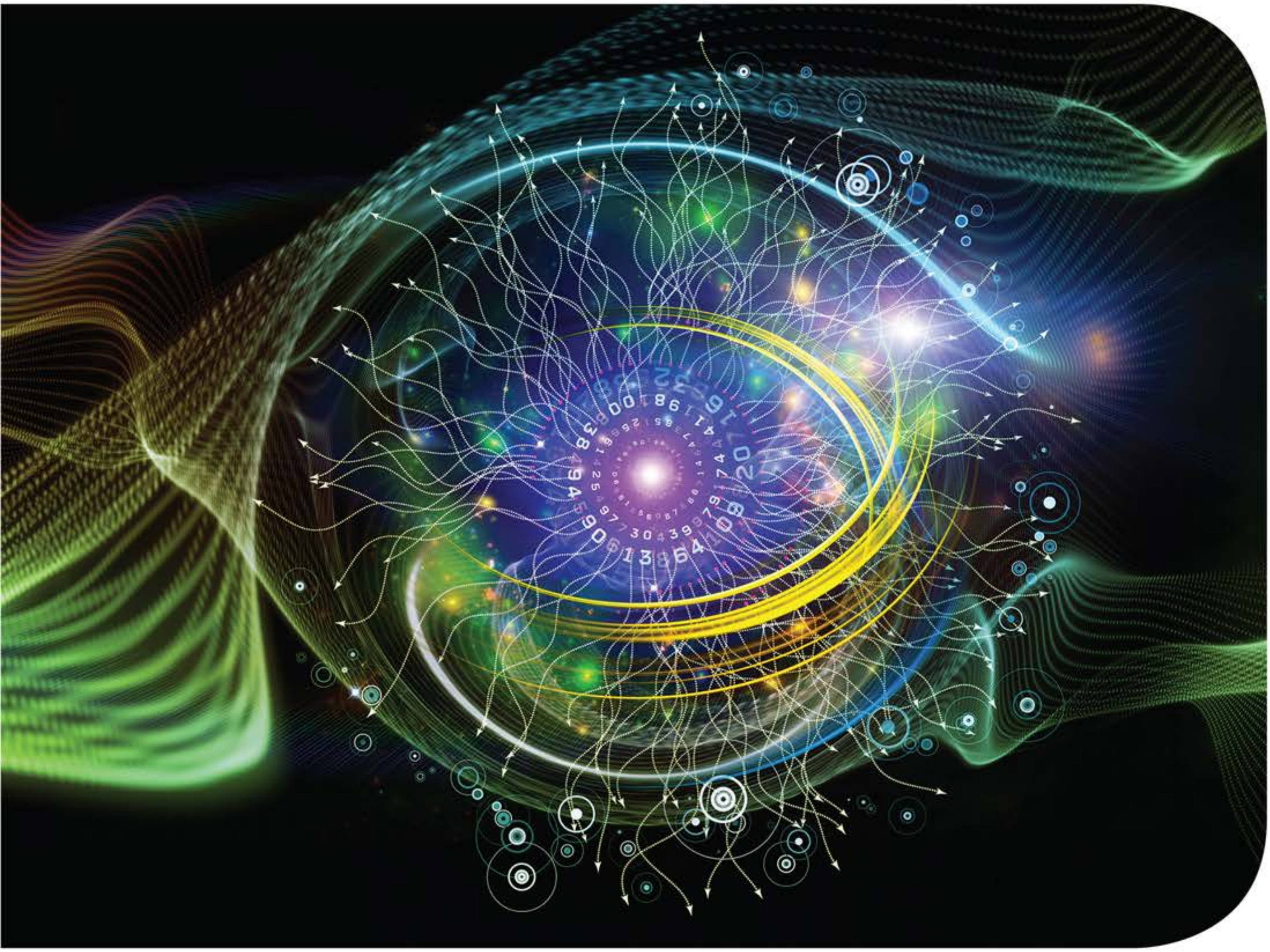
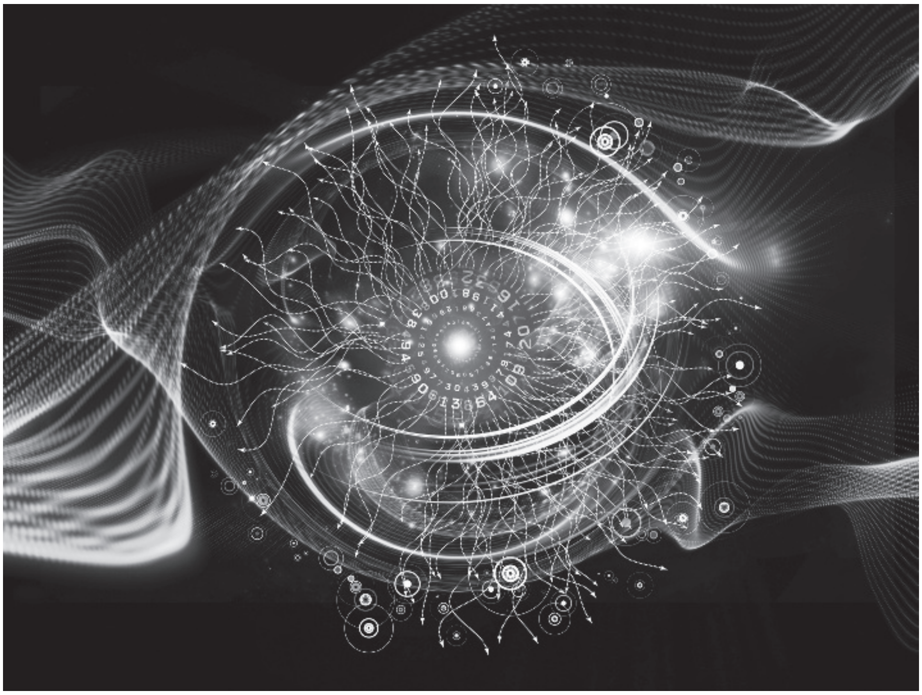
DIGITAL SIGNAL PROCESSING using MATLAB®

THIRD EDITION



ROBERT J. SCHILLING

SANDRA L. HARRIS



Digital Signal Processing

Using MATLAB®

Third Edition

Robert J. Schilling and Sandra L. Harris
Clarkson University
Potsdam, NY

CENGAGE Learning®

Australia • Brazil • Mexico • Singapore • United Kingdom • United States

This is an electronic version of the print textbook. Due to electronic rights restrictions, some third party content may be suppressed. Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. The publisher reserves the right to remove content from this title at any time if subsequent rights restrictions require it. For valuable information on pricing, previous editions, changes to current editions, and alternate formats, please visit www.cengage.com/highered to search by ISBN#, author, title, or keyword for materials in your areas of interest.

Important Notice: Media content referenced within the product description or the product text may not be available in the eBook version.

CENGAGE
Learning®

Digital Signal Processing Using MATLAB®, Third Edition

Robert J. Schilling and Sandra L. Harris

Product Director, Global Engineering: Timothy L. Anderson

Senior Content Developer: Mona Zeftel

Associate Media Content Developer: Ashley Kaupert

Product Assistant: Teresa Versaggi

Marketing Manager: Kristin Stine

Director, Content and Media Production: Sharon L. Smith

Content Project Manager: Nadia Saloom

Production Service: RPK Editorial Services, Inc.

