
**DESIGN AND ANALYSIS OF
ANALOG FILTERS**
A Signal Processing Perspective

**THE KLUWER INTERNATIONAL SERIES
IN ENGINEERING AND COMPUTER SCIENCE**

Design and Analysis of
ANALOG FILTERS:
A Signal Processing Perspective

Larry D. Paarmann
Associate Professor of Electrical Engineering
Wichita State University
Wichita, Kansas

KLUWER ACADEMIC PUBLISHERS
NEW YORK, BOSTON, DORDRECHT, LONDON, MOSCOW

eBook ISBN: 0-306-48012-3
Print ISBN: 0-7923-7373-1

©2003 Kluwer Academic Publishers
New York, Boston, Dordrecht, London, Moscow

Print ©2001 Kluwer Academic Publishers
Dordrecht

All rights reserved

No part of this eBook may be reproduced or transmitted in any form or by any means, electronic, mechanical, recording, or otherwise, without written consent from the Publisher

Created in the United States of America

Visit Kluwer Online at: <http://kluweronline.com>
and Kluwer's eBookstore at: <http://ebooks.kluweronline.com>

PREFACE

Analog filters, that is continuous-time filters, or filters that can be implemented with resistors, capacitors, inductors, specialized elements or devices, etc., have enjoyed a long history of use in electrical engineering applications. In fact, it can be said without fear of contradiction that the modern technological world, as we know it, would not exist without analog filters. Even though digital filters, and digital signal processing in general, has experienced great growth and development in recent years, analog filters are an important topic. At the university where the author is an associate professor, a course in analog filters, taught at the first-year graduate / senior level, is one course in the graduate program of signal processing. Many of the concepts in analog filter theory help establish a foundation of understanding that assists in more advanced courses on digital filters, modern filters, adaptive filters, spectral estimation, etc. And, of course, analog filter concepts, and the ability to design and analyze them, is important in the additional area of analog circuit design, mixed signal circuit¹ design, and in integrated circuit development.

Therefore, this textbook presents analog filter theory from a signal processing perspective (i.e., stressing the signals and systems concepts), but also including analog circuit design and analysis as well. Concepts such as the relationships among the time domain, frequency domain, and s domain are stressed. Other things stressed are inherent trade-offs dictated by theory, that have nothing to do with implementation. For example, attempting to eliminate the time-domain ringing of a high-order Butterworth bandpass filter is an exercise in futility, as theory clearly reveals. Chebyshev and elliptic filters will ring even more: s domain analysis, and the equivalent time-domain analysis clearly reveal this, whereas frequency domain analysis alone may not suggest it. As an educator, and one who concentrates in the area of signal processing, the author believes these concepts to be of vital importance. Almost any book on analog filters will include signal processing / systems concepts

¹ Mixed signal circuits are those that include analog signals, and analog circuitry to process and amplify them, and also digital signals and associated digital circuitry. An example is an integrated circuit with an analog input signal and an analog output signal, but with some digital signal processing in between, which requires an A/D with some analog signal conditioning on the input, perhaps including an anti-aliasing filter, and a D/A with some signal conditioning on the output, perhaps including a reconstruction filter.

as well as implementation, and the present book is no exception. Most books on analog filter design briefly present the signal processing / systems concepts, and then concentrate on a variety of filter implementation methods. The present book reverses the emphasis, stressing signal processing concepts. The present book does not ignore implementation, as it does present filter implementation topics in **Part II**: passive filters, and operational amplifier active filters. However, greater emphasis on signal processing / systems concepts are included in **Part I** of the book than is typical. As suggested above, this emphasis makes the book more appropriate as part of a signal processing curriculum, but should also be of interest to those in analog circuit design as well.

The intended audience for this book includes anyone with a standard electrical engineering background, with a B.S. degree or beyond, or at the senior level. The most important background subjects are Laplace and Fourier transform theory, and concepts in basic systems and signals, and, of course, basic circuits as well. A background in communications systems would be helpful in fully appreciating the application examples given in **Chapter 1**, but these examples are given to illustrate analog filter applications, and a communications background is not a prerequisite for understanding the developments in the book. While MATLAB² and SPICE³ are software packages used, and familiarity with them would be an asset, it is assumed that the adept student can learn these software tools on his own, or with minimal guidance, if they are not already known. A brief introduction to MATLAB is given in **Appendix A**.

Analog electrical signals are so named because they are analogous to some other signal from which they are derived: acoustic, electromagnetic, mechanical motion, etc. Analog filters process analog signals. However, they are also analogous in another respect. The physically-constructed filter, i.e. the realized filter, responds in the time-domain and frequency-domain in a manner that is analogous to the theoretical filter, as defined by, say, as is often done, the magnitude-squared frequency response. This suggests an important concept. A particular filter response, as perhaps defined by the magnitude-squared frequency response, such as a Butterworth response, is mathematically defined. The realization is, at best, an approximation. Therefore, the “filter” is defined mathematically and abstractly. All realizations are approximations. A “circle” often refers to a geometrical drawing representing a circle, as a “filter” often refers to a physical realization of a filter. Hence, in this textbook, theory is stressed and presented first; implementation (a schematic drawing) follows, and, in practice, a realization (physical circuit) would often follow that. It is a fascinating confirmation of the value of theory, that trial and error, and experimentation, would never come up with a Butterworth filter design, but theory elegantly and clearly develops it, and states the result in a very simple form.

²MATLAB is a registered trademark of The Math Works, Inc., and is a high-level language for technical computing.

³SPICE is an abbreviation for Simulation Program with Integrated Circuit Emphasis, and is a powerful circuit simulation computing program. Many commercially available circuit simulation programs are based on SPICE.

The term “approximation” is used in two ways in this book. In **Part I** it refers to a filter design $H(s)$ that only approximates an ideal filter. As pointed out in **Chapter 2**, the term “ideal filter” is an unfortunate choice of words, as a conventional “ideal filter” can only be conceived of as being ideal in the frequency domain. A conventional “ideal filter” has some very non-ideal characteristics, such as being non-causal, for example. Nevertheless, such “ideal filters” are often a starting point, and then classical filter designs are referred to as approximations, since their magnitude frequency response will only approximate that of the ideal response.

The term “approximation” is also used in this book in the sense in which it was used two paragraphs above. A physical realization will only approximate the filter design $H(s)$. This is because of physical limitations, such as component value tolerances, etc. So a realized filter may be thought of as doubly an approximation. The physical realization only approximates $H(s)$, and $H(s)$ only approximates some “ideal” response.

A valuable relationship between analog filter theory and analysis and modern digital signal processing is made by the application of MATLAB to both the design and analysis of analog filters. MATLAB was used significantly in developing the material presented in this book, and throughout the textbook computer-oriented problems are assigned. The disk that accompanies this book contains MATLAB functions and m-files written specifically for this book. The MATLAB functions on the disk extend basic MATLAB capabilities in terms of the design and analysis of analog filters. The m-files are used in a number of examples in the book. They are included on the disk as an instructional aid. See **Appendix B** for a description of the contents of the accompanying disk. These functions and m-files are intended to be used with MATLAB, version 5, *Student Edition*, and are not stand-alone. Therefore, familiarity with MATLAB is essential, or the willingness to study it on one's own, for maximum benefit from the study of this book.

In **Chapter 1, Introduction**, basic filtering concepts are presented, such as how a filter is used to estimate a signal from a noisy version of it, or to separate signals based on their frequency content. **Chapter 1** also gives a number of practical examples of where a properly designed analog filter can be of significant practical use. It also gives an overview of the text, and therefore chapters of the book will only be briefly introduced here.

In **PART I, Approximation Design and Analysis**, consisting in **Chapters 2** through **9**, fundamental concepts and the design and analysis all of the common classical filters are theoretically presented: Butterworth, Chebyshev, elliptic and Bessel. Some filter designs, such as Gaussian and Legendre, which are not as well known, are also covered.

In **PART II, Implementation and Analysis**, consisting of **Chapters 10** and **11**, implementation of a filter in a circuit schematic diagram is presented. **Chapter 10** introduces passive filter design, and **Chapter 11** introduces active filter design.

Features of this book that may be of interest include the following:

- There are over 200 figures in the book. Many of these figures report basic characteristics of given analog filter design methods: these data graphs were obtained from MATLAB simulations.
- The data graphs mentioned immediately above include the magnitude frequency response, the phase response, phase delay, group delay, unit impulse response, and unit step response, for several filter orders. These data graphs are for filters with a normalized 3 dB cutoff frequency for ease of comparing different filters.
- Not only are all of the classical filter design methods covered (Butterworth, Chebyshev Type I, Chebyshev Type II, Bessel, and elliptic), but other methods are also included: Gaussian, Legendre, ultraspherical, Papoulis, and Halpern.
- There are over 100 examples in the book.
- There is a total of 345 homework problems in the book, appearing at the ends of the chapters.
- On the accompanying disk (standard 3 1/2 inch PC floppy) there is over 30 MATLAB m-files and functions written specifically for this book. The functions include filter designs for Gaussian, Legendre, ultraspherical, Papoulis, and Halpern filters. See **Appendix B** for a complete list of the contents of the disk.
- A solutions manual, containing the solutions for selected homework problems, is available from the publisher for qualified instructors who have adopted the book for classroom use.

This book has grown out of the author's experience of teaching a course on analog filters over the past ten years. The author would like to express his appreciation to the classes of students at Wichita State University who have taken the course on analog filters with the author, have suffered through earlier manuscript versions that preceded this book, and offered comments and suggestions toward improving the final result. Being their teacher has been a rewarding experience.

Larry D. Paarmann

TABLE OF CONTENTS

	Page
PREFACE	v
Chapter	
1. INTRODUCTION	1
1.1 Filtering Concepts	1
1.2 Classes of Filters	4
1.3 Applications of Analog Filters	8
1.4 Historical Perspective	15
1.5 A Note on MATLAB	16
1.6 Overview of the Text	17
1.7 Chapter 1 Problems	19

PART I Approximation Design and Analysis

2. ANALOG FILTER DESIGN AND ANALYSIS CONCEPTS	23
2.1 Time, Frequency, and s Domains	24
2.2 The Paley-Wiener Theorem	34
2.3 Time-Bandwidth Products	40
2.4 Frequency Band Definitions	51
2.5 Filter Selectivity and Shaping Factor	52
2.6 Imposed Constraints	54
2.7 Analog Filter Design Theorem	58
2.8 First-Order Transfer Functions	66
2.9 Second-Order Transfer Functions	70
2.10 Transfer Functions with Orders Greater than Two ...	76
2.11 Minimum-Phase Transfer Functions	77
2.12 All-Pass Transfer Functions	78
2.13 Time-Domain Response	80
2.14 Phase Delay and Group Delay	81
2.15 Hilbert Transform Relations	88
2.16 Frequency Scaling	98
2.17 Chapter 2 Problems	102

TABLE OF CONTENTS (continued)

Chapter	Page
3.	BUTTERWORTH FILTERS 113 3.1 Maximally-Flat Magnitude 113 3.2 Filter Selectivity and Shaping Factor 115 3.3 Determination of Order 117 3.4 Pole Locations 119 3.5 Phase Response, Phase Delay, and Group Delay ... 122 3.6 Time-Domain Response 125 3.7 Chapter 3 Problems 126
4.	CHEBYSHEV TYPE I FILTERS 131 4.1 Equiripple Passband Magnitude 131 4.2 Filter Selectivity and Shaping Factor 136 4.3 Determination of Order 137 4.4 Chebyshev Polynomials 138 4.5 Pole Locations 140 4.6 Phase Response, Phase Delay, and Group Delay ... 143 4.7 Time-Domain Response 146 4.8 Comparison with Butterworth Filters 148 4.9 Chapter 4 Problems 150
5.	CHEBYSHEV TYPE II FILTERS 155 5.1 Equiripple Stopband Magnitude 155 5.2 Filter Selectivity and Shaping Factor 160 5.3 Determination of Order 161 5.4 Inverse Chebyshev Polynomials 162 5.5 Location of the Poles and Zeros 163 5.6 Phase Response, Phase Delay, and Group Delay ... 166 5.7 Time-Domain Response 169 5.8 Comparison with Butterworth and Chebyshev Type I Filters 170 5.9 Chapter 5 Problems 172
6.	ELLIPTIC FILTERS 177 6.1 Introduction 177 6.2 Elliptic Integrals and Jacobian Elliptic Functions ... 182 6.3 Equiripple Passband and Stopband Magnitude 188 6.4 Computing ω_s Given A_p , A_s , N and ω_p 195 6.5 Filter Selectivity and Shaping Factor 196 6.6 Determination of Order 197

TABLE OF CONTENTS (continued)

	Page
6.7	Chebyshev Rational Functions 198
6.8	Location of the Poles and Zeros 200
6.9	Phase Response, Phase Delay, and Group Delay . . . 202
6.10	Time-Domain Response 205
6.11	Comparison with Preceding Filters 207
6.12	Chapter 6 Problems 208
7.	BESSEL FILTERS 215
7.1	Introduction 215
7.2	Maximally-Flat Group Delay 217
7.3	Filter Selectivity and Shaping Factor 223
7.4	Determination of Order 224
7.5	Pole Locations 225
7.6	Phase Response, Phase Delay, and Group Delay . . . 225
7.7	Time-Domain Response 228
7.8	Comparison with Preceding Filters 228
7.9	Chapter 7 Problems 230
8.	OTHER FILTERS 233
8.1	Transitional Filters 233
8.2	Gaussian Filters 235
8.3	Legendre Filters 238
8.4	Ultraspherical Filters 245
8.5	Papoulis Filters 249
8.6	Halpern Filters 255
8.7	Phase-Compensated Filters 259
8.8	Chapter 8 Problems 266
9.	FREQUENCY TRANSFORMATIONS 271
9.1	Lowpass-to-Lowpass 271
9.2	Lowpass-to-Highpass 274
9.3	Lowpass-to-Bandpass 287
9.4	Lowpass-to-Bandstop 303
9.5	Chapter 9 Problems 320

TABLE OF CONTENTS (continued)

	Page
PART II Implementation and Analysis	
10. PASSIVE FILTERS	329
10.1 Introduction	329
10.2 Continued-Fraction Ladder Implementation	331
10.3 Frequency Transformation Circuit Operations	345
10.4 Impedance Scaling	348
10.5 Summary and Examples of Passive Filter Implementation	349
10.6 Chapter 10 Problems	354
11. ACTIVE FILTERS	359
11.1 First-Order Stages	360
11.2 Second-Order Stages	364
11.3 Summary and Examples of Active Filter Implementation	377
11.4 Chapter 11 Problems	383
 APPENDICES	
APPENDIX A: Introduction to MATLAB	393
APPENDIX B: Contents of the Accompanying Disk	409
APPENDIX C: The MATLAB m-File EXAMP6_1.m	413
APPENDIX D: The MATLAB m-File EXAMP6_2.m	417
APPENDIX E: The MATLAB m-File EXAMP6_6.m	423
 REFERENCES	425
 INDEX	433

CHAPTER 1

INTRODUCTION

In this chapter very basic concepts of analog filters are presented in order to provide motivation for the rest of the book, and to give some indication of the importance of the topic. Therefore, defining what is meant by a “filter,” and how an analog filter contrasts with other filter types is given: the topic or filtering is broad, including modern, or statistical filters, adaptive filters, etc. Also, to strengthen the basic concept of what a filter is and how it is used, several examples of analog filter applications are presented: these examples indicate the importance and pervasiveness of analog filters. A brief historical perspective is presented on analog filter theory and design. Also, a brief note on the use of MATLAB¹ in this book is given. Finally, to provide a scope of how the material of the book is to be unfolded, a brief overview of the text is given.

1.1 FILTERING CONCEPTS

The word “filter” is in common use, such as an oil filter used in an automobile. Also used in an automobile is an air filter and a fuel filter. An air filter is also used in home heating / air conditioning systems. A lint filter is used in a clothes dryer. Photographers frequently make use of a lens filter. In all of these applications the filter is a device that removes something: small metal particles, dust, lint, etc. The photographic filter suppresses a certain band of wavelengths, or is designed to pass light of a particular polarity, etc.

Electric filters may be thought of in a similar way. An electric analog filter is typically designed to pass certain things and attenuate if not completely block other things. Since an analog filter is typically time-invariant, what it passes or blocks is not time-dependent *per se*. Rather, similar to the photographic filter, it is typically designed to pass certain wavelengths, or frequencies, and attenuate or block others. Therefore, many of the concepts, and specifications, of analog filters are defined or

¹MATLAB is a registered trademark of The MathWorks, Inc.

explained in the frequency domain. Just what a given filter accomplishes is much more readily comprehended in the frequency domain than in the time domain.

A general area of application of analog filters is passing one signal while suppressing others that are *non-overlapping* in the frequency domain. These same signals, generally, are overlapping in the time domain, i.e., they occur at the same time and are added together. A time-domain plot of such a composite signal would not suggest any convenient way of separating them. On the other hand, a frequency domain plot of such a composite signal does suggest a convenient way of separating them, as shown below.

As an example of signals that overlap in the time domain but not in the frequency domain, consider **Figure 1.1**, where $x(t)$ is as follows:

$$x(t) = \sum_{i=1}^N A_i [1 + \mu_i f_i(t)] \cos(\omega_i t) . \quad (1.1)$$

That is, $x(t)$ is the sum of N amplitude-modulated (AM) signals, which will be recognized by anyone who has studied basic modulation techniques: A_i is the i -th amplitude coefficient, μ_i is the i -th modulation coefficient, $f_i(t)$ is the i -th information signal (e.g., speech or music), and ω_i is the i -th transmitter carrier frequency (Proakis and Salehi, 1994; Stremler, 1990). Suppose the output signal $y(t)$ of **Figure 1.1** is as follows:

$$y(t) = A_3 [1 + \mu_3 f_3(t)] \cos(\omega_3 t) . \quad (1.2)$$

That is, $y(t)$ is just one of the terms ($i = 3$) in the sum shown in (1.1). The signal $y(t)$ could then be applied to an AM demodulator to recover $f_3(t)$, and $f_3(t)$ could then be amplified and listened to on a loudspeaker, assuming $f_3(t)$ is speech or music. This example, of course, illustrates how an AM broadcast receiver is able to select one of many AM broadcast signals, all of which are simultaneously impinging upon the radio receiver antenna. This is illustrated in **Figure 1.2**. Note that **Figure 1.2** clearly reveals the operation of the filter shown in **Figure 1.1**: the filter is a bandpass filter; only certain frequencies are allowed to pass through it. **Figure 1.2** is an idealized

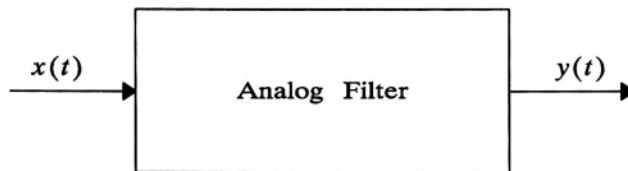


Figure 1.1 A simple block diagram representation of a filter.

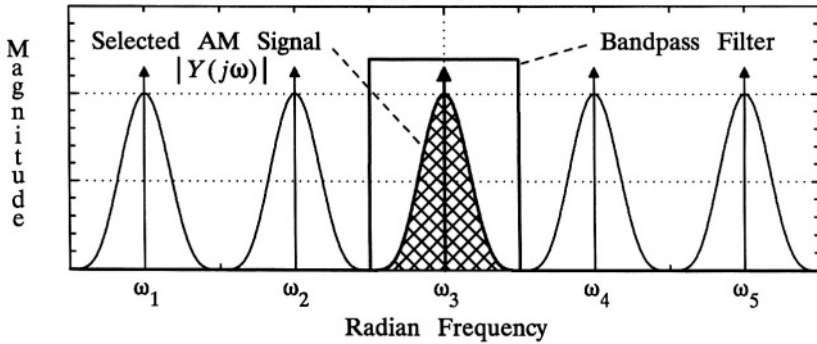


Figure 1.2 Illustration of several AM broadcast signals impinging upon a radio receiver antenna, and a bandpass filter selecting one for demodulation and listening.

illustration: (1) Five carrier frequencies are shown, ω_1 through ω_5 ; (2) The impulses represent those five carrier frequencies; (3) The smooth symmetrical curves on either side of each impulse represent the upper and lower sidebands of the transmission; (4) These five carrier frequencies and associated sidebands represent five AM broadcast transmissions adjacent to one another in the frequency domain, which would be five adjacent radio stations on a radio receiver tuning dial; (5) The analog filter of **Figure 1.1** has an idealized frequency response illustrated in **Figure 1.2** by the rectangular function; (6) In the frequency domain, the output of the analog filter of **Figure 1.1** consists only of the *product* of the bandpass filter response and the composite frequency spectrum shown in **Figure 1.2**, which consists only of the crosshatched transmission with carrier frequency ω_3 ; (7) Therefore, in the time domain, the output of the filter is as shown in (1.2), assuming the bandpass filter gain across the passband is unity.

Note that showing the relationship between $y(t)$ and $x(t)$ in the time domain, $y(t) = h(t) * x(t)$, where $h(t)$ is the unit impulse response of the filter, and $*$ denotes convolution, would not so readily reveal the operation of the filter. In the frequency domain, $Y(j\omega) = H(j\omega)X(j\omega)$, where the magnitudes are illustrated in **Figure 1.2**, is much more illustrative.

Another general area of application of analog filters is estimating a desired signal while suppressing others that are *overlapping* in the frequency domain. This is often done to improve the signal-to-noise ratio of a noisy signal. As an example, consider **Figure 1.1** again, but this time let $x(t)$ be as follows:

$$x(t) = s(t) + n(t), \quad (1.3)$$

where $s(t)$ is a signal of interest, and $n(t)$ is additive noise. Suppose it is desired to design the filter in **Figure 1.1** to estimate the signal $s(t)$ from the noisy version of it, $x(t)$. Or, in other words, it is desired that the filter be designed to reduce the noise in $x(t)$ while having a minimal degradation effect upon the signal $s(t)$. Or put another way, it is desired that the filter be designed in such a way as to improve the signal-to-noise ratio.² In this context, the analog filter is a signal estimator. In the time domain it may be difficult to visualize what characteristics the filter should have. Suppose that **Figure 1.3** (a) illustrates the power spectral density (PSD)³ of $x(t)$, denoted $S_x(\omega)$. Suppose, for the purposes of this example, that it is known that the power spectral density of $s(t)$, denoted $S_s(\omega)$, is as illustrated in **Figure 1.3** (b), and that $n(t)$ is white noise⁴ with power spectral density $N_0/2 = 1.0$ (note that the level of the line between 10 rad/s and 30 rad/s in **Figure 1.3** (a) is unity). In the frequency domain it is apparent that the signal-to-noise ratio can be improved by designing the analog filter as a lowpass filter with a cutoff frequency at the maximum frequency content of $s(t)$. This is illustrated in **Figure 1.3** (b). The signal-to-noise ratio at the output of the filter can be readily determined to be 10.⁵ If the noise $n(t)$ on the input of the filter is not truly white, but rather bandlimited white noise with a bandwidth of 1000 rad/s, then the signal-to-noise ratio on the filter input can be readily shown to be only 0.1. Therefore, the lowpass filter has significantly increased the signal-to-noise ratio.

1.2 CLASSES OF FILTERS

There are many ways in which to classify filters. Below are given several ways in which to do so. Filtering is a broad topic, and classification helps one to gain an overall perspective, and to appreciate how analog filters fit into this larger scheme of filtering.

²Signal-to-noise ratio is defined as the power of the signal divided by the power of the noise. For this concept, as well as others in this illustration, refer to an introductory textbook on communications systems, such as, for example, the one by Stremler (1990).

³The power spectral density of a signal indicates how the power in a signal is a function of frequency, and is a topic generally included in introductory textbooks on communication systems (Proakis and Salehi, 1994; Stremler, 1990), and in books on signal processing (Ambardar, 1995; Oppenheim and Schaffer, 1975). The power in a given frequency range, say from ω_1 to ω_2 , is directly related to the integral of the PSD across that frequency range, hence power spectral density.

⁴white noise is uncorrelated such that its PSD is constant.

⁵Given that the level of $S_s(\omega)$ for frequencies from 4 rad/s to 8 rad/s is 6 2/3, the ratio of the area under $S_s(\omega)$ to the area under $N_0/2$ for frequencies from 0 to 10 rad/s can readily be shown to be 10.

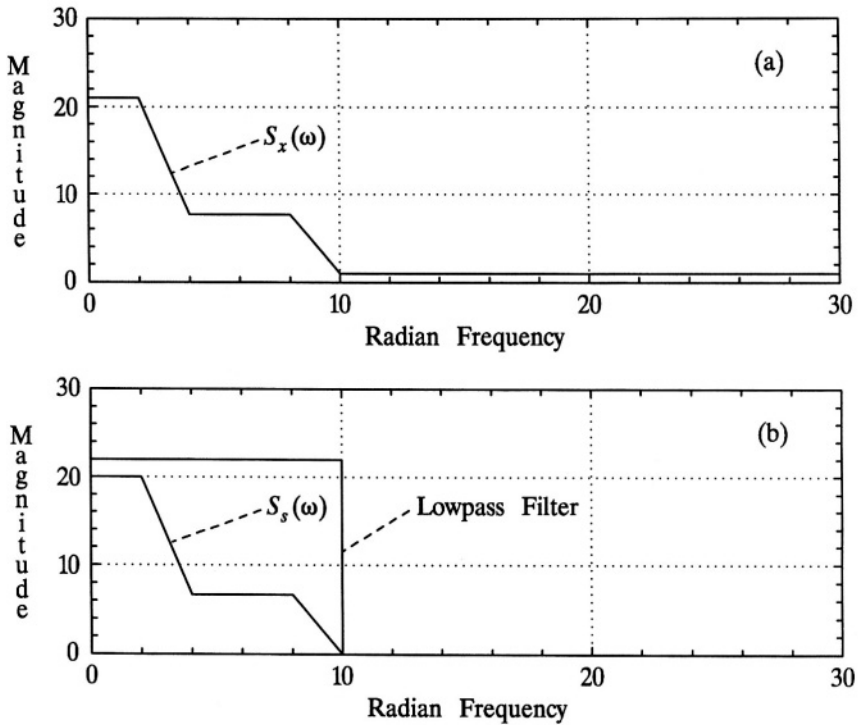


Figure 1.3 The power spectral density (PSD) of a noisy signal is shown in part (a), the PSD of the signal itself is shown in part (b) as is a lowpass filter that will pass the signal without distortion.

Frequency Selective Filters

One class of filters is denoted as frequency selective. What is meant by *frequency selective* is that (1) the frequency response of the filter is generally fixed, and (2) the filter is designed to meet certain frequency specifications that are determined mostly by engineering judgment. While a frequency selective filter may have certain parameters adjustable, such as the 3 dB cutoff frequency, by having an adjustable component or switchable components, generally the frequency response is taken to be fixed. The filter is designed to meet certain specifications such as the 3 dB cutoff frequency and the stopband edge frequency and attenuation.⁶ However, those specifications are dictated by engineering judgment, and are not generally

⁶Definitions of such terms as passband, transition band, and stopband are given in **Chapter 2**.

optimal in any sense. That is, an engineer, in designing, say, an anti-aliasing filter,⁷ may somewhat arbitrarily choose a 3 dB cutoff frequency of 3000 Hz, and a stopband edge of 4000 Hz with an attenuation of 40 dB. Once the specifications have been chosen, then the filter is designed to meet those specifications. Not only are the specifications somewhat arbitrary, but certain filter characteristics are more-or-less ignored. That is, magnitude frequency response passband and stopband edges may be specified, while transition band response characteristics are not specified, and neither are the phase response characteristics nor the time-domain response characteristics. However, a small set of design parameters leads to a tractable design procedure, and then analysis of the proposed design will yield other filter characteristics for consideration as to their acceptability. The majority of analog filters are in this class. However, so are many digital filters.

Statistical Filters

Statistical filters have a fixed frequency response, but the shape of that frequency response is not chosen *a priori*, nor using engineering judgment, and they, in general, do not have flat passband characteristics. In fact, it may even be difficult to define a passband, a stopband, etc. Rather, these filters are designed to optimize some statistical design criterion. They are sometimes referred to as *modern* filters, however the term modern has come to denote a broader class, often including Kalman and other filters. Consider the noisy signal represented by (1.3). As was noted in **Section 1.1**, a filter with an *a priori* chosen frequency response can indeed improve the signal-to-noise ratio, but is the *a priori* filter optimal? Probably not. Suppose that statistics of the signal $s(t)$ and of the noise $n(t)$ are known, or can be obtained; then the filter parameters could be optimized for the largest signal-to-noise ratio at the filter output, or for the minimum mean-squared error between $y(t)$ and $s(t)$. This is the statistical filter design approach. The most well-known statistical filter is the Wiener filter (Davenport and Root, 1987; Gardner, 1986; Haykin, 1989; Therrien, 1992), but other statistical filters, such as the eigenfilter (Haykin, 1996) are also in this class.

Adaptive Filters

In a sense, statistical filters are adaptive, or data dependent, as the filter parameters depend upon statistics of the signal data, but since they assume that the data are stationary they are not referred to as adaptive filters. Adaptive filters operate in a non-stationary environment and therefore the filter parameters change with time as the statistics of the data change with time. Otherwise they are similar to statistical filters, and are designed to optimize some design criterion. Common filters of this class are the Least-Mean-Square (LMS) and Recursive-Least-Square (RLS) filters (Alexander, 1986; Haykin, 1996; Honig and Messerschmitt, 1984; Widrow and

⁷An anti-aliasing filter is used just prior to the input of an analog-to-digital converter to limit the frequency bandwidth of the signal to prevent aliasing. See **Example 1.6** below, and, for example, Oppenheim and Schaffer (1989).

Stearns, 1985). Kalman filters may be put in this class. Kalman filters differ somewhat in that they are usually cast as a state estimator rather than a signal estimator, but are mathematically similar to the RLS filter (Brown and Hwang, 1992; Candy, 1986; Chui and Chen, 1991). It has been suggested that the RLS filter may be viewed as the measurement update of the Kalman filter (Haykin, 1996).

Digital Filters

Statistical and adaptive filters are typically digital filters, however the term *digital filters* usually refers to digital frequency-selective filters. Digital filters are implemented in software and execute in microprocessors, personal computers, mainframe computers, etc. Often, digital filters are used to process analog signals by first going through an analog-to-digital converter. After processing, the output of the digital filter may well then be converted back to an analog signal. In such a real-time filtering situation, usually accomplished with a microprocessor, and commonly with a microprocessor designed especially for signal processing applications, the filtering application is analog-in and analog-out. However, digital filtering is also often accomplished off-line in personal or mainframe computers. Many digital filter design procedures are based on analog prototypes, and others are also frequency selective (Cunningham, 1992; Hamming, 1989; Loy, 1988; Oppenheim and Schaffer, 1975; Oppenheim and Schaffer, 1989; Parks and Burrus, 1987; Proakis and Manolakis, 1988; Terrell, 1988; Williams, 1986).

Analog Filters

Within the class of frequency-selective analog filters, filters may be further categorized according to the specific design method and the way they are implemented. Specific design methods that have been developed include Butterworth, Chebyshev Type I, Chebyshev Type II, Cauer (or elliptic), and Bessel,⁸ all of which are presented in **Part I** of this book, as well as others. Each design method has advantages and disadvantages, which are discussed and illustrated as the material of this book unfolds.

Analog filters may be referred to as *passive*, which indicates that there are no active elements in the filter implementation, but usually also is further restricted to an implementation that is made up of R's, L's and C's. Analog filters that are also passive, but more specialized, would include surface acoustic wave (SAW) filters, mechanical resonators and quartz crystal filters (Sheahan and Johnson, 1977).

Analog filters may also be referred to as *active*, which indicates that the implementation includes active elements, such as operational amplifiers (op amps), or possibly other active elements such as transistors. The main advantage of op amp active filters, due to the very low output impedance characteristic of op amps, and also very high open loop gain, and high input impedance, is that op amp stages have

⁸Filters designed using these methods are sometimes referred to as *classical* filters.

inherent buffering, which means that the overall transfer function of several op amp stages is the product of the individual stage transfer functions, ignoring loading effects of subsequent stages. This greatly simplifies the theoretical implementation. That is, for example, a sixth-order op amp filter can be implemented by cascading three second-order op amp stages, where each second-order stage is implemented independently of the other two stages. Passive analog filters do not enjoy this simplification, and the entire transfer function must be implemented as one non-separable whole. Passive and active (op amp) filter implementation is presented in **Part II** of this book.

1.3 APPLICATIONS OF ANALOG FILTERS

In this section, several examples are given that illustrate the application of frequency selective analog filters to practical engineering use. Selection of a signal from others separated in frequency, estimating a signal in noise, frequency selection decoding, intentionally frequency-limiting a signal, contributing to the demodulation of signals, rejection of interference signals, and separation of signals according to frequency bands, are all illustrated.

Examples 1.1 and 1.2

The first two examples were given above in **Section 1.1**. More specifically, **Example 1.1** illustrates the use of a bandpass filter to extract one desired signal from the sum of several signals, where the individual signals are separated in the frequency domain (see **Figure 1.2**). **Example 1.2** illustrates using a lowpass filter to improve the signal-to-noise ratio of a signal imbedded in noise, but where the noise has a much wider bandwidth than does the signal (see **Figure 1.3**). □

Example 1.3

Consider a high-gain instrumentation amplifier used to measure electroencephalogram (EEG) signals. EEG signals are low-level with an equivalent high source impedance. Electrodes are applied to the scalp of the subject in order to measure these signals. The electrodes are very high impedance devices so as to not present much of a load for the measurement signals. Due to the high impedance, the low-level EEG signal may well be corrupted by additive 60 Hz and/or 120 Hz, derived from fluorescent lighting and other electrical appliances and equipment. Since the additive 60/120 Hz noise tends to be mostly common mode, a *differential* instrumentation amplifier may significantly suppress the additive noise, but will not completely remove it, especially since the EEG signal is likely much lower in amplitude than the additive noise (sometimes referred to as hum). EEG signals, in the frequency domain, have most of the signal energy below 60 Hz. Since the noise (hum) components are above the EEG frequency components, a lowpass filter may be

used to suppress the 60 Hz and 120 Hz components. Since it would be difficult to design and implement a lowpass filter that would fall off fast enough to significantly suppress a strong 60 Hz component while passing all desired signal components, perhaps a better alternative would be to use notch filters tuned to 60 Hz and 120 Hz. A notch filter is a special case of a bandstop filter and passes all frequencies except for a narrow range. A simple solution is a shunt series LC circuit that shorts out at series resonance.

Other biomedical instrumentation amplifier applications have similar filtering needs. The amplifier used in an electrocardiogram (EKG) signal recorder is one such application, or amplifiers used in electromyogram (EMG) signals, in general. □

Example 1.4

Consider the segment of a speech signal shown in part (a) of **Figure 1.4**. In part (b) of the figure is shown the same segment of speech with a significant amount of additive noise. A practical situation, for example, where this may occur, is in the output signal from the microphone of a pilot in a noisy cockpit such as a jet fighter plane. Since the additive noise may well be much wider in frequency bandwidth than the minimum required for intelligible speech, one solution is to filter the noisy speech to eliminate as much of the additive noise as possible while preserving enough of the speech spectrum for good intelligibility. This is a practical application of the more theoretical situation described in **Example 1.2**. Therefore, if the noisy speech signal, as shown in part (b) of the figure, is lowpass filtered, the filter output should be “cleaner,” i.e., have a higher signal-to-noise ratio than does the filter input. The output of a lowpass filter with the noisy speech signal as the input is shown in part (c) of **Figure 1.4**. □

Example 1.5

Consider a touch-tone telephone tone decoder. A touch-tone encoder consists, for each key pressed, of the combination of two tones. There are seven individual tones, but a total of 12 keys (Bigelow, 1994; Huelsman, 1993; Lindquist, 1977). Each tone-pair identifies the unique key that has been pressed. See **Figure 1.5** for an illustration of the encoder. As an example, if the 8 key is pressed (also labeled “TUV”), a signal consisting of the sum of two sinusoids at frequencies 852 Hz and 1336 Hz would be sent to the telephone switching station. The tone decoder consists of seven bandpass filters, each one tuned to one of the seven possible tones. The output of each filter is rectified and filtered, so that the voltage levels indicate which two tones are present at any given time a key is pressed. This is illustrated in **Figure 1.6**. The tone decoder is located in the telephone switching station. If a caller presses the 8 key, then the decision logic block in **Figure 1.6** would so indicate. The highpass and lowpass filters in **Figure 1.6** relax the design constraints on the bandpass filters, and thereby help in the decoding process. Note that the tone decoder, then, includes a highpass and a lowpass filter, and seven bandpass filters. □

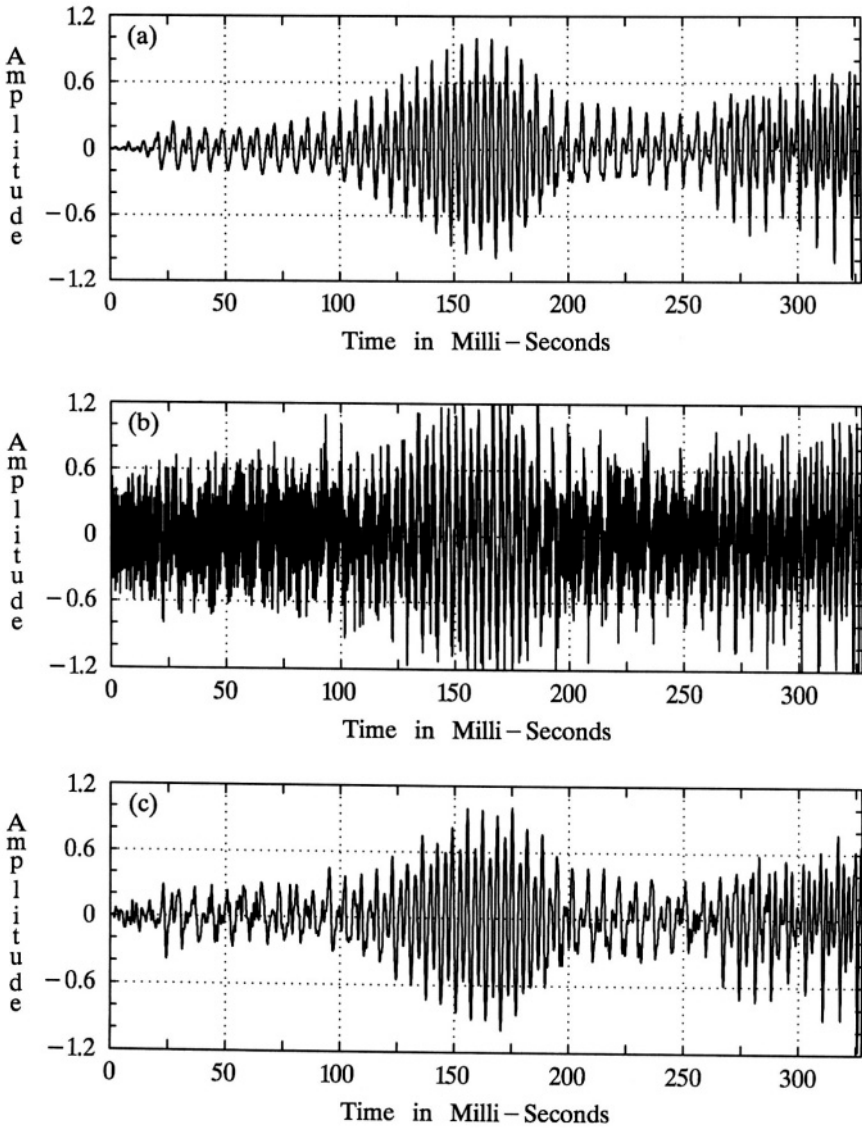


Figure 1.4 Speech data for *Example 1.4*. Part (a) is a segment of clean speech data. Part (b) is the clean speech of part (a) corrupted by additive noise. Part (c) is the noisy speech data of part (b) after lowpass filtering.

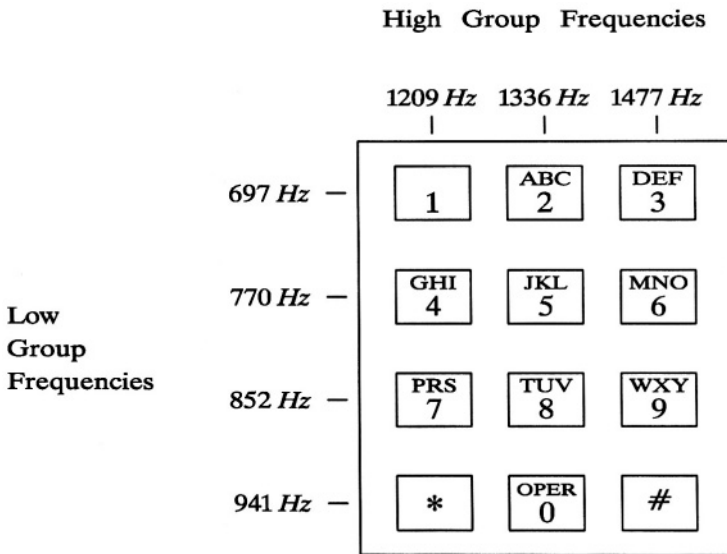


Figure 1.5 A touch-tone telephone key pad and the corresponding tone frequencies. See *Example 1.5*.

Example 1.6

The digital signals that are processed by various digital signal processing methods are often obtained from analog signals by means of an analog-to-digital converter. To satisfy the Nyquist Theorem, the sample rate must be greater than twice the highest frequency in the analog signal. Often, to help insure that the Nyquist Theorem is satisfied, the analog signal is first filtered by an analog lowpass filter, in order to deliberately limit the signal bandwidth. This is especially necessary when the bandwidth is limited to less than what the signal naturally has, in order to use a lower sample rate. In speech processing this is often done, where the sample rate is 8000 *samples/s* and the lowpass filter has, perhaps, a 3 dB cutoff frequency of 3000 Hz. This type of filter is referred to as an anti-aliasing filter. See, for example, Oppenheim and Schaffer (1989). □

Example 1.7

Consider the demodulation of frequency modulated (FM) stereo-multiplexed signals, as transmitted by commercial, broadcast FM radio stations. After a received stereo-multiplexed signal is FM demodulated, the frequency spectrum could be represented as shown in **Figure 1.7**, where L+R indicates the sum of the left and

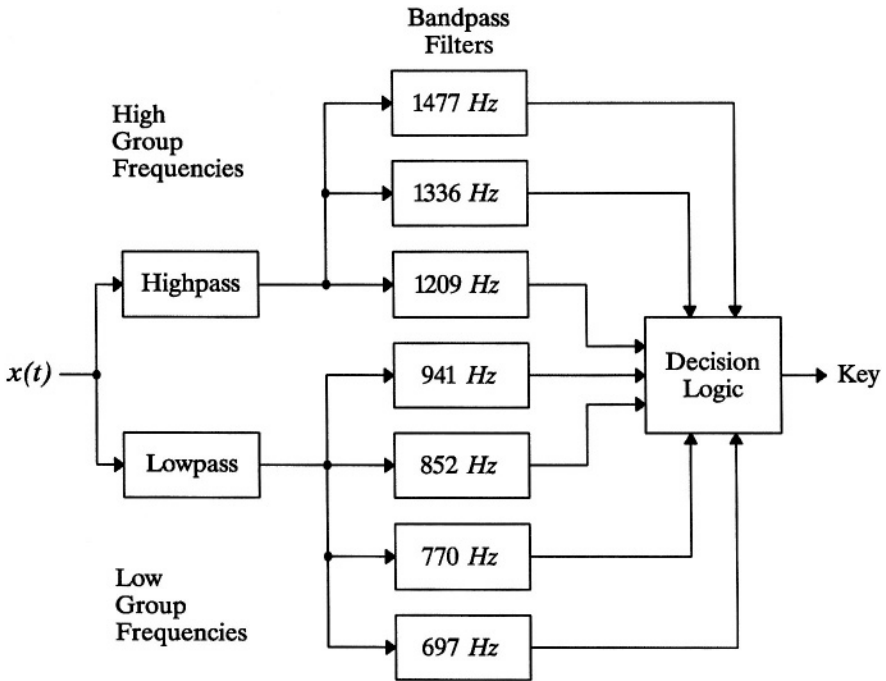


Figure 1.6 A touch-tone telephone tone decoder. See Example 1.5.

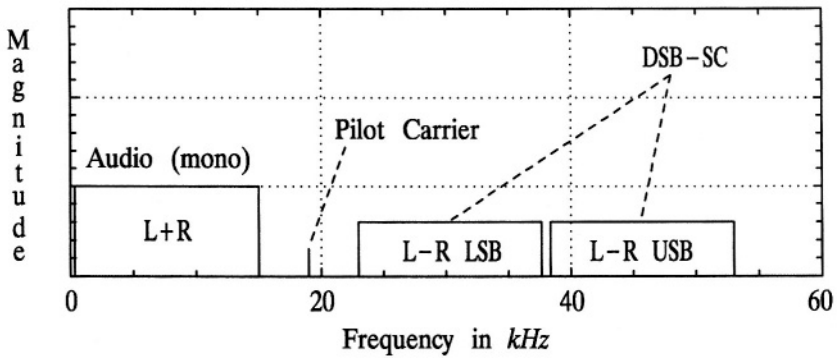


Figure 1.7 Illustration of the frequency spectrum of an FM demodulated stereo multiplexed signal. See Example 1.7.

right channels (i.e., monophonic), L-R indicates the left-channel signal minus the right-channel signal, LSB and USB indicates the lower-sideband and upper-sideband, respectively, DSB-SC indicates double-sideband suppressed carrier, and the pilot carrier is a 19 kHz sinusoid (Stremmer, 1990). Note that the suppressed carrier has a frequency of 38 kHz . Note that if this signal (the entire signal illustrated in **Figure 1.7**) was simply amplified and applied to a loudspeaker, that a monophonic signal would be heard. The left and right channels would be added together. All frequencies above 20 kHz would not be audible, and the 19 kHz pilot carrier would also likely not be heard, or it could be removed with an appropriate filter. This is how monophonic FM receivers are capable of receiving FM stereo broadcasts and still be capable of recovering the full monophonic signal. The Federal Communications Commission (FCC) regulated that FM stereo broadcasts must be capable of being received by FM monophonic receivers.

In an FM stereo receiver the FM demodulated signal, with spectrum illustrated in **Figure 1.7**, is lowpass filtered to recover the L+R signal, and is also applied to the phase-locked loop stereo demodulator shown in **Figure 1.8** (as an example; other demodulation schemes also exist) (Stremmer, 1990). The bandpass filter shown, tuned to 19 kHz , allows only the pilot carrier to pass (this signal also lights the “stereo” light on some FM receivers) and phase-locks the 38 kHz voltage controlled oscillator. The 38 kHz serves as the carrier to demodulate the double-sideband suppressed carrier modulation for the L-R signal. Note that the 19 kHz pilot is used rather than transmitting a 38 kHz carrier for several reasons, including the fact it would be much more difficult to extract for phase synchronization because the L-R LSB and USB are much closer together than is the upper edge of the L+R signal and the lower edge of the L-R LSB signal. If the L+R and L-R signals are added, only the left channel signal remains. If the L+R and L-R signals are subtracted, only the right channel signal remains. This is how the right and left channel signals are separated. Note that several analog filters have been used to help accomplish this task. □

Example 1.8

Consider the crossover network used in a three-way high-fidelity loudspeaker illustrated in **Figure 1.9**. Such a crossover network, as they are usually referred to, consists of a lowpass, a bandpass, and a highpass filter. The reasons for using such filters, rather than simply connecting the three speakers (woofer, midrange, and tweeter) in parallel, are for power efficiency and to avoid exceeding the power limitations of individual speakers (and also for impedance considerations). High-powered bass signals are applied only to the woofer, for example, and the midrange and tweeter speakers need not wastefully dissipate part of this power. Each speaker receives frequencies in the range in which it is most efficient. Special consideration is given to the design of such filters, in that in the transitional region between two speakers, for example frequencies where both the woofer and midrange respond, care

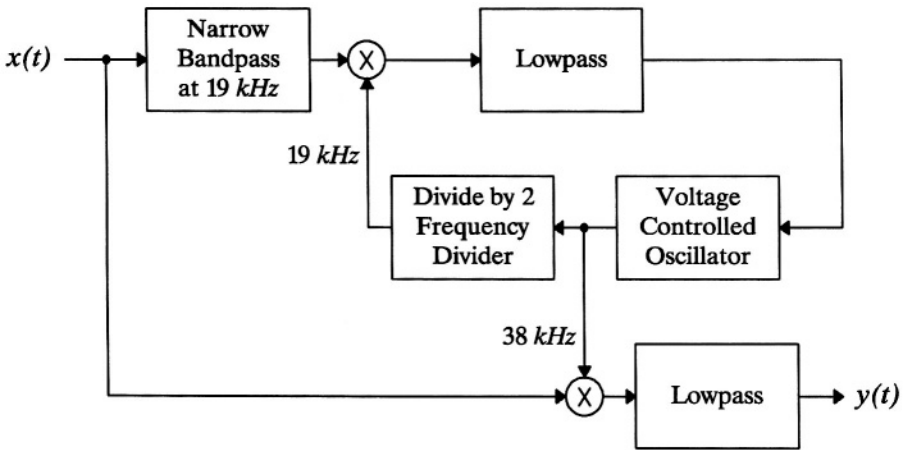


Figure 1.8 Illustration of a phase-locked loop FM stereo demodulator (Stremler, 1990). See *Example 1.7*.

must be taken that the two speakers are in phase, otherwise their acoustic outputs could interfere destructively. □

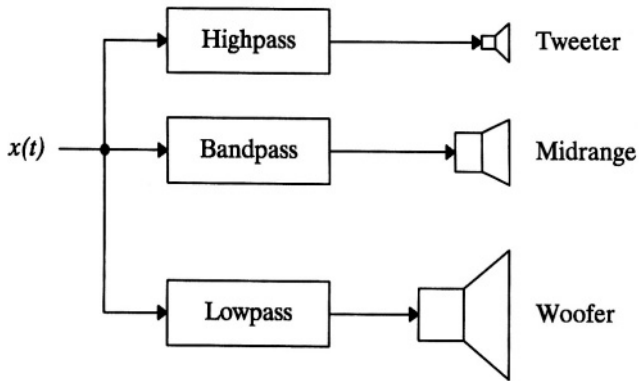


Figure 1.9 Three-way high-fidelity loudspeaker crossover network. See *Example 1.8*.

1.4 HISTORICAL PERSPECTIVE

Analog electric filters have been in use since the very early days of electrical engineering. Simple resonant filters, although perhaps not explicitly called such at the time, were used in radio communications before the introduction of the first active devices. For example, an early interference suppression circuit introduced by Marconi, called the X Stopper, was actually a bandpass filter (Blake, 1974).

However, it appears that the first systematic design approaches date from the first couple decades of this century when K.W. Wagner in Germany, and George Ashley Campbell (Campbell, 1911; Campbell, 1922) in the United States, independently developed such approaches to filter design (Darlington, 1984; Stephenson, 1985; Van Valkenburg, 1982).⁹ Other methods (such as insertion-loss synthesis) were introduced by Darlington in the United States and Cauer in Germany in the 1930s (Cauer, 1939; Darlington, 1939) with contributions also by Brune (Brune, 1931; Brune, 1932). Also during the 1930s and 1940s, the beginnings of active filter design occurred, as introduced by Black, Bode (Bode, 1945), and others.

During this same time period analog filter design procedures were formalized, such as what is now known as the Butterworth filter (Butterworth, 1930), and elliptic filters (Cauer, 1931; Norton, 1937; Weinberg, 1962). Bessel filters, based upon Bessel (lived 1784-1846) polynomials (Krall and Frink, 1949; Grosswald, 1951; Burchnall, 1951), were introduced by Thomson and Storch (Thomson, 1949; Thomson, 1959; Storch, 1954). Chebyshev filters, and inverse Chebyshev (also called Chebyshev type II) filters, based upon Chebyshev polynomials (Chebyshev, 1899), were developed during the 1950s (Cauer, 1958; Henderson and Kautz, 1958; Stephenson, 1985; Storer, 1957; Weinberg, 1962).¹⁰

In more recent years there has been development in analog filter design primarily along three lines: (1) introducing additional filter transfer function types exhibiting refinements, in some sense, compared with classical filters, (2) introducing computer-aided procedures for simulation, optimization, etc., and (3) circuit implementation. Since (1) is of primary concern with the emphasis of this book, with a signal processing perspective, only it will be expanded on in this historical review. Although it would not be useful to attempt an exhaustive list of filter transfer function types, a few of the more significant ones (in the author's opinion) will serve to indicate this further development. The author apologizes in advance for any significant filter transfer function types inadvertently left out. The order of mention is mostly chronological within two groupings: (1) general, where the primary concern

⁹ A brief biographical sketch of Campbell, with some discussion of his early contributions to filter design, is given by Brittain (1992).

¹⁰ A brief biographical sketch of Cauer, with a tribute to his contributions, has been given by Fettweis (1995).

is the magnitude response, and (2) where the primary concern is constant time delay / linear phase.

General Filter Design

Papoulis introduced filters with a maximum magnitude slope at the passband edge for a monotonic response falloff for a given order (1958). Halpern extended the work of Papoulis for optimum monotonic transition band falloff, based on Jacobi polynomials (1969). Ku and Drubin introduced filters based on Legendre and Hermite polynomials (1962). Scanlan introduced filters with poles that fall on an ellipse with equal frequency spacing, and noted the tradeoff between magnitude response characteristics and time-domain response characteristics as the eccentricity of the ellipse is varied (1965). Filter transfer functions based on ultraspherical polynomials, where Chebyshev, Butterworth, and Legendre filters are shown to be special cases, was introduced by Johnson and Johnson (1966). This was extended by ultraspherical and modified ultraspherical polynomials where a single parameter determines many transitional forms (Attikiouzel and Phuc, 1978). Extensions to Cauer filters have recently been made in two ways: lowering the pole Q_s by using quasi-elliptic functions (Rabrenovic and Lutovac, 1992), and by significantly reducing the complexity of designing elliptic filters without reference to elliptic functions (Lutovac and Rabrenovic, 1992).

Constant Time-Delay Design

Whereas Bessel filters are designed for a maximally-flat time delay characteristic, Macnee introduced filters that use a Chebyshev approximation to a constant time delay (1963). By allowing small amounts of ripple in the group delay or phase response (based on Chebyshev polynomials), similar to Macnee's objectives, Bunker made enhancements in delay filters (1970). Ariga and Masamitsu developed a method to extend the magnitude bandwidth of constant-delay filters (1970). By using hyperbolic function approximation, Halpern improved on Bessel filters, at least for low orders (1976). The so-called Hourglass filter design (Bennett, 1988) may be used to obtain transfer functions that have simultaneously equiripple time-delay and equiripple magnitude characteristics. Gaussian filters have magnitude and phase characteristics very similar to Bessel filters, but with less delay for the same order (Dishal, 1959; Young and van Vliet, 1995).

1.5 A NOTE ON MATLAB

Although a variety of programming languages and high-level software could be used to design, analyze, and simulate analog filters, MATLAB has been selected in this book because of its ease of use, wide-range availability, and because it includes many high-level analog filter functions, and good graphics capabilities.

Many homework problems in this book require the application of MATLAB. The MATLAB m-files on the disk that accompanies this book, requires as a minimum *The Student Edition of MATLAB*. A brief introduction to MATLAB is given in **Appendix A**.

1.6 OVERVIEW OF THE TEXT

A brief overview of the book was given in the **Preface**, however an expanded overview is given here. The main body of this book is divided into two parts. **Part I, Approximation Design and Analysis** is concerned with various design methods to arrive at the desired filter transfer function $H(s)$. In the design phase, a small number of design criteria are established, and then a minimum-order transfer function of specified type is determined. The design criteria are often specified in terms of the desired magnitude frequency response, such as the 3 dB cutoff frequency, stopband edge frequency, and the minimum attenuation in the stopband. However, other design criteria may be used, such as with Bessel filters where a maximally-flat delay characteristic is desired. The filter type, such as Butterworth, Chebyshev Type I, etc., is chosen based on engineering judgment (knowledge and experience with the various filter types). In the analysis phase, the tentative filter design is analyzed to determine its characteristics not specified by the design criteria, such as the phase response, group delay, impulse response, etc. Several competing designs may be compared in terms of analysis results for final selection. The more knowledge and experience an engineer has with the characteristics of various filter types, the less time and effort would need to be spent on analysis.

Chapter 2, Analog Filter Design and Analysis Concepts, the first chapter of **Part I**, presents basic concepts on filter design and analysis applicable to all filter types. The attempt is made to present most of the theoretical concepts that are useful in the following chapters in **Part I**. These concepts include the existence, or lack thereof, of a causal impulse response for a given magnitude frequency response, as expressed by the *Paley-Wiener Theorem*, and the relationship between the magnitude frequency response and the corresponding phase response, as expressed by the Hilbert transform. Since analog filters are usually designed starting with a magnitude response, this is the approach that is used in **Chapter 2**, and the *Analog Filter Design Theorem* is developed, giving insight as to what magnitude frequency responses can be designed, and which cannot. Two basic questions are answered: what are permissible magnitude responses for analog filters, and given a permissible magnitude response, what is the procedure for determining the transfer function $H(s)$ that will have that permissible magnitude response? Some filters, such as Bessel filters, are not specified in terms of the magnitude response, but their design procedures are similar in concept and will be, where appropriate, considered as a special case.

Chapter 3, Butterworth Filters, is a presentation of the first specific filter type presented in the book. It is historically one of the first developed methods, and is very commonly used in practice. It is designed to yield a maximally-flat magnitude response in the passband (actually at DC) and is frequently referred to as the *maximally-flat* design. The design is based on Butterworth polynomials.

Chapter 4, Chebyshev Type I Filters, is a presentation of the first of two filter designs based on Chebyshev polynomials. Type I filters have ripple in the passband of an equal across-the-band magnitude and of a specified amount, with the response monotonically falling off through the transition band and the stopband. These filters will usually meet a set of magnitude specifications with a lower order than will a comparable Butterworth design, but have less desirable phase response and time-domain characteristics.

Chapter 5, Chebyshev Type II Filters, covers the second filter design based on Chebyshev polynomials, having a flat magnitude response in the passband, but having ripple in the stopband of an equal across-the-band magnitude (referred to as equiripple) and of a specified amount. As with Type I filters, these filters will usually meet a set of magnitude specifications with a lower order than will a comparable Butterworth design, in fact, with the identical order as a Chebyshev Type I design. However, this design requires specified finite-value zeros in the transfer function, whereas a Chebyshev Type I design has no finite-value zeros for a lowpass filter. Therefore a Chebyshev Type II design has a somewhat more complex transfer function than does either a Butterworth or a Chebyshev Type I design. It also has less desirable phase response and time-domain characteristics than does a Butterworth design.

Chapter 6, Elliptic Filters, presents filters that have equiripple characteristics in both the passband and the stopband, but fall off monotonically through the transition band. This design is based on Chebyshev rational functions, which in turn are dependent upon Jacobian elliptic functions. Among common filter types, elliptic filters will meet given magnitude specifications with the lowest order. However, they have very poor phase response and time-domain characteristics. As will be seen in **Part II** (more specifically, **Chapter 12**), they also can have implementation components with large sensitivities (very critical component values).

Chapter 7, Bessel Filters, covers filters that are designed for maximally-flat group delay, and are closely related to Bessel polynomials. While optimized for flat group delay, the magnitude response is that of a lowpass filter. Since flat group delay is analogous to a linear phase response, Bessel filters have very good phase characteristics. They also have very good time-domain characteristics. However, for a given set of magnitude specifications, a Bessel filter requires the highest order of any of the common filter types.

Chapter 8, Other Filters, presents a few of the filter design methods that have been reported, other than the above classical filters, but are less known. These filters have certain desirable characteristics, but are generally only marginally superior to the

above classical filters. The chapter makes no attempt to give an exhaustive presentation, but does include a few additional design methods that should be considered when selecting a filter type.

Chapter 9, Frequency Transformations, presents methods for transforming a lowpass filter design prototype into another lowpass filter with a different cutoff frequency, a highpass filter, a bandpass filter, or a bandstop filter. Frequency transformations significantly simplify filter design, as all filter design can then begin with a lowpass prototype, and the prototype design alone receives a detailed design development.

Part II, Implementation and Analysis, consists of two chapters concerned with determining a circuit schematic diagram that implements a given filter design (the transfer function $H(s)$) and with the analysis of that design. These two chapters of **Part II** are on passive filter implementation, and active filter implementation. The chapters also include some analysis, in the form of SPICE (Simulation Program with Integrated Circuit Emphasis) analysis, but this serves mainly as verification of the implementation, as the magnitude frequency response, phase response, impulse response, etc. should all be the same, assuming the implementation is correct, as found using **Part I** techniques.

Chapter 10, Passive Filters, presents a ladder implementation method whereby a desired transfer function may be implemented using passive components. These filters are of particular importance in high power or very high frequency applications.

Chapter 11, Active Filters, presents analog filter implementation using operational amplifiers (op amps). These filter implementations are of particular importance in small-signal, low-frequency applications. They have the important advantage, due to the very low output impedance of op amps, of allowing each op amp stage to be designed independently of the other cascaded stages.

1.7 CHAPTER 1 PROBLEMS

- 1.1** Given that $x(t) = s(t) + n(t)$, as in (1.2), and that $S_f(\omega)$ is illustrated in **Figure 1.3** (b) (only one side shown), and that $N_0/2 = 1.0$, such that $S_x(\omega)$ is illustrated in **Figure 1.3** (a), verify that the signal-to-noise ratio (SNR) at the output of the lowpass filter shown in **Figure 1.3** (b) is 10. Note that the value of $S_f(\omega)$ between $\omega = 4$ and 8 rad/s is $6.6\bar{6}$.
- 1.2** The signal $y(t)$ in **Figure 1.8** is the L-R signal. The frequency spectrum of $x(t)$ is illustrated in **Figure 1.7**. Suppose that $x(t)$, in addition to being processed by that shown in **Figure 1.8**, is also filtered directly by another

lowpass filter with identical characteristics as the lower lowpass filter shown in **Figure 1.8**, producing a signal denoted as $z(t)$, which is the L+R signal.

- (a) Argue that the 38 kHz oscillator waveform in **Figure 1.8** must be in proper phase synchronization in order for $y(t) + z(t)$ to be equal to the left-channel signal and for $z(t) - y(t)$ to be equal to the right-channel signal.
- (b) If the 38 kHz oscillator waveform is 180° out of phase, what would $y(t) + z(t)$ and $z(t) - y(t)$ be equal to?
- (c) If the 38 kHz oscillator waveform is out of phase somewhere between 0° and 180°, what would be the effect?

1.3 Consider the ideal bandpass filter shown in **Figure 1.2**. Given that the center frequency, ω_3 , is $2\pi \times 10^6$ rad/sec, and the bandwidth is $2\pi \times 10^4$ rad/sec, and that the magnitude gain across the passband is unity, determine the envelope of the unit impulse response of this filter: this would be the demodulated audio signal response to a unit impulse.

1.4 Using MATLAB, generate samples of the $x(t)$ of **(1.1)** for $N = 1$, $A_1 = 1$, $\mu_1 = 1$, $f_1(t) = \sin(2\pi 100 t)$, and $\omega_1 = 2\pi 2000$. Generate 1000 points for 20 ms of $x(t)$. Plot $x(t)$.

1.5 Using MATLAB, generate samples of the $x(t)$ of **(1.1)** for $N = 2$, $A_1 = 1$, $A_2 = 1$, $\mu_1 = 1$, $\mu_2 = 1$, $f_1(t) = \sin(2\pi 100 t)$, $f_2(t) = \sin(2\pi 200 t)$, $\omega_1 = 2\pi 2000$, and $\omega_2 = 2\pi 2500$. Generate 1000 points for 20 ms of $x(t)$. Plot $x(t)$. It is interesting to note that the two amplitude-modulated signals added together in this problem do not have overlapping frequency spectra and could easily be separated by appropriate filtering.

Clearly, *Case 2*, where $G(s) = s + \beta$, is a combination of a first-order and a second-order transfer function. First-order transfer functions were considered in **Section 2.8**. The contribution of $G(s)$ to the magnitude response of $|H(j\omega)|$ is independent of the polarity of β , however, the phase response, $h(t)$, and $h_u(t)$ do depend on the polarity, as well as the magnitude, of β .

Case 3: $G(s) = s^2 + \alpha s + \beta$ in (2.56)

In the case where $G(s) = s^2 + \alpha s + \beta$, then the contribution of $G(s)$ to $|H(j\omega)|$ and $\angle H(j\omega)$ will be closely related to that already considered above for *Case 1*. Note, however, that since $G(s)$ is in the numerator, its effect on the magnitude response will be the inverse of that considered for *Case 1* (the opposite polarity when the magnitude response in *dB* is considered). It will also have the opposite polarity effect on the phase response. Also note that the roots of $G(s)$, in general, are not restricted to the left-half plane. See the next section for a more-general consideration of cascaded transfer functions.

2.10 TRANSFER FUNCTIONS WITH ORDERS GREATER THAN TWO

The transfer function, $H(s)$, expressed in factored form as in (2.40), where all coefficients are real, consists of the product of first- and second-order terms. This may be expressed as follows:

$$H(s) = H_1(s) H_2(s) \cdots H_{N_1+N_2}(s), \quad (2.71)$$

where each $H_i(s)$ is either a first-order or second-order transfer function that conforms to the imposed constraints of **Section 2.6**.

Therefore,

$$|H(j\omega)| = |H_1(j\omega)| |H_2(j\omega)| \cdots |H_{N_1+N_2}(j\omega)|,$$

or

$$\begin{aligned} 20 \log |H(j\omega)| &= 20 \log |H_1(j\omega)| + 20 \log |H_2(j\omega)| \\ &+ \cdots + 20 \log |H_{N_1+N_2}(j\omega)|. \end{aligned}$$

The magnitude frequency response is the product of $N_1 + N_2$ magnitude responses (or the sum of $N_1 + N_2$ responses if in *dB*). When designing an analog filter, it is usually the overall magnitude response that is designed, as mentioned in the opening paragraph of **Section 2.7**. For convenience, at least when designing active analog filters as in **Chapter 11**, individual first- and second-order stages will be designed, and then the stages will be cascaded as in (2.71). It is interesting to note, as will be

seen in subsequent chapters, that individual first- and second-order stages of a well-designed analog filter, such as, for example, a Butterworth filter, will not be particularly good filters, yet cascaded together they form a well-designed filter. The magnitude frequency response of one stage will, in part, make up for the deficiencies of other stages in the filter.

The phase response of (2.71) may be expressed as follows:

$$\angle H(j\omega) = \angle H_1(j\omega) + \angle H_2(j\omega) + \dots + \angle H_{N_1+N_2}(j\omega) .$$

The phase response is the sum of the $N_1 + N_2$ phase responses of the individual stages. Each of these individual phase responses may be further separated into the sum of the phase response of the numerator minus the phase response of the denominator.

The filter unit impulse response may be expressed as

$$h(t) = h_1(t) * h_2(t) * \dots * h_{N_1+N_2}(t) , \tag{2.72}$$

and the filter unit step response may be expressed as

$$h_u(t) = h_{u_1}(t) * h_{u_2}(t) * \dots * h_{u_{N_1+N_2}}(t) . \tag{2.73}$$

See Section 2.13 below, for more discussion on the time-domain response of analog filters.

2.11 MINIMUM-PHASE TRANSFER FUNCTIONS

Very simply stated, a minimum-phase transfer function is one where there are no zeros in the right-half of the s plane. This section demonstrates why such transfer functions are denoted as *minimum-phase*.

Since any rational $H(s)$ may be expressed as in (2.40), it is sufficient to consider first- and second-order minimum-phase transfer functions; $H(s)$ is minimum-phase only if each $H_i(s)$ is minimum-phase.¹³

First-Order

Consider (2.50), and refer to (2.54) and (2.55). Note in (2.54) that the magnitude frequency response is independent of the polarity of Υ . Therefore, $\Upsilon = \pm\chi$, where χ is an arbitrary positive real number, will have identical magnitude frequency responses. However, from (2.55), note that the phase response is dependent on the polarity of Υ . If $\Upsilon = +\chi$, then

$$\angle H(j\omega) = \angle K + \arctan(\omega/\chi) - \arctan(\omega/\mu) , \tag{2.74}$$

¹³Assuming that each $H_i(s)$ is stable, then no pole in $H_k(s)$ could cancel a right-half plane zero in $H_m(s)$. Therefore $H(s)$ is minimum-phase only if each $H_i(s)$ is minimum-phase.

whereas if $\gamma = -\chi$, then

$$\angle H(j\omega) = \angle K - \arctan(\omega/\chi) - \arctan(\omega/\mu). \quad (2.75)$$

Since μ and χ are positive real values, as ω varies from 0 to ∞ , $\arctan(\omega/\mu)$ and $\arctan(\omega/\chi)$ are both positive, and vary from 0 to 90° (or $\pi/2$). If $\angle K = 0$, then $|\angle H(j\omega)|_{\max}$ for (2.74) will be 90° , whereas for (2.75) it will be 180° .¹⁴ Therefore, (2.74) represents *minimum* phase, compared to (2.75): γ is positive, and therefore the zero is in the left-half plane.

Second-Order

Consider *Case 2* for (2.56), and refer to (2.69) and (2.70). Note in (2.69) that the magnitude frequency response is independent of the polarity of β . Therefore, $\beta = \pm\chi$, where χ is an arbitrary positive real number, will have identical magnitude frequency responses. However, from (2.70), note that the phase response is dependent upon the polarity of β . From (2.59), and the condition of stability, it is clear that both ξ and λ must be positive values. Note, from (2.70), that as ω varies from 0 to ∞ , that $\angle[(\lambda - \omega^2) + j\xi\omega]$ varies from 0 to 180° . Note that the real part, $\lambda - \omega^2$, becomes negative for $\omega > \sqrt{\lambda}$, and therefore the phase angle shifts to the second quadrant. Therefore, if $\angle K = 0$, $\angle H(j\omega)|_{\omega=\infty}$ is -90° if β is positive (zero is in the left-half plane), and is -270° if β is negative (zero is in the right-half plane). Therefore, again, the zero in the left-half plane corresponds with the *minimum* phase.

Consider *Case 3* for (2.56), where $G(s) = s^2 + \alpha s + \beta$. If the roots of $G(s)$ are real, then the results will be very similar to *Case 2* discussed above. If the roots of $G(s)$ are complex, the real parts of those roots are $-\alpha/2$. Clearly, the magnitude frequency response is independent of the polarity of α . However, the phase response of $G(j\omega)$, $\angle[(\beta - \omega^2) + j\alpha\omega]$, is dependent on the polarity of α . If $\alpha > 0$, then as ω varies from 0 to ∞ , $\angle G(j\omega)$ will vary from 0 to $+180^\circ$, whereas if $\alpha < 0$, then $\angle G(j\omega)$ will vary from 0 to -180° . Therefore, if $\angle K = 0$, $\angle H(j\omega)|_{\omega=\infty}$ is 0 if α is positive (zeros in the left-half plane), and is -360° if α is negative (zeros in the right-half plane). Therefore, zeros in the left-half plane correspond with *minimum* phase.

2.12 ALL-PASS TRANSFER FUNCTIONS

An all-pass transfer function is one where the magnitude frequency response is constant for all frequencies but the phase is not. Such filters are used for phase compensation; they effect the phase response but not the magnitude response. For

¹⁴The actual maximum magnitude phase angle for (2.74) depends upon the values of χ and μ , but the maximum possible value is 90° ; the maximum value for (2.75) occurs at $\omega = \infty$, and is 180° .

example, a well-designed all-pass filter may, in part, correct for the nonlinear phase response of a Butterworth filter which has an acceptable magnitude frequency response.

The importance of the phase response may be demonstrated by considering the properties of ideal transmission. Suppose $x(t)$ is the input to an analog filter. While the filter may be designed to attenuate certain unwanted components in $x(t)$, that part of $x(t)$ that falls within the passband of the filter is desired to be passed with minimal effect on the waveshape of the signal. Suppose $x_p(t)$ is that part of $x(t)$ that falls within the passband of the filter, then the desired filter output would be $y(t) = K_p x_p(t - t_d)$, where K_p is a gain term (could be unity) and t_d is a time delay (a time delay of zero is not practical, and a small delay will not effect the waveshape). From the basic properties of Fourier transforms, it follows that the required filter frequency response, magnitude and phase, across the passband would be as follows:

$$H(j\omega) = K_p e^{-jt_d\omega}$$

Therefore, the magnitude frequency response should be constant across the passband and the phase response should be linear ($\angle H(j\omega) = -t_d\omega$). Therefore in applications where preserving the waveshape of signals is important, as in, as examples, radar, sonar, an oscilloscope amplifier, etc., the phase response is of special importance.

Since any all-pass transfer function $H(s)$ may be expressed as the product of first- and second-order transfer functions such as in (2.71), it is sufficient to consider first- and second-order all-pass transfer functions.¹⁵

First-Order

Consider (2.50) with $\gamma = -\mu$ and, for convenience, $K = -1$. From (2.54) it is easy to see that $|H(j\omega)| = 1$, or $20 \log |H(j\omega)| = 0 \text{ dB}$, for all ω . From (2.55), $\angle H(j\omega) = -180^\circ + \angle(-\mu + j\omega) - \angle(\mu + j\omega)$:

$$\begin{aligned} &0^\circ \text{ at } \omega = 0, \\ &-90^\circ \text{ at } \omega = \mu, \\ &\text{and } -180^\circ \text{ at } \omega = \infty. \end{aligned}$$

$$\begin{aligned} \text{If } K = 1, \angle H(j\omega) &= \angle(-\mu + j\omega) - \angle(\mu + j\omega): \\ &180^\circ \text{ at } \omega = 0, \\ &90^\circ \text{ at } \omega = \mu, \\ &\text{and } 0^\circ \text{ at } \omega = \infty. \end{aligned}$$

¹⁵Although it is possible to conceive of an all-pass $H(s)$ where there are $H_i(s)$ that are not all-pass, it is always possible to order the poles and zeros such that all $H_i(s)$ are all-pass.

Therefore, the only design parameter is μ , which determines the frequency at which the phase will be $\pm 90^\circ$. Note that the phase response monotonically decreases for $0 < \omega < \infty$.

Second-Order

Consider (2.56) with $G(s) = s^2 + \alpha s + \beta$ (i.e., Case 3), but with $\beta = \lambda$ and $\alpha = -\xi$. For convenience, let $K = 1$. It is easy to show that $|H(j\omega)| = 1$, or $20 \log |H(j\omega)| = 0 \text{ dB}$, for all ω . Similar to (2.70), $\angle H(j\omega) = \angle [(\lambda - \omega^2) - j\xi\omega] - \angle [(\lambda - \omega^2) + j\xi\omega]$:

$$\begin{aligned} & 0^\circ \text{ at } \omega = 0, \\ & -90^\circ \text{ at } \omega = -(\xi/2) + (1/2)\sqrt{\xi^2 + 4\lambda}, \\ & -180^\circ \text{ at } \omega = \sqrt{\lambda}, \\ & -270^\circ \text{ at } \omega = (\xi/2) + (1/2)\sqrt{\xi^2 + 4\lambda}, \\ & \text{and } -360^\circ \text{ at } \omega = \infty. \end{aligned}$$

If the definitions of (2.60) and (2.61) are used, then $\angle H(j\omega) = \angle [(\omega_o^2 - \omega^2) - j(\omega_o\omega)/Q] - \angle [(\omega_o^2 - \omega^2) + j(\omega_o\omega)/Q]$:

$$\begin{aligned} & 0^\circ \text{ at } \omega = 0, \\ & -90^\circ \text{ at } \omega = [\omega_o/(2Q)] [-1 + \sqrt{1 + 4Q^2}], \\ & -180^\circ \text{ at } \omega = \omega_o, \\ & -270^\circ \text{ at } \omega = [\omega_o/(2Q)] [1 + \sqrt{1 + 4Q^2}], \\ & \text{and } -360^\circ \text{ at } \omega = \infty. \end{aligned}$$

Note there are two design parameters: ω_o and Q (or λ and ξ). The frequency at which $\angle H(j\omega)$ is -180° can be set with ω_o (or λ), and how rapidly the phase changes on either side of ω_o (the phase response slope) can be set with Q (or ξ). Note that the phase monotonically decreases for $0 < \omega < \infty$.

2.13 TIME-DOMAIN RESPONSE

The time-domain response of first- and second-order transfer functions was briefly treated in Sections 2.8 and 2.9, respectively. The time-domain response of transfer functions with orders greater than two was briefly treated in Section 2.10: (2.72) and (2.73) are important general equations for the impulse response and the step response, respectively, of the overall transfer function. It should be noted that: (1) the effective time width of $h(t)$ will be longer than the longest $h_i(t)$, (2) whatever ringing there is associated with $h_i(t)$, it will appear in $h(t)$ and in $h_u(t)$,

(3) the shape of $h(t)$ is as dependent upon (or related to) $\angle H(j\omega)$ as it is upon $|H(j\omega)|$.

More generally, the filter response $y(t)$ may always be represented as the convolution of the filter input $x(t)$ and the filter unit impulse response $h(t)$:

$$y(t) = \int_0^t h(\tau) x(t-\tau) d\tau ,$$

where the lower limit on the integral of 0 is used since $h(t)$ is causal. To evaluate a particular filter response $y(t)$ requires a specified filter input $x(t)$. However, the other term in the convolution integral, $h(t)$, is independent of any specific $x(t)$,¹⁶ and is commonly evaluated and plotted as an indication of a filter's time-domain response characteristics. Since the unit step response, $h_u(t)$, is closely related to the unit impulse response,

$$h_u(t) = \int_0^t h(\tau) d\tau ,$$

the unit step response is also commonly evaluated and plotted as an indication of a filter's time-domain response characteristics.

2.14 PHASE DELAY AND GROUP DELAY

Consider the general transfer function of the form expressed in (2.39), which is repeated here for convenience:

$$H(s) = \frac{\sum_{k=0}^M b_k s^k}{\sum_{k=0}^N a_k s^k} . \tag{2.76}$$

The frequency response of (2.76) may be expressed as follows:

$$H(j\omega) = \frac{\sum_{k=0}^M b_k (j\omega)^k}{\sum_{k=0}^N a_k (j\omega)^k} = \frac{\sum_{k \text{ even}} b_k (-1)^{k/2} \omega^k + j \sum_{k \text{ odd}} b_k (-1)^{(k-1)/2} \omega^k}{\sum_{k \text{ even}} a_k (-1)^{k/2} \omega^k + j \sum_{k \text{ odd}} a_k (-1)^{(k-1)/2} \omega^k} . \tag{2.77}$$

Equation (2.77) may be expressed:

¹⁶Except, of course, $\delta(t)$.

$$H(j\omega) = R(\omega) + jI(\omega) = \frac{R_n(\omega) + jI_n(\omega)}{R_d(\omega) + jI_d(\omega)}, \quad (2.78)$$

where

$$R_n(\omega) = \sum_{k \text{ even}} b_k (-1)^{k/2} \omega^k,$$

$$I_n(\omega) = \sum_{k \text{ odd}} b_k (-1)^{(k-1)/2} \omega^k,$$

$$R_d(\omega) = \sum_{k \text{ even}} a_k (-1)^{k/2} \omega^k,$$

$$I_d(\omega) = \sum_{k \text{ odd}} a_k (-1)^{(k-1)/2} \omega^k,$$

$$R(\omega) = \frac{R_n(\omega)R_d(\omega) + I_n(\omega)I_d(\omega)}{R_d^2(\omega) + I_d^2(\omega)},$$

and

$$I(\omega) = \frac{R_d(\omega)I_n(\omega) - R_n(\omega)I_d(\omega)}{R_d^2(\omega) + I_d^2(\omega)}$$

That is, $R_n(\omega)$ is the real part of the numerator of (2.77), $I_n(\omega)$ is the imaginary part of the numerator of (2.77), $R_d(\omega)$ is the real part of the denominator, $I_d(\omega)$ is the imaginary part of the denominator, $R(\omega)$ is the real part of (2.77) and $I(\omega)$ is the imaginary part. Being careful to maintain the proper quadrant, the phase response, based on (2.78), is

$$\angle H(j\omega) = \arctan\left(\frac{I(\omega)}{R(\omega)}\right) = \arctan\left(\frac{R_d(\omega)I_n(\omega) - R_n(\omega)I_d(\omega)}{R_n(\omega)R_d(\omega) + I_n(\omega)I_d(\omega)}\right). \quad (2.79)$$

This equation, (2.79), can be useful in evaluating the *phase delay* and the *group delay* of an analog filter, and is referred to below.

Phase Delay

The *phase delay*, $t_{pd}(\omega)$, for a filter is defined as follows:

$$t_{pd}(\omega) = -\frac{\angle H(j\omega)}{\omega}. \quad (2.80)$$

From Section 2.10, recall that

$$\angle H(j\omega) = \sum_{k=1}^N \angle H_k(j\omega) .$$

Therefore, phase delay is additive:

$$t_{pd}(\omega) = \sum_{k=1}^N t_{pd}^{(k)}(\omega) ,$$

where $t_{pd}^{(k)}(\omega)$ is the phase delay of the k -th stage.

For any linear filter, a sinusoid on the input results in a sinusoid on the output. That is, if $x(t) = \cos(\omega_o t)$ (the filter input), then the output, $y(t)$, may be expressed as

$$\begin{aligned} y(t) &= |H(j\omega_o)| \cos[\omega_o t + \angle H(j\omega_o)] \\ &= |H(j\omega_o)| \cos[\omega_o \{t + \angle H(j\omega_o)/\omega_o\}] . \end{aligned}$$

Therefore, the phase shift as the signal propagates through the filter is $\angle H(j\omega_o)$, which corresponds to a time shift of $\angle H(j\omega_o)/\omega_o$. Since it is generally assumed that $\angle H(j\omega_o)$ will be negative, the time delay would be $-\angle H(j\omega_o)/\omega_o$. The frequency ω_o could be any value of ω , therefore (2.80) is defined as it is. Briefly stated, (2.80) is the time delay of the output relative to the input that a sinusoid with frequency ω will have. It is referred to as *phase delay* because of its direct relationship to the phase.

Group Delay

The *group delay*, $t_{gd}(\omega)$, for a filter is defined as follows:

$$t_{gd}(\omega) = - \frac{d \angle H(j\omega)}{d\omega} . \tag{2.81}$$

As is the case for phase delay, group delay is additive:

$$t_{gd}(\omega) = \sum_{k=1}^N t_{gd}^{(k)}(\omega) ,$$

where $t_{gd}^{(k)}(\omega)$ is the group delay of the k -th stage.

Applying (2.81) to (2.78) and (2.79):

$$t_{gd}(\omega) = - \frac{d \{ \arctan[I(\omega)/R(\omega)] \}}{d\omega} = \frac{I(\omega) \frac{dR(\omega)}{d\omega} - R(\omega) \frac{dI(\omega)}{d\omega}}{R^2(\omega) + I^2(\omega)} . \tag{2.82}$$

Note that if a filter should have an ideal linear phase response, as indicated in Section 2.12, i.e., $\angle H(j\omega) = -t_d \omega$, then, according to (2.80) and (2.81), $t_{pd}(\omega) = t_{gd}(\omega) = t_d$. That is, for the ideal linear phase response case, phase delay

and group delay would be identical. However, in general, phase delay and group delay differ, and both definitions are useful. Consider the following illustration.

An Illustration

Consider the following sinusoidally modulated sine wave:

$$\begin{aligned} x(t) &= [1 + \cos(\omega_m t + \theta_m)] \cos(\omega_o t + \theta_o) \\ &= \cos(\omega_o t + \theta_o) + \frac{1}{2} \cos[(\omega_o - \omega_m)t + \theta_o - \theta_m] \\ &\quad + \frac{1}{2} \cos[(\omega_o + \omega_m)t + \theta_o + \theta_m] . \end{aligned} \quad (2.83)$$

Recall that (2.83) is an amplitude modulated (AM) carrier (see, e.g., Stremmer (1990)). The carrier frequency is ω_o , the arbitrary carrier phase is θ_o , the frequency of the modulating sinusoid is ω_m , and the arbitrary phase of the modulating sinusoid is θ_m . As is usually the case, it is assumed that $\omega_o \gg \omega_m$. For convenience, the modulation factor is unity; $x(t)$ has a percentage modulation of 100%. Recall that the frequency bandwidth of $x(t)$ is $2 \times \omega_m$; the passband extends from $\omega_o - \omega_m$ to $\omega_o + \omega_m$. In fact, there are only three frequencies (on each side of the double-sided spectrum) present in $x(t)$, which is clear from the second line of (2.83): $\omega_o - \omega_m$, ω_o , and $\omega_o + \omega_m$. Suppose that $x(t)$ is the input to a filter and is within the magnitude passband of the filter such that there are no changes in frequency-component magnitudes. In this illustration, for simplicity in notation, suppose that $\angle H(j\omega_o) = \phi_o$ and that the phase response is locally linear. That is, suppose that across the frequency range of $\omega_o - \omega_m$ through $\omega_o + \omega_m$ the phase response may be expressed as:

$$\angle H(j\omega) = -\alpha \omega + \beta, \quad \omega_o - \omega_m \leq \omega \leq \omega_o + \omega_m .$$

Therefore,

$$\angle H(j[\omega_o - \omega_m]) = -\alpha(\omega_o - \omega_m) + \beta = \phi_o + \alpha \omega_m ,$$

and

$$\angle H(j[\omega_o + \omega_m]) = \phi_o - \alpha \omega_m .$$

Applying (2.80):

$$t_{pd}(\omega_o - \omega_m) = -(\phi_o + \alpha \omega_m) / (\omega_o - \omega_m) ,$$

$$t_{pd}(\omega_o) = -\phi_o / \omega_o ,$$

and

$$t_{pd}(\omega_o + \omega_m) = -(\phi_o - \alpha \omega_m) / (\omega_o + \omega_m) .$$

It is therefore observed that $t_{pd}(\omega)$ differs, in general, for the three frequency components. Applying (2.81):

$$t_{gd}(\omega) = \alpha ,$$

where it is noted that the group delay is the same for all three frequency components. To illustrate further, the filter output may be expressed as follows:

$$\begin{aligned} y(t) &= \cos(\omega_o t + \theta_o + \phi_o) + \frac{1}{2} \cos[(\omega_o - \omega_m)t + \theta_o + \phi_o + \alpha\omega_m - \theta_m] \\ &\quad + \frac{1}{2} \cos[(\omega_o + \omega_m)t + \theta_o + \phi_o - \alpha\omega_m + \theta_m] \\ &= [1 + \cos(\omega_m t + \theta_m - \alpha\omega_m)] \cos(\omega_o t + \theta_o + \phi_o) . \end{aligned} \tag{2.84}$$

Note that the *time delay for the carrier* is the *phase delay* evaluated at $\omega = \omega_o$, $t_{pd}(\omega_o) = -\phi_o/\omega_o$ (compare (2.84) with (2.83)). Also note that the *time delay of the envelope*,¹⁷ which also may be observed by comparing (2.84) with (2.83), is $\alpha\omega_m/\omega_m = \alpha$, which is identical with the *group delay*. Recall that the parameter ϕ_o is the phase shift of the filter frequency response at $\omega = \omega_o$, and the parameter α is the negative of the slope of the phase response at $\omega = \omega_o$. In general, of course, $\alpha \neq -\phi_o/\omega_o$: the group delay could be greater than the phase delay, or vice versa.

The above is illustrated in **Figure 2.37**, where the carrier at $\omega = \omega_o$ has a time delay that differs from the group, or envelope, delay. In this illustration the group delay is greater than the phase delay. If one were to measure the group delay and the phase delay for an analog filter, in a laboratory situation, with a filter input similar to (2.83), it can be seen that there would be ambiguity in the measurements. It is assumed in **Figure 2.37** that numerical values of the group delay and the phase delay are known *a priori*, and the waveforms in the figure merely illustrate the two delays. Otherwise the phase delay, t_{pd} , could be taken as slightly greater than t_{gd} , instead of less than t_{gd} , as illustrated. Therefore, measurements of group delay and phase delay with an input similar to (2.83) would require some *a priori* knowledge of at least approximate values for the two delays. \square

Briefly, the phase delay at a given ω is simply the negative of the phase angle at that ω divided by ω . The group delay at that same given ω is the negative of the phase response slope. In general, these two values are not the same. However, as mentioned above, they are the same for the ideal linear phase response case. Therefore, if a filter has a phase response that approximates linear phase over some range of ω with a zero intercept, i.e., the phase response may be approximated as $\angle H(j\omega) = -\alpha\omega$ over that range of ω , it should be expected that the phase delay

¹⁷*Envelope delay* is another name for *group delay*. The term “group” delay is used, since if the modulating envelope is of short time duration, such as in a transmitted radar modulated pulse, the envelope delay is indeed the delay of the “group,” or the pulse taken as a whole, whereas the phase delay would be the delay associated with the carrier “inside” the “group.”

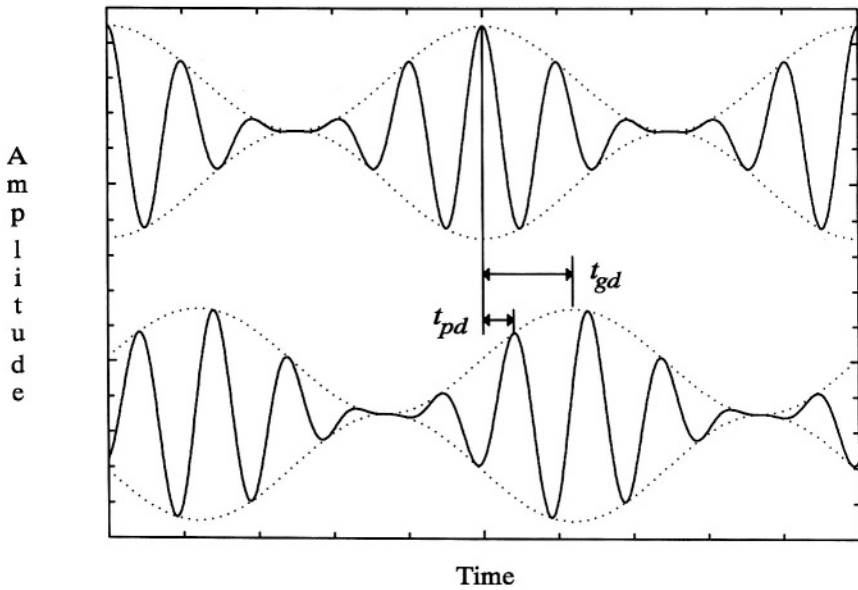


Figure 2.37 Illustration of the difference between phase delay and group delay. The top waveform is an example of (2.83) and the bottom waveform is the corresponding (2.84).

and group delay should be similar for that frequency range. If however, such is not the case, then significant differences may exist between the phase delay and the group delay.

Example 2.23

Given that

$$H(s) = \frac{10^5}{s + 10^5} ,$$

then, from (2.78) and (2.79), $R_d(\omega) = 10^5$, $I_d(\omega) = \omega$, $R_n(\omega) = 10^5$, $I_n(\omega) = 0$, $R(\omega) = 10^{10}/(10^{10} + \omega^2)$, $I(\omega) = -10^5\omega/(10^{10} + \omega^2)$, and $\angle H(j\omega) = -\arctan(\omega/10^5)$. Therefore,

$$t_{pd}(\omega) = \frac{\arctan(\omega/10^5)}{\omega} . \tag{2.85}$$

Using (2.79) in (2.82),

$$t_{gd}(\omega) = \frac{10^5}{10^{10} + \omega^2} . \quad (2.86)$$

Suppose that the filter input, $x(t)$, is as expressed in (2.83) with $\omega_o = 10^5$, $\omega_m = 100$, and $\theta_o = \theta_m = 0$, then, the phase delay and the group delay may be computed at $\omega = \omega_o = 10^5$ from (2.85) and (2.86) to be

$$t_{pd}(\omega_o) = 7.854 \mu s ,$$

and

$$t_{gd}(\omega_o) = 5 \mu s .$$

Note that across the frequency range of $10^5 - 100$ through $10^5 + 100$ that both the phase delay and the group delay varies little from the two values indicated above. In fact, $t_{pd}(\omega_o - 100) = 7.857 \mu s$, $t_{pd}(\omega_o + 100) = 7.851 \mu s$, $t_{gd}(\omega_o - 100) = 5.005 \mu s$, and $t_{gd}(\omega_o + 100) = 4.995 \mu s$. Therefore the filter output, $y(t)$, may be approximately expressed as follows:

$$y(t) \cong \{1 + \cos(\omega_m [t - 5 \mu s])\} \cos(\omega_o [t - 7.854 \mu s]) . \quad \square$$

Example 2.24

Given that

$$H(s) = \frac{10^{10}}{s^2 + \sqrt{2} 10^5 s + 10^{10}} ,$$

then, from (2.78) and (2.79), $R_d(\omega) = 10^{10} - \omega^2$, $I_d(\omega) = \sqrt{2} 10^5 \omega$, $R_n(\omega) = 10^{10}$, $I_n(\omega) = 0$,

$$R(\omega) = \frac{10^{10} (10^{10} - \omega^2)}{10^{20} + \omega^4} ,$$

$$I(\omega) = -\frac{\sqrt{2} \times 10^{15} \omega}{10^{20} + \omega^4} ,$$

and

$$\angle H(j\omega) = -\arctan\left(\frac{\sqrt{2} \times 10^5 \omega}{10^{10} - \omega^2}\right) .$$

Therefore,

$$t_{pd}(\omega) = \frac{\arctan\left(\frac{\sqrt{2} \times 10^5 \omega}{10^{10} - \omega^2}\right)}{\omega} . \quad (2.87)$$

Using (2.79) in (2.82),

$$t_{gd}(\omega) = \sqrt{2} \times 10^5 \frac{10^{30} + 10^{20} \omega^2 + 10^{10} \omega^4 + \omega^6}{(10^{20} + \omega^4)^2} \quad (2.88)$$

Using (2.87) and (2.88), the phase delay and the group delay at $\omega = 10^5$ are as follows: $t_{pd}(10^5) = 15.71 \mu s$, and $t_{gd}(10^5) = 14.14 \mu s$. \square

Summary of Phase Delay and Group Delay

- Phase delay is defined by (2.80). It is referred to as *phase delay* because of its direct relationship to the phase.
- Group delay is defined by (2.81). *Envelope delay* is another name for *group delay*. *Group delay* is the delay of the *group*, or the *envelope*, as opposed to the delay of the carrier, which is the *phase delay*.
- If a filter or system has an ideal linear phase response, then *group delay* and *phase delay* will be identical.
- For filters that have a phase response that approximates linear phase, plots of the *group delay* and the *phase delay* will be similar. As the phase response deviates further from linear phase, plots of the *group delay* and the *phase delay* will differ to a greater extent.

2.15 HILBERT TRANSFORM RELATIONS

From the above, primarily from **Sections 2.1, 2.6, and 2.7**, it should be clear that the magnitude frequency response and the phase response are not independent; they both depend upon the poles and the zeros of the transfer function. In this section

Hilbert transform relations, first between the real and imaginary parts of $H(j\omega)$, and then between the magnitude frequency response and the phase response, are developed.

The impulse response of an analog filter, which is constrained to be causal, may be expressed as follows:

$$h(t) = k_o \delta(t) + h_+(t) u(t) , \quad (2.89)$$

where k_o will be zero (no impulse) for lowpass and bandpass filters, but will be non-zero for highpass and bandstop filters,¹⁸

$$h_+(t) = \begin{cases} h(t), & t > 0^+ \\ 0, & t < 0^+ \end{cases} ,$$

and $u(t)$ is the unit step function:

$$u(t) = \begin{cases} 1, & t > 0 \\ 0, & t < 0 \end{cases} .$$

Multiplying $h_+(t)$ by $u(t)$ in (2.89), while perhaps appearing to add unnecessary redundancy, is very useful in the development that follows immediately.

Let $H(j\omega)$ be expressed in terms of real and imaginary parts:

$$H(j\omega) = R(\omega) + j I(\omega) , \quad (2.90)$$

where $R(\omega)$ is the real part and $I(\omega)$ is the imaginary part of $H(j\omega)$, and let $H_+(j\omega)$ be the Fourier transform of $h_+(t)$. Let $h(t) = h_e(t) + h_o(t)$, where $h_e(t)$ is the even part, and $h_o(t)$ is the odd part of $h(t)$. Note, since $h(t)$ is causal, that for $t < 0^-$, $h_e(t) = -h_o(t)$, and for $t > 0^+$, $h_e(t) = h_o(t)$. Note that if there is an impulse ($k_o \neq 0$), that it will be included in $h_e(t)$, since it is even, but that it will not appear in $h_o(t)$. From basic properties of Fourier transforms,

$$\mathcal{F}\{h_e(t)\} = R(\omega) = k_o + R_k(\omega) ,$$

where $R_k(\omega)$ is the Fourier transform of $h_e(t) - k_o \delta(t)$. Making use of the frequency convolution property of Fourier transforms:

$$h_+(t) u(t) \iff \frac{1}{2\pi} [H_+(j\omega) * U(j\omega)] , \quad (2.91)$$

where

¹⁸Refer to the unit impulse response graphs in **Section 2.1** for ideal filters. In terms of the presence, or lack thereof, of impulses in the impulse response, the graphs in **Section 2.1** accurately reflect practical, as well as ideal filters. Note that if the magnitude frequency response of the lowpass filter (or the lowpass prototype for the bandpass case) is not zero at $\omega = \infty$, such as may be the case for a Chebyshev Type II or an elliptic filter, there would theoretically be an impulse in (2.89), but the value of k_o would be very small and insignificant.

$$U(j\omega) = \mathcal{F}\{u(t)\} = \pi \delta(\omega) + \frac{1}{j\omega} .$$

Combining (2.89), (2.90) and (2.91) (Papoulis, 1977):

$$\begin{aligned} H(j\omega) &= R(\omega) + jI(\omega) = k_o + \frac{1}{2\pi} [H_c(j\omega) * U(j\omega)] \\ &= k_o + \frac{1}{2\pi} [R_k(\omega) + jI(\omega)] * [\pi \delta(\omega) + \frac{1}{j\omega}] \\ &= k_o + \frac{1}{2\pi} [R_k(\omega) * \pi \delta(\omega) + I(\omega) * \frac{1}{\omega}] \\ &\quad + \frac{j}{2\pi} [I(\omega) * \pi \delta(\omega) - R_k(\omega) * \frac{1}{\omega}] . \end{aligned} \tag{2.92}$$

Equating real and imaginary parts on the two sides of (2.92), simplifying, and making use of $R_k(\omega) = R(\omega) - k_o$, results in the following Hilbert transform relations for real and imaginary parts:

$$R(\omega) = k_o + \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{I(\zeta)}{\omega - \zeta} d\zeta , \tag{2.93}$$

and

$$I(\omega) = - \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{R(\zeta)}{\omega - \zeta} d\zeta . \tag{2.94}$$

It needs to be stressed that (2.93) and (2.94) are valid Hilbert transform relations only for causal systems, as the development made use of causality. In fact, the converse is true: any frequency response where the real and imaginary parts are related by (2.93) and (2.94) must be a causal system (Papoulis, 1962).