Copyeditor: Harlan James

Proofreader: Martha McMaster

Indexer: Rose Kernan

Compositor: MPS Limited

Senior Art Director: Michelle Kunkler

Internal Designer: Carmela Pereira

Cover Designer: Tin Box Studio

Cover Image: agsandrew/Shutterstock.com

Intellectual Property

Analyst: Christine Myaskovsky

Project Manager: Sarah Shainwald

Text and Image Permissions Researcher: Kristiina Paul

Senior Manufacturing Planner: Doug Wilke

© 2017, 2012 Cengage Learning®

WCN: 02-200-203

ALL RIGHTS RESERVED. No part of this work covered by the copyright herein may be reproduced, transmitted, stored, or used in any form or by any means graphic, electronic, or mechanical, including but not limited to photocopying, recording, scanning, digitizing, taping, Web distribution, information networks, or information storage and retrieval systems, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without the prior written permission of the publisher.

For product information and technology assistance, contact us at **Cengage Learning Customer & Sales Support, 1-800-354-9706**.

For permission to use material from this text or product, submit all requests online at **www.cengage.com/permissions**
Further permissions questions can be emailed to
permissionrequest@cengage.com

Library of Congress Control Number: 2015949697

ISBN: 978-1-305-63519-7

Cengage Learning
20 Channel Center Street
Boston, MA 02210
USA

Cengage Learning is a leading provider of customized learning solutions with employees residing in nearly 40 different countries and sales in more than 125 countries around the world. Find your local representative at **www.cengage.com**

Cengage Learning products are represented in Canada by Nelson Education, Ltd.

To learn more about Cengage Learning Solutions, visit **www.cengage.com/engineering**

Purchase any of our products at your local college store or at our preferred online store **www.cengagebrain.com**

MATLAB is a registered trademark of The MathWorks, 3 Apple Hill Drive, Natick MA 01760.

Unless otherwise noted, all items © Cengage Learning.

Printed in the United States of America
Print Number: 01 Print Year: 2015

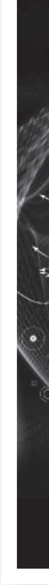
To our mothers

for all they have done for us

Bette Rose Schilling

and

Florence E. Harris



Preface

Digital signal processing, more commonly known as DSP, is a field of study with increasingly widespread applications in our technological world. This book focuses on the development, implementation, and application of modern DSP techniques. The textbook structure consists of three major parts as summarized in Table 1.

Audience and Prerequisites

This book is targeted primarily toward second-semester juniors, seniors, and beginning graduate students in electrical and computer engineering and related fields that rely on digital signal processing. It is assumed that the students have taken a circuits course, or a signals and systems course, or a mathematics course that includes an introduction to both the Laplace transform and the Fourier transform. There is enough material, and sufficient flexibility in the way it can be covered, to provide for courses of different lengths without adding supplementary material. Exposure to MATLAB programming is useful, but it is not

Table 1: Textbook Structure

Part	Name and Chapters
I	Signal and System Analysis
	1. Signal Processing
	2. Discrete-time Systems in the Time Domain
	3. Discrete-time Systems in the Frequency Domain
	4. Fourier Transforms and Signal Spectra
II	Filter Design
	5. Filter Types and Characteristics
	6. FIR Filter Design
	7. IIR Filter Design
III	Advanced Signal Processing
	8. Multirate Signal Processing
	9. Adaptive Signal Processing

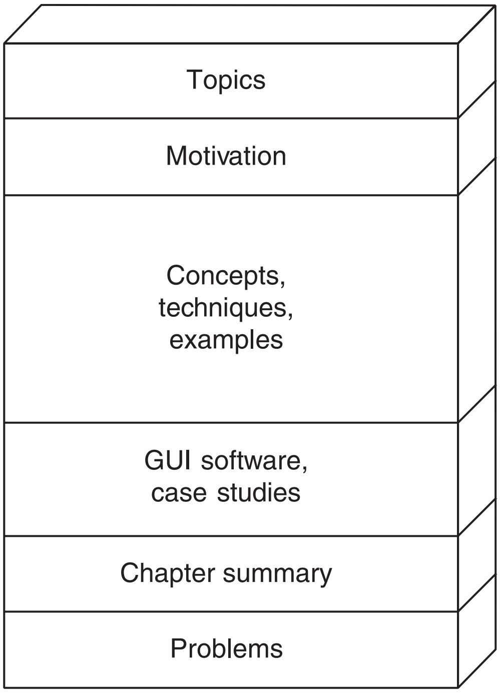
essential. Graphical user interface (GUI) modules are included at the end of each chapter that allow students to interactively explore signal processing concepts and techniques without any need for programming. MATLAB computation problems are supplied for those users who are familiar with MATLAB and are interested in developing their own DSP programs.

This book is written in an engaging informal style that endeavors to provide motivation for each new topic and features a careful transition between topics. Significant terms are set apart for convenient reference using Margin Notes and Definitions. Important results are stated as Propositions in order to highlight their significance, and Algorithms are included to summarize the steps used to implement design procedures. In order to motivate students with examples that are of direct interest, many of the examples feature the processing of speech and music. This theme is also a focus of the DSP Companion course software, which includes a facility for recording and playing back speech and sound. This way, students can experience directly the effects of various signal processing techniques.

Chapter Structure

All of the chapters follow the template shown in Figure 1. Each chapter starts with a brief list of the topics covered. This is followed by a motivation section that introduces one or more examples of practical problems that can be solved using techniques covered in the chapter. The main body of each chapter is used to introduce a series of analysis tools and signal processing techniques. Within these sections, the analysis methods and processing techniques evolve from simple to more complex. Sections near the end of the chapter marked with a ∗ denote more advanced or more specialized material that can be

Figure 1: Chapter Structure



skipped without loss of continuity. Numerous examples are used throughout to illustrate the principles involved.

Near the end of each chapter is a GUI software and Case Studies section that introduces GUI Modules designed to allow the student to interactively explore the chapter concepts and techniques without any need for programming. The GUI Modules feature a common user interface that is simple to use and easy to learn. Results exported from one module can be imported into other modules. This section also includes Case Study examples that present complete solutions to practical problems in the form of MATLAB programs. The Chapter Summary concisely reviews important concepts and includes a table of student learning outcomes for each section. The chapter concludes with an extensive set of homework problems separated into three categories and cross-referenced to the sections. The Analysis and Design problems can be done by hand or with a calculator. They are designed to test (and in some cases extend) student understanding of the chapter material. The GUI Simulation problems allow the student to interactively explore processing and design techniques using the chapter GUI modules. No programming is required for these problems. MATLAB Computation problems are provided that require students to write programs that apply the signal processing techniques covered in the chapter. Complete solutions to selected problems, marked with the ✓ symbol, are available using the DSP Companion software.

DSP Companion Software

One of the unique features of this textbook is a highly integrated collection of course software called the DSP Companion. It is available on the publisher's companion web site, and it features a menu-based graphical user interface driver program called *g_dsp*. The DSP Companion runs under MATLAB and features supplementary course material that can be used both inside the classroom by the instructor and outside the classroom by the student. The DSP Companion provides direct access to the textbook material as well as additional features that allow for class demonstrations and interactive student exploration of analysis and design concepts. The DSP Companion is self-contained in the sense that only MATLAB itself is required; there is no need for access to optional MATLAB toolboxes.

The menu options of the DSP Companion are listed in Table 2. The Settings option allows the user to configure the DSP Companion by selecting operating modes and default folders for exporting, importing, and printing results. The GUI Modules option is used to run the chapter graphical user interface modules. In the Examples option, MATLAB code for all of the examples appearing in the text can be viewed and executed. The Figures and the Tables options are used to display pdf files of all of the figures and tables that appear in the text. Similarly, the Definitions option displays definitions, propositions, and algorithms from the text. The next two menu options are only available with the Instructor version of DSP Companion. The Presentations option displays PowerPoint lectures, with each presentation covering a section of a chapter, while the Solutions option displays solutions to all of the end of chapter problems. For the Student version of DSP Companion, there is a Marked Problems option that display solutions to selected end of chapter problems. The Documentation option provides user help for the DSP Companion functions and the GUI modules. Finally, the Web option allows the user to download the latest version of the DSP Companion from the publisher web site.

Table 2: DSP Companion Menu Options

Option	Description	Type	Links
Settings	Adjust default settings		
GUI Modules	Graphical user interface modules	.m, .mat	11
Examples	View and run MATLAB examples	.m, .mat	120
Figures	View all figures	.pdf	431
Tables	View all tables	.pdf	75
Definitions	View definitions, propositions, algorithms	.pdf	58
Presentations	Display PowerPoint lectures (instructor)	.pptx	91
Solutions	Solutions to all problems (instructor)	.pdf	487
Marked Problems	Solutions to selected problems (student)	.pdf	54
Documentation	Help for DSP Companion functions	.m	124
Web	Software updates	url	6
Exit	Exit DSP Companion		

Acknowledgments

This project has been several years in the making, and many individuals have contributed to its completion. The reviewers commissioned by Brooks/Cole and Cengage Learning made numerous thoughtful and insightful suggestions that were incorporated into the final draft. Thanks to graduate students Joe Tari, Rui Guo, and Lingyun Bai for helping review the initial DSP Companion software. Special thanks to Bill Stenquist of Brooks/Cole who worked closely with us to bring the first edition to completion. We thank the reviewers for this edition for their helpful comments: Selim Awad, University of Michigan-Dearborn; Frederick Harris, San Diego State University; Thomas Hebert, University of Houston; Vishal Monga, Penn State; and Tokunbo Ogunfunmi, Santa Clara University. The latest edition has a number of important improvements.

We wish to acknowledge and thank our Global Engineering team at Cengage Learning for their dedication to this new edition: Timothy Anderson, Product Director; Mona Zeftel, Senior Content Developer; Jana Lewis, Content Project Manager; Kristin Stine, Marketing Manager; Elizabeth Murphy, Engagement Specialist; Ashley Kaupert, Associate Media Content Developer; Teresa Versaggi and Alexander Sham, Product Assistants; and Rose Kernan of RPK Editorial Services, Inc. They have skillfully guided every aspect of this text's development and production to successful completion.

Robert J. Schilling
Sandra L. Harris
Potsdam, NY

Contents

Margin Contents xix

Part I: Signal and System Analysis 1

1. Signal Processing 3

1.1 Motivation 3
1.1.1 Digital and Analog Processing 4
1.1.2 Total Harmonic Distortion (THD) 6
1.1.3 A Notch Filter 7
1.1.4 Active Noise Control 7
1.1.5 Video Aliasing 9

1.2 Signals and Systems 11
1.2.1 Signal Classification 11
1.2.2 System Classification 16

1.3 Sampling of Continuous-time Signals 21
1.3.1 Sampling as Modulation 21
1.3.2 Aliasing 23

1.4 Reconstruction of Continuous-time Signals 27
1.4.1 Reconstruction Formula 27
1.4.2 Zero-order Hold 29
1.4.3 Delayed First-order Hold 32

1.5 Prefilters and Postfilters 35
1.5.1 Anti-aliasing Filter 35
1.5.2 Anti-imaging Filter 39

*1.6 DAC and ADC Circuits 41
1.6.1 Digital-to-analog Converter (DAC) 41
1.6.2 Analog-to-digital Converter (ADC) 43

*Sections marked with a * contain more advanced or specialized material that can be skipped without loss of continuity.

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

1.7 **DSP Companion 48**
1.7.1 Installation 48
1.7.2 Menu Options 48
1.7.3 GUI Modules 50
1.7.4 Functions 50

1.8 **GUI Modules and Case Studies 51**

1.9 **Chapter Summary 59**

1.10 **Problems 61**
1.10.1 Analysis and Design 61
1.10.2 GUI Simulation 66
1.10.3 MATLAB Computation 68

2. **Discrete-time Systems in the Time Domain 70**

2.1 **Motivation 70**
2.1.1 Home Mortgage 71
2.1.2 Range Measurement with Radar 72

2.2 **Discrete-time Signals 74**
2.2.1 Signal Classification 74
2.2.2 Common Signals 79

2.3 **Discrete-time Systems 82**

2.4 **Difference Equations 86**
2.4.1 Zero-input Response 87
2.4.2 Zero-state Response 90

2.5 **Block Diagrams 95**

2.6 **The Impulse Response 97**
2.6.1 FIR Systems 97
2.6.2 IIR Systems 99

2.7 **Convolution 102**
2.7.1 Linear Convolution 102
2.7.2 Circular Convolution 105
2.7.3 Zero Padding 107
2.7.4 Deconvolution 110
2.7.5 Polynomial Arithmetic 111

2.8 **Correlation 113**
2.8.1 Linear Cross-correlation 113
2.8.2 Circular Cross-correlation 116

2.9 **Stability in the Time Domain 119**

2.10 **GUI Modules and Case Studies 121**

2.11 **Chapter Summary 130**

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

2.12 **Problems** 133
- 2.12.1 Analysis and Design 134
- 2.12.2 GUI Simulation 141
- 2.12.3 MATLAB Computation 143

3. **Discrete-time Systems in the Frequency Domain** 147

3.1 **Motivation** 147
- 3.1.1 Satellite Attitude Control 148
- 3.1.2 Modeling the Vocal Tract 149

3.2 **Z-transform Pairs** 151
- 3.2.1 Region of Convergence 152
- 3.2.2 Common Z-transform Pairs 154

3.3 **Z-transform Properties** 159
- 3.3.1 General Properties 159
- 3.3.2 Causal Properties 164

3.4 **Inverse Z-transform** 166
- 3.4.1 Noncausal Signals 166
- 3.4.2 Synthetic Division 167
- 3.4.3 Partial Fractions 169
- 3.4.4 Residue Method 174

3.5 **Transfer Functions** 177
- 3.5.1 The Transfer Function 177
- 3.5.2 Zero-state Response 179
- 3.5.3 Poles, Zeros, and Modes 181
- 3.5.4 DC Gain 183

3.6 **Signal Flow Graphs** 185

3.7 **Stability in the Frequency Domain** 188
- 3.7.1 Input-output Representations 188
- 3.7.2 BIBO Stability 189
- 3.7.3 The Jury Test 192

3.8 **Frequency Response** 195
- 3.8.1 Frequency Response 195
- 3.8.2 Sinusoidal Inputs 197
- 3.8.3 Periodic Inputs 200

*3.9 **System Identification** 203
- 3.9.1 Least-squares Fit 204
- 3.9.2 Persistently Exciting Inputs 207

3.10 **GUI Modules and Case Studies** 209

3.11 **Chapter Summary** 217

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

3.12 **Problems 220**
3.12.1 Analysis and Design 220
3.12.2 GUI Simulation 231
3.12.3 MATLAB Computation 232

4. **Fourier Transforms and Spectral Analysis 234**

4.1 **Motivation 234**
4.1.1 Fourier Series 235
4.1.2 DC Wall Transformer 236
4.1.3 Frequency Response 238