Any integral of the form shown in (2.93) and (2.94) is called a Hilbert transform. That is, if

$$g(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(\zeta)}{t - \zeta} d\zeta ,$$

then $g(t)$ is said to be the Hilbert transform of $f(t)$. An interesting aside is that if $z(t) = f(t) + jg(t)$, where $g(t)$ is the Hilbert transform of $f(t)$, then the Fourier transform of $z(t)$ will be one-sided, which is the mathematical basis for the theoretical study of single side-band radio communications (Stremler, 1990). The one-sided Fourier transform of $z(t)$ may be shown by noting that $g(t)$ is the convolution of $f(t)$ and $1/(\pi t)$. Therefore, $G(j\omega) = F(j\omega) \times [-j \operatorname{sgn}(\omega)]$, where $\operatorname{sgn}(\omega)$ is the sign function:

$$\text{sgn}(\omega) = \begin{cases} 1, & \omega > 0 \\ -1, & \omega < 0 \end{cases}.$$

Therefore, the Fourier transform of $z(t)$ is one-sided:

$$Z(j\omega) = \begin{cases} 2F(j\omega), & \omega > 0 \\ 0, & \omega < 0 \end{cases}.$$

Example 2.25

Given that $R(\omega) = \frac{100}{100 + \omega^2}$, then, the integrand of (2.94) may be expressed as follows:

$$\frac{100}{(100 + \zeta^2)(\omega - \zeta)} = \frac{100}{100 + \omega^2} \left[\frac{\zeta + \omega}{100 + \zeta^2} + \frac{1}{\omega - \zeta} \right]. \tag{2.95}$$

Using (2.95) in (2.94) results in

$$I(\omega) = -\frac{10\omega}{100 + \omega^2}.$$

Therefore,

$$\begin{aligned} H(j\omega) &= R(\omega) + jI(\omega) = \frac{100}{100 + \omega^2} - j\frac{10\omega}{100 + \omega^2} \\ &= \frac{10(10 - j\omega)}{(10 + j\omega)(10 - j\omega)} = \frac{10}{10 + j\omega}. \end{aligned} \tag{2.96}$$

From (2.96),

$$H(s) = \frac{10}{s + 10},$$

which is recognized as a lowpass filter with a 3 dB cutoff frequency of 10. □

Example 2.26

Given that

$$R(\omega) = \frac{\omega^2}{100 + \omega^2},$$

then, the integrand of (2.94) may be expressed as follows:

$$\frac{\zeta^2}{(100 + \zeta^2)(\omega - \zeta)} = \frac{1}{100 + \omega^2} \left[\frac{-100(\zeta + \omega)}{100 + \zeta^2} + \frac{\omega^2}{\omega - \zeta} \right]. \tag{2.97}$$

Using (2.97) in (2.94) results in

$$I(\omega) = \frac{10 \omega}{100 + \omega^2} .$$

Therefore,

$$\begin{aligned} H(j\omega) &= R(\omega) + jI(\omega) = \frac{\omega^2}{100 + \omega^2} + j \frac{10 \omega}{100 + \omega^2} \\ &= \frac{j\omega(10 - j\omega)}{(10 + j\omega)(10 - j\omega)} = \frac{j\omega}{10 + j\omega} . \end{aligned} \quad (2.98)$$

From (2.98),

$$H(s) = \frac{s}{s + 10} , \quad (2.99)$$

which is recognized as a highpass filter with a 3 dB cutoff frequency of 10. The inverse Laplace transform of (2.99) includes an impulse with a weight of unity. Note that $R(\omega)$ could be expressed as follows:

$$R(\omega) = \frac{\omega^2}{100 + \omega^2} = 1 - \frac{100}{100 + \omega^2} ,$$

and therefore $k_o = 1$, and

$$R_k(\omega) = \frac{-100}{100 + \omega^2} . \quad \square$$

The above Hilbert transform relations for the real and imaginary parts of $H(j\omega)$ place no restrictions on the phase; the phase response need not be minimum phase; there may be zeros in the right-half plane. However, the Hilbert transform relations for the magnitude frequency response and the phase response restrict the phase response to be minimum phase. Mathematical reasons for this restriction are given below, however, given there are finite-valued complex zeros with non-zero real parts, it is intuitive that the phase response is not unique for a given magnitude response. For example, the magnitude response for an $H(s)$ that has an $s + 100$ term in the numerator is identical if the term is replaced by $s - 100$, but the phase response would differ. More generally, $H(s)$ multiplied by any all-pass transfer function (see Section 2.12) will have the same magnitude frequency response as $H(j\omega)$ itself, but the phase response will differ.

As was done in Section 2.2, on page 33, let the analog filter frequency response be expressed as follows (Lam, 1979; Papoulis, 1962):

$$H(j\omega) = e^{-[\rho(\omega) + j\theta(\omega)]} , \quad (2.100)$$

and recall that $\rho(\omega)$ is called the *attenuation* or *loss function*, and $\theta(\omega)$ is called the *phase function*. The negative of the natural logarithm of (2.100) is a complex function of ω , and may be, therefore, expressed as a new frequency response, denoted as $H_{ln}(j\omega)$:

$$H_{ln}(j\omega) = -\ln\{H(j\omega)\} = \rho(\omega) + j\theta(\omega) .$$

The real and imaginary parts of $H_{ln}(j\omega)$, $\rho(\omega)$ and $\theta(\omega)$, respectively, are related by (2.93) and (2.94): however $H(j\omega)$ is constrained to be minimum phase. The reason why is that every pole and zero in $H(s)$ is a pole in $H_{ln}(s)$: $|\ln\{x\}| = \infty$, for $x = \infty$, and for $x = 0$. In order for the $j\omega$ axis to be in the region of convergence of $H_{ln}(s)$ and the corresponding impulse response to be causal there can be no right-half plane poles in $H_{ln}(s)$: $H(j\omega)$ is constrained to be minimum phase. With $H(j\omega)$ minimum phase:

$$\rho(\omega) = \rho(\infty) + \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\theta(\zeta)}{\omega - \zeta} d\zeta , \tag{2.101}$$

and

$$\theta(\omega) = -\frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\rho(\zeta)}{\omega - \zeta} d\zeta . \tag{2.102}$$

Note that the constant in (2.101) is $\rho(\infty)$ rather than k_o : $R(\infty) = k_o$. However, $\rho(\infty)$, while useful for highpass and bandstop filters, is not in the best form for lowpass and bandpass filters. Another useful form for the attenuation function may be obtained as follows. From (2.101),

$$\rho(0) = \rho(\infty) - \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\theta(\zeta)}{\zeta} d\zeta ,$$

from which

$$\rho(\infty) = \rho(0) + \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\theta(\zeta)}{\zeta} d\zeta . \tag{2.103}$$

Substituting (2.103) into (2.101) and simplifying results in:

$$\rho(\omega) = \rho(0) + \frac{\omega}{\pi} \int_{-\infty}^{\infty} \frac{\theta(\zeta)}{\zeta(\omega - \zeta)} d\zeta . \tag{2.104}$$

Example 2.27

Consider the ideal lowpass filter shown in **Figure 2.2**, except that, for computational reasons, the response for $\omega > \omega_c$ is $\epsilon > 0$ rather than 0.¹⁹ That is:

$$|H_{LP}(j\omega)| = \begin{cases} 1, & |\omega| < \omega_c \\ \epsilon, & |\omega| > \omega_c \end{cases},$$

where ϵ is a small positive number. The purpose of this example is to compute the corresponding phase response, given the modified ideal lowpass magnitude response above, and assuming $H(s)$ is minimum phase. This example is computationally interesting and relatively easy, but as discussed in **Section 2.1**, the transfer function is not realizable.

The attenuation function, $\rho(\omega)$, is as follows:

$$\rho(\omega) = \begin{cases} 0, & |\omega| < \omega_c \\ -\ln(\epsilon), & |\omega| > \omega_c \end{cases},$$

and

$$\begin{aligned} \theta(\omega) &= \frac{1}{\pi} \int_{-\infty}^{-\omega_c} \frac{\ln(\epsilon)}{\omega - \zeta} d\zeta + \frac{1}{\pi} \int_{\omega_c}^{\infty} \frac{\ln(\epsilon)}{\omega - \zeta} d\zeta \\ &= \frac{2\omega \ln(\epsilon)}{\pi} \int_{\omega_c}^{\infty} \frac{1}{\omega^2 - \zeta^2} d\zeta \\ &= \frac{\ln(\epsilon)}{\pi} \ln \left| \frac{\omega - \omega_c}{\omega + \omega_c} \right|. \end{aligned} \tag{2.105}$$

The phase response is the negative of (2.105):

$$\angle H_{LP}(j\omega) = -\frac{\ln(\epsilon)}{\pi} \ln \left| \frac{\omega - \omega_c}{\omega + \omega_c} \right|. \tag{2.106}$$

Note that the phase response has odd symmetry, and that the phase is $\mp \infty$ at $\omega = \pm \omega_c$. Also note that in the limit as ϵ goes to 0, the phase response approaches $-\infty$ for all $\omega > 0$.

The group delay for this filter may be obtained by applying (2.81) to (2.106) resulting in

¹⁹This example is adapted from (Lindquist, 1977) and (Papoulis, 1962).

$$t_{gd}(\omega) = \frac{2 \ln(\epsilon)}{\pi} \frac{\omega_c}{\omega^2 - \omega_c^2} \tag{2.107}$$

Note that the group delay goes to infinity at $\omega = \omega_c$, and reverses polarity for frequencies beyond that: another indication that the transfer function is not realizable. Note also that the group delay approaches infinity at DC as ϵ approaches zero. The phase response and the group delay, (2.106) and (2.107), are plotted in Figures 2.38 and 2.39 for $\omega_c = 100$ and $\epsilon = 0.001$. \square

Example 2.28

Consider the following phase function, which is linear over a given range of frequencies:²⁰

$$\theta(\omega) = \begin{cases} t_d \omega & |\omega| < \omega_c \\ t_d \omega_c & \omega > \omega_c \\ -t_d \omega_c & \omega < -\omega_c \end{cases} \tag{2.108}$$

The corresponding attenuation function may be found using (2.104):

$$\begin{aligned} \rho(\omega) &= \rho(0) + \frac{\omega t_d}{\pi} \int_{-\omega_c}^{\omega_c} \frac{1}{\omega - \zeta} d\zeta + \frac{2 t_d \omega^2 \omega_c}{\pi} \int_{\omega_c}^{\infty} \frac{1}{\zeta(\omega^2 - \zeta^2)} d\zeta \\ &= \rho(0) + \frac{\omega t_d}{\pi} \ln \left| \frac{\omega + \omega_c}{\omega - \omega_c} \right| + \frac{\omega_c t_d}{\pi} \ln \left| 1 - \left(\frac{\omega}{\omega_c} \right)^2 \right|. \end{aligned} \tag{2.109}$$

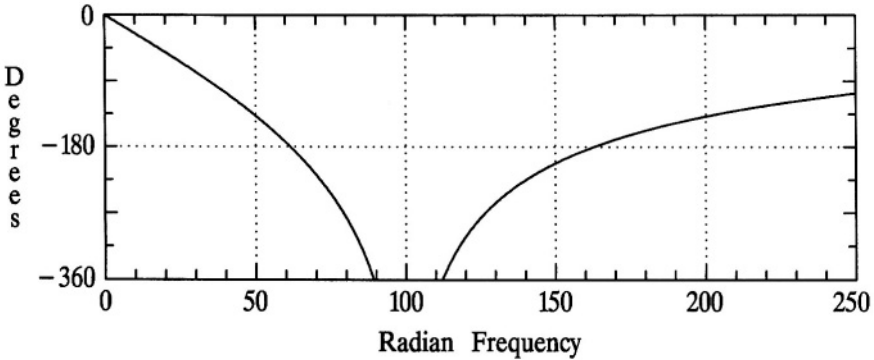


Figure 2.38 The phase response, $\angle H_{LP}(j\omega)$, for Example 2.27, with $\omega_c = 100$ and $\epsilon = 0.001$: a plot of (2.106).

²⁰This example is adapted from (Lam, 1979) and (Papoulis, 1962).

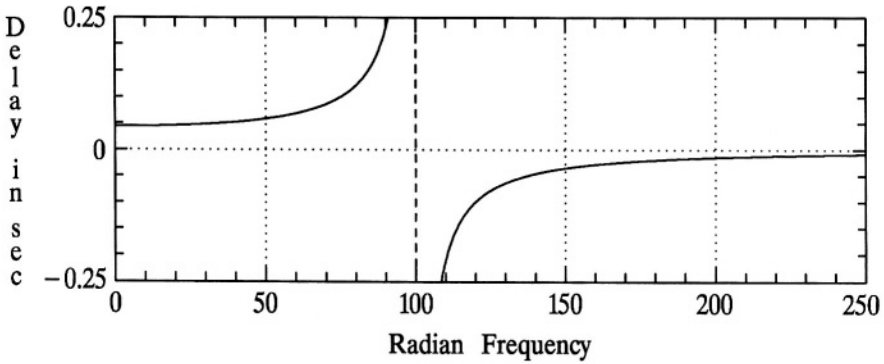


Figure 2.39 The group delay, $t_{gd}(\omega)$, for *Example 2.27*, with $\omega_c = 100$ and $\epsilon = 0.001$: a plot of (2.107).

The corresponding phase response and magnitude frequency response for (2.108) and (2.109) with $\rho(0) = 0$, $\omega_c = 1$, and $t_d = \pi$ are shown in **Figures 2.40** and **2.41**, respectively. Note that the group delay is a constant for this example, $t_{gd}(\omega) = t_d$ over the frequency range of $-\omega_c < \omega < \omega_c$, and is also a constant of zero outside that range. □

Obtaining the phase response from the magnitude frequency response for practical filters is very difficult to do using Hilbert transform relations. A more practical approach for practical filters would be to use the *Analog Filter Design*

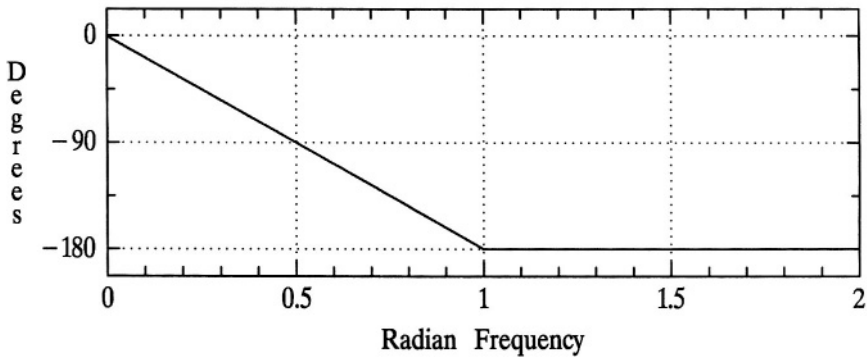


Figure 2.40 The phase response, $\angle H(j\omega)$, for *Example 2.28*, with $\omega_c = 1$ and $t_d = \pi$: a plot of the negative of (2.108).

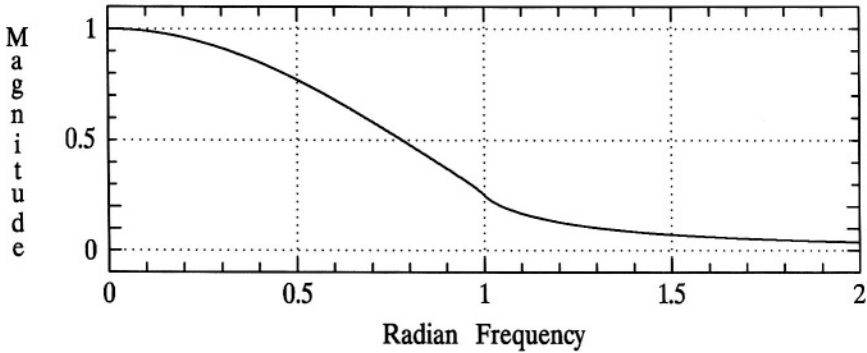


Figure 2.41 The magnitude frequency response, $|H(j\omega)|$, for *Example 2.28*, with $\omega_c = 1$, $t_d = \pi$ and $\rho(0) = 0$, corresponding to (2.109): $|H(j\omega)| = \exp[-\rho(\omega)]$.

Theorem and the method developed in **Section 2.7**: given the magnitude-squared frequency response, obtain the minimum-phase transfer function $H(s)$,²¹ from which $\angle H(j\omega)$ is readily found. For example, the simple one-pole lowpass filter of *Example 2.18*, on page 62, beginning with the magnitude-squared frequency response of $1/(\omega^2 + 1)$ readily yields the $H(s)$ of $1/(s + 1)$, from which $\angle H(j\omega) = -\arctan(\omega)$. However, for this simple problem, $\rho(\omega) = (1/2) \ln(\omega^2 + 1)$, which is difficult to evaluate in (2.102) to find the phase.

Therefore, the Hilbert transform method is of primary value in theoretical analysis, such as done in the above examples. In *Example 2.27*, as indicated in the example, the results give further evidence that the ideal lowpass response can not be realized, and can therefore only be approximated. Also, *Example 2.28* yields the magnitude frequency response for a desired linear-phase response over a given frequency range. The theoretical result suggests that to approximate the linear phase response, the resulting magnitude frequency response must be approximated. As will be seen in **Chapter 7**, Bessel filters, indirectly, do approximate this response.

²¹ Actually, as noted in **Section 2.7**, the resultant $H(s)$ is not restricted to be minimum phase. Therefore, the procedure can identify various phase realizations, assuming there are finite-valued transfer function zeros with non-zero real parts.

Summary of Hilbert Transform Relations

The most important concepts and results of this section are as follows:

- Given the imaginary part of $H(j\omega)$, denoted $I(\omega)$, the real part of $H(j\omega)$ is expressed by (2.93).
- Given the real part of $H(j\omega)$, denoted $R(\omega)$, the imaginary part of $H(j\omega)$ is expressed by (2.94).
- The Hilbert transform relations for real and imaginary parts, as expressed by (2.93) and (2.94), are not restricted to minimum-phase transfer functions.
- Given the attenuation function, $\rho(\omega) = -\ln |H(j\omega)|$, the phase function, $\theta(\omega)$, is expressed by (2.102).
- Given the phase function, $\theta(\omega) = -\angle H(j\omega)$, the attenuation function, $\rho(\omega)$, is expressed by (2.104).
- The Hilbert transform relations for the attenuation function and the phase function, as expressed by (2.102) and (2.104), are restricted to minimum-phase transfer functions.
- If the magnitude-squared frequency response is expressed as a polynomial in ω over a polynomial in ω , satisfying the *Analog Filter Design Theorem*, the method developed in Section 2.7 may be used to determine the phase response.

2.16 FREQUENCY SCALING

The general form of a rational transfer function has been given in (2.39), and when s is replaced by $j\omega$, by (2.42). Let k_f be a frequency scaling factor, where k_f is real and positive, and let $H^{(fs)}(j\omega)$ be a frequency scaled version of $H(j\omega)$.

Let all variables, functions, and coefficients with a superscript of (fs) be frequency-scaled values, and those without the superscript be associated with $H(j\omega)$. Making use of (2.42):

$$H^{(fs)}(j\omega) = H(j\omega) \Big|_{\omega \rightarrow \omega/k_f} = \frac{\sum_{k=0}^M b_k (j\omega/k_f)^k}{\sum_{k=0}^N a_k (j\omega/k_f)^k}, \quad (2.110)$$

where $H(j\omega)$ is the original, non-frequency scaled frequency response. Or, in general, starting with (2.39), or replacing $j\omega$ with s in (2.110):

$$H^{(fs)}(s) = H(s) \Big|_{s \rightarrow s/k_f} = \frac{\sum_{k=0}^M b_k (s/k_f)^k}{\sum_{k=0}^N a_k (s/k_f)^k}. \quad (2.111)$$

Therefore, relating the coefficients of $H^{(fs)}(s)$ in (2.111) to $H(s)$:

$$b_k^{(fs)} = b_k/k_f^k, \quad a_k^{(fs)} = a_k/k_f^k,$$

and so

$$H^{(fs)}(s) = \frac{\sum_{k=0}^M b_k^{(fs)} s^k}{\sum_{k=0}^N a_k^{(fs)} s^k}.$$

As an alternative, which may be more convenient to apply, the frequency-scaled transfer function may be expressed as follows:

$$H^{(fs)}(s) = \frac{k_f^{N-M} \sum_{k=0}^M k_f^{M-k} b_k s^k}{\sum_{k=0}^N k_f^{N-k} a_k s^k}. \quad (2.112)$$

Therefore, in this case, relating the coefficients of $H^{(fs)}(s)$ in (2.112) to $H(s)$:

$$\beta_k^{(fs)} = k_f^{M-k} b_k, \quad \alpha_k^{(fs)} = k_f^{N-k} a_k,$$

and so

$$H^{(fs)}(s) = \frac{k_f^{N-M} \sum_{k=0}^M \beta_k^{(fs)} s^k}{\sum_{k=0}^N \alpha_k^{(fs)} s^k} .$$

If $H(s)$ is shown in factored form:

$$H(s) = \frac{K \prod_{k=1}^M (s + \gamma_k)}{\prod_{k=1}^N (s + \mu_k)} ,$$

where the zeros, $-\gamma_k$, and the poles, $-\mu_k$, may be, and in general are, complex, then the frequency-scaled transfer function may be expressed as follows:

$$H^{(fs)}(s) = \frac{K k_f^{N-M} \prod_{k=1}^M (s + k_f \gamma_k)}{\prod_{k=1}^N (s + k_f \mu_k)} . \quad (2.113)$$

In the form of (2.113), it is apparent that frequency scaling multiplies the poles and zeros by the frequency-scaling factor, k_f . Since k_f is real and positive, the magnitudes of the poles and zeros are scaled but not the phase angles. A pole/zero plot of $H^{(fs)}(s)$ will be identical to that of $H(s)$ except that the axes will be scaled by k_f .

If $H(s)$ is expressed as the product of first-order and second-order terms over the product of first-order and second-order terms as in (2.40), then the frequency-scaled transfer function may be expressed as follows:

$$H^{(fs)}(s) = \frac{K k_f^{N-M} \prod_{k=1}^{M_1} (s + k_f \gamma_k) \prod_{k=1}^{M_2} (s^2 + k_f \alpha_k s + k_f^2 \beta_k)}{\prod_{k=1}^{N_1} (s + k_f \mu_k) \prod_{k=1}^{N_2} (s^2 + k_f \xi_k s + k_f^2 \lambda_k)} . \quad (2.114)$$

In (2.114), $M = M_1 + 2 M_2$, $N = N_1 + 2 N_2$, and all coefficients are real (they are complex, in general, in (2.113)).

Magnitude Frequency Response and Phase Response

Any critical frequency of interest for $H(j\omega)$, denoted ω_{cr} , such as the passband edge frequency, or the frequency for some given attenuation, is affected as

follows by frequency scaling:

$$\omega_{cr}^{(fs)} = k_f \omega_{cr} \quad ,$$

as can be seen from the basic frequency scaling operation. It follows that

$$H^{(fs)}(j\omega_{cr}^{(fs)}) = H(j\omega_{cr}) \quad .$$

That is, $H^{(fs)}(j\omega)$ will have the same magnitude and phase at $\omega = \omega_{cr}^{(fs)}$ as does $H(j\omega)$ at $\omega = \omega_{cr}$. Since ω_{cr} could be any frequency, the above may be generalized as follows:

$$|H^{(fs)}(j\omega)| = |H(j\omega/k_f)| \quad , \tag{2.115}$$

and

$$\angle H^{(fs)}(j\omega) = \angle H(j\omega/k_f) \quad . \tag{2.116}$$

Therefore, plots of the frequency-scaled magnitude frequency response, (2.115), and phase response, (2.116), are identical to those obtained prior to frequency scaling except for the scaling of the frequency axes by k_f .

Phase Delay and Group Delay

Phase delay is defined by (2.80), and group delay by (2.81). Phase delay for a frequency-scaled transfer function at some critical frequency may be expressed as follows:

$$t_{pd}^{(fs)}(\omega_{cr}^{(fs)}) = - \frac{\angle H^{(fs)}(j\omega_{cr}^{(fs)})}{\omega_{cr}^{(fs)}} = - \frac{\angle H(j\omega_{cr})}{k_f \omega_{cr}} = - t_{pd}(\omega_{cr})/k_f \quad .$$

Group delay for a frequency-scaled transfer function at some critical frequency may be expressed as follows:

$$t_{gd}^{(fs)}(\omega_{cr}^{(fs)}) = - \frac{d \angle H^{(fs)}(j\omega_{cr}^{(fs)})}{d \omega_{cr}^{(fs)}} = - \frac{d \angle H(j\omega_{cr})}{k_f d \omega_{cr}} = - t_{gd}(\omega_{cr})/k_f \quad .$$

Since ω_{cr} could be any frequency, the above may be generalized as follows:

$$t_{pd}^{(fs)}(\omega) = t_{pd}(\omega/k_f)/k_f \quad , \tag{2.117}$$

and

$$t_{gd}^{(fs)}(\omega) = t_{gd}(\omega/k_f)/k_f \quad . \tag{2.118}$$

Therefore, plots of the frequency-scaled phase delay, (2.117), and group delay, (2.118), not only have the frequency axes scaled by k_f , but the amplitude axes are scaled by $1/k_f$. For example, for $k_f = 1000$, $t_{gd}^{(fs)}(2000) = 0.001 t_{gd}(2)$.

Time-Domain Response

From the scaling property of Fourier transforms,

$$h(k_f t) \Leftrightarrow \frac{1}{k_f} H(j\omega/k_f) .$$

It therefore follows that

$$h^{(fs)}(t) = k_f h(k_f t) \Leftrightarrow H^{(fs)}(j\omega) . \quad (2.119)$$

Note that the frequency-scaled unit impulse response, as shown in (2.119), is a time-scaled version of the original unit impulse response and is also amplitude scaled by k_f . If $k_f > 1$, then $h^{(fs)}(t)$ will be greater in amplitude and time-compressed, compared to $h(t)$.

Recall that the unit step response may be expressed as the integral of the unit impulse response. Assuming causality,

$$h_u^{(fs)}(t) = \int_0^t h^{(fs)}(\tau) d\tau = k_f \int_0^t h(k_f \tau) d\tau = \int_0^{k_f t} h(\beta) d\beta = h_u(k_f t) . \quad (2.120)$$

Note that the frequency-scaled unit step response, as shown in (2.120), is a time-scaled version of the original unit step response with no corresponding amplitude scaling.

In the following chapters, there are many graphical plots showing the magnitude frequency response, phase response, phase delay, group delay, unit impulse response and unit step response for various filter types. Often a single graph displays a family of plots, displaying the response for the same filter type with many values of filter order. Usually these plots are for a normalized frequency axis, that is the passband edge frequency is normalized to unity. By applying the concepts of frequency scaling, as presented in this section, the graphs in following chapters give relevant characteristic information for any desired passband edge frequency, i.e., for any desired frequency axis by appropriate frequency scaling.

2.17 CHAPTER 2 PROBLEMS

- 2.1** Given the ideal magnitude frequency response for a lowpass filter as expressed in (2.1), and assuming that the phase response is zero, determine an expression for the unit impulse response. That is, verify (2.2).
- 2.2** Verify that the area under $h_{LP}(t)$, as expressed in (2.2) and plotted in Figure 2.3, is unity independent of ω_c . Also, argue that, in the limit as $\omega_c \rightarrow \infty$, that $h_{LP}(t) \rightarrow \delta(t)$.

- 2.3** Determine an expression for the unit step response of an ideal lowpass filter with zero phase response and with cutoff frequency ω_c . Using MATLAB, plot this unit step response for $\omega_c = 2\pi 1000 \text{ rad/s}$ and for $-2.5\text{ms} < t < 2.5\text{ms}$, and thereby verify the plot shown in **Figure 2.4**. Determine the percentage overshoot in the step response.
- 2.4** Verify that the overshoot of the step response for the ideal lowpass filter, illustrated in **Figure 2.4** and **Figure 2.5**, is independent of ω_c . How is this related to Gibb's phenomenon?
- 2.5** Given the unit impulse response of the ideal lowpass filter as expressed by (2.2), determine the minimum-value to maximum-value rise time for the step response of this filter.
- 2.6** Given the ideal magnitude frequency response for a highpass filter as expressed in (2.3), and assuming that the phase response is zero, determine an expression for the unit impulse response. That is, verify (2.6).
- 2.7** Determine an expression for the unit step response of an ideal highpass filter with zero phase response and cutoff frequency ω_c . Using MATLAB, plot this unit step response for $\omega_c = 2\pi 1000 \text{ rad/s}$, for $-2.5\text{ms} < t < 2.5\text{ms}$.
- 2.8** Given the ideal magnitude frequency response for a bandpass filter as expressed in (2.4), and assuming that the phase response is zero, determine an expression for the unit impulse response. That is, verify (2.7). Using MATLAB, plot this unit impulse response for $f_1 = 9 \text{ kHz}$, $f_2 = 11 \text{ kHz}$, and for $-2.5\text{ms} < t < 2.5\text{ms}$, and thereby verify the plot shown in **Figure 2.10**.
- 2.9** Given the ideal magnitude frequency response for a bandstop filter as expressed in (2.5), and assuming that the phase response is zero, determine an expression for the unit impulse response. That is, verify (2.8). Using MATLAB, plot this unit impulse response for $f_1 = 600 \text{ Hz}$, $f_2 = 1400 \text{ Hz}$, and for $-2.5\text{ms} < t < 2.5\text{ms}$, and thereby verify the $BW = 800 \text{ Hz}$ plot shown in **Figure 2.11**.
- 2.10** Given the family of raised-cosine lowpass filters expressed by (2.9), determine an expression for the corresponding unit impulse response. That is, verify (2.10).

2.11 If the effective time duration of pulses as shown in **Figure 2.14** is defined as the time duration over which the magnitude first becomes 1 % of the peak value until the magnitude drops below, and stays below, 1% of the peak value:

- (a) What is the effective time duration of the pulse shown in **Figure 2.14** when $\alpha = 0$?
- (b) What is the effective time duration of the pulse shown in **Figure 2.14** when $\alpha = 1$?

2.12 **Comment 6** on page 39 has relevance to sampling theory and digital signal processing. It is common in digital signal processing to sample a time-limited analog signal, such as a segment of speech, and perform filtering or some other operation in the digital domain. An anti-aliasing filter is frequently used to band-limit the analog signal prior to sampling. Comment on Nyquist sampling theory²² and the relevance of the **Paley-Wiener Theorem** to sampling of time-limited signals.

2.13 Show that

$$|H(j\omega)| = e^{-\sqrt{|\omega|}},$$

satisfies the **Paley-Wiener Theorem**. That is, verify the results of **Example 2.1**.

2.14 For the following magnitude frequency response:

$$|H(j\omega)| = e^{-k|\omega|},$$

where k is a positive real number,

- (a) Show that this function is square-integrable.
- (b) Show that there is no $H(j\omega)$ with the given magnitude frequency response that has a corresponding $h(t)$ that is causal. That is, show that the **Paley-Wiener Theorem** is not satisfied.
- (c) Determine the half-power frequency, i.e., the frequency where $|H(j\omega)| = |H(0)|/\sqrt{2}$, in terms of k .
- (d) Using MATLAB, generate 1000 equally-spaced samples of $|H(j\omega)|$ with $k = \ln(2)/200$, for $0 \leq \omega \leq 2000$. Plot the results with (i) a linear vertical scale, and, on a second plot, (ii) also with a dB vertical scale.

Note, in this problem, that the magnitude frequency response is smooth, and

²² Sampling theory is often attributed to Nyquist, or Shannon, or more often simply referred to as sampling theory, and the sampling theorem. See the interesting historical footnote on page 517 of Oppenheim, Willsky and Young (1983).

there are no frequency bands where the response is zero, and yet the *Paley-Wiener Theorem* is not satisfied.

2.15 For the following magnitude frequency response:

$$|H(j\omega)| = e^{-\frac{\alpha^2 \omega^2}{2}},$$

- Show that this function is square-integrable for all real $\alpha \neq 0$.
- Show that there is no $H(j\omega)$ with the given magnitude frequency response that has a corresponding $h(t)$ that is causal. That is, show that the **Paley-Wiener Theorem** is not satisfied.
- Determine the half-power frequency, i.e., the frequency where $|H(j\omega)| = |H(0)|/\sqrt{2}$, in terms of α .
- Using MATLAB, generate 1000 equally-spaced samples of $|H(j\omega)|$ with $\alpha = \sqrt{\ln(2)}/100$, for $0 \leq \omega \leq 2000$. Plot the results with (i) a linear vertical scale, and, on a second plot, (ii) also with a *dB* vertical scale.

Note, in this problem, that the magnitude frequency response is smooth, and there are no frequency bands where the response is zero, and yet the *Paley-Wiener Theorem* is not satisfied.

2.16 Given that

$$h(t) = \begin{cases} 5, & 2 \leq t \leq 5 \\ 0, & \text{elsewhere} \end{cases},$$

determine numerical values for t_{o_1} , t_{w_1} , t_{o_2} , t_{w_2} , and t_{w_3} . That is, verify the results of *Example 2.2*.

2.17 Given that

$$h(t) = \gamma e^{-\beta|t - \alpha|},$$

determine expressions for t_{o_1} , t_{w_1} , t_{o_2} , t_{w_2} , and t_{w_3} . That is, verify the results of *Example 2.3*.

2.18 Given that

$$h(t) = \gamma e^{-\frac{(t - \alpha)^2}{2\sigma^2}},$$

determine expressions for t_{o_1} , t_{w_1} , t_{o_2} , t_{w_2} , and t_{w_3} . That is, verify the results of *Example 2.5*.

2.19 Given that

$$h(t) = \begin{cases} 5, & 2 < t < 4 \\ -3, & 4 < t < 6 \\ 0, & \text{elsewhere} \end{cases}$$

determine numerical values for t_{o_1} , t_{w_1} , t_{o_2} , t_{w_2} , and t_{w_3} . That is, verify the results of **Example 2.7**.

2.20 Given the raised-cosine frequency response:

$$H(j\omega) = \begin{cases} \frac{1}{2} \left[1 + \cos \left(\frac{\pi \omega}{2\omega_c} \right) \right], & |\omega| \leq 2\omega_c \\ 0, & \text{elsewhere} \end{cases}$$

determine expressions for ω_{w_1} , ω_{w_2} , and ω_{w_3} . That is, verify the results of **Example 2.8**.

2.21 Given that

$$h(t) = \gamma e^{-\beta|t - \alpha|}$$

determine expressions for ω_{w_1} and ω_{w_3} . That is, verify the results of **Example 2.9**.

2.22 Given that

$$h(t) = \gamma e^{-\frac{(t - \alpha)^2}{2\sigma^2}}$$

determine expressions for ω_{w_1} , ω_{w_2} , and ω_{w_3} . That is, verify the results of **Example 2.11**.

2.23 Given that

$$|H(j\omega)|^2 = \frac{1}{1 + (\omega/\omega_c)^8}$$

and that $\omega_c = 1000$, determine the **Filter Selectivity**, F_S , and the **Shaping Factor**, S_a^b , where $a = 3$ dB, and $b = 60$ dB. Using MATLAB, plot $|H(j\omega)|$ for $0 < \omega < 10000$: use (a) linear scales for both the frequency and magnitude axes, (b) a linear frequency axis and dB for the magnitude axis, and (c) a logarithmic frequency axis and dB for the magnitude axis. All three plots are of the same information, but note the difference in the display formats.

2.24 Given that

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 [4(\omega/\omega_p)^3 - 3(\omega/\omega_p)]^2},$$

where $\epsilon = 0.508847$ and $\omega_p = 913.352$, determine the *Filter Selectivity*, F_s , and the *Shaping Factor*, S_a^b , where $a = 3$ dB, and $b = 60$ dB. Using MATLAB, plot $|H(j\omega)|$, in dB, for $0 < \omega < 10000$.

2.25 Given $H(s) = 1 / (s + 1)$:

- Determine $h(t)$.
- Determine $h(t)$, where $h(t) = \mathcal{L}^{-1}\{H(-s)\}$.
- Determine the autocorrelation function $r(\tau)$ of the response of the filter to a white noise input with unit variance using (2.46).
- Show that $r(\tau) = h(\tau) * \tilde{h}(\tau)$.

2.26 Repeat **Problem 2.25** for

$$H(s) = \frac{s}{s + 1}.$$

2.27 Given $X(\omega) = 4 / (\omega^2 + 4)$:

- Determine whether the *Analog Filter Design Theorem* is or is not satisfied for the given function of ω .
- If the *Theorem* is satisfied, determine $H(s)$.

2.28 Repeat **Problem 2.27** for $X(\omega) = 64 / (\omega^4 + 16)$.

2.29 Repeat **Problem 2.27** for $X(\omega) = 1 / (\omega^4 - 1)$.

2.30 Repeat **Problem 2.27** for $X(\omega) = (\omega^4 - 1) / (\omega^4 + 16)$.

2.31 Using MATLAB, and given that

$$H(s) = \frac{s^2 + 4}{s^2 + 3s + 2},$$

- Plot the magnitude and phase of $H(j\omega)$.
- Determine and plot $h(t)$.
- Determine and plot $\tilde{h}(t)$.
- Determine and plot $r(\tau) = h(\tau) * \tilde{h}(\tau)$.

2.32 Suppose $G(\omega) = e^{-\sigma^2 \omega^2 / 2}$. Note that from **Problem 2.15** there is no causal transfer function that has this given magnitude-squared, nor magnitude,

frequency response. However, as this problem shows, it can be approximated. Let the following approximation be used (Young and van Vliet, 1995):

$$X(\omega) = \frac{A_0}{a_0 + a_2(\sigma\omega)^2 + a_4(\sigma\omega)^4 + a_6(\sigma\omega)^6} \cong G(\omega) ,$$

where $a_0 = 2.490895$, $a_2 = 1.466003$, $a_4 = -0.024393$, $a_6 = 0.178257$, and $A_0 = a_0$. This $X(\omega)$ function satisfies the *Analog Filter Design Theorem*, and the corresponding $H(s)$, substituting numerical values for a_i , $i = 0, 2, 4, 6$, may be expressed as follows:

$$H(s) = \frac{A_1}{(1.1668 + \sigma s)(3.20373 + 2.21567\sigma s + \sigma^2 s^2)} ,$$

where $A_1 = 3.738108$. Note that this type of filter is referred to as a Gaussian filter (an approximation to it) since the magnitude frequency response as well as the impulse response are approximately Gaussian. Using MATLAB, and given that $\sigma = 1.17741$:

- Determine and plot $20 \log |H(j\omega)|$. Use a linear horizontal scale from 0 to 5 rad/s.
- Determine and plot the phase response $\angle H(j\omega)$ over the same frequency range as in part (a).
- Plot the unit impulse response $h(t)$.
- Plot the unit step response $h_u(t)$.
- Plot the poles of $H(s)$.

2.33 Given that

$$H(s) = \frac{1}{s^2 + (\omega_o/Q)s + \omega_o^2} ,$$

compute the poles of $H(s)$ for the following values of Q : 0, 0.25, 0.5, 0.6, $1/\sqrt{2}$, 5, 10, and ∞ . On an s -plane plot, sketch the locus (the path) of the poles as Q varies from 0 to infinity, labeling the points on the locus for the computed values.

2.34 Given that

$$H(s) = \frac{10^4}{s^2 + 10s + 10^4} ,$$

using MATLAB,

- Plot the magnitude and phase frequency response of $H(j\omega)$. Use a logarithmic frequency axis from 1 to 10^4 for both plots. Plot the magnitude in dB.

- (b) Plot the unit impulse response and the unit step response. Use a time axis from 0 to 1 second for both plots.

2.35 Given that

$$H_1(s) = \frac{s + 10}{s + 100}, \quad H_2(s) = \frac{s - 10}{s + 100},$$

and using MATLAB,

- (a) Plot the magnitude and phase response of $H_1(j\omega)$.
 (b) Plot $h_1(t)$ and $h_{u_1}(t)$.
 (c) Plot the magnitude and phase response of $H_2(j\omega)$.
 (d) Plot $h_2(t)$ and $h_{u_2}(t)$.

2.36 Given that

$$H(s) = \frac{s^2 + \alpha s + 10^8}{s^2 + 100s + 10^4},$$

using MATLAB,

- (a) Plot the magnitude and phase frequency response of $H(j\omega)$ when $\alpha = 10^4$. Also plot the unit impulse response and the unit step response.
 (b) Plot the magnitude and phase frequency response of $H(j\omega)$ when $\alpha = -10^4$. Also plot the unit impulse response and the unit step response.
 (c) Compare the phase responses of parts (a) and (b) and comment.
 (d) Comment on the presence of impulses in the time-domain plots of parts (a) and (b).

2.37 Given that

$$H(s) = \frac{s^2 - 10s + 10^4}{s^2 + 10s + 10^4},$$

using MATLAB,

- (a) Plot the magnitude and phase frequency response of $H(j\omega)$.
 (b) Plot the unit impulse response and the unit step response.
 (c) Verify the unit impulse response by computing $h(t)$ by hand, i.e., using Laplace transform theory.

Note that while the magnitude frequency response is constant (this is an all-pass filter), the impulse response will significantly affect the filter input signal $x(t)$, that is, $y(t) \neq x(t - \alpha)$, where $y(t)$ is the filter output. This example stresses the importance of the phase response (if the phase response was $-\alpha\omega$, then $y(t) = x(t - \alpha)$).

2.38 Determine expressions for the phase delay and the group delay of the transfer function, $H(s)$, given in **Problem 2.32**, for $A_1 = 3.738108$ and $\sigma = 1.17741$. Using MATLAB, plot $t_{pd}(\omega)$ and $t_{gd}(\omega)$, using a frequency range of 0 to 5 rad/s.

2.39 Determine expressions for the phase delay and the group delay of the transfer functions, $H_1(s)$ and $H_2(s)$, given in **Problem 2.35**. Using MATLAB, plot both phase delay functions and both group delay functions, using a frequency range of 0 to 1000 rad/s.

2.40 Determine numerical values for $t_{pd}(\omega)$ and $t_{gd}(\omega)$ at $\omega = 10^5$ for the following transfer function:

$$H(s) = \frac{10^5}{s + 10^5}.$$

That is, verify the results of **Example 2.23**.

2.41 Determine numerical values for $t_{pd}(\omega)$ and $t_{gd}(\omega)$ at $\omega = 10^5$ for the following transfer function:

$$H(s) = \frac{10^{10}}{s^2 + \sqrt{2} 10^5 s + 10^{10}}.$$

That is, verify the results of **Example 2.24**.

2.42 Given that the real part of $H(j\omega)$ may be expressed as

$$R(\omega) = \frac{400}{400 + \omega^2},$$

using Hilbert transform relations, determine the transfer function $H(s)$.

2.43 Given that the real part of $H(j\omega)$ may be expressed as

$$R(\omega) = \frac{\omega^2}{900 + \omega^2},$$

using Hilbert transform relations, determine the transfer function $H(s)$.

2.44 Given that

$$|H(j\omega)| = \frac{1}{\sqrt{1 + (\omega/100)^4}},$$

determine the corresponding phase response. HINT: use the **Analog Filter Design Theorem** and the method of **Section 2.7**.

- 2.45** Given the phase function expressed in (2.108) with $\omega_c = 1$, $t_d = \pi$, and that $\rho(0) = 0$, determine the frequency at which the magnitude frequency response is $1/\sqrt{2}$ times the maximum value (the half-power frequency).
- 2.46** Repeat **Problem 2.45** for an arbitrary ω_c ($t_d = \pi$ and $\rho(0)$ arbitrary).
- 2.47** Given the phase function expressed in (2.108) with $\omega_c = 1$, $t_d = \pi$, and that $\rho(0) = 0$, using MATLAB, determine and plot the corresponding unit impulse response.
- 2.48** Given that

$$H(s) = \frac{s^2 + 4}{s^2 + 3s + 2},$$

- (a) Determine, by hand, expressions for the magnitude frequency response and the phase response.
- (b) Determine, by hand, an expression for the unit impulse response.
- (c) For a frequency scaling factor $k_f = 10$ applied to $H(s)$, determine expressions for the frequency scaled transfer function, the frequency scaled magnitude frequency response, the frequency scaled phase response, and the frequency scaled unit impulse response.

CHAPTER 3

BUTTERWORTH FILTERS

Butterworth (1930) introduced one of the earliest systematic analog filter design methods, and it is still one of the most widely used. The Butterworth filter design method is one of the *classical* filter design procedures. Other classical filter design methods are Chebyshev Type I, Chebyshev Type II, elliptic (or Cauer), and Bessel; each of these will be studied in following chapters. In this chapter the Butterworth, or maximally-flat magnitude, response will be defined, and it will be observed that it satisfies the *Analog Filter Design Theorem*. Explicit formulas for the design and analysis of Butterworth filters, such as *Filter Selectivity*, *Shaping Factor*, the minimum required order to meet design specifications, etc., will be obtained. From the defining $|H(j\omega)|^2$ the corresponding $H(s)$ will be determined, and a very simple and convenient means for determining the filter poles will be found. To complete the study of lowpass, prototype Butterworth filters, the phase response, phase delay, group delay, and time-domain response characteristics will be investigated.

3.1 MAXIMALLY-FLAT MAGNITUDE

Consider a general magnitude-squared frequency response for a transfer function that only has finite poles (all zeros at infinity) with arbitrary coefficients that satisfies the *Analog Filter Design Theorem* of Chapter 2, following (2.39), (2.44) and (2.49):

$$|H(j\omega)|^2 = \frac{K^2}{\sum_{k=0}^N \alpha_k \omega^{2k}}, \quad (3.1)$$

where the coefficients K , and $\{\alpha_k\}_{k=0}^N$ are to be determined. For convenience, let (3.1) be expressed as follows:

$$|H(j\omega)|^2 = \frac{K^2}{\alpha_0} \times \frac{1}{\sum_{k=0}^N \beta_k \omega^{2k}} \quad (3.2)$$

where $\beta_k = \alpha_k / \alpha_0$. Following Lindquist (1977), by long division (3.2) may be expressed:

$$|H(j\omega)|^2 = \frac{K^2}{\alpha_0} \times [1 - \beta_1 \omega^2 + (\beta_1^2 - \beta_2) \omega^4 - (\beta_1^3 - 2\beta_1 \beta_2 + \beta_3) \omega^6 + (\beta_1^4 - 3\beta_1^2 \beta_2 + 2\beta_1 \beta_3 + \beta_2^2 - \beta_4) \omega^8 - \dots] \quad (3.3)$$

Expanding (3.2) by a Maclaurin series, the function may also be expressed:

$$|H(j\omega)|^2 = \frac{K^2}{\alpha_0} \times \sum_{k=0}^{\infty} \frac{G^{(k)}(0)}{k!} \omega^k \quad (3.4)$$

where

$$G^{(k)}(0) = \left. \frac{d^k |H(j\omega)|^2}{d\omega^k} \right|_{\omega=0}$$

By equating coefficients in (3.3) and (3.4), it can be seen that

$$G^{(k)}(0) = 0 \quad \forall k \text{ odd} \quad ,$$

and

$$G^{(0)}(0) = 1, \quad G^{(2)}(0) = -\beta_1 2!, \quad G^{(4)}(0) = (\beta_1^2 - \beta_2) 4!, \\ G^{(6)}(0) = -(\beta_1^3 - 2\beta_1 \beta_2 + \beta_3) 6!, \quad \dots$$

It is desired to make the magnitude-squared response of (3.2) maximally flat. One way to accomplish this, at least at DC, is to set as many derivatives evaluated at DC to zero ($G^{(k)}(0)$) as possible. If all $\{\beta_k\}_{k=1}^N = 0$ then (3.2) would not be a function of ω . Therefore, with only $\beta_0 = 1$ and β_N non-zero, the first $N-1$ derivatives of (3.2) at $\omega = 0$ will be zero:

$$G^{(k)}(0) = 0, \quad 1 \leq k \leq N-1 \quad .$$

This will result in what is denoted as a *maximally-flat* function. Maximally flat at DC, since the first $N-1$ derivatives of (3.2) at DC are zero. Therefore,

$$|H(j\omega)|^2 = \frac{K^2}{\alpha_0} \times \frac{1}{1 + \beta_N \omega^{2N}} \quad (3.5)$$

Note that the DC value of (3.5) is K^2 / α_0 . For convenience let K^2 / α_0 be unity and β_N be $1/\omega_c^{2N}$. With these simplifications, the Butterworth response may be defined.

Definition of the magnitude-squared Butterworth response:

$$|H(j\omega)|^2 = \frac{1}{1 + (\omega/\omega_c)^{2N}} \tag{3.6}$$

Note that in (3.6) N is the Butterworth filter order,¹ and the magnitude-squared response is 1/2 when $\omega = \omega_c$ independent of N . This is referred to as the 3 dB frequency.²

Note, again, that by simple inspection, it is observed that (3.6) satisfies the **Analog Filter Design Theorem**. Therefore, since the **Theorem** is satisfied, there exists a transfer function that is time-invariant, causal, linear, rational with real coefficients, and is stable³ with the magnitude-squared frequency response of (3.6).

See **Figure 3.1** for plots of (3.6) for several values of N . When plotted as shown, with the vertical scale in dB and a logarithmic frequency scale, the fall-off for $\omega \gg \omega_c$ is a straight line, and the slope is $-N \times 20$ dB/decade. This is, of course, well known from a study of Bode plots (Siebert, 1986)⁴. Also see **Figure 3.2** for detailed plots of (3.6) across the passband; note that larger values of N yield a closer approximation to the ideal lowpass response.

3.2 FILTER SELECTIVITY AND SHAPING FACTOR

Applying (2.37), the definition of **Filter Selectivity**, to the square root of (3.6) results in

$$F_s = \frac{N}{2\sqrt{2} \omega_c} \tag{3.7}$$

Therefore, for a Butterworth filter, **Filter Selectivity** is directly related to filter order N , and inversely related to the 3 dB cutoff frequency ω_c .

¹In **Section 3.4** it is shown that N is the number of poles in $H(s)$.
²Actually, the attenuation is slightly greater than 3 dB. It is precisely $-10 \log(1/2)$. Throughout this book, the term “3 dB” precisely means $-10 \log(1/2)$.
³See **Section 2.6** for a discussion of imposed constraints: the constraints are summarized on page 57. Satisfaction of the **Analog Filter Design Theorem**, summarized on page 61, implies the imposed constraints are met.
⁴See **Sections 2.8** and **2.9**, especially page 67.

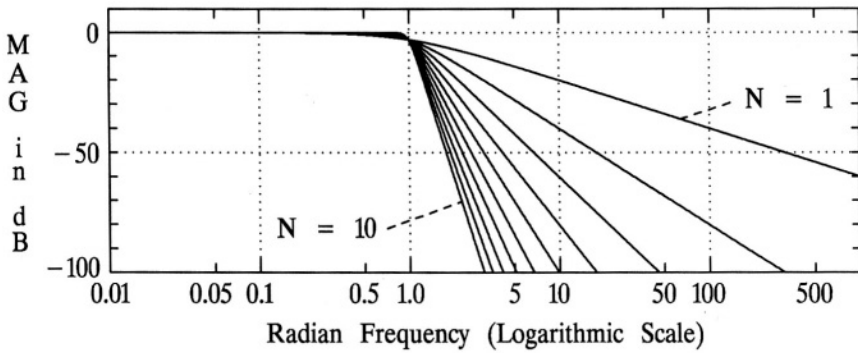


Figure 3.1 The Butterworth magnitude response. Plots of (3.6) for $\omega_c = 1$ and values of N from 1 through 10.

Let A be an arbitrary attenuation in dB relative to the DC value. From (3.6):

$$A = 10 \log (1 + (\omega/\omega_c)^{2N}) . \tag{3.8}$$

For a given A , solving (3.8) for ω would be equivalent to solving for the bandwidth at that attenuation A :

$$BW = \omega_c (10^{A/10} - 1)^{1/(2N)} . \tag{3.9}$$

Using (3.9) and applying (2.38), the definition of *Shaping Factor*, the Butterworth filter *Shaping Factor* may be readily found:

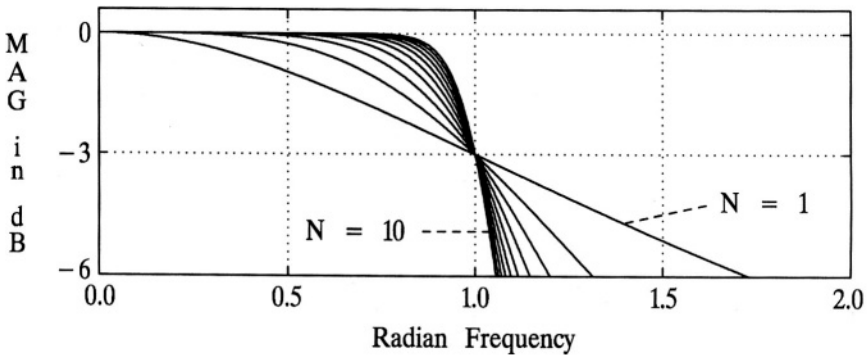


Figure 3.2 The Butterworth magnitude response. Passband details of (3.6) for $\omega_c = 1$ and values of N from 1 through 10.

$$S_a^b = \frac{BW_b}{BW_a} = \left(\frac{10^{b/10} - 1}{10^{a/10} - 1} \right)^{1/(2N)}, \quad (3.10)$$

where b is the attenuation in dB at the wider bandwidth, a is the attenuation in dB at the narrower bandwidth, BW_b is the wider bandwidth, and BW_a is the narrower bandwidth.

Example 3.1

For this example, let $a = 3$ dB and $b = 80$ dB . Then S_3^{80} may be computed for a Butterworth filter using (3.10) for $N = 1, 2, \dots, 10$. The results are as follows: 10000.0, 100.0, 21.54, 10.0, 6.31, 4.64, 3.73, 3.16, 2.78, 2.51, respectively. \square

3.3 DETERMINATION OF ORDER

In the design of an analog filter, an important step is determining the minimum required order to meet given specifications. Refer to **Figure 2.15** on page 47 in specifying the desired filter magnitude characteristics. As long as the filter magnitude frequency response fits within the acceptable corridor indicated in **Figure 2.15**, it satisfies the specifications. Note that A_p , the permitted magnitude deviation within the passband ($0 \leq \omega \leq \omega_p$), and that A_s , the minimum attenuation relative to the passband peak response within the stopband ($\omega \geq \omega_s$), are in dB . The *transition band* is the range of frequencies between ω_p and ω_s ($\omega_p < \omega < \omega_s$). Note that, for convenience, the peak response in the passband is often taken to be 0 dB , however, it could be any value without effecting the following development. Therefore, **Figure 2.15** serves as a convenient definition of several important variables and terms: A_p , A_s , ω_p , ω_s , *passband*, *transition band*, and *stopband*. Refer to **Section 2.4** for more detail.

Starting with (3.6):

$$10 \log |H(j\omega)|^2 = -10 \log [1 + (\omega/\omega_c)^{2N}] . \quad (3.11)$$

Temporarily let η , a real variable, assume the role of N , an integer. This will allow for a solution of the filter order that just meets the filter specifications, even though that solution will, in general, not be an integer. From (3.11), let

$$A_p = 10 \log [1 + (\omega_p/\omega_c)^{2\eta}] . \quad (3.12)$$

From (3.12), the following is easily obtained:

$$(\omega_p/\omega_c)^{2\eta} = 10^{A_p/10} - 1 . \quad (3.13)$$

Similarly,

$$(\omega_s / \omega_c)^{2\eta} = 10^{A_s/10} - 1 \quad (3.14)$$

Note that (3.13) and (3.14) are directly related to (3.9), and could have been obtained from (3.9) by appropriately defining BW and A . Now note that the ratio of (3.14) to (3.13) may be expressed as follows:

$$\left(\frac{\omega_s}{\omega_p} \right)^{2\eta} = \frac{10^{A_s/10} - 1}{10^{A_p/10} - 1},$$

from which η may be solved for. Letting $N = \lceil \eta \rceil$, where $\lceil \eta \rceil$ is the smallest integer equal to or larger than η ($\eta \leq \lceil \eta \rceil < \eta + 1$), the minimum order required to meet the specifications may be determined from the following:

$$N = \left\lceil \frac{\log \{ [(10^{A_s/10} - 1) / (10^{A_p/10} - 1)]^{1/2} \}}{\log (\omega_s / \omega_p)} \right\rceil \quad (3.15)$$

If η is not an integer, then $N > \eta$ and the specifications will be exceeded. Note that N is independent of ω_c .

Example 3.2

Suppose the following specifications are given: $f_p = 3,000 \text{ Hz}$, $f_s = 7,000 \text{ Hz}$, $A_p = 2 \text{ dB}$, and $A_s = 60 \text{ dB}$. From the right side of (3.15), $\eta = 8.469$. If it were possible to have a filter order equal to η , the filter specifications would be exactly met, without exceeding them. However, applying (3.15), $N = 9$, and the stated filter specifications will be exceeded. There are a family of possible 9th-order Butterworth filters that exceed the stated specifications. Two cases are shown in Figure 3.3. One case just meets the specifications at the passband edge, the other case just meets the specifications at the stopband edge. There are, theoretically, by adjusting the value of ω_c , an infinite number of other cases in between the two cases shown in Figure 3.3. □

To make the design solution unique, precisely meeting the design specifications at the passband edge will be adhered to. As in Example 3.2, the determination of the order N does not require knowledge of the 3 dB frequency ω_c . However, as will be seen, to carry out the design of a Butterworth filter does require knowing ω_c . Generally, a Butterworth filter is initially designed for a normalized $\omega_c = 1$, and then frequency scaling is performed to obtain the desired ω_c .⁵ Assuming the filter is

⁵The numerical values of the transfer function poles are dependent on a knowledge of the numerical value of ω_c . See Section 2.16 for a presentation of frequency scaling.

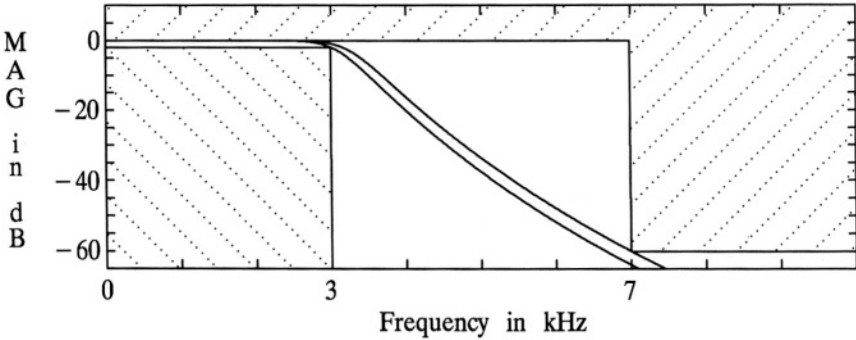


Figure 3.3 Magnitude frequency response plots for *Example 3.2*. Both responses are for a 9th-order Butterworth filter that meets the design specifications.

designed to precisely meet the stated magnitude specifications at the passband edge, (3.12) may be used, substituting N for η , to compute ω_c

$$\omega_c = \frac{\omega_p}{\left(10^{A_p/10} - 1\right)^{1/(2N)}} \tag{3.16}$$

3.4 POLE LOCATIONS

Starting with (3.6) and following the procedure used in Section 2.7:

$$Y(s) = H(s)H(-s) = \frac{1}{1 + (s/j\omega_c)^{2N}} \tag{3.17}$$

The poles of (3.17) may be found by setting

$$(s_k/j\omega_c)^{2N} = -1 \ ,$$

and solving for the values of s_k . Noting that $j^{2N} = (-1)^N$,

$$s_k = \omega_c [-(-1)^N]^{1/(2N)} \tag{3.18}$$

Suppose $N = 1$, then by application of (3.18) it is easy to see that $s_k = \pm \omega_c$. If

$N = 2$, then $s_k = \omega_c e^{j\pi(2k-1)/4}$, for $k = 1, 2, 3, 4$. For $k > 4$ the poles repeat. Equations to find the poles may be generalized as follows:

$$s_k = \omega_c e^{j\pi(2k-1)/2N}, \quad 1 \leq k \leq 2N, \quad N \text{ even}, \quad (3.19)$$

and

$$s_k = \omega_c e^{j\pi(k-1)/N}, \quad 1 \leq k \leq 2N, \quad N \text{ odd}. \quad (3.20)$$

Note that (3.19) and (3.20) give the poles for $H(s)H(-s)$; only those in the left-half of the s plane are for $H(s)$. Note that the poles, for N odd or N even, are equally spaced on a circle of radius ω_c . Note also that there are poles on the real axis only if N is odd.

See Figure 3.4 for a plot of the poles for $N = 3$ and $\omega_c = 1$. See Figure 3.5 for a plot of the poles for $N = 4$ and $\omega_c = 1$. As mentioned on page 55, note the *quadrantal symmetry*.

If Euler's relation is made use of, (3.19) and (3.20) may be both expressed as follows for left-half plane poles:

$$s_k = \omega_c \left[-\sin\left[\pi(2k-1)/2N\right] + j \cos\left[\pi(2k-1)/2N\right] \right], \quad (3.21)$$

$$1 \leq k \leq N, \quad N \text{ even or odd}.$$

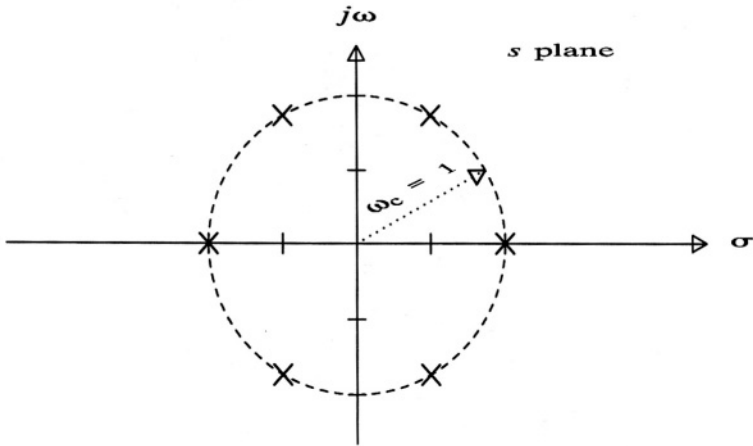


Figure 3.4 A plot of the poles of $H(s)H(-s)$ for $N = 3$ and $\omega_c = 1$.

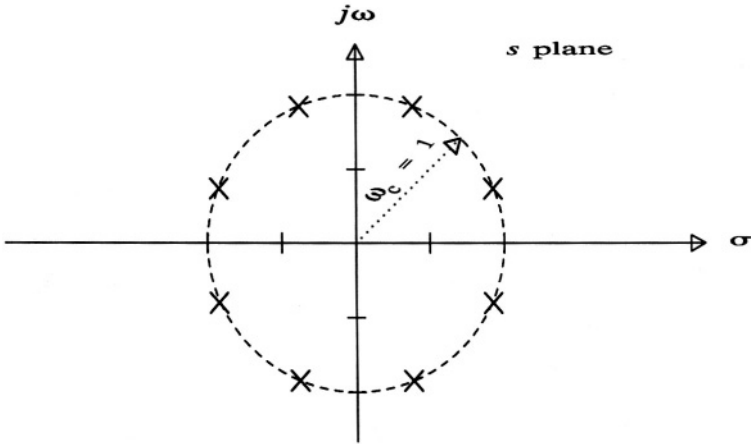


Figure 3.5 A plot of the poles of $H(s)H(-s)$ for $N = 4$ and $\omega_c = 1$.

Example 3.3

In this example, the transfer function of a third-order Butterworth filter normalized for $\omega_c = 1$ is found. From (3.20) or (3.21), the three poles of $H(s)$ are as follows: $-1, -1/2 \pm j\sqrt{3}/2$. Therefore,

$$H(s) = \frac{1}{(s + 1)(s^2 + s + 1)},$$

or

$$H(s) = \frac{1}{s^3 + 2s^2 + 2s + 1}. \tag{3.22}$$

The third-order polynomial in the denominator of (3.22) is denoted as a *third-order Butterworth polynomial*. □

Example 3.4

In this example, the transfer function of a fourth-order Butterworth filter normalized for $\omega_c = 1$ is found. From (3.19) or (3.21), the four poles of $H(s)$ are as follows: $e^{\pm j5\pi/8}, e^{\pm j7\pi/8}$. Therefore,

$$H(s) = \frac{1}{(s^2 + 0.7653669s + 1)(s^2 + 1.847759s + 1)},$$

or

$$H(s) = \frac{1}{s^4 + 2.613126s^3 + 3.414213s^2 + 2.613126s + 1} \quad (3.23)$$

The denominator of (3.23) is a *fourth-order Butterworth polynomial*. □

3.5 PHASE RESPONSE, PHASE DELAY, AND GROUP DELAY

A Butterworth filter, as seen above, is designed to meet given magnitude response specifications. Once the transfer function is determined, it may be put in the following form:

$$H(s) = \frac{1}{\sum_{i=0}^N a_i s^i} \quad (3.24)$$

which is an all-pole form of (2.76). Given (3.24), the phase response, from (2.79) may be stated as follows:

$$\angle H(j\omega) = -\arctan \frac{I_d(\omega)}{R_d(\omega)} \quad (3.25)$$

where $R_d(\omega)$ denotes the real part of the denominator of (3.24) evaluated with $s = j\omega$, and $I_d(\omega)$ denotes the imaginary part. The phase response of a Butterworth filter, with a normalized $\omega_c = 1$, for several values of N , is shown in **Figure 3.6**. Taking the initial phase slope as a linear-phase reference, deviations from linear phase, for several values of N , are shown in **Figure 3.7**.

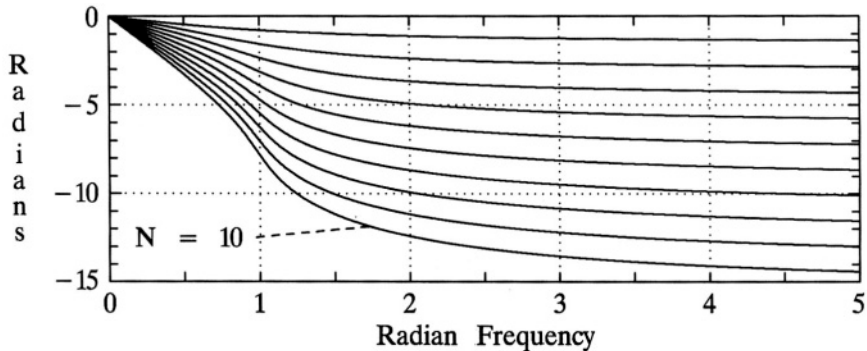


Figure 3.6 A plot of the phase response for a Butterworth filter with normalized $\omega_c = 1$ for values of N from 1 through 10.

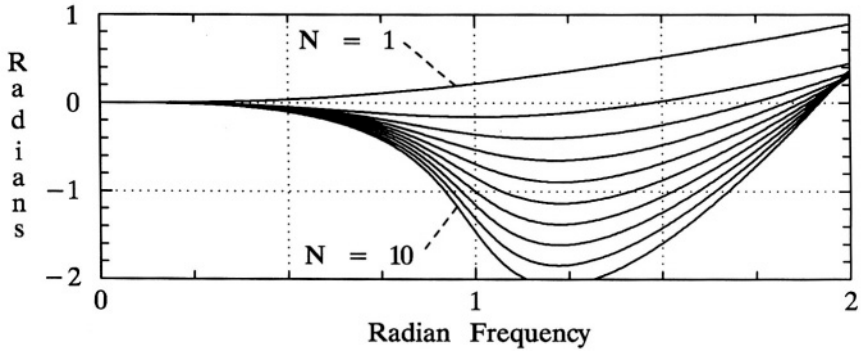


Figure 3.7 Phase deviation from linear for a Butterworth filter with normalized $\omega_c = 1$ for values of N from 1 through 10.

The *phase delay*, $t_{pd}(\omega)$, for a filter is defined in (2.80), which is repeated here for convenience:

$$t_{pd}(\omega) = - \frac{\angle H(j\omega)}{\omega} . \tag{3.26}$$

Using (3.25) in (3.26), the phase delay for a lowpass Butterworth filter may be expressed as

$$t_{pd}(\omega) = \frac{1}{\omega} \arctan \frac{I_d(\omega)}{R_d(\omega)} . \tag{3.27}$$

The *group delay* for a filter, $t_{gd}(\omega)$, is defined by (2.81) and is repeated here for convenience:

$$t_{gd}(\omega) = - \frac{d \angle H(j\omega)}{d\omega} . \tag{3.28}$$

The phase delay of a Butterworth filter, with a normalized $\omega_c = 1$, for several values of N , is shown in **Figure 3.8**. The group delay of a Butterworth filter, with a normalized $\omega_c = 1$, for several values of N , is shown in **Figure 3.9**. Note that for values of $\omega < 0.5$ the phase delay values shown in **Figure 3.8** agree very closely with the group delay values shown in **Figure 3.9**. This is expected since, as shown in **Figures 3.6** and **3.7**, the phase response closely approximates a linear response for that frequency range.

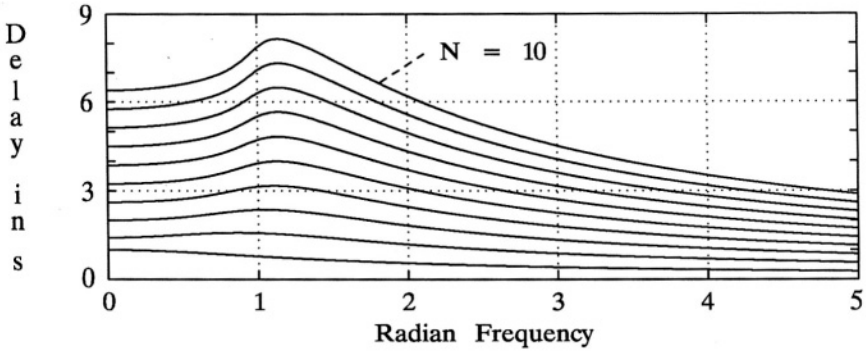


Figure 3.8 A plot of the phase delay for a Butterworth filter with normalized $\omega_c = 1$ for values of N from 1 through 10.

Note also that for values of $\omega > 0.5$ the phase delay values shown in **Figure 3.8** differ significantly from the group delay values shown in **Figure 3.9**. This also is expected since, as shown in **Figures 3.6** and **3.7**, the phase response curves are nonlinear for that frequency range.

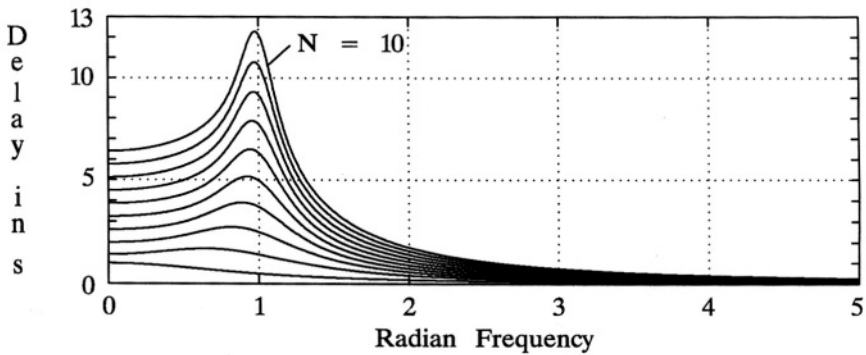


Figure 3.9 A plot of the group delay for a Butterworth filter with normalized $\omega_c = 1$ for values of N from 1 through 10.

3.6 TIME-DOMAIN RESPONSE

The unit impulse response of a Butterworth filter,⁶ with a normalized $\omega_c = 1$, for several values of N , is shown in **Figure 3.10**. Compared to the impulse responses shown in **Figure 2.3** for the ideal lowpass filter, those shown in **Figure 3.10** have similarities and differences. They differ in that they are causal and do not have even symmetry. However they do bear some similarity in the waveshape, especially for larger values of N . Also note that the period of the oscillations is approximately equal to $1/(2\pi)$ ($\omega_c/(2\pi)$ in general), the same as in **Figure 2.3**. As an interesting observation, note that the delay, taken as the time at which the maximum value of the response occurs, agrees very favorably with the phase delay at $\omega = 1$. This is expected since, referring to (2.2) for the ideal case, the impulse response may be modeled as a sinusoid at the 3 dB frequency modulated by an envelope. Therefore the impulse response will be delayed by approximately the phase delay evaluated at $\omega = \omega_c$. Note that the delay does not correlate well with the group delay, since the maximum of the impulse response occurs at the peak of the “carrier,” not the peak of the envelope.

The unit step response of a Butterworth filter, with normalized $\omega_c = 1$, for several values of N , is shown in **Figure 3.11**. Compared to the step responses shown in **Figure 2.5** for the ideal lowpass filter, those shown in **Figure 3.11** have similarities

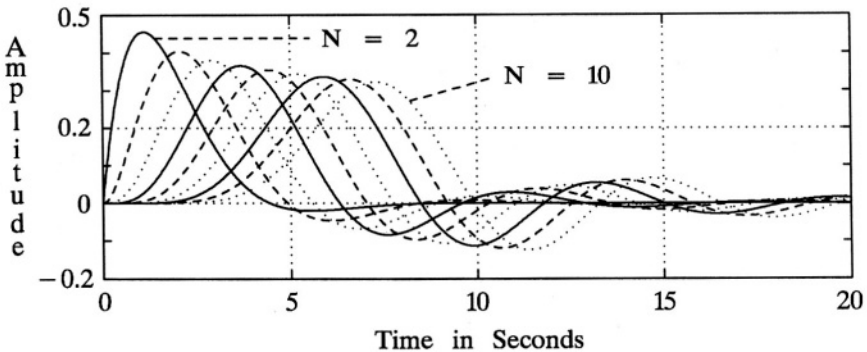


Figure 3.10 A plot of the unit impulse response for a Butterworth filter with normalized $\omega_c = 1$ for values of N from 2 through 10.

⁶See Section 2.13 for a general introduction to time-domain responses.

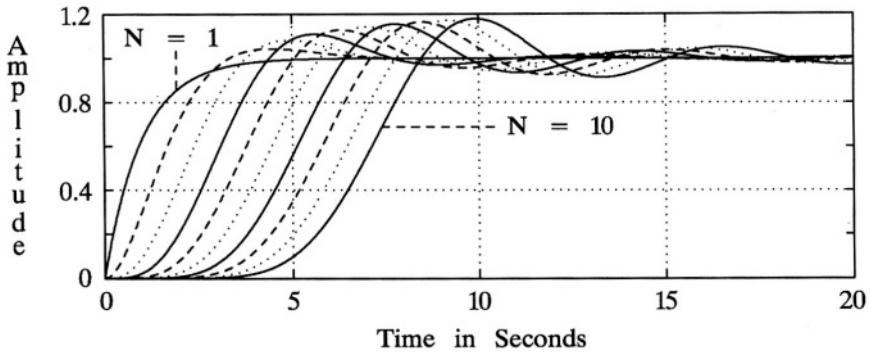


Figure 3.11 A plot of the unit step response for a Butterworth filter with normalized $\omega_c = 1$ for values of N from 1 through 10.

and differences. They differ in that they are causal. However they are quite similar in waveshape for amplitudes greater than 0.5 for larger values of N .

3.7 CHAPTER 3 PROBLEMS

3.1 Given that

$$H(s) = \frac{s + \beta}{s^2 + \alpha s + 1},$$

determine the relationship between α and β , without a pole-zero cancellation, such that the magnitude frequency response will be *maximally flat*, as defined in **Section 3.1**.

3.2. Given that

$$H(s) = \frac{s^2 + \alpha s + \beta}{s^2 + \gamma s + 1},$$

determine the relationship among α , β , and γ , without a pole-zero cancellation, such that the magnitude frequency response will be *maximally flat*, as defined in **Section 3.1**.

3.3 Verify that the first $N-1$ derivatives of (3.6), evaluated at $\omega = 0$, are zero.

- 3.4** By referring to **Figure 3.1**, verify that the transition band slope of a Butterworth filter, for $N = 1$ and $N = 5$, is $-N \times 20$ dB/decade.
- 3.5** By direct application of (2.37) to (3.6), determine *Filter Selectivity* for a Butterworth filter of order 1, and also order 5, for an arbitrary ω_c . Compare your results with that found from a direct use of (3.7).
- 3.6** Estimate the *Shaping Factor* for a Butterworth filter of order 1, $a = 3$ dB, and $b = 60$ dB, from Figure 3.1. Compute the *Shaping Factor* from (3.10) and compare the result with your estimated value from the figure.
- 3.7** Estimate the *Shaping Factor* for a Butterworth filter of order 5, $a = 3$ dB, and $b = 100$ dB, from **Figure 3.1**. Compute the *Shaping Factor* from (3.10) and compare the result with your estimated value from the figure.
- 3.8** Determine the minimum required Butterworth filter order to meet the following set of specifications: $A_p = 1$ dB, $A_s = 45$ dB, $\omega_p = 520$ rad/s, and $\omega_s = 12,500$ rad/s.
- 3.9** Determine the minimum required Butterworth filter order to meet the following set of specifications: $A_p = 2.5$ dB, $A_s = 60$ dB, $f_p = 1000$ Hz, and $f_s = 5,000$ Hz.
- 3.10** For the specifications of **Problem 3.8** and the determined order, find the maximum value that A_s may be increased to without increasing the order.
- 3.11** For the specifications of **Problem 3.9** and the determined order, find the maximum value that A_s may be increased to without increasing the order.
- 3.12** For the specifications of **Problem 3.8** and the determined order, find the maximum value that ω_p may be increased to without increasing the order.
- 3.13** For the specifications of **Problem 3.9** and the determined order, find the maximum value that ω_p may be increased to without increasing the order.
- 3.14** For the specifications of **Problem 3.8** and the determined order, determine the numerical value of ω_c .
- 3.15** For the specifications of **Problem 3.9** and the determined order, determine the numerical value of ω_c .

- 3.16** Determine the pole locations of a third-order Butterworth filter with $\omega_c = 500 \text{ rad/s}$. Determine and state the transfer function, $H(s)$, with (a) the denominator expressed as a non-factored third-order polynomial, and (b) the denominator expressed as the product of a first-order and a second-order polynomial both with real coefficients, and (c) the denominator expressed as the product of three first-order terms.
- 3.17** Determine the pole locations of a fourth-order Butterworth filter with $f_c = 5000 \text{ Hz}$. Determine and state the transfer function, $H(s)$, with (a) the denominator expressed as a non-factored fourth-order polynomial, and (b) the denominator expressed as the product of two second-order polynomials both with real coefficients, and (c) the denominator expressed as the product of four first-order terms.
- 3.18** Based on **Figure 3.7**, which order of Butterworth filter has the most linear phase response? By using a straight-edge applied to **Figure 3.6**, do you arrive at the same conclusion? Do the results shown in **Figure 3.8** agree with your conclusion? Do the results shown in **Figure 3.9** agree with your conclusion?
- 3.19** The minimum value of the phase deviation shown in **Figure 3.7** for $N = 10$ is off scale. Compute the numerical value of the minimum value.
- 3.20** Compute the maximum value of the phase delay (do not just read it from the plot) and the radian frequency at which it occurs, as graphically shown in **Figure 3.8**, for $N = 10$.
- 3.21** Compute the maximum value of the group delay (do not just read it from the plot) and the radian frequency at which it occurs, as graphically shown in **Figure 3.9**, for $N = 10$.
- 3.22** By making use of the scaling property of Fourier transforms (see **Appendix B**), and referring to **Figure 3.6**, at what approximate radian frequency would an eighth-order Butterworth filter with $\omega_c = 1250 \text{ rad/s}$ have a phase of -10 radians ? What would be the corresponding frequency in Hertz?
- 3.23** By making use of the scaling property of Fourier transforms and **Figures 3.8** and **3.9**, what are the approximate values of phase delay and group delay at 1000 Hz for a tenth-order Butterworth filter with $f_c = 2000 \text{ Hz}$?
- 3.24** Suppose an amplitude-modulated “carrier” of 1000 Hz , with a very narrow bandwidth, is applied to the filter described in **Problem 3.23**. Explain the

significance of the values obtained for phase delay and group delay in **Problem 3.23**.

- 3.25** By making use of the scaling property of Fourier transforms and **Figure 3.10**, determine, for an eighth-order Butterworth filter with $f_c = 3000 \text{ Hz}$, approximate values for the following:
- The time at which the unit impulse response is a maximum.
 - The amplitude of the unit impulse response maximum.
 - The width of the unit impulse response, defined as the time during which $0.5 h_{\max} \leq h(t) \leq h_{\max}$.
- 3.26** Repeat **Problem 3.25** for $f_c = 10,000 \text{ Hz}$.
- 3.27** Note from **Problems 3.25** and **3.26** that the time at which the unit impulse response is a maximum is *inversely related* to the frequency scaling factor and the maximum value of the unit impulse response is *directly related* to the frequency scaling factor. However, the unit step response, graphically shown in **Figure 3.11**, while having a time axis that is *inversely related* to the frequency scaling factor, has an amplitude scale that is not effected by frequency scaling. Explain, mathematically and intuitively, why the unit step response amplitude is not effected by frequency scaling.
- 3.28** The purpose of this problem is to compare a 3rd-order Butterworth filter with $\omega_c = 1$ to the 3rd-order Gaussian filter of **Problem 2.30**. Using MATLAB,
- Using a linear frequency scale from 0 to 10 *rad/s*, and a magnitude scale in *dB*, overlay-plot on the same graph the magnitude frequency response of the Butterworth filter and the Gaussian filter.
 - Using a linear frequency scale from 0 to 10 *rad/s*, and a vertical scale in *radians*, overlay-plot on the same graph the phase response of the Butterworth filter and the Gaussian filter.
 - Using a linear frequency scale from 0 to 10 *rad/s*, and a vertical scale in *seconds*, overlay-plot on the same graph the phase delay of the Butterworth filter and the Gaussian filter.
 - Using a linear frequency scale from 0 to 10 *rad/s*, and a vertical scale in *seconds*, overlay-plot on the same graph the group delay of the Butterworth filter and the Gaussian filter.
 - Using a linear time scale from 0 to 10 *s*, overlay-plot on the same graph the unit impulse response of the Butterworth filter and the Gaussian filter.

- (f) Using a linear time scale from 0 to 10 s , overlay-plot on the same graph the unit step response of the Butterworth filter and the Gaussian filter.
- (g) Overlay-plot on the same complex s plane graph the poles of the Butterworth filter and the Gaussian filter.
- (h) Based upon the above graphs:
 - (h1) In what ways does the Butterworth filter appear to be superior to the Gaussian filter?
 - (h2) In what ways does the Gaussian filter appear to be superior to the Butterworth filter?

Note: for each graph in parts (a) through (g), indicate which plot is for the Butterworth filter and which is for the Gaussian filter, properly label the x axis and the y axis for each figure, and include a title for each graph.

CHAPTER 4

CHEBYSHEV TYPE I FILTERS

Although a logical presentation of classical analog filters frequently follows the order of Butterworth, Chebyshev Type I, Chebyshev Type II, and elliptic, such as is done in this book, this is not a chronological order, as was shown in **Section 1.4**. In fact, Chebyshev filters were not developed until the 1950s, some twenty years after the development of Butterworth and elliptic filters. However, the order indicated above is logical, and is followed here, since Butterworth filters have a maximally-flat passband magnitude and monotonically fall off outside the passband, Chebyshev Type I filters have a magnitude response that ripples in the passband and monotonically falls off outside the passband, Chebyshev Type II filters are flat in the passband and ripple in the stopband, and elliptic filters ripple in both the passband and the stopband.

In this chapter the Chebyshev Type I, or simply *Chebyshev*, response is defined, and it will be observed that it satisfies the **Analog Filter Design Theorem**. Explicit formulas for the design and analysis of Chebyshev Type I filters, such as **Filter Selectivity**, **Shaping Factor**, the minimum required order to meet design specifications, etc., will be obtained. From the defining $|H(j\omega)|^2$ the corresponding $H(s)$ will be determined, and means for determining the filter poles are found. To complete the study of lowpass, prototype *Chebyshev* filters, the phase response, phase delay, group delay, and time-domain response characteristics are investigated.

4.1 EQUIRIPPLE PASSBAND MAGNITUDE

The success of the Butterworth response is based on a polynomial in the denominator of $|H(j\omega)|^2$ that remains close to unity for small values of ω and increases rapidly in value for $\omega > \omega_c$. Or, stated another way, the denominator is unity plus a polynomial, $(\omega/\omega_c)^{2N}$, that remains very small for small values of ω and increases rapidly in value for $\omega > \omega_c$. Using this basic concept, let the magnitude-squared frequency response for a Chebyshev Type I response be defined as follows:

Definition of the magnitude-squared Chebyshev Type I response:

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 C_N^2(\omega/\omega_p)} \quad (4.1)$$

where

$$C_N(\omega/\omega_p) = \begin{cases} \cos[N \cos^{-1}(\omega/\omega_p)] , & |\omega| \leq \omega_p \\ \cosh[N \cosh^{-1}(\omega/\omega_p)] , & |\omega| \geq \omega_p \end{cases} \quad (4.2)$$

and ω_p is a frequency scaling constant, and ϵ is a constant that adjusts the influence of $C_N(\omega/\omega_p)$ in the denominator of $|H(j\omega)|^2$.

In due course it will be shown that (4.2) can be expressed as a polynomial, in fact a Chebyshev polynomial (see Section 4.4), and that as such (4.1) will satisfy the *Analog Filter Design Theorem*, and therefore the imposed constraints of Section 2.6 will be satisfied. It will be shown that N is the order of the Chebyshev polynomial, and in Section 4.5 it will be shown that N is the order of the filter, i.e., the number of poles of the transfer function $H(s)$. The form shown for $C_N(\omega/\omega_p)$ in (4.2) is very convenient for analytical investigation purposes, revealing the characteristics of the Chebyshev Type I response, and also yielding design formulae such as for the minimum required order to meet design specifications.

Note that $0 \leq C_N^2(\omega/\omega_p) \leq 1$, for $0 \leq \omega \leq \omega_p$, and $C_N^2(\omega/\omega_p) \geq 1$, for $\omega \geq \omega_p$. Therefore, $0 \leq \omega \leq \omega_p$ defines the passband, and $|H(j\omega)|^2$ monotonically falls off for $\omega > \omega_p$. Within the passband, as can be seen from (4.2), the magnitude-squared frequency response “ripples,” following the cosine function. However, the hyperbolic cosine function monotonically increases, causing the magnitude-squared response to monotonically decrease, beyond the passband.

It is easy to see that

$$|H(j\omega)|_{max}^2 = 1 \quad ,$$

and that

$$|H(j\omega)|_{min}^2 = \frac{1}{1 + \epsilon^2} \quad .$$

In terms of dB,

$$10 \log |H(j\omega)|_{max}^2 = 0 \quad ,$$

and

$$10 \log |H(j\omega)|_{min}^2 = -10 \log(1 + \epsilon^2) . \tag{4.3}$$

When (4.3) is compared with the general magnitude specifications for the design of a lowpass filter illustrated in **Figure 2.15** on page 52, setting A_p equal to the negative of (4.3) results in

$$\epsilon = \sqrt{10^{A_p/10} - 1} . \tag{4.4}$$

Several values of A_p and corresponding values of ϵ are shown in **Table 4.1**. It is common practice to restrict A_p to less than 3 dB, since a ripple greater than that would be excessive. Usually A_p is 2 dB or less, in practice. In the following, except for the 3 dB entry in **Table 4.1**, it is assumed that A_p is less than 3 dB.

Note that magnitude-squared response peaks occur in the passband when $C_N^2(\omega/\omega_p) = 0$. The frequencies of the peaks may be found as follows:

$$C_N^2(\omega/\omega_p) = \cos^2[N \cos^{-1}(\omega/\omega_p)] = 0 ,$$

from which

$$N \cos^{-1}(\omega/\omega_p) = \frac{\pi}{2} , \frac{3\pi}{2} , \frac{5\pi}{2} , \dots , \frac{k\pi}{2} , \quad k = 1, 3, 5, \dots .$$

Table 4.1

Several values of Chebyshev ripple factor, ϵ , versus passband attenuation specification, A_p .

$A_p, \text{ dB}$	ϵ
0.1	0.15262
0.2	0.21709
0.5	0.34931
1.0	0.50885
1.5	0.64229
2.0	0.76478
2.5	0.88220
3.0103	1.0

Therefore, the frequencies of the peaks are as follows:

$$\omega_k^{(peak)} = \omega_p \cos[(2k - 1)\pi/(2N)] , \quad k = 1, 2, 3, \dots, N_p, \quad (4.5)$$

where $N_p = (N+1)/2$ if N is odd, and $N_p = N/2$ if N is even.

Similarly, the magnitude-squared response valleys, i.e., the minimum values, occur in the passband when $C_N^2(\omega/\omega_p) = 1$. The frequencies of the valleys may be found to be as follows:

$$\omega_k^{(valley)} = \omega_p \cos[(k\pi)/N] , \quad k = 1, 2, 3, \dots, N_v, \quad (4.6)$$

where $N_v = (N-1)/2$ if N is odd, and $N_v = N/2$ if N is even. It is noted that the response is also equal to

$$|H(j\omega)|_{min}^2 = \frac{1}{1 + \epsilon^2}$$

when $\omega = \omega_p$, but this is not the frequency of a “valley,” but rather the edge of the passband, and therefore ω_p is not included in (4.6). The passband response is denoted as “equiripple” since all of the passband peaks are the same magnitude, and all of the passband valleys are of the same magnitude. It is noted that the frequency spacing between peaks, as well as between valleys, are *not* equal: it is the magnitudes of the peaks that are equal, and the magnitudes of the valleys.

See **Figure 4.1** for plots of (4.1) for a normalized ω_c of unity,¹ a somewhat arbitrary, but common value of $A_p = 1$ dB ($\epsilon = 0.50885$), and several values of N . Also see **Figure 4.2** for detailed plots of (4.1) across the passband. In the figures solid lines are for even orders, and dashed lines are for odd orders. Note that all odd orders have a DC response of unity (0 dB), and that all even orders have a DC response given by (4.3). This can be easily seen from (4.1), since $\cos[N \cos^{-1}(0)]$ is zero for N odd, and is ± 1 for N even. Note also the number of peaks and valleys across the passband, including the peak or valley at DC, equals the order.

Assuming that, as stated above, A_p is less than 3 dB, and defining ω_c , as is done in **Chapter 3**, as the 3dB corner frequency, it follows that $\omega_c > \omega_p$. Therefore, at $\omega = \omega_c$ the hyperbolic form of (4.2) applies. It follows that

$$\epsilon \cosh[N \cosh^{-1}(\omega_c/\omega_p)] = 1 ,$$

¹It is common to show plots of the magnitude response, as well as other plots, of Chebyshev filters with a normalized ω_p , which indeed is justifiable. However, so that Chebyshev plots may be more easily compared with those of other filters, such as Butterworth filters, ω_c will be normalized to unity here.

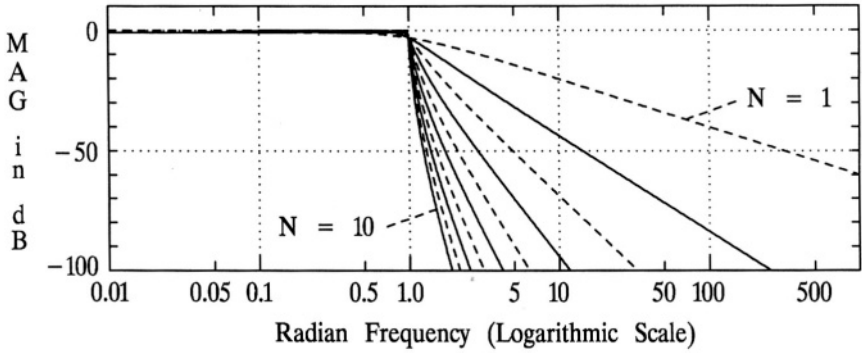


Figure 4.1 The Chebyshev Type I magnitude response. Plots of (4.1) for $\omega_c = 1$, $A_p = 1$ dB, and values of N from 1 through 10.

from which it follow that

$$\omega_c = \omega_p \cosh[(1/N) \cosh^{-1}(1/\epsilon)] . \tag{4.7}$$

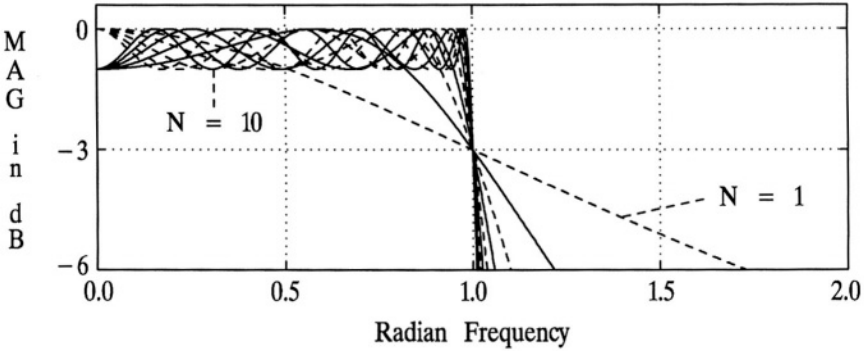


Figure 4.2 The Chebyshev Type I magnitude response. Passband details of (4.1) for $\omega_c = 1$, $A_p = 1$ dB, and values of N from 1 through 10.

Example 4.1

Suppose $N = 5$, $\omega_p = 1000 \text{ rad/s}$, and $\epsilon = 0.50885$ ($A_p = 1 \text{ dB}$), then, from (4.5), the frequencies of the peaks are 0, 587.79 rad/s, and 951.06 rad/s. From (4.6) the frequencies of the valleys are 309.02 rad/s and 809.02 rad/s. From (4.7), $\omega_c = 1033.81 \text{ rad/s}$. \square

4.2 FILTER SELECTIVITY AND SHAPING FACTOR

Applying (2.37), the definition of *Filter Selectivity*, to the square root of (4.1) results in

$$F_S = \frac{\epsilon^2 N}{\sqrt{\omega_c^2 - \omega_p^2}} \frac{\cosh [N \cosh^{-1}(\omega_c/\omega_p)] \sinh [N \cosh^{-1}(\omega_c/\omega_p)]}{(1 + \epsilon^2 \cosh^2 [N \cosh^{-1}(\omega_c/\omega_p)])^{3/2}}. \quad (4.8)$$

If (4.7) is used in (4.8) to eliminate any direct reference to ω_p , then (4.8) may be expressed as follows:

$$F_S = \frac{\epsilon N \cosh [(1/N) \cosh^{-1}(1/\epsilon)] \sinh [\cosh^{-1}(1/\epsilon)]}{2\sqrt{2} \omega_c \sinh [(1/N) \cosh^{-1}(1/\epsilon)]}. \quad (4.9)$$

Let A be an arbitrary attenuation in dB relative to the peak value greater than A_p . From (4.1):

$$A = 10 \log [1 + \epsilon^2 C_N^2(\omega/\omega_p)]. \quad (4.10)$$

For a given A , solving (4.10) for ω would be equivalent to solving for the bandwidth at that attenuation A :

$$BW = \omega_p \cosh [(1/N) \cosh^{-1}([1/\epsilon] \sqrt{10^{A/10} - 1})]. \quad (4.11)$$

Using (4.11) and applying (2.38), the definition of *Shaping Factor*, the Chebyshev Type I filter *Shaping Factor* may be readily found:

$$S_a^b = \frac{BW_b}{BW_a} = \frac{\cosh [(1/N) \cosh^{-1}([1/\epsilon] \sqrt{10^{b/10} - 1})]}{\cosh [(1/N) \cosh^{-1}([1/\epsilon] \sqrt{10^{a/10} - 1})]}. \quad (4.12)$$

Example 4.2

Suppose $a = 3 \text{ dB}$, $b = 80 \text{ dB}$, $\omega_c = 1$, and $\epsilon = 0.50885$. From (4.9), for $N = 1, 2, \dots, 10$, F_s may be computed to be 0.35, 1.07, 2.24, 3.89, 6.00, 8.58, 11.63, 15.15, 19.14 and 23.60 respectively. From (4.12), for N from 1 through 10, S_3^{80} may be computed to be 10000.0, 81.41, 15.54, 6.72, 4.07, 2.93, 2.34, 1.98, 1.76 and 1.60 respectively. \square

4.3 DETERMINATION OF ORDER

An important step in the design of an analog filter is determining the minimum required order to meet given specifications. Refer to **Figure 2.15** on page 52 in specifying the desired filter magnitude characteristics. As long as the filter magnitude frequency response fits within the acceptable corridor indicated in **Figure 2.15**, it satisfies the specifications. Note that A_p , the permitted magnitude deviation within the passband ($0 \leq \omega \leq \omega_p$), and that A_s , the minimum attenuation relative to the passband peak response within the stopband ($\omega \geq \omega_s$), are in dB.