4.2 **Discrete-time Fourier Transform (DTFT) 239**
4.2.1 DTFT 239
4.2.2 Properties of the DTFT 243

4.3 **Discrete Fourier Transform (DFT) 247**
4.3.1 DFT 247
4.3.2 Matrix Formulation 250
4.3.3 Fourier Series and Discrete Spectra 252
4.3.4 DFT Properties 255

4.4 **Fast Fourier Transform (FFT) 262**
4.4.1 Decimation in Time FFT 262
4.4.2 FFT Computational Effort 266
4.4.3 Alternative FFT Implementations 268

4.5 **Fast Convolution and Correlation 269**
4.5.1 Fast Convolution 269
*4.5.2 Fast Block Convolution 273
4.5.3 Fast Correlation 276

4.6 **White Noise 281**
4.6.1 Uniform White Noise 281
4.6.2 Gaussian White Noise 285

4.7 **Auto-correlation 289**
4.7.1 Auto-correlation of White Noise 289
4.7.2 Power Density Spectrum 291
4.7.3 Extracting Periodic Signals from Noise 292

4.8 **Zero Padding and Spectral Resolution 298**
4.8.1 Frequency Response Using the DFT 298
4.8.2 Zero Padding 302
4.8.3 Spectral Resolution 303

4.9 **The Spectrogram 307**
4.9.1 Data Windows 307
4.9.2 Spectrogram 308

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

*4.10 **Power Density Spectrum Estimation 312**
4.10.1 Bartlett's Method 312
4.10.2 Welch's Method 316

4.11 **GUI Modules and Case Studies 320**

4.12 **Chapter Summary 326**

4.13 **Problems 329**
4.13.1 Analysis and Design 329
4.13.2 GUI Simulation 336
4.13.3 MATLAB Computation 338

Part II: Filter Design 343

5. **Filter Types and Characteristics 345**

5.1 **Motivation 345**
5.1.1 Filter Design Specifications 346
5.1.2 Filter Realization Structures 348

5.2 **Frequency-selective Filters 351**
5.2.1 Linear Design Specifications 352
5.2.2 Logarithmic Design Specifications (dB) 357

5.3 **Linear-phase and Zero-phase Filters 359**
5.3.1 Linear Phase 359
5.3.2 Amplitude Response 360
5.3.3 Linear-phase Zeros 363
5.3.4 Zero-phase Filters 365

5.4 **Minimum-phase and Allpass Filters 368**
5.4.1 Minimum-phase Filters 368
5.4.2 Allpass Filters 372
5.4.3 Inverse Systems and Equalization 376

5.5 **Quadrature Filters 377**
5.5.1 Differentiator 377
5.5.2 Hilbert Transformer 378
5.5.3 Digital Oscillator 381

5.6 **Notch Filters and Resonators 384**
5.6.1 Notch Filters 384
5.6.2 Resonators 387

5.7 **Narrowband Filters and Filter Banks 389**
5.7.1 Narrowband Filters 389
5.7.2 Filter Banks 391

5.8 **Adaptive Filters 394**
5.8.1 Transversal Filters 394
5.8.2 Pseudo-filters 396

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

5.9 GUI Modules and Case Studies 397

5.10 Chapter Summary 402

5.11 Problems 405
5.11.1 Analysis and Design 405
5.11.2 GUI Simulation 413
5.11.3 MATLAB Computation 415

6. FIR Filter Design 417

6.1 Motivation 417
6.1.1 Numerical Differentiators 418
6.1.2 Signal-to-noise Ratio 420

6.2 Windowing Method 422
6.2.1 Truncated Impulse Response 423
6.2.2 Windowing 426

6.3 Frequency-sampling Method 434
6.3.1 Frequency Sampling 434
6.3.2 Transition-band Optimization 437

6.4 Least-squares Method 441

6.5 Optimal Equiripple Filter Design 445
6.5.1 Minimax Error Criterion 445
6.5.2 Parks-McClellan Algorithm 447

6.6 Differentiators and Hilbert Transformers 454
6.6.1 Differentiator Design 454
6.6.2 Hilbert Transformer Design 457

6.7 Quadrature Filter Design 460
6.7.1 Generation of a Quadrature Pair 460
6.7.2 Quadrature Filter Design 462
6.7.3 Equalizer Design 466

6.8 Filter Realization Structures 470
6.8.1 Direct Forms 470
6.8.2 Cascade Form 472
6.8.3 Lattice Form 474

*6.9 Finite Word Length Effects 477
6.9.1 Binary Number Representation 478
6.9.2 Input Quantization Error 479
6.9.3 Coefficient Quantization Error 482
6.9.4 Roundoff Error, Overflow, and Scaling 486

6.10 GUI Modules and Case Studies 489

6.11 Chapter Summary 495

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

6.12 **Problems 499**
6.12.1 Analysis and Design 499
6.12.2 GUI Simulation 504
6.12.3 MATLAB Computation 506

7. **IIR Filter Design 511**

7.1 **Motivation 511**
7.1.1 Tunable Plucked-string Filter 512
7.1.2 Colored Noise 514

7.2 **Filter Design by Pole-zero Placement 516**
7.2.1 Resonator 516
7.2.2 Notch Filter 519
7.2.3 Comb Filters 522

7.3 **Filter Design Parameters 526**

7.4 **Classical Analog Filters 528**
7.4.1 Butterworth Filters 529
7.4.2 Chebyshev-I Filters 533
7.4.3 Chebyshev-II Filters 536
7.4.4 Elliptic Filters 538

7.5 **Bilinear Transformation Method 540**

7.6 **Frequency Transformations 547**
7.6.1 Analog Frequency Transformations 547
7.6.2 Digital Frequency Transformations 550

7.7 **Filter Realization Structures 553**
7.7.1 Direct Forms 553
7.7.2 Parallel Form 556
7.7.3 Cascade Form 559

*7.8 **Finite Word Length Effects 562**
7.8.1 Coefficient Quantization Error 563
7.8.2 Roundoff Error, Overflow, and Scaling 566
7.8.3 Limit Cycles 570

7.9 **GUI Modules and Case Studies 572**

7.10 **Chapter Summary 579**

7.11 **Problems 582**
7.11.1 Analysis and Design 582
7.11.2 GUI Simulation 587
7.11.3 MATLAB Computation 590

Part III: Advanced Signal Processing 593

8. **Multirate Signal Processing 595**

8.1 **Motivation 595**

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

8.1.1 Narrowband Filters 596
8.1.2 Intersample Delay Systems 598

8.2 Integer Sampling Rate Converters 599
8.2.1 Sampling Rate Decimator 599
8.2.2 Sampling Rate Interpolator 600

8.3 Rational Sampling Rate Converters 603
8.3.1 Single-stage Converters 603
8.3.2 Multistage Converters 606

8.4 Polyphase Filters 608
8.4.1 Polyphase Decimator 608
8.4.2 Polyphase Interpolator 612

8.5 Narrowband Filters 614

8.6 Filter Banks 616
8.6.1 Analysis and Synthesis Banks 616
8.6.2 Subfilter Design 620
8.6.3 Polyphase Representation 621