Starting with (4.1):

$$-10 \log |H(j\omega_s)|^2 = A_s = 10 \log [1 + \epsilon^2 C_N^2(\omega_s/\omega_p)] \tag{4.13}$$

Temporarily let η , a real variable, assume the role of N , an integer, as is done in **Chapter 3**. Therefore, from (4.13):

$$\epsilon^2 \cosh^2 [\eta \cosh^{-1}(\omega_s/\omega_p)] = 10^{A_s/10} - 1 ,$$

from which

$$\eta = \frac{\cosh^{-1} \left[(1/\epsilon) \sqrt{10^{A_s/10} - 1} \right]}{\cosh^{-1}(\omega_s/\omega_p)} .$$

Letting $N = \lceil \eta \rceil$, where $\lceil \eta \rceil$ is the smallest integer equal to or larger than η ($\eta \leq \lceil \eta \rceil < \eta + 1$), the minimum order required to meet the specifications may be determined from the following:

$$N = \left\lceil \frac{\cosh^{-1} \left[\sqrt{10^{A_s/10} - 1} / \sqrt{10^{A_p/10} - 1} \right]}{\cosh^{-1}(\omega_s/\omega_p)} \right\rceil \tag{4.14}$$

Example 4.3

Suppose the following specifications are given: $f_p = 3,000 \text{ Hz}$, $f_s = 7,000 \text{ Hz}$, $A_p = 2 \text{ dB}$ ($\epsilon = 0.76478$), and $A_s = 60 \text{ dB}$. From the right side of (4.14), $\eta = 5.278$. Therefore, $N = 6$. \square

4.4 CHEBYSHEV POLYNOMIALS

Observe that

$$\cos[(N+1) \cos^{-1}(\omega/\omega_p)] = \cos[N \cos^{-1}(\omega/\omega_p) + \cos^{-1}(\omega/\omega_p)] , \quad (4.15)$$

and

$$\cosh[(N+1) \cosh^{-1}(\omega/\omega_p)] = \cosh[N \cosh^{-1}(\omega/\omega_p) + \cosh^{-1}(\omega/\omega_p)] .$$

Therefore,

$$C_{N+1}(\omega/\omega_p) = \begin{cases} \cos[N \cos^{-1}(\omega/\omega_p) + \cos^{-1}(\omega/\omega_p)] , & |\omega| \leq \omega_p \\ \cosh[N \cosh^{-1}(\omega/\omega_p) + \cosh^{-1}(\omega/\omega_p)] , & |\omega| \geq \omega_p \end{cases} ,$$

But,

$$\begin{aligned} \cos[N \cos^{-1}(\omega/\omega_p) + \cos^{-1}(\omega/\omega_p)] &= (\omega/\omega_p) \cos[N \cos^{-1}(\omega/\omega_p)] \\ &\quad - \sin[N \cos^{-1}(\omega/\omega_p)] \sin[\cos^{-1}(\omega/\omega_p)] , \end{aligned} \quad (4.16)$$

Also note that,

$$\begin{aligned} \cos[(N-1) \cos^{-1}(\omega/\omega_p)] &= \cos[N \cos^{-1}(\omega/\omega_p) - \cos^{-1}(\omega/\omega_p)] = \\ &(\omega/\omega_p) \cos[N \cos^{-1}(\omega/\omega_p)] + \sin[N \cos^{-1}(\omega/\omega_p)] \sin[\cos^{-1}(\omega/\omega_p)] . \end{aligned} \quad (4.17)$$

From (4.17) it follows that,

$$\begin{aligned} \sin[N \cos^{-1}(\omega/\omega_p)] \sin[\cos^{-1}(\omega/\omega_p)] &= \cos[(N-1) \cos^{-1}(\omega/\omega_p)] \\ &\quad - (\omega/\omega_p) \cos[N \cos^{-1}(\omega/\omega_p)] . \end{aligned} \quad (4.18)$$

Combining (4.15), (4.16) and (4.18),

$$\begin{aligned} \cos[(N+1) \cos^{-1}(\omega/\omega_p)] &= 2 (\omega/\omega_p) \cos[N \cos^{-1}(\omega/\omega_p)] \\ &\quad - \cos[(N-1) \cos^{-1}(\omega/\omega_p)] . \end{aligned} \quad (4.19)$$

Similarly,

$$\begin{aligned} \cosh[(N+1) \cosh^{-1}(\omega/\omega_p)] &= 2 (\omega/\omega_p) \cosh[N \cosh^{-1}(\omega/\omega_p)] \\ &\quad - \cosh[(N-1) \cosh^{-1}(\omega/\omega_p)] . \end{aligned} \quad (4.20)$$

Therefore, from (4.19) and (4.20), a general recursion can be stated for $C_N(\omega/\omega_p)$:

$$C_{N+1}(\omega/\omega_p) = 2 (\omega/\omega_p) C_N(\omega/\omega_p) - C_{N-1}(\omega/\omega_p) . \tag{4.21}$$

It is important to note that (4.21) applies equally to the cosine and hyperbolic forms of $C_N(\omega/\omega_p)$.

If $N = 0$, and for convenience ω_p is normalized to unity,

$$\cos[N \cos^{-1}(\omega)] = \cosh[N \cosh^{-1}(\omega)] = 1 \quad \forall \omega ,$$

and therefore

$$C_0(\omega) = 1 \quad \forall \omega .$$

If $N = 1$,

$$\cos[N \cos^{-1}(\omega)] = \cosh[N \cosh^{-1}(\omega)] = \omega ,$$

and therefore

$$C_1(\omega) = \omega \quad \forall \omega .$$

For $N > 1$ the recursion (4.21) may be used. For $N = 2$,

$$C_2(\omega) = 2 \omega C_1(\omega) - C_0(\omega) = 2 \omega^2 - 1 .$$

The function $C_2(\omega)$ expressed as $2 \omega^2 - 1$ rather than as in (4.2) with $N = 2$ and $\omega_p = 1$, is clearly a polynomial, and is numerically equal: as such it is referred to as a second-order Chebyshev polynomial. Several Chebyshev polynomials are shown in Table 4.2. Note that N is the order of the Chebyshev polynomial. Also note that if ω_p is not normalized to unity, the Chebyshev polynomial is as shown in Table 4.2 with ω replaced by ω/ω_p .

It can be shown that the square of all Chebyshev polynomials have only even powers of ω , and that multiplied by ϵ^2 and added to unity they have no real roots. That is, the *Analog Filter Design Theorem* is satisfied. For example, see Example 4.4.

Example 4.4

Suppose $N = 3$, $\omega_p = 100$, and $\epsilon = 0.5$. Then

$$C_3(\omega/100) = 4 (\omega/100)^3 - 3 (\omega/100) ,$$

and

$$|H(j\omega)|^2 = \frac{1}{4 (\omega/100)^6 - 6 (\omega/100)^4 + (9/4) (\omega/100)^2 + 1} .$$

A root of the denominator requires that

$$6 (\omega/100)^4 = 4 (\omega/100)^6 + (9/4) (\omega/100)^2 + 1 ,$$

Table 4.2
Chebyshev polynomials for several values of N .

N	Chebyshev Polynomial
1	ω
2	$2 \omega^2 - 1$
3	$4 \omega^3 - 3 \omega$
4	$8 \omega^4 - 8 \omega^2 + 1$
5	$16 \omega^5 - 20 \omega^3 + 5 \omega$
6	$32 \omega^6 - 48 \omega^4 + 18 \omega^2 - 1$
7	$64 \omega^7 - 112 \omega^5 + 56 \omega^3 - 7 \omega$
8	$128 \omega^8 - 256 \omega^6 + 160 \omega^4 - 32 \omega^2 + 1$

which has no real solution, i.e., no real roots. Therefore, the *Analog Filter Design Theorem* is satisfied: there is a corresponding $H(s)$ that meets all of the imposed constraints of **Section 2.6**. Therefore a circuit can be implemented with the desired third-order Chebyshev Type I response. □

4.5 POLE LOCATIONS

Starting with (4.1) and following the procedure used in **Section 2.7**:

$$Y(s) = H(s)H(-s) = \frac{1}{1 + \epsilon^2 C_N^2(-js/\omega_p)} \tag{4.22}$$

The poles of (4.22) may be found by setting

$$\epsilon^2 C_N^2(-js_k/\omega_p) = -1 ,$$

and solving for the values of s_k . First, it is noted that

$$C_N(-js_k/\omega_p) = \pm j/\epsilon .$$

Assuming that $\epsilon < 1$ ($A_p < 3$ dB), as noted above in **Section 4.1**, then $|\pm j/\epsilon| > 1$, and the hyperbolic form of C_N is perhaps the more appropriate:

$$C_N(-js_k/\omega_p) = \cosh[N \cosh^{-1}(-js_k/\omega_p)] = \pm j/\epsilon .$$

Note that if $\epsilon > 1$ is used, then the cosine form of C_N would perhaps be the more appropriate:² a development similar to that used here will yield the same results, i.e., the same equations for finding the poles. That is, even though $\epsilon < 1$ is used here, the results are valid for all real ϵ .

Note that $-js_k/\omega_p$ is a complex number. Let

$$-js_k/\omega_p = \cosh(w) ,$$

where

$$w = u + jv ,$$

and u and v are both real. Then

$$\begin{aligned} C_N(-js_k/\omega_p) &= \cosh(Nw) = \cosh(Nu + jNv) \\ &= \cosh(Nu) \cosh(jNv) + \sinh(Nu) \sinh(jNv) . \end{aligned} \tag{4.23}$$

Note that

$$\cosh(jNv) = \frac{e^{jNv} + e^{-jNv}}{2} = \cos(Nv) , \tag{4.24}$$

and

$$\sinh(jNv) = \frac{e^{jNv} - e^{-jNv}}{2} = j \sin(Nv) . \tag{4.25}$$

Using (4.24) and (4.25) in (4.23),

$$C_N(-js_k/\omega_p) = \cosh(Nu) \cos(Nv) + j \sinh(Nu) \sin(Nv) = \pm j/\epsilon . \tag{4.26}$$

By observation of (4.26),

$$\cosh(Nu) \cos(Nv) = 0 , \tag{4.27}$$

and

$$\sinh(Nu) \sin(Nv) = \pm \frac{1}{\epsilon} . \tag{4.28}$$

From (4.27), $\cos(Nv) = 0$, and therefore,

$$v = \frac{\pi}{2N} (2k - 1) , \quad k = 1, 2, 3, \dots, 2N . \tag{4.29}$$

At the values of v expressed in (4.29), $\sin(Nv) = \pm 1$, and therefore

$$u = \pm (1/N) \sinh^{-1}(1/\epsilon) . \tag{4.30}$$

Since $-js_k/\omega_p = \cosh(w)$,

²Since $\pm j/\epsilon$ is complex, either form, the cosine or the hyperbolic, is equally valid.

$$s_k = j\omega_p \cosh(w) = j\omega_p \cosh(u + jv) . \quad (4.31)$$

Using (4.29) and the positive form of (4.30) (to avoid redundancy) in (4.31),

$$\begin{aligned} s_k &= j\omega_p \cosh[(1/N) \sinh^{-1}(1/\epsilon) + j(\pi/[2N])(2k - 1)] \\ &= \omega_p \left\{ -\sinh[(1/N) \sinh^{-1}(1/\epsilon)] \sin[(\pi/[2N])(2k - 1)] \right. \\ &\quad \left. + j \cosh[(1/N) \sinh^{-1}(1/\epsilon)] \cos[(\pi/[2N])(2k - 1)] \right\} . \end{aligned} \quad (4.32)$$

For left-half plane poles:

$$\begin{aligned} s_k &= \sigma_k + j\omega_k , \\ \sigma_k &= -\omega_p \sinh[(1/N) \sinh^{-1}(1/\epsilon)] \sin[(\pi/[2N])(2k - 1)] , \\ \omega_k &= \omega_p \cosh[(1/N) \sinh^{-1}(1/\epsilon)] \cos[(\pi/[2N])(2k - 1)] , \\ &\quad k = 1, 2, 3, \dots, N . \end{aligned} \quad (4.33)$$

Example 4.5

Suppose $N = 4$, $\omega_p = 1000 \text{ rad/s}$, and $\epsilon = 0.5$. From (4.33),

$$s_k = -141.13 \pm j 984.71 , \quad -340.72 \pm j 407.88 .$$

From the poles, and also noting that the DC gain is $1/\sqrt{1 + \epsilon^2}$,

$$H(s) = \frac{2.5 \times 10^{11}}{(s^2 + 282.26s + 989,571.46)(s^2 + 681.44s + 282,456.21)} ,$$

or,

$$H(s) = \frac{2.5 \times 10^{11}}{s^4 + 963.7s^3 + 1,464,370.92s^2 + 754,059,665.5s + 2.795 \times 10^{11}} .$$

□

Note that, from (4.33),

$$\frac{\sigma_k^2}{\omega_p^2 \sinh^2[(1/N) \sinh^{-1}(1/\epsilon)]} = \sin^2[(\pi/[2N])(2k - 1)] , \quad (4.34)$$

and,

$$\frac{\omega_k^2}{\omega_p^2 \cosh^2[(1/N) \sinh^{-1}(1/\epsilon)]} = \cos^2[(\pi/[2N])(2k - 1)] . \quad (4.35)$$

Adding (4.34) and (4.35),

$$\frac{\sigma_k^2}{\omega_p^2 \sinh^2[(1/N) \sinh^{-1}(1/\epsilon)]} + \frac{\omega_k^2}{\omega_p^2 \cosh^2[(1/N) \sinh^{-1}(1/\epsilon)]} = 1. \quad (4.36)$$

If σ_k and ω_k are considered as continuous variables, then (4.36) describes an ellipse with axes that are the real and imaginary axes of the s plane. The *major* semiaxis of the ellipse has the value of

$$SA_{major} = \omega_p \cosh[(1/N) \sinh^{-1}(1/\epsilon)] , \quad (4.37)$$

and the *minor* semiaxis has the value of

$$SA_{minor} = \omega_p \sinh[(1/N) \sinh^{-1}(1/\epsilon)] , \quad (4.38)$$

and the *foci* are located at $\omega_k = \pm j \omega_p$. Therefore, the poles of a Chebyshev Type I filter fall on an ellipse described by (4.36).

Example 4.6

Consider the filter described in *Example 4.5*. The *major* semiaxis of the ellipse, obtained from (4.37), is 1065.84. The *minor* semiaxis of the ellipse, obtained from (4.38), is 368.8. The *foci* are located at $\pm j 1000$. The poles and the ellipse are shown in **Figure 4.3**. For symmetry, the right-half plane poles of $Y(s) = H(s)H(-s)$ are also shown. □

4.6 PHASE RESPONSE, PHASE DELAY, AND GROUP DELAY

A Chebyshev Type I filter, as seen above, is designed to meet given magnitude response specifications. Once the transfer function is determined, it may be put in the following form:

$$H(s) = \frac{K}{\sum_{i=0}^N a_i s^i} , \quad (4.39)$$

which is an all-pole form of (2.76). Given (4.39), the phase response, from (2.79), may be stated as follows:

$$\angle H(j\omega) = -\arctan \frac{I_d(\omega)}{R_d(\omega)} , \quad (4.40)$$

where $R_d(\omega)$ denotes the real part of the denominator of (4.39) evaluated with

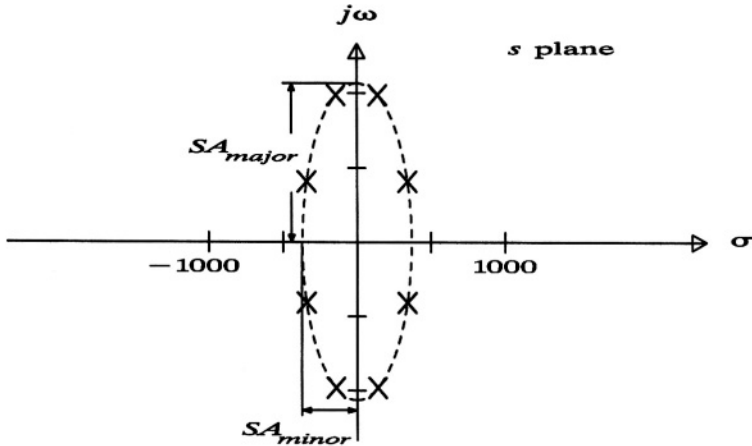


Figure 4.3 A plot of the poles of $H(s)H(-s)$ for $N = 4$, $\omega_p = 1000 \text{ rad/s}$, and $\epsilon = 0.5$. See *Examples 4.5 and 4.6*.

$s = j\omega$, and $I_d(\omega)$ denotes the imaginary part. The phase response of a Chebyshev Type I filter, with a normalized $\omega_c = 1$,³ a somewhat arbitrary, but common value of $A_p = 1 \text{ dB}$ ($\epsilon = 0.50885$), and several values of N , is shown in **Figure 4.4**. Taking the initial phase slope as a linear-phase reference, deviations from linear

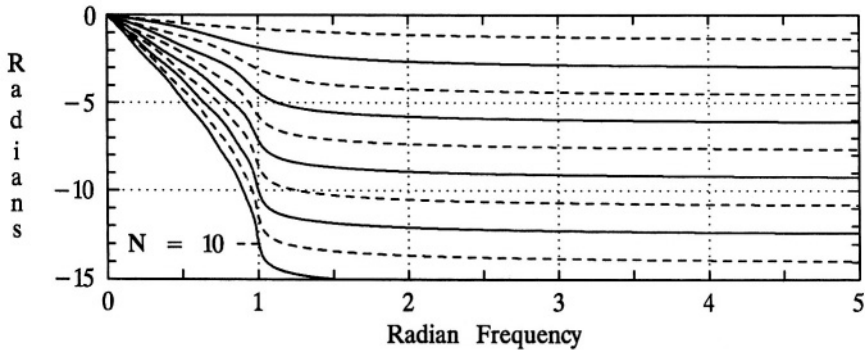


Figure 4.4 A plot of the phase response for a Chebyshev Type I filter with normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, and for values of N from 1 through 10.

³ See Footnote 1.

phase, for a normalized $\omega_c = 1$, $A_p = 1$ dB, and for several values of N , are shown in **Figure 4.5**. In the figure, solid lines are for even orders, and dashed lines are for odd orders.

The *phase delay*, $t_{pd}(\omega)$, for a filter is defined in (2.80), which is repeated here for convenience:

$$t_{pd}(\omega) = - \frac{\angle H(j\omega)}{\omega} . \tag{4.41}$$

Using (4.40) in (4.41), the phase delay for a lowpass Chebyshev Type I filter may be expressed as

$$t_{pd}(\omega) = \frac{1}{\omega} \arctan \frac{I_d(\omega)}{R_d(\omega)} . \tag{4.42}$$

The *group delay* for a filter, $t_{gd}(\omega)$, is defined by (2.81) and is repeated here for convenience:

$$t_{gd}(\omega) = - \frac{d \angle H(j\omega)}{d\omega} . \tag{4.43}$$

The phase delay of a Chebyshev Type I filter, with a normalized $\omega_c = 1$, $A_p = 1$ dB, and for several values of N , is shown in **Figure 4.6**. The group delay is shown in **Figure 4.7**. In both figures, solid lines are for even orders, and dashed lines are for odd orders. Note that the phase delay values at DC are very close to the group delay values at DC, and that for all $\omega < 0.5$ the phase delay and the group delay are comparable. However, as ω approaches the 3 dB corner frequency, in this case unity, the group delay becomes very large due to the nonlinearity of the phase response near the corner frequency (see **Figures 4.4** and **4.5**).

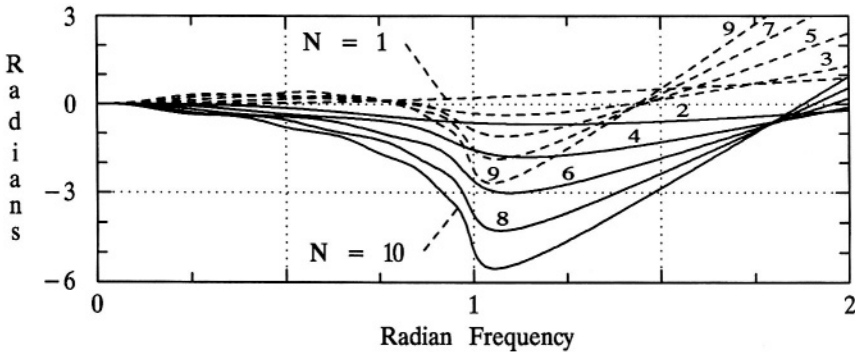


Figure 4.5 Phase deviation from linear for a Chebyshev Type I filter with normalized $\omega_c = 1$, $A_p = 1$ dB, and for values of N from 1 through 10.

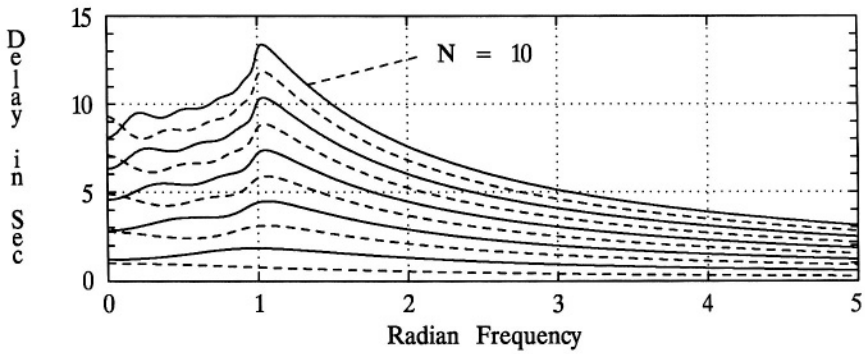


Figure 4.6 A plot of the phase delay for a Chebyshev Type I filter with normalized $\omega_c = 1$, $A_p = 1$ dB, and for values of N from 1 through 10.

4.7 TIME-DOMAIN RESPONSE

The unit impulse response of a Chebyshev Type I filter,⁴ with a normalized $\omega_c = 1$, $A_p = 1$ dB, and for several values of N , is shown in **Figure 4.8**. The unit step response of a Chebyshev Type I filter, with normalized $\omega_c = 1$, $A_p = 1$ dB,

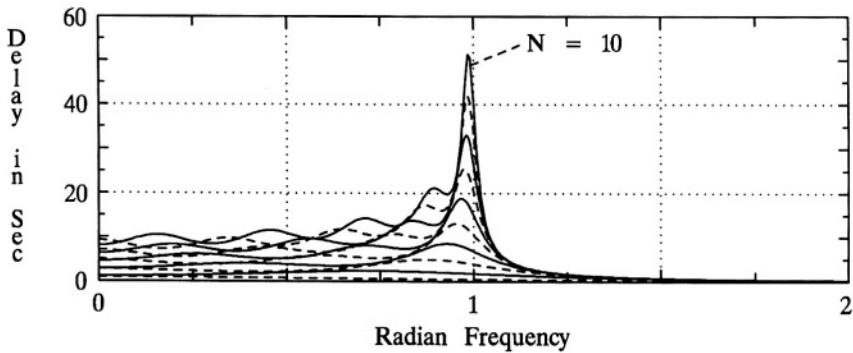


Figure 4.7 A plot of the group delay for a Chebyshev Type I filter with normalized $\omega_c = 1$, $A_p = 1$ dB, and for values of N from 1 through 10.

⁴See **Section 2.13** for a general introduction to time-domain responses.

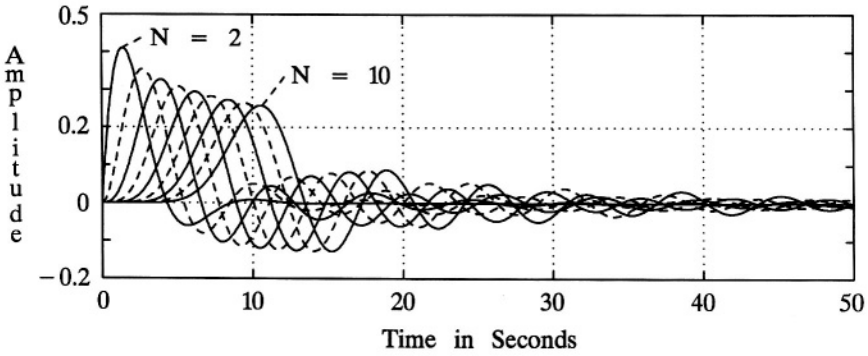


Figure 4.8 A plot of the unit impulse response for a Chebyshev Type I filter with normalized $\omega_c = 1$, $A_p = 1$ dB, and for values of N from 2 through 10.

and for several values of N , is shown in **Figure 4.9**. Note that the settled value of the step response is unity for odd orders, and $1/\sqrt{1 + \epsilon^2}$ (0.89125 in this case) for even orders; these values are, of course, the same as the *DC* gain of the filter, the input being unity.

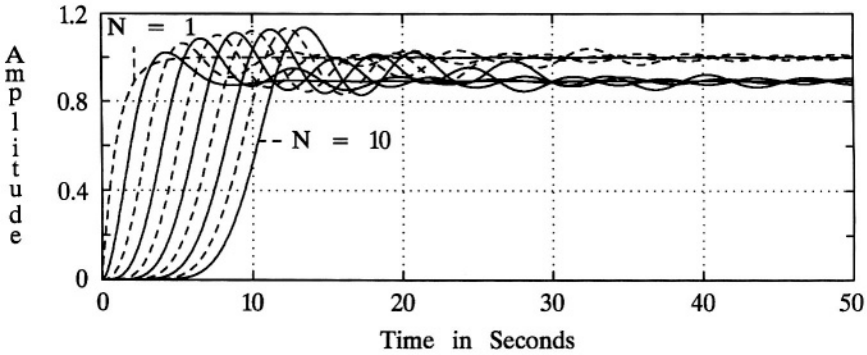


Figure 4.9 A plot of the unit step response for a Chebyshev Type I filter with normalized $\omega_c = 1$, $A_p = 1$ dB, and for values of N from 1 through 10.

4.8 COMPARISON WITH BUTTERWORTH FILTERS

There are several ways in which Chebyshev Type I filters may be compared with Butterworth filters. The magnitude frequency responses may be compared, the phase responses, the phase delays, the group delays, the unit impulse responses, and the unit step responses, for example.

It is noted that, for a given set of filter specifications, the minimum order required for a Chebyshev Type I response will never be greater than that required for a Butterworth response: it will frequently be less. It is not always less, since the orders are restricted to integers.

It can be shown that, for all practical values of ϵ , that the *Shaping Factor* of a Chebyshev Type I filter is always less than that for a Butterworth filter of the same order. Similarly, *Filter Selectivity* can be shown to be greater for a Chebyshev Type I response than for a Butterworth response of the same order. To illustrate this, compare the results of *Example 4.2* to that of *Example 3.1*; in both cases $a = 6\text{dB}$ and $b = 60\text{dB}$. For example, for $N = 8$, S_6^{60} for the Butterworth filter is 2.2, whereas it is only 1.54 for the Chebyshev Type I filter, which indicates that the attenuation in the transition band increases more rapidly as a function of ω for the Chebyshev filter than it does for the Butterworth. This, of course, can readily be observed by comparing *Figures 3.1* and *4.1*.

The above may imply that a Chebyshev Type I filter is superior to a Butterworth filter of the same order. However, the phase response of a Butterworth filter is more nearly linear than is that of a Chebyshev Type I filter of the same order. This is, of course, reflected in the phase delay and the group delay. By comparing the maximum group delay, near the cutoff frequency, of the Chebyshev filter to that of the Butterworth filter indicates a much greater value for the Chebyshev, due, of course, to the greater nonlinearity of the phase response for the Chebyshev.

Also, the unit impulse response and the unit step response of a Butterworth filter are more desirable than that of a Chebyshev Type I filter of the same order. The Chebyshev filter exhibits ringing that dies out much more slowly than that for the Butterworth filter.

Therefore, it is not possible to state emphatically that one filter type is superior to the other, without first stating very clearly the design objectives. Perhaps a *minimum* order is of low priority, as long as it is less than some given N_{max} , and approximating linear phase is of high priority; then a Butterworth response would be the better choice. On the other hand, if minimum order is of high priority and the phase response is more-or-less irrelevant, then a Chebyshev Type I response would likely be the better choice.

Note that by combining (4.7) and (4.37), the *major* semiaxis of the s -plane ellipse that the poles fall on may be expressed as follows:

$$SA_{major} = \omega_c \frac{\cosh[(1/N) \sinh^{-1}(1/\epsilon)]}{\cosh[(1/N) \cosh^{-1}(1/\epsilon)]} . \tag{4.44}$$

Note that

$$\cosh^{-1}(x) = \ln(x + \sqrt{x^2 - 1}) ,$$

and

$$\sinh^{-1}(x) = \ln(x + \sqrt{x^2 + 1}) .$$

It is noted that for large x , $\cosh^{-1}(x) \cong \sinh^{-1}(x)$. Therefore, as $\epsilon \rightarrow 0$,

$$SA_{major} \rightarrow \omega_c .$$

Note that by combining (4.7) and (4.38), the *minor* semiaxis of the s -plane ellipse that the poles fall on may be expressed as follows:

$$SA_{minor} = \omega_c \frac{\sinh[(1/N) \sinh^{-1}(1/\epsilon)]}{\cosh[(1/N) \cosh^{-1}(1/\epsilon)]} . \tag{4.45}$$

Note that

$$\cosh(x) = \frac{e^x + e^{-x}}{2} ,$$

and

$$\sinh(x) = \frac{e^x - e^{-x}}{2} .$$

It is noted that for large x , $\cosh(x) \cong \sinh(x)$. Therefore, as $\epsilon \rightarrow 0$,

$$SA_{minor} \rightarrow \omega_c .$$

Therefore, as $\epsilon \rightarrow 0$, the ellipse approaches a circle of radius ω_c . Similarly, from (4.30), in the limit as $\epsilon \rightarrow 0$,

$$\sigma_k = -\omega_c \sin[(\pi/[2N])(2k - 1)] ,$$

and

$$\omega_k = \omega_c \cos[(\pi/[2N])(2k - 1)] ,$$

which are identical to Butterworth poles (see (3.21)).

Therefore, in the limit as $\epsilon \rightarrow 0$, the Chebyshev Type I response, and poles, become identical to the Butterworth. Or, the Butterworth filter is a special case (a limiting case) of the Chebyshev Type I.

4.9 CHAPTER 4 PROBLEMS

- 4.1 Given the defining equations for a Chebyshev Type I response, (4.1) and (4.2), and given that $\omega_c = 1000 \text{ rad/s}$, $A_p = 1.8 \text{ dB}$, and $N = 3$:
- Determine the value of ϵ .
 - Determine the value of ω_p .
 - Determine the value of $|H(j\omega)|_{min}^2$.
 - Determine the frequencies of the peaks in the passband.
 - Determine the frequencies of the valleys in the passband.
 - Accurately sketch the magnitude frequency response. Use only a calculator for the necessary calculation. Use a vertical scale in dB (0 to -50dB), and a linear radian frequency scale from 0 to 5000 rad/s .
 - Accurately sketch the magnitude frequency response. Use only a calculator for the necessary calculations. Use a linear vertical scale from 0 to 1, and a linear radian frequency scale from 0 to 2000 rad/s .
- 4.2 Given the defining equations for a Chebyshev Type I response, (4.1) and (4.2), and given that $\omega_c = 1000 \text{ rad/s}$, $A_p = 1.5 \text{ dB}$, and $N = 6$:
- Determine the value of ϵ .
 - Determine the value of ω_p .
 - Determine the value of $|H(j\omega)|_{min}^2$.
 - Determine the frequencies of the peaks in the passband.
 - Determine the frequencies of the valleys in the passband.
 - Accurately sketch the magnitude frequency response. Use only a calculator for the necessary calculations. Use a vertical scale in dB (0 to -100dB), and a linear radian frequency scale from 0 to 5000 rad/s .
 - Accurately sketch the magnitude frequency response. Use only a calculator for the necessary calculations. Use a linear vertical scale from 0 to 1, and a linear radian frequency scale from 0 to 1200 rad/s .
- 4.3 In Figure 4.1 it may appear that the response fall-off is greater than $N \times 20 \text{ dB/decade}$. However, this is not the case for very high frequencies. Demonstrate that for very high frequencies the fall-off is indeed $-N \times 20 \text{ dB/decade}$.
- 4.4 Determine the value of *Filter Selectivity* for the Chebyshev Type I filter specified in Problem 4.2. Compare this value with the *Filter Selectivity* for a Butterworth filter with similar specifications: $\omega_c = 1000 \text{ rad/s}$, and $N = 6$.

- 4.5** Determine the value of the *Shaping Factor* for the Chebyshev Type I filter specified in **Problem 4.2**, for $a = 3 \text{ dB}$ and $b = 80 \text{ dB}$. Compare this value with the *Shaping Factor* for a Butterworth filter with similar specifications: $\omega_c = 1000 \text{ rad/s}$, and $N = 6$.
- 4.6** Estimate the *Shaping Factor* for a Chebyshev Type I filter with $N = 5$, $\omega_c = 1$, and $A_p = 1 \text{ dB}$, from **Figure 4.1**. Compare your results with that obtained from (4.12).
- 4.7** Suppose filter specifications are stated as follows: $f_c = 3500 \text{ Hz}$, $f_s = 7000 \text{ Hz}$, and $A_s = 60 \text{ dB}$. Note that (4.14) requires knowledge of ω_p to determine the required order, and when it is ω_c that is specified the order is required to determine ω_p from ω_c via (4.7). However, it is very practical to use (4.14), using ω_c in place of ω_p . This is justified since, (1) in practical problems there is usually not much difference between ω_c and ω_p , especially for higher order filters, (2) since the orders are restricted to integers it seldom makes any difference, and (3) since $\omega_c > \omega_p$, any error will result in exceeding the specifications rather than failing to meet the specifications. Making use of (4.14), determine the required filter order to meet the given specifications:
- For a Chebyshev Type I filter with 0.1 dB of ripple.
 - For a Chebyshev Type I filter with 0.5 dB of ripple.
 - For a Chebyshev Type I filter with 1.0 dB of ripple.
 - For a Chebyshev Type I filter with 1.5 dB of ripple.
 - For a Chebyshev Type I filter with 2.0 dB of ripple.
 - For a Chebyshev Type I filter with 2.5 dB of ripple.
 - For comparison purposes, for a Butterworth filter.
- 4.8** Given that $N = 4$, $\omega_p = 10$, and $\epsilon = 0.4$, express $|H(j\omega)|^2$ in polynomial form similar to **Example 4.4**, and demonstrate that it satisfies the *Analog Filter Design Theorem*.
- 4.9** Prove that the magnitude-squared frequency response of **Problem 4.1** satisfies the *Analog Filter Design Theorem*.
- 4.10** Prove that the magnitude-squared frequency response of **Problem 4.2** satisfies the *Analog Filter Design Theorem*.
- 4.11** Determine the 9-th and 10-th order Chebyshev polynomials.

- 4.12** Sketch the square of the 4-th order Chebyshev polynomial given in **Table 4.2** for ω from 0 to 1.1 *rad/s*. Compute the square of (4.2) over this same radian frequency range for $\omega_p = 1$, and verify that it is numerically the same as the Chebyshev polynomial.
- 4.13** Determine the poles of the filter specified in **Problem 4.7** (a).
- 4.14** Determine the poles of the filter specified in **Problem 4.7** (f).
- 4.15** Determine the poles of the filter specified in **Problem 4.8**.
- 4.16** Determine the transfer function $H(s)$ for the Chebyshev Type I filter specified in **Problem 4.1**. Express the denominator of $H(s)$ in two ways: (1) As a polynomial in s . (2) As the product of a second-order polynomial in s , the roots of which being complex conjugates, and a first-order term. State the numerical values of the three poles. Sketch the six poles of $H(-s)H(s)$ on an s plane, and include a sketch of the ellipse that the poles fall on. State the numerical values of the major and minor semiaxes of the ellipse.
- 4.17** Determine the transfer function $H(s)$ for the Chebyshev Type I filter specified in **Problem 4.2**. Express the denominator of $H(s)$ in two ways: (1) As a polynomial in s . (2) As the product of three second-order polynomials in s , the roots of each second-order polynomial being complex conjugates. State the numerical values of the six poles. Sketch the twelve poles of $H(-s)H(s)$ on an s plane, and include a sketch of the ellipse that the poles fall on. State the numerical values of the major and minor semiaxes of the ellipse.
- 4.18** Suppose an anti-aliasing filter is needed prior to an analog-to-digital converter to be used in a speech processing system. Suppose a sampling rate of 8000 *samples/s* is to be used, and therefore it is decided that the anti-aliasing filter should have a minimum attenuation of 60 *dB* at 4000 *Hz*. Suppose a Chebyshev Type I filter is to be used.
- If $f_p = 3000$ *Hz* and $A_p = 1$ *dB*, what is the minimum order required?
 - If $A_p = 1$ *dB* and $N = 10$, what is maximum value of f_p ?
 - If $f_p = 3000$ *Hz* and $N = 10$, what is the minimum value of A_p ?
- 4.19** Under the conditions of part (c) of **Problem 4.18**, determine the transfer function $H(s)$, and give numerical values for all the poles.
- 4.20** Sketch the step response of a 10-th order Chebyshev Type I filter with $f_c = 1000$ *Hz* and $A_p = 1$ *dB*. Refer to **Figure 4.9** and make use of the

scaling property of Fourier transforms. What would the maximum group delay be for this filter, and at what frequency would it occur? At what time would the peak of the unit impulse response of this filter be, and what would be the value of that peak?

4.21 Using the MATLAB functions *cheblap*, *impulse* and *step*:

- (a) Determine the transfer function in polynomial form, and also factored to indicate the poles, of a Chebyshev Type I filter with $\omega_p = 1$, $A_p = 1.2 \text{ dB}$, and $N = 6$.
- (b) Determine the impulse response and the step response for the filter of part (a).
- (c) By multiplying the pole vector found in part (a) by $2\pi 1000$ determine the transfer function of a Chebyshev Type I filter with $f_p = 1000 \text{ Hz}$, $A_p = 1.2 \text{ dB}$, and $N = 6$.
- (d) Determine and plot the magnitude frequency response of the filter of part (c) by using the MATLAB function *freqs*. Use a vertical scale in *dB* and a linear horizontal scale from 0 to 5000 *Hz*. Also determine and plot the phase response over this same frequency range. Use the MATLAB function *unwrap* to display the smooth phase response rather than the principle phase.
- (e) By appropriately scaling the impulse response and the step response of part (b), determine and plot the impulse response and the step response of the filter of part (c). That is, the time axis for the step response needs to be scaled by $1/(2\pi 1000)$, and the unit impulse response needs the same time-axis scaling and requires an amplitude scaling of $2\pi 1000$.
- (f) Determine and plot the phase delay of the filter of part (c). Note that this is easily obtained from the phase response of part (d).
- (g) Determine and plot the group delay of the filter of part (c). Note that this also is easily obtained from the phase response of part (d): $t_{gd}(n) \cong -[\Phi(n) - \Phi(n-1)]/S_s$, where $\Phi(n)$ is the phase in *radians* at step n , and S_s is the step size in *rad/s*.

CHAPTER 5

Chebyshev Type II Filters

Chebyshev Type II filters are closely related to Chebyshev Type I filters, and are noted for having a flat passband magnitude response, and an equiripple response in the stopband. As was noted in **Chapter 4**, the Chebyshev Type I response is often simply referred to as the *Chebyshev* response. Similarly, the Chebyshev Type II response is often referred to as the *Inverse Chebyshev* response, for reasons that will become clear as the response is developed below.

In this chapter the Chebyshev Type II response is defined, and it will be observed that it satisfies the *Analog Filter Design Theorem*. Explicit formulas for the design and analysis of Chebyshev Type II filters, such as *Filter Selectivity*, *Shaping Factor*, the minimum required order to meet design specifications, etc., will be obtained. From the defining $|H(j\omega)|^2$ the corresponding $H(s)$ will be determined, and means for determining the filter poles and zeros are found. To complete the study of lowpass, prototype Chebyshev Type II filters, the phase response, phase delay, group delay, and time-domain response characteristics are investigated.

5.1 EQUIRRIPLE STOPBAND MAGNITUDE

Suppose that $|G(j\omega)|^2$ is the magnitude-squared frequency response of a Chebyshev Type I filter according to (4.1):

$$|G(j\omega)|^2 = \frac{1}{1 + \varepsilon^2 C_N^2(\omega/\omega_p)} \quad (5.1)$$

Let $|F(j\omega)|^2$ be obtained from (5.1) as follows:

$$|F(j\omega)|^2 = 1 - |G(j\omega)|^2 = \frac{\varepsilon^2 C_N^2(\omega/\omega_p)}{1 + \varepsilon^2 C_N^2(\omega/\omega_p)} \quad (5.2)$$

Finally, the desired magnitude-squared response is obtained by replacing ω/ω_p by ω_s/ω in (5.2):

Definition of the magnitude-squared Chebyshev Type II response:

$$|H(j\omega)|^2 = \frac{\epsilon^2 C_N^2(\omega_s/\omega)}{1 + \epsilon^2 C_N^2(\omega_s/\omega)} \quad (5.3)$$

where

$$C_N(\omega_s/\omega) = \begin{cases} \cos[N \cos^{-1}(\omega_s/\omega)] , & |\omega| \geq \omega_s \\ \cosh[N \cosh^{-1}(\omega_s/\omega)] , & |\omega| \leq \omega_s \end{cases} \quad (5.4)$$

and ω_s is a frequency scaling constant, and ϵ is a constant that adjusts the influence of $C_N(\omega_s/\omega)$ in the denominator of $|H(j\omega)|^2$. Therefore, it is observed that the hyperbolic cosine is used in (5.4) for low frequencies, and, from (5.3) that this results in a response near unity; the trigonometric cosine is used for high frequencies beyond ω_s resulting in a rippling response of small magnitude.

In due course it will be shown that (5.4) can be expressed as a polynomial, in fact very closely related to the Chebyshev polynomials of Section 4.4, and that as such (5.3) will satisfy the *Analog Filter Design Theorem*, and therefore the imposed constraints of Section 2.6 will be satisfied. It will be shown that N is the order of the Chebyshev polynomial, and in Section 5.5 it will be shown that N is the order of the filter, i.e., the number of poles of the transfer function $H(s)$. The form shown for $C_N(\omega_s/\omega)$ in (5.4) is very convenient for analytical investigation purposes, revealing the characteristics of the Chebyshev Type II response, and also yielding design formulae such as for the minimum required order to meet design specifications.

Note that $0 \leq C_N^2(\omega_s/\omega) \leq 1$, for $\omega \geq \omega_s$, and $C_N^2(\omega_s/\omega) \geq 1$, for $0 \leq \omega \leq \omega_s$. Therefore, $\omega \geq \omega_s$ defines the stopband, and $|H(j\omega)|^2$ ripples within the stopband following the cosine function. Within the passband, as can be seen from (5.3) and (5.4), the magnitude-squared frequency response follows the hyperbolic cosine function and falls off monotonically for increasing ω .

It is easy to see that

$$|H(j\omega)|_{\omega=0}^2 = 1 ,$$

independent of N , and that

$$|H(j\omega)|_{\omega=\omega_s}^2 = \frac{\epsilon^2}{1 + \epsilon^2} .$$

In terms of dB ,

$$10 \log |H(j\omega)|_{\omega=0} = 0 ,$$

and

$$10 \log |H(j\omega)|_{\omega=\omega_s} = 10 \log [\epsilon^2 / (1 + \epsilon^2)] . \tag{5.5}$$

Note that (5.5) is the minimum attenuation for all $\omega \geq \omega_s$. When (5.5) is compared with the general magnitude specifications for the design of a lowpass filter illustrated in Figure 2.15 on page 52, setting A_s equal to the negative of (5.5) results in

$$\epsilon = \frac{1}{\sqrt{10^{A_s/10} - 1}} . \tag{5.6}$$

Several values of A_s and corresponding values of ϵ are shown in Table 5.1. Note that A_s is the minimum attenuation in the stopband. At frequencies where the numerator of (5.3) is zero, the attenuation is infinity.

Note that the magnitude-squared response of (5.3) is zero in the stopband when $C_N^2(\omega_s/\omega) = 0$. The frequencies where the response is zero may be found as follows:

$$C_N^2(\omega_s/\omega) = \cos^2 [N \cos^{-1}(\omega_s/\omega)] = 0 ,$$

from which

$$N \cos^{-1}(\omega_s/\omega) = \frac{\pi}{2}, \frac{3\pi}{2}, \frac{5\pi}{2}, \dots, \frac{k\pi}{2}, \quad k = 1, 3, 5, \dots .$$

Table 5.1

Several values of Chebyshev Type II ripple factor, ϵ , versus minimum stopband attenuation specification, A_s .

$A_s, \text{ dB}$	ϵ
20	0.1005038
30	0.0316386
40	0.0100005
50	0.0031623
60	0.0010000

Therefore, the frequencies where the response is zero are as follows:

$$\omega_k^{(zero)} = \omega_s / \cos[(2k - 1)\pi / (2N)] , \quad k = 1, 2, 3, \dots, N_p, \quad (5.7)$$

where $N_p = (N + 1) / 2$ if N is odd, and $N_p = N / 2$ if N is even. Note that if N is odd the highest frequency where the response is zero is infinity: there are only $(N - 1) / 2$ finite frequencies where the response is zero. If N is even there are $N / 2$ finite frequencies where the response is zero.

Similarly, the attenuation equals A_s in the stopband when $C_N^2(\omega_s / \omega) = 1$. The frequencies of these minimum attenuation points may be found as follows:

$$\omega_k^{(min)} = \omega_s / \cos[(k\pi) / N] , \quad k = 1, 2, 3, \dots, N_v, \quad (5.8)$$

where $N_v = (N - 1) / 2$ if N is odd, and $N_v = N / 2$ if N is even. Note that if N is even the highest frequency where the attenuation is equal to A_s is infinity.

The stopband response is denoted as “equiripple” since all of the stopband peaks (the points of minimum attenuation) are the same magnitude. It is noted that the frequency spacing between peaks are *not* equal: it is the magnitudes of the peaks that are equal.

The frequency at which the attenuation is equal to a given A_p may be found from (5.3):

$$A_p = 10 \log \left(\frac{1 + \epsilon^2 C_N^2(\omega_s / \omega_p)}{\epsilon^2 C_N^2(\omega_s / \omega_p)} \right) ,$$

and then solve for ω_p , making use of the hyperbolic form of (5.4):

$$\omega_p = \omega_s / \cosh[(1/N) \cosh^{-1}(1 / [\epsilon \sqrt{10^{A_p/10}} - 1])] . \quad (5.9)$$

See **Figure 5.1** for plots of (5.3) for a normalized ω_c of unity ($A_p = 3$ dB), $A_s = 80$ dB, and several values of N . Recall that Butterworth and Chebyshev Type I filters both have magnitude frequency responses that monotonically decrease with increasing frequency throughout the transition band and the stopband. However, due to the rippling in the stopband, this is not the case for Chebyshev Type II filters, as can be seen in **Figure 5.1**. Also, as can be seen from (5.3), $|H(j\omega)|^2_{\omega \rightarrow \infty}$ is zero

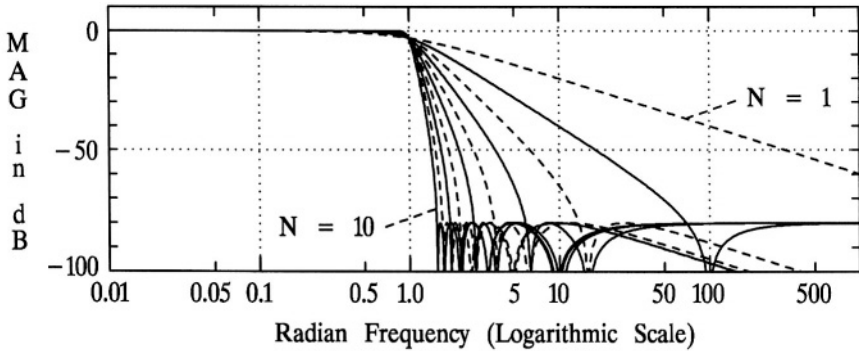


Figure 5.1 The Chebyshev Type II magnitude response. Plots of (5.3) for $\omega_c = 1$, $A_s = 80 \text{ dB}$, and values of N from 1 through 10.

if N is odd, and is $\epsilon^2/(1 + \epsilon^2)$ if N is even. Even though the frequency range does not go to infinity in **Figure 5.1**, this phenomenon is observable.

See **Figure 5.2** for detailed plots of (5.3) across the passband. Note that the passband magnitude response is very flat. It is very comparable to the Butterworth passband magnitude response shown in **Figure 3.2**. In fact, for a large range of ϵ it is superior: see **Sections 5.8** and **5.9**. In **Figures 5.1** and **5.2**, solid lines are for even orders, and dashed lines are for odd orders.

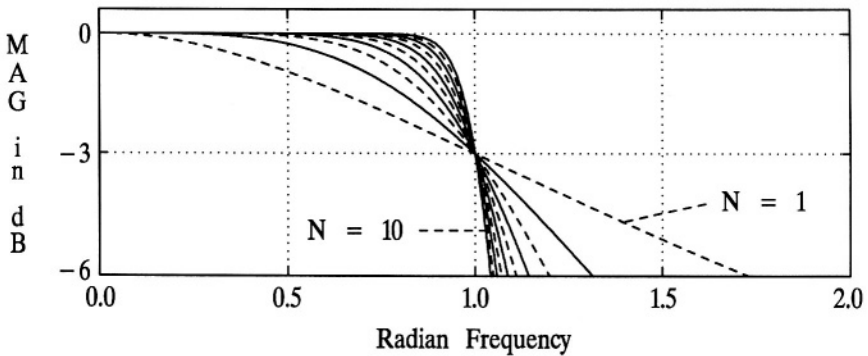


Figure 5.2 The Chebyshev Type II magnitude response. Passband details of (5.3) for $\omega_c = 1$, $A_s = 80 \text{ dB}$, and values of N from 1 through 10.

Example 5.1

Suppose $N = 5$, $\omega_s = 1000 \text{ rad/s}$, and $\epsilon = 0.001$ ($A_s = 60 \text{ dB}$), then, from (5.7), the frequencies where the magnitude frequency response is zero are 1051.46 rad/s , 1701.3 rad/s , and infinity. From (5.8) the frequencies where the attenuation in the stopband ripples to a minimum of A_s are 1236.07 rad/s and 3236.07 rad/s . From (5.9), $\omega_c = 417.39 \text{ rad/s}$. \square

5.2 FILTER SELECTIVITY AND SHAPING FACTOR

Applying (2.37), the definition of *Filter Selectivity*, to the square root of (5.3) results in

$$F_s = \frac{\epsilon N \omega_s}{\omega_c \sqrt{\omega_s^2 - \omega_c^2}} \frac{\sinh [N \cosh^{-1}(\omega_s/\omega_c)]}{(1 + \epsilon^2 \cosh^2 [N \cosh^{-1}(\omega_s/\omega_c)])^{3/2}} \quad (5.10)$$

If (5.9) is used, with $A_p = 3 \text{ dB}$, and therefore $\omega_p = \omega_c$, in (5.10) to eliminate any direct reference to ω_s , then (5.10) may be expressed as follows:

$$F_s = \frac{\epsilon N \cosh[(1/N) \cosh^{-1}(1/\epsilon)] \sinh[\cosh^{-1}(1/\epsilon)]}{2\sqrt{2} \omega_c \sinh[(1/N) \cosh^{-1}(1/\epsilon)]} \quad (5.11)$$

It is noted that (5.11) is identical to (4.9).

Let A be an arbitrary attenuation in dB relative to the DC value such that $0 < A \leq A_s$. From (5.3):

$$A = 10 \log \left\{ \left[1 + \epsilon^2 C_N^2(\omega_s/\omega) \right] / \left[\epsilon^2 C_N^2(\omega_s/\omega) \right] \right\} \quad (5.12)$$

For a given A , solving (5.12) for ω would be equivalent to solving for the bandwidth at that attenuation A :

$$BW = \omega_s / \cosh \left[(1/N) \cosh^{-1} \left(1 / \left\{ \epsilon \sqrt{10^{A/10} - 1} \right\} \right) \right] \quad (5.13)$$

Using (5.13) and applying (2.38), the definition of *Shaping Factor*, the Chebyshev Type II filter *Shaping Factor* may be readily found:

$$S_a^b = \frac{BW_b}{BW_a} = \frac{\cosh \left[(1/N) \cosh^{-1} \left(1 / \left\{ \epsilon \sqrt{10^{a/10} - 1} \right\} \right) \right]}{\cosh \left[(1/N) \cosh^{-1} \left(1 / \left\{ \epsilon \sqrt{10^{b/10} - 1} \right\} \right) \right]} \quad (5.14)$$

Example 5.2

Suppose $a = 3 \text{ dB}$, $b = 80 \text{ dB}$, $\omega_c = 1$, and $\epsilon = 0.0001$ ($A_s = 80 \text{ dB}$). From (5.11), for $N = 1, 2, \dots, 10$, F_s may be computed to be 0.35, 0.71, 1.06, 1.43, 1.84, 2.28, 2.79, 3.35, 3.97 and 4.67 respectively. From (5.14), for N from 1 through 10, S_3^{80} may be computed to be 10000.0, 70.71, 13.59, 5.99, 3.69, 2.70, 2.18, 1.87, 1.67 and 1.53 respectively. \square

5.3 DETERMINATION OF ORDER

An important step in the design of an analog filter is determining the minimum required order to meet given specifications. Refer to **Figure 2.15** on page 52 in specifying the desired filter magnitude characteristics. As long as the filter magnitude frequency response fits within the acceptable corridor indicated in **Figure 2.15**, it satisfies the specifications.

Starting with (5.3):

$$-10 \log |H(j\omega_p)|^2 = A_p = 10 \log [\{ 1 + \epsilon^2 C_N^2(\omega_s/\omega_p) \} / \epsilon^2 C_N^2(\omega_s/\omega_p)] . \quad (5.15)$$

Temporarily let η , a real variable, assume the role of N , an integer, as is done in **Chapters 3** and **4**. Therefore, from (5.15):

$$1 / \epsilon^2 \cosh^2 [\eta \cosh^{-1}(\omega_s/\omega_p)] = 10^{A_p/10} - 1 ,$$

from which, making use of (5.6),

$$\eta = \frac{\cosh^{-1} \left[\sqrt{10^{A_s/10} - 1} / \sqrt{10^{A_p/10} - 1} \right]}{\cosh^{-1}(\omega_s/\omega_p)} .$$

Letting $N = \lceil \eta \rceil$, where $\lceil \eta \rceil$ is the smallest integer equal to or larger than η ($\eta \leq \lceil \eta \rceil < \eta + 1$), the minimum order required to meet the specifications may be determined from the following:

$$N = \left\lceil \frac{\cosh^{-1} \left[\sqrt{10^{A_s/10} - 1} / \sqrt{10^{A_p/10} - 1} \right]}{\cosh^{-1}(\omega_s/\omega_p)} \right\rceil . \quad (5.16)$$

Note that (5.16) is identical to (4.14): for the same specifications, the minimum order required for a Chebyshev Type II filter is the same as that for a Chebyshev Type I filter.

Example 5.3

Suppose the following specifications are given: $f_p = 3,000 \text{ Hz}$, $f_s = 7,000 \text{ Hz}$, $A_p = 2 \text{ dB}$, and $A_s = 60 \text{ dB}$. From the right side of (5.16), $\eta = 5.278$. Therefore, $N = 6$. \square

5.4 INVERSE CHEBYSHEV POLYNOMIALS

A recursion for (5.4) may readily be developed, similar to that which was done in Section 4.4 for Chebyshev Type I filters. The resultant recursion is similar to (4.21):

$$C_{N+1}(\omega_s/\omega) = 2(\omega_s/\omega) C_N(\omega_s/\omega) - C_{N-1}(\omega_s/\omega) . \quad (5.17)$$

It is important to note that (5.17) applies equally to the cosine and hyperbolic forms of $C_N(\omega_s/\omega)$.

If $N = 0$, and for convenience ω_s is normalized to unity,

$$\cos[N \cos^{-1}(1/\omega)] = \cosh[N \cosh^{-1}(1/\omega)] = 1 \quad \forall \omega ,$$

and therefore

$$C_0(1/\omega) = 1 \quad \forall \omega .$$

If $N = 1$,

$$\cos[N \cos^{-1}(1/\omega)] = \cosh[N \cosh^{-1}(1/\omega)] = 1/\omega ,$$

and therefore

$$C_1(1/\omega) = 1/\omega \quad \forall \omega .$$

For $N > 1$ the recursion (5.17) may be used. For $N = 2$,

$$C_2(1/\omega) = (2/\omega) C_1(1/\omega) - C_0(1/\omega) = (2/\omega^2) - 1 .$$

Several Chebyshev polynomials are shown in Table 4.2. Inverse Chebyshev polynomials may be obtained from Table 4.2 by replacing ω with $1/\omega$. Note that if ω_s is not normalized to unity, the inverse Chebyshev polynomial is as shown in Table 4.2 with ω replaced by ω_s/ω .

It can be shown that the square of all inverse Chebyshev polynomials have only even powers of ω , and that multiplied by ϵ^2 and added to unity they have no real roots. If they are not added to unity, as appears in the numerator of (5.3), then they do have real roots: those roots may be found by (5.7). Therefore, the *Analog Filter Design Theorem* is satisfied. For example, see Example 5.4.

Example 5.4

Suppose $N = 3$, $\omega_s = 100$, and $\epsilon = 0.01$. Then

$$C_3(100/\omega) = 4(100/\omega)^3 - 3(100/\omega) ,$$

and

$$|H(j\omega)|^2 = \frac{9\omega^4 - 2.4 \times 10^5 \omega^2 + 1.6 \times 10^9}{\omega^6 + 9\omega^4 - 2.4 \times 10^5 \omega^2 + 1.6 \times 10^9} .$$

A root of the denominator requires that

$$2.4 \times 10^5 \omega^2 = \omega^6 + 9\omega^4 + 1.6 \times 10^9 ,$$

which has no real solution, i.e., no real roots. The numerator has one real repeated root, which is 115.47. Therefore, the *Analog Filter Design Theorem* is satisfied: there is a corresponding $H(s)$ that meets all of the imposed constraints of **Section 2.6**. Therefore a circuit can be implemented with the desired third-order Chebyshev Type II response. □

5.5 LOCATION OF THE POLES AND ZEROS

Starting with (5.3) and following the procedure used in **Section 2.7**:

$$Y(s) = H(s)H(-s) = \frac{\epsilon^2 C_N^2(\omega_s/-js)}{1 + \epsilon^2 C_N^2(\omega_s/-js)} . \tag{5.18}$$

The zeros of (5.18) may be found by setting

$$C_N(\omega_s/-js_k) = 0 ,$$

and solving for the values of s_k . The trigonometric cosine form of (5.4) must be used since the inverse hyperbolic cosine of zero doesn't exist:

$$N \cos^{-1}(\omega_s/-js_k) = (\pi/2)(2k - 1), \quad k = 1, 2, 3, \dots . \tag{5.19}$$

Solving (5.19) for s_k results in

$$s_k = \frac{j\omega_s}{\cos [(2k - 1)\pi/(2N)]} ,$$

which, except for the j , is identical with (5.7). Therefore, the zeros of the transfer function of a Chebyshev Type II filter are as follows:

$$s_k = \pm j\omega_k^{(zero)} , \quad k = 1, 2, 3, \dots, N_p, \tag{5.20}$$

where $\omega_k^{(zero)}$ is given by (5.7).

The poles of (5.18) may be found by setting

$$C_N(\omega_s / -js) = \pm j/\epsilon .$$

Since $\epsilon < 1$, then $|\pm j/\epsilon| > 1$, and the hyperbolic form of C_N is perhaps the more appropriate:

$$C_N(\omega_s / -js_k) = \cosh[N \cosh^{-1}(\omega_s / -js_k)] = \pm j/\epsilon .$$

As noted in **Chapter 4**, since $\pm j/\epsilon$ is complex, either form, the cosine or the hyperbolic, is equally valid. Either approach will yield the same equations for finding the poles.

Since the approach here is identical to the approach used in **Section 4.5**, except that $-js_k/\omega_p$ is replaced by $\omega_s / -js_k$, it follows that the equivalent to (4.32) would be as follows:

$$\begin{aligned} s_k &= j\omega_s / \cosh[(1/N) \sinh^{-1}(1/\epsilon) + j(\pi/[2N])(2k - 1)] \\ &= \omega_s / \left\{ -\sinh[(1/N) \sinh^{-1}(1/\epsilon)] \sin[(\pi/[2N])(2k - 1)] \right. \\ &\quad \left. + j \cosh[(1/N) \sinh^{-1}(1/\epsilon)] \cos[(\pi/[2N])(2k - 1)] \right\} . \end{aligned} \quad (5.21)$$

For left-half plane poles:

$$\begin{aligned} s_k &= \sigma_k + j\omega_k , \\ \sigma_k &= -\omega_s \sinh[(1/N) \sinh^{-1}(1/\epsilon)] \sin[(\pi/[2N])(2k - 1)] / D(k) , \\ \omega_k &= -\omega_s \cosh[(1/N) \sinh^{-1}(1/\epsilon)] \cos[(\pi/[2N])(2k - 1)] / D(k) , \\ &\quad k = 1, 2, 3, \dots, N, \end{aligned} \quad (5.22)$$

where

$$\begin{aligned} D(k) &= \sinh^2[(1/N) \sinh^{-1}(1/\epsilon)] \sin^2[(\pi/[2N])(2k - 1)] \\ &\quad + \cosh^2[(1/N) \sinh^{-1}(1/\epsilon)] \cos^2[(\pi/[2N])(2k - 1)] . \end{aligned}$$

It is interesting to note that the poles of the Chebyshev Type I filter, for a normalized $\omega_p = 1$, may be expressed as follows:

$$s_k^{I(norm)} = \sqrt{D(k)} e^{j\theta_k} ,$$

where

$$\theta_k = -\tan^{-1} \left\{ \frac{\cosh[(1/N) \sinh^{-1}(1/\epsilon)] \cos[(\pi/[2N])(2k - 1)]}{\sinh[(1/N) \sinh^{-1}(1/\epsilon)] \sin[(\pi/[2N])(2k - 1)]} \right\} ,$$

and the poles of the Chebyshev Type II filter, for a normalized $\omega_s = 1$, may be expressed as follows:

$$s_k^{II(norm)} = \frac{1}{\sqrt{D(k)}} e^{-j\theta_k},$$

and therefore, the $s_k^{II(norm)}$ poles are the $s_k^{I(norm)}$ poles reflected about a unit circle in the s plane: the magnitudes are inversely related and the phase angles have opposite polarity. This inverse relationship between the normalized poles of the Chebyshev Type I transfer function and those of the Chebyshev Type II is often noted in the literature, and is implied, as noted above, by contrasting (4.33) and (5.22). It is possible to find the poles of a Chebyshev Type II transfer function by first finding the poles of a Chebyshev Type I transfer function and then converting them into Chebyshev Type II poles. However, this relationship between normalized poles doesn't seem to yield any practical advantage. Direct use of (5.22) is quite adequate for finding the poles of a Chebyshev Type II transfer function. Also note, by comparing Tables 4.1 and 5.1, that practical values for ϵ are very different for the two transfer functions.

Example 5.5

Suppose $N = 4$, $\omega_s = 1000 \text{ rad/s}$, and $\epsilon = 0.01$. From (5.9),

$$\omega_c = 496.71 .$$

From (5.22), the poles are

$$s_k = -171.16 \pm j 476.10, \quad -504.53 \pm j 240.79 .$$

From (5.20), the zeros are

$$s_k = \pm j 1082.39, \quad \pm j 2613.13 .$$

From the poles and zeros, and noting that the DC gain is unity:

$$H(s) = \frac{0.01 (s^2 + 1,171,572.88) (s^2 + 6,828,427.13)}{(s^2 + 342.32 s + 255,963.37) (s^2 + 1,009.06 s + 312,529.09)},$$

or,

$$H(s) = \frac{0.01 (s^4 + 8 \times 10^6 s^2 + 8 \times 10^{12})}{s^4 + 1,351.38 s^3 + 913,911.63 s^2 + 365,266,792.0 s + 8 \times 10^{10}} .$$

□

5.6 PHASE RESPONSE, PHASE DELAY, AND GROUP DELAY

A Chebyshev Type II filter, as seen above, is designed to meet given magnitude response specifications. Once the transfer function is determined, it may be put in the following form:

$$H(s) = \frac{K \sum_{k=0}^M b_k s^k}{\sum_{i=0}^N a_i s^i} \tag{5.23}$$

which is of the form of (2.39). Given (5.23), the phase response, from (2.79), may be stated as follows:

$$\angle H(j\omega) = \arctan \left(\frac{R_d(\omega) I_n(\omega) - R_n(\omega) I_d(\omega)}{R_n(\omega) R_d(\omega) + I_n(\omega) I_d(\omega)} \right) \tag{5.24}$$

where $R_d(\omega)$ and $I_d(\omega)$ denote the real and imaginary parts of the denominator, respectively, and $R_n(\omega)$ and $I_n(\omega)$ denote the real and imaginary parts of the numerator of (5.23) evaluated with $s = j\omega$.

The phase response of a Chebyshev Type II filter, with a normalized $\omega_c = 1$, a somewhat arbitrary, but common value of $A_s = 80 \text{ dB}$ ($\epsilon = 0.0001$), and several values of N , is shown in Figure 5.3. The phase response, from $\omega = 0$ until the first phase discontinuity, which occurs at $\omega = 1.55 \text{ rad/s}$ for the tenth-order response,

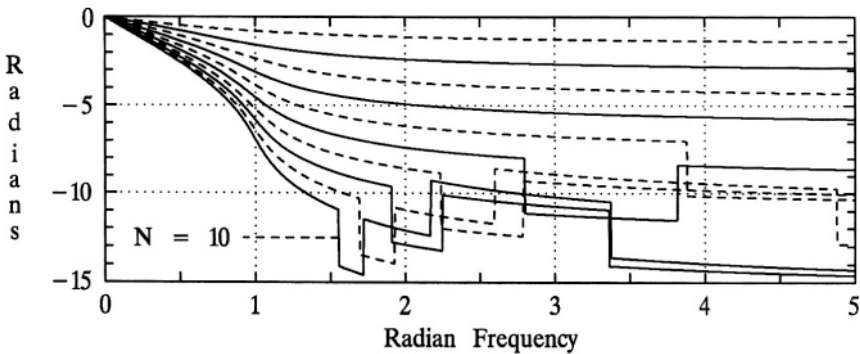


Figure 5.3 A plot of the phase response for a Chebyshev Type II filter with normalized $\omega_c = 1$, $A_s = 80 \text{ dB}$, and for values of N from 1 through 10.

is the total phase, in contrast to the principal phase.¹ The total phase, as shown for Butterworth filters in **Figure 3.6**, and for Chebyshev Type I filters in **Figure 4.4**, and and for Chebyshev Type II filters until the first phase discontinuity in **Figure 5.3**, is important because phase delay and group delay are directed related to the total phase response. Each of the phase discontinuities seen in **Figure 5.3** are π rad. The phase response in **Figure 5.3** for ω beyond, and including, the first phase discontinuity is not total phase, but rather *pseudo-principal* phase. That is, the phase shown is the total phase plus $m2\pi$ rad, where m is an integer. This technique allows for a less congested set of plots that is easier to read. In fact, each of the phase discontinuities, if total phase was to be preserved, are $+\pi$ rad. The phase discontinuities occur at transmission zeros, which are on the $j\omega$ axis; as ω increases through a zero, the phase response encounters a $+\pi$ rad discontinuity. It is interesting to note that while Butterworth and Chebyshev Type I filters each have a phase response in the limit, as ω approaches infinity, of $-N\pi/2$ rad, this is not true for Chebyshev Type II filters. Due to the finite transmission zeros, the phase response in the limit, as ω approaches infinity, for Chebyshev Type II filters, is zero, for N even, and is $-\pi/2$ rad, for N odd.

Taking the initial phase slope as a linear-phase reference, deviations from linear phase, for a normalized $\omega_c = 1$, $A_s = 80$ dB, and for several values of N , are shown in **Figure 5.4**. In the figure, solid lines are for even orders, and dashed lines are for odd orders. The phase deviation is shown in the figure from $\omega = 0$ until just before the first phase discontinuity occurs. Each phase discontinuity causes a $+\pi$ discontinuity in the phase deviation, but if plotted is somewhat misleading,

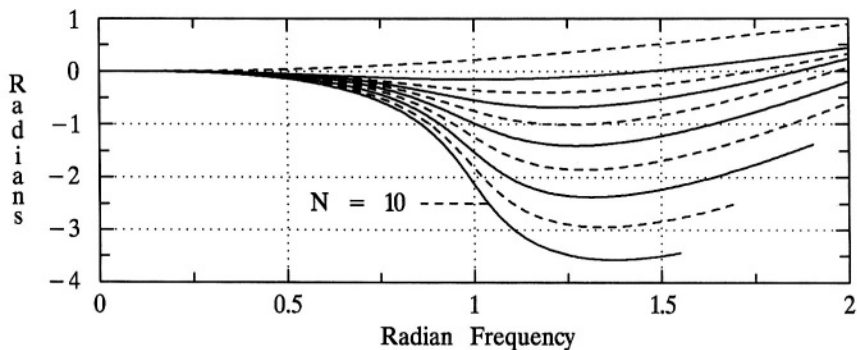


Figure 5.4 Phase deviation from linear for a Chebyshev Type II filter with normalized $\omega_c = 1$, $A_s = 80$ dB, and for values of N from 1 through 10.

¹The principal phase is restricted to the range of $\pm\pi$.

since the magnitude response is zero at the same frequency and is, in general, in the stopband. Over the frequency range of the figure, $0 \leq \omega \leq 2 \text{ rad/s}$, phase discontinuities only effect the plots for orders 8, 9, and 10, as can be seen in the figure.

The *phase delay*, $t_{pd}(\omega)$, for a filter is defined in (2.80), which is repeated here for convenience:

$$t_{pd}(\omega) = - \frac{\angle H(j\omega)}{\omega} \tag{5.25}$$

The *group delay* for a filter, $t_{gd}(\omega)$, is defined by (2.81) and is repeated here for convenience:

$$t_{gd}(\omega) = - \frac{d \angle H(j\omega)}{d\omega} \tag{5.26}$$

The phase delay of a Chebyshev Type II filter, with a normalized $\omega_c = 1$, $A_s = 80 \text{ dB}$, and for several values of N , is shown in **Figure 5.5**. Note that the phase delay discontinuities occur at the frequencies where there are phase discontinuities, that is, at the frequencies of transmission zeros. Since each phase discontinuity is $+\pi \text{ rad}$, each phase delay discontinuity is $-\pi/\omega \text{ s}$, where ω is the frequency of the discontinuity. The effect, therefore, of the transmission zeros, is that the phase delay for frequencies well beyond the passband approaches zero much more rapidly than it does for either Butterworth or Chebyshev Type I filters (see **Figures 3.8 and 4.6**).

The group delay is shown in **Figure 5.6**. Note that the group delay values at *DC* are very close to the phase delay values at *DC*, and that for all $\omega < 0.5$ the phase delay and the group delay are comparable. However, as ω approaches the 3

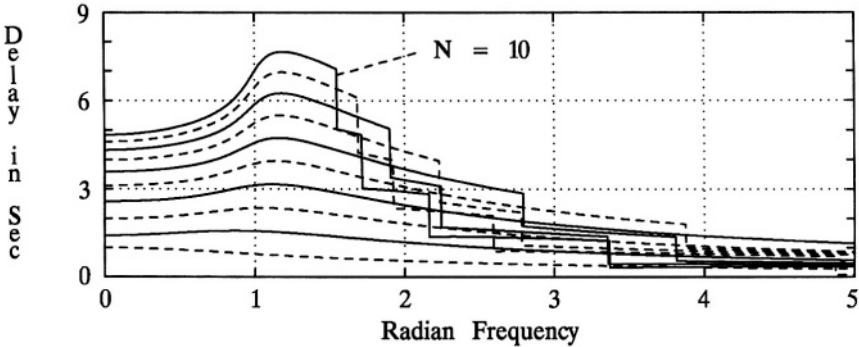


Figure 5.5 A plot of the phase delay for a Chebyshev Type II filter with normalized $\omega_c = 1$, $A_s = 80 \text{ dB}$, and for values of N from 1 through 10.

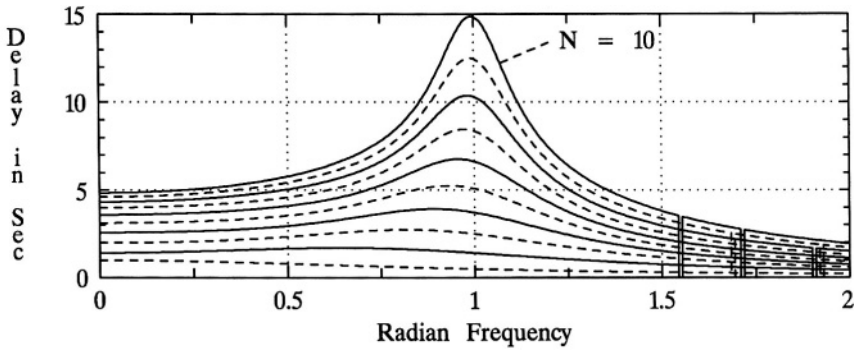


Figure 5.6 A plot of the group delay for a Chebyshev Type II filter with normalized $\omega_c = 1$, $A_s = 80 \text{ dB}$, and for values of N from 1 through 10.

dB corner frequency, in this case unity, the group delay becomes large due to the nonlinearity of the phase response near the corner frequency (see **Figures 5.3** and **5.4**). Note the points of discontinuity in the group delay, for example at $\omega = 1.55 \text{ rad/s}$ for $N = 10$. As can be seen from (5.26), at each $+\pi \text{ rad}$ phase discontinuity the group delay is theoretically $-\infty \text{ s}$. That is, the group delay is theoretically an infinite time *advance*, rather than a delay, at the point of a phase discontinuity; however, since this occurs only at a *point* along the frequency axis, and at a point of a transmission zero, the filter magnitude response at that point is zero; there is nothing to advance. In **Figure 5.6**, these *points* of infinite time advance are plotted with non-zero width; this is a result of the plotting software and from the fact that the calculation frequency-sample width is non-zero. Also, for convenience, the minimum delay value of the figure is zero.

5.7 TIME-DOMAIN RESPONSE

The unit impulse response of a Chebyshev Type II filter, with a normalized $\omega_c = 1$, $A_s = 80 \text{ dB}$, and for several values of N , is shown in **Figure 5.7**. Note that for even orders there is an impulse at the origin (not shown in the figure), but the weight of these impulses is ϵ (in this case, since $A_s = 80 \text{ dB}$, $\epsilon = 0.0001$), and therefore these impulses are insignificant. The unit step response of a Chebyshev Type II filter, with normalized $\omega_c = 1$, $A_s = 80 \text{ dB}$, and for several values of N , is shown in **Figure 5.8**.

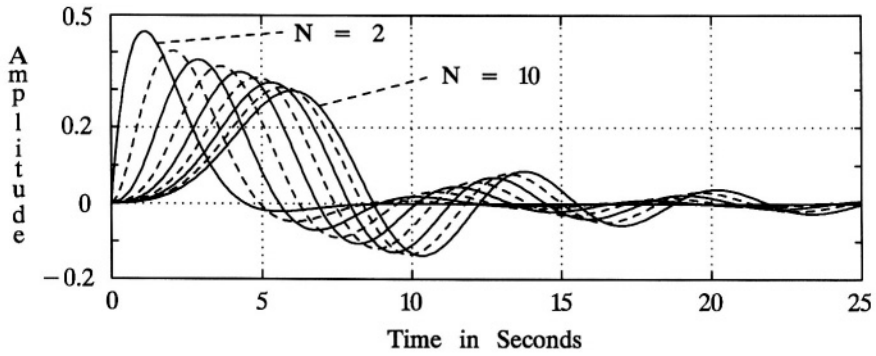


Figure 5.7 A plot of the unit impulse response for a Chebyshev Type II filter with normalized $\omega_c = 1$, $A_s = 80 \text{ dB}$, and for values of N from 2 through 10.

5.8 COMPARISON WITH BUTTERWORTH AND CHEBYSHEV TYPE I FILTERS

There are several ways in which Chebyshev Type II filters may be compared with Butterworth and Chebyshev Type I filters. The magnitude frequency responses may be compared, the phase responses, the phase delays, the group delays, the unit impulse responses, and the unit step responses, for example.

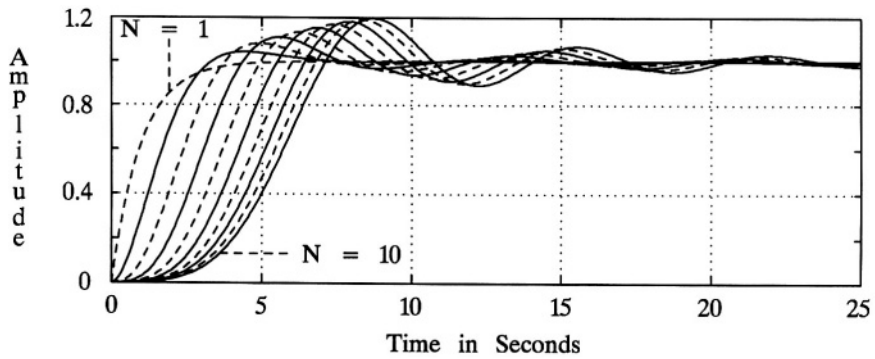


Figure 5.8 A plot of the unit step response for a Chebyshev Type II filter with normalized $\omega_c = 1$, $A_s = 80 \text{ dB}$, and for values of N from 1 through 10.

It is noted that, for a given set of filter specifications, the minimum order required for a Chebyshev Type II response will never be greater than that required for a Butterworth response: it will frequently be less. It is not always less, since the orders are restricted to integers. As noted in **Section 5.3**, the minimum order required for a Chebyshev Type II filter is identical with that for a Chebyshev Type I filter.

Although the passband of a Chebyshev Type II response was not designed to be maximally flat in any sense, yet, as can be seen by comparing **Figures 5.2** and **3.2**, the passband magnitude response of a Chebyshev Type II filter is comparable with that of a Butterworth. In fact, over a wide range of filter specifications, the passband magnitude response of a Chebyshev Type II filter, with the same order and same 3 dB corner frequency, is more flat than that of a Butterworth filter. One way of demonstrating this is by comparing *Filter Selectivity* for the two filters. Let (5.11) be denoted $F_S^{(CII)}$ and (3.7) be denoted $F_S^{(B)}$. It can be shown that

$$F_S^{(CII)} \geq F_S^{(B)} \tag{5.27}$$

for all ϵ , N , and ω_c . Equality in (5.27) is achieved only for $N = 1$ or for very small values of ϵ . For example, for $\epsilon = 0.0001$, $N = 10$, and $\omega_c = 1$, then

$$F_S^{(CII)} = 4.667, \quad F_S^{(B)} = 3.536 .$$

It can be shown that, for all practical values of ϵ , that the *Shaping Factor* of a Chebyshev Type II filter is always less than that for a Butterworth filter of the same order. To illustrate this, compare the results of **Example 5.2** to that of **Example 3.1**; in both cases $a = 3$ dB and $b = 80$ dB. For example, for $N = 8$, S_3^{80} for the Butterworth filter is 3.16, whereas it is only 1.87 for the Chebyshev Type II filter, which indicates that the attenuation in the transition band increases more rapidly as a function of ω for the Chebyshev Type II filter than it does for the Butterworth. This, of course, can readily be observed by comparing **Figures 3.1** and **5.1**.

The above may imply that a Chebyshev Type II filter is superior to a Butterworth filter of the same order. However, the phase response of a Butterworth filter is more nearly linear than is that of a Chebyshev Type II filter of the same order. The differences, however, are not great. The phase deviation from linear for a Chebyshev Type II filter, as shown in **Figure 5.4**, is greater than it is for a Butterworth filter, as shown in **Figure 3.7**. This is, of course, reflected in the phase delay and the group delay, but comparing **Figure 5.5** with **3.8**, and **5.6** with **3.9**, shows that the differences are not great. Comparing plots of the unit impulse response and the unit step response of the two filters also shows that the differences are not great.

The comparison between Chebyshev Type II filters and Chebyshev Type I filters is more objective and straight-forward, especially since the minimum required

order to meet design specifications is identical for both filters. A Chebyshev Type II filter, of the same order, has a more constant magnitude response in the passband, a more nearly linear phase response, a more nearly constant phase delay and group delay, and less ringing in the impulse and step responses, than does a Chebyshev Type I filter: compare **Figures 5.2 and 4.2, 5.4 and 4.5, 5.5 and 4.6, 5.6 and 4.7, 5.7 and 4.8, and 5.8 and 4.9**. However, while (5.11) and (4.9), equations for *Filter Selectivity*, are the same, the numerical values for ϵ differ significantly for the two filters. For all practical filters, ϵ for the Chebyshev Type II filter will be significantly smaller than that for the comparable Chebyshev Type I filter. To illustrate this, compare **Table 5.1 to Table 4.1**. The result is that, for all practical filters, F_s for the Chebyshev Type I filter will be significantly larger than for the comparable Chebyshev Type II filter: compare **Figures 5.2 and 4.2**. However, the comparison of *Shaping Factor* values is not consistent. It is possible for a Chebyshev Type II filter to have a smaller S_a^b than a corresponding Chebyshev Type I filter. To illustrate this, compare 10th-order responses in **Figures 5.1 and 4.1** with $a = 3$ dB and $b = 80$ dB.

5.9 CHAPTER 5 PROBLEMS

- 5.1** Given the defining equations for a Chebyshev Type II response, (5.3) and (5.4), and given that $\omega_c = 1000$ rad/s, $A_s = 50$ dB, and $N = 3$:
- Determine the value of ϵ .
 - Determine the value of ω_s .
 - Determine the frequencies where the response is zero.
 - Determine the frequencies in the stopband where the attenuation is A_s .
 - Accurately sketch the magnitude frequency response. Use only a calculator for the necessary calculations. Use a vertical scale in dB (0 to -60dB), and a linear radian frequency scale from 0 to 5000 rad/s.
 - Accurately sketch the magnitude frequency response. Use only a calculator for the necessary calculations. Use a linear vertical scale from 0 to 1, and a linear radian frequency scale from 0 to 2000 rad/s.
- 5.2** Given the defining equations for a Chebyshev Type II response, (5.3) and (5.4), and given that $\omega_c = 1000$ rad/s, $A_s = 60$ dB, and $N = 6$:
- Determine the value of ϵ .
 - Determine the value of ω_s .
 - Determine the frequencies where the response is zero.
 - Determine the frequencies in the stopband where the attenuation is A_s .
 - Accurately sketch the magnitude frequency response. Use only a calculator for the necessary calculations. Use a vertical scale in dB (0

- (f) to -80dB), and a linear radian frequency scale from 0 to 5000 rad/s .
 (f) Accurately sketch the magnitude frequency response. Use only a calculator for the necessary calculations. Use a linear vertical scale from 0 to 1, and a linear radian frequency scale from 0 to 1200 rad/s .

5.3 Starting with (2.37) and the square root of (5.3), derive (5.11).

5.4 On page 159 it is mentioned that for a large range of ϵ the passband magnitude response for a Chebyshev Type II filter is more flat than that of a Butterworth filter of the same order. On page 171 this concept is expanded upon. Since both responses are relatively flat in the passband, if they both have the same ω_c , then the one with the larger *Filter Selectivity* implies that the magnitude frequency response remains closer to unity as ω approaches ω_c then does the other one. Verify that, for the same order and the same ω_c , *Filter Selectivity* for a Chebyshev Type II filter is greater than or equal to *Filter Selectivity* for a Butterworth filter, and that they are equal only for $N = 1$ or for very small values of ϵ . That is, verify (5.27).

5.5 Determine the value of *Filter Selectivity* for the Chebyshev Type II filter specified in **Problem 5.2**. Compare this value with the *Filter Selectivity* for a Butterworth filter with similar specifications: $\omega_c = 1000\text{ rad/s}$, and $N = 6$. Compare this value with the *Filter Selectivity* for a Chebyshev Type I filter with similar specifications: $\omega_c = 1000\text{ rad/s}$, $A_p = 1.5\text{ dB}$, and $N = 6$.

5.6 Determine the value of the *Shaping Factor* for the Chebyshev Type II filter specified in **Problem 5.2**, for $a = 3\text{ dB}$ and $b = 60\text{ dB}$. Compare this value with the *Shaping Factor* for a Butterworth filter with similar specifications: $\omega_c = 1000\text{ rad/s}$, and $N = 6$. Compare this value with the *Shaping Factor* for a Chebyshev Type I filter with similar specifications: $\omega_c = 1000\text{ rad/s}$, $A_p = 1.5\text{ dB}$, and $N = 6$.

5.7 Estimate the *Shaping Factor* for a Chebyshev Type II filter with $N = 5$, $\omega_c = 1$, and $A_s = 80\text{ dB}$, from **Figure 5.1**. Compare your results with that obtained from (5.14).

5.8 Suppose filter specifications are stated as follows: $f_c = 3500\text{ Hz}$, $f_s = 7000\text{ Hz}$. Determine the required filter order to meet the given specifications:

- For a Chebyshev Type II filter with $A_s = 40\text{ dB}$.
- For a Chebyshev Type II filter with $A_s = 60\text{ dB}$.
- For a Chebyshev Type II filter with $A_s = 80\text{ dB}$.

- (d) For a Chebyshev Type II filter with $A_s = 100 \text{ dB}$.
 - (e) For comparison purposes, repeat parts (a) through (d) for a Butterworth filter.
 - (f) For comparison purposes, repeat parts (a) through (d) for a Chebyshev Type I filter with 0.1 dB of ripple.
 - (g) For comparison purposes, repeat parts (a) through (d) for a Chebyshev Type I filter with 0.5 dB of ripple.
 - (h) For comparison purposes, repeat parts (a) through (d) for a Chebyshev Type I filter with 1.5 dB of ripple.
- 5.9** Given that $N = 4$, $\omega_s = 100$, and $\epsilon = 0.01$, express $|H(j\omega)|^2$ in polynomial form similar to *Example 5.4*, and demonstrate that it satisfies the *Analog Filter Design Theorem*.
- 5.10** Prove that the magnitude-squared frequency response of **Problem 5.1** satisfies the *Analog Filter Design Theorem*.
- 5.11** Prove that the magnitude-squared frequency response of **Problem 5.2** satisfies the *Analog Filter Design Theorem*.
- 5.12** Determine the 9-th and 10-th order inverse Chebyshev polynomials.
- 5.13** Sketch the square of the 4-th order inverse Chebyshev polynomial for ω from 0 to 1.1 rad/s . Compute the square of (5.4) over this same radian frequency range for $\omega_s = 1$, and verify that it is numerically the same as the inverse Chebyshev polynomial.
- 5.14** Determine the poles and zeros of the filter specified in **Problem 5.8** (a).
- 5.15** Determine the poles and zeros of the filter specified in **Problem 5.8** (d).
- 5.16** Determine the poles and zeros of the filter specified in **Problem 5.9**.
- 5.17** Suppose $N = 4$, $\omega_s = 1000 \text{ rad/s}$, and $\epsilon = 0.01$. Determine the transfer function $H(s)$. That is, verify the results of *Example 5.5*.
- 5.18** Determine the transfer function $H(s)$ for the Chebyshev Type II filter specified in **Problem 5.1**. Express the denominator of $H(s)$ in two ways: (1) As a polynomial in s . (2) As the product of a second-order polynomial in s , the roots of which being complex conjugates, and a first-order term. State the numerical values of the three poles and the finite-value zeros. Sketch the six poles and the zeros of $H(-s)H(s)$ on an s plane.

- 5.19** Determine the transfer function $H(s)$ for the Chebyshev Type II filter specified in **Problem 5.2**. Express the denominator of $H(s)$ in two ways: (1) As a polynomial in s . (2) As the product of three second-order polynomials in s , the roots of each second-order polynomial being complex conjugates. Express the numerator of $H(s)$ in two ways: (1) As a polynomial in s . (2) As the product of three second-order polynomials in s , the roots of each second-order polynomial being complex conjugates. State the numerical values of the six poles and six zeros. Sketch the twelve poles and zeros of $H(-s)H(s)$ on an s plane.
- 5.20** Suppose a Chebyshev Type II filter is used that has a minimum attenuation of 60 dB at 4000 Hz ($A_s = 60$ dB).
- If $f_p = 3000$ Hz and $A_p = 1$ dB, what is the minimum order required?
 - If $A_p = 1$ dB and $N = 10$, what is the maximum value of f_p ?
 - If $f_p = 3000$ Hz and $N = 10$, what is the minimum value of A_p ?
- 5.21** Under the conditions of part (c) of **Problem 5.20**, determine the transfer function $H(s)$, and give numerical values for all the poles and zeros.
- 5.22** Sketch the step response of a 10-th order Chebyshev Type II filter with $f_c = 1000$ Hz and $A_s = 80$ dB. Refer to **Figure 5.8** and make use of the scaling property of Fourier transforms. What would the maximum group delay be for this filter, and at what frequency would it occur? At what time would the peak of the unit impulse response of this filter be, and what would be the value of that peak?
- 5.23** Using the MATLAB functions *cheb2ap*, *impz* and *step*:
- Determine the transfer function in polynomial form, and also factored to indicate the poles and zeros, of a Chebyshev Type II filter with $\omega_p = 1$, $A_p = 1.2$ dB, and $N = 6$.
 - Determine the impulse response and the step response for the filter of part (a).
 - By multiplying the pole vector and the zero vector found in part (a) by $2\pi 1000$ determine the transfer function of a Chebyshev Type II filter with $f_p = 1000$ Hz, $A_p = 1.2$ dB, and $N = 6$.
 - Determine and plot the magnitude frequency response of the filter of part (c) by using the MATLAB function *freqz*. Use a vertical scale in dB and a linear horizontal scale from 0 to 5000 Hz. Also determine and plot the phase response over this same frequency range. Use the MATLAB function *unwrap* rather than plotting the principle phase.
 - By appropriately scaling the impulse response and the step response of

part (b), determine and plot the impulse response and the step response of the filter of part (c). That is, the time axis for the step response needs to be scaled by $1/(2\pi 1000)$, and the unit impulse response needs the same time-axis scaling and requires an amplitude scaling of $2\pi 1000$.

- (f) Determine and plot the phase delay of the filter of part (c). Note that this is easily obtained from the phase response of part (d).
- (g) Determine and plot the group delay of the filter of part (c). Note that this also is easily obtained from the phase response of part (d): $t_{gd}(n) \cong -[\phi(n) - \phi(n-1)]/S_s$, where $\phi(n)$ is the phase in radians at step n , and S_s is the step size in rad/s.

5.24 To demonstrate the critical nature of a filter design, this problem experiments some with the filter of **Problem 5.23** (c). Multiply the highest-frequency pole-pair of the filter of **Problem 5.23** (c) by 1.1, leaving all other poles and zeros unchanged. Determine and plot the magnitude frequency response of the filter by using the MATLAB function *freqs*. Use a vertical scale in dB and a linear horizontal scale from 0 to 5000 Hz. Also determine and plot the phase response over this same frequency range. Use the MATLAB function *unwrap* rather than plotting the principle phase. Compare these results with that obtained for **Problem 5.23** (d).

5.25 This problem continues to demonstrate the critical nature of a filter design, and is a continuation of **Problem 5.24**. Multiply the lowest-frequency zero-pair of the filter of **Problem 5.23** (c) by 1.1, leaving all other poles and zeros unchanged. Determine and plot the magnitude frequency response of the filter by using the MATLAB function *freqs*. Use a vertical scale in dB and a linear horizontal scale from 0 to 5000 Hz. Also determine and plot the phase response over this same frequency range. Use the MATLAB function *unwrap* rather than plotting the principle phase. Compare these results with that obtained for **Problem 5.23** (d).

CHAPTER 6

ELLIPTIC FILTERS

Elliptic filters, also known as Cauer filters in recognition of the contributions of Wilhelm Cauer to elliptic, as well as other filters, are noted for having an equiripple passband magnitude response similar to Chebyshev Type I filters, and an equiripple stopband magnitude response similar to Chebyshev Type II filters. They require the lowest order of all the classical filters to meet given design specifications. As was noted at the opening to **Chapter 4**, although a logical presentation of classical analog filters frequently follows the order of Butterworth, Chebyshev Type I, Chebyshev Type II, and elliptic, such as is done in this book, this is not the chronological order, as was shown in **Section 1.4**.

In this chapter the elliptic filter magnitude response is defined, and it will be observed that it satisfies the *Analog Filter Design Theorem*. Explicit formulas for the design and analysis of elliptic filters, such as *Filter Selectivity*, *Shaping Factor*, the minimum required order to meet design specifications, etc., will be obtained. From the defining $|H(j\omega)|^2$ the corresponding $H(s)$ will be determined, and means for determining the filter poles and zeros are found. To complete the study of lowpass, prototype elliptic filters, the phase response, phase delay, group delay, and time-domain response characteristics are investigated.

6.1 INTRODUCTION

The magnitude-squared frequency response of an elliptic filter is defined below.

Definition of the magnitude-squared elliptic response:

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon_p^2 R_N^2(\omega, \omega_p, \omega_s, \epsilon_p, \epsilon_s)} \quad (6.1)$$

where

$$R_N(\omega, \omega_p, \omega_s, \epsilon_p, \epsilon_s) = sn[\kappa sn^{-1}(\omega/\omega_p, \omega_p/\omega_s) + q X_1, \epsilon_p \epsilon_s] \quad (6.2)$$

$sn(x, \tau)$ is the Jacobian elliptic sine of x with modulus τ , ω_p denotes the frequency of the passband edge, ω_s denotes the frequency of the stopband edge, κ denotes a parameter that is a function of the filter order N and complete elliptic integral values to be described below, ϵ_p is a positive real constant that specifies the magnitude of the ripple in the passband, ϵ_s is a positive real constant that specifies the magnitude of the ripple in the stopband, q is zero if N is odd and unity if N is even, and X_1 is the complete elliptic integral of $\tau_1 = \epsilon_p \epsilon_s$ to be described below.

The alert reader may observe that (6.2) is in general impossible, since there are no degrees of freedom in the equation. That is, (6.2) is dependent on all four design parameters¹ and on the order N as well. Contrasting (6.2) with previously-presented filters, note that the corresponding function for a Butterworth filter is $(\omega/\omega_c)^N$, which is parameter-dependent only on the passband edge frequency (ω_c) and the order; the corresponding function for a Chebyshev Type I filter is $C_N(\omega/\omega_p)$; the corresponding function for a Chebyshev Type II filter is $C_N(\omega_s/\omega)$. For a Butterworth filter, for a given ω_c and N , ω_s and A_s have a one-to-one correspondance: there is one degree of freedom in the filter design. Similarly for Chebyshev Type I and Chebyshev Type II filters, there is one degree of freedom. This degree of freedom is not present in (6.2). To stress this point: If all design parameters were arbitrary, the logical thing to do would be to let the order be unity for any given set of the other four parameters. Hopefully this is recognized as being absurd. Although (6.2) may be evaluated for any given set of parameters, it will not in general yield desirable nor expected results: in general, the results will neither be real nor meet the stated specifications. Certain requirements must be imposed. It is necessary that (6.2) meet the following restrictions for the purpose of elliptic filter design:

- (1) R_N squared must be real,
- (2) R_N squared must have equiripple in the passband,
- (3) R_N squared must have equiripple in the stopband,
- (4) R_N squared must be such that (6.1) will satisfy the *Analog Filter Design Theorem*. Clearly (1) is redundant and included in (4).

The reason why (6.2) does not in general meet the above restrictions is that elliptic sine functions are doubly periodic, having a real period and an imaginary period, and those periods are dependent upon both the argument x and the modulus τ ,

¹See Figure 2.15 on page 52, and note that ϵ_p is directly related to A_p , and ϵ_s is related to A_s .

or, in terms of (6.2), the periods are dependent upon the argument $sn^{-1}(\omega/\omega_p, \omega_p/\omega_s) + qX_1$, and the modulus $\epsilon_p \epsilon_s$. Therefore, in order to understand these dependencies it is necessary to consider an overview of those elements of elliptic function theory relevant to elliptic filter design. Elliptic functions were intensely studied in the 19th century. A relatively recent and commendable book on the subject is by Lawden (1989)². The treatment to follow, beginning in Section 6.2, is very brief, focusing on the immediate needs of elliptic filter theory. But before considering elliptic function theory, two examples that illustrate and elaborate on the above discussion are given.

Example 6.1

This example illustrates three things: (1) that a Chebyshev filter is a special case of an elliptic filter, (2) that in this case, the elliptic sine form and the elliptic cosine form of (6.2) are the same, and (3) that in this case, $\kappa = N$.

Let $\omega_p = 1$, $\omega_s = \infty$, $\epsilon_p = 1$, $\epsilon_s = 0$, and $N = 3$. It can be shown that, with these specifications,

$$R_3^2 = sn^2[3 sn^{-1}(\omega, 0), 0] = cn^2[3 cn^{-1}(\omega, 0), 0] = C_3^2(\omega) , \quad (6.3)$$

where $cn(x, \tau)$ is the Jacobian elliptic cosine function of argument x and modulus τ , and C_N is given by (4.2)³. In Figure 6.1 are shown (6.1) with R_3^2 experimentally obtained from the elliptic sine form, the elliptic cosine form, and from the standard Chebyshev Type I response expressed by (4.1). The three plots in Figure 6.1 are indistinguishable. The data plotted in Figure 6.1 were obtained from the MATLAB⁴ m-file *EXAMP6_1.m*, as found on the accompanying disk and printed in Appendix C. □

Example 6.2

This example illustrates that when properly set up, (6.1) yields a real magnitude-squared frequency response that meets the design specifications, and that

²The last paragraph of the preface to Lawden’s book is rather interesting: “I am now approaching the termination of a life, one of whose major enjoyments has been the study of mathematics. The three jewels whose effulgence has most dazzled me are Maxwell’s theory of electromagnetism, Einstein’s theory of relativity, and the theory of elliptic functions. I have now published textbooks on each of these topics, but the one from whose preparation and writing I have derived the greatest pleasure is the present work on elliptic functions. How enviable are Jacobi and Weierstrass to have been the creators of such a work of art! As a lesser mortal lays down his pen, he salutes them and hopes that his execution of their composition does not offend any who have ears to hear the music of the spheres.”

³In (4.2), C_N is expressed as a trigonometric cosine for frequencies up to ω_p , and as a hyperbolic cosine for frequencies beyond that. This is correct, convenient, and as usually expressed. However, it is noted that if would be equally accurate to express C_N in the trigonometric cosine form for all frequencies, and perhaps make the equivalence with the elliptic cosine form of (6.3) of greater interest.

⁴MATLAB is a registered trademark of The Math Works, Inc.

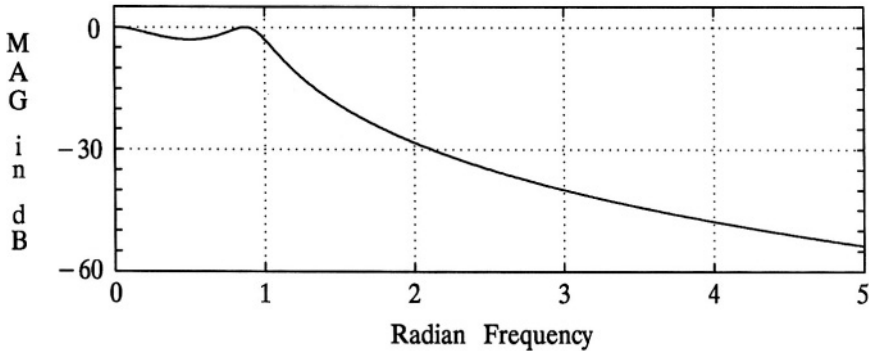


Figure 6.1 A demonstration that a Chebyshev filter is a special case of an elliptic filter. Overlay of the magnitude response of (6.1) based on the elliptic sine form shown in (6.2), based on the elliptic cosine form (see (6.3)), and based on $C_3^2(\omega)$: plots of *Example 6.1*.

when design parameters are arbitrary chosen the result, in general, will not be real, and even the magnitude of the result does not meet the design specifications. This example, plus the previous one, serve to illustrate the previous discussion in this section, and also to motivate the study of elliptic integrals and Jacobian elliptic functions to follow.

Let $\omega_p = 1$, $\omega_s = 1.9789$, $\epsilon_p = 1$ ($A_p = 3$ dB), $\epsilon_s = 0.0100005$ (as will be seen, $A_s = 40$ dB), and $\kappa = 2.7907$. As will be seen in following sections, these critical values will yield a real result, that can be realized with a third-order transfer function (three poles, and two zeros on the $j\omega$ axis). These parameters are denoted as *Set 1* in this example, and are the *properly set-up parameters*.

For comparison purposes, let $\omega_p = 1$, $\omega_s = 2.5$, $\epsilon_p = 1$, $\epsilon_s = 0.01$, and $\kappa = 3$. These parameters yield complex results, and cannot be realized with a rational transfer function with real coefficients. These parameters are denoted as *Set 2* in this example, and are the *first set of arbitrary parameters*.

Also for comparison purposes, let $\omega_p = 1$, $\omega_s = 1.5$, $\epsilon_p = 1$, $\epsilon_s = 0.008$, and $\kappa = 3$. These parameters also yield complex results, and cannot be realized with a rational transfer function with real coefficients. These parameters are denoted as *Set 3* in this example, and are the *second set of arbitrary parameters*.

Also for comparison purposes, the MATLAB function *ellipap* is used to design an elliptic filter with specified parameters of $N = 3$, $A_p = 10 \log(2)$, and

$A_s = 40$.⁵ The procedure for designing elliptic filters, which is accomplished by the MATLAB function *ellipap*, is developed below in this chapter. It is included in this example to illustrate the identical magnitude-squared frequency response results using parameter *Set 1* in (6.1).

The four frequency responses are shown in **Figure 6.2**. Since *Set 2* and *Set 3* each yield complex results for the “magnitude-squared” response of (6.1), what is plotted in **Figure 6.2** for those two cases are the magnitude of the result. Note that the response for *Set 1* is identical to the response for the elliptic filter designed by the MATLAB function *ellipap*. Not only do *Set 2* and *Set 3* yield complex results, but, as can be seen in **Figure 6.2**, they also do not have equiripple in the stopband.

In **Figure 6.3** are shown the real and imaginary parts of (6.2) for parameter *Sets 1, 2* and *3*. As the magnitude of the real part of (6.2) for *Set 1*, near $\omega = 2.245$, the transmission zero, is considerably larger than that of either *Set 2* or *Set 3*, in the figure the real and imaginary parts of (6.2) for *Set 1* are divided by 40. Ideally, (6.2) would be infinity at the transmission zero, and the imaginary part would be zero for all ω . Computationally, for *Set 1*, the imaginary part of (6.2) is insignificant compared with the real part for all ω . However, as can be seen in **Figure 6.3**, the real and imaginary parts for both *Sets 2* and *3* are comparable, and

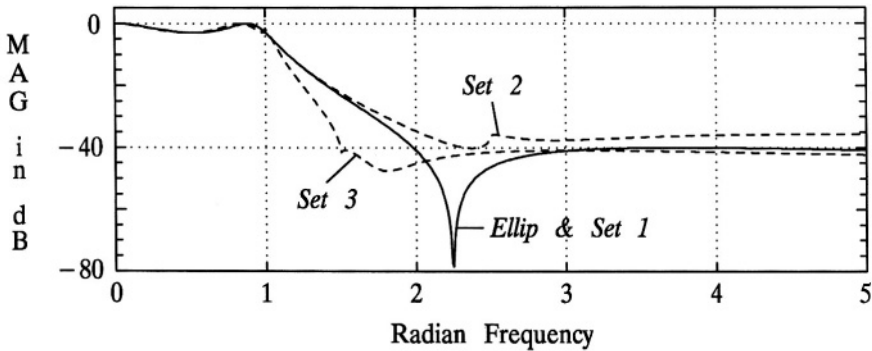


Figure 6.2 A demonstration that setting up the parameters of (6.2) for proper elliptic filter results is critical, and that arbitrarily chosen values yield, in general, complex results, and magnitude frequency responses that are not equiripple. Note that *Set 1* results are indistinguishable from the third-order elliptic filter design results. These are magnitude plots for *Example 6.2*.

⁵The MATLAB function *ellipap* requires three inputs: order, passband ripple in dB, and minimum stopband attenuation in dB.

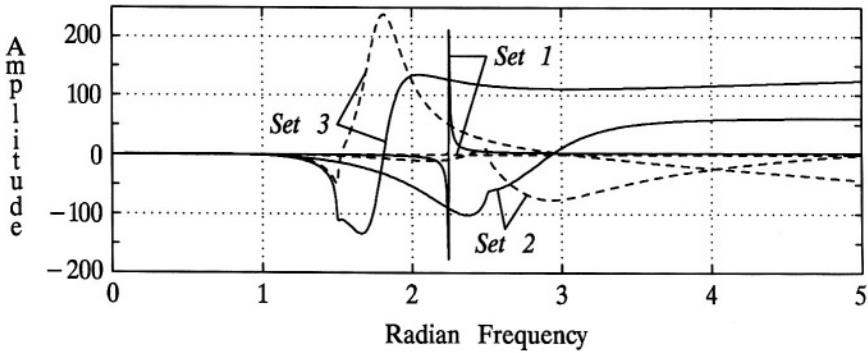


Figure 6.3 A demonstration that setting up the parameters of (6.2) such as to yield a real result is critical, and that arbitrarily chosen values yield, in general, complex results. Real parts are in solid lines, and imaginary parts are dashed. *Set 1* results, for compatible scaling, are divided by 40. These are real and imaginary plots of sn for *Example 6.2*.

therefore the imaginary parts cannot be ignored. The data plotted in **Figures 6.2** and **6.3** were obtained from the MATLAB m-file *EXAMP6_2.m*, as found on the accompanying disk and printed in **Appendix D**. □

6.2 ELLIPTIC INTEGRALS AND JACOBIAN ELLIPTIC FUNCTIONS

In order to formalize, and simplify, the design procedure for elliptic filters, with at least a basic understanding of the underlying theory, some background information on elliptic integrals and functions is necessary. Details on elliptic integrals and functions may be found elsewhere (Byrd and Friedman, 1954; Calahan, 1964; Lawden, 1989).

Basic Definitions

The elliptic integral of the first kind is defined as follows:

$$x(\phi, \tau) = \int_0^\phi \frac{d\xi}{\sqrt{1 - \tau^2 \sin^2(\xi)}} \quad , \quad (6.4)$$

where τ , a fixed parameter, generally real and $0 \leq \tau \leq 1$, is denoted as the *modulus*.⁶ If $\phi = \pi/2$, the integral is denoted as the *complete* elliptic integral of the first kind, and denoted as X .⁷

$$X = x(\phi, \tau) \Big|_{\phi = \pi/2} = x(\pi/2, \tau) \quad ,$$

for reasons that will be seen below, and if $\phi \neq \pi/2$, the integral is denoted as an *incomplete* elliptic integral of the first kind. As will be seen below, ϕ need not be real, and therefore $x(\phi, \tau)$ is in general complex, in which case the integral is a line integral in the complex plane.

The inverse of the elliptic integral of the first kind is denoted $\phi(x, \tau)$ and is the value of ϕ given x and τ . The Jacobian elliptic sine of x with modulus τ is denoted as follows:

$$sn(x, \tau) = \sin[\phi(x, \tau)] \quad , \quad (6.5)$$

and the Jacobian elliptic cosine of x with modulus τ is denoted:

$$cn(x, \tau) = \cos[\phi(x, \tau)] \quad .$$

It is also useful to note the inverse Jacobian elliptic sine and cosine, respectively:

$$sn^{-1}[\sin(\phi), \tau] = \int_0^\phi \frac{d\xi}{\sqrt{1 - \tau^2 \sin^2(\xi)}} \quad , \quad (6.6)$$

and

$$cn^{-1}[\cos(\phi), \tau] = \int_0^\phi \frac{d\xi}{\sqrt{1 - \tau^2 \sin^2(\xi)}} \quad .$$

⁶Often in the literature, the modulus is denoted by k . In signal processing, k usually denotes an integer: τ is used here to denote the modulus. Also, in the literature, since the “modulus,” as defined here, appears squared in the integrand, it is not uncommon to see the “modulus” defined as m , also usually denoting an integer in signal processing, and is equal to the square of the “modulus” as defined here. Therefore, caution should be used to insure that the definition is used consistently in practice.

⁷In the literature the complete elliptic integral is often denoted as K , but X is used here to be more suggestive of its relationship to x .

Information on the elliptic cosine is included here because it was mentioned in **Section 6.1**. It is not, however, necessary for the presentation in this section relevant to elliptic filter theory. There are other *kinds* of elliptic integrals, and many other Jacobian elliptic functions, but these will suffice for an understanding of the design and analysis of elliptic filters.

Double Periodicity of Jacobian Elliptic Functions

An important property of Jacobian elliptic functions is that they are periodic with respect to the argument along the real axis and along the imaginary axis (Lawden, 1989, sec. 2.2), and hence the term *double periodicity*. It can be shown that the real period for both sn and cn , as well as other Jacobian elliptic functions, is $4X$:

$$\text{Real Period} = 4X ,$$

where X , as noted above, is the complete elliptic integral of the modulus τ . That is,

$$sn(x + 4mX, \tau) = sn(x, \tau), \quad \forall x ,$$

where m is an integer. It can also be shown that the imaginary period is $j2X'$:

$$\text{Imaginary Period} = j2X' ,$$

where X' is the complete elliptic integral of the *complementary modulus* τ' :

$$\tau' = \sqrt{1 - \tau^2} ,$$

$$X' = \int_0^{\pi/2} \frac{d\xi}{\sqrt{1 - (\tau')^2 \sin^2(\xi)}} .$$

That is,

$$sn(x + j2mX', \tau) = sn(x, \tau), \quad \forall x ,$$

where m is an integer.

To illustrate the periodicity of the elliptic sine function with respect to real x , plots of the function for several values of τ , with respect to real x , are shown in **Figure 6.4**. Since the *period* is a function of τ , and it is the *shape* of the function that is stressed, the horizontal axis is normalized to X , the complete elliptic integral, or the *quarter-period*, as it is also known. As can be seen, two complete periods, or $8X$, are shown for each value of τ . In each plot, the function plotted is $sn(x, \tau)$. As can be readily seen from (6.4) and (6.5),

$$sn(x, 0) = \sin(x) ;$$

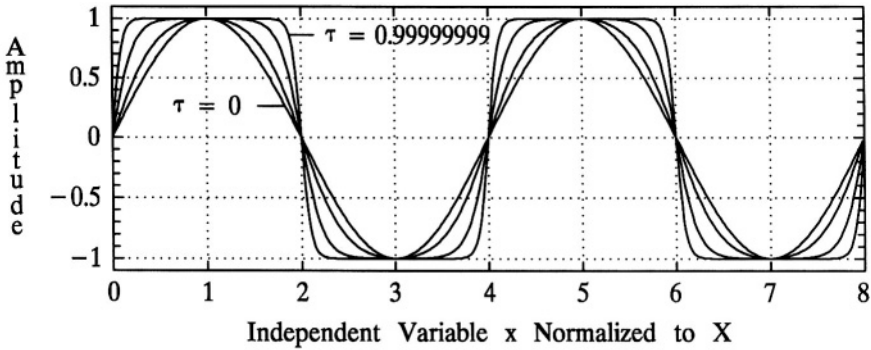


Figure 6.4 Plots of the elliptic sine function for several values of the modulus τ , versus real x : $sn(x, \tau)$.

this can be observed in the figure. The effect, then, that τ has on the elliptic sine function, is that as τ increases, the period increases and the shape of the function becomes more “square.” For the four modulus values in the figure, 0, 0.9, 0.999, and 0.99999999, the corresponding values of X are 1.5708, 2.2805, 4.4956, and 10.2501, respectively. The corresponding values of X' , the complete elliptic integral of the complementary modulus, are 3.6956, 3.0161, 2.3593, and 1.9953, respectively.

By reference to (6.2), and noting that $0 \leq \epsilon_p \leq 1$, and that ϵ_s is typically very small (e.g., 0.01), it can be seen τ is typically very small in elliptic filter applications. Therefore, for real x , $sn(x, \tau)$ is close to sinusoidal in shape for elliptic filters. The larger values of τ in Figure 6.4 are given for information purposes, to illustrate the effect of τ on the shape. This property yields, when properly set up, the equiripple magnitude frequency response in the passband of elliptic filters.

To illustrate the periodicity of the elliptic sine function with respect to an imaginary argument, plots of the function for several values of τ , with respect to jx , are shown in Figure 6.5. The horizontal axis is normalized to jX' . As can be seen, two complete periods, or $4X'$, are shown for each value of the modulus. In each plot, the function plotted is $sn(X + jx, \tau)$. It can be shown (e.g., Lawden, 1989) that

$$sn[(2m + 1)X + j(2n + 1)X', \tau] = \pm 1/\tau, \tag{6.7}$$

where m and n are integers. Note that this property, in part, is illustrated in Figure 6.5. The maximum values of the plots, occurring at X' and $3X'$, are equal to $1/\tau$, for each value of τ . It should be stressed, that even though the argument is complex for each of the plots in Figure 6.5, the elliptic sine function is real.

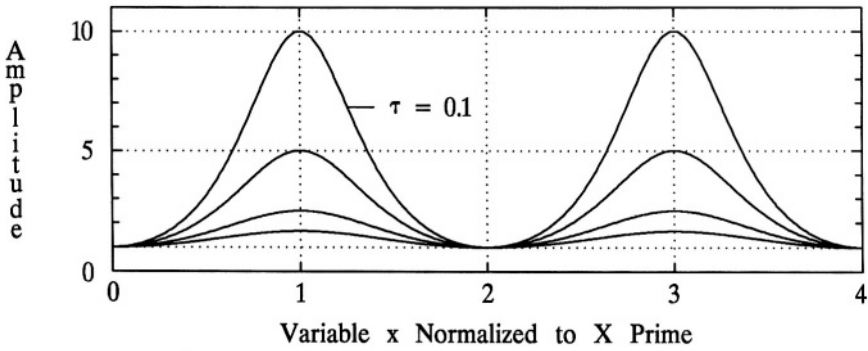


Figure 6.5 Plots of the elliptic sine function for several values of the modulus τ , versus an imaginary argument: $sn(X + jx, \tau)$.

For the four modulus values in the figure, 0.1, 0.2, 0.4, and 0.6, the corresponding X values are 1.5747, 1.5869, 1.6400, and 1.7508, respectively. The corresponding X' values are 3.6956, 3.0161, 2.3593, and 1.9953, respectively.

An even more important illustration of periodicity of the elliptic sine function with respect to real x , at least in the study of elliptic filters, is shown in **Figure 6.6**. It can readily be shown (e.g., Lawden, 1989) that

$$sn [2mX + j(2n + 1)X', \tau] = \pm \infty, \tag{6.8}$$

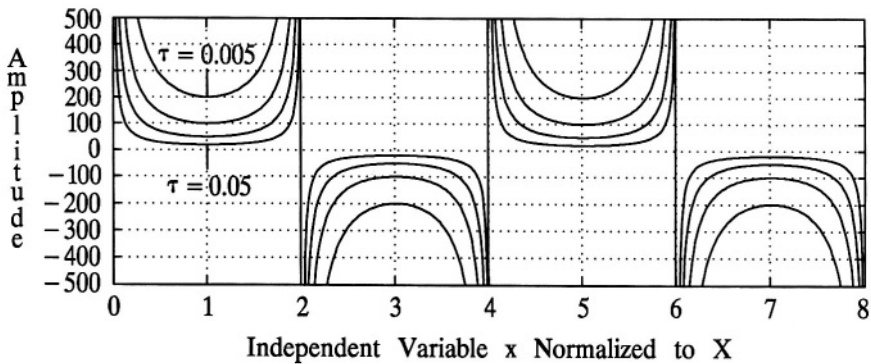


Figure 6.6 Plots of the elliptic sine function for several values of the modulus τ , versus real x : $sn(x + jX', \tau)$.

where m and n are integers. Therefore, plotting $sn(x + jX', \tau)$ with respect to real x , as is done in **Figure 6.6**, will exhibit poles as x goes through the $2mX$ points. Note that each plot in the figure goes to infinity at even multiples of X , but that the minimum-magnitude values, occurring at odd multiples of X , as noted in (6.7), are dependent on τ , in fact are equal to $1/\tau$. This property yields, when properly set up, the equiripple magnitude frequency response in the stopband of elliptic filters. It should be stressed, that even though the argument is complex for each of the plots in **Figure 6.6**, the elliptic sine function is real.

For the four modulus values in the figure, 0.005, 0.01, 0.02, and 0.05, the corresponding X values are 1.5708, 1.5708, 1.5710, and 1.5718, respectively. The corresponding X' values are 6.6846, 5.9916, 5.2987, and 4.3841, respectively.

Inverse Elliptic Sine Function with a Real Argument

Since the argument of the elliptic sine function in (6.2) involves a scaled inverse elliptic sine with a real argument, it is instructive to graphically consider such inverse elliptic sine functions. From (6.6), it is clear that for real $-1 \leq z \leq 1$ that, since $asin(z)$ is real, $sn^{-1}(z, \tau)$ will also be real. However, if $|z| > 1$, then $asin(z)$ will be complex. This is illustrated in **Figures 6.7** and **6.8** for several values of τ . In **Figure 6.7**, the real part of $sn^{-1}(z, \tau)$, the amplitude is normalized to X , whereas in **Figure 6.8**, the imaginary part of $sn^{-1}(z, \tau)$, the amplitude is normalized to X' . Note that the imaginary part is zero for $z < 1$, and that the imaginary part is $-X'$ for large values of z . Note also that the real part increases from zero to X as z increases from zero to unity, and then remains equal to X until z equals the value where the imaginary part becomes equal to X' . The value of z where the

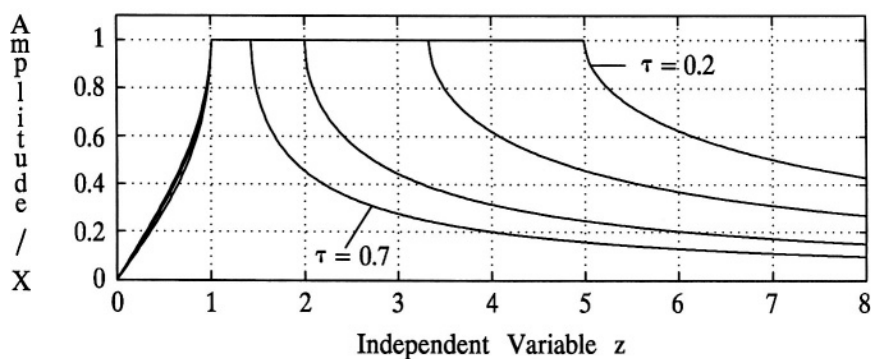


Figure 6.7 Plots of the real part of the inverse elliptic sine function for several values of the modulus τ , versus real z : $\Re e [sn^{-1}(z, \tau)]$.

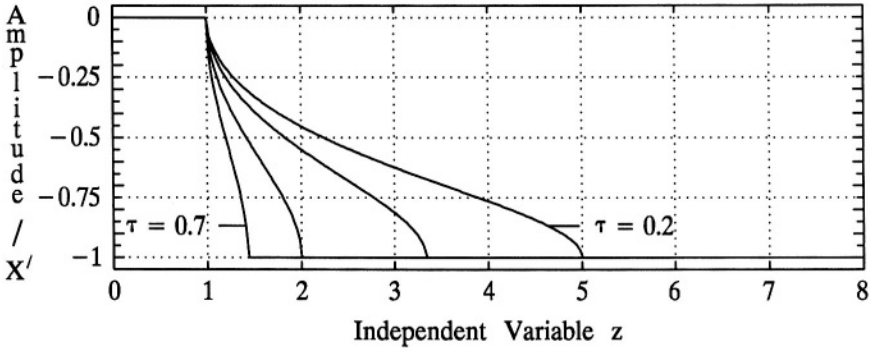


Figure 6.8 Plots of the imaginary part of the inverse elliptic sine function for several values of the modulus τ , versus real z : $\Im m[sn^{-1}(z, \tau)]$.

imaginary part just becomes X' , denoted as z_c , and is the maximum value of z where the real part is X , is

$$z_c = sn(X + jX', \tau) . \tag{6.9}$$

Note that (6.9) is the same as (6.7) with $m = n = 0$, and that, therefore,

$$z_c = 1/\tau .$$

It follows that

$$sn^{-1}(z_c, \tau) = sn^{-1}(1/\tau, \tau) = X + jX' . \tag{6.10}$$

By reference to (6.2), and noting that τ , here, is for the inverse elliptic sine, the values of 0.2, 0.3, 0.5 and 0.7 in Figures 6.7 and 6.8, are realistic values for elliptic filters. The corresponding X values are 1.5869, 1.6080, 1.6858, and 1.8457 respectively, and the corresponding X' values are 3.0161, 2.6278, 2.1565, and 1.8626 respectively.

6.3 EQUIRIPPLE PASSBAND AND STOPBAND MAGNITUDE

By comparing (6.5) and (6.6) with (6.2), define elliptic filter moduli as follows:

$$\tau_1 = \epsilon_p \epsilon_s , \quad \tau_2 = \omega_p / \omega_s . \tag{6.11}$$

Also, let

$$x = \kappa \operatorname{sn}^{-1}(\omega/\omega_p, \omega_p/\omega_s) , \quad z = \omega/\omega_p . \tag{6.12}$$

With definitions (6.11) and (6.12), (6.2) may be written

$$R_N = \operatorname{sn}[\kappa \operatorname{sn}^{-1}(z, \tau_2) + qX_1, \tau_1] = \operatorname{sn}(x + qX_1, \tau_1) . \tag{6.13}$$

Since, in the passband, $-1 \leq R_N \leq 1$ in (6.1) (see, e. g., Figure 6.4), A_p (see Figure 2.15 on page 52) equals $10 \log(1 + \epsilon_p^2)$. Therefore, given the A_p specification:

$$\epsilon_p = \sqrt{10^{A_p/10} - 1} . \tag{6.14}$$

Also, in the stopband, the minimum magnitude of R_N is $1/\tau_1 = 1/(\epsilon_p \epsilon_s)$, (see, e. g., Figure 6.6), and $A_s = 10 \log(1 + 1/\epsilon_s^2)$. Therefore, given the A_s specification:

$$\epsilon_s = \frac{1}{\sqrt{10^{A_s/10} - 1}} . \tag{6.15}$$

Since it is required that (6.1) have equiripple in the passband similar to a Chebyshev Type I response (see, e. g., Figure 6.2), it is clear that (6.2) squared must equal unity at $\omega = \omega_p$ ($x = 1$ in (6.13)). From (6.6) and (6.12) it is clear that

$$\operatorname{sn}^{-1}(1, \tau_2) = \operatorname{sn}^{-1}(1, \omega_p/\omega_s) = X_2 ,$$

where X_2 is the complete elliptic integral of τ_2 . Also, at $\omega = \omega_p$, from (6.2), $\operatorname{sn}(\gamma, \tau_1) = \pm 1$. Therefore, $\gamma = (2m + 1)X_1$, where X_1 is the complete elliptic integral of τ_1 and m is an integer (see, e. g., Figure 6.4). It follows that

$$\kappa X_2 = (2m + 1 - q)X_1 ,$$

or,

$$\kappa = \frac{(2m + 1 - q)X_1}{X_2} . \tag{6.16}$$

Also, from (6.7),

$$\operatorname{sn}[(2m + 1)X_1 + jX_1', \tau_1] = 1/\tau_1 ,$$

and it is required, by the specifications, that this equality hold when $\omega = \omega_s$.

Therefore,

$$\begin{aligned} \kappa sn^{-1}(\omega_s/\omega_p, \omega_p/\omega_s) + qX_1 &= \kappa sn^{-1}(1/\tau_2, \tau_2) + qX_1 \\ &= (2m + 1)X_1 + jX_1' \end{aligned} \quad (6.17)$$

From (6.17), it follows that

$$sn^{-1}(1/\tau_2, \tau_2) = \frac{(2m + 1 - q)X_1}{\kappa} + \frac{jX_1'}{\kappa} \quad (6.18)$$

However, from (6.16),

$$\frac{(2m + 1 - q)X_1}{\kappa} = X_2 \quad ,$$

and therefore (6.18) may be expressed

$$sn^{-1}(1/\tau_2, \tau_2) = X_2 + \frac{jX_1'}{\kappa} \quad (6.19)$$

However, from (6.10),

$$sn^{-1}(1/\tau_2, \tau_2) = X_2 + jX_2' \quad (6.20)$$

By comparing (6.19) with (6.20) it is apparent that

$$\kappa = \frac{X_1'}{X_2'} \quad (6.21)$$

Combining (6.16) and (6.21):

$$\kappa = \frac{(2m + 1 - q)X_1}{X_2} = \frac{X_1'}{X_2'} \quad (6.22)$$

Equation (6.22) is a statement of the critical conditions required by an elliptic filter design, in terms of the four complete elliptic integrals, that were discussed in general terms in **Section 6.1**. The complete elliptic integrals X_1 and X_1' are dependent upon the design parameters ϵ_p and ϵ_s , whereas X_2 and X_2' are dependent upon ω_p and ω_s , and the equality in (6.22) is also dependent upon m . Note that $2m + 1 - q$ is an odd integer if q is zero, and is an even integer if q is unity; it is equal, as will be shown, to the filter order N .

The reason why X_1 must be added to the elliptic sine argument in (6.2) when the order is even, is because the response at DC for even orders must be equal to the minimum value in the passband. Refer to **Figure 4.2** and note that even order Chebyshev Type I filters have a magnitude response that starts low, that is, the DC response is equal to $-A_p$. The same is true for even order elliptic filters.

Letting $N = 2m + 1 - q$ in (6.22):

$$\kappa = \frac{NX_1}{X_2} = \frac{X_1'}{X_2'} \quad (6.23)$$

which is valid for all N .

Example 6.3

Suppose the following design specifications for an elliptic filter are given:

$$N = 4, \quad \omega_p = 1, \quad A_p = 3 \text{ dB}, \quad A_s = 40 \text{ dB} .$$

It can be shown that an ω_s of 1.3461 will yield equality in (6.23) with $\kappa = 3.3058$. The complete elliptic integral values are: $X_1 = 1.5708$, $X_1' = 5.9915$, $X_2 = 1.9007$, and $X_2' = 1.8124$. A plot of (6.1) using (6.23) ($q = 1$) is shown in Figure 6.9. □

For the remainder of this section, it is assumed that (6.23) is satisfied. Note that magnitude-squared response peaks occur in the passband when (6.2) is zero. The frequencies of the response peaks may be found as follows. Observe from Figure 6.4 that $sn(x, \tau_1)$ is zero when $x = \kappa sn^{-1}(\omega/\omega_p, \omega_p/\omega_s) + qX_1 = 2mX_1$, or, substituting (6.23),

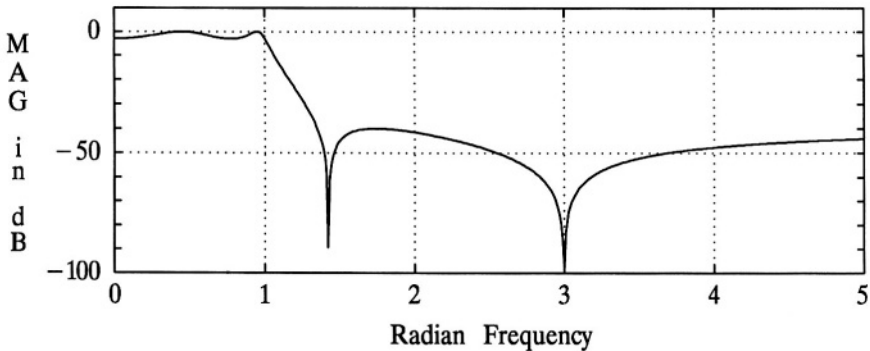


Figure 6.9 A plot of (6.1) for a fourth-order elliptic response. A plot of the data for Example 6.3.

$$\operatorname{sn}^{-1}(\omega/\omega_p, \omega_p/\omega_s) = \frac{2m - q}{N} X_2 \quad (6.24)$$

From (6.24),

$$\omega_m^{(\text{peak})} = \omega_p \operatorname{sn}[(2m - q)X_2/N, \omega_p/\omega_s] \quad (6.25)$$

where $m = 0, 1, \dots, (N-1)/2$ for N odd, and $m = 1, 2, \dots, N/2$ for N even.

Note also that magnitude-squared response valleys, i.e., minimum values, occur in the passband when (6.2) squared is unity. The frequencies of the passband response valleys may be found as follows. Observe from Figure 6.4 that $\operatorname{sn}(x, \tau_1)$ is ± 1 when $x = \kappa \operatorname{sn}^{-1}(\omega/\omega_p, \omega_p/\omega_s) + qX_1 = (2m + 1)X_1$, or, substituting (6.23),

$$\operatorname{sn}^{-1}(\omega/\omega_p, \omega_p/\omega_s) = \frac{2m + 1 - q}{N} X_2 \quad (6.26)$$

From (6.26),

$$\omega_m^{(\text{valley})} = \omega_p \operatorname{sn}[(2m + 1 - q)X_2/N, \omega_p/\omega_s] \quad (6.27)$$

where $m = 0, 1, \dots, (N-3)/2$ for N odd, and $m = 0, 1, \dots, (N-2)/2$ for N even.

Assuming that $A_p < 3 \text{ dB}$, and defining ω_c as the 3 dB corner frequency, it follows that $\omega_c > \omega_p$. This is the normal case, i.e., $A_p < 3 \text{ dB}$. If $A_p = 3 \text{ dB}$, then, of course, $\omega_c = \omega_p$; this was the case earlier in this chapter, where a large amount of passband ripple was desirable so that it was easily visible in the graphs. From (6.1) and (6.2), and noting from Figures 6.7 and 6.8 that ω_c only slightly greater than ω_p means that the inverse elliptic sine function in (6.2) will be

$$\operatorname{sn}^{-1}(\omega_c/\omega_p, \omega_p/\omega_s) = X_2 + jx \quad (6.28)$$

where x is small ($x = 0$ if $\omega_c = \omega_p$), and also noting that

$$\varepsilon_p \operatorname{sn}[\kappa \operatorname{sn}^{-1}(\omega_c/\omega_p, \omega_p/\omega_s) + qX_1, \varepsilon_p \varepsilon_s] = 1 \quad ,$$

it follows, making use of (6.23), that

$$\operatorname{sn}^{-1}(\omega_c/\omega_p, \omega_p/\omega_s) = X_2 + j \Im m \left\{ \frac{\operatorname{sn}^{-1}(1/\varepsilon_p, \varepsilon_p \varepsilon_s) - qX_1}{NX_1} X_2 \right\} \quad (6.29)$$

From (6.29) it follows that

$$\omega_c = \omega_p \operatorname{sn} \left[X_2 + j \operatorname{Im} \left\{ \frac{\operatorname{sn}^{-1}(1/\epsilon_p, \epsilon_p \epsilon_s) - q X_1}{N X_1} X_2 \right\}, \omega_p/\omega_s \right] \quad (6.30)$$

Equation (6.30) may be conveniently computed using the MATLAB function *ELLIPWC*, found on the accompanying disk.

The zero-transmission frequencies of (6.1) occur when (6.2) is infinity. By reference to Figure 6.6 it is noted that the elliptic sine function is infinity every $2X$:

$$\operatorname{sn}(2m X_1 + j X_1', \tau_1) = \pm \infty \quad .$$

By reference to Figures 6.7 and 6.8, and recalling (6.10), where it was noted that the frequency where the inverse elliptic sine function is $X_2 + j X_2'$ is $1/\tau_2 = \omega_s/\omega_p$, it follows that for frequencies greater than ω_s/ω_p , the inverse elliptic sine function will be $x + j X_2'$, where $x < X_2$. Therefore, for frequencies $> \omega_s/\omega_p$, the argument of the elliptic sine function will be

$$\kappa \operatorname{sn}^{-1}(\omega/\omega_p, \omega_p/\omega_s) + q X_1 = 2m X_1 + j X_1' \quad ,$$

and therefore,

$$\kappa(x + j X_2') = (2m - q) X_1 + j X_1' \quad . \quad (6.31)$$

Applying (6.23) to (6.31), it follows that

$$x = (2m - q) X_2/N \quad ,$$

from which

$$\operatorname{sn}^{-1}(\omega/\omega_p, \omega_p/\omega_s) = (2m - q) X_2/N + j X_2' \quad . \quad (6.32)$$

From (6.32) it follows that

$$\omega_m^{(zero)} = \omega_p \operatorname{sn} [(2m - q) X_2/N + j X_2', \omega_p/\omega_s] \quad , \quad (6.33)$$

where $m = 1, 2, \dots, (N - 1)/2$ for N odd, and $m = 1, 2, \dots, N/2$ for N even. Equation (6.33) may be conveniently computed using the MATLAB function *ELLIPWZ*, found on the accompanying disk. It should be noted that (6.33) computes only finite-value zero-transmission frequencies. Odd-order elliptic filters also have a transmission zero at infinite frequency.

Frequencies between the zero-transmission frequencies where the stopband attenuation is minimum are also of interest. The difference here, compared to the zero-transmission analysis above, may be seen by reference to **Figure 6.6**, where it is noted that minimum-attenuation frequencies occur at odd-multiples of X , as opposed to previous even-multiples. Therefore, the analysis is very similar resulting in the following:

$$\omega_m^{(min)} = \omega_p \operatorname{sn}[(2m + 1 - q) X_2 / N - j X_2', \omega_p / \omega_s] \quad , \quad (6.34)$$

where $m = 1, 2, \dots, (N - 1)/2$ for N odd, and $m = 1, 2, \dots, (N - 2)/2$ for N even. Equation (6.34) may be conveniently computed using the MATLAB function *ELLIPWM*, found on the accompanying disk. It should be noted that (6.34) computes only finite-value minimum-attenuation frequencies. Even-order elliptic filters also have a minimum-attenuation at infinite frequency.

See **Figure 6.10** for plots of (6.1) for a normalized ω_c of unity, a somewhat arbitrary, but common value of $A_p = 1 \text{ dB}$, an also somewhat arbitrary $A_s = 80 \text{ dB}$, and several values of N . These values are comparable to those used in **Figures 4.1** and **5.1**, for ready comparison. See **Figure 6.11** for detailed plots of (6.1) across the passband. In the figures solid lines are for even orders, and dashed lines are for odd orders.

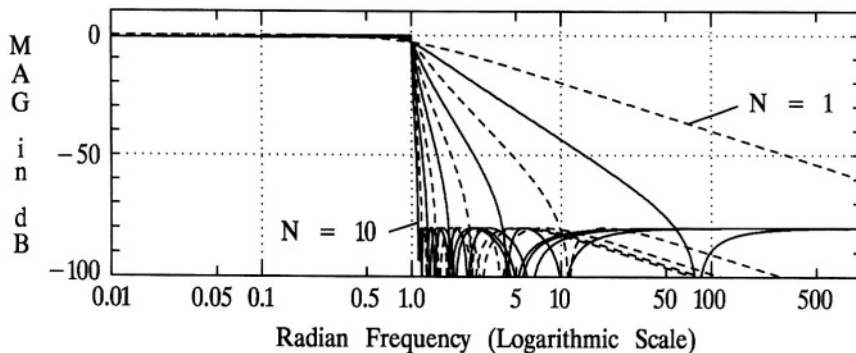


Figure 6.10 The elliptic filter magnitude response. Plots of (6.1) for $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and values of N from 1 through 10.

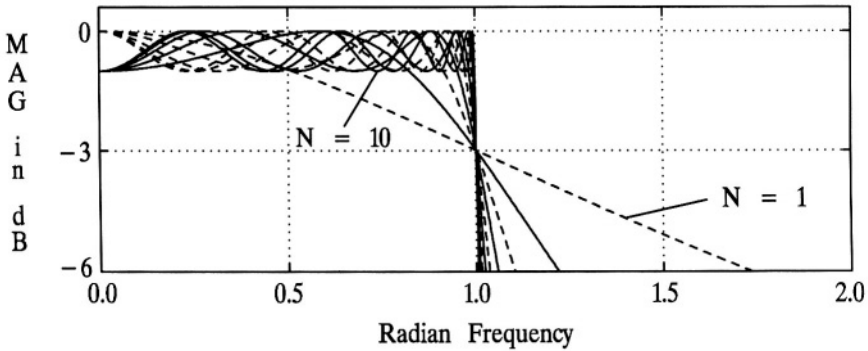


Figure 6.11 The elliptic filter magnitude response. Passband details of (6.1) for $\omega_c = 1$, $A_p = 1$ dB, $A_s = 80$ dB, and values of N from 1 through 10.

6.4 COMPUTING ω_s GIVEN A_p , A_s , N , AND ω_p

Given that (6.23) is not satisfied for an initial set of design parameters, the question arises as to how to make adjustments so that (6.23) is satisfied. Clearly the possibilities are infinite, as A_p , A_s , N , ω_p , and ω_s could all be varied, either in combination or individually.⁸ However, here, ω_s will be varied to achieve equality in (6.23), leaving the other design parameters fixed.

The following is obtainable directly from (6.23):

$$\frac{X_2}{X_2'} = \frac{NX_1}{X_1'} \quad , \quad (6.35)$$

and it is noted that the ratio X_2/X_2' is dependent only on ω_s , given that ω_p is known, and that NX_1/X_1' is dependent on the other design parameters which are also assumed known. Note that X_2 is inversely related to ω_s , and that X_2' is directly related. Therefore an algorithmic solution can easily be obtained, varying ω_s until (6.35), and therefore (6.23), is satisfied. One such algorithm, that enables a convenient computation of ω_s , is the MATLAB function *ELLIPWS*, found on the accompanying disk.

⁸The order N , of course, is restricted to the integers, and therefore can not, generally, achieve equality in (6.23) as an individual variable adjustment.

Example 6.4

Suppose the following design specifications for an elliptic filter are given:

$$N = 5, \quad \omega_p = 1, \quad A_p = 1 \text{ dB}, \quad A_s = 80 \text{ dB} \quad .$$

Using *ELLIPWS* and equations (6.25), (6.27), (6.30), (6.33), and (6.34), the following critical frequencies may be found: $\omega_s = 2.4880$, $\omega^{(peak)} = 0, 0.6047, 0.9549$, $\omega^{(valley)} = 0.3214, 0.8211$, $\omega_c = 1.0308$, $\omega^{(zero)} = 2.6054, 4.1147, \infty$, and $\omega^{(min)} = 3.0300, 7.7409$. \square

Example 6.5

Suppose the following design specifications for an elliptic filter are given:

$$N = 8, \quad \omega_p = 1, \quad A_p = 0.4 \text{ dB}, \quad A_s = 90 \text{ dB} \quad .$$

Using *ELLIPWS* and equations (6.25), (6.27), (6.30), (6.33), and (6.34), the following critical frequencies may be found: $\omega_s = 1.4588$, $\omega^{(peak)} = 0.2258, 0.6156, 0.8692, 0.9861$, $\omega^{(valley)} = 0, 0.4351, 0.7607, 0.9433$, $\omega_c = 1.0188$, $\omega^{(zero)} = 1.4794, 1.6783, 2.3696, 6.4610$, and $\omega^{(min)} = 1.5465, 1.9177, 3.3525, \infty$. \square

6.5 FILTER SELECTIVITY AND SHAPING FACTOR

Applying (2.37), the definition of *Filter Selectivity*, to the square root of (6.1), and making use of the fact that $1 + \epsilon_p^2 R_N^2 = 2$ by definition at $\omega = \omega_c$, results in

$$F_s = \frac{j \epsilon_p \kappa \operatorname{cn}(x, \epsilon_p \epsilon_s) \sqrt{1 - (\epsilon_p \epsilon_s)^2 \operatorname{sn}^2(x, \epsilon_p \epsilon_s)}}{2 \sqrt{2} \sqrt{1 - (\omega_p / \omega_s)^2} \cosh^2(\Im m[\operatorname{asin}(\omega_c / \omega_p)]) \sqrt{\omega_c^2 - \omega_p^2}} \quad , \quad (6.36)$$

where $x = \kappa \operatorname{sn}^{-1}(\omega_c / \omega_p, \omega_p / \omega_s) + q X_1$. A numerical solution, computing the *Filter Selectivity* for an elliptic filter, is available in the MATLAB function *ELLIPFS*, found on the accompanying disk.

Let A be an arbitrary attenuation in *dB* relative to the peak value such that $A_p \leq A \leq A_s$. From (6.1) and (6.2):

$$A = 10 \log (1 + \epsilon_p^2 \operatorname{sn}^2[\kappa \operatorname{sn}^{-1}(\omega / \omega_p, \omega_p / \omega_s) + q X_1, \epsilon_p \epsilon_s]) \quad . \quad (6.37)$$

For a given A , and making use of (6.14) and (6.23), solving (6.37) for ω would be equivalent to solving for the bandwidth at that attenuation A :

$$BW = \omega_p \operatorname{sn} \left[\frac{\operatorname{sn}^{-1} [\sqrt{10^{A/10}} - 1] / \sqrt{10^{A_p/10}} - 1, \epsilon_p \epsilon_s}{NX_1} X_2, \omega_p / \omega_s \right] . \quad (6.38)$$

Using (6.38) and applying (2.38), the definition of *Shaping Factor*, the elliptic filter *Shaping Factor* may be readily found:

$$S_a^b = \frac{BW_b}{BW_a} , \quad (6.39)$$

where BW_a is (6.38) evaluated with $A = a$, and BW_b is evaluated with $A = b$. A numerical solution, computing the *Shaping Factor* for an elliptic filter, is available in the MATLAB function *ELLIPSF*, found on the accompanying disk.

Example 6.6

Suppose $a = 3 \text{ dB}$, $b = 80 \text{ dB}$, $\omega_c = 1$, $A_p = 1 \text{ dB}$, and $A_s = 80 \text{ dB}$. From (6.36), for $N = 3, 4, \dots, 0$, F_s may be computed to be 2.25, 4.01, 6.60, 10.48, 16.42, 25.55, 39.65, and 61.48 respectively. From (6.39), for N from 3 through 10, S_3^{80} may be computed to be 9.81, 4.04, 2.41, 1.76, 1.44, 1.26, 1.16 and 1.10 respectively. The data in this example were obtained from the MATLAB m-file *EXAMP6_6.m*, as found on the accompanying disk and printed in **Appendix E**. □

6.6 DETERMINATION OF ORDER

An important step in the design of an analog filter is determining the minimum required order to meet given specifications. Refer to **Figure 2.15** on page 52 in specifying the desired filter magnitude characteristics. As long as the filter magnitude frequency response fits within the acceptable corridor indicated in **Figure 2.15**, it satisfies the specifications.

In the event that the filter specifications are exceeded, for elliptic filters there are several ways in which the specifications can be adjusted: allow ϵ_s to be made smaller (greater attenuation in the stopband than specified), allow ϵ_p to be made smaller (less ripple in the passband than specified), allow ω_s to be made smaller (a narrower transition band than specified), or some combination of these changes. An infinite combination exists, as was discussed in **Section 6.4**. If the initial specifications are to be exceeded, a smaller value for ω_s will be adhered to here, and all other specifications remaining unchanged. This is consistent with earlier chapters on Butterworth, Chebyshev Type I, and Chebyshev Type II filters, where a narrower

transition band was adhered to, keeping the other specifications unchanged. Therefore, from (6.35),

$$N \geq \frac{X_1' X_2}{X_1 X_2'}$$

or, following earlier chapters,

$$N = \left\lceil \frac{X_1' X_2}{X_1 X_2'} \right\rceil, \tag{6.40}$$

where $\lceil x \rceil$ is the smallest integer equal to or greater than x ($x \leq \lceil x \rceil < x + 1$). A numerical solution of (6.40), computing the minimum order for a lowpass elliptic filter, is available in the MATLAB function *ELLIPOR*, found on the accompanying disk. It should be noted that in using (6.40) the initial specification values are used. However, in using any of the analysis equations developed in this chapter, such as, for example, (6.30) or (6.33), the actual value of ω_s must be used (see Section 6.4).

Example 6.7

Suppose the following specifications are given: $f_p = 3,000$ Hz, $f_s = 7,000$ Hz, $A_p = 2$ dB, and $A_s = 60$ dB. From (6.40), $N = 4$. The actual value of f_s is found to be 6,733 Hz. □

6.7 CHEBYSHEV RATIONAL FUNCTIONS

In Section 4.4 a polynomial representation for the trigonometric and hyperbolic cosine form of C_N , as shown in (4.2), was developed. In this section, it is desirable to develop a similar representation for the elliptic sine form of R_N , as shown in (6.2). However, in this case a polynomial will not suffice, since R_N is not only equal to zero for certain values of ω , those identified as $\omega_m^{(peak)}$ in (6.25), but is also equal to infinity for certain values of ω , those identified as $\omega_m^{(zero)}$ in (6.33). What is required here is a ratio of polynomials, and that ratio that equals R_N in (6.2) is referred to as a *Chebyshev Rational Function*. Since the roots of the numerator and denominator are known (as expressed in (6.25) and (6.33)), R_N may be expressed as follows:

$$R_N = K \omega^{(1-q)} \prod_{m=1}^{\lfloor N/2 \rfloor} \frac{\omega^2 - (\omega_m^{(peak)})^2}{\omega^2 - (\omega_m^{(zero)})^2} , \tag{6.41}$$

where $\lceil x \rceil$ is the largest integer equal to or less than x ($x - 1 < \lceil x \rceil \leq x$), that is, $\lfloor N/2 \rfloor = N/2$ if N is even, and $\lfloor N/2 \rfloor = (N - 1)/2$ if N is odd, and K is a scaling constant such that $|R_N| = 1$ at each of the frequencies identified as $\omega_m^{(valley)}$ in (6.27) and at the passband edge ω_p . When N is even there is a valley at $\omega = 0$ and therefore a convenient means of computing K is as follows:

$$K = \prod_{m=1}^{\lfloor N/2 \rfloor} \frac{(\omega_m^{(zero)})^2}{(\omega_m^{(peak)})^2} , \quad N \text{ even} . \tag{6.42}$$

If N is odd the following may be used:

$$K = \frac{1}{\omega_p} \prod_{m=1}^{\lfloor N/2 \rfloor} \frac{(\omega_m^{(zero)})^2 - \omega_p^2}{\omega_p^2 - (\omega_m^{(peak)})^2} , \quad N \text{ odd} . \tag{6.43}$$

Note in (6.41) that the order N is the order of the numerator polynomial. If N is odd the order of the numerator polynomial inside the product is $N - 1$, and the additional ω term is supplied by $\omega^{(1-q)}$ ($q = 0$), which is associated with $\omega_0^{(peak)} = 0$.

Example 6.8

Let $\omega_p = 1$, $\epsilon_p = 1$ ($A_p = 3 \text{ dB}$), $A_s = 40 \text{ dB}$, and $N = 3$. These design parameters are the same as in Example 6.2, and therefore $\omega_s = 1.9789$. Using (6.25), the peak frequencies are found to be zero and 0.8814. Using (6.33), the zero frequency is found to be 2.2451. Using (6.43), $K = 18.1076$. Therefore,

$$R_N = K \omega \frac{\omega^2 - (\omega_1^{(peak)})^2}{\omega^2 - (\omega_1^{(zero)})^2} ,$$

and

$$|H(j\omega)|^2 = \frac{\omega^4 - 2(\omega_1^{(zero)})^2 \omega^2 + (\omega_1^{(zero)})^4}{K^2 \omega^6 + [1 - 2K^2(\omega_1^{(peak)})^2] \omega^4 + [K^2(\omega_1^{(peak)})^4 - 2(\omega_1^{(zero)})^2] \omega^2 + (\omega_1^{(zero)})^4} . \tag{6.44}$$

A plot of (6.44) is identical to the “*Ellip & Set 1*” graph in Figure 6.2. A MATLAB computation of this example may be found in the m-file *EXAMP6_8.m* on the accompanying disk. □

Example 6.9

Let $\omega_p = 1$, $A_p = 3 \text{ dB}$, $A_s = 40 \text{ dB}$, and $N = 4$. These design parameters are the same as in **Example 6.3**, and therefore $\omega_s = 1.3461$. Using (6.25), the peak frequencies are found to be 0.4491 and 0.9478. Using (6.33), the zero frequencies are found to be 1.4203 and 2.9971. Using (6.43), $K = 99.9952$. Therefore,

$$R_N = K \frac{(\omega^2 - (\omega_1^{(peak)})^2)(\omega^2 - (\omega_2^{(peak)})^2)}{(\omega^2 - (\omega_1^{(zero)})^2)(\omega^2 - (\omega_2^{(zero)})^2)} = K \frac{\omega^4 - a\omega^2 + b}{\omega^4 - c\omega^2 + d},$$

and

$$|H(j\omega)|^2 = \frac{\omega^8 - 2a\omega^6 + (c^2 + 2d)\omega^4 - 2cd\omega^2 + d^2}{(K^2 + 1)\omega^8 - (2aK^2 + 2c)\omega^6 + K^2(a^2 + 2b + c^2 + 2d)\omega^4 - 2(K^2ab + cd)\omega^2 + K^2b^2 + d^2}. \quad (6.45)$$

A plot of (6.45) is identical to that shown in **Figure 6.9**. A MATLAB computation of this example may be found in the m-file *EXAMP6_9.m* on the accompanying disk. \square

Note that (6.44) and (6.45) each satisfy the *Analog Filter Design Theorem*, as summarized on page 61. Each function is a polynomial in ω over a polynomial in ω with only real coefficients. There are only even powers of ω in those polynomials. The order of the numerator is no greater than the order of the denominator. The denominator has no real roots, and the roots of the numerator occur with even order. A consideration (6.41) in (6.1) will show that this is true in general.

6.8 LOCATION OF THE POLES AND ZEROS

Starting with (6.1) and following the procedure used in **Section 2.7**:

$$\begin{aligned} Y(s) &= H(s)H(-s) = \frac{1}{1 + \epsilon_p^2 R_N^2(-js)} \\ &= \frac{1}{1 + \epsilon_p^2 K^2 (-1)^{q+1} s^{2(1-q)} \left\{ \prod_{m=1}^{\lfloor N/2 \rfloor} \frac{s^2 + (\omega_m^{(peak)})^2}{s^2 + (\omega_m^{(zero)})^2} \right\}^2}. \end{aligned} \quad (6.46)$$

By rearranging (6.46) the numerator may be expressed as follows:

$$\prod_{m=1}^{\lfloor N/2 \rfloor} [s^2 + (\omega_m^{(zero)})^2]^2 \quad (6.47)$$

From (6.47) it immediately follows that the transfer function zeros of an elliptic filter are given by

$$s_m = \pm j \omega_m^{(zero)} \quad (6.48)$$

where $\omega_m^{(zero)}$ is given by (6.33).

An expression for the transfer function poles could be obtained from (6.46), however, a more effective approach is to begin with (6.1) and (6.2), and let

$$R_N = \frac{j}{\epsilon_p} = sn[\kappa sn^{-1}(-js_m/\omega_p, \omega_p/\omega_s) + (q + 4m)X_1, \epsilon_p \epsilon_s] \quad (6.49)$$

where the $4mX_1$ term reflects the real periodicity of the elliptic sine function. From (6.49) an expression for the poles follows:

$$s_m = j \omega_p sn[(sn^{-1}(j/\epsilon_p, \epsilon_p \epsilon_s) - (q + 4m)X_1)/\kappa, \omega_p/\omega_s] \quad (6.50)$$

where $m = 1, 2, \dots, N$. A MATLAB function, *ELLIPPZ*, found on the accompanying disk, computes the poles and zeros according to (6.48) and (6.50).

Example 6.10

Let $\omega_p = 1$, $A_p = 3 \text{ dB}$, $A_s = 40 \text{ dB}$, and $N = 3$. These design parameters are the same as in *Examples 6.2* and *6.8*. Using the MATLAB function *ELLIPPZ* mentioned above, the poles and zeros of the elliptic filter are as follows.

poles: -0.3225, -0.1337 ± j 0.9194 ,
 zeros: ± j 2.2451 .

□

Example 6.11

Let $\omega_p = 1$, $A_p = 3 \text{ dB}$, $A_s = 40 \text{ dB}$, and $N = 4$. These design parameters are the same as in *Examples 6.3* and *6.9*. Using the MATLAB function *ELLIPPZ*, the poles and zeros of the elliptic filter are as follows.

$$\begin{aligned} \text{poles: } & -0.0593 \pm j 0.9666, \quad -0.2269 \pm j 0.4709, \\ \text{zeros: } & \pm j 1.4203, \quad \pm j 2.9971. \end{aligned}$$

□

Example 6.12

Suppose it is desired to use a 5th-order elliptic filter with $f_p = 3000 \text{ Hz}$, $A_p = 1 \text{ dB}$, and $A_s = 70 \text{ dB}$. Using the MATLAB function *ELLIPZ*, the poles and zeros are found to be as follows.

$$\begin{aligned} \text{poles: } & -5,879, \quad -1,469 \pm j 18,715, \quad -4,381 \pm j 12,166, \\ \text{zeros: } & \pm j 39,894, \quad \pm j 62,142. \end{aligned}$$

□

6.9 PHASE RESPONSE, PHASE DELAY, AND GROUP DELAY

The phase response of an elliptic filter, with a normalized $\omega_c = 1$, a somewhat arbitrary, but common value of $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and several values of N , is shown in **Figure 6.12**. The phase response, from $\omega = 0$ until the first phase discontinuity, which occurs at $\omega = 1.11 \text{ rad/s}$ for the tenth-order

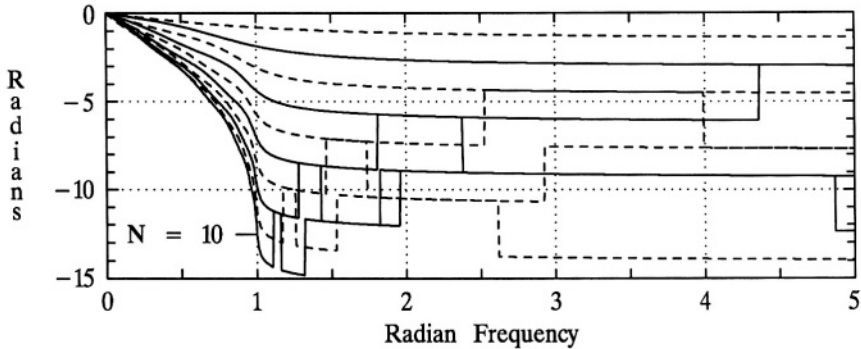


Figure 6.12 A plot of the phase response for an elliptic lowpass filter with normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and for values of N from 1 through 10.

response, is the total phase, in contrast to the principal phase. The phase response in **Figure 6.12** for ω beyond, and including, the first phase discontinuity is not total phase, but rather *pseudo-principal* phase. That is, the phase shown is the total phase plus $m2\pi \text{ rad}$, where m is an integer. This technique allows for a less congested set of plots that is easier to read. See **Section 5.6** for further discussion.

Taking the initial phase slope as a linear-phase reference, deviations from linear phase, for a normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and for several values of N , are shown in **Figure 6.13**. The phase deviation is shown in the figure from $\omega = 0$ until just before the first phase discontinuity occurs. Each phase discontinuity causes a $+\pi$ discontinuity in the phase deviation, but if plotted is somewhat misleading, since the magnitude response is zero at the same frequency and is, in general, in the stopband. Over the frequency range of the figure, $0 \leq \omega \leq 2 \text{ rad/s}$, phase discontinuities only effect the plots for orders 6 through 10, as can be seen in the figure.

The *phase delay*, $t_{pd}(\omega)$, for a filter is defined in (2.80), which is repeated here for convenience:

$$t_{pd}(\omega) = - \frac{\sphericalangle H(j\omega)}{\omega} .$$

The *group delay* for a filter, $t_{gd}(\omega)$, is defined by (2.81) and is repeated here for convenience:

$$t_{gd}(\omega) = - \frac{d \sphericalangle H(j\omega)}{d\omega} .$$

The phase delay of an elliptic filter, with a normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and for several values of N , is shown in **Figure 6.14**. Note that the

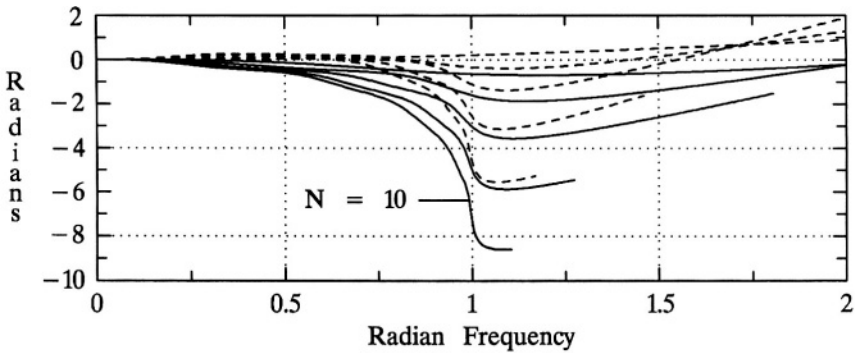


Figure 6.13 Phase deviation from linear for an elliptic filter with normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and for values of N from 1 through 10.

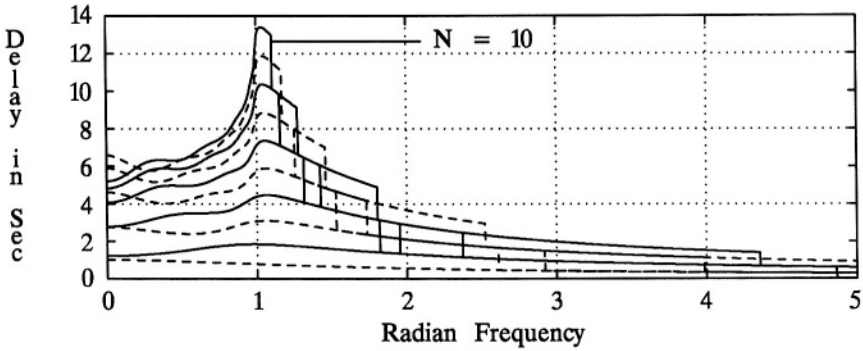


Figure 6.14 A plot of the phase delay for an elliptic filter with normalized $\omega_c = 1$, $A_p = 1$ dB, $A_s = 80$ dB, and for values of N from 1 through 10.

phase delay discontinuities occur at the frequencies where there are phase discontinuities, that is, at the frequencies of transmission zeros, which are, of course, in the stopband. Since each phase discontinuity is $+\pi$ rad, each phase delay discontinuity is $-\pi/\omega$ s, where ω is the frequency of the discontinuity.

The group delay is shown in **Figure 6.15**. The group delay values for lower radian frequencies are obscured due to the very large group delays for frequencies near unity for the larger orders. More detail may be seen in the “close-up” view

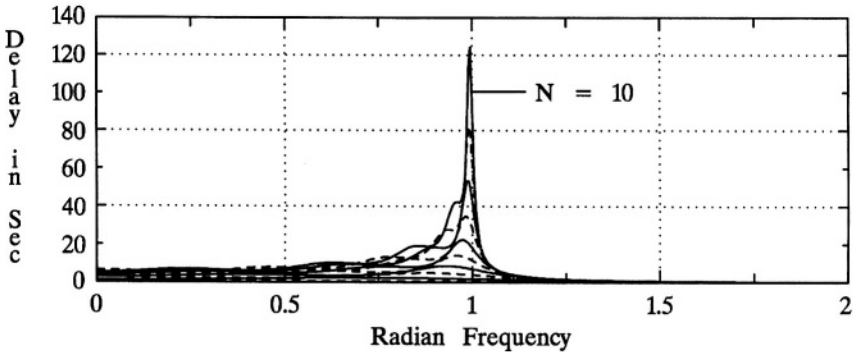


Figure 6.15 A plot of the group delay for an elliptic filter with normalized $\omega_c = 1$, $A_p = 1$ dB, $A_s = 80$ dB, and for values of N from 1 through 10.

shown in **Figure 6.16**. Note that the group delay values at *DC* are very close to the phase delay values at *DC*. However, as ω approaches the 3 dB corner frequency, in this case unity, the group delay becomes considerably larger than the phase delay due to the increasingly nonlinear phase response near the corner frequency (see **Figures 6.12** and **6.13**). Note the points of discontinuity in the group delay, for example at $\omega = 1.11 \text{ rad/s}$ for $N = 10$. As can be seen from the definition of group delay, at each $+\pi \text{ rad}$ phase discontinuity the group delay is theoretically $-\infty \text{ s}$. That is, the group delay is theoretically an infinite time *advance*, rather than a delay, at the point of a phase discontinuity; however, since this occurs only at a *point* along the frequency axis, and at a point of a transmission zero, the filter magnitude response at that point is zero; there is nothing to advance. In **Figures 6.15** and **6.16**, these *points* of infinite time advance are plotted with non-zero width; this is a result of the plotting software and from the fact that the calculation frequency-sample width is non-zero. Also, for convenience, the minimum delay value of the figure is zero.

6.10 TIME-DOMAIN RESPONSE

The unit impulse response of an elliptic filter, with a normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and for several values of N , is shown in **Figure 6.17**. Note that, as was the case for Chebyshev Type II filters, for even orders there is an impulse at the origin (not shown in the figure), but the weight of these impulses is insignificant. The unit step response of an elliptic filter, with normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and for several values of N , is shown in **Figure 6.18**.

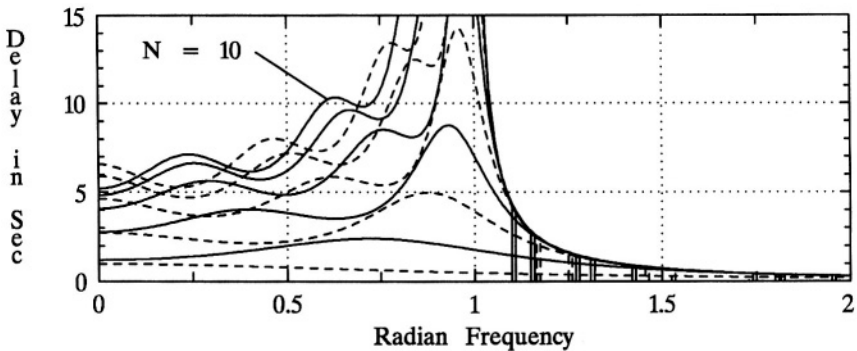


Figure 6.16 A “close-up” plot of the group delay for an elliptic filter with normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and for values of N from 1 through 10.

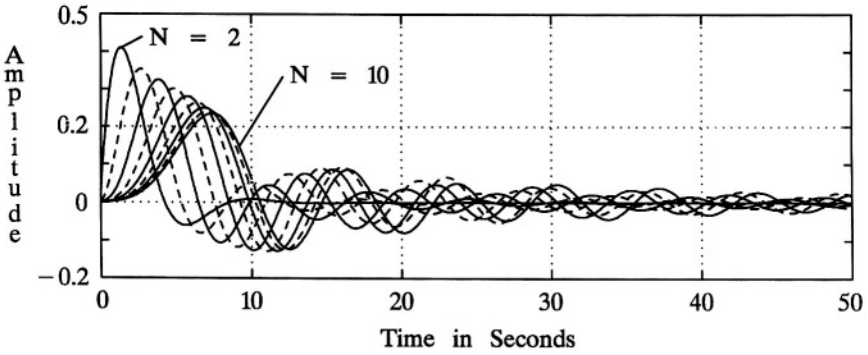


Figure 6.17 A plot of the unit impulse response for an elliptic filter with normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and for values of N from 2 through 10.

Note that the settled value of the step response is unity for odd orders, and $1/\sqrt{1 + \epsilon_p^2}$ (0.89125 in this case), the input being unity.

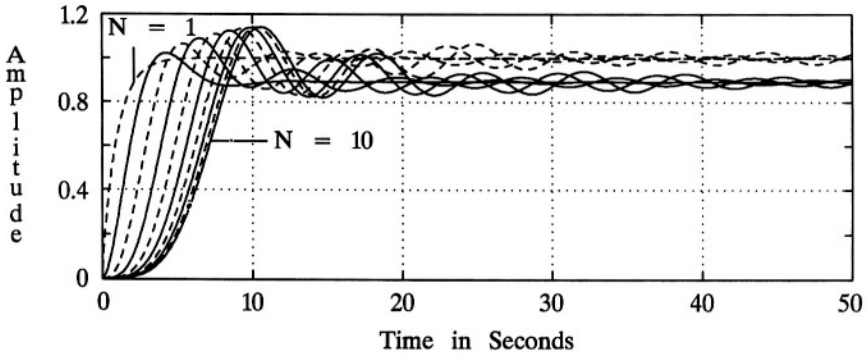


Figure 6.18 A plot of the unit step response for an elliptic filter with normalized $\omega_c = 1$, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, and for values of N from 1 through 10.

6.11 COMPARISON WITH PRECEDING FILTERS

There are several ways in which elliptic filters may be compared with the previously presented filters: Butterworth, Chebyshev Type I, and Chebyshev Type II filters. The magnitude frequency responses may be compared, the phase responses, the phase delays, the group delays, the unit impulse responses, and the unit step responses, for example.

It is noted that, for a given set of filter specifications, the minimum order required for an elliptic filter will never be greater than that required for any of the other classical filters presented in this book: it will usually be less. Consider the following two examples.

Example 6.13

Suppose the following magnitude frequency response specifications are given:

$$f_p = 3000 \text{ Hz}, \quad f_s = 10000 \text{ Hz}, \quad A_p = 1 \text{ dB}, \quad A_s = 70 \text{ dB} .$$

The minimum required order for a Butterworth filter is 8. The minimum required order for either a Chebyshev Type I filter or a Chebyshev Type II filter is 6. The minimum required order for an elliptic filter is 4. \square

Example 6.14

Suppose the following magnitude frequency response specifications are given:

$$f_p = 3000 \text{ Hz}, \quad f_s = 5000 \text{ Hz}, \quad A_p = 1 \text{ dB}, \quad A_s = 70 \text{ dB} .$$

The minimum required order for a Butterworth filter is 18. The minimum required order for either a Chebyshev Type I filter or a Chebyshev Type II filter is 9. The minimum required order for an elliptic filter is 6. \square

From *Examples 6.13* and *6.14*, and from the general statement at the top of the page that “the minimum order required for an elliptic filter will never be greater than that required for any of the other classical filters presented in this book: it will usually be less”, it may be concluded that if only the magnitude frequency response is of interest, then an elliptic filter would always be the one to choose. If only the minimum order were of interest, this would be true, however there are often other issues beyond the scope of this book to be considered. One of those issues is that some components in an elliptic filter realization, due to high Q stages, will have very

critical and sensitive values compared, for example, to a Butterworth filter, which means it will be more difficult to implement or may require more expensive components. These comments serve to suggest that even when only the magnitude frequency response is of interest, there are other issues besides minimum order that need to be considered in practical engineering design.

In terms of the phase response, an elliptic filter is the most nonlinear of any of the classical filter designs. This can be seen by comparing **Figure 6.12** with **Figures 5.3, 4.4, and 3.6**, or by comparing **Figure 6.13** with **Figures 5.4, 4.5, and 3.7**. This nonlinear phase response is reflected in the phase delay and especially in the group delay. Compare **Figure 6.15** with **Figures 5.6, 4.7, and 3.9**. Also note that the amount of ringing in the impulse response and the step response of an elliptic filter is comparable to that of a Chebyshev Type I, and perhaps slightly worse: compare **Figure 6.17** with **4.8**, and **Figure 6.18** with **4.9**.

In summary, when the primary consideration in selecting a filter type is using a minimum order to meet the magnitude response specifications, then an elliptic filter may be the best choice. However, frequently there are other factors to be considered as well, and this stresses the importance of having the design objectives clearly and completely stated.

6.12 CHAPTER 6 PROBLEMS

- 6.1** Confirm the results of *Example 6.1* by executing the MATLAB m-file *EXAMP6_1.m* and reviewing the contents of the file. Compare your graphical results with that shown in **Figure 6.1**.
- 6.2** Conduct an experiment similar to *Example 6.1*, but for $N = 4$. That is, let $\omega_p = 1$, $\omega_s = \infty$, $\epsilon_p = 1$, $\epsilon_s = 0$, $N = 4$, and show that a fourth-order Chebyshev Type I filter is a special case of an elliptic filter, that in this case the elliptic sine form and the elliptic cosine form of (6.2) are the same, and that in this case $\kappa = N$. To do this problem, modify *EXAMP6_1.m*. Note that $qX_1 = \pi/2$ in (6.2) but that no similar term is required for the *cn* case.
- 6.3** Conduct an experiment similar to *Example 6.1*, but for $N = 5$. That is, let $\omega_p = 1$, $\omega_s = \infty$, $\epsilon_p = 1$, $\epsilon_s = 0$, $N = 5$, and show that a fourth-order Chebyshev Type I filter is a special case of an elliptic filter, that in this case the elliptic sine form and the elliptic cosine form of (6.2) are the same, and that in this case $\kappa = N$. To do this problem, modify *EXAMP6_1.m*.

- 6.4** Confirm the results of *Example 6.2* by executing the MATLAB m-file *EXAMP6_2.m* and reviewing the contents of the file. Compare your graphical results with that shown in **Figures 6.2** and **6.3**.
- 6.5** Compute by hand, to five significant places, the numerical value of the complete elliptic integral for τ equals 0, 0.8, 0.999, and 1.
- 6.6** Compute by hand the numerical value of the Jacobian elliptic sine and cosine functions when $x = 1.8541$ and $\tau = 1/\sqrt{2}$. Note that $\phi = \pi/2$.
- 6.7** Compute by hand numerical values for the inverse Jacobian elliptic sine and cosine for $\phi = \pi/2$ and $\tau = 0, 0.8, 0.999$, and 1.
- 6.8** Determine numerical values for the imaginary period and the real period of Jacobian elliptic functions if $\tau = 0.5$.
- 6.9** Confirm the results of *Example 6.3*, given that $\omega_s = 1.3461$. That is, given that $N = 4$, $\omega_p = 1$, $A_p = 3 \text{ dB}$ (actually, $10 \log(2) \text{ dB}$), and $A_s = 40 \text{ dB}$, show that the complete elliptic integral values are as shown, and therefore that κ is as shown, and then use those results to plot **(6.1)** and compare your graphical results to that shown in **Figure 6.9**. Use the MATLAB function *ELLIPKE* to evaluate the complete elliptic integrals, and use *ELLIPSN* and *ELLIPINT* to evaluate the elliptic sine function and the inverse elliptic sine.
- 6.10** Confirm the results of *Example 6.4*. That is, given that $N = 5$, $\omega_p = 1$, $A_p = 1 \text{ dB}$, and $A_s = 80 \text{ dB}$, determine, using **(6.25)**, **(6.27)**, **(6.30)**, **(6.33)** and **(6.34)**, the critical frequencies ω_s , $\omega^{(peak)}$, $\omega^{(valley)}$, ω_c , $\omega^{(zero)}$, and $\omega^{(min)}$. Use the following MATLAB functions on the accompanying disk: *ELLIPWS*, *ELLIPPV*, *ELLIPWC*, *ELLIPWZ*, and *ELLIPMN*.
- 6.11** Confirm the results of *Example 6.5*. That is, given that $N = 8$, $\omega_p = 1$, $A_p = 0.4 \text{ dB}$, and $A_s = 90 \text{ dB}$, determine, using **(6.25)**, **(6.27)**, **(6.30)**, **(6.33)** and **(6.34)**, the critical frequencies ω_s , $\omega^{(peak)}$, $\omega^{(valley)}$, ω_c , $\omega^{(zero)}$, and $\omega^{(min)}$. Use the following MATLAB functions on the accompanying disk: *ELLIPWS*, *ELLIPPV*, *ELLIPWC*, *ELLIPWZ*, and *ELLIPMN*.
- 6.12** Similar to *Examples 6.4* and *6.5*, and **Problems 6.6** and **6.7**, given that $N = 7$, $f_p = 3,000 \text{ Hz}$, $A_p = 1.5 \text{ dB}$, and $A_s = 80 \text{ dB}$, determine the critical frequencies frequencies ω_s , $\omega^{(peak)}$, $\omega^{(valley)}$, ω_c , $\omega^{(zero)}$, and $\omega^{(min)}$.

- 6.13** Confirm the results of *Example 6.6* using the MATLAB functions *ELLIPFS* and *ELLIPSF*. That is, given that $a = 6 \text{ dB}$, $b = 60 \text{ dB}$, $\omega_c = 1$, $A_p = 1 \text{ dB}$, and $A_s = 80 \text{ dB}$, determine the *Filter Selectivity and Shaping Factor* for $N = 3, 4, \dots, 10$.
- 6.14** Suppose the following specifications are given: $f_p = 3,000 \text{ Hz}$, $f_s = 7,000 \text{ Hz}$, $A_p = 2 \text{ dB}$, and $A_s = 60 \text{ dB}$. From (6.40), determine the minimum required elliptic filter order to meet these specifications. Also determine the actual value of f_s that the elliptic filter would have. That is, confirm the results of *Example 6.7*.
- 6.15** Suppose the following specifications are given: $f_p = 5,000 \text{ Hz}$, $f_s = 7,000 \text{ Hz}$, $A_p = 1 \text{ dB}$, and $A_s = 75 \text{ dB}$. From (6.40), determine the minimum required elliptic filter order to meet these specifications. Also determine the actual value of f_s that the elliptic filter would have.
- 6.16** Similar to *Example 6.8*, let $\omega_p = 1$, $A_p = 1 \text{ dB}$, $A_s = 40 \text{ dB}$, and $N = 3$. Using these design parameters determine the rational function form for $|H(j\omega)|^2$, similar to (6.44) but with numerical values. Plot this function, using a dB vertical scale.
- 6.17** Let $\omega_p = 1$, $A_p = 3 \text{ dB}$, $A_s = 40 \text{ dB}$, and $N = 3$. Using the MATLAB function *ELLIPZ* which implements (6.48) and (6.50), determine the poles and zeros of an elliptic filter with the given specifications. This will confirm the results of *Example 6.10*.
- 6.18** Given the poles and zeros indicated in *Example 6.10* and the form of the transfer function indicated in (2.40):
- Determine K such that the peak magnitude frequency response in the passband is unity.
 - Express the transfer function in the form of (2.40) with numerical values given for all coefficients.
 - Frequency scale the transfer function so that $\omega_p = 1000$, and express the new transfer function in the form of (2.40) with numerical values given for all coefficients.
- 6.19** Let $\omega_p = 1$, $A_p = 3 \text{ dB}$, $A_s = 40 \text{ dB}$, and $N = 4$. Using the MATLAB function *ELLIPZ*, determine the poles and zeros of an elliptic filter with the given specifications. This will confirm the results of *Example 6.11*.

- 6.20** Given the poles and zeros indicated in *Example 6.11* and the form of the transfer function indicated in (2.40):
- Determine K such that the peak magnitude frequency response in the passband is unity.
 - Express the transfer function in the form of (2.40) with numerical values given for all coefficients.
 - Frequency scale the transfer function so that $\omega_p = 1000$, and express the new transfer function in the form of (2.40) with numerical values given for all coefficients.
- 6.21** Suppose it is desired to use a 5th-order elliptic filter with $f_p = 3000$ Hz, $A_p = 1$ dB, and $A_s = 70$ dB. Using the MATLAB function *ELLIPZ*, determine the poles and zeros of an elliptic filter with the given specifications. This will confirm the results of *Example 6.12*.
- 6.22** Given the poles and zeros indicated in *Example 6.12* and the form of the transfer function indicated in (2.40):
- Determine K such that the peak magnitude frequency response in the passband is unity.
 - Express the transfer function in the form of (2.40) with numerical values given for all coefficients.
 - Frequency scale the transfer function so that $f_p = 5000$ Hz, and express the new transfer function in the form of (2.40) with numerical values given for all coefficients.
- 6.23** Suppose it is desired to use a 6th-order elliptic filter with $f_p = 3000$ Hz, $A_p = 1$ dB, and $A_s = 70$ dB. Using the MATLAB function *ELLIPZ*, determine the poles and zeros of an elliptic filter with the given specifications.
- 6.24** Suppose the following filter specifications are given: $\omega_c = 1000$ rad/s, $A_s = 60$ dB, and $N = 6$.
- Determine the value of *Filter Selectivity* for a Chebyshev Type II filter with the given specifications.
 - Determine the value of *Filter Selectivity* for a Butterworth filter with similar specifications: $\omega_c = 1000$ rad/s, and $N = 6$.
 - Compare the above values with the value of the *Filter Selectivity* for a Chebyshev Type I filter with similar specifications: $\omega_c = 1000$ rad/s, $A_p = 1.5$ dB, and $N = 6$.
 - Compare the above values with the value of the *Filter Selectivity* for an elliptic filter with similar specifications: $\omega_c = 1000$ rad/s, $A_p = 1.5$ dB, $A_s = 60$ dB, and $N = 6$.

- 6.25** Repeat **Problem 6.24** for the *Shaping Factor* with $a = 3 \text{ dB}$ and $b = 60 \text{ dB}$. That is, suppose the following filter specifications are given: $\omega_c = 1000 \text{ rad/s}$, $A_s = 60 \text{ dB}$, and $N = 6$.
- Determine the value of the *Shaping Factor* for a Chebyshev Type II filter with the given specifications.
 - Determine the value of the *Shaping Factor* for a Butterworth filter with similar specifications: $\omega_c = 1000 \text{ rad/s}$, and $N = 6$.
 - Compare the above values with the value of the *Shaping Factor* for a Chebyshev Type I filter with similar specifications: $\omega_c = 1000 \text{ rad/s}$, $A_p = 1.5 \text{ dB}$, and $N = 6$.
 - Compare the above values with the value of the *Shaping Factor* for an elliptic filter with similar specifications: $\omega_c = 1000 \text{ rad/s}$, $A_p = 1.5 \text{ dB}$, $A_s = 60 \text{ dB}$, and $N = 6$.
- 6.26** Suppose filter specifications are stated as follows: $f_c = 3500 \text{ Hz}$, $f_s = 7000 \text{ Hz}$. Determine the required filter order to meet the given specifications:
- For an elliptic filter with $A_p = 1 \text{ dB}$ and $A_s = 40 \text{ dB}$.
 - For an elliptic filter with $A_p = 1 \text{ dB}$ and $A_s = 60 \text{ dB}$.
 - For an elliptic filter with $A_p = 1 \text{ dB}$ and $A_s = 80 \text{ dB}$.
 - For an elliptic filter with $A_p = 1 \text{ dB}$ and $A_s = 100 \text{ dB}$.
 - For comparison purposes, repeat parts (a) through (d) for a Butterworth filter ($A_p = 3 \text{ dB}$).
 - For comparison purposes, repeat parts (a) through (d) for a Chebyshev Type II filter ($A_p = 3 \text{ dB}$).
 - For comparison purposes, repeat parts (a) through (d) for an elliptic filter, but with $A_p = 0.1 \text{ dB}$.
 - For comparison purposes, repeat parts (a) through (d) for an elliptic filter, but with $A_p = 0.5 \text{ dB}$.
 - For comparison purposes, repeat parts (a) through (d) for an elliptic filter, but with $A_p = 1.5 \text{ dB}$.
 - For comparison purposes, repeat parts (a) through (d) for a Chebyshev Type I filter with 0.1 dB of ripple.
 - For comparison purposes, repeat parts (a) through (d) for a Chebyshev Type I filter with 0.5 dB of ripple.
 - For comparison purposes, repeat parts (a) through (d) for a Chebyshev Type I filter with 1.5 dB of ripple.
- 6.27** By making use of the scaling property of Fourier transforms and **Figure 6.17**, determine, for a tenth-order elliptic filter with $f_c = 3,000 \text{ Hz}$, approximate values for the following:
- The time at which the unit impulse response is a maximum.

- (b) The amplitude of the unit impulse response maximum.
- (c) The half-value width of the unit impulse response, defined as the time during which $0.5 h_{\max} \leq h(t) \leq h_{\max}$.
- (d) The *duration*, t_D , of the unit impulse response, defined as the time from $t = 0$ until $|h(t)| < 0.1 h_{\max}$ for all $t > t_D$.

6.28 Repeat **Problem 6.27** for $f_c = 10,000 \text{ Hz}$.

6.29 Sketch the step response of a 10th-order elliptic filter with $f_c = 1000 \text{ Hz}$, $A_p = 1 \text{ dB}$, and $A_s = 80 \text{ dB}$. Refer to **Figure 6.18** and make use of the scaling property of Fourier transforms. What would the maximum group delay be for this filter, and at what frequency would it occur? At what time would the peak of the unit impulse response of this filter be, and what would be the value of that peak?

6.30 Suppose an anti-aliasing filter is needed prior to an analog-to-digital converter to be used in a speech processing system. Suppose a sampling rate of 8000 *samples/s* is to be used, and therefore it is decided that the anti-aliasing filter should have a minimum attenuation of 60 *dB* at 4000 *Hz*. Suppose an elliptic filter is to be used.

- (a) If $f_p = 3000 \text{ Hz}$ and $A_p = 1 \text{ dB}$, what is the minimum order required?
- (b) If $A_p = 1 \text{ dB}$ and $N = 10$, what is maximum value of f_p ?
- (c) If $f_p = 3000 \text{ Hz}$ and $N = 10$, what is the minimum value of A_p ?

6.31 Under the conditions of part (c) of **Problem 6.30**, determine the transfer function $H(s)$, and give numerical values for all the poles and zeros.

6.32 Using the MATLAB functions *ellipap*, *impz* and *step*:

- (a) Determine the transfer function in polynomial form, and also factored to indicate the poles and zeros, of an elliptic filter with $\omega_p = 1$, $A_p = 1.2 \text{ dB}$, $A_s = 70 \text{ dB}$, and $N = 6$.
- (b) Determine the impulse response and the step response for the filter of part (a).
- (c) By multiplying the pole vector and the zero vector found in part (a) by $2\pi 1000$ determine the transfer function of an elliptic filter with $f_p = 1000 \text{ Hz}$, $A_p = 1.2 \text{ dB}$, $A_s = 70 \text{ dB}$, and $N = 6$.
- (d) Determine and plot the magnitude frequency response of the filter of part (c) by using the MATLAB function *freqs*. Use a vertical scale in *dB* and a linear horizontal scale from 0 to 5000 *Hz*. Also determine and plot the phase response over this same frequency range. Use the MATLAB function *unwrap* rather than plotting the principle phase.

- (e) By appropriately scaling the impulse response and the step response of part (b), determine and plot the impulse response and the step response of the filter of part (c). That is, the time axis for the step response needs to be scaled by $1/(2\pi 1000)$, and the unit impulse response needs the same time-axis scaling and requires an amplitude scaling of $2\pi 1000$.
- (f) Determine and plot the phase delay of the filter of part (c). Note that this is easily obtained from the phase response of part (d).
- (g) Determine and plot the group delay of the filter of part (c). Note that this also is easily obtained from the phase response of part (d):

$$t_{gd}(n) \cong -[\phi(n) - \phi(n-1)]/S_s$$
, where $\phi(n)$ is the phase in radians at step n , and S_s is the step size in *rad/s*.

6.33 To demonstrate the critical nature of a filter design, this problem experiments some with the filter of **Problem 6.32** (c). Multiply the highest-frequency pole-pair of the filter of **Problem 6.32** (c) by 1.1, leaving all other poles and zeros unchanged. Determine and plot the magnitude frequency response of the filter by using the MATLAB function *freqs*. Use a vertical scale in *dB* and a linear horizontal scale from 0 to 5000 *Hz*. Also determine and plot the phase response over this same frequency range. Use the MATLAB function *unwrap* rather than plotting the principle phase. Compare these results with that obtained for **Problem 6.32** (d).

6.34 This problem continues to demonstrate the critical nature of a filter design, and is a continuation of **Problem 6.33**. Multiply the lowest-frequency zero-pair of the filter of **Problem 6.33** (c) by 1.1, leaving all other poles and zeros unchanged. Determine and plot the magnitude frequency response of the filter by using the MATLAB function *freqs*. Use a vertical scale in *dB* and a linear horizontal scale from 0 to 5000 *Hz*. Also determine and plot the phase response over this same frequency range. Use the MATLAB function *unwrap* rather than plotting the principle phase. Compare these results with that obtained for **Problem 6.32** (d).

CHAPTER 7

BESSEL FILTERS

Although all of the classical filters presented up to this point are defined in terms of the magnitude frequency response, and are designed primarily to meet given magnitude frequency response specifications, such is not the case with Bessel filters. Bessel filters are designed to achieve a maximum frequency bandwidth while maintaining a constant time delay: as initially introduced, a Bessel filter is a constant time-delay network.

As noted in **Section 1.4**, Bessel filters are based on Bessel (lived 1784-1846) polynomials (Krall and Frink, 1949; Grosswald, 1951; Burchnell, 1951), and were introduced by Thomson and Storch (Thomson, 1949; Thomson, 1959; Storch, 1954). In fact, because of Thomson's contributions to the development of Bessel filters they are sometimes referred to as Thomson filters, or Bessel-Thomson filters.

Although, as mentioned above, Bessel filters are designed for a constant time-delay independent of frequency, as will be seen below in the development, nevertheless they result in a lowpass filter. Therefore, the presentation of Bessel filters below will proceed along two paths: as a constant time-delay circuit, and as a lowpass filter that can be compared with other classical filter types.

In this chapter, after some introductory material, the Bessel filter response is developed and defined, and it will be observed that it satisfies the *Analog Filter Design Theorem*. Explicit formulas for the design and analysis of Bessel filters, such as *Filter Selectivity*, *Shaping Factor*, the minimum required order to meet design specifications, etc., will be obtained. The transfer function $H(s)$ will be determined, and to complete the study of lowpass, prototype Bessel filters, the phase response, phase delay, and time-domain response characteristics are investigated.

7.1 INTRODUCTION

In **Section 2.12** the importance of the phase response was briefly considered. It was noted that, for ideal transmission from the input to the output of a filter the phase response must be linear. That is, suppose $x(t)$ is the input to an analog filter. While the filter may be designed to attenuate certain unwanted components in $x(t)$,

that part of $x(t)$ that falls within the passband of the filter is desired to be passed with minimal effect on the waveshape of the signal. Suppose $x_p(t)$ is that part of $x(t)$ that falls within the passband of the filter, then the desired filter output would be $y(t) = K_p x_p(t - t_d)$, where K_p is a gain term (could be unity) and t_d is a time delay (a time delay of zero is not practical, and a small delay will not affect the waveshape). From the basic properties of Fourier transforms, it follows that the required filter frequency response, magnitude and phase, across the passband would be as follows:

$$H(j\omega) = K_p e^{-jt_d\omega} \quad (7.1)$$

Therefore, the magnitude frequency response should be constant across the passband and the phase response should be linear ($\angle H(j\omega) = -t_d\omega$). Therefore in applications where preserving the waveshape of signals is important, the phase response is of special importance.

An application where having a linear phase response so as to preserve the waveshape is of special importance is in radar and sonar receivers. In both radar and sonar, pulses are transmitted which in turn reflect off objects that may be of interest. Analysis of the reflected pulses, which are received, can yield a variety of informative data about the target from which the reflection came, such as distance, bearing and velocity. To preserve the waveshape of the reflections, it is important that any filters in the radar and sonar receivers have good (nearly linear) phase characteristics.

Another application is a filter in an oscilloscope amplifier, or any filtering done prior to the signal being applied to an oscilloscope input. Since the oscilloscope is being used to electronically display a graph of the signal waveform, it is important to be able to assume that the displayed graph is accurate, and that any “ringing” or distortion caused by filtering is minimal.

Another application is in state-of-the-art “high-end” high fidelity audio systems, such as in a graphics equalizer, which is a bank of filters, allowing the gain to be adjusted in multiple frequency bands across the audio frequency spectrum. Such a graphics equalizer is intended to compensate for room acoustics or non-ideal frequency response characteristics of loudspeakers. If the phase response of the filters are not linear, or equivalently the group delay characteristics are not constant, then the equalizer will alter the waveshape of the audio signal even when all filters have the same gain, resulting in a distorted signal.

In some applications what is desired is a time delay only, without any magnitude frequency response alterations. This may be desired in order to time-align a signal with another signal which has experienced an inevitable time delay itself. An example of such a situation is in the practical production of an *analytic signal*. The study of *analytic signals* often arises in the analysis and production of single side-band modulation (Proakis and Salehi, 1994; Stremler, 1990). The analytic signal may be defined as follows:

$$z(t) = f(t) + jg(t) \quad (7.2)$$

where $g(t)$ is the Hilbert transform of $f(t)$, as discussed in **Section 2.15**, beginning near the bottom of page 90. An approximation to a Hilbert transform filter can be designed, which is an all-pass filter (as discussed in **Section 2.12**) with an ideal phase response of -90° for all positive frequencies and $+90^\circ$ for all negative frequencies. However, any analog filter that approximates a Hilbert transform will have some inevitable time delay, which means that $g(t)$ in (7.2) will approximate the Hilbert transform of $f(t-t_d)$ rather than of $f(t)$. Therefore, the analytic signal may better be represented as follows:

$$z(t) = f(t-t_d) + j g(t-t_d) .$$

Therefore, in a practical implementation of the analytic signal, it is necessary to delay $f(t)$ by t_d seconds. One way to accomplish this delay, especially at lower frequencies, is with a Bessel filter.

7.2 MAXIMALLY-FLAT GROUP DELAY

Consider a general all-pole transfer function with a DC gain of unity, of the form indicated in (2.39):

$$H(s) = \frac{1}{\sum_{k=0}^N a_k s^k} , \tag{7.3}$$

where $a_0 = 1$. From (7.3) it follows that:

$$H(j\omega) = \frac{1}{\sum_{k \text{ even}}^N (-1)^{k/2} a_k \omega^k + j\omega \sum_{k \text{ odd}}^N (-1)^{(k-1)/2} a_k \omega^{k-1}} , \tag{7.4}$$

where $k \text{ even}$ includes $k = 0$. From (7.4):

$$\angle H(j\omega) = -\tan^{-1} \left[\frac{\omega \sum_{k \text{ odd}}^N (-1)^{(k-1)/2} a_k \omega^{k-1}}{\sum_{k \text{ even}}^N (-1)^{k/2} a_k \omega^k} \right] . \tag{7.5}$$

Applying the definition of group delay, as given by (2.81), to (7.5):

$$t_{gd}(\omega) = \frac{d}{d\omega} (\tan^{-1}[u]) = \frac{1}{1 + u^2} \frac{du}{d\omega} , \tag{7.6}$$

where $u = x/y$, and

$$x = \omega \sum_{k \text{ odd}}^N (-1)^{(k-1)/2} a_k \omega^{k-1} ,$$

$$y = \sum_{k \text{ even}}^N (-1)^{k/2} a_k \omega^k .$$

From (7.6),

$$t_{gd}(\omega) = \frac{1}{x^2 + y^2} \left[y \frac{dx}{d\omega} - x \frac{dy}{d\omega} \right] , \quad (7.7)$$

where

$$\frac{dx}{d\omega} = \sum_{k \text{ odd}}^N (-1)^{(k-1)/2} k a_k \omega^{k-1} ,$$

$$\frac{dy}{d\omega} = \sum_{k \text{ even}}^N (-1)^{k/2} k a_k \omega^{k-1} .$$

It can be shown that

$$x^2 + y^2 = 1 + (a_1^2 - 2a_2)\omega^2 + (a_2^2 - 2a_1a_3 + 2a_4)\omega^4$$

$$+ (a_3^2 + 2a_1a_5 - 2a_2a_4 - 2a_6)\omega^6 + \dots , \quad (7.8)$$

and that

$$y \frac{dx}{d\omega} - x \frac{dy}{d\omega} = a_1 + (a_1a_2 - 3a_3)\omega^2 + (5a_5 - 3a_1a_4 + a_2a_3)\omega^4$$

$$+ (5a_1a_6 + a_3a_4 - 3a_2a_5 - 7a_7)\omega^6 + \dots . \quad (7.9)$$

By applying (7.8) and (7.9) to (7.7) it is clear that

$$t_{gd}(0) = a_1 .$$

Without loss of generality, let $a_1 = 1$, i.e., the delay at DC is normalized to unity.

It is desired to make the group delay maximally flat for a given order. One way to accomplish this is by equating the coefficients of (7.8) and (7.9) that multiply the same powers of ω . That is, let

$$a_2 - 3a_3 = 1 - 2a_2 ,$$

$$5a_5 - 3a_4 + a_2a_3 = a_2^2 - 2a_3 + 2a_4 ,$$

$$5a_6 + a_3a_4 - 3a_2a_5 - 7a_7 = a_3^2 + 2a_5 - 2a_2a_4 - 2a_6 ,$$

$$\vdots$$

$$\vdots$$

$$\vdots \quad (7.10)$$

From (7.10) it follows that

$$\begin{aligned}
 a_2 - a_3 &= 1/3 \quad , \\
 3(a_5 - a_4) + a_3 &= 1/15 \quad , \\
 15(a_6 - a_7) + a_4 - 6a_5 &= 1/105 \quad , \\
 &\vdots \\
 &\vdots \\
 &\vdots
 \end{aligned}
 \tag{7.11}$$

Example 7.1

If $N = 1$, then $a_k = 0$ for all $k \geq 2$. Then

$$a_0 = 1, \quad a_1 = 1 .$$

Therefore,

$$H(s) = \frac{1}{s + 1} . \tag{7.12}$$

□

Example 7.2

If $N = 2$, then $a_k = 0$ for all $k \geq 3$. Then, from (7.11), it follows that

$$a_0 = 1, \quad a_1 = 1, \quad a_2 = 1/3 .$$

Therefore,

$$H(s) = \frac{3}{s^2 + 3s + 3} . \tag{7.13}$$

□

Example 7.3

If $N = 3$, then $a_k = 0$ for all $k \geq 4$. Then, from (7.11), it follows that

$$a_0 = 1, \quad a_1 = 1, \quad a_2 = 2/5, \quad a_3 = 1/15 .$$

Therefore,

$$H(s) = \frac{15}{s^3 + 6s^2 + 15s + 15} . \tag{7.14}$$

□

Example 7.4

If $N = 4$, then $a_k = 0$ for all $k \geq 5$. Then, from (7.11), it follows that

$$a_0 = 1, \quad a_1 = 1, \quad a_2 = 3/7, \quad a_3 = 2/21, \quad a_4 = 1/105 \quad .$$

It follows that

$$H(s) = \frac{105}{s^4 + 10s^3 + 45s^2 + 105s + 105} \quad . \quad (7.15)$$

□

Example 7.5

If $N = 5$, then $a_k = 0$ for all $k \geq 6$. Then, from (7.11), it follows that

$$a_0 = 1, \quad a_1 = 1, \quad a_2 = 4/9, \quad a_3 = 1/9, \quad a_4 = 1/63, \quad a_5 = 1/945 \quad .$$

It follows that

$$H(s) = \frac{945}{s^5 + 15s^4 + 105s^3 + 420s^2 + 945s + 945} \quad . \quad (7.16)$$

□

Therefore, (7.12), (7.13), (7.14), (7.15), and (7.16) is, respectively, the transfer function of a 1st-order, 2nd-order, 3rd-order, 4th-order, and 5th-order Bessel filter with a normalized delay of 1 second. Note that in *Example 7.1*, $a_1/a_0 = 1$, and in *Example 7.2*, $a_2/a_1 = 1/3$, and in *Example 7.3*, $a_2/a_1 = 4/15$ and $a_3/a_2 = 1/4$, and in *Example 7.4*, $a_2/a_1 = 3/7$, $a_3/a_2 = 2/9$, $a_4/a_3 = 1/10$, and in *Example 7.5*, $a_2/a_1 = 4/9$, $a_3/a_2 = 1/4$, $a_4/a_3 = 1/7$, and $a_5/a_4 = 1/15$. It can be observed from the above examples that

$$\frac{a_{k+1}}{a_k} = \frac{2(N-k)}{(2N-k)(k+1)} \quad , \quad (7.17)$$

where N is the filter order. Therefore, it follows from (7.17) that a Bessel filter with a normalized delay of unity can readily be designed using the following formula.