8.7 Perfect Reconstruction Filter Banks 624
8.7.1 Time-division Multiplexing 624
8.7.2 Perfect Reconstruction 626

8.8 Transmultiplexors 630

***8.9 Oversampled A-to-D Converters 634**
8.9.1 Anti-aliasing Filters 634
8.9.2 A Sigma-delta ADC 638

***8.10 Oversampled D-to-A Converters 642**
8.10.1 Anti-imaging Filters 642
8.10.2 Passband Equalization 644

8.11 GUI Modules and Case Studies 646

8.12 Chapter Summary 652

8.13 Problems 655
8.13.1 Analysis and Design 655
8.13.2 GUI Simulation 663
8.13.3 MATLAB Computation 664

9. Adaptive Signal Processing 667

9.1 Motivation 667
9.1.1 System Identification 668
9.1.2 Channel Equalization 669
9.1.3 Signal Prediction 670
9.1.4 Noise Cancellation 670

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

9.2 Mean Square Error 671

9.2.1 Adaptive Transversal Filters 671
9.2.2 Cross-correlation Revisited 672
9.2.3 Mean Square Error 673

9.3 Least Mean Square (LMS) Method 678

9.4 Performance Analysis of LMS Method 682

9.4.1 Step Size 682
9.4.2 Convergence Rate 686
9.4.3 Excess Mean Square Error 689

9.5 Modified LMS Methods 692

9.5.1 Normalized LMS Method 692
9.5.2 Correlation LMS Method 694
9.5.3 Leaky LMS Method 698

9.6 Adaptive Filter Design with Pseudo-filters 701

9.6.1 Pseudo-filters 701
9.6.2 Adaptive Filter Design 703
9.6.3 Linear-phase Adaptive Filters 705

9.7 Recursive Least Squares (RLS) Method 708

9.7.1 Performance Criterion 708
9.7.2 Recursive Formulation 709

***9.8 Active Noise Control 714**

9.8.1 The Filtered-*x* LMS Method 714
9.8.2 Secondary-path Identification 716
9.8.3 Signal-synthesis Method 719

***9.9 Adaptive Function Approximation 723**

9.9.1 Nonlinear Functions 724
9.9.2 Radial Basis Functions (RBFs) 726
9.9.3 Raised-cosine RBF Networks 728

***9.10 Nonlinear System Identification (NLMS) 734**

9.11 GUI Modules and Case Studies 740

9.12 Chapter Summary 744

9.13 Problems 748

9.13.1 Analysis and Design 748
9.13.2 GUI Simulation 753
9.13.3 MATLAB Computation 754

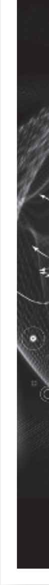
References and Further Reading 761

Appendix 1 Transform Tables 765

Appendix 2 Mathematical Identities 774

Index 777

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.



Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

Margin Contents

Table I: Definitions

Number	Term	Symbol	Page
1.1	Causal signal	$x_a(t)$	14
1.2	Linear system	S	16
1.3	Time-invariant system	S	17
1.4	Stable system	S	18
1.5	Frequency response	$H_a(f)$	19
1.6	Impulse response	$h_a(t)$	19
1.7	Bandlimited signal	$x_a(t)$	24
1.8	Transfer function	$H_a(S)$	29
2.1	Impulse response	$h(k)$	97
2.2	FIR and IIR systems	S	98
2.3	Linear convolution	$h(k) * x(k)$	103
2.4	Circular convolution	$h(k) \circ x(k)$	106
2.5	Linear cross-correlation	$r_{yx}(k)$	113
2.6	Circular cross-correlation	$c_{yz}(k)$	116
2.7	BIBO stability	$\|h\|_1 < \infty$	120
3.1	Z-transform	$X(z)$	151
3.2	Transfer function	$H(z)$	178
3.3	Frequency response	$H(f)$	195
4.1	Discrete-time Fourier transform (DTFT)	$X(f)$	239
4.2	Discrete Fourier transform (DFT)	$X(i)$	249
4.3	Expected value	$E[fx(k)]$	282
4.4	Circular auto-correlation	$c_{xx}(k)$	289
4.5	Spectrogram	$G(m, i)$	308
5.1	Group delay	$D(f)$	359
5.2	Linear-phase filter	$H(z)$	360
5.3	Minimum-phase filter	$H(z)$	369
5.4	Allpass Filter	$H(z)$	372
6.1	Signal-to-noise ratio	$\mathrm{SNR}(y)$	420
6.2	Quantization operator	$Q_N(x)$	479
9.1	Random cross-correlation	$r_{yx}(i)$	672

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

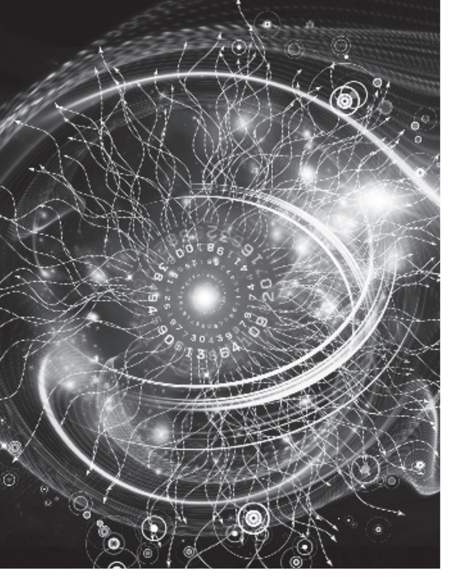
Table II: Propositions

Number	Description	Page
1.1	Signal Sampling	25
1.2	Signal Reconstruction	28
2.1	Stability in the Time Domain	120
3.1	Stability in the Frequency Domain	190
3.2	Frequency Response	198
4.1	Parseval's Identity: DTFT	244
4.2	Parseval's Identity: DFT	260
5.1	Paley-Wiener Theorem	352
5.2	Linear-phase Filter	362
5.3	Minimum-phase Allpass Decomposition	373
6.1	Alternation Theorem	447
6.2	Flow Graph Reversal Theorem	471
8.1	Perfect Reconstruction Filter Bank	627
8.2	Perfect Reconstruction Frequency Domain	628
9.1	Least Mean Square (LMS) Convergence	684
9.2	Raised-cosine RBF Network	731

Table III: Algorithms

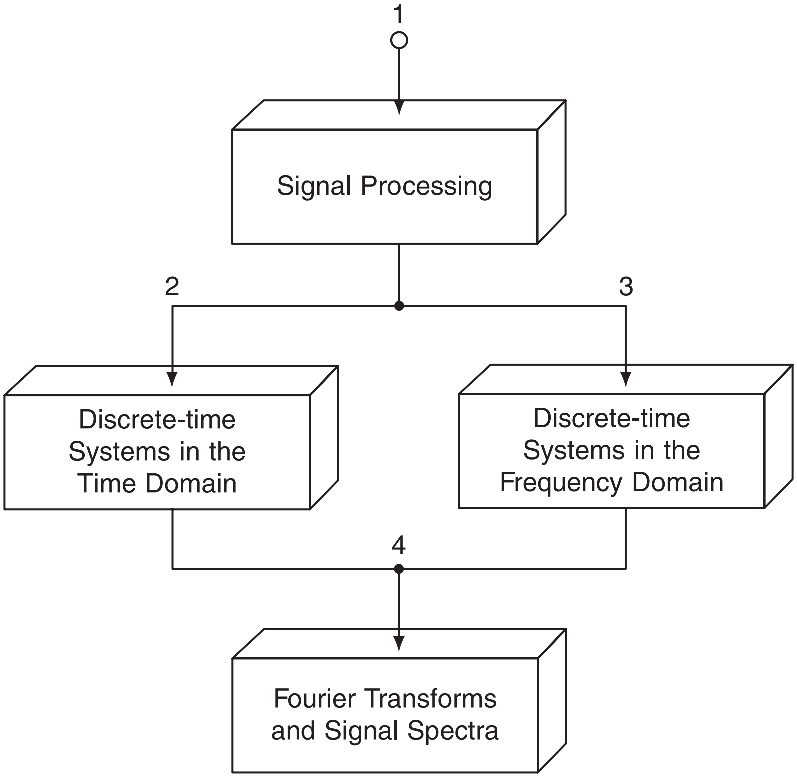
Number	Description	Page
1.1	Successive Approximation	45
3.1	Residue Method	175
4.1	Bit Reversal	265
4.2	FFT	266
4.3	Problem Domain	267
4.4	IFFT	268
4.5	Fast Block Convolution	274
5.1	Zero-phase Filter	366
5.2	Minimum-phase Allpass Decomposition	373
6.1	Windowed FIR Filter	431
6.2	Ripple Size	449
6.3	Equiripple FIR Filter	451
6.4	Lattice-form Realization	475
7.1	Bilinear Transformation Method	544
9.1	RLS Method	711
9.2	Raised-cosine RBF Network Evaluation	731

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

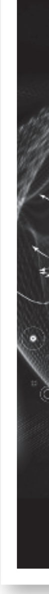


PART 1

Signal and System Analysis


1
Signal Processing
2
3
Discrete-time Systems in the Time Domain
Discrete-time Systems in the Frequency Domain
4
Fourier Transforms and Signal Spectra

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.



Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

CHAPTER 1

Signal Processing

CHAPTER TOPICS

1.1 Motivation
1.2 Signals and Systems
1.3 Sampling of Continuous-time Signals
1.4 Reconstruction of Continuous-time Signals
1.5 Prefilters and Postfilters
*1.6 DAC and ADC Circuits
1.7 DSP Companion
1.8 GUI Modules and Case Studies
1.9 Chapter Summary
1.10 Problems

1.1 Motivation

A signal is a physical variable whose value varies with time or space. When the value of the signal is available over a continuum of times, it is referred to as a *continuous-time signal*. Continuous-time signals whose amplitudes also vary over a continuous range are called *analog signals*. Everyday examples of analog signals include temperature, pressure, liquid level, chemical concentration, voltage and current, position, velocity, acceleration, force and torque. If the value of the signal is available only at discrete instants of time, it is called a *discrete-time signal*. Although some signals, for example economic data, are inherently discrete-time signals, a more common way to produce a discrete-time signal, $x(k)$, is to take samples of an underlying analog signal, $x_a(t)$.

Continuous-time signal

Analog signal

Discrete-time signal

Sampling interval

Digital signal

$$x(k) \overset{\Delta}{=} x_a(kT), \quad |k| = 0, 1, 2, \cdots$$

Here T denotes the *sampling interval* or time between samples, and $\overset{\Delta}{=}$ means equals by definition. When finite precision is used to represent the value of $x(k)$, the sequence of quantized values is then called a *digital signal*. A system or algorithm which processes one digital signal $x(k)$ as its input and produces a second digital signal $y(k)$ as its output is a digital signal processor. Digital signal processing (DSP) techniques have widespread applications, and they play an increasingly important role in the modern world. Application areas include speech recognition, detection of targets with radar and sonar, processing of music and video, seismic exploration for oil and gas deposits, medical

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

signal processing including EEG, EKG, and ultrasound, communication channel equalization, and satellite image processing. The focus of this book is the development, implementation, and application of modern DSP techniques.

We begin this introductory chapter with a comparison of digital and analog signal processing. Next, some practical problems are posed that can be solved using DSP techniques. This is followed by characterization and classification of signals. The fundamental notion of the spectrum of a signal is then presented including the concepts of bandlimited and white noise signals. This leads naturally to the sampling process which takes a continuous-time signal and produces a corresponding discrete-time signal. Simple conditions are presented that ensure that an analog signal can be reconstructed from its samples. When these conditions are violated, a phenomenon called aliasing occurs. The use of guard filters to reduce the effects of aliasing is discussed. Next DSP hardware in the form of analog-to-digital converters (ADCs) and digital-to-analog converters (DACs) is examined. The hardware discussion includes ways to model the quantization error associated with finite precision converters. A menu-based graphical user interface (GUI) program called the *DSP Companion* is introduced that provides direct access to an extensive set of supplementary course materials. The DSP Companion allows the student and the instructor to run and view chapter GUI modules, PowerPoint lecture slides, examples, figures, tables, definitions, propositions, algorithms, and selected problem solutions that appear throughout the text. The *GUI modules* can be used to interactively explore the digital signal processing techniques covered in each chapter without any need for programming. For example, in this chapter GUI module *g_sample* allows the user to investigate the sampling of continuous-time signals including aliasing and quantization effects. The module *g_reconstruct* then allows the user to explore the reconstruction of continuous-time signals from their samples. The chapter concludes with a case study example, and a summary of signal sampling and reconstruction.

DSP Companion

GUI modules

1.1.1 Digital and Analog Processing

For many years, almost all signal processing was done with analog circuits as shown in Figure 1.1. For example, operational amplifiers, resistors, and capacitors are used to realize frequency-selective filters.

With the advent of specialized microprocessors with built-in data conversion circuits (Papamichalis, 1990), it is now commonplace to perform signal processing digitally as shown in Figure 1.2. Digital processing of analog signals is more complex because it typically requires the three components shown in Figure 1.2. The analog-to-digital converter or ADC at the front end converts the analog input $x_a(t)$ into an equivalent digital signal

Figure 1.1: Analog Signal Processing

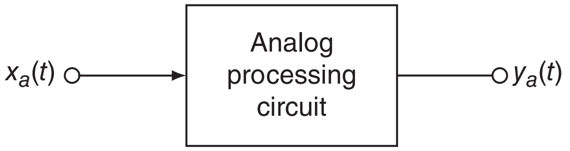
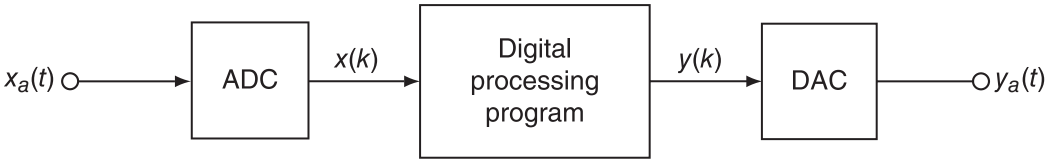


Figure 1.2: Digital Signal Processing



Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

Table 1.1: Comparison of Analog and Digital Signal Processing

Feature	Analog Processing	Digital Processing
Speed	Fast	Moderate
Cost	Low to moderate	Moderate
Flexibility	Low	High
Performance	Moderate	High
Self-calibration	No	Yes
Data logging capability	No	Yes
Adaptive capability	Limited	Yes

$x(k)$. The processing of $x(k)$ is then achieved with an algorithm that is implemented in software. For a filtering operation, the DSP algorithm consists of a difference equation, but other types of processing are also possible and are often used. The digital output signal $y(k)$ is then converted back to an equivalent analog signal $y_a(t)$ by the digital-to-analog converter or DAC.

Although the DSP approach requires more steps than analog signal processing, there are many important benefits to working with signals in digital form. A comparison of the relative advantages and disadvantages of the two approaches is summarized in Table 1.1.