Bessel filter design formula:

$$a_{k+1} = \frac{2(N-k)}{(2N-k)(k+1)} a_k, \quad a_0 = 1, \quad k = 0, 1, \dots, N-1 \quad , \quad (7.18)$$

It can readily be seen that (7.18) yields the same filter coefficient values as obtained in the above examples.

Note that $H(s)$, as indicated in (7.3), may be expressed as follows:

$$H(s) = \frac{1}{1 + s + \frac{a_2}{a_1} \left\{ s^2 + \frac{a_3}{a_2} \left[s^3 + \frac{a_4}{a_3} \left(s^4 + \dots \right) \right] \right\}} \quad (7.19)$$

Applying (7.17) to (7.19):

$$H(s) = \frac{1}{1 + s + \frac{N+1}{2N-1} \left\{ s^2 + \frac{2(N-2)}{3(2N-2)} \left[s^3 + \frac{2(N-3)}{4(2N-3)} \left(s^4 + \dots \right) \right] \right\}} \quad (7.20)$$

The denominator of (7.20) is recognized as being a Bessel polynomial (Krall and Frink, 1949; Storch, 1954; Van Valkenburg, 1960). Therefore, these filters are called Bessel filters.

Note that, from (7.17),

$$\lim_{N \rightarrow \infty} \frac{a_{k+1}}{a_k} = \frac{1}{k + 1} .$$

Therefore, it follows that

$$\lim_{N \rightarrow \infty} H(s) = \frac{1}{1 + s + \frac{1}{2} s^2 + \frac{1}{6} s^3 + \frac{1}{24} s^4 + \dots} \quad (7.21)$$

The denominator of (7.21) is recognized as being the Maclaurin series expansion of e^s . Therefore,

$$\lim_{N \rightarrow \infty} H(s) = e^{-s} .$$

It follows that

$$\lim_{N \rightarrow \infty} H(j\omega) = e^{-j\omega} \quad (7.22)$$

Note that (7.22) is precisely equal to (7.1) with $K_p = t_d = 1$. Therefore, as the order is made higher and higher, a Bessel filter more and more closely approximates an ideal delay circuit.

The delay characteristics of Bessel filters with practical values for the order are shown in **Figure 7.1**. As can be readily seen, as the order becomes larger the frequency range over which the delay is relatively constant becomes larger. The corresponding magnitude frequency response is shown in **Figure 7.2**.

A delay circuit, assuming that the order is known, is readily designed as follows. Equation (7.18) may be used to obtain the required transfer function

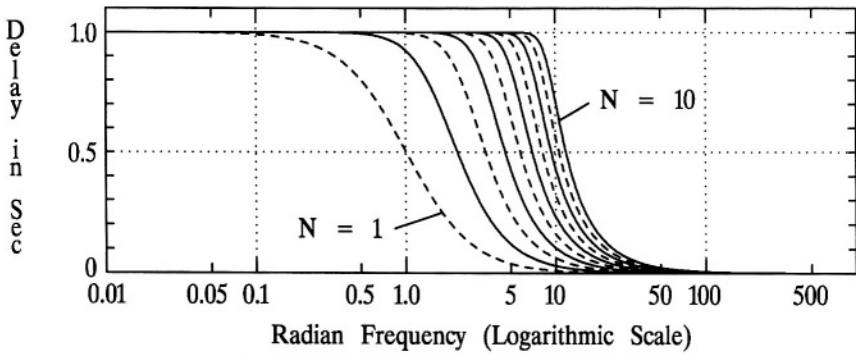


Figure 7.1 The Bessel filter normalized group delay response for values of N from 1 through 10.

coefficients for a normalized delay of 1 second. The required time-scaling parameter is simply equal to the desired actual time delay, which is inversely related to the equivalent frequency-scaling parameter. If the actual time delay desired is t_d seconds, then a frequency scaling parameter of $1/t_d$ is required.

Example 7.6

Suppose a 4th-order Bessel filter is desired with a time delay of 1 ms. The transfer function with a normalized delay of 1 second is given in **Example 7.4**. The

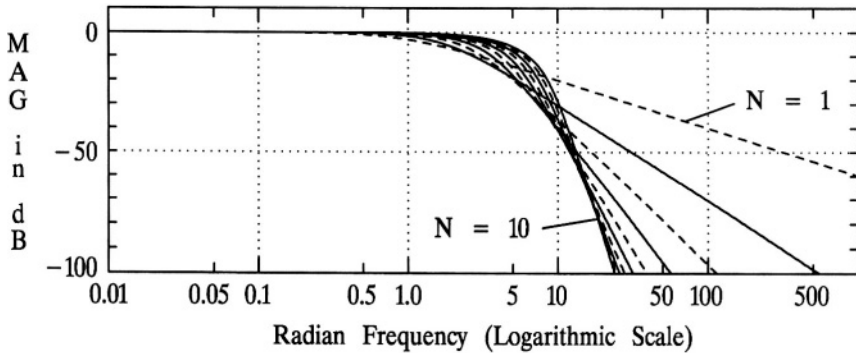


Figure 7.2 The Bessel filter magnitude frequency response with normalized group delay, for values of N from 1 through 10.

required frequency scaling factor is 1000 rad/s . Replacing each s in (7.15) with $s/1000$ results in the following transfer function:

$$H(s) = \frac{1.05 \times 10^{14}}{s^4 + 10^4 s^3 + 4.5 \times 10^7 s^2 + 1.05 \times 10^{11} s + 1.05 \times 10^{14}} \quad (7.23)$$

Therefore, (7.23) is the transfer function of a 4th-order Bessel filter with a delay of 1 ms . □

Unlike filters presented in earlier chapters, Bessel filters do not have a convenient closed-form expression for the magnitude frequency response. Therefore, such expressions as the 3 dB corner frequency, etc., are not readily available. However, they are easy to obtain with an algorithm. On the accompanying disk is the MATLAB function *BESSELDE*, which will design a Bessel filter for any given order, passband corner frequency ω_p , and passband attenuation limit A_p .

Using the MATLAB function *BESSELDE*, the magnitude frequency response for Bessel filters normalized for a 3 dB corner frequency of unity, for several values of N have been computed. The results are plotted in Figure 7.3. Also see Figure 7.4 for detailed plots of the magnitude frequency response across the passband.

7.3 FILTER SELECTIVITY AND SHAPING FACTOR

Since a closed-form convenient expression for the magnitude frequency response for a Bessel filter of arbitrary order is not readily available, neither are expressions for *Filter Selectivity* and *Shaping Factor*. However, these can be readily

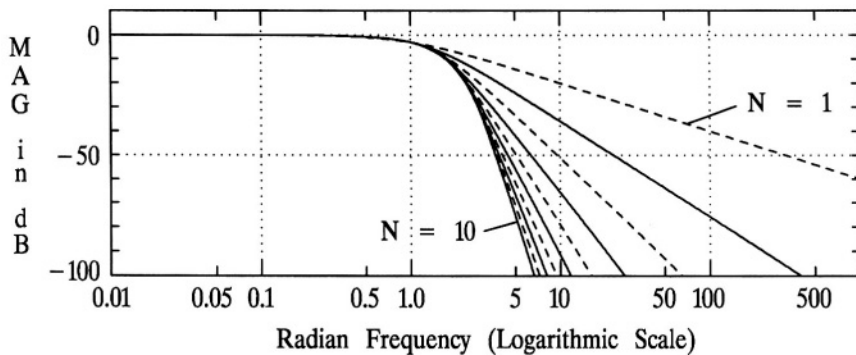


Figure 7.3 The Bessel filter magnitude frequency response with normalized $\omega_c = 1$, for values of N from 1 through 10.

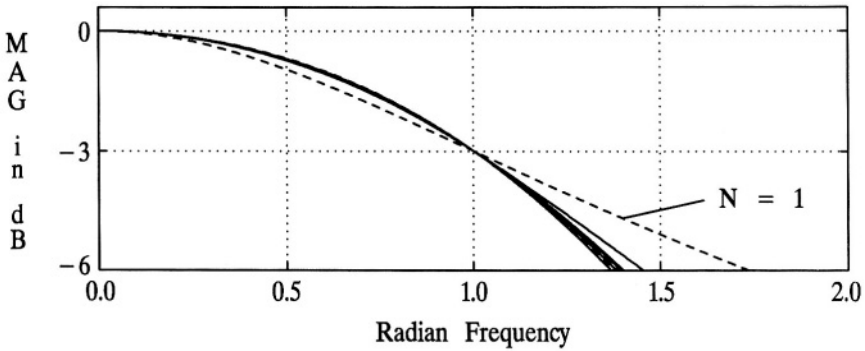


Figure 7.4 Passband details of the Bessel filter magnitude frequency response with normalized $\omega_c = 1$, for values of N from 1 through 10.

computed. The MATLAB functions *BESSELS* and *BESSELSF* on the accompanying disk may be used to compute these filter measures of performance.

Example 7.7

Suppose $a = 3$ dB, $b = 80$ dB, and $\omega_c = 1$. From *BESSELS*, for $N = 1, 2, \dots, 10$, F_s may be computed to be 0.35, 0.49, 0.54, 0.54, 0.52, 0.51, 0.51, 0.51, 0.50 and 0.50 respectively. From *BESSELSF*, for N from 2 through 10, S_3^{80} may be computed to be 127.19, 30.26, 15.13, 10.21, 8.00, 6.79, 6.08, 5.58 and 5.29 respectively. \square

7.4 DETERMINATION OF ORDER

Since a closed-form convenient expression for the magnitude frequency response for a Bessel filter of arbitrary order is not readily available, neither is a convenient expression for the minimum required order to meet given magnitude frequency response specifications as indicated in **Figure 2.15**. However, note that the **Shaping Factor** is inversely related to filter order and that parameters a and b could be the design parameters A_p and A_s . With these observations, and making use of the MATLAB function *BESSELSF*, the MATLAB function *BESSELSOR* has been written, and is on the accompanying disk, that will compute the minimum required order to meet given design specifications.

Example 7.8

Suppose the following specifications are given: $f_p = 3,000 \text{ Hz}$, $f_s = 16,000 \text{ Hz}$, $A_p = 2 \text{ dB}$, and $A_s = 60 \text{ dB}$. From the MATLAB function *BESSELOR*, $N = 9$. \square

7.5 POLE LOCATIONS

Previous filters presented, Butterworth, Chebyshev Type I, Chebyshev Type II, and elliptic, all have closed-form convenient expressions for the magnitude frequency response, from which expressions are found for the poles and zeros. The design of Bessel filters has been presented above, either as a delay network with some given desired time delay, or as a lowpass filter with given magnitude frequency response specifications. As mentioned above, no convenient closed-form expression exists for the magnitude frequency response of a Bessel filter.

The design of a lowpass Bessel filter using the MATLAB function *BESSELDE* yields the poles of the transfer function directly. The design of a Bessel time-delay filter using (7.18) followed by frequency scaling, as in *Example 7.6*, yields the transfer function expressed as a constant over a polynomial in s . The poles may be found by determining the roots of the transfer function denominator polynomial. This may be accomplished using the MATLAB function *ROOTS*. All zeros, of course, are at infinity.

7.6 PHASE RESPONSE, PHASE DELAY, AND GROUP DELAY

A Bessel filter, as seen above, is designed for a maximally-flat group delay response. The group delay response, normalized for a unit delay, is shown in *Figure 7.1* above. The magnitude frequency response, normalized for a 3 dB corner frequency of unity, is also shown above in *Figures 7.3* and *7.4*.

The phase response of a Bessel filter, with a normalized $\omega_c = 1$, and several values of N , is shown in *Figure 7.5*. Taking the initial phase slope as a linear-phase reference, deviations from linear phase, for a normalized $\omega_c = 1$, and for several values of N , are shown in *Figure 7.6*. In the figure, solid lines are for even orders, and dashed lines are for odd orders.

The *phase delay*, $t_{pd}(\omega)$, for a filter is defined in (2.80), which is repeated here for convenience:

$$t_{pd}(\omega) = -\frac{\angle H(j\omega)}{\omega} . \quad (7.24)$$

The *group delay* for a filter, $t_{gd}(\omega)$, is defined by (2.81) and is repeated here for convenience:

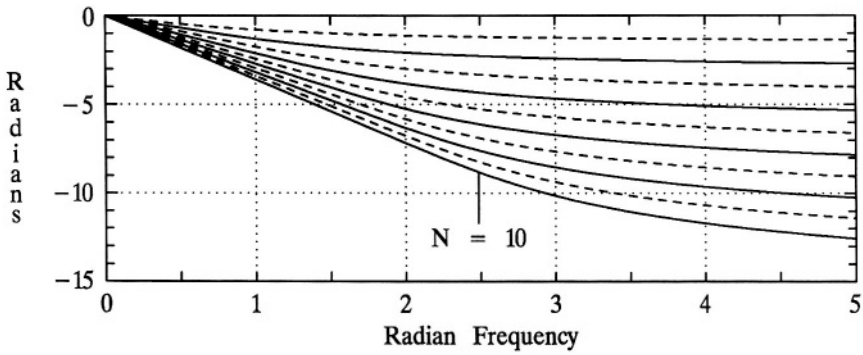


Figure 7.5 A plot of the phase response for a Bessel filter with normalized $\omega_c = 1$, for values of N from 1 through 10.

$$t_{gd}(\omega) = - \frac{d \angle H(j\omega)}{d\omega} \quad (7.25)$$

The phase delay of a Bessel filter, with a normalized $\omega_c = 1$, for several values of N , is shown in **Figure 7.7**. The group delay is shown in **Figure 7.8**. In both figures, solid lines are for even orders, and dashed lines are for odd orders.

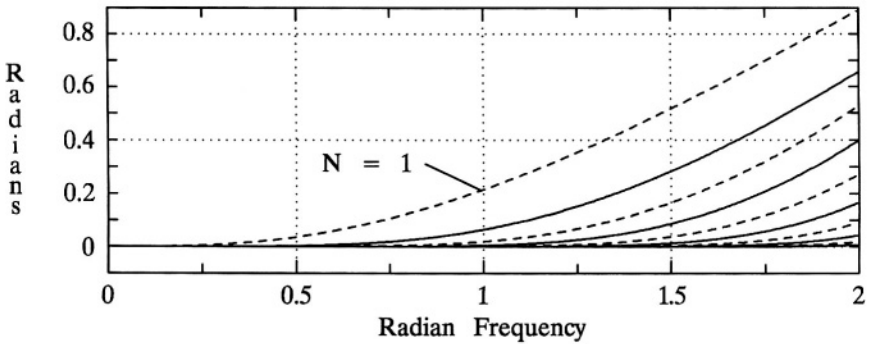


Figure 7.6 Phase deviation from linear for a Bessel filter with normalized $\omega_c = 1$, for values of N from 1 through 10.

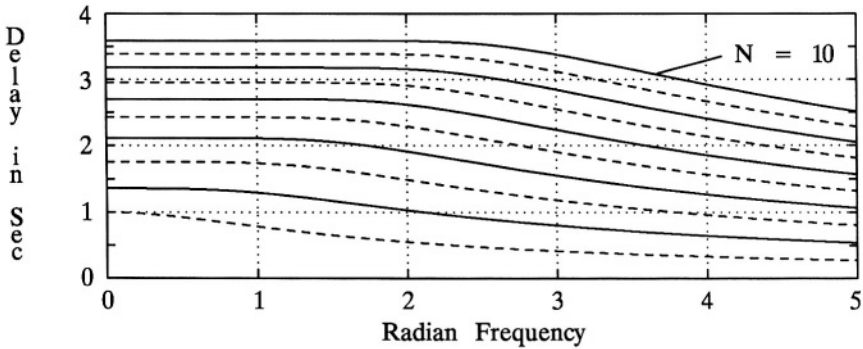


Figure 7.7 A plot of the phase delay for a Bessel filter with normalized $\omega_c = 1$, for values of N from 1 through 10.

Note that the phase delay values at *DC* are essentially identical to the group delay values at *DC*. The comparison between phase delay and group delay for $\omega > 0$ depends on the order, but note that they are, in general, very comparable. The group delay, as can be seen in the figures, begins to decrease from its' low-frequency value at a lower frequency than does the phase delay, but both are flat for a wide frequency range. The fact that the group delay begins to differ from the phase delay for higher frequencies, is an indication of the phase beginning to deviate from linear: this is

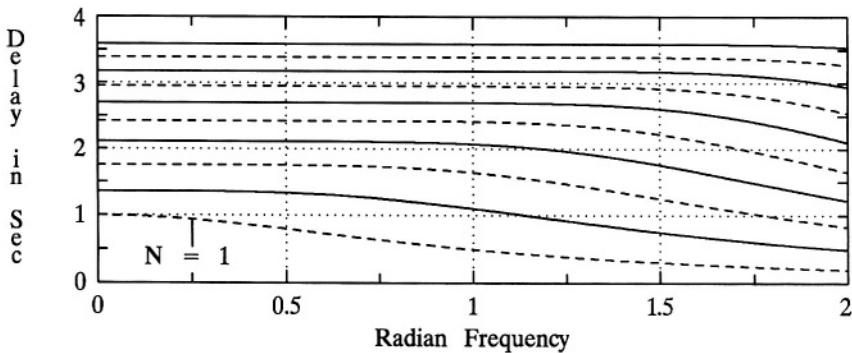


Figure 7.8 A plot of the group delay for a Bessel filter with normalized $\omega_c = 1$, for values of N from 1 through 10.

observable in **Figures 7.5** and **7.6** and illustrates how sensitive this phenomenon is to phase nonlinearity. For example, for $N = 10$, it is observed by comparing **Figure 7.8** with **7.7**, that the group delay differs from the phase delay slightly at $\omega = 2$. It is difficult in **Figure 7.5** to observe any deviation from linearity at $\omega = 2$, for $N = 10$, however that deviation is detectable in **Figure 7.6**.

7.7 TIME-DOMAIN RESPONSE

The unit impulse response of a Bessel filter, with a normalized $\omega_c = 1$, and for several values of N , is shown in **Figure 7.9**. The unit step response of a Bessel filter, with normalized $\omega_c = 1$, and for several values of N , is shown in **Figure 7.10**.

7.8 COMPARISON WITH PRECEDING FILTERS

As noted in earlier chapters, there are several ways in which filters can be compared. In terms of the phase response, phase delay, group delay, impulse response, and step response, Bessel filters are clearly superior to all of the other classical filters presented in earlier chapters. Therefore, when any of these responses are of high priority, a Bessel filter should be carefully considered. It should also be noted that all of these superior response characteristics of Bessel filters are closely related. The linear phase response across the passband of a Bessel filter is what results in excellent phase delay and group delay characteristics. It also contributes to the excellent impulse response and step response characteristics.

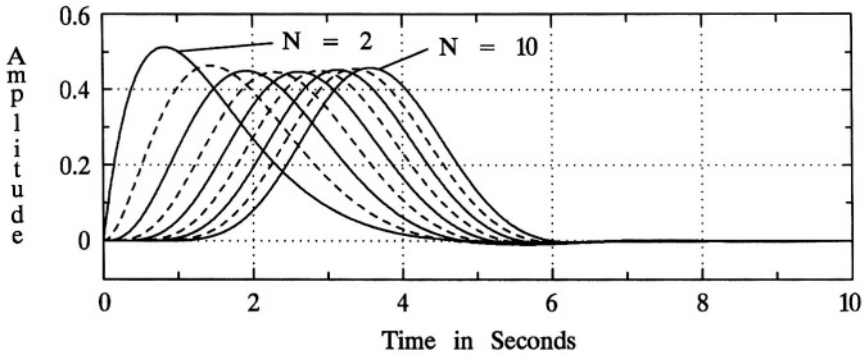


Figure 7.9 A plot of the unit impulse response for a Bessel filter with normalized $\omega_c = 1$, for values of N from 2 through 10.

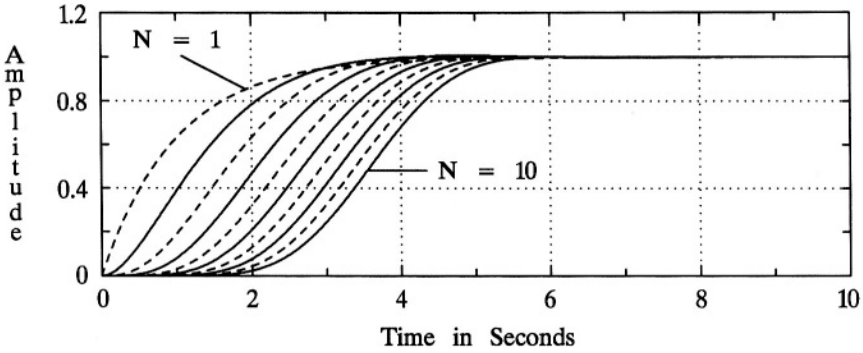


Figure 7.10 A plot of the unit step response for a Bessel filter with normalized $\omega_c = 1$, for values of N from 1 through 10.

However, in terms of the magnitude frequency response characteristics Bessel filters are less desirable than any of the previously presented classical filters in this book. This is so in terms of the minimum required order to meet given magnitude response specifications, and also in terms of *Filter Selectivity*, i.e., the sharpness of the passband edge. As can be seen in *Example 7.7*, as well as directly in **Figure 7.4**, *Filter Selectivity*, the negative of the slope of the magnitude response at $\omega = \omega_c$, remains relatively constant for a Bessel filter for orders of two and greater. This differs from any of the previously studied filters. The filter *Shaping Factor* of a Bessel filter is also not as good, for the same order, as any of the previously studied filters. If *Example 7.7* is compared with *Examples 3.1, 4.2, 5.2* and *6.6*, it is observed that the *Shaping Factor* of a tenth-order Bessel filter, with $a = 3\text{ dB}$ and $b = 80\text{ dB}$, can be met or exceeded by a sixth-order Butterworth filter, a fifth-order Chebyshev Type I or Chebyshev Type II filter, or by a fourth-order elliptic filter. Therefore, for given magnitude response specifications, a higher order is required with a Bessel filter: when the superior characteristics, phase response, etc., of a Bessel filter are of relatively low priority, a Bessel filter is probably not the best choice among filter types.

It is interesting to note how Bessel filters and elliptic filters are somewhat at opposite extremes among filter types. Elliptic filters require the lowest order among all classical filter types to meet given magnitude response specifications. In terms of *Filter Selectivity* and *Shaping Factor*, elliptic filters are clearly superior to all other classical filter types. However, the phase response and corresponding phase delay and group delay, as well as time domain responses, for elliptic filters, are the worst of all classical filter types. Whereas Bessel filters require the largest order among all

classical filter types presented so far to meet given magnitude response specifications, and have the worst *Filter Selectivity* and *Shaping Factor*, their phase response and corresponding phase delay and group delay, and the time domain responses as well, are by far vastly superior to all other filter types considered so far. It is in this sense that Butterworth and Chebyshev filters are compromises between the two extremes of Bessel and elliptic filters. The performance of Butterworth filters is closer to that of Bessel filters than are Chebyshev filters: the phase response is more linear but the order required to meet given magnitude response specifications is greater. Conversely, the performance of Chebyshev filters, Types I and II, fall between Butterworth and elliptic filters: Type II being closer to Butterworth and Type I closer to elliptic.

7.9 CHAPTER 7 PROBLEMS

- 7.1 Similar to *Examples 7.1* through *7.5*, determine, using (7.18), the transfer function of a sixth-order Bessel filter with a normalized delay of unity.
- 7.2 Repeat **Problem 7.1** for a seventh-order Bessel filter.
- 7.3 Repeat **Problem 7.1** for an eighth-order Bessel filter.
- 7.4 Similar to *Example 7.6*, determine the transfer function of a third-order Bessel filter with a time delay of 1 *ms*.
- 7.5 Determine the transfer function (the order is to be determined) of a Bessel filter with the following specifications: time delay at low frequencies of 2 *ms*, and a minimum time delay of 1.8 *ms* at 400 *Hz*.
- 7.6 Determine the transfer function (the order is to be determined) of a Bessel filter with the following specifications: time delay at low frequencies of 10 μs , and a minimum time delay of 8 μs at 80 *kHz*.
- 7.7 Using the MATLAB function *BESSELDE*, determine the transfer function of a second-order Bessel filter with $A_p = 2$ *dB* and $\omega_p = 1000$ *rad/s*.
- 7.8 Using the MATLAB function *BESSELDE*, determine the transfer function of a third-order Bessel filter with $A_p = 1$ *dB* and $\omega_p = 2000$ *rad/s*.
- 7.9 Using the MATLAB function *BESSELDE*, determine the transfer function of a fourth-order Bessel filter with $A_p = 2$ *dB* and $f_p = 2000$ *Hz*.

- 7.10** Using MATLAB functions *BESSELFS* and *BESSELSF*, determine the *Filter Selectivity* and *Shaping Factor* for the Bessel filter of **Problem 7.7**.
- 7.11** Using MATLAB functions *BESSELFS* and *BESSELSF*, determine the *Filter Selectivity* and *Shaping Factor* for the Bessel filter of **Problem 7.8**.
- 7.12** Using MATLAB functions *BESSELFS* and *BESSELSF*, determine the *Filter Selectivity* and *Shaping Factor* for the Bessel filter of **Problem 7.9**.
- 7.13** Similar to *Example 7.8*, using the MATLAB function *BESSELOR*, determine the minimum required order to meet the following set of specifications:
 $\omega_p = 1,000$, $\omega_s = 10,000$, $A_p = 1.5$ dB, $A_s = 70$ dB .
- 7.14** Similar to *Example 7.8*, using the MATLAB function *BESSELOR*, determine the minimum required order to meet the following set of specifications:
 $f_p = 3,500$ Hz, $f_s = 20,000$ Hz, $A_p = 2.5$ dB, $A_s = 60$ dB .
- 7.15** Using the MATLAB function *BESSELDE*, determine the transfer function for the filter of **Problem 7.13**. Express $H(s)$ as a constant over a polynomial in s .
- 7.16** Using the MATLAB function *BESSELDE*, determine the transfer function for the filter of **Problem 7.14**. Express $H(s)$ as a constant over a polynomial in s .
- 7.17** By referring to **Figure 7.6**, and making use of frequency scaling, what would be the phase deviation from linear, in degrees, for a 4th-order Bessel filter with an f_c of 1000 Hz, at $f = 2000$ Hz? By also making use of **Figure 7.7**, what percentage error in the phase response does this phase deviation represent?
- 7.18** By referring to **Figure 7.6**, and making use of frequency scaling, what would be the phase deviation from linear, in degrees, for a 6th-order Bessel filter with an f_c of 1500 Hz, at $f = 3000$ Hz? By also making use of **Figure 7.7**, what percentage error in the phase response does this phase deviation represent?
- 7.19** Refer to the tenth-order impulse response plotted in **Figure 7.9**. Suppose that a particular measure for effective time duration yields 4 s for the tenth-order impulse response. Also suppose that the peak value of the impulse response is 0.46. Determine the effective time duration and the peak value of the tenth-order impulse response for the following values of 3 dB cutoff frequency:

- (a) $\omega_c = 1000 \text{ rad/s}$
- (b) $\omega_c = 10,000 \text{ rad/s}$
- (c) $f_c = 30 \text{ kHz}$.

7.20 Using the MATLAB functions *BESSELDE*, *impulse* and *step*:

- (a) Determine the transfer function in polynomial form, and also factored to indicate the poles, of a Bessel filter with $\omega_p = 1$, $A_p = 2 \text{ dB}$, and $N = 6$.
- (b) Determine the impulse response and the step response for the filter of part (a).
- (c) By multiplying the pole vector found in part (a) by $2\pi 1000$ determine the transfer function of a Bessel filter with $f_p = 1000 \text{ Hz}$, $A_p = 2 \text{ dB}$, and $N = 6$.
- (d) Determine and plot the magnitude frequency response of the filter of part (c) by using the MATLAB function *freqs*. Use a vertical scale in *dB* and a linear horizontal scale from 0 to 5000 *Hz*. Also determine and plot the phase response over this same frequency range. Use the MATLAB function *unwrap* to display the smooth phase response rather than the principle phase.
- (e) By appropriately scaling the impulse response and the step response of part (b), determine and plot the impulse response and the step response of the filter of part (c). That is, the time axis for the step response needs to be scaled by $1/(2\pi 1000)$, and the unit impulse response needs the same time-axis scaling and requires an amplitude scaling of $2\pi 1000$.
- (f) Determine and plot the phase delay of the filter of part (c). Note that this is easily obtained from the phase response of part (d).
- (g) Determine and plot the group delay of the filter of part (c). Note that this also is easily obtained from the phase response of part (d): $t_{gd}(n) \cong -[\phi(n) - \phi(n-1)]/S_s$, where $\phi(n)$ is the phase in *radians* at step n , and S_s is the step size in *rad/s*.

CHAPTER 8

OTHER FILTERS

In this chapter, several other filters are briefly presented, selected from the many that have been proposed. These filters have not enjoyed the popularity of those presented in earlier chapters, but are presented here for two reasons: (1) they are representative of other filter types that have been proposed, and indicate fruitful research in filter development, and (2) they are interesting and useful filter design methods. The filter presentations in this chapter are not as detailed as in earlier chapters, but do, hopefully, include relevant and interesting material that indicate important characteristics of the various methods presented.

8.1 TRANSITIONAL FILTERS

A transitional filter is a filter that is explicitly designed such that it is, in some sense, in between two existing filter designs. A number of ways have been proposed to form a transitional filter (Lindquist, 1977). One particular method will be presented here, that operates directly on the filter poles (Aiello and Angelo, 1974; Peless and Murakami, 1957).

Suppose $p_{D1}(k)$, $k = 1, 2, \dots, N$, are the poles for filter design 1, and $p_{D2}(k)$, $k = 1, 2, \dots, N$, are the poles for filter design 2, and the poles are arranged such that $p_{D2}(k)$ is close to $p_{D1}(k)$ for each k . Let the transitional filter poles be expressed as follows:

$$p_{TR}(k) = p_{D2}^m(k) p_{D1}^{1-m}(k) \quad , \quad (8.1)$$

where $0 \leq m \leq 1$. Note that when $m = 0$ the transitional filter is identical to design 1, when $m = 1$ the transitional filter is identical to design 2, and otherwise the transitional filter is somewhere in between the two. The term $p_{D2}^m(k)$ in (8.1) may be expressed as follows:

$$p_{D2}^m(k) = \left\{ |p_{D2}(k)| e^{j\angle p_{D2}(k)} \right\}^m = |p_{D2}(k)|^m e^{jm\angle p_{D2}(k)} \quad , \quad (8.2)$$

and also

$$p_{D1}^{1-m}(k) = |p_{D1}(k)|^{1-m} e^{j(1-m)\angle p_{D1}(k)} \quad . \quad (8.3)$$

Combining (8.1) through (8.3):

$$|p_{TR}(k)| = |p_{D2}(k)|^m |p_{D1}(k)|^{1-m} \quad (8.4)$$

and

$$\angle p_{TR}(k) = \angle p_{D1}(k) - m \{ \angle p_{D1}(k) - \angle p_{D2}(k) \} \quad (8.5)$$

Except for Bessel filters, Butterworth filters have the best group delay characteristics of all the filters presented in previous chapters. However, while Butterworth filters have much better magnitude frequency response characteristics than do Bessel filters, the group delay characteristics are clearly inferior. Therefore, as an example of transitional filters, (8.4) and (8.5) are applied to Butterworth and Bessel filters. Specifically, let design 1 be a 10th-order Butterworth filter with a normalized ω_c of unity, let design 2 be a 10th-order Bessel filter also with a normalized ω_c of unity, and let $m = 0.4$.

In Figure 8.1 are shown the magnitude frequency responses for the Butterworth, Bessel, and transitional filters. The transitional magnitude frequency response is clearly in between that of the Butterworth and the Bessel. In Figure 8.2 are shown the passband details of the magnitude frequency response for the three filters. In Figure 8.3 the phase responses are shown. In Figure 8.4 the phase delay responses are shown, and in Figure 8.5 the group delay responses. As can be seen, the phase delay and the group delay of the transitional filter is significantly improved over the Butterworth, however at the expense of a less desirable magnitude frequency response. In Figure 8.6 the unit impulse responses are shown, and in Figure 8.7 the unit step responses. Note that the unit impulse response and the unit step response of the transitional filter is significantly improved over the Butterworth (less ringing).

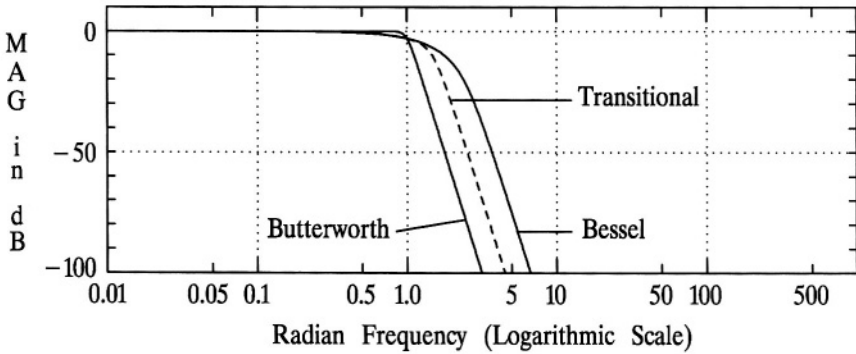


Figure 8.1 The magnitude frequency response of a 10th-order Butterworth filter, a 10th-order Bessel filter, and a 10th-order transitional filter with $m = 0.4$.

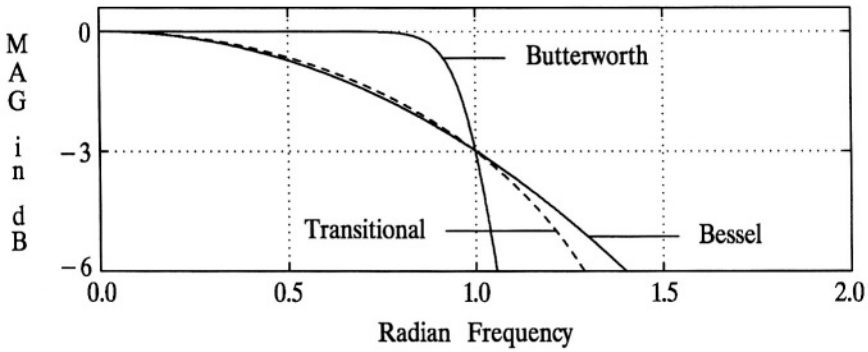


Figure 8.2 Passband details of that shown in **Figure 8.1**: a 10th-order Butterworth filter, a 10th-order Bessel filter, and a 10th-order transitional filter with $m = 0.4$.

8.2 GAUSSIAN FILTERS

The ideal Gaussian filter has a magnitude-squared frequency response defined as follows:

$$|H(j\omega)|^2 = e^{-\gamma\omega^2} \quad , \quad (8.6)$$

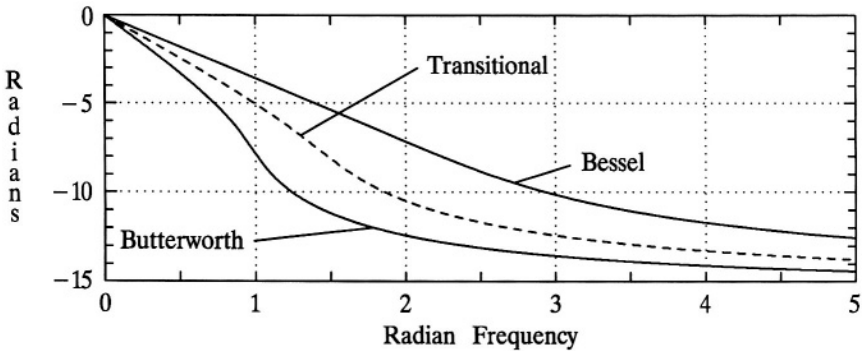


Figure 8.3 The phase response of a 10th-order Butterworth filter, a 10th-order Bessel filter, and a 10th-order transitional filter with $m = 0.4$.

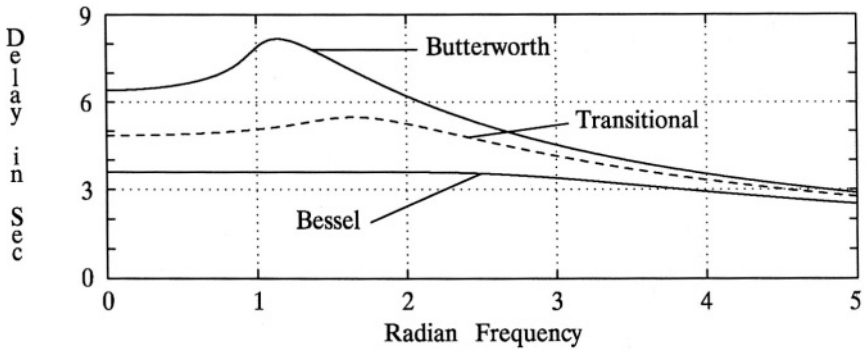


Figure 8.4 The phase delay response of a 10th-order Butterworth filter, a 10th-order Bessel filter, and a 10th-order transitional filter with $m = 0.4$.

where $\gamma = \ln(2)/\omega_c^2$. An approximation to $e^{\gamma\omega^2}$, using a Maclaurin series, may be expressed as follows (Dishal, 1959; Lindquist, 1977):

$$e^{\gamma\omega^2} = 1 + \gamma\omega^2 + \frac{\gamma^2}{2!}\omega^4 + \frac{\gamma^3}{3!}\omega^6 + \dots + \frac{\gamma^N}{N!}\omega^{2N} + \dots$$

If the series is truncated at the N -th term, (8.6) may be approximated by the following:

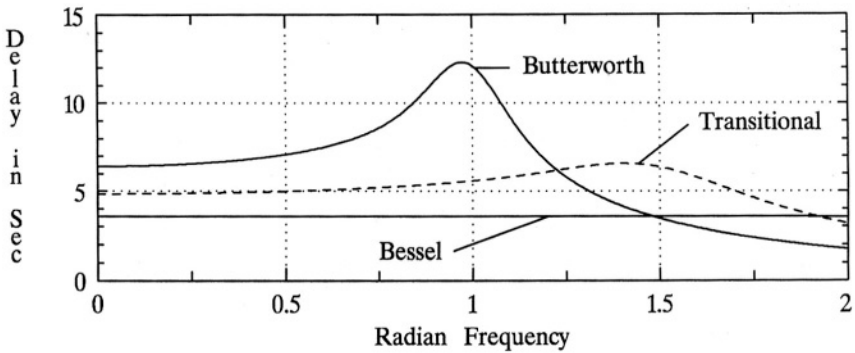


Figure 8.5 The group delay response of a 10th-order Butterworth filter, a 10th-order Bessel filter, and a 10th-order transitional filter with $m = 0.4$.

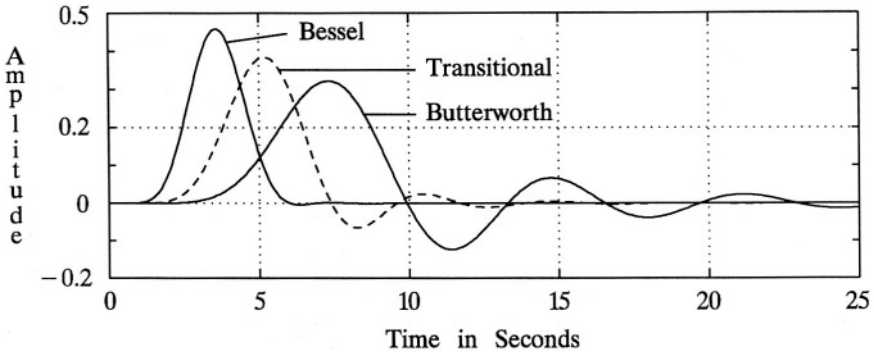


Figure 8.6 The unit impulse response of a 10th-order Butterworth filter, a 10th-order Bessel filter, and a 10th-order transitional filter with $m = 0.4$.

$$|H(j\omega)|^2 = \frac{1}{1 + \sum_{k=1}^N \frac{\gamma^k}{k!} \omega^{2k}} \tag{8.7}$$

Following the procedure given in Section 2.7, $H(s)$ may be found from (8.7)¹.

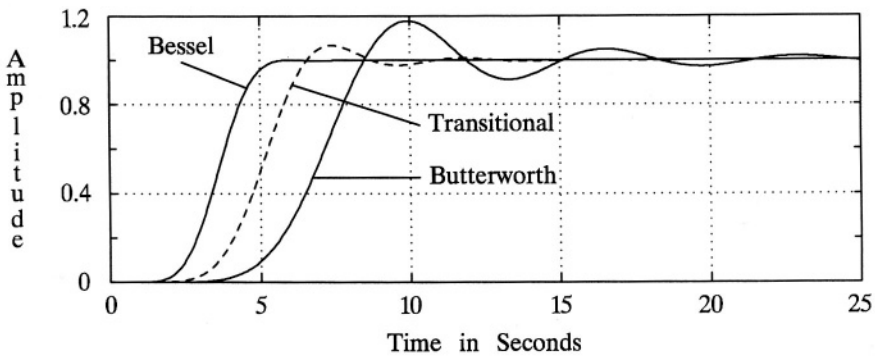


Figure 8.7 The unit step response of a 10th-order Butterworth filter, a 10th-order Bessel filter, and a 10th-order transitional filter with $m = 0.4$.

¹A convenient method for computing the transfer function is by using MATLAB to find the roots of the denominator of $H(s)H(-s)$ and then assigning those with negative real parts to $H(s)$.

See **Figure 8.8** for plots of (8.7) for a normalized ω_c of unity, and several values of N . Also see **Figure 8.9** for detailed plots of (8.7) across the passband. The phase response for a normalized ω_c of unity, and several values of N , is shown in **Figure 8.10**. The phase delay is shown in **Figure 8.11**, and the group delay in **Figure 8.12**. The unit impulse response is shown in **Figure 8.13**, and the unit step response in **Figure 8.14**.

It is noted that the magnitude frequency response, phase response, phase delay, group delay, unit impulse response, and unit step response for the Gaussian filter are all very similar to those of the Bessel filter. Close examination shows that the Bessel filter has slightly better *Shaping Factor*, flatter phase delay, and flatter group delay than that of a Gaussian filter of equal order. However, the Gaussian filter has less time delay, as noted by the unit impulse response peaks occurring sooner than they do for Bessel filters of equal order. This smaller time delay is also observable in the step response plots for the Gaussian filter compared with those for the Bessel filter.

On the accompanying disk is the MATLAB function *GAUSDE*. This function may be used to design a Gaussian lowpass filter for a given order, passband corner frequency ω_p , and passband corner attenuation A_p .

8.3 LEGENDRE FILTERS

Legendre filters, and *associated Legendre filters*, were introduced by Ku and Drubin (1962), and are noted for having a very good *Shaping Factor*, ripple in the passband (but not equiripple), and having less group delay distortion (a flatter response, for the first and second associated Legendre filters) than Chebyshev Type I filters. The magnitude-squared frequency response is defined as follows:

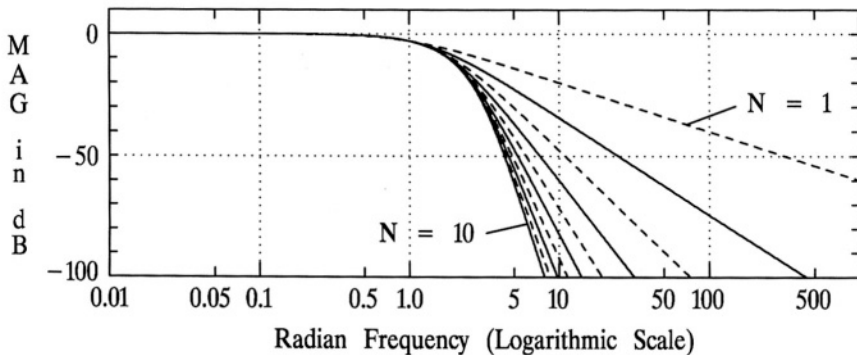


Figure 8.8 The Gaussian magnitude response. Plots of (8.7) for $\omega_c = 1$ and values of N from 1 through 10.

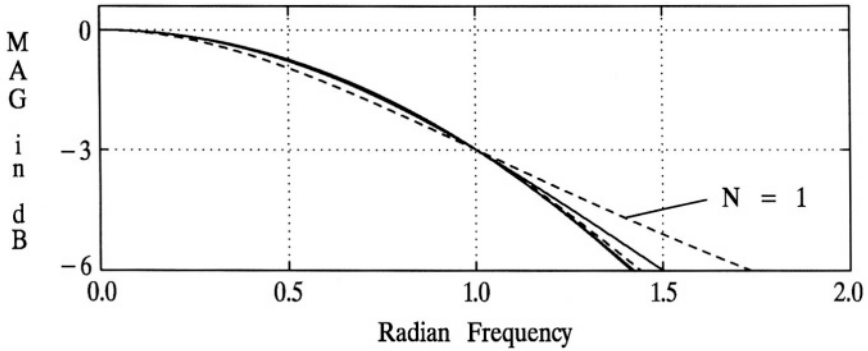


Figure 8.9 Passband details of the Gaussian magnitude response. Plots of (8.7) for $\omega_c = 1$ and values of N from 1 through 10.

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 [P_N^{(m)}(\omega/\omega_p)]^2}, \tag{8.8}$$

where $P_N^{(m)}(\omega/\omega_p)$ is the modified m -th associated Legendre polynomial of order N , (N is also the order of the filter), ω_p is the passband edge frequency, and ϵ is a parameter that controls the attenuation (A_p in dB) at $\omega = \omega_p$. Legendre polynomials

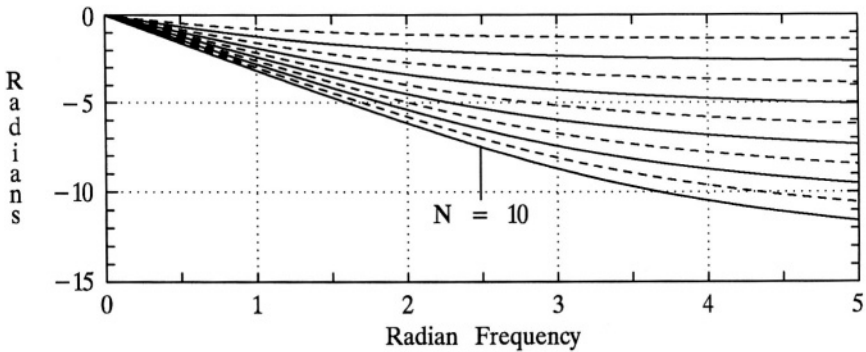


Figure 8.10 The phase response of a Gaussian filter with $\omega_c = 1$, for values of N from 1 through 10.

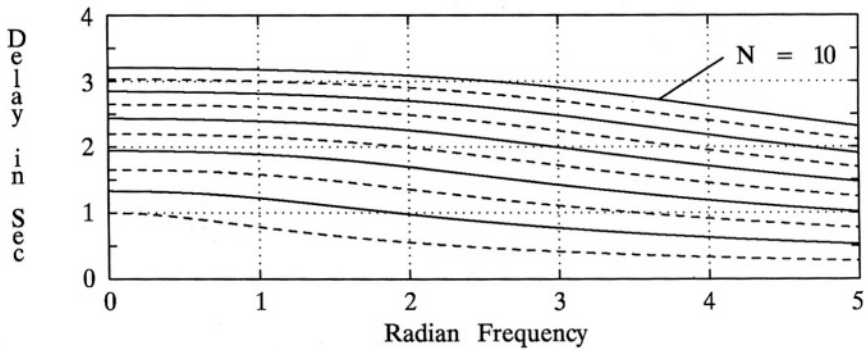


Figure 8.11 The phase delay response of a Gaussian filter with $\omega_c = 1$, for values of N from 1 through 10.

($m = 0$) may be readily obtained from the following recursion (Selby, 1970), where ω_p has been normalized to unity:

$$P_{N+1}(\omega) = \frac{2N+1}{N+1} \omega P_N(\omega) - \frac{N}{N+1} P_{N-1}(\omega) \quad ,$$

where $P_0(\omega) = 1$, and $P_1(\omega) = \omega$. The m -th associated Legendre polynomial of order N is obtained from the Legendre polynomial of order N as follows:

$$P_{N,m}(\omega) = \frac{d^m}{d\omega^m} P_N(\omega) \quad .$$

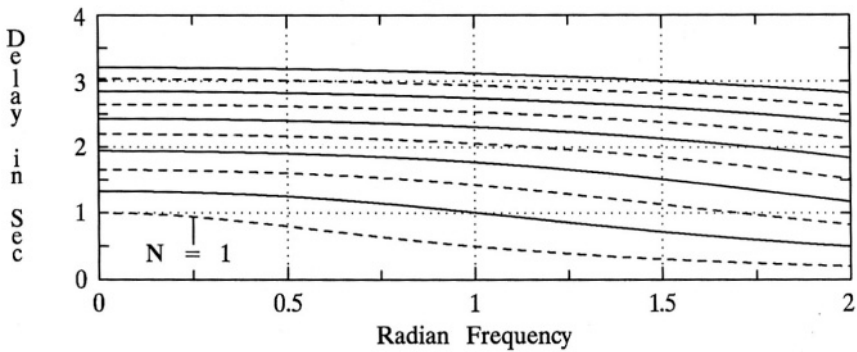


Figure 8.12 The group delay response of a Gaussian filter with $\omega_c = 1$, for values of N from 1 through 10.

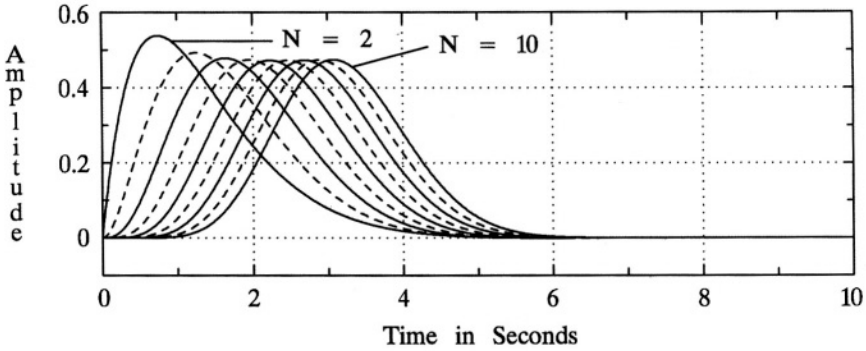


Figure 8.13 The unit impulse response of a Gaussian filter with $\omega_c = 1$, for values of N from 2 through 10.

For example,

$$\begin{aligned}
 P_{3,2}(\omega) &= \frac{d^2}{d\omega^2} P_5(\omega) = \frac{d^2}{d\omega^2} \frac{1}{8} (63\omega^5 - 70\omega^3 + 15\omega) \\
 &= \frac{105}{2} (3\omega^3 - \omega) .
 \end{aligned}$$

It is noted that $-1 \leq P_N(\omega) \leq 1$ for $-1 \leq \omega \leq 1$, and is equal to one at $\omega = 1$ (at $\omega = \omega_p$, in general), but that this is not so for $P_{N,m}(\omega)$ when $m > 0$.

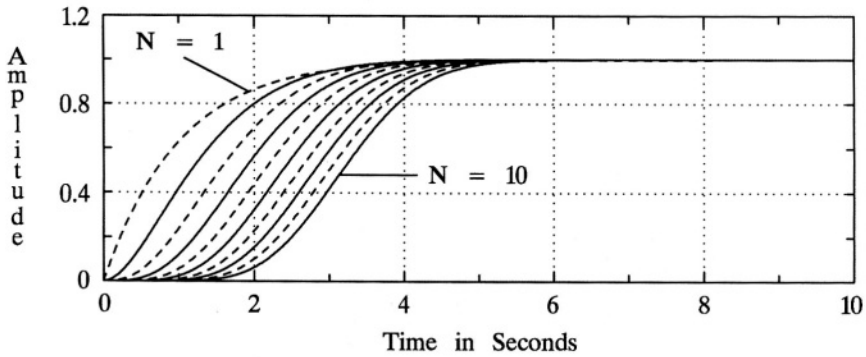


Figure 8.14 The unit step response of a Gaussian filter with $\omega_c = 1$, for values of N from 1 through 10.

Therefore, for consistent use of ϵ in adjusting the desired attenuation at the edge of the passband, it is necessary to adjust the amplitude of $P_{N,m}(\omega)$. This is most conveniently accomplished at $\omega = 1$, where it is desired that $[P_{N,m}(1)]^2 = 1$. For example, as can be seen from above, $P_{3,2}(1) = 105$. The modified second-associated Legendre polynomial of order 3, denoted $P_3^{(2)}(\omega)$, modified so that the function is properly bounded is as follows:

$$P_3^{(2)}(\omega) = \frac{1}{2} (3\omega^3 - \omega) \quad .$$

Therefore, the *modified* m -th associated Legendre polynomial of order N is obtained from the Legendre polynomial of order N as follows:

$$P_N^{(m)}(\omega) = K \frac{d^m}{d\omega^m} P_{N,m}(\omega) \quad ,$$

where K is such that $P_N^{(m)}(1) = 1$.

Under the above conditions, i.e., using Legendre and modified associated Legendre polynomials, the required value for ϵ is the same for Legendre and associated Legendre filters as it is for Chebyshev Type I filters:

$$\epsilon = \sqrt{10^{A_p/10} - 1} \quad , \quad (8.9)$$

where A_p in (8.9) is the attenuation in dB at the passband edge, i.e., at $\omega = 1$ (at $\omega = \omega_p$, in general).

While Legendre filters differ from Chebyshev Type I filters, all of the characteristics considered in this book (magnitude frequency response, phase response, phase delay, group delay, unit impulse response, unit step response) are very similar. In fact, with $\epsilon = 1$, the unmodified Legendre filter differs very little indeed from a Chebyshev Type I filter with 1 dB ripple in all ways considered in this book except that the passband ripple is not equiripple and is less (it varies with order, but is about 0.7 dB). Modified first-associated Legendre filters with $\epsilon = 1$ have a **Shaping Factor** a little less desirable than Chebyshev Type I filters of the same order and with a ripple of 1 dB . While the ripple in the passband is not equiripple, it is less than 0.2 dB for orders greater than two. The phase response is more linear than the Chebyshev filter, as can most readily be seen in the group delay response.

While the modified second-associated Legendre filter has a **Shaping Factor** that is not as desirable as a Chebyshev Type I filter of the same order, it exceeds that of a Butterworth filter of the same order, and will be used for more detailed presentation. See **Figure 8.15** for plots (8.8) for the modified second-associated Legendre filter, with a normalized ω_c of unity ($A_p = 3$ dB), and for several values of N . See **Figure 8.16** for detailed plots across the passband: the passband ripple is less than 0.06 dB for orders greater than two. The phase response is shown in **Figure 8.17**. The phase delay is shown in **Figure 8.18** and the group delay in **Figure**

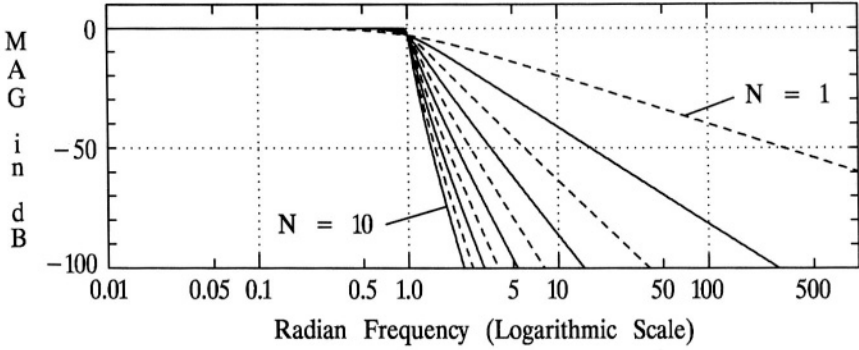


Figure 8.15 The Modified Second-Associated Legendre magnitude frequency response. Plots of (8.8) for $\omega_p = 1$, $A_p = 3$ dB, and for values of N from 1 through 10.

8.19. Note that the group delay is significantly more constant across the passband, and that the peak delay near the normalized passband edge frequency of unity is significantly less than that of a Chebyshev Type I filter (see Figure 4.7). The unit impulse response is shown in Figure 8.20 and the unit step response in Figure 8.21.

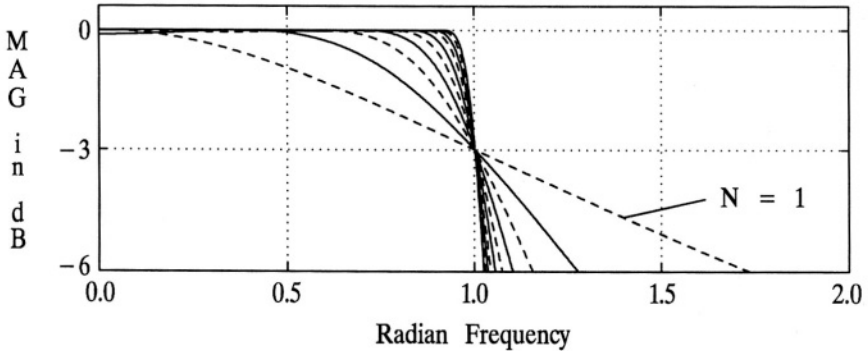


Figure 8.16 Passband details of the Modified Second-Associated Legendre magnitude response. Plots of (8.8) for $\omega_p = 1$, $A_p = 3$ dB, and values of N from 1 through 10.

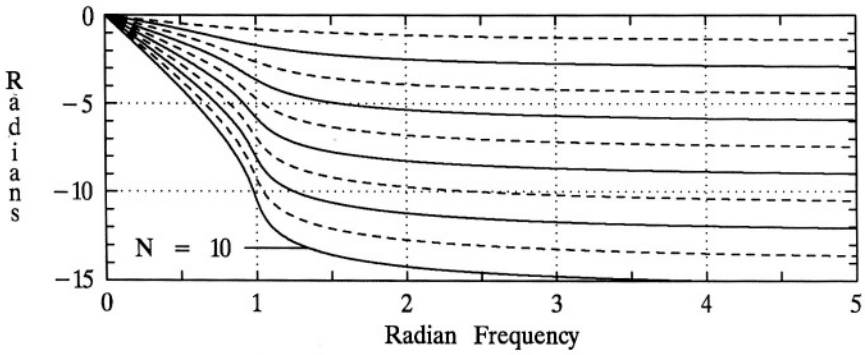


Figure 8.17 Phase response of the Modified Second-Associated Legendre filter with $\omega_p = 1$, $A_p = 3 \text{ dB}$, and values of N from 1 through 10.

On the accompanying disk is the MATLAB function *LEGENDE*. This function may be used to design a Legendre lowpass filter, for a given order, passband corner frequency ω_p , passband corner attenuation A_p , and coefficient m : 0 for the standard Legendre filter, 1 for first-associated, and 2 for the second-associated Legendre filter.

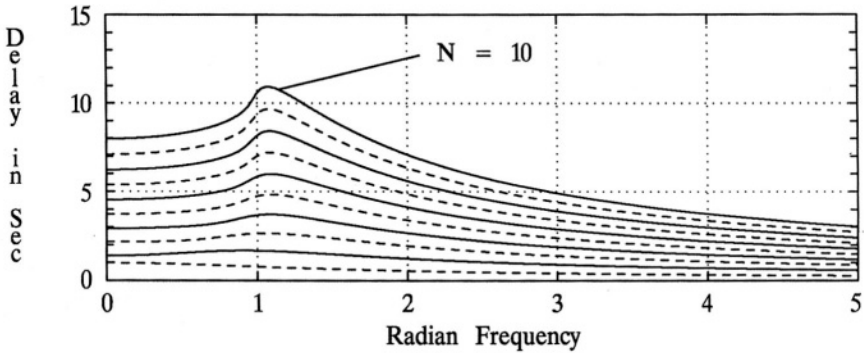


Figure 8.18 The phase delay response of the Modified Second-Associated Legendre filter with $\omega_p = 1$, $A_p = 3 \text{ dB}$, and values of N from 1 through 10.

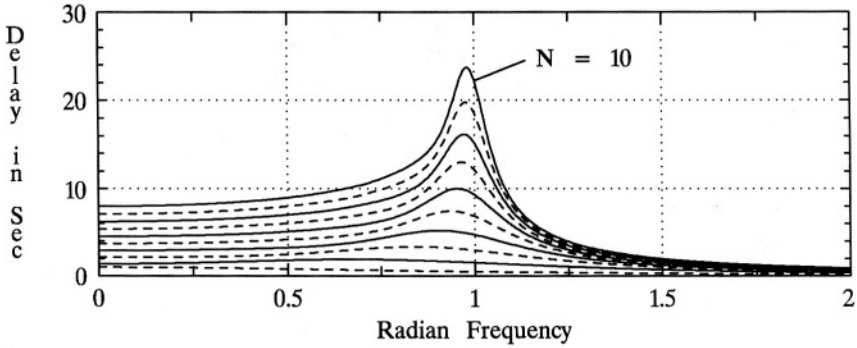


Figure 8.19 The group delay response of the Modified Second-Associated Legendre filter with $\omega_p = 1$, $A_p = 3 \text{ dB}$, and values of N from 1 through 10.

8.4 ULTRASPHERICAL FILTERS

Ultraspherical filters are based on ultraspherical (or Gegenbauer) polynomials (Johnson and Johnson, 1966; Lindquist, 1977), which are a special case of Jacobi polynomials. Ultraspherical filters have several design parameters: the order of the

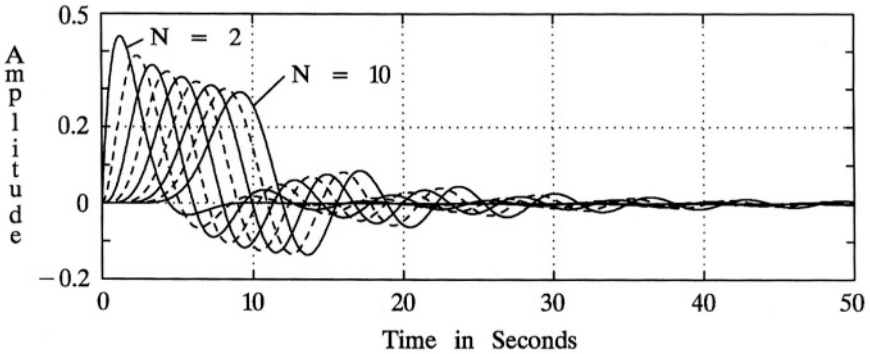


Figure 8.20 The unit impulse response of the Modified Second-Associated Legendre filter with $\omega_p = 1$, $A_p = 3 \text{ dB}$, and values of N from 2 through 10.

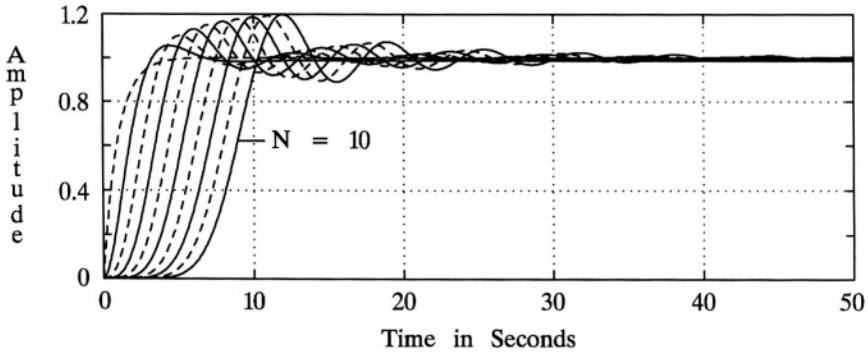


Figure 8.21 The unit step response of the Modified Second-Associated Legendre filter with $\omega_p = 1$, $A_p = 3$ dB, and values of N from 1 through 10.

filter N , the passband corner frequency ω_p , a parameter that controls the attenuation at the passband corner frequency ϵ , and a parameter that controls the response shape α . An interesting feature of ultraspherical filters is that they include Butterworth, Chebyshev Type I, and Legendre filters as special cases, with the appropriate value of α .

The magnitude-squared frequency response is defined as follows:

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 [F_N^{(\alpha)}(\omega/\omega_p)]^2} \quad (8.10)$$

where

$$F_N^{(\alpha)}(\omega/\omega_p) = \frac{N!}{(1+\alpha)_N} \Gamma_N^{(\alpha, \alpha)}(\omega/\omega_p) \quad ,$$

N is the order, α is a real number > -1 , and $\Gamma_N^{(\alpha, \alpha)}(\omega/\omega_p)$ is an ultraspherical polynomial which is the special case, $\beta = \alpha$, of the Jacobi polynomial $\Gamma_N^{(\alpha, \beta)}(\omega/\omega_p)$ (Selby, 1970),

$$(1 + \alpha)_N = (1 + \alpha)(2 + \alpha) \cdots (N + \alpha) \quad ,$$

and $(1 + \alpha)_0 = 1$.

It can be shown that

$$\Gamma_N^{(\alpha, \beta)}(-\omega/\omega_p) = (-1)^N \Gamma_N^{(\beta, \alpha)}(\omega/\omega_p) \quad ,$$

and therefore, to have an even function, which is necessary to satisfy the *Analog*

Design Theorem. it is necessary that $\beta = \alpha$. It can also be shown that to insure the zeros of $F_N^{(\alpha)}(\omega/\omega_p)$ are within $|\omega| < \omega_p$, and that $|F_N^{(\alpha)}(\omega/\omega_p)|$ monotonically increases for $|\omega| > \omega_p$, it is necessary that $\alpha > -1$.

An explicit formula for the ultraspherical filter polynomials, where, for convenience, ω_p has been normalized to unity, is as follows:

$$F_N^{(\alpha)}(\omega) = \sum_{k=0}^N \frac{(-N)_k (2\alpha + N + 1)_k}{k! (1 + \alpha)_k} \left(\frac{1 - \omega}{2} \right)^k \quad (8.11)$$

An important property readily follows from (8.11): $F_N^{(\alpha)}(1) = 1$ for all N and α . Therefore, the required value for ϵ is as follows:

$$\epsilon = \sqrt{10^{A_p/10} - 1} \quad (8.12)$$

where A_p in (8.12) is the attenuation in dB at the passband edge, i.e., at $\omega = 1$ (at $\omega = \omega_p$, in general).

A recursion may also be used:

$$(2\alpha + N) F_N^{(\alpha)}(\omega) = (2\alpha + 2N - 1) \omega F_{N-1}^{(\alpha)}(\omega) - (N - 1) F_{N-2}^{(\alpha)}(\omega) \quad (8.13)$$

where $F_0^{(\alpha)}(\omega) = 1$ and $F_1^{(\alpha)}(\omega) = \omega$. It can readily be seen from (8.13) that, if $\alpha = \infty$, $F_N^{(\alpha)}(\omega) = \omega^N$, which would make (8.10) a Butterworth response.

It can also be shown that if $\alpha = -0.5$ that (8.10) is a Chebyshev Type I response. If $\alpha = m$, where $m = 0, 1, \dots$, any non-negative integer, then (8.10) is a Legendre response: if $m = 0$ the response would be the standard Legendre response, otherwise the modified m -th associated Legendre response.

Values of α other than those mentioned immediately above yield responses unique to the ultraspherical response. One will be chosen here for display of response characteristics: let $\alpha = 1$, and $\epsilon = 1$ ($A_p = 3$ dB). This value of α is selected because it is an interesting compromise between the Butterworth and the Chebyshev Type I response. The ultraspherical magnitude response has a **Shaping Factor** and **Filter Selectivity** considerably better than a Butterworth response of the same order, but not as good as a Chebyshev Type I with 1 dB of ripple, but the ripple in the passband of the ultraspherical response is much less than in the Chebyshev response. The other responses are also compromises between that of the Butterworth and the Chebyshev Type I. See **Figure 8.22** for plots of (8.10) for the ultraspherical filter, for several values of N . See **Figure 8.23** for detailed plots across the passband: the passband ripple is less than 0.2 dB for orders greater than two and becomes less and less as the order becomes greater. The phase response is shown in **Figure 8.24**. The phase delay is shown in **Figure 8.25** and the group delay in **Figure 8.26**. Note that the group delay is significantly more constant across the passband, and that the peak delay near the normalized passband edge frequency of unity is significantly less than

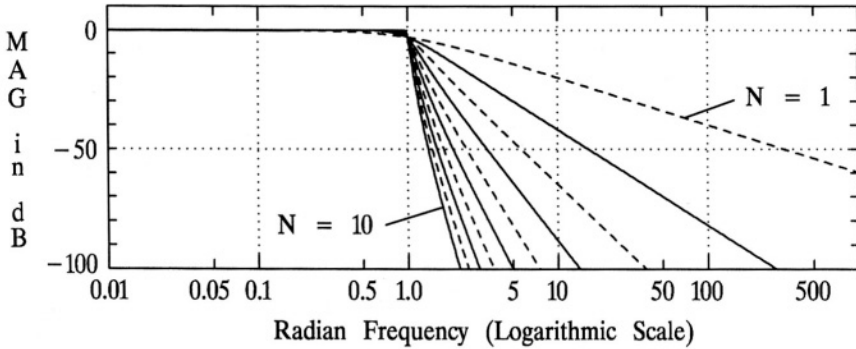


Figure 8.22 The ultraspherical magnitude frequency response. Plots of (8.10) for $\omega_p = 1$, $A_p = 3 \text{ dB}$, $\alpha = 1$, and for values of N from 1 through 10.

that of a Chebyshev Type I filter (see **Figure 4.7**). The unit impulse response is shown in **Figure 8.27** and the unit step response in **Figure 8.28**.

On the accompanying disk is the MATLAB function *ULTRADE*. This function may be used to design an ultraspherical lowpass filter, for a given order, passband corner frequency ω_p , passband corner attenuation A_p , and coefficient α .

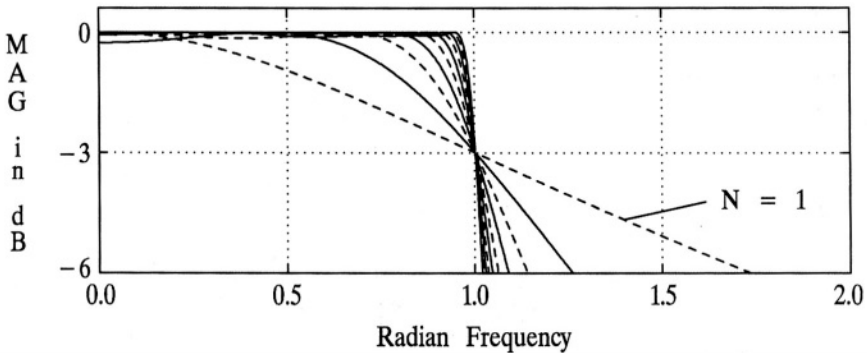


Figure 8.23 Passband details of the ultraspherical magnitude response. Plots of (8.10) for $\omega_p = 1$, $A_p = 3 \text{ dB}$, $\alpha = 1$, and for values of N from 1 through 10.

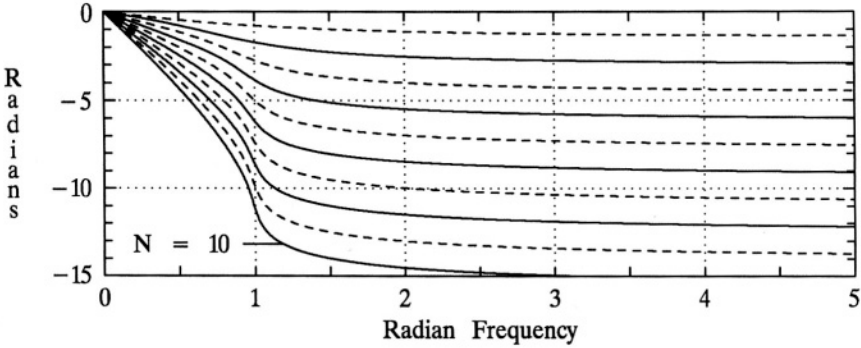


Figure 8.24 Phase response of the ultraspherical filter with $\omega_p = 1$, $A_p = 3 \text{ dB}$, $\alpha = 1$, and for values of N from 1 through 10.

8.5 PAPOULIS FILTERS

Papoulis filters (Lindquist, 1977; Papoulis, 1958; Papoulis, 1959) have a magnitude-squared frequency response defined as follows:

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 L_N^2(\omega/\omega_p)} \quad (8.14)$$

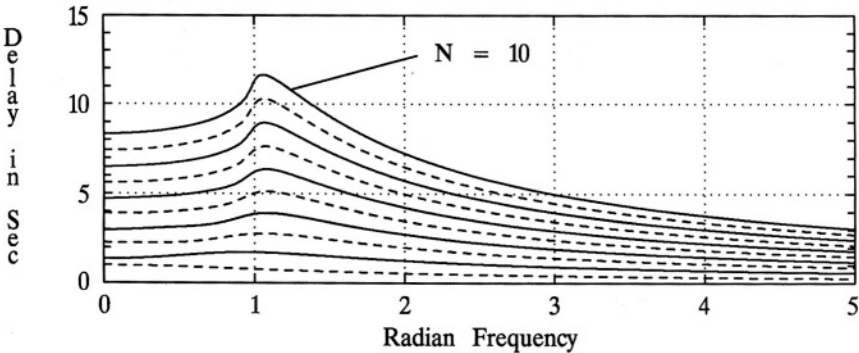


Figure 8.25 The phase delay response of the ultraspherical filter with $\omega_p = 1$, $A_p = 3 \text{ dB}$, $\alpha = 1$, and for values of N from 1 through 10.

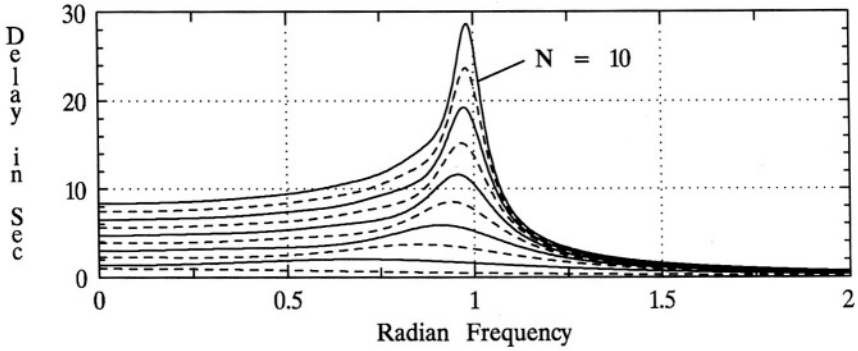


Figure 8.26 The group delay response of the ultraspherical filter with $\omega_p = 1$, $A_p = 3 \text{ dB}$, $\alpha = 1$, and for values of N from 1 through 10.

where ω_p defines the edge of the passband, ϵ controls the filter attenuation at $\omega = \omega_p$, and $L_N(\omega/\omega_p)$ is a polynomial related to the Legendre polynomials as described below. This filter has the characteristic of having the maximum negative response slope at $\omega = \omega_p$ under the conditions of a monotonically nonincreasing response. That is,

$$-\left. \frac{d|H(j\omega)|}{d\omega} \right|_{\omega = \omega_p}$$

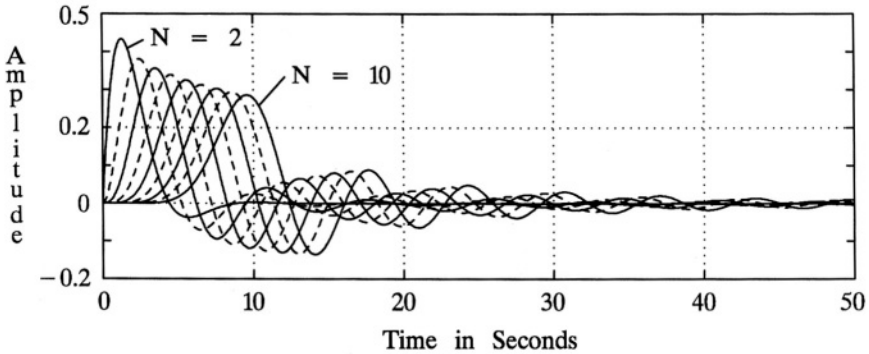


Figure 8.27 The unit impulse response of the ultraspherical filter with $\omega_p = 1$, $A_p = 3 \text{ dB}$, $\alpha = 1$, and for values of N from 2 through 10.

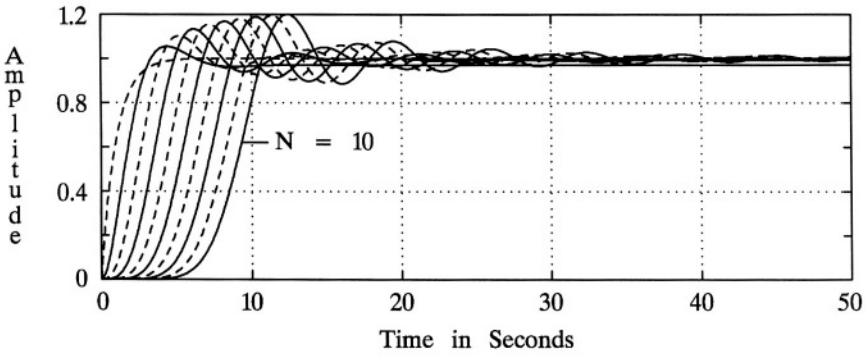


Figure 8.28 The unit step response of the ultraspherical filter with $\omega_p = 1$, $A_p = 3 \text{ dB}$, $\alpha = 1$, and for values of N from 1 through 10.

will be the maximum for all filters with a monotonically nonincreasing frequency response. Note that this differs from the definition of *Filter Selectivity* given in (2.37), which is evaluated at $\omega = \omega_c$. If $\epsilon = 1$ in (8.14), then $\omega_p = \omega_c$, and the Papoulis filter would have the largest *Filter Selectivity* of all filters of equal order that have a monotonically nonincreasing magnitude response.

For odd orders, where $N = 2k + 1$, explicit expressions for $L_N^2(\omega)$, where, for convenience, ω_p has been normalized to unity, may be obtained as follows:

$$L_N^2(\omega) = \int_{-1}^{2\omega^2-1} v^2(x) dx \quad , \quad (8.15)$$

where

$$v(x) = a_0 + a_1 P_1(x) + \dots + a_k P_k(x) \quad ,$$

$$a_0 = \frac{a_1}{3} = \frac{a_2}{5} = \dots = \frac{a_r}{2r+1} = \frac{1}{\sqrt{2}(k+1)} \quad ,$$

and $P_k(x)$ is the unmodified Legendre polynomial of order k , as defined in Section 8.3.

For example, consider $N = 3$, and therefore $k = 1$:

$$v(x) = \frac{1}{2\sqrt{2}} + \frac{3}{2\sqrt{2}} x \quad ,$$

and applying (8.15),

$$L_3^2(\omega) = 3\omega^6 - 3\omega^4 + \omega^2 \quad .$$

$$v^2(x) = \frac{1}{8} + \frac{3}{4}x + \frac{9}{8}x^2 \quad ,$$

For even orders, where $N = 2k + 2$, explicit expressions for $L_N^2(\omega)$ may be obtained as follows:

$$L_N^2(\omega) = \int_{-1}^{2\omega^2-1} \phi(x) dx \quad , \quad (8.16)$$

where

$$\phi(x) = K(x+1) \left(\frac{dP_{k+1}(x)}{dx} \right)^2 \quad ,$$

and K is such as to make $L_N^2(1) = 1$.

For example, consider $N = 4$, and therefore $k = 1$:

$$\phi(x) = K(x+1) \left(\frac{dP_2(x)}{dx} \right)^2 = K(9x^3 + 9x^2) \quad ,$$

and applying (8.16),

$$L_4^2(\omega) = K[(9/4)(2\omega^2 - 1)^4 + 3(2\omega^2 - 1)^3 + (3/4)]$$

$$= K[36\omega^8 - 48\omega^6 + 18\omega^4] \quad ,$$

and therefore,

$$L_4^2(\omega) = 6\omega^8 - 8\omega^6 + 3\omega^4 \quad .$$

It should be noted that the required value for ϵ is as follows:

$$\epsilon = \sqrt{10^{A_p/10} - 1} \quad , \quad (8.17)$$

where A_p in (8.17) is the attenuation in dB at the passband edge, i.e., at $\omega = 1$ (at $\omega = \omega_p$, in general).

While the Papoulis filter has a *Shaping Factor* that is not as desirable as a Chebyshev Type I filter of the same order, it exceeds that of a Butterworth filter of the same order. See Figure 8.29 for plots of (8.14) for the Papoulis filter with $A_p = 1$ dB ($\epsilon = 0.5088$), with a normalized ω_c of unity, and for several values of N . See Figure 8.30 for detailed plots across the passband: note the sharp passband corners for the higher orders, much sharper than that of a Butterworth filter (see

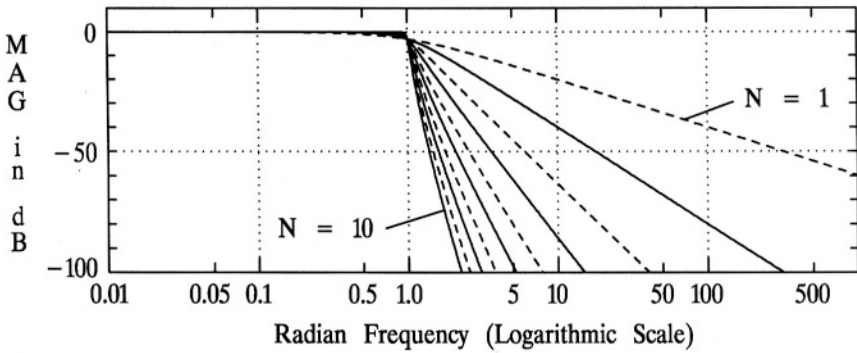


Figure 8.29 The Papoulis filter magnitude frequency response. Plots of (8.14) for $\omega_c = 1$, $A_p = 1$ dB, and for values of N from 1 through 10.

Figure 3.2). Note also that the ripple, or unevenness, across the passband is less than 0.2 dB, whereas, for comparison, the Chebyshev Type I ripple in Figure 4.2 is 1 dB. The phase response is shown in Figure 8.31. The phase delay is shown in Figure 8.32 and the group delay in Figure 8.33. Note that the group delay is significantly more constant across the passband, and that the peak delay near the normalized passband edge frequency of unity is significantly less than that of a Chebyshev Type

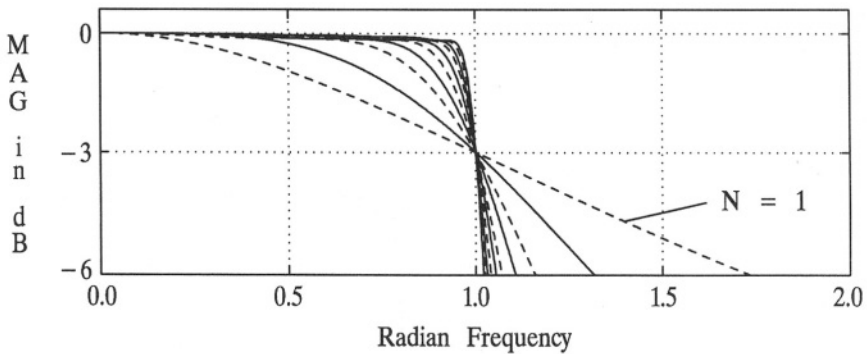


Figure 8.30 Passband details of the Papoulis filter magnitude response. Plots of (8.14) for $\omega_c = 1$, $A_p = 1$ dB, and for values of N from 1 through 10.

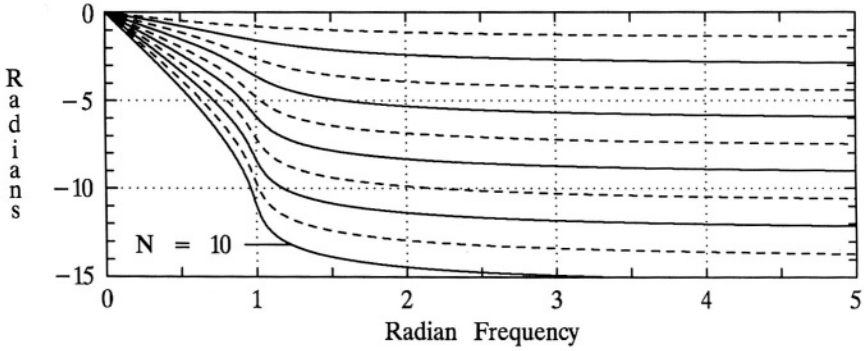


Figure 8.31 Phase response of the Papoulis filter with $\omega_c = 1$, $A_p = 1 \text{ dB}$, and values of N from 1 through 10.

I filter (see **Figure 4.7**), however it is less desirable than that of a Butterworth filter (see **Figure 3.9**). The unit impulse response is shown in **Figure 8.34** and the unit step response in **Figure 8.35**.

On the accompanying disk is the MATLAB function *PAPOULDE*. This function may be used to design a Papoulis lowpass filter, for a given order, passband corner frequency ω_p , and passband corner attenuation A_p .

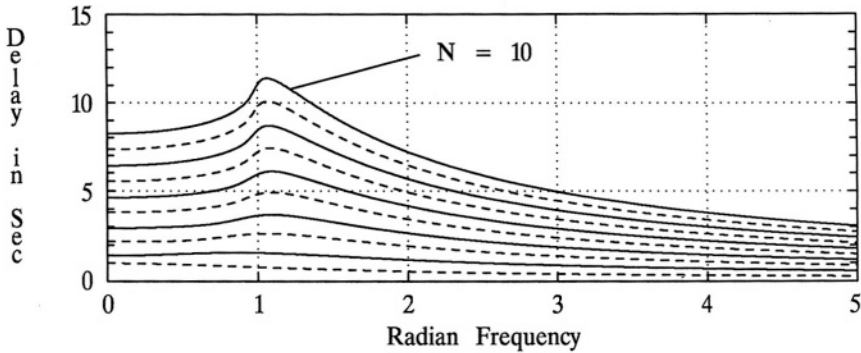


Figure 8.32 Phase delay of the Papoulis filter with $\omega_c = 1$, $A_p = 1 \text{ dB}$, and values of N from 1 through 10.

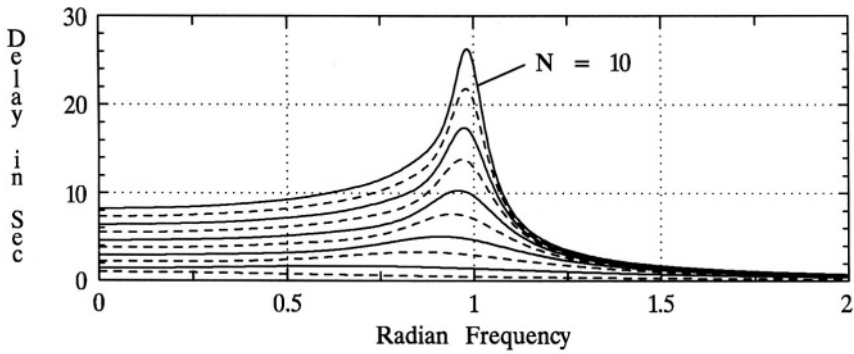


Figure 8.33 Group delay of the Papoulis filter with $\omega_c = 1$, $A_p = 1 \text{ dB}$, and values of N from 1 through 10.

8.6 HALPERN FILTERS

Halpern filters (Halpern, 1969) are closely related to Papoulis filters, but optimize the *Shaping Factor* under the conditions of a monotonically nonincreasing response. The magnitude-squared frequency response is defined as follows:

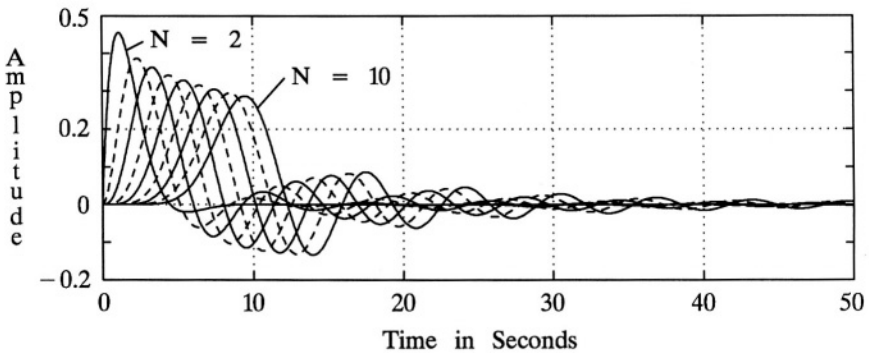


Figure 8.34 The unit impulse response of the Papoulis filter with $\omega_c = 1$, $A_p = 1 \text{ dB}$, and values of N from 2 through 10.

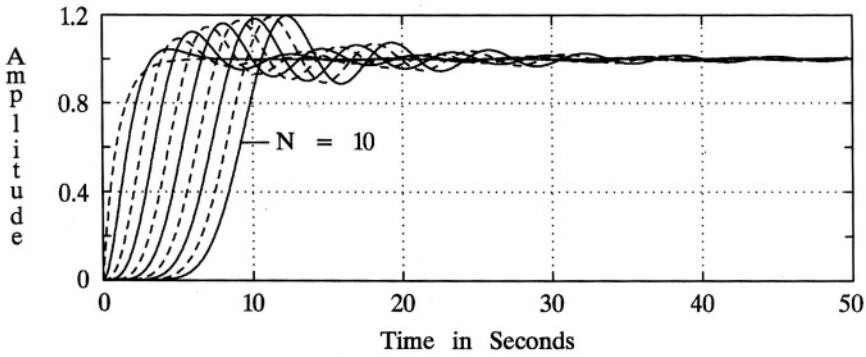


Figure 8.35 The unit step response of the Papoulis filter with $\omega_c = 1$, $A_p = 1$ dB, and values of N from 1 through 10.

$$|H(j\omega)|^2 = \frac{1}{1 + \epsilon^2 T_N^2(\omega/\omega_p)} \tag{8.18}$$

where ω_p defines the edge of the passband, ϵ controls the filter attenuation at $\omega = \omega_p$, and $T_N(\omega/\omega_p)$ is a polynomial related to the Jacobi polynomials. This filter has the characteristic of having the maximum asymptotic negative magnitude response slope in the transition/stop band under the conditions of a monotonically nonincreasing response that includes the passband. This will result in the smallest **Shaping Factor** for a given order of all filters with a monotonically nonincreasing frequency response.

Explicit expressions for $T_N^2(\omega)$, where, for convenience, ω_p has been normalized to unity, may be obtained as follows:

$$T_N^2(\omega) = \int_0^\omega x U_{N-1}^2(x) dx \tag{8.19}$$

where,

$$U_{2k}(x) = \sqrt{4k+2} \sum_{m=0}^k (-1)^{k-m} x^{2m} \frac{(m+k)!}{(k-m)!(m!)^2} \tag{8.19}$$

and,

$$U_{2k+1}(x) = 2\sqrt{k+1} \sum_{m=0}^k (-1)^m x^{2(k-m)+1} \frac{(2k+1-m)!}{m!(k+1-m)!(k-m)!} .$$

For example, consider $N = 3$, and therefore $k = 1$:

$$U_2(x) = \sqrt{6} (2x^2 - 1) ,$$

and applying (8.19),

$$T_3^2(\omega) = 4\omega^6 - 6\omega^4 + 3\omega^2 .$$

While the Halpern filter has a *Shaping Factor* that is not as desirable as a Chebyshev Type I filter of the same order, it exceeds that of a Butterworth filter, and a Papoulis filter of the same order. See **Figure 8.36** for plots of (8.18) for the Halpern filter with $A_p = 1 \text{ dB}$ ($\epsilon = 0.5088$), with a normalized ω_c of unity, and for several values of N . See **Figure 8.37** for detailed plots across the passband: note the sharp passband corners for the higher orders, but that there is a decreasing response trend across the passband. The phase response is shown in **Figure 8.38**. The phase delay is shown in **Figure 8.39** and the group delay in **Figure 8.40**. Note that the group delay is significantly more constant across the passband, and that the peak delay near the normalized passband edge frequency of unity is significantly less than that of a Chebyshev Type I filter (see **Figure 4.7**), however it is less desirable than that of a Butterworth filter (see **Figure 3.9**), and slightly less desirable than that of a Papoulis filter. The unit impulse response is shown in **Figure 8.41** and the unit step response in **Figure 8.42**.

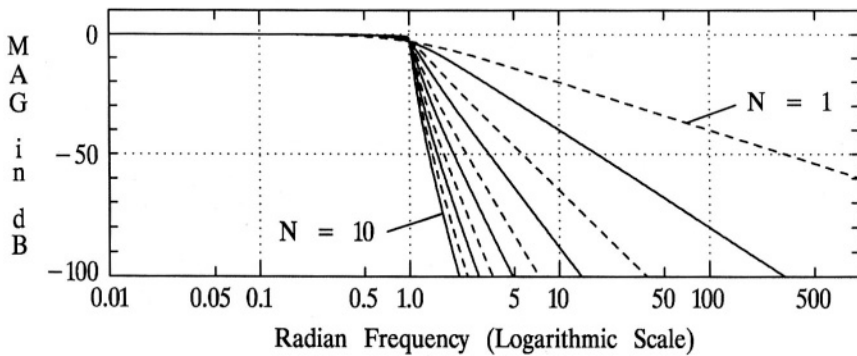


Figure 8.36 The Halpern filter magnitude frequency response. Plots of (8.18) for $\omega_c = 1$, $A_p = 1 \text{ dB}$, and for values of N from 1 through 10.

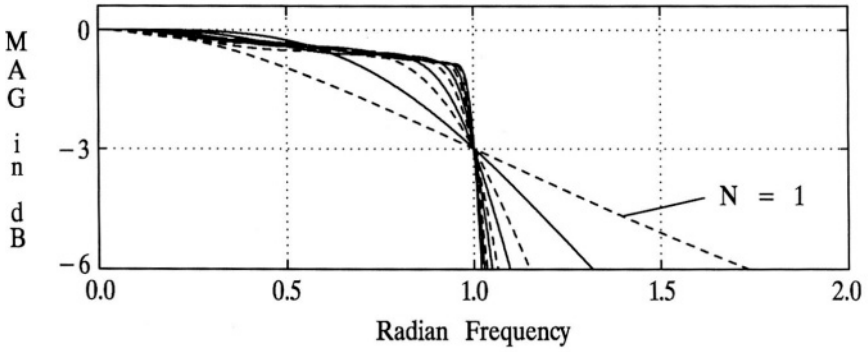


Figure 8.37 Passband details of the Halpern filter magnitude response. Plots of (8.18) for $\omega_c = 1$, $A_p = 1$ dB, and for values of N from 1 through 10.

On the accompanying disk is the MATLAB function *HALPDE*. This function may be used to design a Halpern lowpass filter, for a given order, passband corner frequency ω_p , and passband corner attenuation A_p .

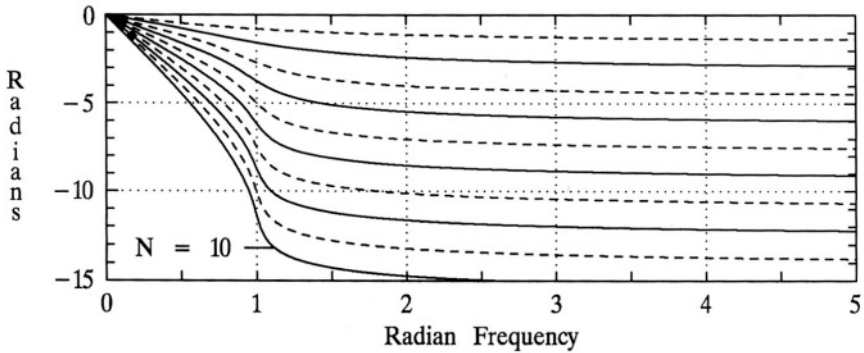


Figure 8.38 Phase response of the Halpern filter with $\omega_c = 1$, $A_p = 1$ dB, and values of N from 1 through 10.

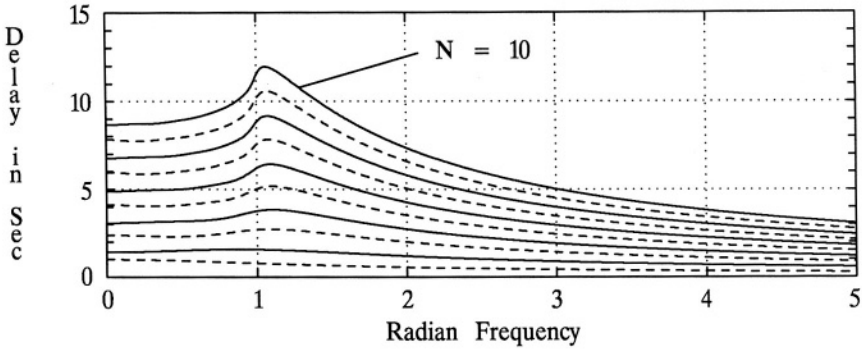


Figure 8.39 Phase delay of the Halpern filter with $\omega_c = 1$, $A_p = 1$ dB, and values of N from 1 through 10.