Real time

Latency

The primary advantages of analog signal processing are speed and cost. Digital signal processing is not as fast due to the limits on the sampling rates of the converter circuits. In addition, if substantial computations are to be performed between samples, then the clock rate of the processor also can be a limiting factor. Speed can be an issue in *real-time* applications where the kth output sample $y(k)$ must be computed and sent to the DAC as soon as possible after the kth input sample $x(k)$ is available from the ADC. The delay is sometimes referred to as *latency*. However, there are also applications where the entire input signal is available ahead of time for processing off-line. For this batch mode type of processing, speed is less critical.

DSP hardware is often somewhat more expensive than analog hardware because analog hardware can consist of as little as a few discrete components on a stand-alone printed circuit board. The cost of DSP hardware varies depending on the performance characteristics required. In some cases, a PC may already be available to perform other functions for a given application, and in these instances the marginal expense of adding DSP hardware is not large.

In spite of these limitations, there are great benefits to using DSP techniques. Indeed, DSP is superior to analog processing with respect to virtually all of the remaining features listed in Table 1.1. One of the most important advantages is the inherent flexibility available with a software implementation. Whereas an analog circuit might be tuned with a potentiometer to vary its performance over a limited range, a DSP algorithm can be completely replaced, on the fly, when circumstance warrant. DSP also offers considerably higher performance than analog signal processing. For example, digital filters with arbitrary magnitude responses and linear phase responses can be designed easily whereas this is not feasible with analog filters.

A common problem that plagues analog systems is the fact that the component values tend to drift with age and with changes in environmental conditions such as temperature. This leads to a need for periodic calibration or tuning. With DSP there is no drift problem and therefore no need to manually recalibrate.

Since data are already available in digital form in a DSP system, with little or no additional expense one can log the data associated with the operation of the system so that its performance can be monitored, either locally or remotely over a network connection. If an unusual operating condition is detected, its exact time and nature can be determined,

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

and a higher-level control system can be alerted. Although strip chart recorders can be added to an analog system, this substantially increases the expense thereby negating one of its potential advantages.

The flexibility inherent in software can be exploited by having the parameters of the DSP algorithm vary with time and adapt as the characteristics of the input signal or the processing task change. Applications, like system identification and active noise control, exploit adaptive signal processing, a topic that is addressed in Chapter 9.

1.1.2 Total Harmonic Distortion (THD)

With the widespread use of digital computers, DSP applications are now commonplace. As a simple initial example, consider the problem of designing an audio amplifier to boost signal strength without distorting the shape of the input signal. For the amplifier shown in Figure 1.3, suppose the input signal $x_a(t)$ is a pure sinusoidal tone of amplitude a and frequency F_0 Hz.

$$x_a(t) = a\cos(2\pi F_0 t) \tag{1.1.1}$$

Gain

An ideal amplifier will produce a desired output signal $y_d(t)$ that is a scaled and delayed version of the input signal. For example, if the scale factor or amplifier *gain* is K and the delay is τ, then the desired output is

$$\begin{aligned} y_d(t) &= Kx_a(t-\tau) \\ &= Ka\cos\left[2\pi F_0(t-\tau)\right] \end{aligned} \tag{1.1.2}$$

In a practical amplifier, the relationship between the input and the output is only approximately linear, so some additional terms are present in the actual output y_a.

$$\begin{aligned} y_a(t) &= A[x_a(t)] \\ &\approx \frac{d_0}{2} + \sum_{i=1}^{M-1} d_i \cos(2\pi i F_0 t + \theta_i) \end{aligned} \tag{1.1.3}$$

Average power

The presence of the additional harmonics indicates that there is distortion in the amplified signal due to nonlinearities within the amplifier. For example, if the amplifier is driven with an input whose amplitude a is too large, then the amplifier will saturate with the result that the output is a clipped sine wave that sounds distorted when played through a speaker. To quantify the amount of distortion, the average power contained in the ith harmonic is $d_i^2/2$ for $i \geq 1$ and $d_i^2/4$ for $i = 0$. Thus the *average power* of the signal $y_a(t)$ is

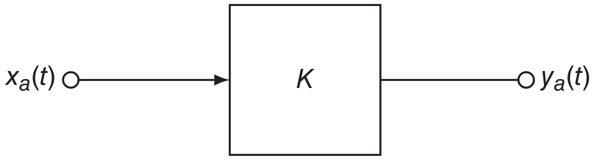
$$P_y = \frac{d_0^2}{4} + \frac{1}{2}\sum_{i=1}^{M-1} d_i^2 \tag{1.1.4}$$

Total harmonic distortion

The *total harmonic distortion* or THD of the output signal $y_a(t)$ is defined as the power in the spurious harmonic components, expressed as a percentage of the total power. Thus the following can be used to measure the quality of the amplifier output.

$$\text{THD} \triangleq \frac{100(P_y - d_1^2/2)}{P_y}\ \% \tag{1.1.5}$$

Figure 1.3: An Audio Amplifier



Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

For an ideal amplifier $d_i = 0$ for $i \neq 1$ and

$$d_1 = Ka \tag{1.1.6a}$$

$$\theta_1 = -2\pi F_0 \tau \tag{1.1.6b}$$

Consequently, for a high-quality amplifier, the THD is small, and when no distortion is present THD = 0. Suppose the amplifier output is sampled to produce the following digital signal of length $N = 2M$.

$$y(k) = y_a(kT), \quad 0 \leq k < N \tag{1.1.7}$$

If the sampling interval is set to $T = 1/(NF_0)$, then this corresponds to one period of $y_a(t)$. By processing the digital signal $x(k)$ with something called the discrete Fourier transform or DFT, it is possible to determine d_i and θ_i for $0 \leq i < M$. In this way the total harmonic distortion can be measured. The DFT is a key analytic tool that is introduced in Chapter 4.

1.1.3 A Notch Filter

As a second example of a DSP application, suppose one is performing sensitive acoustic measurements in a laboratory setting using a microphone. Here, any ambient background sounds in the range of frequencies of interest have the potential to corrupt the measurements with unwanted noise. Preliminary measurements reveal that the overhead fluorescent lights are emitting a 120 Hz hum, which corresponds to the second harmonic of the 60 Hz commercial AC power. The problem then is to remove the 120 Hz frequency component while affecting the other nearby frequency components as little as possible. Consequently, you want to process the acoustic data samples with a *notch filter* designed to remove the effects of the fluorescent lights. After some calculations, you arrive at the following digital filter to process the measurements $x(k)$ to produce a filtered signal $y(k)$.

Notch filter

$$\begin{aligned} y(k) = 1.6466y(k-1) - .9805y(k-2) + .9905x(k) \\ - 1.6471x(k-1) + .9905x(k-2) \end{aligned} \tag{1.1.8}$$

The filter in (1.1.8) is a notch filter with a bandwidth of 4 Hz, a notch frequency of $F_n = 120$ Hz, and a sampling frequency of $f_s = 1280$ Hz. A plot of the frequency response of this filter is shown in Figure 1.4 where a sharp notch at 120 Hz is apparent. Notice that except for frequencies very close to F_n, all other frequency components of $x(k)$ are passed through the filter without attenuation. The design of notch filters is discussed in Chapter 7.

1.1.4 Active Noise Control

An application area of DSP that makes use of adaptive signal processing is active control of acoustic noise (Kuo and Morgan, 1996). Examples include industrial noise from rotating machines, propeller and jet engine noise, road noise in an automobile, and noise caused by air flow in heating, ventilation, and air conditioning systems. As an illustration of the latter, consider the active noise control system shown in Figure 1.5, which consists of an air duct with two microphones and a speaker. The basic principle of active noise control is to inject a secondary sound into the environment so as to cancel the primary sound using destructive interference.

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

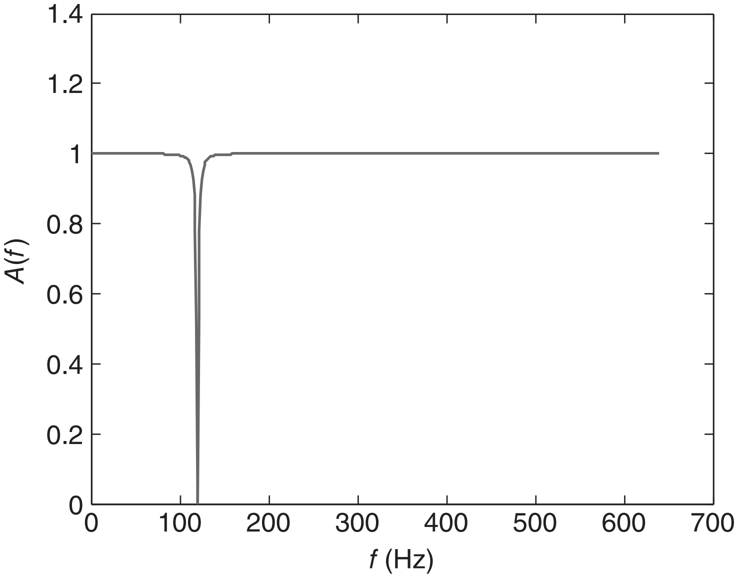


Figure 1.4: Magnitude Response of a Notch Filter with $F_n = 120$ Hz

The purpose of the reference microphone in Figure 1.5 is to detect the primary noise $x(k)$ generated by the noise source or blower. The primary noise signal is then passed through a digital filter of the following form.

$$y(k) = \sum_{i=0}^{m} w_i(k)x(k-i) \tag{1.1.9}$$

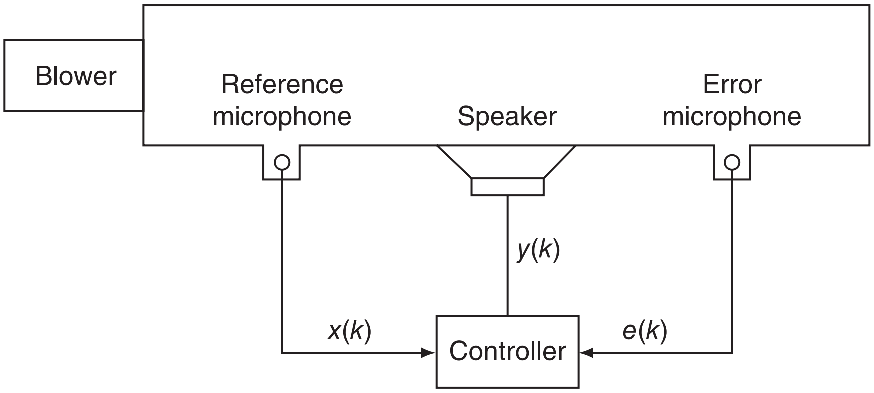
Antisound

The output of the filter $y(k)$ drives a speaker that creates the secondary sound, sometimes called *antisound.* The error microphone, located downstream of the speaker, detects the sum of the primary and secondary sounds and produces an error signal $e(k)$. The objective of the adaptive algorithm is to take $x(k)$ and $e(k)$ as inputs and adjust the filter weights $w(k)$ so as to drive $e^2(k)$ to zero. If zero error can be achieved, then silence is observed at the error microphone. In practical systems, the error or residual sound is significantly reduced by active noise control.

To illustrate the operation of this adaptive DSP system, suppose the blower noise is modeled as a periodic signal with fundamental frequency F_0 and r harmonics plus some random white noise $v(k)$.

$$x(k) = \sum_{i=1}^{r} a_i \cos(2\pi ikF_0T + \theta_i) + v(k), \quad 0 \le k < p \tag{1.1.10}$$

Figure 1.5: Active Control of Acoustic Noise in an Air Duct



Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s). Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.

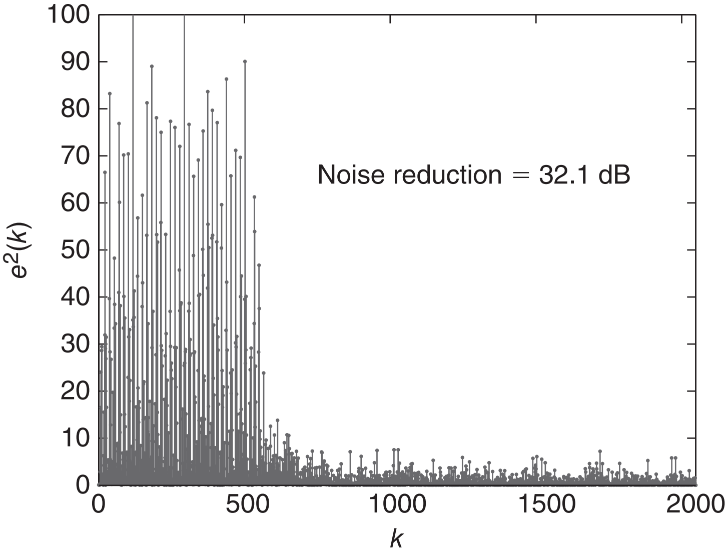


Figure 1.6: Squared Error Signal with Active Noise Control Activated at $k = 512$

For example, suppose $F_0 = 100$ Hz and there are $r = 4$ harmonics with amplitudes $a_i = 1/i$ and random phase angles. Suppose the random white noise term is distributed uniformly over the interval $[-.5, .5]$. Let $p = 2048$ samples, and suppose the sampling interval is $T = 1/1600$ s and the filter order is $m = 40$. The adaptive algorithm used to adjust the filter weights is called the FXLMS method, and it is discussed in detail in Chapter 9. The results of applying this algorithm are shown in Figure 1.6.

Initially the filter weights are set to $w(0) = 0$, which corresponds to no noise control at all. The adaptive algorithm is not activated until sample $k = 512$, so the first quarter of the plot in Figure 1.6 represents the ambient or primary noise detected at the error microphone. When adaptation is activated, the error begins to decrease rapidly, and after a short transient period it reaches a steady-state level that is more than an order of magnitude quieter than the primary noise itself. We can quantify the noise reduction by using the following measure of overall noise cancellation.

$$E = 10 \log_{10}\left(\sum_{i=0}^{p/4-1} e^2(i)\right) - 10 \log_{10}\left(\sum_{i=3p/4}^{p-1} e^2(i)\right) \quad \text{dB} \tag{1.1.11}$$

The overall noise cancellation E is the log of the ratio of the average power of the noise during the first quarter of the samples divided by the average power of the noise during the last quarter of the samples, expressed in units of decibels. Using this measure, the noise cancellation observed in Figure 1.6 is $E = 32.1$ dB.

1.1.5 Video Aliasing

Later in Chapter 1 we focus on the problem of sampling a continuous-time signal $x_a(t)$ to produce the following discrete-time signal, where $T > 0$ is the sampling interval and $f_s = 1/T$ is the sampling frequency.

$$x(k) = x_a(kT), \; |k| = 0, 1, 2, \ldots \tag{1.1.12}$$

Copyright 2017 Cengage Learning. All Rights Reserved. May not be copied, scanned, or duplicated, in whole or in part. Due to electronic rights, some third party content may be suppressed from the eBook and/or eChapter(s).
Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. Cengage Learning reserves the right to remove additional content at any time if subsequent rights restrictions require it.