8.7 PHASE-COMPENSATED FILTERS

All-pass transfer functions were briefly presented in **Section 2.12**, and as indicated there, an important application for all-pass filters is in the phase compensation of an existing filter that has an acceptable magnitude frequency response but the phase response is less than desirable. In this section, specific results will be given, illustrating phase compensation. The disadvantages of phase compensation

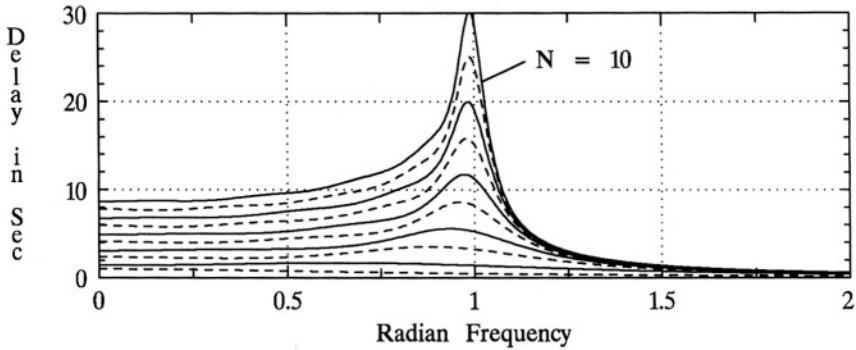


Figure 8.40 Group delay of the Halpern filter with $\omega_c = 1$, $A_p = 1$ dB, and values of N from 1 through 10.

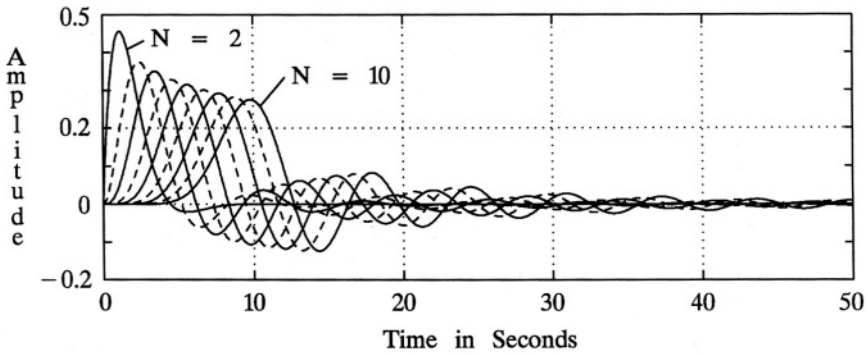


Figure 8.41 The unit impulse response of the Halpern filter with $\omega_c = 1$, $A_p = 1$ dB, and values of N from 2 through 10.

include the additional design necessary, and the higher overall transfer function order that results, indicating the required increased complexity of implementation. However, as seen in this section, the phase response and related responses are significantly improved, assuming the increased delay is acceptable, and time-domain responses are also significantly improved.

The specific results illustrated in this section are for phase compensation applied to a tenth-order Butterworth filter. This particular filter is selected because

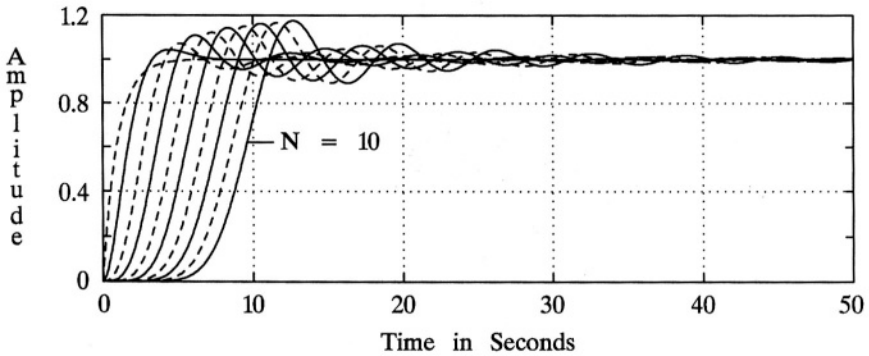


Figure 8.42 The unit step response of the Halpern filter with $\omega_c = 1$, $A_p = 1$ dB, and values of N from 1 through 10.

of its relatively good magnitude frequency response combined with a relatively good phase response. Only a Bessel filter or a Gaussian filter of the same order would have superior phase characteristics, compared with all other filter types in this book, and yet the Butterworth magnitude frequency response is significantly superior to either the Bessel or the Gaussian filter. Since the phase compensation filters are all-pass, and either implemented in cascade with the Butterworth filter or combined with the Butterworth transfer function prior to implementation (transfer functions multiplied), the phase compensation has no effect on the magnitude frequency response. Therefore, no magnitude frequency response figures are required, since they would be identical to that given in **Chapter 3**: ω_c , here, is also normalized to unity. However, the phase response, phase delay, group delay, unit impulse response, and unit step response are all effected by phase compensation.

The results of four compensation filters are illustrated in the figures that follow. The four compensation filters are all-pass filters of order 1, 2, 3, and 4. They were each empirically designed (trial and error), to achieve a group delay response superior to that of the basic Butterworth filter (all-pass of order one), or of the immediately lower order phase-compensated filter.

The 1st-order phase-compensation filter has the following transfer function:

$$H_1(s) = \frac{-(s - \mu_1)}{s + \mu_1} ,$$

where $\mu_1 = 0.9$. The 2nd-order phase-compensation filter has the following transfer function:

$$H_2(s) = \frac{s^2 - \xi_1 s + \lambda_1}{s^2 + \xi_1 s + \lambda_1} = \frac{s^2 - (\omega_{o1}/Q_1)s + \omega_{o1}^2}{s^2 + (\omega_{o1}/Q_1)s + \omega_{o1}^2} ,$$

where $Q_1 = 0.6$ and $\omega_{o1} = 0.7$. The 3rd-order phase-compensation filter is as follows:

$$H_3(s) = -H_2(s) \frac{s - \mu_2}{s + \mu_2} = -\frac{s^2 - (\omega_{o1}/Q_1)s + \omega_{o1}^2}{s^2 + (\omega_{o1}/Q_1)s + \omega_{o1}^2} \frac{s - \mu_2}{s + \mu_2} ,$$

where $\mu_2 = 1.4$, and as above, $Q_1 = 0.6$ and $\omega_{o1} = 0.7$. The 4th-order phase-compensation filter is as follows:

$$H_4(s) = H_2(s) \frac{s^2 - (\omega_{o2}/Q_2)s + \omega_{o2}^2}{s^2 + (\omega_{o2}/Q_2)s + \omega_{o2}^2} ,$$

where $Q_2 = 0.6$ and $\omega_{o2} = 0.95$.

In **Figure 8.43** is shown the phase response of the uncompensated Butterworth filter, and the phase response of the phase-compensated Butterworth filter using the

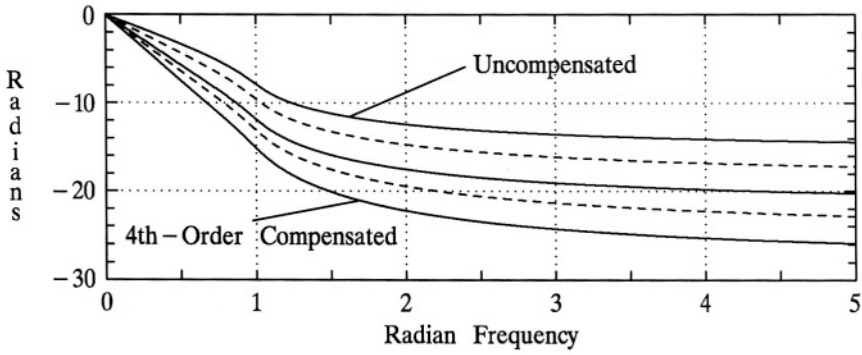


Figure 8.43 The phase response of a 10th-order Butterworth filter with $\omega_c = 1$, and the same filter phase-compensated with an all-pass filter of order 1, 2, 3, and 4.

1st-order, 2nd-order, 3rd-order and 4th-order all-pass filters. As can be seen in the figure, particularly with a straight-edge, the phase response becomes increasingly linear as the order of the phase-compensation filter is increased.

In **Figure 8.44** is shown the phase delay response of the uncompensated Butterworth filter, and the phase delay response of the phase-compensated Butterworth filter using the 1st-order, 2nd-order, 3rd-order and 4th-order all-pass filters. As can

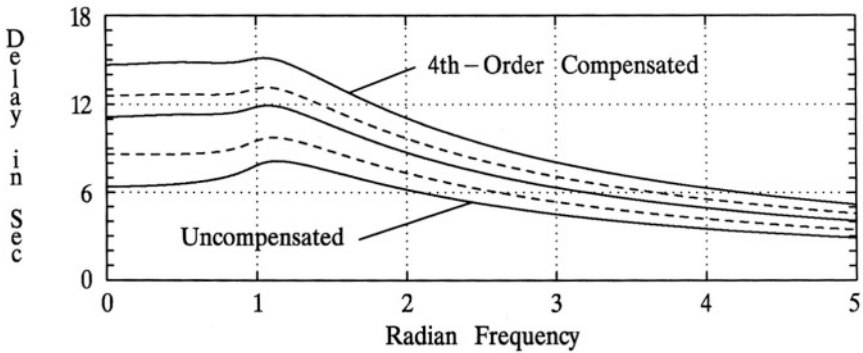


Figure 8.44 The phase delay of a 10th-order Butterworth filter with $\omega_c = 1$, and the same filter phase-compensated with an all-pass filter of order 1, 2, 3, and 4.

be seen in the figure, the phase delay response becomes increasingly flat across the passband as the order of the phase-compensation filter is increased. More precise comparisons may be achieved by comparing the maximum delay (for ω a little greater than unity) to the minimum delay across the passband. The maximum values are 8.1551, 9.7460, 11.9252, 13.1433, and 15.1301 for the uncompensated Butterworth filter, the Butterworth filter compensated with the 1st-order all-pass, the 2nd-order all-pass, the 3rd-order all-pass, and the 4th-order all-pass, respectively. The minimum values are 6.3925, 8.6059, 11.1544, 12.5830, and 14.6632 for the uncompensated Butterworth filter, the Butterworth filter compensated with the 1st-order all-pass, the 2nd-order all-pass, the 3rd-order all-pass, and the 4th-order all-pass, respectively. The ratio of the maximum to the minimum values are 1.2757, 1.1325, 1.0691, 1.0445, and 1.0318 for the uncompensated Butterworth filter, the Butterworth filter compensated with the 1st-order all-pass, the 2nd-order all-pass, the 3rd-order all-pass, and the 4th-order all-pass, respectively. The difference between the maximum and minimum values are 1.7626, 1.1401, 0.7708, 0.5603, and 0.4669 for the uncompensated Butterworth filter, the Butterworth filter compensated with the 1st-order all-pass, the 2nd-order all-pass, the 3rd-order all-pass, and the 4th-order all-pass, respectively. The uncompensated Butterworth filter has a phase delay that varies 1.7626 seconds as the radian frequency varies from 0 to about 1.2 (27.6%), whereas the Butterworth filter compensated by the 4th-order all-pass has a phase delay that varies only 0.4669 seconds (3.2%).

In **Figure 8.45** is shown the group delay of the uncompensated Butterworth filter, and the group delay of the phase-compensated Butterworth filter using the 1st-order, 2nd-order, 3rd-order and 4th-order all-pass filters. As can be seen in the

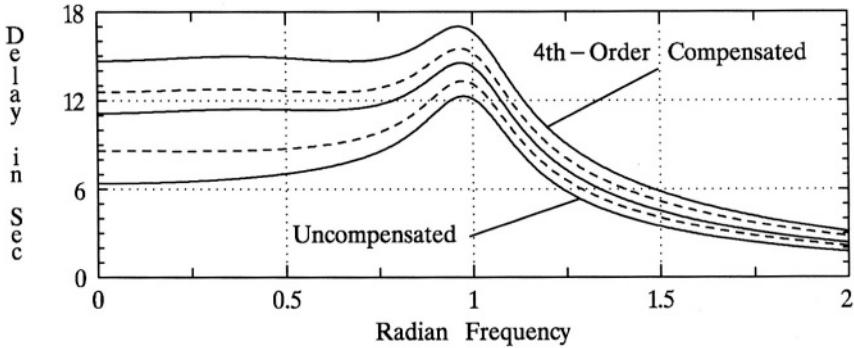


Figure 8.45 The group delay of a 10th-order Butterworth filter with $\omega_c = 1$, and the same filter phase-compensated with an all-pass filter of order 1, 2, 3, and 4.

figure, the group delay becomes increasingly flat across the passband as the order of the phase-compensation filter is increased. More precise comparisons may be achieved by comparing the maximum delay to the minimum delay across the passband. The maximum values are 12.2823, 13.3019, 14.5175, 15.4838, and 16.0075 for the uncompensated Butterworth filter, the Butterworth filter compensated with the 1st-order all-pass, the 2nd-order all-pass, the 3rd-order all-pass, and the 4th-order all-pass, respectively. The minimum values are 6.3925, 8.5989, 11.1544, 12.5616, and 14.6377 for the uncompensated Butterworth filter, the Butterworth filter compensated with the 1st-order all-pass, the 2nd-order all-pass, the 3rd-order all-pass, and the 4th-order all-pass, respectively. The ratio of the maximum to the minimum values are 1.9214, 1.5469, 1.3015, 1.2326, and 1.1612 for the uncompensated Butterworth filter, the Butterworth filter compensated with the 1st-order all-pass, the 2nd-order all-pass, the 3rd-order all-pass, and the 4th-order all-pass, respectively. The difference between the maximum and minimum values are 5.8898, 4.7030, 3.3631, 2.9222, and 2.3598 for the uncompensated Butterworth filter, the Butterworth filter compensated with the 1st-order all-pass, the 2nd-order all-pass, the 3rd-order all-pass, and the 4th-order all-pass, respectively. The uncompensated Butterworth filter has a group delay that varies 5.8898 seconds as the radian frequency varies from 0 to 1 (92.1%), whereas the Butterworth filter compensated by the 4th-order all-pass has a group delay that varies only 2.3598 seconds (16.1%).

In **Figure 8.46** is shown the unit impulse response of the uncompensated Butterworth filter, and the unit impulse response of the phase-compensated Butterworth filter using the 1st-order, 2nd-order, 3rd-order and 4th-order all-pass filters. As can be seen in the figure, as expected, the peak of the unit impulse

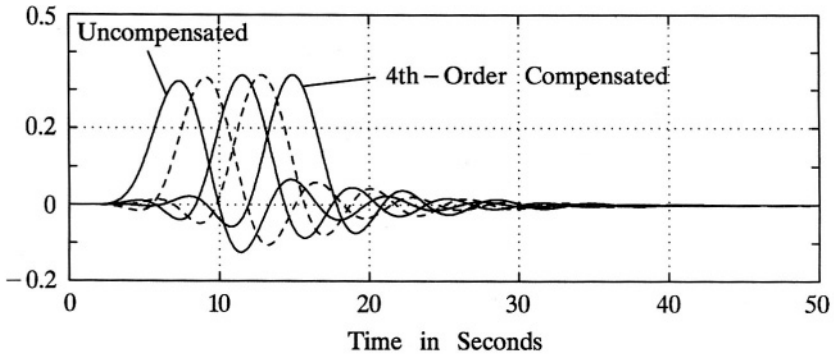


Figure 8.46 The unit impulse response of a 10th-order Butterworth filter with $\omega_c = 1$, and the same filter phase-compensated with an all-pass filter of order 1, 2, 3, and 4.

response shifts farther to the right (increased delay) as the order of the phase-compensation filter is increased. Note that the time at which the impulse response peak occurs correlates well with the low-frequency delay as indicated in either **Figure 8.44** or **8.45**. It is interesting to note that the impulse response increasingly exhibits even symmetry about the time of the response peak as the order of the phase-compensation filter is increased. Compare the unit impulse response of the Butterworth filter compensated by the 4th-order all-pass with that shown in **Figure 2.3** and expressed in (2.2) for the ideal lowpass filter. The peak value for (2.2) with $\omega_c = 1$ is 0.3183, whereas the peak value in **Figure 8.46** is 0.3399. From (2.2), the impulse response first zero-crossings would be at the time of the peak (0 for (2.2)) $\pm \pi \approx 3.1416$, whereas the first zero-crossings in **Figure 8.46** are 11.9 and 17.8, which are 14.9 (the time of the peak) -3 and $+2.9$. Therefore, the unit impulse response of the 10th-order Butterworth filter compensated by the 4th-order all-pass is beginning to approximate the ideal unit impulse response, with a delay, rather well.

In **Figure 8.47** is shown the unit step response of the uncompensated Butterworth filter, and the unit step response of the phase-compensated Butterworth filter using the 1st-order, 2nd-order, 3rd-order and 4th-order all-pass filters. Compare the unit step response of the Butterworth filter compensated by the 4th-order all-pass with that shown in **Figure 2.4** for the ideal lowpass filter. The unit step response of the 10th-order Butterworth filter compensated by the 4th-order all-pass is beginning to approximate the ideal unit step response, with a delay, rather well.

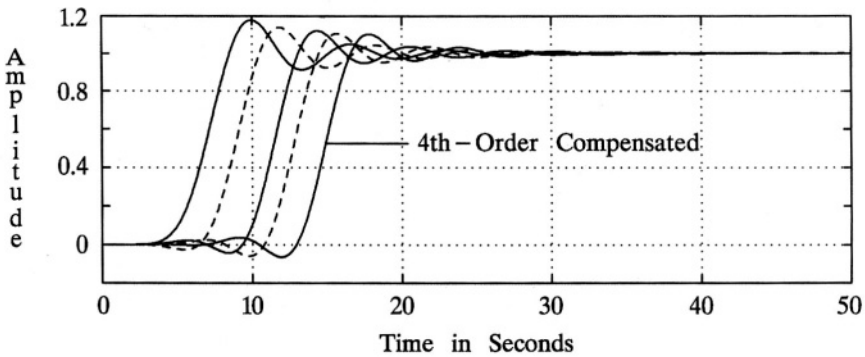


Figure 8.47 The unit step response of a 10th-order Butterworth filter with $\omega_c = 1$, and the same filter phase-compensated with an all-pass filter of order 1, 2, 3, and 4.

8.8 CHAPTER 8 PROBLEMS

- 8.1** Using (8.4) and (8.5), design a transitional filter, where design 1 is a 10th-order Butterworth filter with $\omega_c = 1$, design 2 is a 10th-order Bessel filter with $\omega_c = 1$, and $m = 0.2$. Plot figures similar to Figures 8.1 through 8.7. HINT: use the MATLAB m-files *FIG8_1dat.m* through *FIG8_7dat.m* on the accompanying disk as starting points. Also plot, on one set of axes, the poles of the Butterworth filter, the Bessel filter, and the transitional filter.
- 8.2** Repeat Problem 8.1 with $m = 0.8$.
- 8.3** Beginning with the transfer function of a 3rd-order Butterworth filter with $\omega_c = 1$ (see Chapter 3) denoted as design 1, the transfer function of a 3rd-order Bessel filter normalized for $\omega_c = 1$ (see Chapter 7) denoted as design 2, and $m = 0.5$, using (8.4) and (8.5) manually compute the transfer function of the transitional filter.
- 8.4** Compare the 3rd-order Gaussian filter as given in Section 8.2, based on truncation of the Maclaurin series, with the 3rd-order approximation to a Gaussian filter as given in Problem 2.30: repeat Problem 2.30 for the filter coefficients given in Problem 2.30 and also for those obtained from (8.7).
- 8.5** A convenient feature of a Gaussian filter is the simple expression for the ideal magnitude-squared frequency response given in (8.6). For the ideal Gaussian filter:
- Determine an expression for the power spectral density (PSD) of the filter output given that the input is white noise with variance σ^2 (see Section 2.7).
 - Determine an expression for the autocorrelation function of the filter output given that the input is white noise with variance σ^2 .
 - Determine an expression, as a function of the filter ω_c , for the power in the filter output signal given that the input is white noise with variance σ^2 .
- 8.6** Determine an expression for the unit impulse response for the ideal magnitude-squared frequency response given in (8.6), under the assumption that the phase response is a constant of zero. Discuss how your answer would differ if the phase response was assumed to be linear with a slope of $-\alpha$.

- 8.7** The truncated Maclaurin series approximation to the ideal Gaussian magnitude-squared frequency response given by (8.6) will approximate the ideal with less and less error as the order is increased. However, no realizable filter can ever have a magnitude-squared frequency response that equals (8.6). Show that this statement is true by showing that (8.6) violates the *Paley-Wiener Theorem*.
- 8.8** Determine expressions for the Legendre polynomials ($m = 0$) of orders 2, 3, 4, 5 and 6.
- 8.9** Determine expressions for the first-associated Legendre polynomials, unmodified and modified, of orders 1, 2, 3, 4 and 5.
- 8.10** Determine expressions for the second-associated Legendre polynomials, unmodified and modified, of orders 1, 2, 3, and 4.
- 8.11** In Section 8.3 it is stated that the unmodified Legendre filter with $\epsilon = 1$ differs very little from a Chebyshev Type I filter with 1 dB of ripple except that the Legendre filter has less ripple. Compare the performance of the two filters, each of order 6 and normalized for $\omega_c = 1$. With both filters plotted on the same graph for easy comparison, compute and plot graphs similar to Figures 8.15 through 8.21. In your judgment, what are the advantages and disadvantages of each filter compared with the other?
- 8.12** Under appropriate conditions, a Legendre filter and a Chebyshev Type I filter can be very similar in performance. In this problem, perform a detailed comparison between a 10th-order Chebyshev Type I filter with 0.15 dB of ripple and a normalized ω_c of unity with a 10th-order Modified second-associated Legendre filter with $A_p = 10$ dB and a normalized ω_c of unity. With both filters plotted on the same graph for easy comparison, compute and plot graphs similar to Figures 8.15 through 8.21. In your judgment, what are the advantages and disadvantages of each filter compared with the other?
- 8.13** In Section 8.4 it is stated that a Butterworth filter is a special case of an ultraspherical filter. Demonstrate that if $\alpha = \infty$ the ultraspherical filter is a Butterworth filter, i.e., show that $F_N^{(\infty)}(\omega) = \omega^N$.
- 8.14** In Section 8.4 it is stated that a Chebyshev Type I filter is a special case of an ultraspherical filter. Demonstrate that if $\alpha = -0.5$ the ultraspherical filter is a Chebyshev Type I filter. Specifically, show that $F_N^{(-0.5)}(\omega)$ is equal to the polynomial entries in Table 4.2 for $N = 2, 3, 4$.

- 8.15** In **Problem 8.8** the Legendre polynomials ($m = 0$) are found for orders 2, 3, 4, 5, and 6. Show that these same polynomials are obtainable as special cases of $F_N^{(\alpha)}(\omega)$.
- 8.16** In **Problem 8.9** the modified first-associated Legendre polynomials are found for orders 1, 2, 3, 4, and 5. Show that these same polynomials are obtainable as special cases of $F_N^{(\alpha)}(\omega)$.
- 8.17** In **Problem 8.10** the modified second-associated Legendre polynomials are found for orders 1, 2, 3, and 4. Show that these same polynomials are obtainable as special cases of $F_N^{(\alpha)}(\omega)$.
- 8.18** By making use of special cases of $F_N^{(\alpha)}(\omega)$, determine the modified third-associated Legendre polynomials of orders 3 and 4.
- 8.19** Subjectively compare the performance of a 10th-order ultraspherical filter with $\alpha = 1$, $\epsilon = 1$, and $\omega_c = 1$, as shown in **Figures 8.22** through **8.28**, with that of a 10th-order Chebyshev Type I filter with 1 dB of ripple and $\omega_c = 1$, as shown in the figures in **Chapter 4**. In your judgment, what are the advantages and disadvantages of each filter compared with the other? Plot the poles of each filter, on the same graph for ready comparison.
- 8.20** Determine the polynomial $L_5^2(\omega)$ associated with a 5th-order Papoulis filter.
- 8.21** Determine the polynomial $L_6^2(\omega)$ associated with a 6th-order Papoulis filter.
- 8.22** Given that $\epsilon = 1$, determine the *Filter Selectivity* of a 4th-order Papoulis filter when $\omega_c = 1$. Compare this numerical value with that of a 4th-order Butterworth filter with $\omega_c = 1$. Compare this numerical value with that of a 4th-order Chebyshev Type II filter with $\omega_c = 1$ and $A_s = 60$ dB. Compare this numerical value with that of a 4th-order Chebyshev Type I filter with $\omega_c = 1$ and $A_p = 1$ dB. Interpret the numerical results of this problem and discuss the significance of them.
- 8.23** Show that the Papoulis filter becomes identical to the Butterworth filter when ϵ approaches the limit of zero and, simultaneously, ω_c is maintained normalized to unity. It is sufficient to show this for the 4th-order case.
- 8.24** Determine the polynomial $T_4^2(\omega)$ associated with a 4th-order Halpern filter.
- 8.25** Determine the polynomial $T_5^2(\omega)$ associated with a 5th-order Halpern filter.

- 8.26** Given that $A_p = 1 \text{ dB}$, $a = 3 \text{ dB}$, and $b = 100 \text{ dB}$, determine the *Shaping Factor* of a 4th-order Halpern filter when $\omega_c = 1$. Compare this numerical value with that of a 4th-order Butterworth filter with $\omega_c = 1$. Compare this numerical value with that of a 4th-order Chebyshev Type II filter with $\omega_c = 1$ and $A_s = 60 \text{ dB}$. Compare this numerical value with that of a 4th-order Chebyshev Type I filter with $\omega_c = 1$ and $A_p = 1 \text{ dB}$. Interpret the numerical results of this problem and discuss the significance of them.
- 8.27** Compare the performance of the Halpern filter, as shown in **Figures 8.36** through **8.42**, with that of the related Papoulis filter, as shown in **Figures 8.29** through **8.35**. In your judgment, what are the advantages and disadvantages of each filter compared with the other?
- 8.28** Plot the poles and zeros of the 4th-order phase-compensated 10th-order Butterworth filter, as presented in **Section 8.7**.
- 8.29** Extend the work shown in **Section 8.7** by designing, by trial and error, a 6th-order phase-compensation filter to cascade with the 10th-order Butterworth filter to improve performance over that of the 4th-order phase-compensated filter. Start with the given 4th-order all-pass filter, i.e., add an additional 2nd-order all-pass. Select the 2nd-order all-pass filter parameters to yield improved group delay performance. You may use as a starting point the MATLAB m-file *F8_45dat.m* on the accompanying disk. Plot graphs similar to **Figures 8.43** through **8.47** that include the results of the 6th-order phase compensation. Also plot a graph of the poles and zeros of the 6th-order phase-compensated 10th-order Butterworth filter.

CHAPTER 9

FREQUENCY TRANSFORMATIONS

The standard procedure for analog filter design is to first design (i.e., obtain the transfer function) a lowpass filter with either ω_c or ω_p normalized to unity, and then, by means of a frequency transformation (the topic of this chapter), obtain the transfer function of the desired filter, be that lowpass, highpass, bandpass or bandstop, with desired critical frequency values. The lowpass filter that is initially designed, before performing the frequency transformation, is referred to as the lowpass *prototype* filter. This procedure greatly simplifies and unifies filter design, as then technical detail is concentrated in the prototype filter design, as has been done in earlier chapters of this book.

Since it is usually the case that the desired performance characteristics (the design specifications) are for the final, or actual, filter, it is necessary to determine what the corresponding characteristics or specifications need to be for the lowpass prototype filter. For example, if a Butterworth bandpass filter is specified with a center frequency of 10 kHz, a 3 dB bandwidth of 1 kHz, and a 60 dB bandwidth of 5 kHz, it is necessary to determine the minimum order of the lowpass prototype filter so that the design will satisfy the given specifications after the frequency transformation is applied. Therefore, this chapter is concerned not only with frequency transformation methods, but also in the analysis of what the implications of such a transformation are on the magnitude frequency response, the phase response, phase delay, group delay, unit impulse response, etc.

9.1 LOWPASS-TO-LOWPASS TRANSFORMATION

Lowpass-to-lowpass transformations have already been presented in **Section 2.16 Frequency Scaling**. While frequency scaling, as presented in **Section 2.16**, is not restricted to lowpass transfer functions, it does, of course, include lowpass transfer functions. A lowpass-to-lowpass transformation is simply, since the starting point here is the lowpass prototype filter, frequency scaling. See **Section 2.16** for details on how this frequency transformation affects the magnitude frequency response, the phase response, phase delay, group delay, the unit impulse response, and the unit step

response. Also note that the equations for the minimum order required by any of the classical filters presented in **Chapter 3** through **Chapter 6** may be applied directly.

Example 9.1

Suppose

$$H_{LPP}(s) = \frac{1}{s^3 + 2s^2 + 2s + 1} ,$$

where $H_{LPP}(s)$ denotes the lowpass prototype transfer function. It is noted that this is a 3rd-order Butterworth filter with $\omega_c = 1$. See **Example 3.3** on page 121. If it is desired that the transformed lowpass filter have $\omega_c = 1000$, then k_f , the frequency scaling factor, should be 1000. Then,

$$\begin{aligned} H_{LP}(s) = H_{LPP}(s) \Big|_{s \rightarrow s/1000} &= \frac{1}{(s/1000)^3 + 2(s/1000)^2 + 2(s/1000) + 1} \\ &= \frac{10^9}{s^3 + 2 \times 10^3 s^2 + 2 \times 10^6 s + 10^9} . \end{aligned}$$

□

Poles and Zeros

Since k_f is real and positive, the magnitudes of the poles and zeros are scaled but not the phase angles. A pole/zero plot of $H_{LP}(s)$ will be identical to that of $H_{LPP}(s)$ except that the axes will be scaled by k_f .

Magnitude Frequency Response and Phase Response

The magnitude frequency response and phase response may be summarized as follows:

$$|H_{LP}(j\omega)| = |H_{LPP}(j\omega/k_f)| , \tag{9.1}$$

and

$$\angle H_{LP}(j\omega) = \angle H_{LPP}(j\omega/k_f) . \tag{9.2}$$

Therefore, plots of the transformed (frequency-scaled) magnitude frequency response, (9.1), and phase response, (9.2), are identical to those obtained from the lowpass prototype except for the scaling of the frequency axes by k_f .

Determination of Minimum Order

Since the equations used to determine the minimum order required to meet given specifications in earlier chapters depend upon A_p , A_s , and the ratio ω_s/ω_p , they may be used directly to determine the minimum order required for the frequency-scaled lowpass filter. For example, (3.15) may be used to determine the minimum

order required for a lowpass Butterworth filter, regardless of whether it is a prototype filter with $\omega_p = 1$, or the desired lowpass filter after frequency scaling.

Filter Selectivity

Since the *Filter Selectivity* equations derived in earlier chapters did not start with an assumption that ω_c was normalized to unity, they may be used directly to determine F_S for the frequency-scaled lowpass filter. For example, (3.7) may be used to determine F_S for a lowpass Butterworth filter, regardless of whether it is a prototype filter with $\omega_c = 1$, or the desired lowpass filter after frequency scaling.

Shaping Factor

Since the *Shaping Factor* equations derived in earlier chapters did not start with any assumptions about normalized frequencies, and are the ratio of bandwidths, they may be used directly to determine S_a^b for the frequency-scaled lowpass filter. For example, (4.12) may be used to determine S_a^b for a lowpass Chebyshev Type I filter, regardless of whether it is a prototype filter with $\omega_p = 1$, or the desired lowpass filter after frequency scaling.

Phase Delay and Group Delay

Phase delay and group delay for a frequency-scaled lowpass filter, as related to the prototype with ω_p generally normalized to unity, may be expressed as follows:

$$t_{pd}^{(LP)}(\omega) = t_{pd}^{(LPP)}(\omega/k_f)/k_f \quad (9.3)$$

and

$$t_{gd}^{(LP)}(\omega) = t_{gd}^{(LPP)}(\omega/k_f)/k_f \quad (9.4)$$

Therefore, plots of the transformed (frequency-scaled) phase delay, (9.3), and group delay, (9.4), not only have the frequency axes scaled by k_f , but the amplitude axes are scaled by $1/k_f$.

Time-Domain Response

The unit impulse response of a lowpass-to-lowpass transformed filter, as related to the prototype with ω_p generally normalized to unity, may be expressed as follows:

$$h_{LP}(t) = k_f h_{LPP}(k_f t) \quad (9.5)$$

Note that the transformed (frequency-scaled) unit impulse response, as shown in (9.5), is a time-scaled version of the prototype unit impulse response and is also amplitude scaled by k_f . If $k_f > 1$, then $h_{LP}(t)$ will be greater in amplitude and time-compressed, compared to $h_{LPP}(t)$.

The unit step response may be summarized as follows:

$$h_u^{(LP)}(t) = h_u^{(LPP)}(k_f t) \quad (9.6)$$

Note that the transformed unit step response, as shown in (9.6), is a time-scaled version of the prototype unit step response with no corresponding amplitude scaling.

9.2 LOWPASS-TO-HIGHPASS TRANSFORMATION

The lowpass-to-highpass transformation is stated as follows: replace every s in $H_{LPP}(s)$ by k_f/s , where, again, k_f is the frequency scaling factor. Applying this to the general transfer function expressed in (2.39), treating it as $H_{LPP}(s)$:

$$\begin{aligned} H_{HP}(s) = H_{LPP}(s) \Big|_{s \rightarrow k_f/s} &= \frac{\sum_{k=0}^M b_k (k_f/s)^k}{\sum_{k=0}^N a_k (k_f/s)^k} \\ &= \frac{(s/k_f)^{N-M} \sum_{k=0}^M b_k (s/k_f)^{M-k}}{\sum_{k=0}^N a_k (s/k_f)^{N-k}} \end{aligned} \quad (9.7)$$

Equation (9.7), while general, does not readily reveal the effects of this transformation. Consider the case where the prototype filter is a Butterworth design. In that case, $M = 0$ and there is symmetry to the a_k coefficients: $a_0 = a_N$, $a_1 = a_{N-1}$, etc. Therefore, for the Butterworth case, (9.7) may be expressed as follows:

$$H_{HP}(s) = \frac{(s/k_f)^N}{\sum_{k=0}^N a_k (s/k_f)^k} = \frac{s^N}{k_f^N} H_{LPP}(s/k_f) \quad (9.8)$$

Example 9.2

Suppose

$$H_{LPP}(s) = \frac{1}{s^3 + 2s^2 + 2s + 1} \quad ,$$

where $H_{LPP}(s)$ denotes a 3rd-order Butterworth lowpass prototype transfer function. If it is desired to transform this into a highpass filter with $k_f = 1000$, then,

$$\begin{aligned}
 H_{HP}(s) &= H_{LPP}(s) \Big|_{s \rightarrow 1000/s} = \frac{1}{(1000/s)^3 + 2(1000/s)^2 + 2(1000/s) + 1} \\
 &= \frac{s^3}{s^3 + 2 \times 10^3 s^2 + 2 \times 10^6 s + 10^9} .
 \end{aligned}$$

Compare the results of this example with that of *Example 9.1*, and note the difference. Note that this example is a direct application of (9.8). Also note that

$$|H_{HP}(j\omega)| = \frac{\omega^3}{\sqrt{(10^9 - 2000\omega^2)^2 + \omega^2(2 \times 10^6 - \omega^2)^2}} .$$

Therefore, $|H_{HP}(0)| = 0$, $|H_{HP}(\infty)| = 1$, and $|H_{HP}(j1000)| = 1/\sqrt{2}$. So $H_{HP}(s)$ is clearly a highpass filter with $\omega_c = 1000$. \square

Poles and Zeros

If $H_{LPP}(s)$ is shown in factored form:

$$H_{LPP}(s) = \frac{K \prod_{k=1}^M (s + \gamma_k)}{\prod_{k=1}^N (s + \mu_k)} ,$$

where the zeros, $-\gamma_k$, and the poles, $-\mu_k$, may be, and in general are, complex, then the lowpass-to-highpass transformed transfer function may be expressed as follows:

$$H_{HP}(s) = \frac{K \prod_{k=1}^M (k_f/s + \gamma_k)}{\prod_{k=1}^N (k_f/s + \mu_k)} = \frac{K_{HP} s^{N-M} \prod_{k=1}^M (s + k_f/\gamma_k)}{\prod_{k=1}^N (s + k_f/\mu_k)} , \tag{9.9}$$

where

$$K_{HP} = \frac{K \prod_{k=1}^M \gamma_k}{\prod_{k=1}^N \mu_k} .$$

In the form of (9.9), it is apparent that lowpass-to-highpass transformation has resulted in the following:

- (a) the poles and zeros are inverted and then scaled by k_f ,
- (b) the order of the numerator becomes equal to that of the denominator (N),

$$(c) H_{HP}(0) = H_{LPP}(\infty),$$

$$(d) H_{HP}(\infty) = K_{HP} = H_{LPP}(0),$$

and (e) $H_{HP}(jk_f) = H_{LPP}^*(j)$.

Magnitude Frequency Response and Phase Response

The magnitude frequency response and phase response may be summarized as follows:

$$|H_{HP}(j\omega)| = |H_{LPP}(k_f/j\omega)| \quad , \quad (9.10)$$

and

$$\angle H_{HP}(j\omega) = \angle H_{LPP}(k_f/j\omega) \quad . \quad (9.11)$$

Note that a plot of the highpass transformed frequency-scaled magnitude frequency response, as expressed in (9.10), is a highly nonlinear transformation of the frequency axis. As noted above,

$$H_{HP}(0) = 0, \text{ for the case where } N > M, \text{ and note also}$$

$$H_{HP}(0) = K, \text{ if } N = M.$$

See **Figures 5.1** and **6.10** for examples of lowpass prototype filters where $H_{LPP}(\infty) = K \neq 0$. In such cases, K will typically be very small, such as, for example, 0.0001 (–80 dB).

Also note that

$$H_{HP}(\infty) = K_{HP} = H_{LPP}(0), \text{ and}$$

$$|H_{HP}(jk_f)| = |H_{LPP}(j)|.$$

This nonlinear transformation of the frequency axis significantly changes the appearance of a magnitude frequency response plot. For example, the rippling in the passband of a Chebyshev Type I lowpass prototype filter will have peaks and valleys occurring in the frequency range from 0 to unity. After lowpass-to-highpass transformation, the peaks and valleys in the highpass passband will be greatly stretched out to occupy the frequency range of the passband from k_f to ∞ . More specifically, see the following example.

Example 9.3

Given a 5th-order Chebyshev Type I lowpass prototype filter with 1 dB of passband ripple and $\omega_p = 1$, then, from (4.5), the frequencies of the passband peaks are 0, 0.5878 rad/s, and 0.9511 rad/s. From (4.6) the frequencies of the valleys are 0.309 rad/s and 0.809 rad/s. Given that a lowpass-to-highpass transformation is applied to this lowpass prototype, with $k_f = 1000$, then, from (9.10), the ω_p for the highpass filter will be 1000 rad/s, the frequencies of the passband peaks for the highpass filter are 1051.4 rad/s, 1701.3 rad/s, and ∞ . The frequencies of the

highpass passband valleys are 1236.1 rad/s and 3236.2 rad/s. Note that the peak at 0 in the lowpass prototype becomes a peak at infinity for the highpass filter. □

Note that a plot of the highpass transformed frequency-scaled phase response, as expressed in (9.11), is also a highly nonlinear transformation of the frequency axis. As can readily be seen,

$$\angle H_{HP}(0) = -\angle H_{LPP}(\infty),$$

$$\angle H_{HP}(\infty) = -\angle H_{LPP}(0),$$

and $\angle H_{HP}(jk_f) = -\angle H_{LPP}(j)$.

It should be noted however, that a plot of the phase response may have an additional constant phase offset that is an integer multiple of 2π (or 360°) without introducing any inaccuracies. That is, $\angle H_{HP}(j\omega) = \angle H_{LPP}(k_f/j\omega) + m2\pi$ is equally valid. The main observation is that phase values that occur at high frequencies for the lowpass prototype occur at low frequencies for the highpass filter and vice versa.

Determination of Minimum Order

The equations for determination of minimum order for lowpass filters depend on A_p , A_s , and ω_s/ω_p . Note, however, that under the lowpass-to-highpass transformation, ω_p maps to $-k_f/\omega_p$, $-\omega_p$ maps to k_f/ω_p , ω_s maps to $-k_f/\omega_s$, and $-\omega_s$ maps to k_f/ω_s . Note that k_f/ω_p is ω_{pHP} , the passband edge of the highpass filter, and that k_f/ω_s is ω_{sHP} , the stopband edge of the highpass filter. Therefore,

$$\frac{\omega_{pHP}}{\omega_{sHP}} = \frac{-k_f/\omega_p}{-k_f/\omega_s} = \frac{k_f\omega_s}{k_f\omega_p} = \frac{\omega_s}{\omega_p}.$$

Therefore, the required minimum order to meet design specifications for the lowpass prototype may be determined by using the existing equations in previous chapters, replacing ω_s/ω_p by $\omega_{pHP}/\omega_{sHP}$. For example, to determine the minimum required order for the lowpass prototype of a Butterworth highpass filter, (3.15) may be modified as follows:

$$N_{LPP} = \left\lceil \frac{\log \left(\frac{(10^{A_s/10} - 1) / (10^{A_p/10} - 1)^{1/2}}{\log(\omega_{pHP} / \omega_{sHP})} \right) \right\rceil.$$

Filter Selectivity

The definition of *Filter Selectivity* is defined for lowpass filters in (2.37). In order for F_s to be a positive number for a highpass filter, it is defined with the opposite polarity:

$$F_S^{(HP)} = \left. \frac{d|H_{HP}(j\omega)|}{d\omega} \right|_{\omega = \omega_c}, \quad (9.12)$$

where ω_c is the 3 dB cutoff frequency. Note that (9.12) can be related to the *Filter Selectivity* of the lowpass prototype as follows:

$$F_S^{(HP)} = \left. \frac{d|H_{HP}(j\omega)|}{d\omega} \right|_{\omega = \omega_c} = \left. \frac{d|H_{LPP}(k_f/j\omega)|}{d\omega} \right|_{\omega = \omega_c}. \quad (9.13)$$

By making use of a change of variable, specifically $x = -k_f/\omega$, it follows that

$$F_S^{(HP)} = \frac{k_f}{\omega_c^2} F_S^{(LPP)}, \quad (9.14)$$

where $F_S^{(LPP)}$ is evaluated at $\omega_c^{(LPP)} = k_f/\omega_c$, and ω_c is the 3 dB frequency for the highpass filter. If the lowpass prototype filter has an ω_c normalized to unity, then (9.14) simplifies somewhat to $F_S^{(HP)} = F_S^{(LPP)}/\omega_c$.

Shaping Factor

Since the passband width of a highpass filter is infinite, the definition of *Shaping Factor*, as given in (2.38) for a lowpass filter, is not useful for a highpass filter. However, from the basic lowpass-to-highpass transformation, it follows that:

$$|H_{LPP}(j\omega_a)| = |H_{HP}(jk_f/\omega_a)|$$

and

$$|H_{LPP}(j\omega_b)| = |H_{HP}(jk_f/\omega_b)|,$$

where $\omega_a = BW_a$ and $\omega_b = BW_b$ for the lowpass prototype filter. Denoting S_{LPP} and S_{HP} the *Shaping Factor* for the lowpass prototype filter and the highpass filter, respectively, it follows that

$$S_{LPP} = \frac{BW_b}{BW_a} = \frac{\omega_b}{\omega_a} = \frac{k_f/\omega_a}{k_f/\omega_b} = \frac{\omega_b}{\omega_a} \triangleq S_{HP}. \quad (9.15)$$

That is, the highpass *Shaping Factor* is numerically identical to the lowpass prototype *Shaping Factor*, when defined as in (9.15). Note that, as defined in (9.15), the *Shaping Factor* for a highpass filter is the ratio of the stopband/transition band bandwidth for an attenuation of a in dB (e.g., 3 dB) over the stopband/transition band bandwidth for an attenuation of b in dB (e.g., 60 dB).

Phase Delay and Group Delay

Phase delay is defined by (2.80). Phase delay for a lowpass-to-highpass transformed transfer function may be expressed as follows:

$$t_{pd}^{(HP)}(\omega) = - \frac{\angle H_{HP}(j\omega)}{\omega} = - \frac{\angle H_{LPP}(k_f/j\omega) + m 2 \pi}{\omega}$$

Therefore, it can be seen that

$$t_{pd}^{(HP)}(\infty) = 0,$$

$$t_{pd}^{(HP)}(k_f) = -t_{pd}^{(LPP)}(1) / k_f - m 2 \pi / k_f,$$

and $t_{pd}^{(HP)}(\omega)$ may be very large as ω approaches zero unless $\angle H_{HP}(\omega)$ approaches zero as ω approaches zero, which is often not the case for a highpass filter. Note that the phase is always zero at $\omega = 0$ due to the odd symmetry of the phase response, but that the phase may well not be zero as the discontinuity is approached.¹ Note also that the very large phase delay as zero frequency is approached, which is often the case for highpass filters, occurs when the magnitude response of the filter is very small, and therefore the large phase delay has no practical significance.

Group delay is defined by (2.81). Group delay for a lowpass-to-highpass transformed transfer function may be expressed as follows:

$$t_{gd}^{(HP)}(\omega) = - \frac{d \angle H_{HP}(j\omega)}{d \omega} = - \frac{d \angle H_{LPP}(k_f/j\omega)}{d \omega} \tag{9.16}$$

Making use of an appropriate change of variable (specifically, $jx = k_f/j\omega$), and the even symmetry property of group delay, (9.16) may be expressed as follows:

$$t_{gd}^{(HP)}(k_f/\omega) = \frac{\omega^2}{k_f} t_{gd}^{(LPP)}(\omega) \tag{9.17}$$

From (9.17), the following may be observed:

$$t_{gd}^{(HP)}(\infty) = 0,$$

$$t_{gd}^{(HP)}(k_f) = \frac{1}{k_f} t_{gd}^{(LPP)}(1),$$

$$t_{gd}^{(HP)}(k_f/2) = \frac{4}{k_f} t_{gd}^{(LPP)}(2),$$

¹ Consider, for example, a simple series R-C circuit, where an input voltage is applied across the series combination and the output voltage is taken across the resistor. As the frequency is lowered, approaching zero, the phase approaches $\pi/2$ (or 90°), but, due to odd symmetry, there is a discontinuity at zero frequency and the phase at zero may be taken as being zero.

$$t_{gd}^{(HP)}(k_f/10) = \frac{100}{k_f} t_{gd}^{(LPP)}(10),$$

etc.

Example 9.4

Suppose

$$H_{LPP}(s) = \frac{1}{s^3 + 2s^2 + 2s + 1},$$

where $H_{LPP}(s)$ denotes a 3rd-order Butterworth lowpass prototype transfer function. It follows that

$$\angle H_{LPP}(j\omega) = -\tan^{-1} \left[\frac{\omega(2 - \omega^2)}{1 - 2\omega^2} \right],$$

from which

$$t_{gd}^{(LPP)}(\omega) = -\frac{d \angle H_{LPP}(j\omega)}{d\omega} = \frac{2\omega^4 + \omega^2 + 2}{\omega^6 + 1}.$$

If $k_f = 1$, then $t_{gd}^{(HP)}(\omega)$ may be found by use of fundamental definitions, or, more conveniently, by application of (9.17):

$$t_{gd}^{(HP)}(1/\omega) = \omega^2 t_{gd}^{(LPP)}(\omega) = \omega^2 \frac{2\omega^4 + \omega^2 + 2}{\omega^6 + 1}.$$

Therefore, by a change of variable,

$$t_{gd}^{(HP)}(\omega) = \frac{2\omega^4 + \omega^2 + 2}{\omega^6 + 1} = t_{gd}^{(LPP)}(\omega).$$

If $k_f = 1000$, then by application of (9.17):

$$t_{gd}^{(HP)}(\omega) = 10^3 \frac{2\omega^4 + 10^6\omega^2 + 2 \times 10^{12}}{\omega^6 + 10^{18}}.$$

□

Time-Domain Response

From (9.9):

$$H_{HP}(s) = K_{HP} - \frac{K_{HP} \left[\prod_{k=1}^N (s + k_f/\mu_k) - s^{N-M} \prod_{k=1}^M (s + k_f/\gamma_k) \right]}{\prod_{k=1}^N (s + k_f/\mu_k)}. \quad (9.18)$$

Equation (9.18) may be expressed as follows:

$$H_{HP}(s) = \begin{cases} K_{HP} - \frac{K_{HP} \prod_{k=1}^{N-1} (s + \beta_k)}{\prod_{k=1}^N (s + k_f/\mu_k)} , & M > 0 \\ K_{HP} - \frac{K_{HP} \alpha \prod_{k=1}^{N-1} (s + \beta_k)}{\prod_{k=1}^N (s + k_f/\mu_k)} , & M = 0 \end{cases} , \quad (9.19)$$

where α is the coefficient of the s^{N-1} term in the denominator. Note that (9.18) and (9.19) may be expressed as

$$H_{HP}(s) = K_{HP} - H_{LP}(s) . \quad (9.20)$$

That is, the second term on the right side of (9.18) or (9.19) is recognized as a lowpass transfer function. Note from (9.20), with s replaced by $j\omega$, that the high-frequency gain is K_{HP} , and that $|H_{LP}(j\omega)|$ is a lowpass response that, as its response becomes increasingly small with increasing ω , that the overall response, $|H_{HP}(j\omega)|$, becomes increasingly close to K_{HP} . The inverse Laplace transform of (9.20) may be therefore stated as follows:

$$h_{HP}(t) = K_{HP} \delta(t) - h_{LP}(t) . \quad (9.21)$$

Therefore, the unit impulse response of a highpass filter will have an impulse at $t = 0$, and also a lowpass impulse response component.

Recall that the unit step response is the integral of the unit impulse response. Therefore,

$$h_u^{(HP)}(t) = K_{HP} - \int_0^t h_{LP}(\tau) d\tau = K_{HP} - h_u^{(LP)}(t) . \quad (9.22)$$

Example 9.5

Suppose

$$H_{LPP}(s) = \frac{1}{s^3 + 2s^2 + 2s + 1} ,$$

where $H_{LPP}(s)$ denotes a 3rd-order Butterworth lowpass prototype transfer function. This transfer function may be expressed as follows:

$$H_{LPP}(s) = \frac{1}{s + 1} - \frac{s}{s^2 + s + 1} .$$

From the above,

$$h_{LPP}(t) = e^{-t} [1 - \cos(\{\sqrt{3}/2\}t) + (1/\sqrt{3}) \sin(\{\sqrt{3}/2\}t)] u(t) .$$

Also, for $k_f = 1$,

$$H_{HP}(s) = \frac{s^3}{s^3 + 2s^2 + 2s + 1} = 1 - \frac{1}{s + 1} - \frac{s}{s^2 + s + 1} .$$

It follows that

$$h_{HP}(t) = \delta(t) - e^{-t} [1 + \cos(\{\sqrt{3}/2\}t) - (1/\sqrt{3}) \sin(\{\sqrt{3}/2\}t)] u(t) .$$

Note that whereas $h_{LPP}(0) = 0$, $h_{HP}(0) = \delta(t) - 2$. Therefore, there is a discontinuity in the unit impulse response of the highpass filter at $t = 0$, even if the impulse is not considered. This is typical for a highpass filter. □

Example 9.6

Suppose the lowpass prototype filter is a 10th-order Chebyshev Type I filter with 1 dB of passband ripple. Suppose a high-pass filter is obtained from this prototype with $k_f = 1000$. Whereas the poles of the lowpass prototype filter are as follows:

$$\begin{aligned} & -0.1415 \pm j 0.1580, \\ & -0.1277 \pm j 0.4586, \\ & -0.1013 \pm j 0.7143, \\ & -0.0650 \pm j 0.9001, \end{aligned}$$

and the poles of the highpass filter are:

$$\begin{aligned} & -3144.7 \pm j 3511.7, \\ & -563.3 \pm j 2023.6, \\ & -194.6 \pm j 1372.3, \\ & -79.9 \pm j 1105.2, \end{aligned}$$

and

$$-22.5 \pm j 1001.7.$$

The highpass filter also has 10 zeros at the origin of the s plane. The transfer function of the highpass filter is as follows:

$$H_{HP}(s) = \frac{0.8913 s^{10}}{s^{10} + 8,010 s^9 + 4.237 \times 10^7 s^8 + 1.057 \times 10^{11} s^7 + 2.890 \times 10^{14} s^6 + 3.745 \times 10^{17} s^5 + 6.923 \times 10^{20} s^4 + 4.894 \times 10^{23} s^3 + 6.779 \times 10^{26} s^2 + 2.127 \times 10^{29} s + 2.322 \times 10^{32}} .$$

The magnitude frequency response of the highpass filter is shown in **Figure 9.1**. Compared with the plot of the lowpass prototype shown in **Figure 4.1**, it is noted that **Figure 9.1** appears as a mirror image, even though the frequency axis is

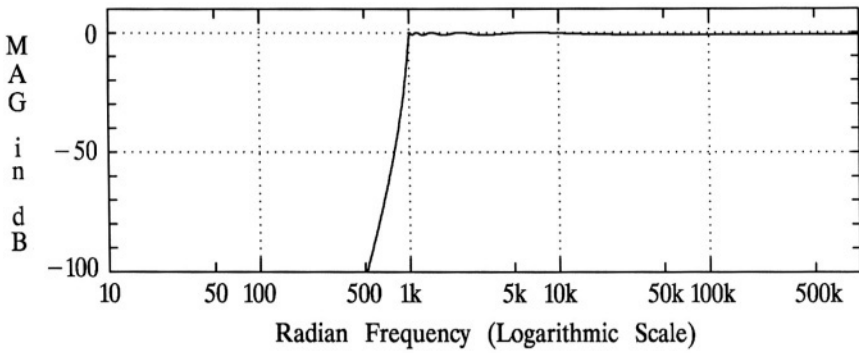


Figure 9.1 The magnitude frequency response of the Chebyshev Type I highpass filter in *Example 9.6*: the order is 10, the passband ripple is 1 dB, and $\omega_p = 1000$.

logarithmic. The “mirror image” is somewhat misleading, however, because of the nonlinear mapping of the lowpass-to-highpass transformation.

It can be shown that $F_s^{(HP)}$ for this filter, as defined in (9.12), and measured from the data plotted in **Figure 9.1**, adheres to (9.14). Also, it can be shown that S_{HP} for this filter, as defined in (9.15), and measured from the data plotted in **Figure 9.1**, adheres to (9.15), for example, when $a = 6$ dB and $b = 60$ dB.

A more detailed plot of the passband response is shown in **Figure 9.2**. Note that the peak and valley frequencies in the passband of the highpass filter are related to (4.5) and (4.6); due to the inverse relationship between the lowpass prototype and the highpass filter in the s domain:

$$\omega_k^{(peak)} = \omega_p / \cos[(2k - 1)\pi / (2N)] , \quad k = 1, 2, 3, \dots, N_p,$$

where $N_p = (N + 1) / 2$ if N is odd, and $N_p = N / 2$ if N is even,

$$\omega_k^{(valley)} = \omega_p / \cos[(k\pi) / N] , \quad k = 1, 2, 3, \dots, N_v,$$

where $N_v = (N - 1) / 2$ if N is odd, and $N_v = N / 2$ if N is even. The relationship between ω_c and ω_p for the Chebyshev Type I highpass filter is related to (4.7):

$$\omega_c = \omega_p / \cosh[(1/N) \cosh^{-1}(1/\epsilon)] .$$

The phase response is shown in **Figure 9.3**. Note that the phase asymptotically approaches zero as the frequency approaches infinity. This can be readily observed from the transfer function. It should, however, be noted that computer computations will likely differ from that shown in **Figure 9.3** by a multiple of 2π . The reason for this is that the principal-value phase angles are likely computed, and perhaps then followed by an unwrapping operation in an attempt to obtain the complete phase. The

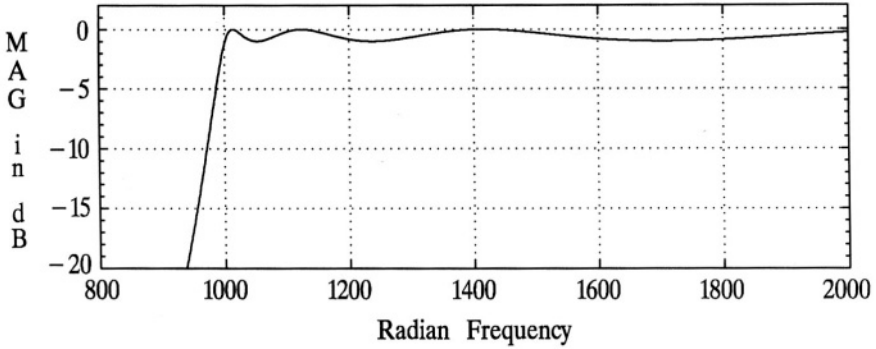


Figure 9.2 A more detailed plot of the passband magnitude frequency response of the Chebyshev Type I highpass filter in *Example 9.6*: the order is 10, the passband ripple is 1 dB, and $\omega_p = 1000$.

unwrapping operation smooths the phase response eliminating discontinuities, but likely lacks data on any true phase reference.

The phase delay is shown in **Figure 9.4**. Based on the definition of phase delay, as given in (2.80), if the phase is positive (the output leading the input) then the phase delay will be negative. A negative phase delay may suggest a time advance, but this would be misleading. Phase delay is defined strictly in terms of phase shifts. Another observation is that an accurate plot of the phase delay requires a *complete*

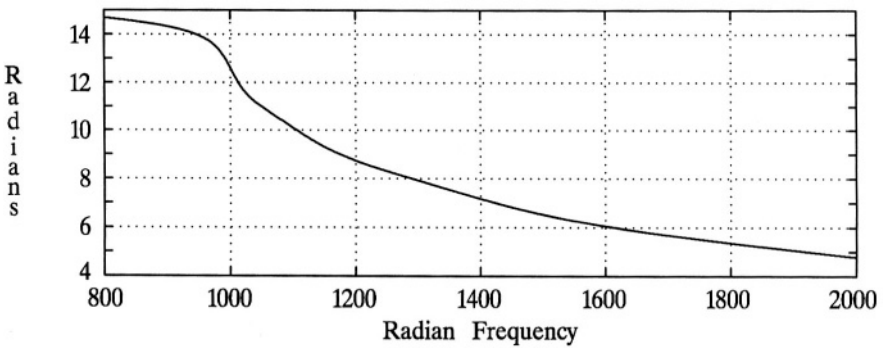


Figure 9.3 The phase response of the Chebyshev Type I highpass filter in *Example 9.6*: the order is 10, the passband ripple is 1 dB, and $\omega_p = 1000$.

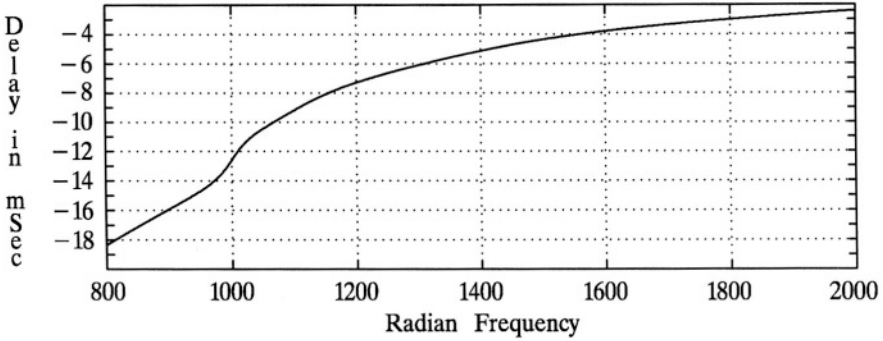


Figure 9.4 The phase delay response of the Chebyshev Type I highpass filter in *Example 9.6*: the order is 10, the passband ripple is 1 dB, and $\omega_p = 1000$.

phase response (see comments above on **Figure 9.3** and also on page 277). Note in **Figure 9.4** that the phase delay asymptotically approaches zero as frequency approaches infinity, conforming to the theory briefly described on page 279. Also, as noted on page 279,

$$t_{pd}^{(HP)}(k_f) = -t_{pd}^{(LPP)}(1) / k_f - m 2 \pi / k_f .$$

By comparing **Figure 4.6** for $N = 10$ with **Figure 9.4** it can be shown that the above equation is satisfied.

The group delay is shown in **Figure 9.5**. It can be observed in **Figure 9.5**, compared with **Figure 4.7**, that, as predicted,

$$t_{gd}^{(HP)}(k_f) = \frac{1}{k_f} t_{gd}^{(LPP)}(1) .$$

The unit impulse response, excluding the impulse at $t = 0$, is shown in **Figure 9.6**. The large negative-value discontinuity at $t = 0^+$ observed in the figure is suggested by (9.21) and by *Example 9.5*.

The unit step response is shown in **Figure 9.7**. Note that in (9.9), for this example, $M = N = 10$ and all $\gamma_k = \infty$. By comparing (9.9) with the transfer function for this example it is clear that $K_{HP} = 0.8913$. By referring to (9.22), it is clear that, in general for a highpass filter, $h_u^{(HP)}(0) = K_{HP}$, and therefore, in this example, $h_u^{(HP)}(0) = 0.8913$, which is what the data plotted in **Figure 9.7** has for the first data point. Since $h_u^{(HP)}(t)$ is the integral of $h_{HP}(t)$, the peaks and minimum-points of $h_u^{(HP)}(t)$ occur at the same time-instants as the zero-crossings of $h_{HP}(t)$, as can be observed by comparing **Figure 9.7** with **Figure 9.6**. Also note that since this

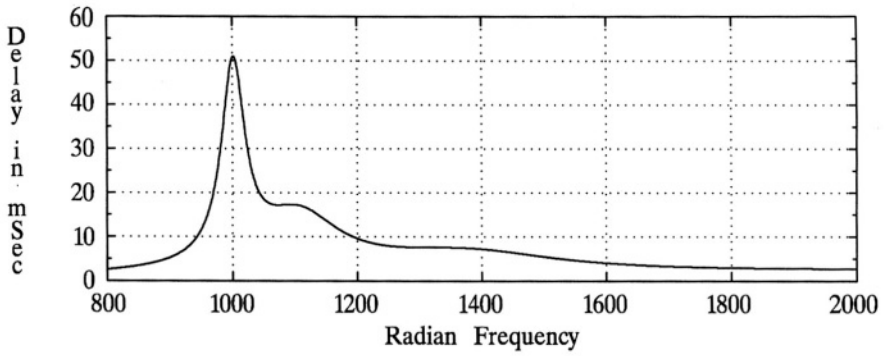


Figure 9.5 The group delay response of the Chebyshev Type I highpass filter in *Example 9.6*: the order is 10, the passband ripple is 1 dB, and $\omega_p = 1000$.

is a highpass filter, the DC gain of the filter is zero, and therefore the unit step response achieves a steady state response of zero, not unity, or near unity, as was the case with the lowpass filters presented in previous chapters. □

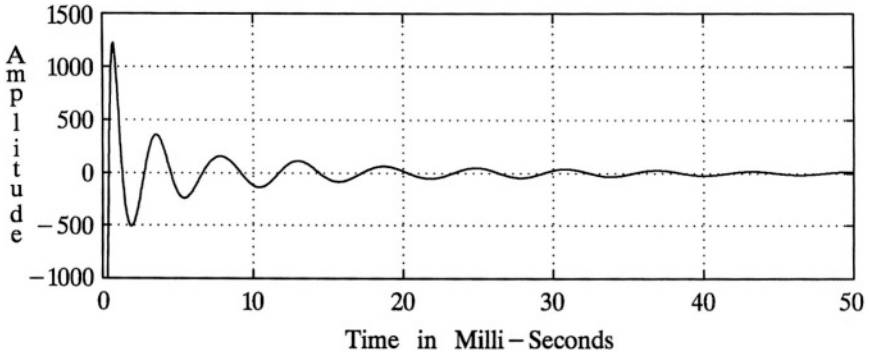


Figure 9.6 The unit impulse response of the Chebyshev Type I highpass filter in *Example 9.6*: the order is 10, the passband ripple is 1 dB, and $\omega_p = 1000$.

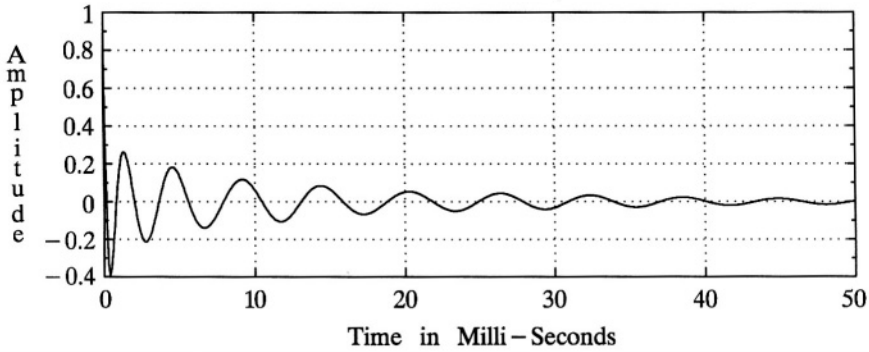


Figure 9.7 The unit step response of the Chebyshev Type I highpass filter in *Example 9.6*: the order is 10, the passband ripple is 1 dB, and $\omega_p = 1000$.

9.3 LOWPASS-TO-BANDPASS TRANSFORMATION

The lowpass-to-bandpass transformation is stated as follows: replace every s in $H_{LPP}(s)$ by $(s^2 + \omega_o^2)/(B_p s)$, where ω_o is the “center frequency” of the bandpass filter frequency response in rad/s and B_p is the passband bandwidth, i.e., $B_p = \omega_{p2} - \omega_{p1}$, where ω_{p2} is the upper edge of the passband where the attenuation is A_p and ω_{p1} is the lower edge of the passband. Note that ω_p must be unity for the lowpass prototype.

Example 9.7

Suppose

$$H_{LPP}(s) = \frac{1}{s^3 + 2s^2 + 2s + 1} ,$$

where $H_{LPP}(s)$ denotes a 3rd-order Butterworth lowpass prototype transfer function. If it is desired to transform this into a bandpass filter with arbitrary ω_o and B_p :

$$\begin{aligned} H_{BP}(s) &= H_{LPP}(s) \Big|_{s \rightarrow \frac{s^2 + \omega_o^2}{B_p s}} \\ &= \frac{1}{\left(\frac{s^2 + \omega_o^2}{B_p s}\right)^3 + 2\left(\frac{s^2 + \omega_o^2}{B_p s}\right)^2 + 2\left(\frac{s^2 + \omega_o^2}{B_p s}\right) + 1} \end{aligned}$$

$$= \frac{B_p^3 s^3}{s^6 + 2 B_p s^5 + (3 \omega_o^2 + 2 B_p^2) s^4 + B_p (4 \omega_o^2 + B_p^2) s^3 + \omega_o^2 (3 \omega_o^2 + 2 B_p^2) s^2 + 2 B_p \omega_o^4 s + \omega_o^6} .$$

By properly factoring the 6th-order denominator of $H_{BP}(s)$:

$$\begin{aligned} H_{BP}(s) &= \frac{k_1}{s^3 + \alpha_1 s^2 + \alpha_2 s + \alpha_3} \times \frac{k_2 s^3}{s^3 + \beta_1 s^2 + \beta_2 s + \beta_3} \\ &= H_{LP}(s) H_{HP}(s) . \end{aligned}$$

Note the following observations: (1) The order of the bandpass transfer function is twice that of the lowpass prototype and therefore will always be even, (2) The bandpass transfer function may be factored into the product of a lowpass transfer function and a highpass transfer function, and (3) For this example, $|H_{BP}(0)| = |H_{BP}(\infty)| = 0$, which is typical for a bandpass filter. If the lowpass prototype filter has rippling in the stopband, such as a Chebyshev Type II or elliptic filter, then $|H_{BP}(0)|$ and $|H_{BP}(\infty)|$ may not be zero, but will be very small values. \square

Poles and Zeros

If $H_{LPP}(s)$ is shown in factored form:

$$H_{LPP}(s) = \frac{K \prod_{k=1}^M (s + \gamma_k)}{\prod_{k=1}^N (s + \mu_k)} ,$$

where the zeros, $-\gamma_k$, and the poles, $-\mu_k$, may be, and in general are, complex, then consider the results of the lowpass-to-bandpass transformation.

Case I

Consider the case where μ_k is real. To determine the corresponding bandpass filter poles, let

$$\frac{s^2 + \omega_o^2}{B_p s} + \mu_k = 0 .$$

It follows that

$$s^2 + B_p \mu_k s + \omega_o^2 = 0 ,$$

or the two poles corresponding to the one lowpass prototype pole, $-\mu_k$, are

$$s_{1,2} = -\frac{B_p \mu_k}{2} \pm \frac{1}{2} \sqrt{B_p^2 \mu_k^2 - 4 \omega_o^2} .$$

Theoretically, $s_{1,2}$ could be real ($B_p^2 \mu_k^2 > 4 \omega_o^2$), however such will not be the case for practical bandpass filters. Therefore, consider $4 \omega_o^2 > B_p^2 \mu_k^2$:

$$s_{1,2} = -\frac{B_p \mu_k}{2} \pm \frac{j}{2} \sqrt{4 \omega_o^2 - B_p^2 \mu_k^2} .$$

Also note that

$$\frac{s^2 + \omega_o^2}{B_p s} + \mu_k = \infty$$

defines a transfer function zero: a zero is introduced at $s = 0$. Therefore, each *real* pole in $H_{LPP}(s)$ will yield a pair of complex conjugate poles in $H_{BP}(s)$, as developed above, and one zero at the origin. Note that if $4 \omega_o^2 \gg B_p^2 \mu_k^2$, then the imaginary part of those complex conjugate poles will be approximately $\pm \omega_o$.

Example 9.8

Suppose $\mu_k = 1$, $\omega_o = 5$, and $B_p = 2$. It follows that

$$s_{1,2} = -1 \pm j\sqrt{24} \cong -1 \pm j4.90$$

will be the resultant poles, and there will one zero at $s = 0$. □

Case II

Consider the case where γ_k is imaginary, that is, where there are zeros on the $j\omega$ axis, such as is the case for a Chebyshev Type II or elliptic lowpass prototype. To make the imaginary property of γ_k explicit, let $\gamma_k = j\rho_k$, where ρ_k is real. To determine the corresponding bandpass filter zeros, let

$$\frac{s^2 + \omega_o^2}{B_p s} + j\rho_k = 0 .$$

It follows that

$$s^2 + jB_p \rho_k s + \omega_o^2 = 0 ,$$

or the two zeros corresponding to the one lowpass prototype zero, $-j\rho_k$, are

$$s_{1,2} = -\frac{jB_p \rho_k}{2} \pm \frac{j}{2} \sqrt{B_p^2 \rho_k^2 + 4 \omega_o^2} ,$$

where it is noted that both zeros are imaginary, but not the same magnitude (not

conjugates). However, if the lowpass prototype has a zero at $-j\rho_k$, there will also be a zero at $j\rho_k$. Therefore, the four bandpass zeros corresponding to the two lowpass prototype zeros at $\pm j\rho_k$ are as follows:

$$-\frac{jB_p \rho_k}{2} \pm \frac{j}{2} \sqrt{B_p^2 \rho_k^2 + 4\omega_o^2} \quad , \quad \frac{jB_p \rho_k}{2} \pm \frac{j}{2} \sqrt{B_p^2 \rho_k^2 + 4\omega_o^2} \quad .$$

Note also that

$$\frac{s^2 + \omega_o^2}{B_p s} \pm j\rho_k = \infty$$

indicates two poles at $s = 0$, canceling two of the zeros introduced at the origin by the lowpass-to-bandpass transformation of two real poles of $H_{LPP}(s)$ or of one complex-conjugate pair of poles.

Therefore, each pair of conjugate imaginary zeros in $H_{LPP}(s)$ will yield two pair of conjugate imaginary zeros in $H_{BPP}(s)$, as developed above, and two poles at the origin. Note that if $4\omega_o^2 \gg B_p^2 \rho_k^2$, then the conjugate imaginary zeros will be approximately $j\omega_o \pm jB_p \rho_k/2$ and $-j\omega_o \pm jB_p \rho_k/2$.

Example 9.9

Suppose $\omega_o = 5$, $B_p = 2$, and there are two lowpass prototype zeros located at $\pm j3$. It follows that the corresponding bandpass filter zeros will be as follows:

$$\pm j 2.83, \quad \text{and} \quad \pm j 8.83,$$

and there will also be two poles introduced at $s = 0$. □

Case III

Consider the case where μ_k is complex, that is, where $\mu_k = \alpha_k + j\beta_k$. To determine the corresponding bandpass filter poles, let

$$\frac{s^2 + \omega_o^2}{B_p s} + \mu_k = 0 \quad .$$

It follows that

$$s^2 + B_p \mu_k s + \omega_o^2 = 0 \quad ,$$

or the two poles corresponding to the one lowpass prototype pole, $-\mu_k = -\alpha_k - j\beta_k$, are

$$\begin{aligned} s_{1,2} &= -\frac{B_p \mu_k}{2} \pm \frac{1}{2} \sqrt{B_p^2 \mu_k^2 - 4\omega_o^2} \\ &= -\frac{B_p \mu_k}{2} \pm \Phi \end{aligned}$$

$$= -\frac{B_p \alpha_k}{2} \pm \Re\{\Phi\} - \frac{j B_p \beta_k}{2} \pm j \Im\{\Phi\} ,$$

where

$$\Phi = \frac{1}{2} \sqrt{M_k} \angle (\phi_k/2) ,$$

and

$$M_k = \sqrt{R_k^2 + I_k^2} ,$$

i.e., M_k is the polar magnitude of, and ϕ_k is the angle of, $R_k + j I_k$, where

$$R_k = B_p^2 (\alpha_k^2 - \beta_k^2) - 4 \omega_o^2 , \quad I_k = 2 B_p^2 \alpha_k \beta_k .$$

Note that Φ may be expressed as follows:

$$\Phi = \frac{1}{2} \sqrt{M_k} \angle (\phi_k/2) = \frac{1}{2} \sqrt{M_k} \cos(\phi_k/2) + \frac{j}{2} \sqrt{M_k} \sin(\phi_k/2) .$$

Therefore,

$$s_1 = \frac{1}{2} [-B_p \alpha_k + \sqrt{M_k} \cos(\phi_k/2) + j(\sqrt{M_k} \sin(\phi_k/2) - B_p \beta_k)]$$

and

$$s_2 = \frac{1}{2} [-B_p \alpha_k - \sqrt{M_k} \cos(\phi_k/2) - j(\sqrt{M_k} \sin(\phi_k/2) + B_p \beta_k)] .$$

If there is a pole in the lowpass prototype at $-\alpha_k - j \beta_k$ there will also be one at $-\alpha_k + j \beta_k$, resulting in the following two poles in the bandpass transfer function:

$$s_3 = \frac{1}{2} [-B_p \alpha_k + \sqrt{M_k} \cos(\phi_k/2) + j(-\sqrt{M_k} \sin(\phi_k/2) + B_p \beta_k)]$$

and

$$s_4 = \frac{1}{2} [-B_p \alpha_k - \sqrt{M_k} \cos(\phi_k/2) - j(-\sqrt{M_k} \sin(\phi_k/2) - B_p \beta_k)] .$$

Note that $s_3 = s_1^*$ and $s_4 = s_2^*$. Therefore, the lowpass-to-bandpass transformation results in two pair of complex-conjugate poles for every complex-conjugate pair of poles in the lowpass prototype. It should be noted that two zeros at the origin are also introduced for every complex-conjugate pair of poles in the lowpass prototype. Note also that for practical lowpass prototype poles that the corresponding bandpass poles will have imaginary parts centered around ω_o . See the following examples for illustrations of this.

In summary, one pair of complex-conjugate poles in the lowpass prototype, $-\alpha_k \pm j\beta_k$, results in the following poles in the bandpass transfer function:

$$\frac{1}{2} [-B_p \alpha_k + \sqrt{M_k} \cos(\phi_k/2) \pm j(-\sqrt{M_k} \sin(\phi_k/2) + B_p \beta_k)]$$

and

$$\frac{1}{2} [-B_p \alpha_k - \sqrt{M_k} \cos(\phi_k/2) \pm j(\sqrt{M_k} \sin(\phi_k/2) + B_p \beta_k)] .$$

Example 9.10

Given that $\mu_k = 1 \pm j$, $\omega_o = 5$, and $B_p = 2$, it follows that $M_k = 100.319$, $\phi_k = 175.426^\circ$, and that the bandpass poles are:
 $-0.8002 \pm j4.004$, and $-1.1998 \pm j6.004$. \square

Example 9.11

Given that the lowpass prototype is a 4th-order Butterworth filter, $\omega_o = 100$, and $B_p = 20$, the corresponding bandpass filter will have the following poles:

$$\begin{aligned} & -3.475 \pm j91.11 , \quad -4.179 \pm j109.59 , \\ & -8.884 \pm j95.82 , \quad -9.594 \pm j103.47 . \end{aligned}$$

The lowpass-to-bandpass transformation will also introduce four zeros at the origin. \square

Example 9.12

Given that the lowpass prototype is a fifth-order elliptic filter with $A_p = 1$ dB and $A_s = 75$ dB (poles: -0.3072 , $-0.0802 \pm j0.9923$, $-0.2329 \pm j0.6386$, and zeros: $\pm j2.3454$, $\pm j3.6815$), $\omega_o = 100$, and $B_p = 20$, the corresponding bandpass filter will have the following poles:

$$\begin{aligned} & -0.7231 \pm j90.56 , \quad -0.8815 \pm j110.41 , \quad -2.1805 \pm j93.79 , \\ & -2.4774 \pm j106.56 , \quad -3.0722 \pm j99.95 . \end{aligned}$$

The bandpass zeros will be as follows:

$$\begin{aligned} & 0 , \quad \pm j69.75 , \quad \pm j79.26 , \\ & \pm j126.17 , \quad \pm j143.38 . \end{aligned}$$

\square

Magnitude Frequency Response and Phase Response

Given that the bandpass transfer function can be factored into the product of a lowpass transfer function and a highpass transfer function, as was done in **Example 9.7**, and given that the ω_p for the lowpass response is greater than that for the highpass response, the magnitude response for a bandpass filter will be very small if not zero at *DC*, rise to some maximum, and then fall off again to some small value as ω is increased. More precisely, consider the basic lowpass-to-bandpass transformation, first at *DC* for the lowpass prototype:

$$\frac{s^2 + \omega_o^2}{B_p s} = 0 .$$

From this, since $s = \pm j \omega_o$ causes the above equation to be satisfied, it is implied that when $\omega = \pm \omega_o$ for the bandpass filter the response will be the same as it at *DC* for the lowpass prototype. Note that $\omega = -\omega_o$ is for the negative side of the two-sided frequency response. Therefore,

$$|H_{BP}(j\omega_o)| = |H_{LPP}(0)| \quad \text{and} \quad \angle H_{BP}(j\omega_o) = \angle H_{LPP}(0) .$$

Now consider the lowpass prototype at $\omega = \pm \omega_p$, or, since ω_p is normalized, at $\omega = \pm 1$, or, in terms of s , at $s = \pm j$:

$$\frac{s^2 + \omega_o^2}{B_p s} = \pm j . \tag{9.23}$$

From (9.23) it follows that

$$\begin{aligned} s_{1,2} &= \frac{j}{2} \left[\pm B_p \pm \sqrt{B_p^2 + 4 \omega_o^2} \right] \\ &= \pm \frac{j}{2} \left[B_p \pm \sqrt{B_p^2 + 4 \omega_o^2} \right] , \quad \pm \frac{j}{2} \left[-B_p \pm \sqrt{B_p^2 + 4 \omega_o^2} \right] . \end{aligned} \tag{9.24}$$

Considering only positive radian frequencies, (9.24) implies the following for ω_{p_1} and ω_{p_2} :

$$\omega_{p_1, p_2} = \frac{1}{2} \sqrt{B_p^2 + 4 \omega_o^2} \pm \frac{B_p}{2} . \tag{9.25}$$

That is, $|H_{BP}(j\omega_{p_1})| = |H_{BP}(j\omega_{p_2})| = |H_{LPP}(j)|$. From (9.25) it readily follows that

$$\omega_o = \sqrt{\omega_{p_1} \omega_{p_2}} .$$

Now consider the lowpass prototype at $\omega = \pm \omega_s$:

$$\frac{s^2 + \omega_o^2}{B_p s} = \pm j \omega_s \quad (9.26)$$

From (9.26) it follows that

$$\begin{aligned} s_{1,2} &= \frac{j}{2} \left[\pm B_p \omega_s \pm \sqrt{B_p^2 \omega_s^2 + 4 \omega_o^2} \right] \\ &= \pm \frac{j}{2} \left[B_p \omega_s \pm \sqrt{B_p^2 \omega_s^2 + 4 \omega_o^2} \right], \pm \frac{j}{2} \left[-B_p \omega_s \pm \sqrt{B_p^2 \omega_s^2 + 4 \omega_o^2} \right] \end{aligned} \quad (9.27)$$

Considering only positive radian frequencies, and introducing ω_{s_1} and ω_{s_2} , (9.27) implies the following:

$$\omega_{s_1, s_2} = \frac{1}{2} \sqrt{B_p^2 \omega_s^2 + 4 \omega_o^2} \pm \frac{B_p \omega_s}{2} \quad (9.28)$$

That is, $|H_{BP}(j\omega_{s_1})| = |H_{BP}(j\omega_{s_2})| = |H_{LPP}(j\omega_s)|$. From (9.28) it readily follows that

$$\omega_o = \sqrt{\omega_{s_1} \omega_{s_2}} \quad .$$

It also readily follows from (9.28) that

$$B_p \omega_s = \omega_{s_2} - \omega_{s_1} \quad .$$

Summarizing the magnitude frequency response:

$$\begin{aligned} |H_{BP}(j\omega_o)| &= |H_{LPP}(0)| \quad , \\ |H_{BP}(j\omega_{p_1})| &= |H_{BP}(j\omega_{p_2})| = |H_{LPP}(j\omega_p)| \quad , \\ |H_{BP}(j\omega_{s_1})| &= |H_{BP}(j\omega_{s_2})| = |H_{LPP}(j\omega_s)| \quad . \end{aligned}$$

Considering the phase response, it can be seen that

$$\begin{aligned} \angle H_{BP}(j\omega_{p_1}) &= -\angle H_{LPP}(j\omega_p) \quad , \\ \angle H_{BP}(j\omega_{p_2}) &= \angle H_{LPP}(j\omega_p) \quad , \\ \angle H_{BP}(j\omega_{s_1}) &= -\angle H_{LPP}(j\omega_s) \quad , \\ \angle H_{BP}(j\omega_{s_2}) &= \angle H_{LPP}(j\omega_s) \quad . \end{aligned}$$

Determination of Minimum Order

The equations for determination of minimum order for lowpass filters depend on A_p , A_s , and ω_s/ω_p . Note, however, that under the lowpass-to-bandpass transformation, ω_p maps to ω_{p_2} , $-\omega_p$ maps to ω_{p_1} , ω_s maps to ω_{s_2} , and $-\omega_s$ maps to ω_{s_1} . Therefore,

$$\frac{\omega_{s_2} - \omega_{s_1}}{\omega_{p_2} - \omega_{p_1}} = \frac{2 \omega_s}{2 \omega_p} = \frac{\omega_s}{\omega_p} .$$

Defining $B_s = \omega_{s_2} - \omega_{s_1}$:

$$\frac{B_s}{B_p} = \frac{\omega_s}{\omega_p} .$$

Therefore, the required minimum order to meet design specifications for the lowpass prototype may be determined by using the existing equations in previous chapters, replacing ω_s/ω_p by B_s/B_p . For example, to determine the minimum required order for the lowpass prototype of a Butterworth bandpass filter, (3.15) may be modified as follows:

$$N_{LPP} = \left\lceil \frac{\log \left(\left[(10^{A_s/10} - 1) / (10^{A_p/10} - 1) \right]^{1/2} \right)}{\log (B_s / B_p)} \right\rceil .$$

Example 9.13

Given the following specifications for a Chebyshev Type I bandpass filter: $f_o = 5 \text{ kHz}$, $B_p = 2 \text{ kHz}$, $B_s = 20 \text{ kHz}$, $A_p = 1 \text{ dB}$, and $A_s = 60 \text{ dB}$, making use of (4.14) it follows that $N_{LPP} = 3$. The minimum required order of the bandpass filter itself is therefore 6. □

Filter Selectivity

The definition of **Filter Selectivity** is defined for lowpass filters in (2.37). For a bandpass filter, it is defined as follows:

$$F_S^{(BP)} = \left| \frac{d |H_{BP}(j\omega)|}{d\omega} \right|_{\omega = \omega_c} , \tag{9.29}$$

where ω_c is either ω_{c_1} , the cutoff frequency on the lower frequency side of the passband, or ω_{c_2} , the cutoff frequency on the higher frequency side of the passband, and the outer magnitude signs are used to make both **Filter Selectivity** values positive. Note that (9.29) can be related to the **Filter Selectivity** of the lowpass prototype as follows:

$$\begin{aligned}
 F_S^{(BP)} &= \left| \frac{d|H_{BP}(j\omega)|}{d\omega} \right|_{\omega = \omega_c} \\
 &= \left| - \frac{d|H_{LPP}(j(\omega^2 - \omega_o^2)/(B_p \omega))|}{d\omega} \right|_{\omega = \omega_c} .
 \end{aligned} \tag{9.30}$$

By making use of a change of variable, specifically $x = (\omega^2 - \omega_o^2)/(B_p \omega)$, it follows that

$$F_S^{(BP)} = \left(\frac{\omega_c^2 + \omega_o^2}{B_p \omega_c^2} \right) F_S^{(LPP)} , \tag{9.31}$$

where $F_S^{(LPP)}$ is evaluated at $\omega_c^{(LPP)} = |(\omega_c^2 - \omega_o^2)/(B_p \omega_c)|$, and ω_c is either ω_{c_1} or ω_{c_2} for the bandpass filter.

Shaping Factor

As was noted above under *Determination of Minimum Order*, under the lowpass-to-bandpass transformation, ω_p maps to ω_{p_2} , $-\omega_p$ maps to ω_{p_1} , ω_s maps to ω_{s_2} , and $-\omega_s$ maps to ω_{s_1} . Therefore,

$$\frac{\omega_{s_2} - \omega_{s_1}}{\omega_{p_2} - \omega_{p_1}} = \frac{2 \omega_s}{2 \omega_p} = \frac{\omega_s}{\omega_p} .$$

and

$$\frac{B_s}{B_p} = \frac{\omega_s}{\omega_p} .$$

This reasoning applies for arbitrary attenuations a and b . That is, the ratio of the bandpass filter bandwidth at attenuation b over the bandwidth at attenuation a is numerically the same as it is for the lowpass prototype. Denoting S_{LPP} and S_{BP} as the *Shaping Factor* for the lowpass prototype filter and the bandpass filter, respectively, it is noted that

$$S_{BP} = S_{LPP} . \tag{9.32}$$

Phase Delay and Group Delay

Phase delay is defined by (2.80). Phase delay for a lowpass-to-bandpass transformed transfer function may be expressed as follows:

$$t_{pd}^{(BP)}(\omega) = - \frac{\angle H_{BP}(j\omega)}{\omega} = - \frac{\angle H_{LPP}(j(\omega^2 - \omega_o^2)/(B_p \omega))}{\omega}$$

Therefore, it can be seen that

$$t_{pd}^{(BP)}(\omega_o) = 0 ,$$

$$t_{pd}^{(BP)}(\omega_{p_2}) = - \frac{\angle H_{LPP}(j)}{\omega_{p_2}} = \frac{t_{pd}^{(LPP)}(1)}{\omega_{p_2}} ,$$

$$t_{pd}^{(BP)}(\omega_{p_1}) = - \frac{\angle H_{LPP}(-j)}{\omega_{p_1}} = - \frac{t_{pd}^{(LPP)}(1)}{\omega_{p_1}} .$$

Group delay is defined by (2.81). Group delay for a lowpass-to-bandpass transformed transfer function may be expressed as follows:

$$t_{gd}^{(BP)}(\omega) = - \frac{d \angle H_{BP}(j\omega)}{d \omega} = - \frac{d \angle H_{LPP}(j(\omega^2 - \omega_o^2)/(B_p \omega))}{d \omega} \quad (9.33)$$

Making use of an appropriate change of variable (specifically, $x = (\omega^2 - \omega_o^2)/(B_p \omega)$), (9.33) may be expressed as follows:

$$t_{gd}^{(BP)}(\omega) = \frac{\omega^2 + \omega_o^2}{B_p \omega^2} t_{gd}^{(LPP)}\left(\frac{\omega^2 - \omega_o^2}{B_p \omega}\right) \quad (9.34)$$

From (9.34), the following may be observed:

$$t_{gd}^{(BP)}(\omega_{p_1}) = \frac{\omega_{p_1} + \omega_{p_2}}{B_p \omega_{p_1}} t_{gd}^{(LPP)}(1),$$

$$t_{gd}^{(BP)}(\omega_{p_2}) = \frac{\omega_{p_1} + \omega_{p_2}}{B_p \omega_{p_2}} t_{gd}^{(LPP)}(1),$$

$$t_{gd}^{(BP)}(\omega_o) = \frac{2}{B_p} t_{gd}^{(LPP)}(0) .$$

Example 9.14

Suppose

$$H_{LPP}(s) = \frac{1}{s^3 + 2s^2 + 2s + 1} ,$$

where $H_{LPP}(s)$ denotes a 3rd-order Butterworth lowpass prototype transfer function. It follows that

$$\angle H_{LPP}(j\omega) = -\tan^{-1} \left[\frac{\omega(2 - \omega^2)}{1 - 2\omega^2} \right],$$

from which

$$t_{gd}^{(LPP)}(\omega) = -\frac{d \angle H_{LPP}(j\omega)}{d\omega} = \frac{2\omega^4 + \omega^2 + 2}{\omega^6 + 1}.$$

Therefore, $t_{gd}^{(LPP)}(0) = 2$, and $t_{gd}^{(LPP)}(1) = 2.5$. From (9.34):

$$t_{gd}^{(BP)}(\omega_{p_1}) = 2.5 \frac{\omega_{p_1} + \omega_{p_2}}{B_p \omega_{p_1}},$$

$$t_{gd}^{(BP)}(\omega_{p_2}) = 2.5 \frac{\omega_{p_1} + \omega_{p_2}}{B_p \omega_{p_2}},$$

$$t_{gd}^{(BP)}(\omega_o) = \frac{4}{B_p}.$$

If $\omega_o = 1000$ and $B_p = 200$, then, from (9.25), $\omega_{p_1} = 904.988$ and $\omega_{p_2} = 1104.988$, and it follows that $t_{gd}^{(BP)}(\omega_{p_1}) = 27.76 \text{ ms}$, $t_{gd}^{(BP)}(\omega_{p_2}) = 22.74 \text{ ms}$, and $t_{gd}^{(BP)}(\omega_o) = 20.0 \text{ ms}$.

Time-Domain Response

As noted in *Example 9.7*, the bandpass transfer function may be represented as the product of a lowpass transfer function and a highpass transfer function:

$$H_{BP}(s) = H_{LP}(s) H_{HP}(s).$$

It follows that

$$h_{BP}(t) = h_{LP}(t) * h_{HP}(t).$$

As was noted in (2.7), and plotted in **Figure 2.10**, the ideal bandpass filter has a unit impulse response that is a sinusoid at the bandpass center frequency modulated by a *sinc* function at a frequency of the bandwidth over two. As was commented on in **Chapter 2**, while the impulse response of the ideal bandpass filter is non-causal, it is still representative of what is expected from a practical bandpass filter (see *Example 9.15* immediately below).

Example 9.15

Suppose the lowpass prototype filter is a 5th-order Chebyshev Type II filter with $A_p = 3 \text{ dB}$ at $\omega_p = 1 \text{ rad/s}$, and $A_s = 80 \text{ dB}$. Suppose a bandpass filter is obtained from this prototype with $\omega_o = 1000 \text{ rad/s}$ and $B_p = 200 \text{ rad/s}$. Whereas the poles of the lowpass filter are as follows:

$-1.0388,$
 $-0.8181 \pm j 0.6174,$
 and $-0.2996 \pm j 0.9578,$
 and the zeros of the lowpass filter are as follows:
 $\pm j 3.8829,$
 and $\pm j 6.2827,$
 the poles of the bandpass filter are:
 $-103.9 \pm j 994.6,$
 $-86.86 \pm j 1060.3,$
 $-76.75 \pm j 936.8,$
 $-32.81 \pm j 1099.9,$
 and $-27.10 \pm j 908.4,$
 and the zeros of the bandpass filter are:
 0
 $\pm j 552.7,$
 $\pm j 684.4,$
 $\pm j 1461.0,$
 and $\pm j 1809.3.$

The magnitude frequency response of the bandpass filter is shown in **Figure 9.8**, where the logarithmic frequency axis extends from 10 rad/s to 10^5 rad/s . A detailed passband magnitude frequency response is shown in **Figure 9.9**, where the linear frequency axis extends from 800 rad/s to 1200 rad/s . Note that ω_{p_1} and ω_{p_2} (ω_{c_1} and ω_{c_2} here since $A_p = 3 \text{ dB}$) may be determined from (9.25)^{p1} to be 904.99 rad/s and 1104.99 rad/s , respectively. These values are also observed in the data plotted in **Figure 9.9**.

Values for *Filter Selectivity* may be obtained from (9.31) and (5.11) to be 0.0204 and 0.0167 at ω_{c_1} and ω_{c_2} , respectively. These values may also be obtained from the data plotted in **Figure 9.9**.

The value of the *Shaping Factor* may be obtained from (9.32) and (5.14), for $a = 3 \text{ dB}$ and $b = 80 \text{ dB}$, to be 3.69. This value may also be obtained from the data plotted in **Figure 9.8**.

For a 5th-order Chebyshev Type II lowpass filter, the frequencies where the magnitude frequency is zero are obtained from (5.7) to be $1.0515 \omega_s$, $1.7013 \omega_s$ and ∞ . The frequencies where the magnitude response equals A_s in the stopband are obtained from (5.8) to be $1.2361 \omega_s$ and $3.2361 \omega_s$. From the basic mapping for a lowpass-to-bandpass frequency transformation,

$$\frac{-\omega_{BP}^2 + \omega_o^2}{B_p j \omega_{BP}} = \pm j \omega_{LPP} \tag{9.35}$$

where ω_{LPP} is the radian frequency variable for the lowpass prototype, and ω_{BP} is the radian frequency variable for the bandpass filter. Multiplying both sides of (9.35)

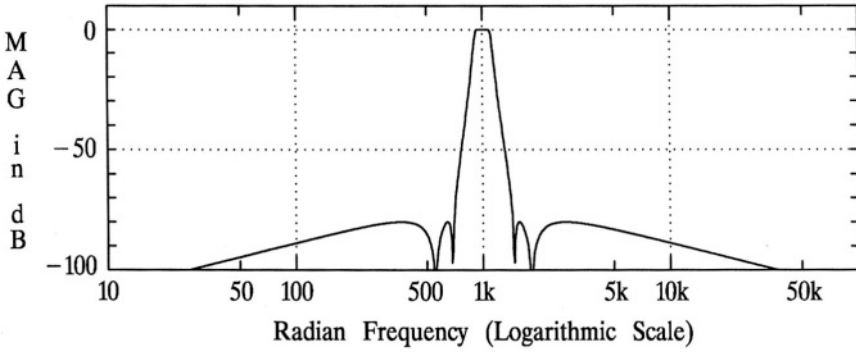


Figure 9.8 The magnitude frequency response of the Chebyshev Type II bandpass filter in *Example 9.15*: the order is 10, $A_p = 3$ dB, $A_s = 80$ dB, $\omega_o = 1000$ rad/s, and $B_p = 200$ rad/s.

by j it follows that

$$\omega_{BP} = \pm \frac{B_p \omega_{LPP}}{2} \pm \frac{1}{2} \sqrt{B_p^2 \omega_{LPP}^2 + 4 \omega_o^2} . \quad (9.36)$$

In this example, ω_s , obtained from (5.9), is 3.6929 rad/s, and therefore the frequencies where the response is zero are 3.8831, 6.2827 and ∞ , and the

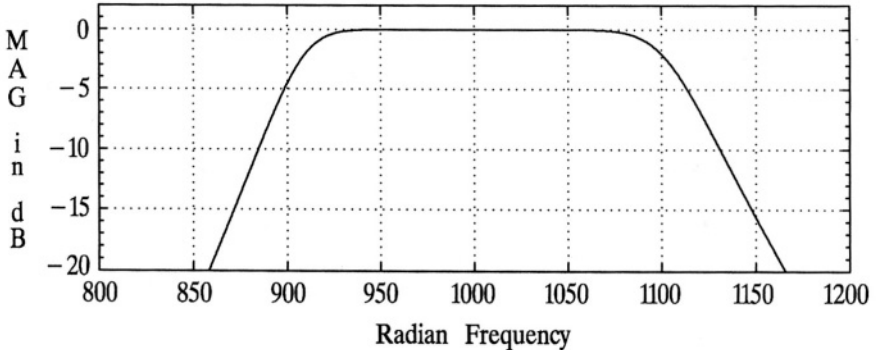


Figure 9.9 A more detailed plot of the passband magnitude frequency response of the Chebyshev Type II bandpass filter in *Example 9.15*: the order is 10, $A_p = 3$ dB, $A_s = 80$ dB, $\omega_o = 1000$ rad/s, and $B_p = 200$ rad/s.

frequencies where the response equals A_s in the stopband are 4.5648 and 11.9506. Using these frequencies in (9.36), the resultant positive values where the bandpass response is zero are as follows: 552.7, 684.4, 1461.0 and 1809.3. The resultant positive values where the bandpass response equals A_s in the stopband are 363.2, 642.8, 1555.7, and 2753.3. These frequencies, where the magnitude response is either zero or equal to A_s , can be observed in the data plotted in Figure 9.8.

The phase response of the bandpass filter is shown in Figure 9.10. Observe that the phase at the center frequency is zero, the same as the phase at DC for the lowpass prototype. Also, observe that the bandpass phase at ω_{p_1} (904.99 rad/s) is equal to the negative of the phase at 1 rad/s for the lowpass prototype (see Figure 5.3). Note the other phase relationships indicated on page 294.

The phase delay of the bandpass filter is shown in Figure 9.11. Observe that the phase delay at the center frequency is zero, as indicated on page 297. Note that $t_{pd}^{(LPP)}(1)$ is 3.8654s, which is obtainable from the data plotted in Figure 5.5. This value divided by ω_{p_1} , as indicated on page 297, is -4.27 ms , the value for $t_{pd}^{(BP)}(\omega_{p_1})$, while this value divided by ω_{p_2} is 3.50 ms, the value for $t_{pd}^{(BP)}(\omega_{p_2})$; these values for the bandpass filter can be observed in Figure 9.11.

The group delay of the bandpass filter is shown in Figure 9.12. The value for $t_{gd}^{(LPP)}(0)$ is 3.1152 s and for $t_{gd}^{(LPP)}(1)$ is 5.1127 s. Therefore, using the equations on page 297, it follows that $t_{gd}^{(BP)}(\omega_{p_1}) = 56.78\text{ ms}$, $t_{gd}^{(BP)}(\omega_{p_2}) = 46.50\text{ ms}$, and $t_{gd}^{(BP)}(\omega_o) = 31.15\text{ ms}$. These group delay values are observable in Figure 9.12.

The unit impulse response of the bandpass filter is shown in Figure 9.13. Note that the frequency of the high-frequency component (the modulated sinusoid) is equal to ω_o (1000 rad/s), and that the envelope is the same as shown in Figure 5.7 for order five, but amplitude scaled by $\omega_o B_p/2$ and frequency scaled by $B_p/2$.

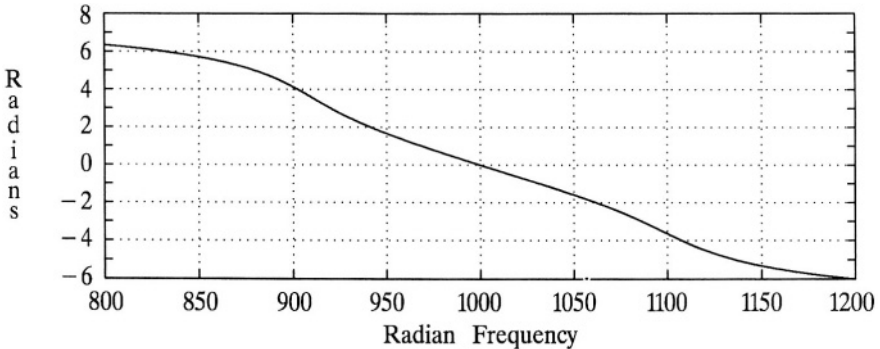


Figure 9.10 The phase response of the Chebyshev Type II bandpass filter in Example 9.15: the order is 10, $A_p = 3\text{ dB}$, $A_s = 80\text{ dB}$, $\omega_o = 1000\text{ rad/s}$, and $B_p = 200\text{ rad/s}$.

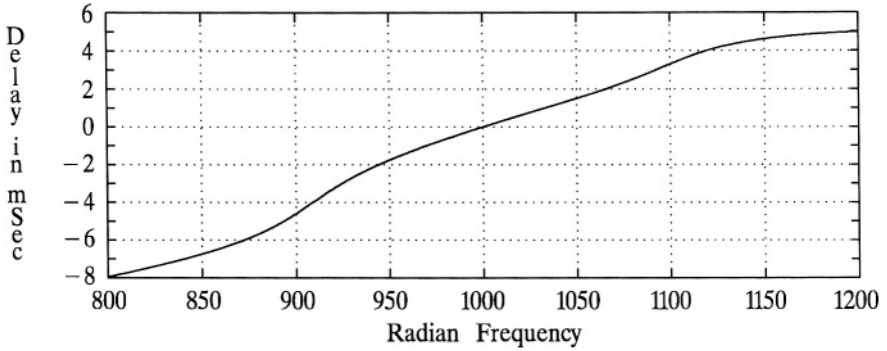


Figure 9.11 The phase delay response of the Chebyshev Type II bandpass filter in *Example 9.15*: the order is 10, $A_p = 3 \text{ dB}$, $A_s = 80 \text{ dB}$, $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$.

The unit step response is shown in **Figure 9.14**. Recall that the unit step response is the integral of the unit impulse response. The unit step response shows the same modulated sinusoid frequency of ω_o , and the same envelope shape as the unit impulse response. □

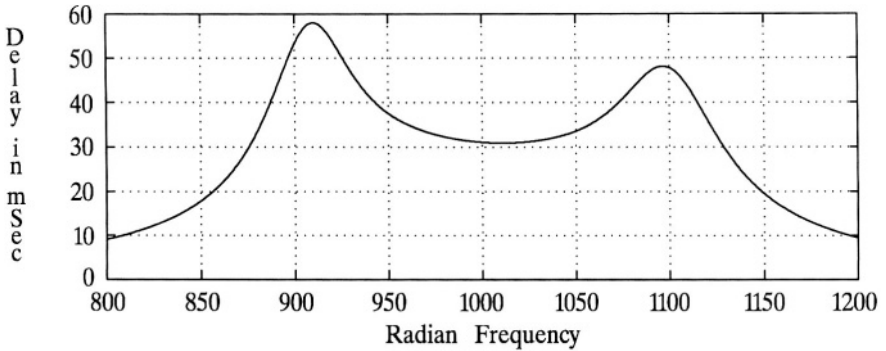


Figure 9.12 The group delay response of the Chebyshev Type II bandpass filter in *Example 9.15*: the order is 10, $A_p = 3 \text{ dB}$, $A_s = 80 \text{ dB}$, $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$.

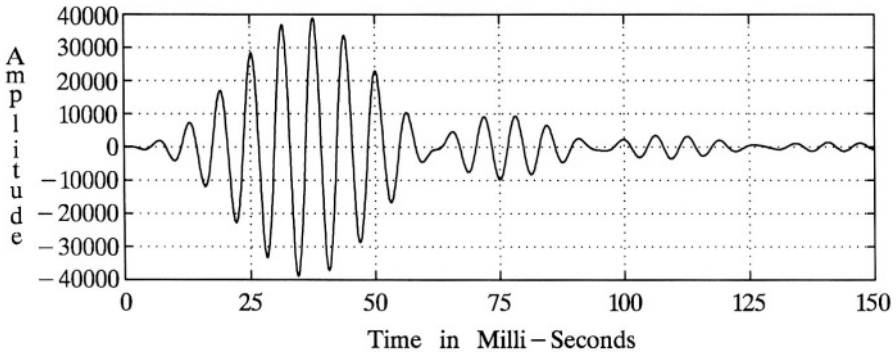


Figure 9.13 The unit impulse response of the Chebyshev Type II bandpass filter in *Example 9.15*: the order is 10, $A_p = 3$ dB, $A_s = 80$ dB, $\omega_o = 1000$ rad/s, and $B_p = 200$ rad/s.

9.4 LOWPASS-TO-BANDSTOP TRANSFORMATION

The lowpass-to-bandstop transformation is stated as follows: replace every s in $H_{LPP}(s)$ by $(B_p s)/(s^2 + \omega_o^2)$, where ω_o is the “center frequency” of the bandstop filter frequency response in rad/s and B_p is the stop bandwidth, i.e.,

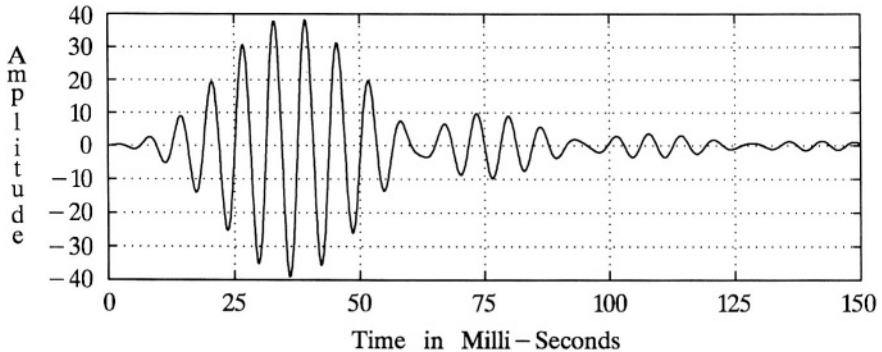


Figure 9.14 The unit step response of the Chebyshev Type II bandpass filter in *Example 9.15*: the order is 10, $A_p = 3$ dB, $A_s = 80$ dB, $\omega_o = 1000$ rad/s, and $B_p = 200$ rad/s.

$B_p = \omega_{p2} - \omega_{p1}$, where ω_{p2} is the upper edge of the stopband where the attenuation is A_p and ω_{p1} is the lower edge of the stopband. Note that ω_p must be unity for the lowpass prototype.

Example 9.16

Suppose

$$H_{LPP}(s) = \frac{1}{s^3 + 2s^2 + 2s + 1} ,$$

where $H_{LPP}(s)$ denotes a 3rd-order Butterworth lowpass prototype transfer function. If it is desired to transform this into a bandstop filter with arbitrary ω_o and B_p :

$$\begin{aligned} H_{BS}(s) &= H_{LPP}(s) \Big|_{s \rightarrow \frac{B_p s}{s^2 + \omega_o^2}} \\ &= \frac{1}{\left(\frac{B_p s}{s^2 + \omega_o^2}\right)^3 + 2\left(\frac{B_p s}{s^2 + \omega_o^2}\right)^2 + 2\left(\frac{B_p s}{s^2 + \omega_o^2}\right) + 1} \\ &= \frac{(s^2 + \omega_o^2)^3}{s^6 + 2B_p s^5 + (3\omega_o^2 + 2B_p^2)s^4 + B_p(4\omega_o^2 + B_p^2)s^3 \\ &\quad + \omega_o^2(3\omega_o^2 + 2B_p^2)s^2 + 2B_p\omega_o^4s + \omega_o^6} , \end{aligned}$$

where it is noted that the denominator is the same here as it is in *Example 9.7*. Note that $H_{BS}(s)$ may be expressed as follows:

$$\begin{aligned} H_{BS}(s) &= \frac{k_1}{s^3 + \alpha_1 s^2 + \alpha_2 s + \alpha_3} + \frac{k_2 s^3}{s^3 + \beta_1 s^2 + \beta_2 s + \beta_3} \\ &= H_{LP}(s) + H_{HP}(s) , \end{aligned}$$

where, since this is a bandstop filter, the ω_c of the lowpass is less than the ω_c of the highpass. Note the following observations: (1) The order of the bandstop transfer function is twice that of the lowpass prototype and therefore will always be even, (2) The bandstop transfer function may be separated into the sum of a lowpass transfer function and a highpass transfer function, and (3) For this example, $|H_{BS}(0)| = |H_{BS}(\infty)| = 1$, which is typical for a bandstop filter. If the lowpass prototype filter has ripples in either the passband or the stopband, then $|H_{BS}(0)|$ and $|H_{BS}(\infty)|$ may not be unity. \square

Poles and Zeros

If $H_{LPP}(s)$ is shown in factored form:

$$H_{LPP}(s) = \frac{K \prod_{k=1}^M (s + \gamma_k)}{\prod_{k=1}^N (s + \mu_k)} ,$$

where the zeros, $-\gamma_k$, and the poles, $-\mu_k$, may be, and in general are, complex, then consider the results of the lowpass-to-bandstop transformation.

Case I

Consider the case where μ_k is real. To determine the corresponding bandstop filter poles, let

$$\frac{B_p s}{s^2 + \omega_o^2} + \mu_k = 0 .$$

It follows that

$$s^2 + \frac{B_p}{\mu_k} s + \omega_o^2 = 0 ,$$

or the two poles corresponding to the one lowpass prototype pole, $-\mu_k$, are

$$s_{1,2} = -\frac{B_p}{2 \mu_k} \pm \frac{1}{2} \sqrt{(B_p^2 / \mu_k^2) - 4 \omega_o^2} .$$

Theoretically, $s_{1,2}$ could be real ($(B_p^2 / \mu_k^2) > 4 \omega_o^2$), however such will not be the case for practical bandstop filters. Therefore, consider $4 \omega_o^2 > (B_p^2 / \mu_k^2)$:

$$s_{1,2} = -\frac{B_p}{2 \mu_k} \pm \frac{j}{2} \sqrt{4 \omega_o^2 - (B_p^2 / \mu_k^2)} .$$

Also, note that since the numerator as well as the denominator of the transfer function is multiplied by $s^2 + \omega_o^2$ in order to form the above quadratic equation, it follows that two transfer function zeros are introduced at $\pm j \omega_o$.

Therefore, each *real* pole in $H_{LPP}(s)$ will yield a pair of complex conjugate poles in $H_{BS}(s)$, as developed above, and two zeros at $\pm j \omega_o$. Note that if $4 \omega_o^2 \gg B_p^2 / \mu_k^2$, then the imaginary part of those complex conjugate poles will be approximately $\pm \omega_o$.

Example 9.17

Suppose $\mu_k = 1$, $\omega_o = 5$, and $B_p = 2$. It follows that

$$s_{1,2} = -1 \pm j \sqrt{24} \cong -1 \pm j 4.90$$

will be the resultant poles, and there will two zeros at $s = \pm j 5$. □

Case II

Consider the case where γ_k is imaginary, that is, where there are zeros on the $j\omega$ axis, such as is the case for a Chebyshev Type II or elliptic lowpass prototype. To make the imaginary property of γ_k explicit, let $\gamma_k = j\rho_k$, where ρ_k is real. To determine the corresponding bandstop filter zeros, let

$$\frac{B_p s}{s^2 + \omega_o^2} + j\rho_k = 0 \quad .$$

It follows that

$$s^2 - j\frac{B_p}{\rho_k}s + \omega_o^2 = 0 \quad ,$$

or the two zeros corresponding to the one lowpass prototype zero, $-j\rho_k$, are

$$s_{1,2} = \frac{jB_p}{2\rho_k} \pm \frac{j}{2}\sqrt{(B_p^2/\rho_k^2) + 4\omega_o^2} \quad ,$$

where it is noted that both zeros are imaginary, but not the same magnitude (not conjugates). However, if the lowpass prototype has a zero at $-j\rho_k$, there will also be a zero at $j\rho_k$. Therefore, the four bandstop zeros corresponding to the two lowpass prototype zeros at $\pm j\rho_k$ are as follows:

$$-\frac{jB_p}{2\rho_k} \pm \frac{j}{2}\sqrt{(B_p^2/\rho_k^2) + 4\omega_o^2} \quad , \quad \frac{jB_p}{2\rho_k} \pm \frac{j}{2}\sqrt{(B_p^2/\rho_k^2) + 4\omega_o^2} \quad .$$

Also, note that since the denominator as well as the numerator is multiplied by $s^2 + \omega_o^2$ for each zero in the lowpass prototype transfer function in order to form the above quadratic equations, it follows that four transfer function poles are introduced at $\pm j\omega_o$. However, it should be noted that these imaginary axis poles will cancel imaginary axis zeros at $\pm j\omega_o$ introduced by real poles (or complex-conjugate poles as discussed below) in the lowpass prototype transfer function.

Therefore, each pair of conjugate imaginary zeros in $H_{LPP}(s)$ will yield two pair of conjugate imaginary zeros in $H_{BS}(s)$, as developed above, and two pair of conjugate imaginary poles at $\pm j\omega_o$.

Example 9.18

Suppose $\omega_o = 5$, $B_p = 2$, and there are two lowpass prototype zeros located at $\pm j3$. It follows that the corresponding bandstop filter zeros will be as follows:

$$\pm j4.68, \quad \text{and} \quad \pm j5.34,$$

and there will also be two conjugate imaginary poles introduced at $s = \pm j5$. \square

Case III

Consider the case where μ_k is complex, that is, where $\mu_k = \alpha_k + j\beta_k$. To determine the corresponding bandstop filter poles, let

$$\frac{B_p s}{s^2 + \omega_o^2} + \mu_k = 0 \quad .$$

It follows that

$$s^2 + \frac{B_p}{\mu_k} s + \omega_o^2 = 0 \quad ,$$

or the two poles corresponding to the one lowpass prototype pole, $-\mu_k = -\alpha_k - j\beta_k$, are

$$\begin{aligned} s_{1,2} &= -\frac{B_p}{2\mu_k} \pm \frac{1}{2} \sqrt{(B_p^2/\mu_k^2) - 4\omega_o^2} \\ &= -\frac{B_p}{2\mu_k} \pm \Phi \\ &= -\frac{B_p \alpha_k}{2(\alpha_k^2 + \beta_k^2)} \pm \Re\{\Phi\} + \frac{j B_p \beta_k}{2(\alpha_k^2 + \beta_k^2)} \pm j \Im\{\Phi\} \quad , \end{aligned}$$

where

$$\Phi = \frac{1}{2} \sqrt{M_k} \angle(\phi_k/2) \quad ,$$

and

$$M_k = \sqrt{R_k^2 + I_k^2} \quad ,$$

i.e., M_k is the polar magnitude of, and ϕ_k is the angle of, $R_k - jI_k$, where

$$R_k = \frac{B_p^2(\alpha_k^2 - \beta_k^2)}{(\alpha_k^2 + \beta_k^2)^2} - 4\omega_o^2 \quad , \quad I_k = \frac{2B_p^2\alpha_k\beta_k}{(\alpha_k^2 + \beta_k^2)^2} \quad .$$

Note that Φ may be expressed as follows:

$$\Phi = \frac{1}{2} \sqrt{M_k} \angle(\phi_k/2) = \frac{1}{2} \sqrt{M_k} \cos(\phi_k/2) + \frac{j}{2} \sqrt{M_k} \sin(\phi_k/2) \quad .$$

Therefore,

$$s_1 = \frac{1}{2} \left[-\frac{B_p \alpha_k}{\alpha_k^2 + \beta_k^2} + \sqrt{M_k} \cos(\phi_k/2) + j \left(\frac{B_p \beta_k}{\alpha_k^2 + \beta_k^2} + \sqrt{M_k} \sin(\phi_k/2) \right) \right]$$

and

$$s_2 = \frac{1}{2} \left[-\frac{B_p \alpha_k}{\alpha_k^2 + \beta_k^2} - \sqrt{M_k} \cos(\phi_k/2) + j \left(\frac{B_p \beta_k}{\alpha_k^2 + \beta_k^2} - \sqrt{M_k} \sin(\phi_k/2) \right) \right] .$$

If there is a pole in the lowpass prototype at $-\alpha_k - j\beta_k$ there will also be one at $-\alpha_k + j\beta_k$, resulting in the following two poles in the bandstop transfer function:

$$s_3 = \frac{1}{2} \left[-\frac{B_p \alpha_k}{\alpha_k^2 + \beta_k^2} + \sqrt{M_k} \cos(\phi_k/2) - j \left(\frac{B_p \beta_k}{\alpha_k^2 + \beta_k^2} + \sqrt{M_k} \sin(\phi_k/2) \right) \right]$$

and

$$s_4 = \frac{1}{2} \left[-\frac{B_p \alpha_k}{\alpha_k^2 + \beta_k^2} - \sqrt{M_k} \cos(\phi_k/2) - j \left(\frac{B_p \beta_k}{\alpha_k^2 + \beta_k^2} - \sqrt{M_k} \sin(\phi_k/2) \right) \right] .$$

Note that $s_3 = s_1^*$ and $s_4 = s_2^*$. Therefore, the lowpass-to-bandstop transformation results in two pair of complex-conjugate poles for every complex-conjugate pair of poles in the lowpass prototype. It should be noted that two pair of conjugate imaginary zeros at $\pm j\omega_o$ are also introduced for every complex-conjugate pair of poles in the lowpass prototype. Note also that for practical lowpass prototype poles that the corresponding bandstop transfer function poles will have imaginary parts centered around ω_o . See the following examples for illustrations of this.

In summary, one pair of complex-conjugate poles in the lowpass prototype, $-\alpha_k \pm j\beta_k$, results in the following poles in the bandstop transfer function:

$$\frac{1}{2} \left[-\frac{B_p \alpha_k}{\alpha_k^2 + \beta_k^2} + \sqrt{M_k} \cos(\phi_k/2) \pm j \left(\frac{B_p \beta_k}{\alpha_k^2 + \beta_k^2} + \sqrt{M_k} \sin(\phi_k/2) \right) \right]$$

and

$$\frac{1}{2} \left[-\frac{B_p \alpha_k}{\alpha_k^2 + \beta_k^2} - \sqrt{M_k} \cos(\phi_k/2) \pm j \left(\frac{B_p \beta_k}{\alpha_k^2 + \beta_k^2} - \sqrt{M_k} \sin(\phi_k/2) \right) \right] .$$

Example 9.19

Given that $\mu_k = 1 \pm j$, $\omega_o = 5$, and $B_p = 2$, it follows that $M_k = 100.02$, $\phi_k = -178.85^\circ$, and that the bandstop poles are: $-0.4498 \pm j 4.500$, and $-0.5502 \pm j 5.500$. \square

Example 9.20

Given that the lowpass prototype is a 4th-order Butterworth filter, $\omega_o = 100$, and $B_p = 20$, the corresponding bandstop filter will have the following poles:

$$\begin{aligned} & -8.884 \pm j 103.473 \quad , \quad -9.594 \pm j 95.820 \quad , \\ & -3.475 \pm j 109.592 \quad , \quad -4.179 \pm j 91.115 \quad . \end{aligned}$$

The lowpass-to-bandstop transformation will also introduce four pair of conjugate imaginary zeros at $\pm j 100$. \square

Magnitude Frequency Response and Phase Response

Given that the bandstop transfer function can be factored into the sum of a lowpass transfer function and a highpass transfer function, as was done in **Example 9.16**, and given that the ω_p for the lowpass response is less than that for the highpass response, the magnitude response for a bandstop filter will be at or near unity (assuming the gain is normalized) at *DC*, fall off in the stopband attenuating a band of frequencies, and then rise again to near unity as ω is increased. More precisely, consider the basic lowpass-to-bandstop transformation, first at *DC* for the lowpass prototype:

$$\frac{B_p s}{s^2 + \omega_o^2} = 0 \quad .$$

From this, since $s = 0$ or $s \rightarrow \infty$ causes the above equation to be satisfied, it is implied that when $\omega = 0$ or $\omega \rightarrow \infty$ for the bandstop filter the response will be the same as it at *DC* for the lowpass prototype. Therefore,

$$|H_{BS}(0)| = |H_{BS}(j\infty)| = |H_{LPP}(0)| \quad ,$$

and

$$\angle H_{BS}(0) = -\angle H_{BS}(j\infty) = \angle H_{LPP}(0) \quad .$$

Now consider the lowpass prototype at $\omega = \pm \omega_p$, or, since ω_p is normalized, at $\omega = \pm 1$, or, in terms of s , at $s = \pm j$:

$$\frac{B_p s}{s^2 + \omega_o^2} = \pm j \quad . \tag{9.37}$$

From (9.37) it follows that

$$\begin{aligned} s_{1,2} &= \frac{j}{2} \left[\pm B_p \pm \sqrt{B_p^2 + 4\omega_o^2} \right] \\ &= \pm \frac{j}{2} \left[B_p \pm \sqrt{B_p^2 + 4\omega_o^2} \right], \quad \pm \frac{j}{2} \left[-B_p \pm \sqrt{B_p^2 + 4\omega_o^2} \right]. \end{aligned} \quad (9.38)$$

Considering only positive radian frequencies, (9.38) implies the following for ω_{p_1} and ω_{p_2} :

$$\omega_{p_1, p_2} = \frac{1}{2} \sqrt{B_p^2 + 4\omega_o^2} \pm \frac{B_p}{2}. \quad (9.39)$$

That is, $|H_{BS}(j\omega_{p_1})| = |H_{BS}(j\omega_{p_2})| = |H_{LPP}(j)|$. From (9.39) it readily follows that

$$\omega_o = \sqrt{\omega_{p_1} \omega_{p_2}}$$

Now consider the lowpass prototype at $\omega = \pm \omega_s$:

$$\frac{B_p s}{s^2 + \omega_o^2} = \pm j \omega_s. \quad (9.40)$$

From (9.40) it follows that

$$\begin{aligned} s_{1,2} &= \frac{j}{2} \left[\pm B_p / \omega_s \pm \sqrt{B_p^2 / \omega_s^2 + 4\omega_o^2} \right] \\ &= \pm \frac{j}{2} \left[B_p / \omega_s \pm \sqrt{B_p^2 / \omega_s^2 + 4\omega_o^2} \right]. \end{aligned} \quad (9.41)$$

Considering only positive radian frequencies, and introducing ω_{s_1} and ω_{s_2} , (9.41) implies the following:

$$\omega_{s_1, s_2} = \frac{1}{2} \sqrt{B_p^2 / \omega_s^2 + 4\omega_o^2} \pm \frac{B_p}{2\omega_s}. \quad (9.42)$$

That is, $|H_{BS}(j\omega_{s_1})| = |H_{BS}(j\omega_{s_2})| = |H_{LPP}(j\omega_s)|$. From (9.42) it readily follows that

$$\omega_o = \sqrt{\omega_{s_1} \omega_{s_2}}.$$

It also readily follows from (9.42) that

$$B_p / \omega_s = \omega_{s_2} - \omega_{s_1}.$$

Summarizing the magnitude frequency response:

$$\begin{aligned} |H_{BS}(0)| &= |H_{BS}(j\infty)| = |H_{LPP}(0)|, \\ |H_{BS}(j\omega_{p_1})| &= |H_{BS}(j\omega_{p_2})| = |H_{LPP}(j\omega_p)|, \\ |H_{BS}(j\omega_{s_1})| &= |H_{BS}(j\omega_{s_2})| = |H_{LPP}(j\omega_s)|. \end{aligned}$$

Considering the phase response, it can be seen that

$$\begin{aligned} \angle H_{BS}(j\omega_{p_1}) &= \angle H_{LPP}(j\omega_p), \\ \angle H_{BS}(j\omega_{p_2}) &= -\angle H_{LPP}(j\omega_p), \\ \angle H_{BS}(j\omega_{s_1}) &= \angle H_{LPP}(j\omega_s), \\ \angle H_{BS}(j\omega_{s_2}) &= -\angle H_{LPP}(j\omega_s). \end{aligned}$$

Determination of Minimum Order

The equations for determination of minimum order for lowpass filters depend on A_p , A_s , and ω_s/ω_p . Note, however, that under the lowpass-to-bandstop transformation, $B_p/\omega_s = \omega_s - \omega_{s_1}$, from which $\omega_s = B_p / (\omega_{s_2} - \omega_{s_1})$. The value of ω_s , here, assumes $\omega_p = 1$. In general, $\omega_s / \omega_p = B_p / (\omega_{s_2} - \omega_{s_1})$. Defining $B_s = \omega_{s_2} - \omega_{s_1}$, it follows that

$$\frac{B_p}{B_s} = \frac{\omega_s}{\omega_p}.$$

Therefore, the required minimum order to meet design specifications for the lowpass prototype may be determined by using the existing equations in previous chapters, replacing ω_s/ω_p by B_p/B_s . For example, to determine the minimum required order for the lowpass prototype of a Butterworth bandstop filter, (3.15) may be modified as follows:

$$N_{LPP} = \left\lceil \frac{\log [(10^{A_s/10} - 1) / (10^{A_p/10} - 1)]^{1/2}}{\log (B_p / B_s)} \right\rceil.$$

Example 9.21

Given the following specifications for a Chebyshev Type I bandstop filter: $f_o = 5 \text{ kHz}$, $B_p = 20 \text{ kHz}$, $B_s = 2 \text{ kHz}$, $A_p = 1 \text{ dB}$, and $A_s = 60 \text{ dB}$, making use of (4.14) it follows that $N_{LPP} = 3$. The minimum required order of the bandstop filter itself is therefore 6. □

Filter Selectivity

The definition of *Filter Selectivity* is defined for lowpass filters in (2.37). For a bandstop filter, it is defined as follows:

$$F_S^{(BS)} = \left| \frac{d|H_{BS}(j\omega)|}{d\omega} \right|_{\omega = \omega_c}, \quad (9.43)$$

where ω_c is either ω_{c_1} , the cutoff frequency on the lower frequency side of the stopband, or ω_{c_2} , the cutoff frequency on the higher frequency side of the stopband, and the outer magnitude signs are used to make both *Filter Selectivity* values positive. Note that (9.43) can be related to the *Filter Selectivity* of the lowpass prototype as follows:

$$\begin{aligned} F_S^{(BS)} &= \left| \frac{d|H_{BS}(j\omega)|}{d\omega} \right|_{\omega = \omega_c} \\ &= \left| \frac{d|H_{LPP}(jB_p \omega / (\omega_o^2 - \omega^2))|}{d\omega} \right|_{\omega = \omega_c}. \end{aligned} \quad (9.44)$$

By making use of a change of variable, specifically $x = B_p \omega / (\omega_o^2 - \omega^2)$, it follows that

$$F_S^{(BS)} = \left(\frac{B_p (\omega_o^2 + \omega_c^2)}{(\omega_o^2 - \omega_c^2)^2} \right) F_S^{(LPP)}, \quad (9.45)$$

where $F_S^{(LPP)}$ is evaluated at $\omega_c^{(LPP)} = |B_p \omega_c / (\omega_o^2 - \omega_c^2)|$, and ω_c is either ω_{c_1} or ω_{c_2} for the bandstop filter.

Shaping Factor

As was noted above under *Determination of Minimum Order*, under the lowpass-to-bandstop transformation,

$$\frac{B_p}{B_s} = \frac{\omega_s}{\omega_p}.$$

This reasoning applies for arbitrary attenuations a and b . That is, the ratio of the bandstop filter bandwidth at attenuation a over the bandwidth at attenuation b is numerically the inverse of what it is for the lowpass prototype. Denoting S_{LPP} and $S_{BS} = S_b^a = BW_a / BW_b$ as the *Shaping Factor* for the lowpass prototype filter and the bandstop filter, respectively, it is noted that

$$S_{BS} = S_{LPP}. \quad (9.46)$$

Phase Delay and Group Delay

Phase delay is defined by (2.80). Phase delay for a lowpass-to-bandstop transformed transfer function may be expressed as follows:

$$t_{pd}^{(BS)}(\omega) = -\frac{\angle H_{BS}(j\omega)}{\omega} = -\frac{\angle H_{LPP}(jB_p\omega / (\omega_o^2 - \omega^2))}{\omega}$$

Therefore, it can be seen that

$$\begin{aligned} t_{pd}^{(BS)}(0) &= t_{pd}^{(LPP)}(0), \\ t_{pd}^{(BS)}(\omega_{p_1}) &= -\frac{\angle H_{LPP}(j)}{\omega_{p_1}} = \frac{t_{pd}^{(LPP)}(1)}{\omega_{p_1}}, \\ t_{pd}^{(BS)}(\omega_{p_2}) &= -\frac{\angle H_{LPP}(-j)}{\omega_{p_2}} = -\frac{t_{pd}^{(LPP)}(1)}{\omega_{p_2}}. \end{aligned}$$

Group delay is defined by (2.81). Group delay for a lowpass-to-bandstop transformed transfer function may be expressed as follows:

$$t_{gd}^{(BS)}(\omega) = -\frac{d \angle H_{BS}(j\omega)}{d\omega} = -\frac{d \angle H_{LPP}(jB_p\omega / (\omega_o^2 - \omega^2))}{d\omega} \quad (9.47)$$

Making use of an appropriate change of variable (specifically, $x = B_p\omega / (\omega_o^2 - \omega^2)$), (9.47) may be expressed as follows:

$$t_{gd}^{(BS)}(\omega) = \left(\frac{B_p(\omega_o^2 + \omega^2)}{(\omega_o^2 - \omega^2)^2} \right) t_{gd}^{(LPP)}\left(\frac{B_p\omega}{\omega_o^2 - \omega^2} \right) \quad (9.48)$$

From (9.48), the following may be observed:

$$\begin{aligned} t_{gd}^{(BS)}(0) &= \left(\frac{B_p}{\omega_o^2} \right) t_{gd}^{(LPP)}(0), \\ t_{gd}^{(BS)}(\omega_{p_1}) &= \left(\frac{\omega_o^2 + \omega_{p_1}^2}{\omega_{p_1}(\omega_o^2 - \omega_{p_1}^2)} \right) t_{gd}^{(LPP)}(1), \\ t_{gd}^{(BS)}(\omega_{p_2}) &= \left(\frac{\omega_o^2 + \omega_{p_2}^2}{\omega_{p_2}(\omega_o^2 - \omega_{p_2}^2)} \right) t_{gd}^{(LPP)}(1). \end{aligned}$$

Example 9.22

Suppose $H_{LPP}(s)$ is a 3rd-order Butterworth lowpass prototype transfer function, as in **Example 9.14**. As was determined in **Example 9.14**, $t_{gd}^{(LPP)}(0) = 2$, and $t_{gd}^{(LPP)}(1) = 2.5$. From (9.48):

$$t_{gd}^{(BS)}(0) = 2 \left(\frac{B_p}{\omega_o^2} \right),$$

$$t_{gd}^{(BS)}(\omega_{p_1}) = 2.5 \left(\frac{\omega_o^2 + \omega_{p_1}^2}{\omega_{p_1}(\omega_o^2 - \omega_{p_1}^2)} \right),$$

$$t_{gd}^{(BS)}(\omega_{p_2}) = 2.5 \left(\frac{\omega_o^2 + \omega_{p_2}^2}{\omega_{p_2}(\omega_{p_2}^2 - \omega_o^2)} \right).$$

If $\omega_o = 1000$ and $B_p = 200$, then, from (9.39), $\omega_{p_1} = 904.988$ and $\omega_{p_2} = 1104.988$, and it follows that $t_{gd}^{(BS)}(0) = 0.4 \text{ ms}$, $t_{gd}^{(BS)}(\omega_{p_1}) = 27.76 \text{ ms}$, and $t_{gd}^{(BS)}(\omega_{p_2}) = 22.74 \text{ ms}$. \square

Time-Domain Response

As noted in **Example 9.16**, the bandstop transfer function may be represented as the sum of a lowpass transfer function and a highpass transfer function:

$$H_{BS}(s) = H_{LP}(s) + H_{HP}(s).$$

It follows that

$$h_{BS}(t) = h_{LP}(t) + h_{HP}(t),$$

and

$$h_u^{(BS)}(t) = h_u^{(LP)}(t) + h_u^{(HP)}(t).$$

Since $h_{HP}(t)$ includes an impulse, so will $h_{BS}(t)$. Since $h_u^{(LP)}(t)$ converges to unity as $t \rightarrow \infty$ and $h_u^{(HP)}(t)$ converges to zero, $h_u^{(BS)}(t)$ converges to unity.

Example 9.23

Suppose the lowpass prototype filter is a 5th-order elliptic filter with $A_p = 1 \text{ dB}$ at $\omega_p = 1 \text{ rad/s}$, and $A_s = 80 \text{ dB}$. This results in $\omega_c = 1.0308 \text{ rad/s}$ and $\omega_s = 2.4880 \text{ rad/s}$. Suppose a bandstop filter is obtained from this prototype with $\omega_o = 1000 \text{ rad/s}$ and $B_p = 200 \text{ rad/s}$. Whereas the poles of the lowpass filter are as follows:

$$\begin{aligned} & -0.3035, \\ & -0.2332 \pm j 0.6331, \end{aligned}$$

and

$$-0.0821 \pm j 0.9919,$$

and the zeros of the lowpass filter are as follows:

$$\pm j 2.6055,$$

and

$$\pm j 4.1150,$$

the poles of the bandstop filter are:

$$-7.459 \pm j 904.8,$$

$$-9.110 \pm j 1105.1,$$

$$-44.17 \pm j 869.3,$$

$$-58.30 \pm j 1147.4,$$

and

$$-329.5 \pm j 944.2,$$

and the zeros of the bandstop filter are:

$$\pm j 962.4,$$

$$\pm j 976.0,$$

$$\pm j 1000.0,$$

$$\pm j 1024.6,$$

and

$$\pm j 1039.1.$$

The magnitude frequency response of the bandstop filter is shown in **Figure 9.15**. A detailed passband magnitude frequency response is shown in **Figure 9.16**. Note that ω_{p_1} and ω_{p_2} may be determined from (9.39) to be 904.99 rad/s and 1104.99 rad/s , respectively. These values are also observed in the data plotted in **Figure 9.16**. Note that ω_{s_1} and ω_{s_2} may be determined from (9.42) to be 960.61 rad/s and 1041.0 rad/s , respectively. These values may be observed in the data plotted in **Figure 9.15**.

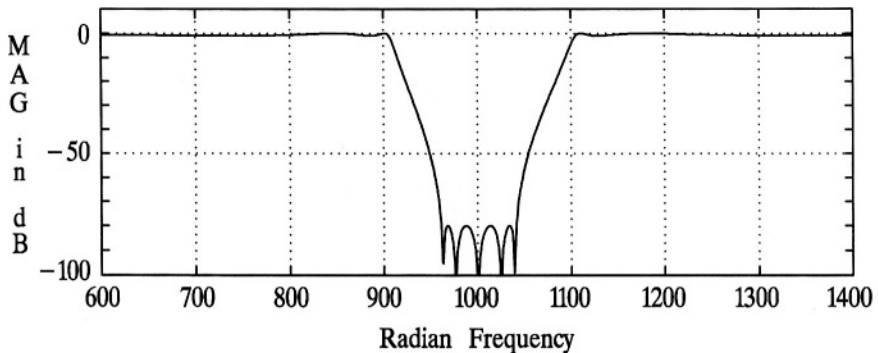


Figure 9.15 The magnitude frequency response of the elliptic bandstop filter in *Example 9.23*: the order is 10, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$.

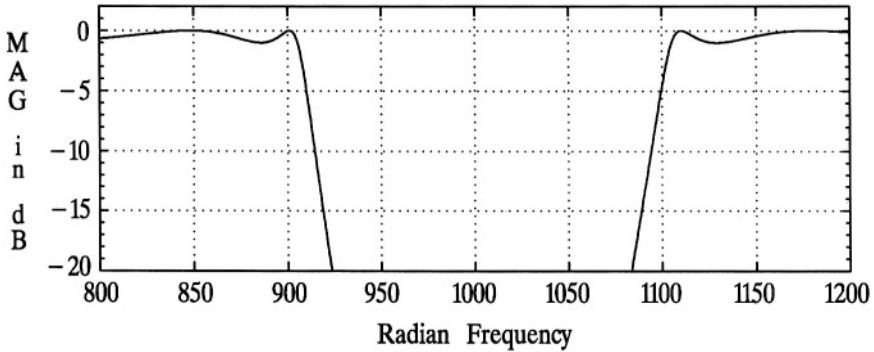


Figure 9.16 A more detailed plot of the passband magnitude frequency response of the elliptic bandstop filter in *Example 9.23*: the order is 10, $A_p = 1$ dB, $A_s = 80$ dB, $\omega_o = 1000$ rad/s, and $B_p = 200$ rad/s.

Values for *Filter Selectivity* may be obtained from (9.45). Values for ω_{c_1} and ω_{c_2} for use in (9.45) may be obtained from an equation derived similarly as (9.42):

$$\omega_{c_1, c_2} = \frac{1}{2} \sqrt{B_p^2 / \omega_c^2 + 4 \omega_o^2} \pm \frac{B_p}{2 \omega_c} \quad (9.49)$$

From (9.49), $\omega_{c_1} = 907.68$ rad/s and $\omega_{c_2} = 1101.71$ rad/s; these values are also observable in the data plotted in **Figure 9.16**. Note that $F_s^{(LPP)} = 6.3999$. It follows from (9.45) that $F_s^{(BS)} = 0.0753$ and 0.0620 for ω_{c_1} and ω_{c_2} , respectively. These values may also be obtained from the data plotted in **Figure 9.16**.

The value of the *Shaping Factor* may be obtained from (9.46). Note that $S_{LPP} = 2.4136$, where $a = 3$ dB and $b = 80$ dB. Therefore, from (9.46), $S_{BS} = 2.4136$, where $BW_a = \omega_{c_2} - \omega_{c_1} = 194.03$ and $BW_b = \omega_{s_2} - \omega_{s_1} = 80.39$. These values are also obtainable from the data plotted in **Figure 9.15**.

For a 5th-order elliptic lowpass filter, the frequencies where the magnitude frequency is zero are obtained from (5.7) to be $1.0515 \omega_s$, $1.7013 \omega_s$ and ∞ . The frequencies where the magnitude response equals A_s in the stopband are obtained from (5.8) to be $1.2361 \omega_s$ and $3.2361 \omega_s$. From the basic mapping for a lowpass-to-bandstop frequency transformation,

$$\frac{B_p j \omega_{BS}}{-\omega_{BS}^2 + \omega_o^2} = \pm j \omega_{LPP} \quad (9.50)$$

where ω_{LPP} is the radian frequency variable for the lowpass prototype, and ω_{BS} is the radian frequency variable for the bandstop filter. From (9.50) it follows that

$$\omega_{BS} = \pm \frac{B_p}{2 \omega_{LPP}} \pm \frac{1}{2} \sqrt{B_p^2 / \omega_{LPP}^2 \pm 4 \omega_o^2} . \tag{9.51}$$

In this example, ω_s , obtained from the MATLAB function ELLIPWS, is 2.4880 rad/s, and therefore the frequencies where the response is zero, obtained from (6.33), are 2.6054, 4.1148 and ∞ , and the frequencies where the response equals A_s in the stopband, obtained from (6.34), are 3.0301 and 7.7409. Using these frequencies in (9.51), the resultant positive values where the bandstop response is zero are as follows: 962.4, 976.0, 1000.0, 1024.6 and 1039.1. The resultant positive values where the bandstop response equals A_s in the stopband are 967.5, 987.2, 1013.0, and 1033.5. These frequencies, where the magnitude response is either zero or equal to A_s , can be observed in the data plotted in Figure 9.15.

The phase response of the bandstop filter is shown in Figure 9.17. Observe that the phase at the center frequency is zero, the same as the phase at DC for the lowpass prototype. Also, observe that the bandstop phase at ω_{p_1} (904.99 rad/s) is equal to the phase at 1 rad/s for the lowpass prototype when the lowpass has a normalized ω_p (see Figure 6.12 where ω_c is normalized). Note the other phase relationships indicated on page 311.

The phase delay of the bandstop filter is shown in Figure 9.18. Observe that the phase delay at the center frequency is zero, the same as the phase delay at infinity for the lowpass prototype. Note that $t_{pd}^{(LPP)}(1)$ is 5.3376 s, where ω_p is normalized. This value divided by ω_{p_1} , as indicated on page 313, is 5.90 ms, the value for $t_{pd}^{(BS)}(\omega_{p_1})$, while the negative of this value divided by ω_{p_2} is -4.83ms, the value for $t_{pd}^{(BS)}(\omega_{p_2})$: these values for the bandstop filter can be observed in Figure 9.18.

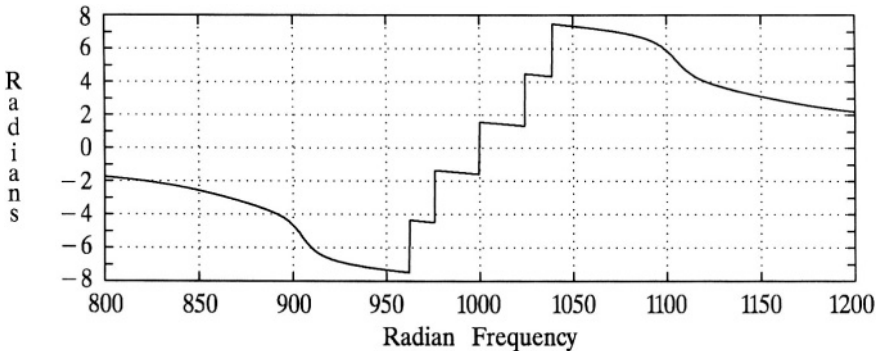


Figure 9.17 The phase response of the elliptic bandstop filter in **Example 9.23**: the order is 10, $A_p = 1$ dB, $A_s = 80$ dB, $\omega_o = 1000$ rad/s, and $B_p = 200$ rad/s.

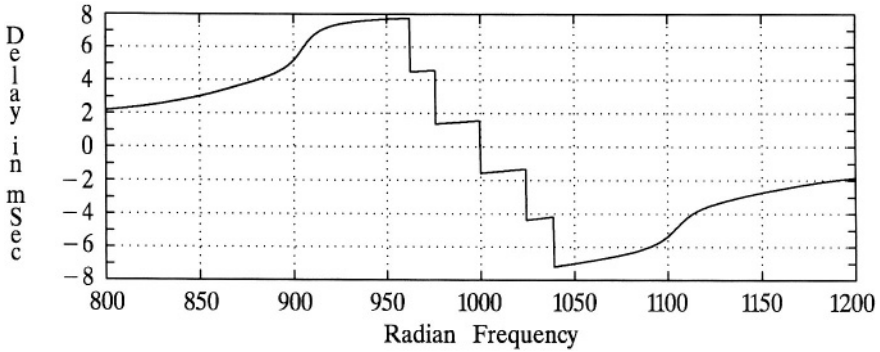


Figure 9.18 The phase delay response of the elliptic bandstop filter in *Example 9.23*: the order is 10, $A_p = 1$ dB, $A_s = 80$ dB, $\omega_o = 1000$ rad/s, and $B_p = 200$ rad/s.

The group delay of the bandstop filter is shown in **Figure 9.19**. The value for $t_{gd}^{(LPP)}(1)$ is 13.5862 s, which is obtained similarly as the data plotted in **Figure 6.15** and **Figure 6.16**, but here ω_p is unity whereas the data in **Chapter 6** was normalized for an ω_c of unity. Therefore, using the equations on page 313, it follows that $t_{gd}^{(BS)}(\omega_{p1}) = 150.9$ ms, and $t_{gd}^{(BS)}(\omega_{p2}) = 123.6$ ms. These group delay values are observable in the data plotted in **Figure 9.19**.

The unit impulse response of the bandstop filter is shown in **Figure 9.20**. This being a bandstop filter, there is an impulse at the origin, but not shown in the figure.

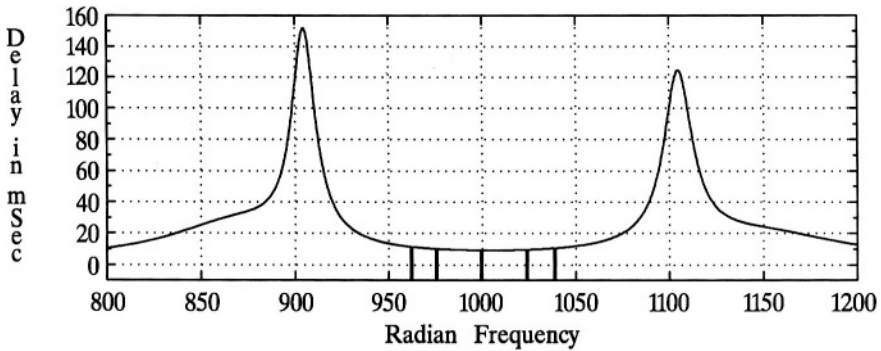


Figure 9.19 The group delay response of the elliptic bandstop filter in *Example 9.23*: the order is 10, $A_p = 1$ dB, $A_s = 80$ dB, $\omega_o = 1000$ rad/s, and $B_p = 200$ rad/s.

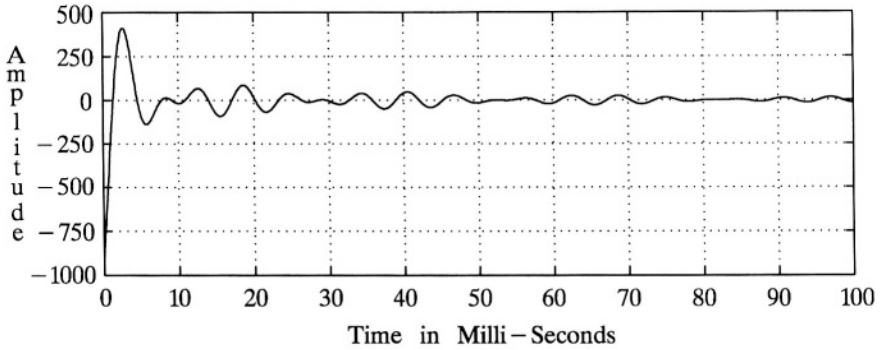


Figure 9.20 The unit impulse response of the elliptic bandstop filter in *Example 9.23*: the order is 10, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$.

The unit step response is shown in **Figure 9.21**. Recall that the unit step response is the integral of the unit impulse response. The unit step response converges to unity as the order of the lowpass prototype is odd. \square

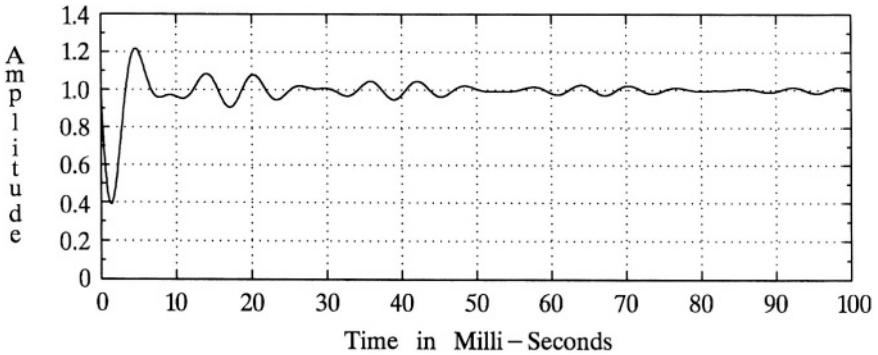


Figure 9.21 The unit step response of the elliptic bandstop filter in *Example 9.23*: the order is 10, $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$.

9.5 CHAPTER 9 PROBLEMS

- 9.1 Similar to *Example 9.1*, determine the transfer function of 3rd-order Butterworth lowpass filter with ω_c of 5000 *rad/s*.
- 9.2 Determine the transfer function of a 3rd-order Chebyshev Type I lowpass filter with $A_p = 1.5$ *dB* and $\omega_p = 1000$ *rad/s*.
- 9.3 Repeat **Problem 9.2** for $\omega_c = 1000$ *rad/s*.
- 9.4 Determine the poles for the transfer function of **Problem 9.1**.
- 9.5 Determine the poles for the transfer function of **Problem 9.2**.
- 9.6 Determine the poles for the transfer function of **Problem 9.3**.
- 9.7 Given that the desired specifications of a Butterworth lowpass filter are as follows: $A_p = 3$ *dB*, $A_s = 70$ *dB*, $\omega_p = 2,500$ *rad/s*, and $\omega_s = 10,000$ *rad/s*, determine the minimum required filter order to meet or exceed these specifications. Repeat the above for $A_p = 3$ *dB*, $A_s = 70$ *dB*, $f_p = 6,500$ *Hz*, and $f_s = 26$ *kHz*.
- 9.8 Given that the desired specifications of a Chebyshev Type I lowpass filter are as follows: $A_p = 1.2$ *dB*, $A_s = 75$ *dB*, $\omega_p = 3,500$ *rad/s*, and $\omega_s = 7,000$ *rad/s*, determine the minimum required filter order to meet or exceed these specifications. Repeat the above for $A_p = 1.2$ *dB*, $A_s = 75$ *dB*, $f_p = 6,500$ *Hz*, and $f_s = 13$ *kHz*.
- 9.9 Determine the *Filter Selectivity*, F_s , for each of the two filters in **Problem 9.7**.
- 9.10 Determine the *Filter Selectivity*, F_s , for each of the two filters in **Problem 9.8**.
- 9.11 Determine the *Shaping Factor*, S_a^b , for each of the two filters in **Problem 9.7**, where $a = 3$ *dB* and $b = 70$ *dB*.
- 9.12 Determine the *Shaping Factor*, S_a^b , for each of the two filters in **Problem 9.8**, where $a = 1.2$ *dB* and $b = 75$ *dB*.

- 9.13** Indicate how **Figure 3.8** could be used to obtain the plot of phase delay for each of the two filters in **Problem 9.7**.
- 9.14** Indicate how **Figure 3.9** could be used to obtain the plot of group delay for each of the two filters in **Problem 9.7**.
- 9.15** Indicate how **Figure 3.10** could be used to obtain the plot of the unit impulse response for each of the two filters in **Problem 9.7**.
- 9.16** Indicate how **Figure 3.11** could be used to obtain the plot of the unit step response for each of the two filters in **Problem 9.7**.
- 9.17** Similar to *Example 9.2*, determine the transfer function of 3rd-order Butterworth highpass filter with an ω_c of 5000 rad/s .
- 9.18** Determine the transfer function of a 3rd-order Chebyshev Type I highpass filter with $A_p = 1.5 \text{ dB}$ and $\omega_p = 1000 \text{ rad/s}$.
- 9.19** Repeat **Problem 9.18** for $\omega_c = 1000 \text{ rad/s}$.
- 9.20** Determine the poles and zeros for the transfer function of **Problem 9.17**.
- 9.21** Determine the poles and zeros for the transfer function of **Problem 9.18**.
- 9.22** Determine the poles and zeros for the transfer function of **Problem 9.19**.
- 9.23** Using MATLAB, plot the magnitude frequency response and the phase response for the filter of **Problem 9.17**.
- 9.24** Using MATLAB, plot the magnitude frequency response and the phase response for the filter of **Problem 9.18**.
- 9.25** Using MATLAB, plot the magnitude frequency response and the phase response for the filter of **Problem 9.19**.
- 9.26** Using MATLAB, plot the magnitude frequency response and the phase response for the highpass filter of *Example 9.3*.
- 9.27** Given that the desired specifications of a Butterworth highpass filter are as follows: $A_p = 3 \text{ dB}$, $A_s = 70 \text{ dB}$, $\omega_{sHP} = 2,500 \text{ rad/s}$, and $\omega_{pHP} = 10,000 \text{ rad/s}$, determine the minimum required filter order to meet

- or exceed these specifications. Repeat the above for $A_p = 3 \text{ dB}$, $A_s = 70 \text{ dB}$, $f_{s_{HP}} = 6,500 \text{ Hz}$, and $f_{p_{HP}} = 26 \text{ kHz}$.
- 9.28** Given that the desired specifications of a Chebyshev Type I highpass filter are as follows: $A_p = 1.2 \text{ dB}$, $A_s = 75 \text{ dB}$, $\omega_{s_{HP}} = 3,500 \text{ rad/s}$, and $\omega_{p_{HP}} = 7,000 \text{ rad/s}$, determine the minimum required filter order to meet or exceed these specifications. Repeat the above for $A_p = 1.2 \text{ dB}$, $A_s = 75 \text{ dB}$, $f_{s_{HP}} = 6,500 \text{ Hz}$, and $f_{p_{HP}} = 13 \text{ kHz}$.
- 9.29** Determine the *Filter Selectivity* of the highpass filter of **Problem 9.17** in two ways: (a) by use of (9.14) and (3.7), and (b) computationally, using MATLAB.
- 9.30** Determine the *Filter Selectivity* of the highpass filter of **Problem 9.18** in two ways: (a) by use of (9.14) and (4.9), and (b) computationally, using MATLAB.
- 9.31** Determine the *Filter Selectivity* of the highpass filter of **Problem 9.19** in two ways: (a) by use of (9.14) and (4.9), and (b) computationally, using MATLAB.
- 9.32** Determine the *Shaping Factor* of the highpass filter of **Problem 9.17** in two ways: (a) by use of (9.15) and (3.10), and (b) computationally, using MATLAB.
- 9.33** Determine the *Shaping Factor* of the highpass filter of **Problem 9.18** in two ways: (a) by use of (9.15) and (4.12), and (b) computationally, using MATLAB.
- 9.34** Determine the *Shaping Factor* of the highpass filter of **Problem 9.19** in two ways: (a) by use of (9.15) and (4.12), and (b) computationally, using MATLAB.
- 9.35** For the Butterworth highpass filter of **Problem 9.17**, determine a closed-form expression for the group delay, similar to *Example 9.4*. Using MATLAB, plot the response of your expression. For comparison, determine and plot the group delay response as obtained by computational manipulation of the phase response (the traditional computational approach).
- 9.36** Using MATLAB, plot the magnitude frequency response, phase response, phase delay response, group delay response, unit impulse response, and unit step response for a 10th-order Chebyshev Type I highpass filter with

$A_p = 1 \text{ dB}$ and $\omega_{pHP} = 1000 \text{ rad/s}$. That is, confirm the results plotted in **Figure 9.1** through **Figure 9.7** (*Example 9.6*).

- 9.37** Similar to *Example 9.11*, determine the poles and zeros of an 8th-order Butterworth bandpass filter, with $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$.
- 9.38** Determine the poles and zeros of a 6th-order Chebyshev Type I bandpass filter, with 1 dB of ripple, $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$.
- 9.39** Using MATLAB, plot the magnitude frequency response and the phase response for the filter of **Problem 9.37**.
- 9.40** Using MATLAB, plot the magnitude frequency response and the phase response for the filter of **Problem 9.38**.
- 9.41** Given that the desired specifications of a Butterworth bandpass filter are as follows: $A_p = 3 \text{ dB}$, $A_s = 70 \text{ dB}$, $B_p = 2,500 \text{ rad/s}$, and $B_s = 10,000 \text{ rad/s}$, determine the minimum required filter order to meet or exceed these specifications. Repeat the above for $A_p = 3 \text{ dB}$, $A_s = 70 \text{ dB}$, $B_p = 6,500 \text{ Hz}$, and $B_s = 26 \text{ kHz}$.
- 9.42** Given that the desired specifications of a Chebyshev Type I bandpass filter are as follows: $A_p = 1.2 \text{ dB}$, $A_s = 75 \text{ dB}$, $B_p = 3,500 \text{ rad/s}$, and $B_s = 7,000 \text{ rad/s}$, determine the minimum required filter order to meet or exceed these specifications. Repeat the above for $A_p = 1.2 \text{ dB}$, $A_s = 75 \text{ dB}$, $B_p = 6,500 \text{ Hz}$, and $B_s = 13 \text{ kHz}$.
- 9.43** Determine the *Filter Selectivity* of the bandpass filter in **Problem 9.41** with $\omega_o = 15,000 \text{ rad/s}$ in two ways: (a) by use of (9.31) and (3.7), and (b) computationally, using MATLAB.
- 9.44** Repeat **Problem 9.43** for $\omega_o = 10,000 \text{ rad/s}$.
- 9.45** Determine the *Filter Selectivity* of the bandpass filter in **Problem 9.42** with $\omega_o = 15,000 \text{ rad/s}$ in two ways: (a) by use of (9.31) and (4.9), and (b) computationally, using MATLAB.
- 9.46** Repeat **Problem 9.45** for $\omega_o = 10,000 \text{ rad/s}$.
- 9.47** Determine the *Shaping Factor* of the bandpass filter in **Problem 9.41** with $\omega_o = 15,000 \text{ rad/s}$ in two ways: (a) by use of (9.32) and (3.10), and (b) computationally, using MATLAB.

- 9.48 Repeat **Problem 9.47** for $\omega_o = 10,000 \text{ rad/s}$.
- 9.49 Determine the *Shaping Factor* of the bandpass filter in **Problem 9.42** with $\omega_o = 15,000 \text{ rad/s}$ in two ways: (a) by use of (9.32) and (4.12), and (b) computationally, using MATLAB.
- 9.50 Repeat **Problem 9.49** for $\omega_o = 10,000 \text{ rad/s}$.
- 9.51 Using the closed-form procedure of *Example 9.14*, compute the group delay of a 6th-order Butterworth bandpass filter at ω_{p_1} , ω_{p_2} , and ω_o where $\omega_o = 5000 \text{ rad/s}$ and $B_p = 500 \text{ rad/s}$.
- 9.52 Using MATLAB, plot the magnitude frequency response, phase response, phase delay response, group delay response, unit impulse response, and unit step response for a 10th-order Chebyshev Type II bandpass filter with $A_p = 3 \text{ dB}$, $A_s = 80 \text{ dB}$, $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$. That is, confirm the results plotted in **Figure 9.8** through **Figure 9.14** (*Example 9.15*).
- 9.53 Similar to *Example 9.20*, determine the poles and zeros of an 8th-order Butterworth bandstop filter, with $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$.
- 9.54 Determine the poles and zeros of a 6th-order Chebyshev Type I bandstop filter, with 1 dB of ripple, $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$.
- 9.55 Using MATLAB, plot the magnitude frequency response and the phase response for the filter of **Problem 9.53**.
- 9.56 Using MATLAB, plot the magnitude frequency response and the phase response for the filter of **Problem 9.54**.
- 9.57 Given that the desired specifications of a Butterworth bandstop filter are as follows: $A_p = 3 \text{ dB}$, $A_s = 70 \text{ dB}$, $B_s = 2,500 \text{ rad/s}$, and $B_p = 10,000 \text{ rad/s}$, determine the minimum required filter order to meet or exceed these specifications. Repeat the above for $A_p = 3 \text{ dB}$, $A_s = 70 \text{ dB}$, $B_s = 6,500 \text{ Hz}$, and $B_p = 26 \text{ kHz}$.
- 9.58 Given that the desired specifications of a Chebyshev Type I bandstop filter are as follows: $A_p = 1.2 \text{ dB}$, $A_s = 75 \text{ dB}$, $B_s = 3,500 \text{ rad/s}$, and $B_p = 7,000 \text{ rad/s}$, determine the minimum required filter order to meet or exceed these specifications. Repeat the above for $A_p = 1.2 \text{ dB}$, $A_s = 75 \text{ dB}$, $B_s = 6,500 \text{ Hz}$, and $B_p = 13 \text{ kHz}$.

- 9.59** Determine the *Filter Selectivity* of the bandstop filter in **Problem 9.57** with $\omega_o = 15,000 \text{ rad/s}$ in two ways: (a) by use of (9.45) and (3.7), and (b) computationally, using MATLAB.
- 9.60** Repeat **Problem 9.59** for $\omega_o = 10,000 \text{ rad/s}$.
- 9.61** Determine the *Filter Selectivity* of the bandstop filter in **Problem 9.58** with $\omega_o = 15,000 \text{ rad/s}$ in two ways: (a) by use of (9.45) and (4.9), and (b) computationally, using MATLAB.
- 9.62** Repeat **Problem 9.61** for $\omega_o = 10,000 \text{ rad/s}$.
- 9.63** Determine the *Shaping Factor* of the bandpass filter in **Problem 9.57** with $\omega_o = 15,000 \text{ rad/s}$ in two ways: (a) by use of (9.46) and (3.10), and (b) computationally, using MATLAB.
- 9.64** Repeat **Problem 9.63** for $\omega_o = 10,000 \text{ rad/s}$.
- 9.65** Determine the *Shaping Factor* of the bandpass filter in **Problem 9.58** with $\omega_o = 15,000 \text{ rad/s}$ in two ways: (a) by use of (9.46) and (4.12), and (b) computationally, using MATLAB.
- 9.66** Repeat **Problem 9.65** for $\omega_o = 10,000 \text{ rad/s}$.
- 9.67** Using (9.48), and the procedure in *Example 9.22*, compute the group delay of a 6th-order Butterworth bandstop filter at ω_{p_1} , ω_{p_2} , and *DC*, where $\omega_o = 5000 \text{ rad/s}$ and $B_p = 500 \text{ rad/s}$.
- 9.68** Using MATLAB, plot the magnitude frequency response, phase response, phase delay response, group delay response, unit impulse response, and unit step response for a 10th-order elliptic bandstop filter with $A_p = 1 \text{ dB}$, $A_s = 80 \text{ dB}$, $\omega_o = 1000 \text{ rad/s}$, and $B_p = 200 \text{ rad/s}$. That is, confirm the results plotted in **Figure 9.15** through **Figure 9.21** (*Example 9.23*).

CHAPTER 10

PASSIVE FILTERS

In this chapter passive filter implementation is introduced. Passive filters contain no active elements in the filter implementation. While passive filter implementation may have many forms, only ladder implementations are considered here, containing only resistors, capacitors and inductors. In fact, only two resistors will be included, no matter what the filter order is, or the filter type. The input resistor is associated with the driving source, and the resistor at the output is the load that the filter supplies energy to. The filter proper, between the source resistor and the load resistor, contains only capacitors and inductors in a ladder configuration.

10.1 INTRODUCTION

The procedure to design and realize an analog filter is to first design the filter, which is to obtain the transfer function, and then determine a desirable implementation, which when realized in hardware completes the procedure. A somewhat complete summary of the procedure may be expressed as follows:

Obtain the desired specifications of the filter.

The specifications may be given by a project engineer, or may be obtained by engineering judgment, based upon the application of the filter. The specifications generally include A_p , A_s , ω_p , and ω_s , if it is a lowpass or highpass filter, or similar specifications if it is a bandpass, or bandstop filter. If the filter is not frequency-selective, but rather, for example, for time delay, such as a Bessel filter, then the specifications would differ. Other possibilities may include specifications as to maximum group delay, settling time for the unit impulse response, etc. The specifications will likely also include the filter type, such as Butterworth, unless the selection of the filter type is considered part of the design.

Design the filter.

The next step is to design the filter, that is, starting with the specifications obtain the transfer function. The design may be obtained by use of MATLAB, using

functions that come with MATLAB, and also perhaps special functions on the disk that accompanies this book. This design procedure would likely also include detailed analysis of the proposed design¹, to reveal as much detail as possible about the proposed design's performance prior to implementation.

Implement the filter.

Implementation may be thought of as arriving at a schematic diagram, or some other representation, of a hardware realization of the filter. Realization may be thought of as the actual physical construction of the filter, although implementation and realization may be terms often used interchangeably. Analog filters may be implemented in many ways. One way to categorize implementation is in terms of *passive* or *active*. *Passive* indicates that there are no active elements in the filter implementation. This implementation may be made up of *R*'s, *L*'s and *C*'s. However, analog filters that are also passive, but more specialized, would include surface acoustic wave (SAW) filters, mechanical resonators and quartz crystal filters (Sheahan and Johnson, 1977). Passive filter implementation, for lossless ladder *LC* implementation, is the subject of this chapter. *Active* indicates that the implementation includes active elements, such as operational amplifiers (op amps), or possibly other active elements such as transistors. Active filter implementation is the subject of the next chapter.

A summary of passive filter implementation, as presented in this chapter, is as follows:

Obtain the lowpass prototype implementation.

After obtaining the desired specifications of the filter, and designing the filter, the lowpass prototype implementation may be obtained using, for example, the procedure given in **Section 10.2** below, or from a table. Several such tables are given in this chapter. Usually, element values in tables are normalized for a critical frequency (ω_c , ω_p , etc.) of unity, and for either a source resistor or load resistor of unity.

Frequency transform and impedance scale the implementation.

As will be developed in this chapter, the frequency transformations of **Chapter 9** may be applied directly to the element values of the lowpass prototype implementation. In addition, impedance scaling (reviewed in this chapter) may be applied for desirable element values.

Implementation is included in this book for the sake of practical completeness. Whereas it is hoped that the theoretical aspects of analog filter design, which has

¹For illustration, see *Example 9.6*, *9.15*, or *9.23*.

occupied the majority of this book to this point, will be judged to be somewhat thorough, implementation is but briefly presented. Any filter design may be realized in a wide variety of implementations. The passive and active filter implementation methods presented in this chapter and the next, while limited, are practical and useful methods.

10.2 CONTINUED-FRACTION LADDER IMPLEMENTATION

More detail on the implementation developed in this section may be found elsewhere (Van Valkenburg, 1960; Weinberg, 1962). Refer to **Figure 10.1**, where the lossless passive filter includes only capacitors and inductors, and ideally, therefore, dissipates no energy. Therefore, $P_{in} = P_{out}$, where P_{out} is the power dissipated in R_L , and P_{in} is the input power to the lossless passive filter. The symbols, with argument $j\omega$, may be thought of either as AC steady state phasors, or as Fourier transforms. Note that $P_{in} = P_{out}$ implies the following:

$$R_{11} |I_1(j\omega)|^2 = \frac{|V_o(j\omega)|^2}{R_L} \quad (10.1)$$

But, since

$$I_1(j\omega) = \frac{V_i(j\omega)}{R_S + Z_{11}} \quad ,$$

it follows that

$$|I_1(j\omega)|^2 = \frac{|V_i(j\omega)|^2}{|R_S + Z_{11}|^2} \quad (10.2)$$

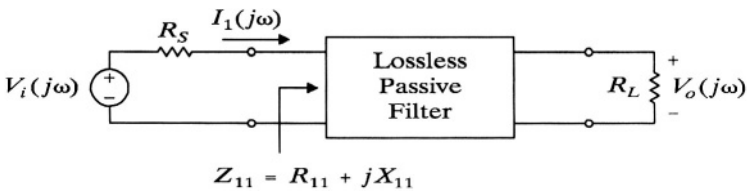


Figure 10.1 Illustration of a lossless, passive filter, referred to in the development of the continued-fraction ladder implementation method.

Substituting (10.2) into (10.1), it follows that

$$\left| \frac{V_o(j\omega)}{V_i(j\omega)} \right|^2 = \frac{R_L R_{11}}{|R_S + Z_{11}|^2} \quad (10.3)$$

Let $|A(j\omega)|^2$ be defined as follows:

$$|A(j\omega)|^2 = 1 - 4 \frac{R_S}{R_L} \left| \frac{V_o(j\omega)}{V_i(j\omega)} \right|^2 \quad (10.4)$$

Substituting (10.3) into (10.4), and making use of $Z_{11} = R_{11} + jX_{11}$, and simplifying, results in the following:

$$|A(j\omega)|^2 = \frac{|R_S - Z_{11}|^2}{|R_S + Z_{11}|^2} \quad (10.5)$$

Since $|A(j\omega)|^2 = A(j\omega)A(-j\omega)$, (10.5) implies that

$$A(j\omega) = \pm \frac{R_S - Z_{11}}{R_S + Z_{11}} \quad (10.6)$$

Equation (10.6), in terms of the Laplace variable s , may be expressed as follows:

$$A(s) = \pm \frac{R_S - Z_{11}(s)}{R_S + Z_{11}(s)} \quad (10.7)$$

From (10.7), it follows that

$$Z_{11}(s) = R_S \frac{1 - A(s)}{1 + A(s)} \quad (10.8)$$

or

$$Z_{11}(s) = R_S \frac{1 + A(s)}{1 - A(s)} \quad (10.9)$$

Combining (10.8) and (10.9):

$$Z_{11}(s) = R_S \left[\frac{1 - A(s)}{1 + A(s)} \right]^{\pm 1} \quad (10.10)$$

To illustrate how $A(s)$ can be known for a particular filter type, and therefore $Z_{11}(s)$ can be known, consider a Butterworth response. The Butterworth lowpass response, in terms of the variables used here, may be expressed as follows:

$$\left| \frac{V_o(j\omega)}{V_i(j\omega)} \right|^2 = \frac{K^2}{1 + (\omega/\omega_c)^{2N}} \quad (10.11)$$

For simplicity, consider $\omega_c = 1$, and $R_S = R_L$, and therefore, by reference to **Figure 10.1**, $K = 1/2$. Substituting (10.11) into (10.4), and using $\omega_c = 1$ and $K = 1/2$, and simplifying, results in

$$|A(j\omega)|^2 = \frac{\omega^{2N}}{1 + \omega^{2N}} \quad (10.12)$$

From **Section 2.7**, **Section 3.4**, and (10.12) it follows that

$$A(s)A(-s) = \frac{(s/j)^{2N}}{1 + (s/j)^{2N}} = \frac{s^N}{\mathcal{P}_N(s)} \frac{(-s)^N}{\mathcal{P}_N(-s)}$$

where $\mathcal{P}_N(s)$ is an N th-order Butterworth polynomial. For example, from **Example 3.3**, $\mathcal{P}_3(s) = s^3 + 2s^2 + 2s + 1$. Therefore, $A(s)$, for a Butterworth lowpass filter with $\omega_c = 1$, $R_S = R_L$, and $K = 1/2$, may be expressed as follows:

$$A(s) = \frac{s^N}{\mathcal{P}_N(s)} \quad (10.13)$$

Given (10.13), then (10.10) may be used to find two acceptable expressions for $Z_{11}(s)$. This, in turn, will result in two circuit implementations.

Example 10.1

Suppose the desired filter to be implemented with a passive ladder circuit is a third-order Butterworth lowpass filter with $\omega_c = 1 \text{ rad/s}$ and $R_S = R_L = 1 \Omega$. Note that later in this chapter impedance and frequency scaling applied directly to the circuit will allow for any desired ω_c and $R_S = R_L$ values. Therefore, from (10.13),

$$A(s) = \frac{s^3}{s^3 + 2s^2 + 2s + 1}$$

From (10.10),

$$Z_{11}(s) = \left[\frac{2s^2 + 2s + 1}{2s^3 + 2s^2 + 2s + 1} \right]^{\pm 1}$$

Case I

Consider the following form for $Z_{11}(s)$:

$$Z_{11}(s) = \frac{2s^2 + 2s + 1}{2s^3 + 2s^2 + 2s + 1} = \frac{1}{\frac{2s^3 + 2s^2 + 2s + 1}{2s^2 + 2s + 1}}$$

Note that

$$\frac{2s^3 + 2s^2 + 2s + 1}{2s^2 + 2s + 1} = s + \frac{s + 1}{2s^2 + 2s + 1},$$

and

$$\frac{2s^2 + 2s + 1}{s + 1} = 2s + \frac{1}{s + 1}.$$

Therefore, it follows that

$$Z_{11}(s) = \frac{2s^3 + 2s^2 + 2s + 1}{2s^3 + 2s^2 + 2s + 1} = \frac{1}{s + \frac{1}{2s + \frac{1}{s + 1}}}. \quad (10.14)$$

This approach is referred to as the continued-fraction method. Equation (10.14) implies the circuit shown in **Figure 10.2**.

Case II

Consider the following form for $Z_{11}(s)$:

$$Z_{11}(s) = \frac{2s^3 + 2s^2 + 2s + 1}{2s^2 + 2s + 1} = s + \frac{1}{2s + \frac{1}{s + 1}}. \quad (10.15)$$

Equation (10.15) implies the circuit shown in **Figure 10.3**. □

Because of the complexity of this procedure, tables have been generated for common filter types. Below, in **Table 10.1** through **Table 10.8**, element values are given for Butterworth lowpass prototype filters, selected Chebyshev Type I filters, Bessel filters, and selected elliptic filters. Refer to **Figure 10.4** through **Figure 10.7** for implementation circuit schematic diagrams for the Butterworth, Chebyshev, and

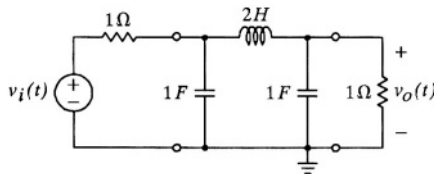


Figure 10.2 Case I implementation for the 3rd-order Butterworth lowpass filter of **Example 10.1**: $\omega_c = 1 \text{ rad/s}$, and $R_S = R_L = 1 \Omega$.

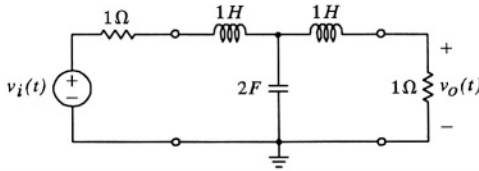


Figure 10.3 Case II implementation for the 3rd-order Butterworth lowpass filter of *Example 10.1*: $\omega_c = 1 \text{ rad/s}$, and $R_S = R_L = 1 \Omega$.

Bessel filters. Refer to **Figure 10.8** and **Figure 10.9** for implementation circuit schematic diagrams for the elliptic filters.

In **Table 10.1**, since $R_S = 1 \Omega$, it follows that the DC gain of each of the filters in the table is 0.5. Therefore, ω_c , which is unity for the table element values, is the radian frequency where the gain is $1/\sqrt{2}$ times the DC gain, or $1/(2\sqrt{2})$. Also note that for any order of 2 through 10, there are element values for two implementations. For example, for $N = 3$, element values for **Figure 10.7** may be read from the table as $C_1 = C_3 = 1.0 \text{ F}$ and $L_2 = 2.0 \text{ H}$. However, element values for **Figure 10.5** may be read from the table as $L'_1 = L'_3 = 1.0 \text{ H}$ and $C'_2 = 2.0 \text{ F}$.

In **Table 10.2** through **Table 10.5**, A_p is the value of the passband ripple, and it is also the value of the attenuation relative to the peak of the passband response at $\omega = \omega_p = 1 \text{ rad/s}$. All of the equations in **Chapter 4** apply to the filters in these tables. That is, for example, ω_c may be computed using (4.7), the value of ω_s , for a given value of A_s (and A_p and N), may be computed from (4.14), etc.

Similarly, in **Table 10.6**, all of the filters are normalized for a group delay at DC of unity. If it is desired to implement a Bessel filter as a frequency selective filter,

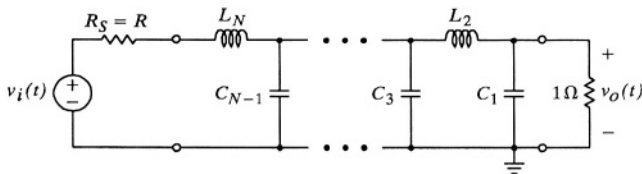


Figure 10.4 Passive lowpass prototype circuit configuration, including the source and load resistances, for even orders, for Butterworth, Chebyshev Type I, and Bessel filters, and beginning with a series inductance.

TABLE 10.1Element values for Butterworth passive lowpass filter implementation:[†] $\omega_c = 1 \text{ rad/s}$, $R_s = 1 \Omega$. See **Figure 10.4** through **Figure 10.7**.

Table entry values are in Farads and Henrys.

Elements in **Figure 10.4** (even order) and **Figure 10.7** (odd order)

N	C_1	L_2	C_3	L_4	C_5	L_6	C_7	L_8	C_9	L_{10}
2	1.4142	1.4142								
3	1.0000	2.0000	1.0000							
4	0.7654	1.8478	1.8478	0.7654						
5	0.6180	1.6180	2.0000	1.6180	0.6180					
6	0.5176	1.4142	1.9319	1.9319	1.4142	0.5176				
7	0.4450	1.2470	1.8019	2.0000	1.8019	1.2470	0.4450			
8	0.3902	1.1111	1.6629	1.9616	1.9616	1.6629	1.1111	0.3902		
9	0.3473	1.0000	1.5321	1.8794	2.0000	1.8794	1.5321	1.0000	0.3473	
10	0.3129	0.9080	1.4142	1.7820	1.9754	1.9754	1.7820	1.4142	0.9080	0.3129
N	L'_1	C'_2	L'_3	C'_4	L'_5	C'_6	L'_7	C'_8	L'_9	C'_{10}

Elements in **Figure 10.5** (odd order) and **Figure 10.6** (even order)

[†]The data in this table is from Louis Weinberg, *Network Analysis and Synthesis*, McGraw-Hill Book Company, New York, 1962.

as opposed to a delay circuit, the MATLAB function *BESSELDD* may be used to design a Bessel filter with a normalized time delay, as the filters are in the table. Then a MATLAB magnitude frequency response may be plotted to obtain the frequency (ω_p) at some desired attenuation (A_p). This method may be used to inform the designer as to what frequency scaling should be applied to achieve a desired ω_p value for the non-normalized circuit.

In **Table 10.7** and **Table 10.8** only a few selected elliptic filters are shown. Since passband ripple and minimum attenuation in the stopband are both design

TABLE 10.2

Element values for Chebyshev Type I passive lowpass filter implementation:[†]

$A_p = 0.1 \text{ dB}$, $\omega_p = 1 \text{ rad/s}$, $R = 0.5$ (R_s is either 0.5Ω or 2Ω).

See Figure 10.4 through Figure 10.7. Table entry values are in Farads and Henrys.

Elements in **Figure 10.4** (even order) and **Figure 10.7** (odd order)

N	C_1	L_2	C_3	L_4	C_5	L_6	C_7	L_8	C_9	L_{10}
2	1.5715	0.2880								
3	2.2746	0.6035	1.3341							
4	2.3545	0.7973	2.6600	0.3626						
5	2.6921	0.8042	3.3882	0.6853	1.4572					
6	2.5561	0.8962	3.3962	0.8761	2.8071	0.3785				
7	2.8260	0.8560	3.7594	0.8685	3.5246	0.7050	1.4932			
8	2.6324	0.9285	3.5762	0.9619	3.5095	0.8950	2.8547	0.3843		
9	2.8839	0.8762	3.8788	0.9121	3.8660	0.8836	3.5703	0.7127	1.5084	
10	2.6688	0.9429	3.6461	0.9887	3.6707	0.9765	3.5472	0.9027	2.8761	0.3870
N	L'_1	C'_2	L'_3	C'_4	L'_5	C'_6	L'_7	C'_8	L'_9	C'_{10}

Elements in **Figure 10.5** (odd order) and **Figure 10.6** (even order)

[†]The data in this table is from Louis Weinberg, *Network Analysis and Synthesis*, McGraw-Hill Book Company, New York, 1962.

parameters for elliptic filters, the number and size of elliptic implementation tables could be large. Those filters selected for inclusion in **Table 10.7** and **Table 10.8** all have minimum stopband attenuations around 60 dB . There are two reasons for this. First, 60 dB is a practical value for filter use, and second, by having all the A_s values in the two tables similar, the influence of filter order and A_p on ω_s becomes clear. Note that it is not necessary for ω_s to be included in the tables, as the values could be computed using the MATLAB function *ELLIPWS* (see **Section 6.4**), but they are included to illustrate its dependance on the filter order and A_p .

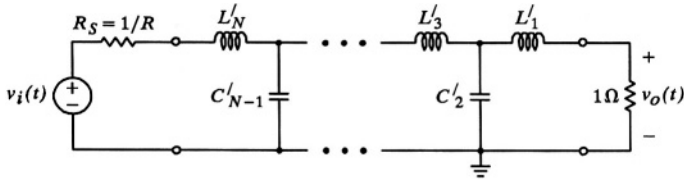


Figure 10.5 Passive lowpass prototype circuit configuration, including the source and load resistances, for **odd orders**, for Butterworth, Chebyshev Type I, and Bessel filters, and beginning with a **series inductance**.

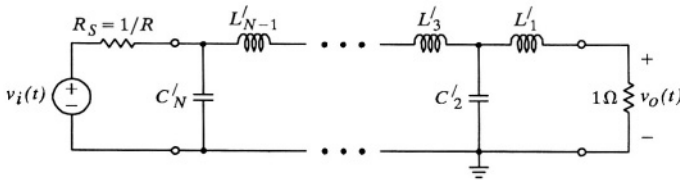


Figure 10.6 Passive lowpass prototype circuit configuration, including the source and load resistances, for **even orders**, for Butterworth, Chebyshev Type I, and Bessel filters, and beginning with a **shunt capacitance**.

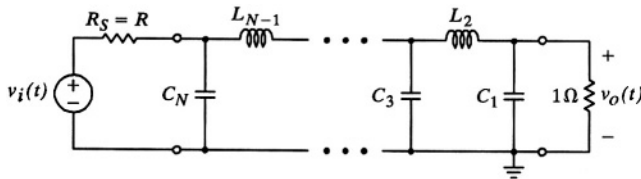


Figure 10.7 Passive lowpass prototype circuit configuration, including the source and load resistances, for **odd orders**, for Butterworth, Chebyshev Type I, and Bessel filters, and beginning with a **shunt capacitance**.

TABLE 10.3

Element values for Chebyshev Type I passive lowpass filter implementation:[†]

$$A_p = 0.25 \text{ dB}, \omega_p = 1 \text{ rad/s}, R = 0.5 \text{ (} R_s \text{ is either } 0.5 \Omega \text{ or } 2 \Omega \text{)}.$$

See **Figure 10.4** through **Figure 10.7**. Table entry values are in Farads and Henrys.

Elements in **Figure 10.4** (even order) and **Figure 10.7** (odd order)

N	C_1	L_2	C_3	L_4	C_5	L_6	C_7	L_8	C_9	L_{10}
2	1.7288	0.4104								
3	2.5965	0.6465	1.7402							
4	2.2884	0.9039	2.7832	0.4930						
5	2.8983	0.8007	3.6926	0.7162	1.8152					
6	2.4162	0.9771	3.2941	0.9837	2.9094	0.5100				
7	3.0294	0.8341	4.0204	0.8546	3.8585	0.7340	1.9007			
8	2.4631	1.0000	3.4072	1.0463	3.3925	1.0015	2.9490	0.5161		
9	3.0724	0.8478	4.1088	0.8843	4.1199	0.8670	3.8989	0.7404	1.9157	
10	2.4852	1.0100	3.4501	1.0651	3.4927	1.0606	3.4235	1.0085	2.9666	0.5190
N	L'_1	C'_2	L'_3	C'_4	L'_5	C'_6	L'_7	C'_8	L'_9	C'_{10}

Elements in **Figure 10.5** (odd order) and **Figure 10.6** (even order)

[†]The data in this table is from Louis Weinberg, *Network Analysis and Synthesis*, McGraw-Hill Book Company, New York, 1962.

TABLE 10.4Element values for Chebyshev Type I passive lowpass filter implementation:[†] $A_p = 0.5 \text{ dB}$, $\omega_p = 1 \text{ rad/s}$, $R = 0.5$ (R_s is either 0.5Ω or 2Ω).See **Figure 10.4** through **Figure 10.7**. Table entry values are in Farads and Henrys.Elements in **Figure 10.4** (even order) and **Figure 10.7** (odd order)

N	C_1	L_2	C_3	L_4	C_5	L_6	C_7	L_8	C_9	L_{10}
2	1.5132	0.6538								
3	2.9431	0.6503	2.1903							
4	1.8158	1.1328	2.4881	0.7732						
5	3.2228	0.7645	4.1228	0.7116	2.3197					
6	1.8786	1.1884	2.7589	1.2403	2.5976	0.7976				
7	3.3055	0.7899	4.3575	0.8132	4.2419	0.7252	2.3566			
8	1.9012	1.2053	2.8152	1.2864	2.8479	1.2628	2.6310	0.8063		
9	3.3403	0.7995	4.4283	0.8341	4.4546	0.8235	4.2795	0.7304	2.3719	
10	1.9117	1.2127	2.8366	1.2999	2.8964	1.3054	2.8744	1.2714	2.6456	0.8104
N	L'_1	C'_2	L'_3	C'_4	L'_5	C'_6	L'_7	C'_8	L'_9	C'_{10}

Elements in **Figure 10.5** (odd order) and **Figure 10.6** (even order)

[†]The data in this table is from Louis Weinberg, *Network Analysis and Synthesis*, McGraw-Hill Book Company, New York, 1962.

TABLE 10.5

Element values for Chebyshev Type I passive lowpass filter implementation:[†]

$$A_p = 1 \text{ dB}, \omega_p = 1 \text{ rad/s}, R = 1/3 \text{ (} R_S \text{ is either } 1/3 \Omega \text{ or } 3 \Omega \text{).}$$

See Figure 10.4 through Figure 10.7. Table entry values are in Farads and Henrys.

Elements in Figure 10.4 (even order) and Figure 10.7 (odd order)

N	C_1	L_2	C_3	L_4	C_5	L_6	C_7	L_8	C_9	L_{10}
2	2.5721	0.4702								
3	4.9893	0.4286	3.8075							
4	3.0355	0.7929	3.7589	0.5347						
5	5.3830	0.4915	6.6673	0.4622	3.9944					
6	3.1307	0.8287	4.1451	0.8467	3.8812	0.5475				
7	5.4978	0.5050	6.9839	0.5177	6.8280	0.4696	4.0473			
8	3.1647	0.8395	4.2237	0.8764	4.2404	0.8580	3.9186	0.5520		
9	5.5459	0.5101	7.0783	0.5288	7.1141	0.5232	6.8785	0.4724	4.0693	
10	3.1806	0.8442	4.2532	0.8851	4.3088	0.8857	4.2691	0.8623	3.9349	0.5541
N	L'_1	C'_2	L'_3	C'_4	L'_5	C'_6	L'_7	C'_8	L'_9	C'_{10}

Elements in Figure 10.5 (odd order) and Figure 10.6 (even order)

[†]The data in this table is from Louis Weinberg, *Network Analysis and Synthesis*, McGraw-Hill Book Company, New York, 1962.

TABLE 10.6

Element values for Bessel passive lowpass filter implementation:[†]
 $t_{gd}(0) = 1 \text{ s}$, $R_S = 1 \Omega$. See **Figure 10.4** through **Figure 10.7**.
 Table entry values are in Farads and Henrys.

Elements in **Figure 10.4** (even order) and **Figure 10.7** (odd order)

N	C_1	L_2	C_3	L_4	C_5	L_6	C_7	L_8	C_9	L_{10}
2	1.5774	0.4226								
3	1.2550	0.5528	0.1922							
4	1.0598	0.5116	0.3181	0.1104						
5	0.9303	0.4577	0.3312	0.2090	0.0718					
6	0.8377	0.4116	0.3158	0.2364	0.1480	0.0505				
7	0.7677	0.3744	0.2944	0.2378	0.1778	0.1104	0.0375			
8	0.7125	0.3446	0.2735	0.2297	0.1867	0.1387	0.0855	0.0289		
9	0.6678	0.3203	0.2547	0.2184	0.1859	0.1506	0.1111	0.0682	0.0230	
10	0.6305	0.3002	0.2384	0.2066	0.1808	0.1539	0.1240	0.0911	0.0557	0.0187
N	L'_1	C'_2	L'_3	C'_4	L'_5	C'_6	L'_7	C'_8	L'_9	C'_{10}

Elements in **Figure 10.5** (odd order) and **Figure 10.6** (even order)

[†]The data in this table is from Louis Weinberg, *Network Analysis and Synthesis*, McGraw-Hill Book Company, New York, 1962.

TABLE 10.7

Element values for selected elliptic passive lowpass filter implementation:[†]
 $\omega_p = 1 \text{ rad/s}$, orders 3 and 5. See **Figure 10.8** and **Figure 10.9**.
 Table entry values are in Farads and Henrys.

Elements in Figure 10.8

N	A_p	A_s	ω_s	C_1	C_2	L_2	C_3	C_4	L_4	C_5
3	0.011	61.6	11.474	0.6354	0.0059	0.9711	0.6354			
3	0.028	61.0	9.567	0.7694	0.0078	1.0587	0.7694			
3	0.044	62.9	9.567	0.8479	0.0075	1.0944	0.8479			
3	0.099	62.5	8.206	1.0212	0.0098	1.1355	1.0212			
3	0.177	61.5	7.185	1.1796	0.0128	1.1404	1.1796			
5	0.011	61.2	2.559	0.7321	0.0441	1.261	1.496	0.1211	1.149	0.662
5	0.028	61.6	2.366	0.8574	0.0505	1.302	1.612	0.1380	1.182	0.779
5	0.044	60.1	2.203	0.9265	0.0587	1.307	1.666	0.1607	1.173	0.836
5	0.099	60.4	2.062	1.089	0.0681	1.299	1.803	0.1858	1.158	0.985
5	0.177	61.4	2.000	1.241	0.0745	1.271	1.936	0.2024	1.132	1.129
5	0.500	66.1	2.000	1.6370	0.0813	1.1639	2.3142	0.2187	1.047	1.515
5	1.000	69.4	2.000	2.0559	0.0915	1.0339	2.7357	0.2449	0.935	1.919
N	A_p	A_s	ω_s	L'_1	L'_2	C'_2	L'_3	L'_4	C'_4	L'_5

Elements in Figure 10.9

[†]The data in this table is from Grant E. Hansell, *Filter Design and Evaluation*, Van Nostrand Reinhold Company, New York, 1969, except for the 0.5 dB and 1.0 dB passband ripple entries, which are from F.W. Stephenson, *RC Active Filter Design Handbook*, John Wiley & Sons, New York, 1985.

TABLE 10.8
 Element values for selected elliptic passive lowpass filter implementation.[†]
 $\omega_p = 1 \text{ rad/s}$, orders 7 and 9. See **Figure 10.8** and **Figure 10.9**.
 Table entry values are in Farads and Henrys.

N	A_p	A_s	ω_s	C_1	C_2	L_2	C_3	C_4	L_4	C_5	C_6	L_6	C_7	C_8	L_8	C_9
7	0.011	60.6	1.466	0.7387	0.0859	1.296	1.435	0.3945	1.136	1.332	0.3081	1.031	0.5612			
7	0.028	60.0	1.390	0.8566	0.0976	1.317	1.494	0.4598	1.087	1.370	0.3477	1.033	0.6599			
7	0.044	60.5	1.367	0.9270	0.1021	1.319	1.536	0.4856	1.064	1.402	0.3614	1.034	0.7236			
7	0.099	61.2	1.325	1.085	0.1134	1.299	1.630	0.5495	1.003	1.474	0.3977	1.012	0.8640			
7	0.177	60.9	1.287	1.229	0.1277	1.258	1.711	0.6286	0.9310	1.531	0.4462	0.9684	0.9748			
7	0.500	58.2	1.200	1.5848	0.1810	1.1123	1.8924	0.9241	0.7312	1.6375	0.6329	0.8139	1.2586			
7	1.000	61.4	1.200	1.9917	0.2045	0.9847	2.2280	1.0486	0.6444	1.9272	0.7055	0.7301	1.6254			
9	0.100	58.7	1.100	1.0723	0.1477	1.2774	1.4640	0.9232	0.7905	1.0015	1.2847	0.6358	1.1496	0.5763	0.8954	0.7702
9	0.500	65.9	1.100	1.6067	0.1668	1.1313	1.8556	1.0651	0.6852	1.2576	1.4676	0.5565	1.4599	0.6257	0.8248	1.2716
9	1.000	69.2	1.100	2.0150	0.1888	0.9998	2.1805	1.2150	0.6006	1.4734	1.6693	0.4893	1.7169	0.6996	0.7376	1.6378
N	A_p	A_s	ω_s	L'_1	L'_2	C'_2	L'_3	L'_4	C'_4	L'_5	L'_6	C'_6	L'_7	L'_8	C'_8	L'_9

[†]The data in this table is from Grant E. Hansell, *Filter Design and Evaluation*, Van Nostrand Reinhold Company, New York, 1969, except for the cases where the passband ripple is 0.100, 0.500, or 1.00 dB, which are from F.W. Stephenson, *RC Active Filter Design Handbook*, John Wiley & Sons, New York, 1985.

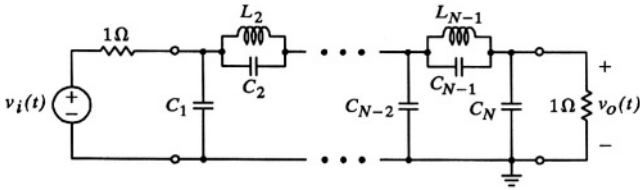


Figure 10.8 Passive lowpass prototype circuit configuration, including the source and load resistances, for elliptic filters beginning with a **shunt capacitance**.

10.3 FREQUENCY TRANSFORMATION CIRCUIT OPERATIONS

As was the case in **Chapter 9**, four types of frequency transformation will be considered here: lowpass-to-lowpass, lowpass-to-highpass, lowpass-to-bandpass, and lowpass-to-bandstop. It is assumed that the initial lowpass filter implementation is the prototype, such as obtained from the tables in the previous section. What differs here, from that presented in **Chapter 9**, is that the transfer function of the frequency-transformed filter need not be found, but rather the concepts of frequency transformation are applied directly to the prototype circuit elements.

Lowpass-to-Lowpass Transformation

As presented in **Section 9.1**, the lowpass-to-lowpass transformation is accomplished by replacing every s in the transfer function of the prototype by s/k_f ,

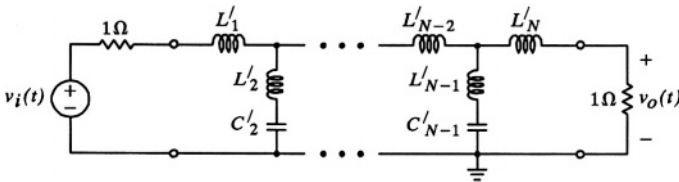


Figure 10.9 Passive lowpass prototype circuit configuration, including the source and load resistances, for elliptic filters beginning with a **series inductance**.

where k_f is the frequency scaling factor. Since, in an s domain circuit diagram of the prototype filter, the complex impedance of an inductor is expressed as sL and that of a capacitor is expressed as $1/(sC)$, it follows that in the s domain circuit diagram of the frequency scaled filter the impedance of an inductor would be expressed as sL/k_f and that of a capacitor would be expressed as $1/(sC/k_f)$. Clearly the resistors in the frequency scaled filter circuit diagram would be the same as in the prototype circuit diagram, as resistor values are not a function of s . Therefore, the procedure to frequency scale a prototype circuit diagram may be summarized as follows:

- Divide all inductor values by k_f , the frequency scaling factor.
- Divide all capacitor values by k_f .
- Leave all resistor values unchanged.

Having applied the above operations to the element values of a lowpass prototype filter circuit schematic diagram, the transformed circuit will be frequency scaled by k_f . Therefore all of the results and observations in **Section 9.1** will apply to the frequency scaled circuit.

Lowpass-to-Highpass Transformation

As presented in **Section 9.2**, the lowpass-to-highpass transformation is accomplished by replacing every s in the transfer function of the prototype by k_f/s , where k_f is the frequency scaling factor. Since, in an s domain circuit diagram of the prototype filter, the complex impedance of an inductor is expressed as sL , it follows that in the s domain circuit diagram of the highpass filter the impedance of an “inductor” would be expressed as $1/(s(1/(k_fL)))$, which represents a capacitor with a value of $1/(k_fL)$. In the prototype filter, a capacitor would be expressed as $1/(sC)$: in the s domain circuit diagram of the highpass filter the impedance of a “capacitor” would be expressed as $s(1/(k_fC))$, which represents an inductor with a value of $1/(k_fC)$. Clearly the resistors in the highpass filter circuit diagram would be the same as in the prototype circuit diagram, as resistor values are not a function of s . Therefore, the procedure to transform a lowpass prototype circuit diagram into a highpass circuit diagram may be summarized as follows:

- Replace each inductor in the lowpass prototype circuit by a capacitor with a value of $1/(k_fL)$.
- Replace each capacitor in the lowpass prototype circuit by an inductor with a value of $1/(k_fC)$.

- Leave all resistor values unchanged.

Having applied the above operations to the element values of a lowpass prototype filter circuit schematic diagram, the transformed circuit will be a highpass filter frequency scaled by k_f . Therefore all of the results and observations in **Section 9.2** will apply to the highpass circuit.

Lowpass-to-Bandpass Transformation

As presented in **Section 9.3**, the lowpass-to-bandpass transformation is accomplished by replacing every s in the transfer function of the prototype by $(s^2 + \omega_o^2)/(B_p s)$, where ω_o is the “center frequency” and B_p is the bandwidth. Since, in an s domain circuit diagram of the prototype filter, the complex impedance of an inductor is expressed as sL , it follows that in the s domain circuit diagram of the bandpass filter the impedance of an “inductor” would be expressed as $sL/B_p + 1/(sB_p/(L\omega_o^2))$, which represents an inductor with a value of L/B_p in series with a capacitor with a value of $B_p/(L\omega_o^2)$. In the prototype filter, a capacitor would be expressed as $1/(sC)$: in the s domain circuit diagram of the bandpass filter the impedance of a “capacitor” would be expressed as $(B_p/C)s/(s^2 + \omega_o^2)$, which represents an inductor with a value of $B_p/(C\omega_o^2)$ in parallel with a capacitor with a value of C/B_p . Again, the resistors in the bandpass filter circuit diagram would be the same as in the prototype circuit diagram, as resistor values are not a function of s . Therefore, the procedure to transform a lowpass prototype circuit diagram into a bandpass circuit diagram may be summarized as follows, where it is assumed that $\omega_p = 1$ for the lowpass prototype circuit:

- Replace each inductor in the lowpass prototype circuit by an inductor with a value of L/B_p in series with a capacitor with a value of $B_p/(L\omega_o^2)$.
- Replace each capacitor in the lowpass prototype circuit by an inductor with a value of $B_p/(C\omega_o^2)$ in parallel with a capacitor with a value of C/B_p .
- Leave all resistor values unchanged.

Having applied the above operations to the element values of a lowpass prototype filter circuit schematic diagram, the transformed circuit will be a bandpass filter with a center frequency of ω_o and have a bandwidth of B_p . Therefore all of the results and observations in **Section 9.3** will apply to the bandpass circuit.

Lowpass-to-Bandstop Transformation

As presented in **Section 9.4**, the lowpass-to-bandstop transformation is accomplished by replacing every s in the transfer function of the prototype by $(B_p s)/(s^2 + \omega_o^2)$, where ω_o is the “center frequency” and B_p is the bandwidth. Since, in an s domain circuit diagram of the prototype filter, the complex impedance of an inductor is expressed as sL , it follows that in the s domain circuit diagram of the bandstop filter the impedance of an “inductor” would be expressed as $B_p L s/(s^2 + \omega_o^2)$, which represents an inductor with a value of $B_p L/\omega_o^2$ in parallel with a capacitor with a value of $1/(B_p L)$. In the prototype filter, a capacitor would be expressed as $1/(sC)$: in the s domain circuit diagram of the bandstop filter the impedance of a “capacitor” would be expressed as $(1/(B_p C))s + 1/(s((B_p C)/\omega_o^2))$, which represents an inductor with a value of $1/(B_p C)$ in series with a capacitor with a value of $(B_p C)/\omega_o^2$. Again, the resistors in the bandstop filter circuit diagram would be the same as in the prototype circuit diagram, as resistor values are not a function of s . Therefore, the procedure to transform a lowpass prototype circuit diagram into a bandstop circuit diagram may be summarized as follows, where it is assumed that $\omega_p = 1$ for the lowpass prototype circuit::

- Replace each inductor in the lowpass prototype circuit by an inductor with a value of $(B_p L)/\omega_o^2$ in parallel with a capacitor with a value of $1/(B_p L)$.
- Replace each capacitor in the lowpass prototype circuit by an inductor with a value of $1/(B_p C)$ in series with a capacitor with a value of $(B_p C)/\omega_o^2$.
- Leave all resistor values unchanged.

Having applied the above operations to the element values of a lowpass prototype filter circuit schematic diagram, the transformed circuit will be a bandstop filter with a center frequency of ω_o and have a bandwidth of B_p . Therefore all of the results and observations in **Section 9.4** will apply to the bandstop circuit.

10.4 IMPEDANCE SCALING

If the output and input variables for a circuit have the same units, for example v_o and v_s , then if all impedances in the circuit are equally scaled it will have no effect on the circuit transfer function. Therefore all measures of circuit performance considered in this book, such as magnitude frequency response, phase response, unit impulse response, etc., will not be affected.

If the output and input variables have units that differ, for example v_o and i_s , then some measures of circuit performance will require modification. For example, if the output is v_o and the input is i_s , and all impedances are increased by 10, then the amplitude of the unit impulse response would also be increased by a factor of 10. However, the time axis would be unchanged and the shape of the graph of $h(t)$ would be unchanged. For the remainder of this chapter it will be assumed that the input and output variables each have units of voltage.

Let the impedance scaling factor² be denoted as k_i . To scale the impedance of an inductor by the factor k_i is equivalent to multiplying the inductance value by k_i , since the impedance of an inductor is sL . To scale the impedance of a capacitor by the factor k_i is equivalent to dividing the capacitance value by k_i , since the impedance of a capacitor is $1/(sC)$. And, to scale the impedance of a resistor by the factor k_i is equivalent to multiplying the resistance value by k_i . Therefore, the procedure to impedance scale the elements in a circuit to achieve convenient or desired values may be summarized as follows:

- Multiply the value of each inductor in the circuit by k_i .
- Divide the value of each capacitor in the circuit by k_i .
- Multiply the value of each resistor in the circuit by k_i .

Having applied the above impedance scaling operations to the element values of a filter circuit schematic diagram, the impedance-scaled circuit, assuming input and output variables of v_s and v_o , the transfer function will be unaffected and all performance measures will be unchanged.

10.5 SUMMARY AND EXAMPLES OF PASSIVE FILTER IMPLEMENTATION

Passive filter implementation, as presented in this chapter, may be summarized as follows:

- Obtain the lowpass prototype circuit schematic diagram. For example by referring to one of the tables in this chapter. This assumes a starting point of knowing the desired filter type (Butterworth, etc.), the type of frequency selection (bandpass, etc.) or time delay, the desired order, and critical design parameters such as passband ripple, cutoff frequency, etc. That is, it is assumed that the desired filter has been fully

²Sometimes “impedance scaling” is denoted “magnitude scaling.”

specified. It should also have been designed and fully analyzed, as mentioned in **Section 10.1**.

- Perform the desired frequency transformation directly to the circuit elements of the lowpass prototype circuit diagram, as described in **Section 10.3**.
- Perform the desired impedance scaling to the circuit element values, as described in **Section 10.4**, to obtain convenient values, a specified load resistance value, etc.
- Perform computer simulation analysis on the proposed circuit diagram to test its accuracy. This step should always be performed prior to realization. A popular circuit simulation analysis tool is SPICE. If a full MATLAB analysis had already been conducted as part of the design procedure, only a brief SPICE analysis may be necessary as a check on the proposed circuit.

Example 10.2

Suppose it is desired to implement a Chebyshev Type I highpass filter that meets the following specifications with minimum order: passband ripple = 0.5 dB, $f_p = 1,000$ Hz, $f_s = 300$ Hz, $A_s = 60$ dB, and $R_L = 1,000$ Ω .

From (4.14) and noting comments on page 277, $N = 5$. From **Table 10.4** the lowpass prototype circuit element values for the circuit schematic diagram shown in **Figure 10.7** are as follows: $R_S = 0.5$ Ω , $C_1 = 3.2228$ F, $L_2 = 0.7645$ H, $C_3 = 4.1228$ F, $L_4 = 0.7116$ H, $C_5 = 2.3197$ F, and $R_L = 1$ Ω . Also from **Table 10.4** the lowpass prototype circuit element values for the circuit schematic diagram shown in **Figure 10.5** are as follows: $R_S = 2$ Ω , $L'_1 = 3.2228$ H, $C'_2 = 0.7645$ F, $L'_3 = 4.1228$ H, $C'_4 = 0.7116$ F, $L'_5 = 2.3197$ H, and $R_L = 1$ Ω .

Since it is desired that $f_p = 1,000$ Hz, it follows that $k_f = 2\pi \cdot 1,000 \cong 6,283.2$. Applying the lowpass-to-highpass transformation of **Section 10.3** to the circuit elements of **Figure 10.7**, it follows that $R_S = 0.5$ Ω , $L_1 = 49.38$ μ H, $C_2 = 208.18$ μ F, $L_3 = 38.60$ μ H, $C_4 = 223.66$ μ F, $L_5 = 68.61$ μ H, and $R_L = 1$ Ω . Applying the lowpass-to-highpass transformation of **Section 10.3** to the circuit elements of **Figure 10.5**, it follows that $R_S = 2$ Ω , $C'_1 = 49.38$ μ F, $L'_2 = 208.18$ μ H, $C'_3 = 38.60$ μ F, $L'_4 = 223.66$ μ H, $C'_5 = 68.61$ μ F, and $R_L = 1$ Ω .

Applying the desired impedance scaling factor of $k_i = 1,000$, according to **Section 10.4**, to the transformed element values of **Figure 10.7** results in $R_S = 500$ Ω , $L_1 = 49.38$ mH, $C_2 = 0.20818$ μ F, $L_3 = 38.60$ mH,

$C_4 = 0.22366 \mu F$, $L_5 = 68.61 mH$, and $R_L = 1,000 \Omega$. This final circuit is shown in **Figure 10.10**.

Applying the desired impedance scaling factor of $k_t = 1,000$, according to **Section 10.4**, to the transformed element values of **Figure 10.5** results in $R_S = 2,000 \Omega$, $C'_1 = 0.04938 \mu F$, $L'_2 = 208.18 mH$, $C'_3 = 0.0386 \mu F$, $L'_4 = 223.66 mH$, $C'_5 = 0.06861 \mu F$, and $R_L = 1,000 \Omega$. This final circuit is shown in **Figure 10.11**.

The magnitude frequency response, as obtained from SPICE analysis applied to **Figure 10.10** is shown in **Figure 10.12**. Note that the peak of the passband response is $-3.52 dB$. This peak passband response is attenuated from $0 dB$ due to the source resistance of 500Ω and the load resistance of $1,000 \Omega$: $20 \log(1,000/1,500) = -3.52$. Note also that for all frequencies less than $300 Hz$ the attenuation is greater than $63.52 dB$ (greater than $60 dB$ below the peak response). The magnitude frequency response for the circuit shown in **Figure 10.11** has the same shape as shown in **Figure 10.12**; however, the peak passband response is $-9.54 dB$: $20 \log(1,000/3,000) = -9.54$. □

Example 10.3

Suppose it is desired to implement a sixth-order elliptic bandpass filter that meets or exceeds the following specifications: passband ripple = $A_p = 0.1 dB$, minimum attenuation in the stopband relative to the peak response = $A_s = 62 dB$, $B_p = 1,000 Hz$, $B_s = 8,500 Hz$, and $f_o = 10,000 Hz$. It can easily be seen that the fourth entry in **Table 10.7** will provide a filter that slightly exceeds the stated specifications: the passband ripple will be $0.099 dB$, the minimum attenuation in the stopband will be $62.5 dB$, and B_s will be $8,206 Hz$ for $B_p = 1,000 Hz$. Suppose it is also desired to use $1,000 \Omega$ resistors.

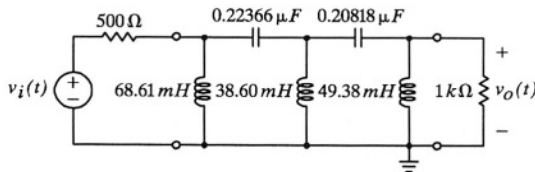


Figure 10.10 The passive 5th-order highpass implementation of the Chebyshev Type I filter of **Example 10.2** based on **Figure 10.7**, with $A_p = 0.5 dB$, $f_p = 1,000 Hz$, $f_s = 300 Hz$, and $A_s = 60 dB$.

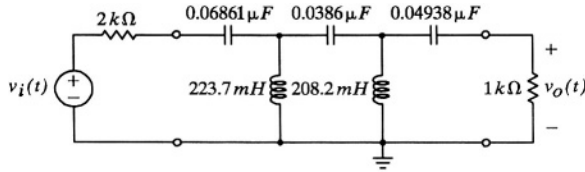


Figure 10.11 The passive 5th-order highpass implementation of the Chebyshev Type I filter of *Example 10.2* based on **Figure 10.5**, with $A_p = 0.5$ dB, $f_p = 1,000$ Hz, $f_s = 300$ Hz, and $A_s = 60$ dB.

Selecting **Figure 10.8** as the prototype circuit (**Figure 10.9** could also be selected, but the results would have a different form), the prototype circuit element values are: $R_S = 1 \Omega$, $C_1 = 1.0212 F$, $C_2 = 0.0098 F$, $L_2 = 1.1355 H$, $C_3 = 1.0212 F$, and $R_L = 1 \Omega$. Applying the element transformations specified for lowpass-to-bandpass in **Section 10.3**, and also applying impedance scaling as specified in **Section 10.4**, results in the final bandpass filter implementation as shown in **Figure 10.13**.

The magnitude frequency response, as obtained from SPICE analysis applied to **Figure 10.13** is shown in **Figure 10.14**. Note that the peak of the passband

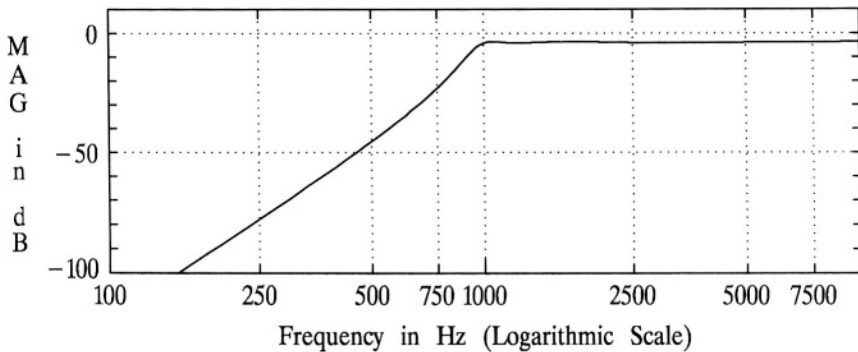


Figure 10.12 The magnitude frequency response, as obtained by SPICE, of the passive 5th-order highpass implementation of the Chebyshev Type I filter of *Example 10.2*, as illustrated in **Figure 10.10**, with $A_p = 0.5$ dB, $f_p = 1,000$ Hz, $f_s = 300$ Hz, and $A_s = 60$ dB.

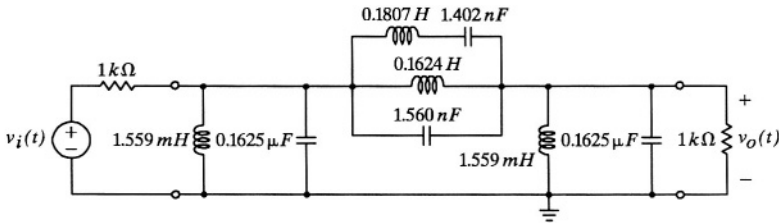


Figure 10.13 The passive 6th-order bandpass implementation of the elliptic filter of *Example 10.3* based on *Figure 10.8*, with $A_p = 0.099 \text{ dB}$, $A_s = 62.5 \text{ dB}$, $B_p = 1,000 \text{ Hz}$, $B_s = 8,206 \text{ Hz}$, and $f_o = 10,000 \text{ Hz}$.

response is -6.02 dB . This peak passband response is attenuated from 0 dB due to the source resistance of $1,000 \Omega$ and the load resistance of $1,000 \Omega$. Note also that for all frequencies in the stopband that the attenuation is no less than 68.5 dB . □

Example 10.4

Suppose it is desired to implement a fourth-order Butterworth bandstop filter that meets the following specifications: $A_p = 3 \text{ dB}$, $B_p = 1,000 \text{ Hz}$, $f_o = 10,000 \text{ Hz}$, and $R_L = 1,000 \Omega$. Selecting *Figure 10.4* as the prototype circuit, the prototype circuit element values, from *Table 10.1*, are as follows: $R_s = 1 \Omega$,

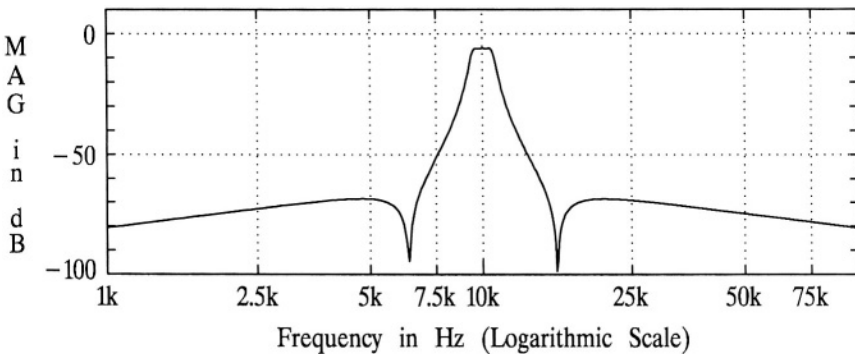


Figure 10.14 The magnitude frequency response, as obtained by SPICE, of the passive 6th-order bandpass implementation of the elliptic filter of *Example 10.3*, as illustrated in *Figure 10.13*, with $A_p = 0.099 \text{ dB}$, $A_s = 62.5 \text{ dB}$, $B_p = 1,000 \text{ Hz}$, $B_s = 8,206 \text{ Hz}$, and $f_o = 10,000 \text{ Hz}$.

$C_1 = 1.4142 F$, $L_2 = 1.4142 H$, and $R_L = 1 \Omega$. Applying the element transformations specified for lowpass-to-bandstop in **Section 10.3**, and also applying impedance scaling as specified in **Section 10.4**, results in the final bandstop filter implementation as shown in **Figure 10.15**.

The magnitude frequency response, as obtained from SPICE analysis applied to **Figure 10.15** is shown in **Figure 10.16**. Note that the peak of the passband response is -6.02 dB , and that between the -9.03 dB points is $1,000 \text{ Hz}$. \square

10.6 CHAPTER 10 PROBLEMS

- 10.1 Summarize the procedure for designing and realizing a passive analog filter.
- 10.2 An important step in *realizing* a passive *RLC* analog filter is *implementing* the filter, that is drawing a circuit schematic diagram of the filter with all element values given. Summarize the implementation procedure as presented in this chapter.
- 10.3 Explain why SPICE simulation is an important step in implementation.
- 10.4 Briefly explain how MATLAB and SPICE each play an important part in analog filter design and implementation. How do the two relate to each other? That is, how does MATLAB provide analysis of an *RLC* analog filter implementation?
- 10.5 Similar to *Example 10.1*, with $\omega_c = 1 \text{ rad/s}$ and $R_S = R_L = 1 \Omega$, determine two passive ladder implementations for a 2nd-order lowpass Butterworth filter using the procedure of **Section 10.2**. Compare your results with that found in **Table 10.1**.

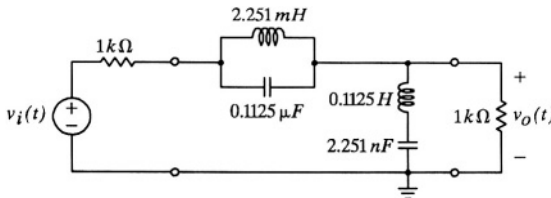


Figure 10.15 The passive 4th-order bandstop implementation of the Butterworth filter of *Example 10.4* based on **Figure 10.4**, with $A_p = 3 \text{ dB}$, $B_p = 1,000 \text{ Hz}$, and $f_o = 10,000 \text{ Hz}$.

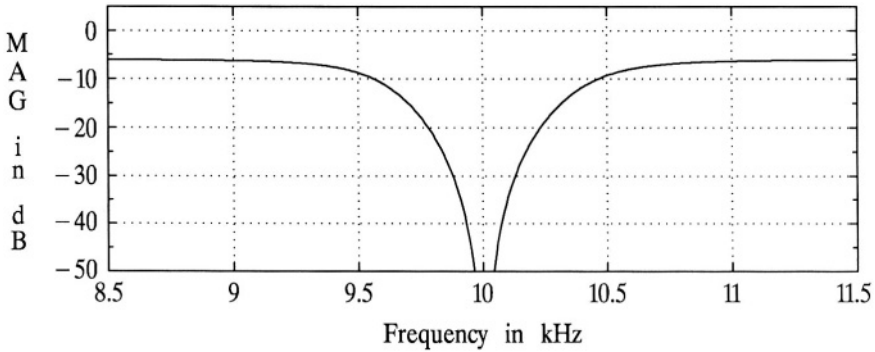


Figure 10.16 The magnitude frequency response, as obtained by SPICE, of the passive 4th-order bandstop implementation of the Butterworth filter of *Example 10.4*, as illustrated in *Figure 10.15*, with $A_p = 3$ dB, $B_p = 1,000$ Hz, and $f_o = 10,000$ Hz.

- 10.6** Similar to *Example 10.1*, with $\omega_c = 1$ rad/s and $R_s = R_L = 1 \Omega$, determine two passive ladder implementations for a 4th-order lowpass Butterworth filter using the procedure of *Section 10.2*. Compare your results with that found in *Table 10.1*.
- 10.7** Similar to *Example 10.1*, with $\omega_p = 1$ rad/s, passband ripple = $A_p = 0.5$ dB, $R_s = 0.5 \Omega$, and $R_L = 1 \Omega$, determine two passive ladder implementations for a 2nd-order lowpass Chebyshev Type I filter using the procedure of *Section 10.2*. Compare your results with that found in *Table 10.4*.
- 10.8** Similar to *Example 10.1*, with $\omega_p = 1$ rad/s, passband ripple = $A_p = 0.5$ dB, $R_s = 0.5 \Omega$, and $R_L = 1 \Omega$, determine two passive ladder implementations for a 3rd-order lowpass Chebyshev Type I filter using the procedure of *Section 10.2*. Compare your results with that found in *Table 10.4*.
- 10.9** Similar to *Example 10.1*, with $\omega_p = 1$ rad/s, passband ripple = $A_p = 0.5$ dB, $R_s = 0.5 \Omega$, and $R_L = 1 \Omega$, determine two passive ladder implementations for a 4th-order lowpass Chebyshev Type I filter using the procedure of *Section 10.2*. Compare your results with that found in *Table 10.4*.
- 10.10** Beginning only with the specifications given in the first paragraph, repeat and confirm all of the results of *Example 10.2* including the SPICE magnitude

frequency plot. Determine and plot the magnitude frequency response for the circuit of **Figure 10.11** as well. Compare the two magnitude responses. In addition, determine and plot the phase response of both circuits, the group delay response, the phase delay response, and the unit impulse response.

- 10.11** Repeat and confirm the results of *Example 10.3*. In addition, determine and plot the phase response, the group delay response, the phase delay response, and the unit impulse response.
- 10.12** Repeat and confirm the results of *Example 10.4*. In addition, determine and plot the phase response, the group delay response, the phase delay response, and the unit impulse response.
- 10.13** Suppose it is desired to implement a Chebyshev Type I highpass filter that meets the following specifications with minimum order: passband ripple = 0.5 dB , $f_p = 1,000 \text{ Hz}$, $f_s = 300 \text{ Hz}$, $A_s = 60 \text{ dB}$, and $R_L = 1,200 \Omega$. Determine two such passive circuit implementations. Using SPICE, determine and plot the magnitude frequency response of each circuit.
- 10.14** Suppose it is desired to implement a sixth-order elliptic bandpass filter that meets or exceeds the following specifications: passband ripple = $A_p = 0.1 \text{ dB}$, minimum attenuation in the stopband relative to the peak response = $A_s = 62 \text{ dB}$, $B_p = 1,000 \text{ Hz}$, $B_s = 8,500 \text{ Hz}$, and $f_o = 20,000 \text{ Hz}$. Suppose it is also desired to use $5,000 \Omega$ resistors. Determine two such passive circuit implementations. Using SPICE, determine and plot the magnitude frequency response of each circuit.
- 10.15** Suppose it is desired to implement a fourth-order Butterworth bandstop filter that meets the following specifications: $A_p = 3 \text{ dB}$, $B_p = 1,000 \text{ Hz}$, $f_o = 25,000 \text{ Hz}$, and $R_L = 600 \Omega$. Determine two such passive circuit implementations. Using SPICE, determine and plot the magnitude frequency response of each circuit.
- 10.16** Suppose it is desired to design and implement a Butterworth bandpass filter that meets the following specifications with minimum order: $A_p = 3 \text{ dB}$, $A_s = 50 \text{ dB}$, $f_o = 455 \text{ kHz}$, $B_p = 10 \text{ kHz}$, and $B_s = 30 \text{ kHz}$. Suppose it is also required that $1,000 \Omega$ resistors be used. Determine two such passive circuit implementations. Using SPICE, determine and plot the magnitude frequency response of each circuit.

- 10.17** Suppose it is desired to design and implement a time delay circuit that has a delay at very low frequencies of $10 \mu s$, and a minimum time delay of $8 \mu s$ at $80 kHz$. Determine two such passive circuit implementations.
- 10.18** Suppose it is desired to design and implement a passive Bessel lowpass filter that meets the following specifications with minimum order: $f_p = 3,500 Hz$, $f_s = 20,000 Hz$, $A_p = 2.5 dB$, and $A_s = 60 dB$. Using MATLAB and **Table 10.6**, design and implement two such passive filters with $1,000 \Omega$ resistors. Using SPICE, plot the magnitude frequency response of the circuit implementation. Also plot the phase response.
- 10.19** Suppose it is desired to implement a sixth-order passive Bessel bandpass filter with the following specifications: $A_p = 1 dB$, $B_p = 1,000 Hz$, and $f_o = 20,000 Hz$. Using **Figure 10.7** as the prototype circuit schematic diagram, and using $1,000 \Omega$ resistors, determine and draw the circuit schematic diagram of the bandpass filter. Using SPICE, plot the magnitude frequency response of the bandpass filter circuit. Also plot the phase response.

CHAPTER 11

ACTIVE FILTERS

Active filter implementation has three distinct advantages over passive implementation. First, as is presented below, active filter implementation need not make use of any inductors: only resistors, capacitors, and active elements (active elements are restricted to operational amplifiers (op amps) here). The reason why this is an advantage is that practical inductors (physical approximations to inductors) tend to be less ideal than are practical capacitors. Practical inductors are coils of wire, often fine wire, on some sort of core material. The core material has losses, the wire has resistance, and there is capacitance between the layers of coil windings. In addition, the coils radiate electromagnetic energy and can result in unwanted mutual inductance.

Second, the output resistance of an op amp is very low, especially with the feedback that is used with common active filter op amp stages. This output resistance (output impedance) is much lower than is the input impedance to the following active filter op amp stage. Even though the input impedance to a typical active filter op amp stage is frequency dependent and non-resistive this is so. Therefore, the individual active filter op amp stages, perhaps several being cascaded together, operate independently of each other (note that this is not true for individual ladder stages of a passive filter). This allows for the implementation of simple first-order and second-order active op amp stages which are then cascaded to yield the overall desired transfer function. All that is required is the introduction to a catalog of active filter op amp stages, and no detailed tables are required.

Third, because of the independence of individual stages the design and implementation engineer is not restricted to filters that are in a set of tables. Any filter in this book, that is, any filter transfer function, with any frequency transformation applied, can, at least in theory, be implemented with an active op amp filter, with complete flexibility as to such parameters as passband ripple, etc.

However, where high power or very high frequencies are involved, active filter implementation with op amps is prohibited. Therefore, passive filter implementation is often desired or required.

Note that what follows in this chapter is somewhat simplistic, in that actual filter implementation can be highly involved and specialized. There are talented engineers and engineering firms that specialize in filter implementation in all of its

various forms. The intent of this chapter, as well as the previous one, is to give the reader an introduction to analog filter implementation, while also presenting methods that are useful in many situations.

For simplicity, only ideal op amps will be considered. The op amp symbol is shown on the left side of **Figure 11.1**, and a more detailed representation of the *ideal* op amp is shown on the right side: the ideal op amp has infinite open-loop gain, zero output resistance, and infinite input resistance.

While amplifiers, *per se*, are not necessary in the implementation of active filters, they may be desirable for gain adjustment. The inverting amplifier, on the left side of **Figure 11.2**, has the gain $v_o/v_i = -R_2/R_1$. The noninverting amplifier, on the right side of **Figure 11.2** has the gain $v_o/v_i = 1 + (R_2/R_1)$.

11.1 FIRST-ORDER STAGES

In **Figure 11.3** is shown a first-order op amp stage that implements one real pole and no finite zero. This circuit and corresponding transfer function may be used where a pole on the negative real axis of the s plane is required, such as in odd-order lowpass filters. It is easy to show that the transfer function for the circuit shown in **Figure 11.3**, is as follows:

$$H(s) = \frac{V_o(s)}{V_i(s)} = -\frac{\frac{1}{R_1 C}}{s + \frac{1}{R_2 C}}, \quad (11.1)$$

where the pole is at $s = -1/(R_2 C)$ and the DC gain for this first-order lowpass transfer function is $-R_2/R_1$. For convenience it is often desirable to normalize at least one of the component values. Let $R_1 = 1 \Omega$, $1/(R_2 C) = \mu$, and $R_2 = G$. With these substitutions, (11.1) may be expressed as follows:

$$H(s) = -\frac{G \mu}{s + \mu}, \quad (11.2)$$



Figure 11.1 An illustration of the ideal op amp.

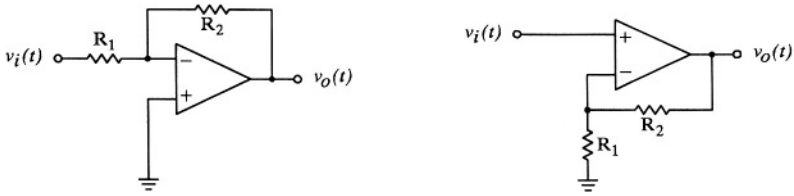


Figure 11.2 Inverting and non-inverting op amp amplifiers.

where the pole is at $s = -\mu$ and the DC gain is $-G$. The circuit shown in **Figure 11.3** may be used to implement (11.2) directly by using the following component values:

$$R_1 = 1 \Omega, \quad R_2 = G \Omega, \quad C = \frac{1}{G\mu} F. \tag{11.3}$$

After obtaining the initial component values indicated in (11.3) for a desired pole value and desired DC gain, then impedance scaling¹ may be applied for more convenient component values.

In **Figure 11.4** is shown a first-order op amp stage that implements one real pole and a zero at the origin. This circuit and corresponding transfer function may be used where a pole on the negative real axis of the s plane is required and a zero at the

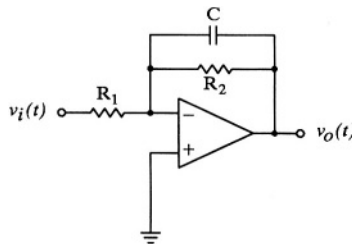


Figure 11.3 A first-order op amp stage that implements one real pole and no finite zero. The general transfer function is (11.1). The transfer function for the component values given in (11.3) is (11.2).

¹See Section 10.4.

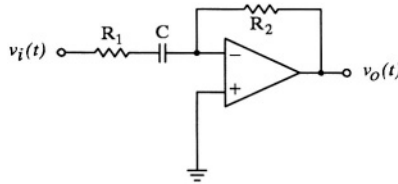


Figure 11.4 A first-order op amp stage that implements one real pole and a zero at the origin. The general transfer function is (11.4). The transfer function for the component values given in (11.6) is (11.5).

origin, such as in odd-order highpass filters. The transfer function for the circuit shown in **Figure 11.4** is as follows:

$$H(s) = \frac{V_o(s)}{V_i(s)} = -\frac{\frac{R_2}{R_1} s}{s + \frac{1}{R_1 C}}, \quad (11.4)$$

where the pole is at $s = -1/(R_1 C)$ and the high-frequency gain for this first-order highpass transfer functions $-R_2/R_1$. For convenience, let $R_1 = 1 \Omega$, $1/(R_1 C) = \mu$, and $R_2 = G$. With these substitutions, (11.4) may be expressed as follows:

$$H(s) = -\frac{G s}{s + \mu}, \quad (11.5)$$

where the pole is at $s = -\mu$ and the high-frequency gain is $-G$. The circuit shown in **Figure 11.4** may be used to implement (11.5) directly by using the following component values:

$$R_1 = 1 \Omega, \quad R_2 = G \Omega, \quad C = \frac{1}{\mu} F. \quad (11.6)$$

After obtaining the initial component values indicated in (11.6) for a desired pole value and desired high-frequency gain, then impedance scaling may be applied for more convenient component values.

In **Figure 11.5** is shown a first-order op amp stage that implements one real pole and one real, finite zero. It allows for the implementation of a first-order all-pass

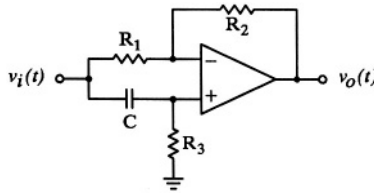


Figure 11.5 A first-order op amp stage that implements one real pole and one real zero. The general transfer function is (11.7). The transfer function for the component values given in (11.9) is (11.8).

transfer function (Van Valkenburg, 1982). The transfer function for the circuit shown in **Figure 11.5** is as follows:

$$H(s) = \frac{V_o(s)}{V_i(s)} = \frac{s - \frac{R_2}{R_1 R_3 C}}{s + \frac{1}{R_3 C}} \quad , \quad (11.7)$$

where the pole is at $s = -1/(R_3 C)$, and the zero is at $s = R_2/(R_1 R_3 C)$. For an all-pass transfer function it is necessary for the magnitude of the pole and zero to be the same. For convenience, let $R_1 = R_2 = R_3 = 1 \Omega$ and $C = 1/\mu F$. With these substitutions, (11.7) may be expressed as follows:

$$H(s) = \frac{s - \mu}{s + \mu} \quad , \quad (11.8)$$

where the pole is at $s = -\mu$, the zero is at $s = \mu$, and the magnitude frequency response is unity for all frequencies. The circuit shown in **Figure 11.5** may be used to implement (11.8) directly by using the following component values:

$$R_1 = R_2 = R_3 = 1 \Omega, \quad C = \frac{1}{\mu} F. \quad (11.9)$$

After obtaining the initial component values indicated in (11.9) for a desired pole/zero value, then impedance scaling may be applied for more convenient component values.

The above first-order stages shown in **Figure 11.3** through **Figure 11.5**, allow for implementation of any first-order transfer function necessary for the filters considered in this book, including all-pass. Next, second-order stages are considered.

11.2 SECOND-ORDER STAGES

The second-order stages presented in this section include stages appropriate for lowpass, highpass, bandpass, bandstop, and all-pass filters. At first thought it may be supposed that lowpass stages would be sufficient, since frequency transformation circuit operations, as covered in **Chapter 10**, could then be applied to transform, for example, a lowpass stage into a bandpass stage. This is indeed possible, however, it would introduce inductors, which can be avoided by including appropriate stages specifically for highpass, bandpass, and bandstop stages in the circuit catalog of active filter op amp stages. This, in turn, requires a somewhat different procedure for active filter implementation than used for passive filter implementation in **Chapter 10**. Those differences become clear in the summary and examples below. Greater detail on each of the second-order stages below may be found in Van Valkenburg (1982).

Sallen and Key Lowpass Circuit

In **Figure 11.6** is shown the Sallen and Key lowpass circuit, which implements one complex-conjugate pair of poles with no finite-valued zeros. This circuit may be used to implement one second-order stage in a lowpass filter that has no zeros on the $j\omega$ axis, such as required with Butterworth, Chebyshev Type I, Bessel, etc. lowpass filters. The transfer function for the circuit shown in **Figure 11.6** is as follows:

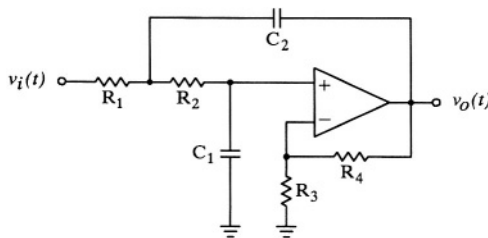


Figure 11.6 The Sallen and Key lowpass second-order op amp stage. The general transfer function is (11.10). The transfer function for the normalized component values in (11.13) is (11.14).

$$\begin{aligned}
 H(s) &= \frac{V_o(s)}{V_i(s)} \\
 &= \frac{K/(R_1 R_2 C_1 C_2)}{s^2 + \left(\frac{1}{R_1 C_2} + \frac{1}{R_2 C_2} + \frac{1}{R_2 C_1} - \frac{K}{R_2 C_1} \right) s + \frac{1}{R_1 R_2 C_1 C_2}} \quad , \quad (11.10)
 \end{aligned}$$

where K , the DC gain, is equal to $1 + R_4/R_3$. Assuming a complex-conjugate pair of poles² it is easy to show that the magnitude of those poles is $1/\sqrt{R_1 R_2 C_1 C_2}$. Therefore let $\omega_o^2 = 1/(R_1 R_2 C_1 C_2)$, where ω_o is defined as the magnitude of the poles. As is often done in the literature, let the transfer function Q be defined as $Q = 1/[2 \cos(\theta)]$, where θ is the angle between the pole and the negative real axis of the s plane, i.e., the poles are as follows: $s_{1,2} = -\omega_o [\cos(\theta) \pm j \sin(\theta)]$. It is noted that ω_o and Q completely specify the pole locations. Note that when $Q = 0.5$ the poles are equal on the negative real axis, i.e. $s_{1,2} = -\omega_o$. When $Q = 1/\sqrt{2}$ the poles are as follows: $s_{1,2} = -\omega_o (1 \pm j)/\sqrt{2}$. And as Q approaches infinity the poles approach $\pm j\omega_o$. From the above it is clear that an alternate expression of Q is as follows: $Q = \omega_o / [-2 \Re e(s_{1,2})]$. Therefore, (11.10) may be expressed as follows:

$$H(s) = \frac{K \omega_o^2}{s^2 + \frac{\omega_o}{Q} s + \omega_o^2} \quad , \quad (11.11)$$

where

$$Q = \frac{\sqrt{R_1 R_2 C_1 C_2}}{C_1 (R_1 + R_2) + R_1 C_2 (1 - K)} \quad . \quad (11.12)$$

For convenience, let all components be normalized to either 1Ω or $1 F$, except for the feedback resistor R_4 , which has a value of $2 - 1/Q \Omega$:

$$R_1 = R_2 = R_3 = 1 \Omega \quad , \quad C_1 = C_2 = 1 F \quad , \quad R_4 = 2 - \frac{1}{Q} \Omega \quad . \quad (11.13)$$

With these substitutions the transfer function for the circuit shown in **Figure 11.6**, as represented by either (11.10) or (11.11), is

²For appropriate component values it is possible for both poles to be on the negative real axis. For example, if $R_1 = R_2 = C_1 = K = 1$ and $C_2 = 1/2$, it follows that the denominator of (11.10) would be $s^2 + 4s + 2$ and the poles would be $s_{1,2} = -2 \pm \sqrt{2}$. However, only complex-conjugate poles are of interest here.

$$H(s) = \frac{K}{s^2 + \frac{1}{Q}s + 1} \quad (11.14)$$

where $K = 3 - 1/Q$ is the DC gain, and $\omega_o = 1$. Therefore, to implement the circuit with normalized component values, as represented by (11.14), all that is required is the value of Q : $Q = \omega_o / [-2 \Re(s_{1,2})]$. After obtaining the initial component values indicated in (11.13) for a desired Q value, then frequency scaling may be applied for the desired ω_o value, and impedance scaling may be applied for more convenient component values. Note that neither frequency scaling nor impedance scaling will affect the value of Q as can be observed from (11.12).

Note that for the normalized circuit (component values as shown in (11.13)), as the feedback resistor, R_4 , value is made closer and closer to 2Ω , Q would be approaching infinity, the two poles, still with a magnitude of unity, would be approaching the $j\omega$ axis of the s plane. With the feedback resistor precisely equal to 2Ω the stage would oscillate at 1 rad/s . With the feedback resistor greater than 2Ω the circuit becomes unstable.

Lowpass Notch Circuit

In Figure 11.7 is shown one form of a second-order lowpass notch circuit, that is, a circuit with a pair of zeros on the $j\omega$ axis (hence, notch) but with a low-frequency gain that is greater than the high-frequency gain. This circuit may be used to implement one second-order stage in any lowpass filter that has finite-valued zeros on the $j\omega$ axis, such as Chebyshev Type II and elliptic filters. Notwithstanding the term *lowpass* notch, it may also be used to implement one second-order stage of a bandpass filter or a bandstop filter that has finite-valued zeros on the $j\omega$ axis. In brief, this circuit may be used in the implementation of any second-order stage that requires a frequency response transmission zero, denoted ω_z , that is *greater* than the magnitude of the poles, denoted ω_o .

The transfer function for the circuit shown in Figure 11.7 is as follows:

$$H(s) = \frac{V_o(s)}{V_i(s)} = \frac{R_3}{R_2 + R_3} \frac{s^2 + \alpha s + \omega_z^2}{s^2 + \frac{\omega_o}{Q}s + \omega_o^2} \quad (11.15)$$

where

$$\omega_o^2 = 1/(R_1 R_5 C_1 C_2) \quad (11.16)$$

$$\omega_o/Q = (C_1 + C_2)/(R_5 C_1 C_2) \quad (11.17)$$

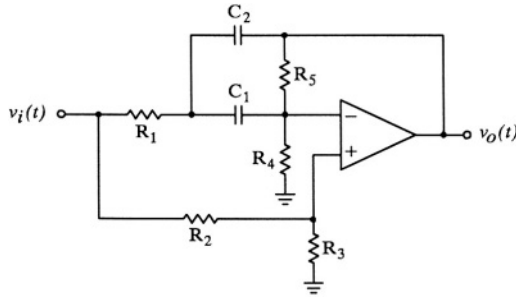


Figure 11.7 One form of the second-order lowpass notch circuit. The general transfer function is (11.15). With the component values shown in (11.25), then $\omega_o = 1$ and $\alpha = 0$.

$$\omega_z^2 = (R_4 + R_5) / (R_1 R_4 R_5 C_1 C_2) , \tag{11.18}$$

and

$$\alpha = \frac{R_1 R_3 (C_1 + C_2) (R_4 + R_5) - R_2 R_4 R_5 C_1}{R_1 R_3 R_4 R_5 C_1 C_2} . \tag{11.19}$$

It is recognized that (11.15) has two poles and two zeros, and as for the Sallen and Key lowpass circuit, ω_o is the magnitude of the poles, and $Q = \omega_o / [-2 \Re(s_{1,2})]$ where $s_{1,2}$ are the poles. Similarly, ω_z is the magnitude of the zeros. In order for the zeros to be on the $j\omega$ axis it is necessary that $\alpha = 0$. Therefore it follows from (11.19) that

$$R_2 R_4 R_5 C_1 = R_1 R_3 (C_1 + C_2) (R_4 + R_5) . \tag{11.20}$$

For convenience, let $R_1 = R_3 = \omega_o = 1$. It follows, therefore, from (11.16) that

$$R_5 C_1 C_2 = 1 . \tag{11.21}$$

Combining $\omega_o = 1$, (11.17), and (11.21):

$$Q = \frac{1}{C_1 + C_2} . \tag{11.22}$$

For convenience, and making use of (11.22), let $C_1 = C_2 = 1/(2Q)$. Therefore,

from (11.21),

$$R_5 = 4Q^2 . \quad (11.23)$$

Substituting (11.23) into (11.18) results in

$$R_4 = \frac{4Q^2}{\omega_z^2 - 1} . \quad (11.24)$$

Substituting (11.22) - (11.24) into (11.20) results in

$$R_2 = \frac{\omega_z^2}{2Q^2} .$$

It was noted above, that in this circuit ω_z must be greater than ω_o . This is clearly seen in (11.24), where it is noted that $\omega_o = 1$ here. Also note that all that is required for initial circuit component values is the value of Q and the value of ω_z relative to a normalized ω_o , that is the value of ω_z for the initial normalized circuit must be equal to the ratio ω_z/ω_o of the actual completed circuit (non-normalized values).

To summarize the design of this stage, what is needed initially is the value of the desired poles of the actual, non-normalized, circuit and also the corresponding value of the non-normalized (actual) transmission zero ω_z . Given that the poles are denoted as $s_{1,2}$, then the non-normalized ω_o value is the magnitude of $s_{1,2}$, and $Q = \omega_o / [-2 \Re(s_{1,2})]$. Then, the normalized ω_z is the ratio ω_z/ω_o of the actual non-normalized values, ω_o is normalized to unity, and Q remains unchanged. The component values of the normalized circuit are expressed above and are repeated here for convenience:

$$R_1 = 1 \Omega, \quad R_2 = \frac{\omega_z^2}{2Q^2} \Omega, \quad R_3 = 1 \Omega, \quad R_4 = \frac{4Q^2}{\omega_z^2 - 1} \Omega, \quad (11.25)$$

$$R_5 = 4Q^2 \Omega, \quad C_1 = C_2 = \frac{1}{2Q} F .$$

After obtaining the initial component values indicated in (11.25) for a desired Q value and normalized ω_z value, then frequency scaling may be applied for the desired ω_o value (and desired ω_z), and impedance scaling may be applied for more convenient component values. Again, note that neither frequency scaling nor impedance scaling will affect the value of Q .

Sallen and Key Highpass Circuit

The development of the Sallen and Key highpass circuit theory follows that of the Sallen and Key lowpass. In **Figure 11.8** is shown the Sallen and Key highpass circuit, which implements one complex-conjugate pair of poles with two zeros at the origin. This circuit may be used to implement one second-order stage in a highpass

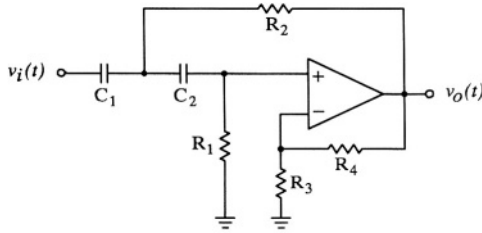


Figure 11.8 The Sallen and Key highpass second-order op amp stage. The general transfer function is (11.26). The transfer function for the normalized component values in (11.29) is (11.27).

filter that has no transmission zeros on the $j\omega$ axis, other than at the origin, such as required with Butterworth, Chebyshev Type I, Bessel, etc. highpass filters. The transfer function for the circuit shown in **Figure 11.8** is as follows:

$$\begin{aligned}
 H(s) &= \frac{V_o(s)}{V_i(s)} \\
 &= \frac{K s^2}{s^2 + \left(\frac{1}{R_1 C_1} + \frac{1}{R_1 C_2} + \frac{1}{R_2 C_1} - \frac{K}{R_2 C_1} \right) s + \frac{1}{R_1 R_2 C_1 C_2}} \quad , \quad (11.26)
 \end{aligned}$$

where K , the high-frequency gain, is equal to $1 + R_4/R_3$. As before, assuming a complex-conjugate pair of poles, it is easy to show that the magnitude of those poles is $1/\sqrt{R_1 R_2 C_1 C_2}$. Therefore let $\omega_o^2 = 1/(R_1 R_2 C_1 C_2)$, where ω_o is defined as the magnitude of the poles. Let the transfer function Q be defined as $Q = 1/[2 \cos(\theta)]$, where θ is the angle between the pole and the negative real axis of the s plane. It is noted that ω_o and Q completely specify the pole locations. An alternate expression of Q is as follows: $Q = \omega_o / [-2 \Re e(s_{1,2})]$. Therefore, (11.26) may be expressed as follows:

$$H(s) = \frac{K s^2}{s^2 + \frac{\omega_o}{Q} s + \omega_o^2} \quad , \quad (11.27)$$

where

$$Q = \frac{\sqrt{R_1 R_2 C_1 C_2}}{R_2(C_1 + C_2) + R_1 C_2(1 - K)} \quad (11.28)$$

For convenience, let all components be normalized to either 1Ω or $1 F$, except for the feedback resistor R_4 , which has a value of $2 - 1/Q \Omega$:

$$R_1 = R_2 = R_3 = 1 \Omega, \quad C_1 = C_2 = 1 F, \quad R_4 = 2 - \frac{1}{Q} \Omega \quad (11.29)$$

With these substitutions the transfer function for the circuit shown in **Figure 11.8**, as represented by either (11.26) or (11.27), is

$$H(s) = \frac{K s^2}{s^2 + \frac{1}{Q} s + 1} \quad (11.30)$$

where $K = 3 - 1/Q$ is the high-frequency gain, and $\omega_o = 1$. Therefore, to implement the circuit with normalized component values, as represented by (11.30), all that is required is the value of Q : $Q = \omega_o / [-2 \Re(s_{1,2})]$. After obtaining the initial component values indicated in (11.29) for a desired Q value, then frequency scaling may be applied for the desired ω_o value, and impedance scaling may be applied for more convenient component values. Note that neither frequency scaling nor impedance scaling will affect the value of Q as can be observed from (11.28).

Highpass Notch Circuit

In **Figure 11.9** is shown one form of a second-order highpass notch circuit, that is, a circuit with a pair of zeros on the $j\omega$ axis (hence, notch) but with a high-frequency gain that is greater than the low-frequency gain. This circuit may be used to implement one second-order stage in any highpass filter that has finite-valued zeros on the $j\omega$ axis, such as Chebyshev Type II and elliptic filters. Notwithstanding the term *highpass notch*, it may also be used to implement one second-order stage of a bandpass filter or a bandstop filter that has finite-valued zeros on the $j\omega$ axis. In brief, this circuit may be used in the implementation of any second-order stage that requires a frequency response transmission zero, denoted ω_z , that is *less than or equal to* the magnitude of the poles, denoted ω_o .

The transfer function for the circuit shown in **Figure 11.9** is as follows:

$$H(s) = \frac{V_o(s)}{V_i(s)} = K \frac{s^2 + \alpha s + \omega_z^2}{s^2 + \frac{\omega_o}{Q} s + \omega_o^2} \quad (11.31)$$

where K , the high-frequency gain, is

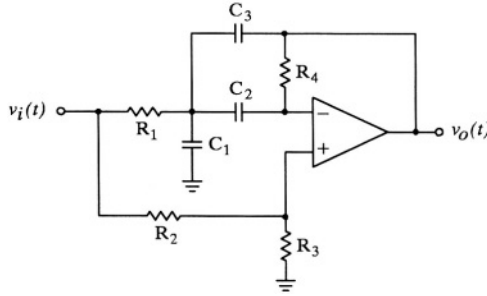


Figure 11.9 One form of the second-order highpass notch circuit. The general transfer function is (11.31). With the component values shown in (11.42), then $\omega_o = 1$ and $\alpha = 0$.

$$K = \frac{R_3 (C_1 + C_3)}{C_3 (R_2 + R_3)} ,$$

and

$$\omega_o^2 = 1 / (R_1 R_4 C_2 C_3) , \tag{11.32}$$

$$\omega_o / Q = (C_1 + C_2 + C_3) / (R_4 C_2 C_3) , \tag{11.33}$$

$$\omega_z^2 = 1 / [R_1 R_4 C_2 (C_1 + C_3)] , \tag{11.34}$$

and

$$\alpha = \frac{R_1 R_3 (C_1 + C_2 + C_3) - R_2 R_4 C_2}{R_1 R_3 R_4 (C_1 + C_3) C_2} . \tag{11.35}$$

It is recognized that (11.31) has two poles and two zeros, ω_o is the magnitude of the poles, and $Q = \omega_o / [-2 \Re(s_{1,2})]$ where $s_{1,2}$ are the poles. Similarly, ω_z is the magnitude of the zeros. In order for the zeros to be on the $j\omega$ axis it is necessary that $\alpha = 0$. Therefore it follows from (11.35) that

$$R_2 R_4 C_2 = R_1 R_3 (C_1 + C_2 + C_3) . \tag{11.36}$$

For convenience, let $R_1 = R_2 = \omega_o = 1$, let $C_2 = C_3 = C$, and let $C_1 = k C$. It follows, therefore, from (11.32) that

$$R_4 C^2 = 1 . \quad (11.37)$$

From (11.34) it follows that $\omega_z^2 = \omega_o^2 / (1 + k)$. It therefore follows that

$$k = \frac{\omega_o^2}{\omega_z^2} - 1 . \quad (11.38)$$

Combining $\omega_o = 1$, (11.33), and (11.37):

$$Q = \frac{1}{(2 + k) C} . \quad (11.39)$$

Making use of (11.39), $C = 1 / [(2 + k) Q]$. Therefore, from (11.37),

$$R_4 = (2 + k)^2 Q^2 . \quad (11.40)$$

Substituting (11.40) into (11.36) results in

$$R_3 = (2 + k) Q^2 . \quad (11.41)$$

It was noted above, that in this circuit ω_z must be less than or equal to ω_o . This is clearly seen in (11.38), where it is noted that k cannot be less than zero (since C_1 cannot have a negative value). However, C_1 can be zero, in which case $\omega_z = \omega_o$, and the circuit becomes a second-order notch where the DC gain and the high-frequency gain are equal.

Note that all that is required for initial circuit component values is the value of Q and the value of ω_z relative to a normalized ω_o . That is, the value of ω_z for the initial normalized circuit must be equal to the ratio ω_z / ω_o of the actual completed circuit (non-normalized values).

To summarize the design of this stage, what is needed initially is the value of the desired poles of the actual, non-normalized, circuit and also the corresponding value of the non-normalized (actual) transmission zero ω_z . Given that the poles are denoted as $s_{1,2}$, then the non-normalized ω_o value is the magnitude of $s_{1,2}$, and $Q = \omega_o / [-2 \Re(s_{1,2})]$. Then, the normalized ω_z is the ratio ω_z / ω_o of the actual non-normalized values, ω_o is normalized to unity, and Q remains unchanged. The component values of the normalized circuit are expressed above and are repeated here for convenience:

$$\begin{aligned} R_1 = R_2 = 1 \Omega, \quad R_3 = (2 + k) Q^2 \Omega, \quad R_4 = (2 + k)^2 Q^2 \Omega, \\ C_1 = \frac{k}{(2 + k) Q} F, \quad C_2 = C_3 = \frac{1}{(2 + k) Q} F, \end{aligned} \quad (11.42)$$

where $k = (\omega_o^2 / \omega_z^2) - 1$. After obtaining the initial component values indicated in (11.42) for a desired Q value and normalized ω_z value, then frequency scaling may be applied for the desired ω_o value (and desired ω_z), and impedance scaling may be

applied for more convenient component values. Note that neither frequency scaling nor impedance scaling will affect the value of Q nor k .

The Friend Bandpass Circuit

In Figure 11.10 is shown one form of a Friend bandpass circuit. This circuit may be used to implement one second-order stage in a bandpass filter that does not have finite-valued zeros on the $j\omega$ axis, other than at the origin, such as Butterworth and Chebyshev Type I filters.

The transfer function for the circuit shown in Figure 11.10 is as follows:

$$H(s) = \frac{V_o(s)}{V_i(s)} = \frac{K s}{s^2 + \frac{\omega_o}{Q} s + \omega_o^2} \quad , \quad (11.43)$$

where

$$\omega_o^2 = 1 / (R_1 R_2 C_1 C_2) \quad , \quad (11.44)$$

$$\omega_o / Q = (C_1 + C_2) / (R_2 C_1 C_2) \quad , \quad (11.45)$$

and

$$K = - 1 / (R_1 C_2) \quad . \quad (11.46)$$

It is recognized that (11.43) has two poles, and a zero at the origin: ω_o is the magnitude of the poles, and $Q = \omega_o / [-2 \Re e(s_{1,2})]$ where $s_{1,2}$ are the poles. It follows from (11.44) and (11.45) that

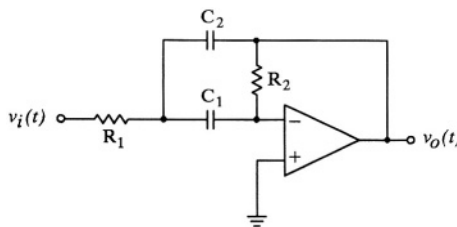


Figure 11.10 One form of the Friend bandpass circuit. The general transfer function is (11.43). With the component values shown in (11.50), then $\omega_o = 1$ and $K = - 2 Q$.

$$Q = \frac{\sqrt{R_1 R_2 C_1 C_2}}{R_1 (C_1 + C_2)} = \frac{1}{\omega_o R_1 (C_1 + C_2)} . \quad (11.47)$$

For convenience, let $R_1 = \omega_o = 1$, and let $C_1 = C_2 = C$. It follows, therefore, from (11.47) that

$$R_2 = 4Q^2 . \quad (11.48)$$

Substituting (11.48) into (11.44) it follows that

$$C = C_1 = C_2 = \frac{1}{2Q} . \quad (11.49)$$

From (11.46) $K = -2Q$.

Note that all that is required for initial circuit component values is the value of Q . That is, $R_1 = 1$, and R_2 and the capacitor values are given by (11.48) and (11.49), respectively.

To summarize the design of this stage, what is needed initially is the value of the desired poles of the actual, non-normalized, circuit. Given that the poles are denoted as $s_{1,2}$, then the non-normalized ω_o value is the magnitude of $s_{1,2}$, and $Q = \omega_o / [-2 \Re e(s_{1,2})]$. Then, ω_o is normalized to unity: Q remains unchanged. The component values of the normalized circuit are expressed above and are repeated here for convenience:

$$R_1 = 1 \Omega, \quad R_2 = 4Q^2 \Omega, \quad C_1 = C_2 = \frac{1}{2Q} F . \quad (11.50)$$

After obtaining the initial component values indicated in (11.50) for a desired Q value and a normalized ω_o , then frequency scaling may be applied for the desired ω_o value of the non-normalized circuit, and impedance scaling may be applied for more convenient component values. Note that neither frequency scaling nor impedance scaling will affect the value of Q .

It may be interesting to note an alternative representation for Q : $Q = \omega_o / B$, where B is the 3 dB bandwidth in rad/s. Note also that the peak gain is $2Q^2$:

$$|H(j\omega)| \Big|_{\omega = \omega_o} = 2Q^2 .$$

Note that bandpass or bandstop filters that have stopband rippling (zeros on the $j\omega$ axis) may be implemented by cascading lowpass notch and highpass notch stages. For a bandpass filter the cutoff frequency of the lowpass notch would be greater than that of the highpass. For a bandstop filter the cutoff frequency of the lowpass notch would be less than that of the highpass.

The Delyiannis Second-Order All-Pass Circuit

In **Figure 11.11** is shown one form of a Delyiannis second-order all-pass circuit. The transfer function for the circuit shown in **Figure 11.11** is as follows

$$H(s) = \frac{V_o(s)}{V_i(s)} = K \frac{s^2 + \alpha s + \omega_o^2}{s^2 + \frac{\omega_o}{Q} s + \omega_o^2} \quad (11.51)$$

where

$$\omega_o^2 = 1 / (R_1 R_4 C_1 C_2) \quad (11.52)$$

$$\omega_o / Q = (C_1 + C_2) / (R_4 C_1 C_2) \quad (11.53)$$

$$\alpha = \frac{R_1 R_3 (C_1 + C_2) - R_2 R_4 C_1}{R_1 R_3 R_4 C_1 C_2} \quad (11.54)$$

and

$$K = R_3 / (R_2 + R_3) \quad (11.55)$$

It is recognized that (11.51) has two poles and ω_o is the magnitude of the poles, and $Q = \omega_o / [-2 \Re e(s_{1,2})]$ where $s_{1,2}$ are the poles. In order for (11.51) to be a second-order all-pass transfer function, it is necessary that $\alpha = -\omega_o / Q$.³ Equating (11.54) to the negative of (11.53), it follows that

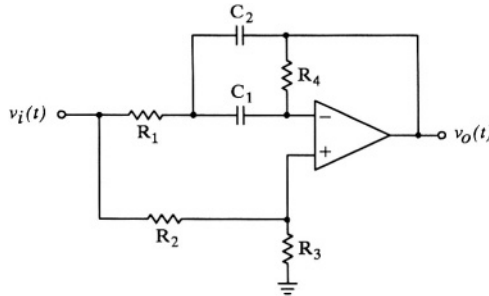


Figure 11.11 One form of the Delyiannis second-order all-pass circuit. The general transfer function is (11.51). With the component values shown in (11.61), then $\omega_o = 1$ and $K = Q^2 / (1 + Q^2)$.

³See Section 2.12.

$$R_2 R_4 C_1 = 2 R_1 R_3 (C_1 + C_2) . \quad (11.56)$$

For convenience, let $R_1 = R_2 = \omega_o = 1$, and let $C_1 = C_2 = C$. It follows, therefore, from (11.56) that

$$R_4 = 4 R_3 . \quad (11.57)$$

Substituting (11.52) into (11.53) it follows that

$$R_4 = 4 Q^2 . \quad (11.58)$$

From (11.57) and (11.58):

$$R_3 = Q^2 . \quad (11.59)$$

From (11.52) it follows that

$$C_1 = C_2 = C = \frac{1}{2Q} . \quad (11.60)$$

Note that all that is required for initial circuit component values is the value of Q . That is, $R_1 = R_2 = 1$, and R_3 , R_4 , and $C_1 = C_2$ are given by (11.59), (11.58) and (11.60), respectively.

To summarize the design of this stage, what is needed initially are the values of the non-normalized ω_o and Q .⁴ Then, ω_o is normalized to unity: Q remains unchanged. The component values of the normalized circuit are expressed above and are repeated here for convenience:

$$\begin{aligned} R_1 = R_2 = 1 \Omega, \quad R_3 = Q^2 \Omega, \\ R_4 = 4 Q^2 \Omega, \quad C_1 = C_2 = \frac{1}{2Q} F . \end{aligned} \quad (11.61)$$

After obtaining the initial component values indicated in (11.61) for a desired Q value and a normalized ω_o , then frequency scaling may be applied for the desired ω_o value of the non-normalized circuit, and impedance scaling may be applied for more convenient component values. Note that neither frequency scaling nor impedance scaling will affect the value of Q . Note also, from (11.55), that the magnitude gain of the circuit is $K = Q^2 / (1 + Q^2)$, independent of frequency.

The above first- and second-order stages shown in **Figure 11.3** through **Figure 11.11**, combined with attenuators and gain stages (**Figure 11.2**) for gain adjustment if required, are sufficient for implementation of any transfer function, including all-pass, presented in this book. The following section illustrates details as to the implementation of desired transfer functions.

⁴See **Section 2.12**.

11.3 SUMMARY AND EXAMPLES OF ACTIVE FILTER IMPLEMENTATION

Active filter implementation may be summarized as follows:

- Perform the filter design, as developed and presented in **Chapter 3** through **Chapter 9**, resulting in the complete transfer function of the filter to be implemented.
- Factor the transfer function so that the numerator and denominator each consist of the product of first-order and second-order terms all with real coefficients, i.e., put the transfer function in the form of (2.40). Arrange this transfer function as the product of possibly one first-order term (there will not be a first-order term if the transfer function order is even) of the form of that shown in (11.2), (11.5) or (11.8), and second-order terms of the form of (11.11), (11.15) (with $\alpha = 0$), (11.27), (11.31) (with $\alpha = 0$), (11.43), and (11.51) (with $\alpha = -\omega_o/Q$). For practical reasons, the terms should be cascaded with increasing peak gain: the lowest peak gain stage should be first, and the highest peak gain stage should be last. This will help prevent saturation of an amplifier in the cascade. High- Q stages may have a very pronounced resonant gain peak near the corner frequency, as illustrated in **Figure 2.33**. If a high- Q stage is early in a cascade of stages, the resonant gain peak may saturate an amplifier (exceed maximum allowable levels for linear operation).
- For each term in the above-factored transfer function, perform the appropriate frequency scaling on the corresponding normalized circuit for the desired pole(s). Impedance scaling may also be applied for convenient element values.
- Perform computer simulation analysis on the cascade of proposed circuit stages to test its accuracy. This step should always be performed prior to realization. If a full MATLAB analysis had already been conducted as part of the design procedure, only a brief SPICE analysis may be necessary as a check on the proposed circuit.

Example 11.1

Suppose it is desired to implement a fourth-order Chebyshev Type I lowpass filter with $\omega_p = 1,000 \text{ rad/s}$ and $\epsilon = 0.5$ (0.969 dB ripple). From **Example 4.5**,

$$H(s) = \frac{2.5 \times 10^{11}}{(s^2 + 282.26s + 989,571.46)(s^2 + 681.44s + 282,456.21)},$$

which may be expressed as:

$$\begin{aligned} H(s) &= \frac{5 \times 10^5}{s^2 + 681.44s + 282,456.21} \times \frac{5 \times 10^5}{s^2 + 282.26s + 989,571.46} \\ &= H_1(s) \times H_2(s). \end{aligned}$$

This transfer function may be implemented by cascading two Sallen and Key lowpass circuits. Note that the Q of $H_1(s)$ is

$$Q_1 = \frac{\sqrt{282,456.21}}{681.44} = 0.77992.$$

Similarly, $Q_2 = 3.5243$. Therefore,

$$H_1^{(norm)}(s) = \frac{1.7178}{s^2 + 1.2822s + 1},$$

and

$$H_2^{(norm)}(s) = \frac{2.7163}{s^2 + 0.28374s + 1},$$

where the *norm* superscript, in both cases, denotes the normalized (ω_o of unity) Sallen and Key lowpass circuit. The difference between the two stages, at this point, is primarily the value of Q . Note also, however, by use of (2.66), that $20 \log |H_1(j\omega)|_{peak} = 4.839 \text{ dB}$, and $20 \log |H_2(j\omega)|_{peak} = 19.709 \text{ dB}$. It is also interesting to note that $20 \log |H_1(j\omega)|_{DC} = 4.699 \text{ dB}$, and $20 \log |H_2(j\omega)|_{DC} = 8.680 \text{ dB}$, illustrating that not only does the second stage have the larger DC gain but also has a more pronounced resonant peak.

All that remains is frequency and impedance scaling. The frequency scaling factor for the first stage is $k_{f_1} = \sqrt{282,456.21} = 531.47$, and for the second stage, $k_{f_2} = \sqrt{989,571.46} = 994.77$. For convenience, let the impedance scaling factor, k_i , be the same for each stage, equal to 1,000. The final active filter implementation schematic diagram is shown in **Figure 11.12**. Note that the DC gain for this active filter is the product of the DC gains of the two stages, which are unaffected by frequency or impedance scaling. If this DC gain is not desirable, it may be adjusted by including an additional gain stage (see **Figure 11.2**).

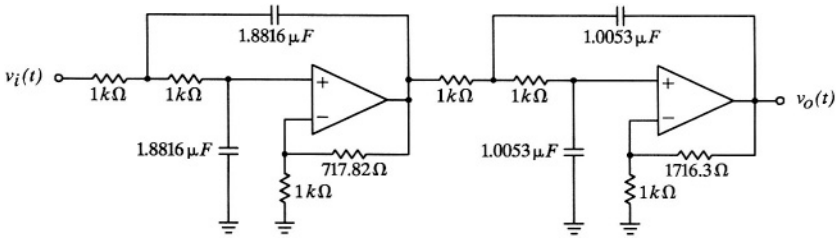


Figure 11.12 The active filter implementation circuit schematic diagram for *Example 11.1*.

The magnitude frequency response, as obtained from SPICE analysis applied to **Figure 11.12** is shown in **Figure 11.13**. Note that the DC gain, in dB, in **Figure 11.13** is 13.38 dB (the product of the two DC gains is 4.666). □

Example 11.2

Suppose it is desired to implement a third-order elliptic lowpass filter with $f_p = 1,000 \text{ Hz}$, $A_p = 3 \text{ dB}$, and $A_s = 40 \text{ dB}$. From *Example 6.10*, the poles and zeros for the normalized ($\omega_p = 1$) filter are as follows:

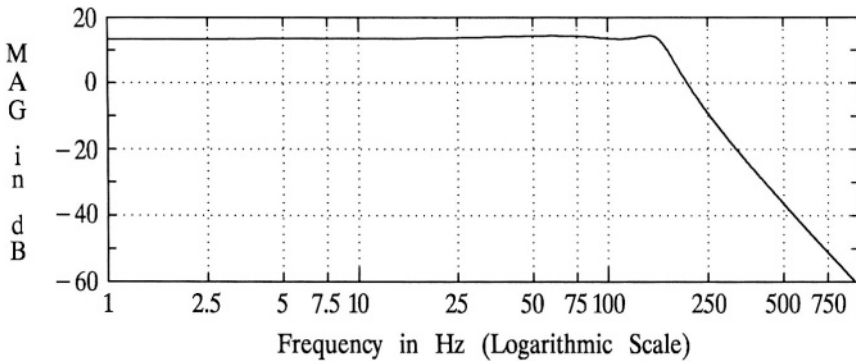


Figure 11.13 The magnitude frequency response of the circuit shown in **Figure 11.12**, as obtained by SPICE.

$$\begin{aligned} \text{poles: } & -0.3225, -0.1337 \pm j 0.9194, \\ \text{zeros: } & \pm j 2.2451 \end{aligned}$$

It follows that the normalized prototype transfer function may be expressed as:

$$\begin{aligned} H^{(norm)}(s) &= \frac{K_1 (s^2 + 5.0405)}{(s + 0.3225)(s^2 + 0.2674s + 0.8632)} \\ &= H_1^{(norm)}(s) H_2^{(norm)}(s) \\ &= \frac{K_2}{s + 0.3225} \frac{K_3 (s^2 + 5.0405)}{s^2 + 0.2674s + 0.8632} \end{aligned}$$

Note that the general form for the above transfer function is as follows:

$$H^{(norm)}(s) = \frac{K_2}{s + \omega_{o_1}} \frac{K_3 (s^2 + \omega_z^2)}{s^2 + \frac{\omega_{o_2}}{Q_2} s + \omega_{o_2}^2}$$

Therefore, $\omega_{o_1} = 0.3225$, $\omega_{o_2} = 0.9291$, and $Q_2 = 3.475$. Frequency scaling by $2\pi 1,000$ results in the desired transfer function:

$$H(s) = \frac{K_2}{s + 2,026.33} \frac{K_3 (s^2 + 1.9899 \times 10^8)}{s^2 + 1,680.12s + 3.4078 \times 10^7} \quad (11.62)$$

If $K_1 = K_2 K_3$ is 347.02 then the *DC* gain, and since the order is odd this will also be the peak gain, will be unity. However, the constants K_1 , K_2 , and K_3 are somewhat arbitrary since they only affect the gain and not the shape of the frequency response. In this example it is possible to easily obtain the desired *DC* gain since a first-order stage is required with an adjustable gain parameter.

Implementation of (11.62) may be accomplished by cascading one first-order stage of the form shown in Figure 11.3 with a lowpass notch circuit, as shown in Figure 11.7. For the second-order stage, it first must be determined what ω_z^2 must be for the normalized circuit. Clearly $\omega_z^2 = 1.9899 \times 10^8 / 3.4078 \times 10^7 = 5.8393$. Also, $Q = \sqrt{3.4078 \times 10^7} / 1,680.12 = 3.475$. For the normalized circuit, the capacitors are 0.14391 *F* each, $R_2 = 0.24185 \Omega$, $R_4 = 9.9784 \Omega$, and $R_5 = 48.289 \Omega$. Frequency scaling by $\sqrt{3.4078 \times 10^7} = 5,837.64$, and impedance scaling by 1,000 at the same time, results in capacitor values of 0.02465 μF , $R_1 = R_3 = 1 k\Omega$, $R_2 = 241.85 \Omega$, $R_4 = 9,978.4 \Omega$, and $R_5 = 48.289 k\Omega$. Note that from (11.15), the *DC* gain for the second-order stage is $(R_3 \omega_z^2) / ((R_2 + R_3) \omega_o^2)$ or in this case, 4.703.

Now consider the first-order stage. Clearly from (11.62), the required value for μ in (11.2) is 2,026.33. Since the *DC* gain for the second-order stage is 4.703,

and the overall desired *DC* gain for the cascade of both stages is unity, it is desirable to have a magnitude *DC* gain for the first-order stage of 0.2126. This, in turn, requires a value for *G* in (11.2) of 0.2126. If an impedance scaling factor of 1,000 is used, the element values for the circuit shown in **Figure 11.3** are as follows: $R_1 = 1,000 \Omega$, $R_2 = 212.6 \Omega$, and the capacitor value is $2.321 \mu F$. The final active filter implementation schematic diagram is shown in **Figure 11.14**. Since the second-order stage has the larger gain, it is connected to the output of the first-order stage.

The magnitude frequency response, as obtained from SPICE analysis applied to **Figure 11.14** is shown in **Figure 11.15**. Note that the *DC* gain, as well as the peak passband gain, in *dB*, in **Figure 11.15** is 0 *dB*, as desired. It is also noted that $f_p = 1,000 \text{ Hz}$, $A_p = 3 \text{ dB}$, and $A_s = 40 \text{ dB}$, as specified. \square

Example 11.3

Suppose it is desired to design and implement a Chebyshev Type II bandpass filter with the following specifications: $f_o = 10,000 \text{ Hz}$, $B_p = 1,000 \text{ Hz}$, $B_s = 9,000 \text{ Hz}$, $A_p = 2 \text{ dB}$, and $A_s = 40 \text{ dB}$. From (5.16) (see page 295), the minimum bandpass filter order to meet the stated specifications is 4, and therefore the order required for the lowpass prototype is 2. For a normalized ω_p of unity for the lowpass prototype, the corresponding ω_s , from (5.9) is 8.1163 rad/s . From (5.20) the zeros are $\pm j 11.478$. From (5.22) the poles are $-0.80756 \pm j 0.81568$. Therefore the lowpass prototype transfer function is

$$H_{LPP}^{(norm)}(s) = \frac{K_1 (s^2 + 131.74)}{s^2 + 1.6151 s + 1.3175}$$

$$= \frac{K_1 (s + j 11.478)(s - j 11.478)}{(s + 0.80756 + j 0.81568)(s + 0.80756 - j 0.81568)}$$

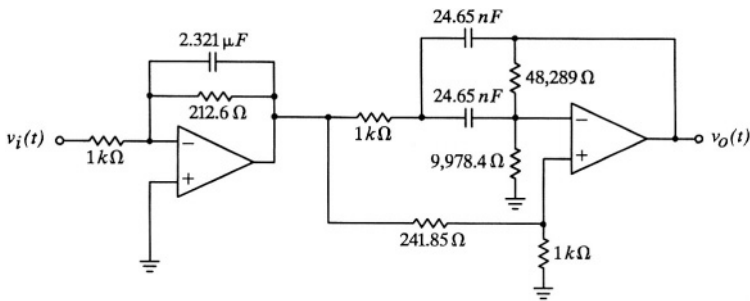


Figure 11.14 The active filter implementation circuit schematic diagram for **Example 11.2**.

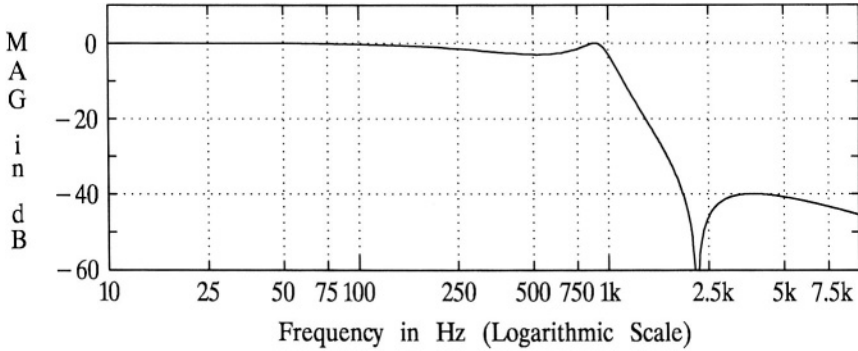


Figure 11.15 The magnitude frequency response of the circuit shown in Figure 11.14, as obtained by SPICE.

From **Section 9.3**, Case II, the new zeros, after applying the lowpass-to-bandpass transformation, are $\pm j 36,384.6$ and $\pm j 108,503.0$. From **Section 9.3**, Case III, the new poles, after applying the lowpass-to-bandpass transformation, are $-2,433.6 \pm j 60,270.5$ and $-2,640.5 \pm j 65,395.5$. Therefore, the bandpass transfer function to be implemented is as follows:

$$\begin{aligned}
 H(s) &= \frac{K_2 (s^2 + 1.3238 \times 10^9)}{s^2 + 4,867.2 s + 3.6385 \times 10^9} \frac{K_3 (s^2 + 1.1773 \times 10^{10})}{s^2 + 5,281 s + 4.2835 \times 10^9} \\
 &= H_1(s) H_2(s) \quad . \quad (11.63)
 \end{aligned}$$

Note that $H_1(s)$ is a highpass notch transfer function since $\omega_z^2 < \omega_o^2$, and that $H_2(s)$ is a lowpass notch transfer function since $\omega_z^2 > \omega_o^2$.

Implementation of (11.63) may be accomplished by cascading a highpass notch circuit and a lowpass notch circuit. Consider the highpass notch first. It must be determined what ω_z^2 must be for the normalized circuit. Clearly $\omega_z^2 = 1.3238 \times 10^9 / 3.6385 \times 10^9 = 0.36383$. Also, $Q = \sqrt{3.6385 \times 10^9} / 4,867.2 = 12.39$. For the normalized circuit, $R_1 = R_2 = 1 \Omega$, $R_3 = 575.4 \Omega$, $R_4 = 2,157.1 \Omega$, $C_1 = 0.03765 F$, and $C_2 = C_3 = 0.02153 F$. Frequency scaling by $\sqrt{3.6385 \times 10^9} = 60,320$, and impedance scaling by 1,000 at the same time, results in $R_1 = R_2 = 1,000 \Omega$, $R_3 = 575.4 k\Omega$, $R_4 = 2.157 M\Omega$, $C_1 = 624.2 pF$, and $C_2 = C_3 = 356.9 pF$.

Now consider the lowpass notch. Again, it must be determined what ω_z^2 must be for the normalized circuit. Clearly $\omega_z^2 = 1.1773 \times 10^{10} / 4.2835 \times 10^9 = 2.7485$. Also, $Q = \sqrt{4.2835 \times 10^9} / 5,281 = 12.39$. For the normalized circuit, $R_1 = R_3 = 1 \Omega$, $R_2 = 8.9521 m\Omega$, $R_4 = 351.19 \Omega$, $R_5 = 614.05 \Omega$, and

$C_1 = C_2 = 0.04036 F$. Frequency scaling by $\sqrt{4.2835 \times 10^9} = 65,448$, and impedance scaling by 10,000 at the same time, results in $R_1 = R_3 = 10 k\Omega$, $R_2 = 89.52 \Omega$, $R_4 = 3.512 M\Omega$, $R_5 = 6.141 M\Omega$, and $C_1 = C_2 = 61.67 pF$. The final active filter implementation schematic diagram is shown in **Figure 11.16**.

The magnitude frequency response, as obtained from SPICE analysis applied to **Figure 11.16** is shown in **Figure 11.17**. Note that the peak gain is 48.7 dB, and that in the stopband it is 8.7 dB, 40 dB below the peak. Note also that $f_o = 10,000 Hz$, $B_p = 1,000 Hz$, $A_p = 2 dB$, $A_s = 40 dB$, and B_s is only about 8 kHz, exceeding the initial specifications. The transmission zeros are at 5,790.8 Hz and 17.27 kHz, outside the frequency range plotted. □

11.4 CHAPTER 11 PROBLEMS

- 11.1 Indicate at least three advantages that active filter implementation has over passive implementation.
- 11.2 Indicate at least two application areas where passive implementation may be desired or required, over active implementation.
- 11.3 Using basic circuit analysis techniques, derive gain expressions for both the inverting and noninverting amplifiers shown in **Figure 11.2**.
- 11.4 Derive the transfer function for the circuit shown in **Figure 11.3**, that is, obtain (11.1).

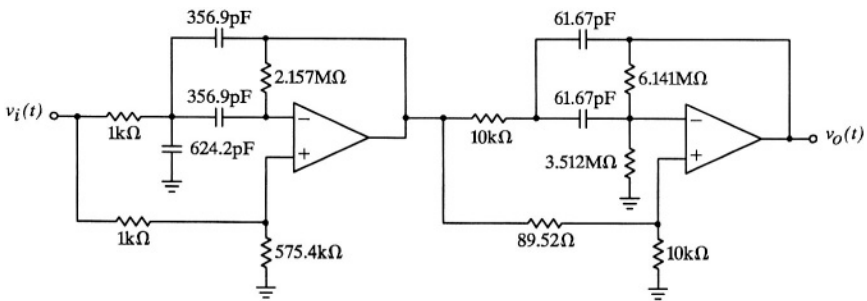


Figure 11.16 The active filter implementation circuit schematic diagram for *Example 11.3*.

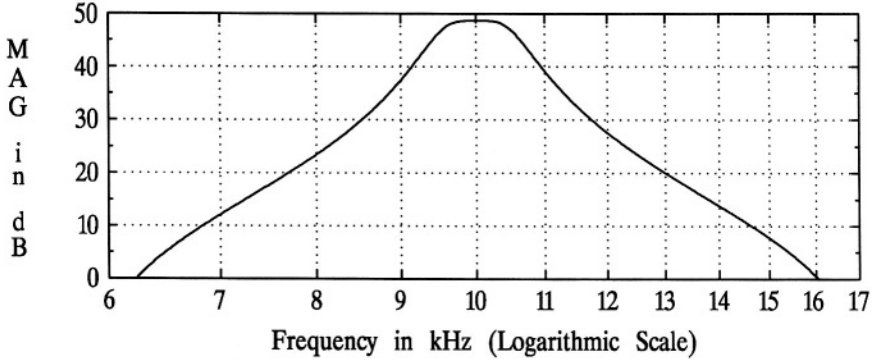


Figure 11.17 The magnitude frequency response of the circuit shown in **Figure 11.16**, as obtained by SPICE.

- 11.5** Suppose a transfer function such as shown in (11.1) is desired with a pole value of $-5,000$ and a DC magnitude gain of 10. Illustrate a least three circuit implementations as shown in **Figure 11.3** (three sets of component values) that will implement the desired transfer function.
- 11.6** Derive the transfer function for the circuit shown in **Figure 11.4**, that is, obtain (11.4).
- 11.7** Suppose a transfer function such as shown in (11.4) is desired with a pole value of $-5,000$ and a high-frequency magnitude gain of 10. Illustrate a least three circuit implementations as shown in **Figure 11.4** (three sets of component values) that will implement the desired transfer function.
- 11.8** Derive the transfer function for the circuit shown in **Figure 11.5**, that is, obtain (11.7).
- 11.9** Suppose a transfer function such as shown in (11.7) and (11.8) is desired (a first-order all-pass) with a pole value of $-5,000$. Illustrate a least three circuit implementations as shown in **Figure 11.5** (three sets of component values) that will implement the desired transfer function.
- 11.10** The Sallen and Key lowpass circuit is shown in **Figure 11.6**. Using basic circuit analysis techniques, derive the transfer function, as shown in (11.10).
- 11.11** Given that $Q = 3$, and beginning with the normalized component values given in (11.13), frequency scale the circuit shown in **Figure 11.6** so that

- $\omega_o = 1,000$. Also impedance scale by a factor of 1,000. Draw the schematic diagram, indicating all element values. Also give the transfer function for the scaled circuit. Indicate the values of the poles. If Q was changed to 2, what would the poles be?
- 11.12** For the circuit implementation obtained in **Problem 11.11** ($Q = 3$), using SPICE obtain the magnitude frequency response of the circuit.
- 11.13** Using the Sallen and Key lowpass second-order stage shown in **Figure 11.6**, design and implement a second-order Butterworth filter that has $f_c = 1,000 \text{ Hz}$. Let $R_1 = 1,000 \Omega$. Draw the complete schematic diagram with all component values shown. Using SPICE, obtain the magnitude frequency response of the circuit.
- 11.14** Using the Sallen and Key lowpass second-order stage shown in **Figure 11.6**, design and implement a second-order Chebyshev Type I filter that has 2 dB of ripple and $f_c = 1,000 \text{ Hz}$. Let $R_1 = 1,000 \Omega$. Draw the complete schematic diagram with all component values shown. Using SPICE, obtain the magnitude frequency response of the circuit.
- 11.15** One form of the second-order lowpass notch circuit is shown in **Figure 11.7**. Using basic circuit analysis techniques, derive the transfer function, as shown in (11.15)-(11.19).
- 11.16** Given that $\alpha = 0$ and ω_o is normalized to unity for (11.15), and that $Q = 3$ and $\omega_z = 2$, frequency scale the circuit shown in **Figure 11.7** so that $\omega_o = 1,000$. Also impedance scale using a scale factor of 1,000. Draw the schematic diagram, indicating all element values. Also give the transfer function for the scaled circuit. Indicate the values of the poles and zeros. If Q was changed to 2, what would the poles and zeros be?
- 11.17** Suppose it is desired to implement a second-order lowpass notch circuit such as shown in **Figure 11.7** with $f_o = 10 \text{ kHz}$, $Q = 12$, and $f_z = 17 \text{ kHz}$. Let $R_1 = 1,000 \Omega$. Determine all component values. Using SPICE, obtain the magnitude frequency response of the circuit. Record the value of the magnitude gain at 5 kHz, 20 kHz, and at the peak. Record the frequency of the peak magnitude gain as well.
- 11.18** One form of the Sallen and Key highpass circuit is shown in **Figure 11.8**. Using basic circuit analysis techniques, derive the transfer function, as shown in (11.26).

- 11.19** Beginning with the values in (11.29) for the normalized circuit, and given that $Q = 3$, frequency scale the circuit shown in **Figure 11.8** so that $\omega_o = 1,000$. Also impedance scale by a factor of 1,000. Draw the schematic diagram, indicating all element values. Also give the transfer function for the scaled circuit. Indicate the values of the poles. If Q was changed to 2, what would the poles be?
- 11.20** For the circuit implementation obtained in **Problem 11.19** ($Q = 3$), using SPICE obtain the magnitude frequency response of the circuit.
- 11.21** Using the Sallen and Key highpass second-order stage shown in **Figure 11.8**, design and implement a second-order Butterworth highpass filter that has $f_c = 1,000 \text{ Hz}$. Let $R_1 = 1,000 \Omega$. Draw the complete schematic diagram with all component values shown. Using SPICE, obtain the magnitude frequency response of the circuit.
- 11.22** Using the Sallen and Key highpass second-order stage shown in **Figure 11.8**, design and implement a second-order Chebyshev Type I highpass filter that has 2 dB of ripple and $f_c = 1,000 \text{ Hz}$. Let $R_1 = 1,000 \Omega$. Draw the complete schematic diagram with all component values shown. Using SPICE, obtain the magnitude frequency response of the circuit.
- 11.23** One form of the second-order highpass notch circuit is shown in **Figure 11.9**. Using basic circuit analysis techniques, derive the transfer function, as shown in (11.31)–(11.35).
- 11.24** Given that $\alpha = 0$ and ω_o is normalized to unity for (11.31), and that $Q = 3$ and $\omega_z = 0.6$, frequency scale the circuit shown in **Figure 11.9** so that $\omega_o = 1,000$. Also impedance scale using a scale factor of 1,000. Draw the schematic diagram, indicating all element values. Also give the transfer function for the scaled circuit. Indicate the values of the poles and zeros. If Q was changed to 2, what would the poles and zeros be?
- 11.25** Suppose it is desired to implement a second-order highpass notch circuit such as shown in **Figure 11.9** with $f_o = 10 \text{ kHz}$, $Q = 12$, and $f_z = 6 \text{ kHz}$. Let $R_1 = 1,000 \Omega$. Determine all component values. Using SPICE, obtain the magnitude frequency response of the circuit. Record the value of the magnitude gain at 5 kHz, 20 kHz, and at the peak. Record the frequency of the peak magnitude gain as well.

- 11.26** One form of the Friend bandpass circuit is shown in **Figure 11.10**. Using basic circuit analysis techniques, derive the transfer function, as shown in (11.43)-(11.46).
- 11.27** Beginning with the values in (11.50) for the normalized circuit, and given that $Q = 3$, frequency scale the circuit shown in **Figure 11.10** so that $\omega_o = 1,000$. Also impedance scale by a factor of 1,000. Draw the schematic diagram, indicating all element values. Also give the transfer function for the scaled circuit. Indicate the values of the poles. If Q was changed to 2, what would the poles be?
- 11.28** For the circuit implementation obtained in **Problem 11.27** ($Q = 3$), using SPICE obtain the magnitude frequency response of the circuit.
- 11.29** Using the Friend bandpass second-order stage shown in **Figure 11.10**, design and implement a second-order bandpass filter that has $f_o = 1,000$ Hz and $Q = 10$. Let $R_1 = 1,000 \Omega$. Draw the complete schematic diagram with all component values shown. Using SPICE, obtain the magnitude frequency response of the circuit. What is the value of the peak magnitude gain, and what is the 3 dB bandwidth of the response?
- 11.30** One form of the Delyiannis second-order all-pass circuit is shown in **Figure 11.11**. Using basic circuit analysis techniques, derive the transfer function, as shown in (11.51)-(11.55).
- 11.31** Given that $\alpha = -\omega_o / Q$ and ω_o is normalized to unity for (11.51), and that $Q = 3$, frequency scale the circuit shown in **Figure 11.11** so that $\omega_o = 1,000$. Also impedance scale by a factor of 1,000. Draw the schematic diagram, indicating all element values. Also give the transfer function for the scaled circuit. Indicate the values of the poles and zeros. If Q was changed to 2, what would the poles and zeros be?
- 11.32** For the circuit implementation obtained in **Problem 11.31** ($Q = 3$), using SPICE obtain the magnitude and phase frequency response of the circuit.
- 11.33** Suppose it is desired to implement a second-order allpass circuit such as shown in **Figure 11.11** with $f_o = 10$ kHz and $Q = 5$. Let $R_1 = 1,000 \Omega$. Determine all component values. Using SPICE, obtain the magnitude and phase frequency response of the circuit. Record the value of the phase at 9 kHz, 10 kHz, and 11 kHz.

- 11.34** Summarize the procedure for active filter implementation presented in this chapter.
- 11.35** Go through the details of implementing, using op amp circuits, a fourth-order Chebyshev Type I lowpass filter with $\omega_p = 1,000 \text{ rad/s}$ and $\epsilon = 0.5$ (0.969 dB ripple). Draw the final circuit schematic diagram indicating all element values. Using SPICE, plot the magnitude frequency response of your resultant circuit. That is, carry out the details of *Example 11.1*.
- 11.36** Suppose it is desired that the circuit shown in **Figure 11.12** be modified such that the magnitude DC gain be unity. Show how this can be done by including an additional gain stage (see **Figure 11.2**).
- 11.37** Suppose it is desired that the circuit shown in **Figure 11.12** be modified such that the magnitude DC gain be unity, but without adding a gain stage such as in **Problem 11.36**, but rather by modifying the input resistors of each stage (R_1 in **Figure 11.6**). For convenience, let the DC gain of each stage be unity. Let each input $1 \text{ k}\Omega$ resistor be replaced by a horizontal resistor as shown with an additional resistor, on the right, to ground. In each case, the resistance looking into the right side of the resistor combination must be $1 \text{ k}\Omega$, but a desired signal attenuation from the input to the output of the resistor combination can be the appropriate value to obtain a DC gain of unity for the stage. That is, the two input $1 \text{ k}\Omega$ resistor values are to be changed, and additional resistors are to be added from the right side of the input resistors to ground. Determine the required values of these four resistors. This method of gain adjustment only requires two more resistors. Using SPICE, plot the magnitude frequency response of your resultant circuit.
- 11.38** Implement a fourth-order Butterworth lowpass filter with $f_c = 1,000 \text{ Hz}$. Show the details of your calculations. Draw the final op amp circuit schematic diagram indicating the values of all circuit elements. Using SPICE, plot the magnitude frequency response of your circuit implementation. Also plot the magnitude frequency response of each stage in your implementation.
- 11.39** Repeat **Problem 11.38** for a fifth-order Butterworth lowpass filter.
- 11.40** Go through the details of implementing, using op amp circuits, a third-order elliptic lowpass filter with $f_p = 1,000 \text{ Hz}$, $A_p = 3 \text{ dB}$, and $A_s = 40 \text{ dB}$. Draw the final circuit schematic diagram indicating all element values. Using SPICE, plot the magnitude frequency response of your resultant circuit. That is, carry out the details of *Example 11.2*.

- 11.41** Implement a fourth-order Butterworth highpass filter with $f_c = 1,000 \text{ Hz}$. Show the details of your calculations. Draw the final op amp circuit schematic diagram indicating the values of all circuit elements. Using SPICE, plot the magnitude frequency response of your circuit implementation. Also plot the magnitude frequency response of each stage in your implementation.
- 11.42** Implement, using op amp circuits, a third-order elliptic highpass filter with $f_p = 1,000 \text{ Hz}$, $A_p = 3 \text{ dB}$, and $A_s = 40 \text{ dB}$. Draw the final circuit schematic diagram indicating all element values. Using SPICE, plot the magnitude frequency response of your resultant circuit.
- 11.43** Go through the details of implementing, using op amp circuits, a fourth-order Chebyshev Type II bandpass filter with the following specifications: $f_o = 10,000 \text{ Hz}$, $B_p = 1,000 \text{ Hz}$, $B_s = 9,000 \text{ Hz}$, $A_p = 2 \text{ dB}$, and $A_s = 40 \text{ dB}$. Draw the final circuit schematic diagram indicating all element values. Using SPICE, plot the magnitude frequency response of your resultant circuit. That is, carry out the details of *Example 11.3*.
- 11.44** Perform SPICE analysis on the circuit shown in **Figure 11.16**, as in **Problem 11.43**, confirming the results shown in **Figure 11.17**. In addition, plot the magnitude frequency response of each of the two stages. For each stage, record the magnitude of the peak gain and the frequency at which it occurs. Perform an overlay plot with the overall magnitude gain for the circuit, and the magnitude gain of each stage, allowing ready comparison of the three frequency response plots.
- 11.45** Suppose it is desired to design and implement a Butterworth bandpass filter with the following specifications: $f_o = 10,000 \text{ Hz}$, $B_p = 1,000 \text{ Hz}$, $B_s = 5,000 \text{ Hz}$, $A_p = 3 \text{ dB}$, and $A_s = 40 \text{ dB}$.
- Determine the minimum required order for the bandpass filter.
 - Using the minimum required order determined in part (a), determine the actual value for B_s that will be obtained, holding all other specifications constant.
 - Determine the transfer function, with all values real and specified except for an arbitrary gain constant, for the bandpass filter.
 - Draw the final active filter implementation schematic diagram, with the order of cascaded stages with increasing Q , with all element values indicated.

- (e) Using SPICE, on the same graph, plot the overall magnitude frequency response, and the magnitude frequency response of each of the individual stages. Note how the individual-stage frequency responses add (in dB) to yield the overall response, and how the individual stages have differing center frequencies (this is known as stagger tuning).
- (f) Plot the overall phase response.
- (g) Plot the overall unit impulse response.
- 11.46** Repeat **Problem 11.45** with $A_s = 50 \text{ dB}$, and all other specifications unchanged.
- 11.47** Repeat **Problem 11.45** for an elliptic bandpass filter that meets or exceeds the following specifications: passband ripple = $A_p = 0.1 \text{ dB}$, minimum attenuation in the stopband relative to the peak response = $A_s = 60 \text{ dB}$, $B_p = 1,000 \text{ Hz}$, $B_s = 8,500 \text{ Hz}$, and $f_o = 10,000 \text{ Hz}$.
- 11.48** Design and implement an elliptic bandstop filter that meets or exceeds the following specifications: $A_p = 1 \text{ dB}$, $B_p = 200 \text{ rad/s}$, $\omega_o = 1000 \text{ rad/s}$ and $A_s = 80 \text{ dB}$. Draw the complete op amp circuit schematic diagram with all element values shown. Using SPICE, plot the magnitude frequency response of the circuit.

APPENDIX A

INTRODUCTION TO MATLAB

MATLAB is a product of The MATH WORKS, Inc., and is a high-performance numeric computation and visualization software package. The name “MATLAB” is an abbreviation for “MATrix LABoratory,” and is noted for the way matrix computations are performed by simple one-line algebraic code. For example, in MATLAB, the command line $A = B * C$ will multiply previously-defined matrices B and C . As another simple example, $A = B / C$ is matrix B times the inverse of matrix C , and is the solution of the equation $A * C = B$. Similarly, $A = C \backslash B$ is the solution of $C * A = B$.

MATLAB, however, has grown far beyond basic matrix algebra. It includes strong graphics capabilities and many m-file functions; m-file functions are ascii files of MATLAB commands that perform high-level computation, similar to a subroutine. Basic MATLAB commands and m-file functions are used without distinction by the user. For example, *buttap* is an m-file function that designs an analog Butterworth filter of a given order and a normalized $\omega_c = 1$: $[z, p, k] = \text{buttap}(n)$, where n is the order, z is a returned vector¹ of transfer function zeros, p is a returned vector of poles, and k is a returned scalar gain value.

A user may create his own m-files of two types: a simple list of MATLAB commands and functions, or user-defined functions. A simple m-file list of commands is simply a time-saving procedure and enables easy editing and de-bugging, similar to a script file in UNIX or other operating systems. A user-defined *function* is a new function written by the user, which then becomes part of the MATLAB library and is used without distinction between it and functions that come with MATLAB.

The most powerful version of MATLAB is the Professional Version; the maximum array sizes are very large, limited only by the machine it is run in, the graphics capabilities include presentation quality fonts, etc. Along with the basic Professional Version are many Toolboxes, which are specialized application m-files. The Toolboxes include Signals and Systems, Optimization, Image Processing, Higher-Order Statistics, Neural Networks, etc. There are more than twenty such Tool Boxes,

¹An array is a collection of numbers, whereas a vector is either an N-by-1 or 1-by-N matrix subject to certain rules of operation. No distinction is made here between an array and a vector.

with more under development. The Student Edition of MATLAB, Version 5, while less capable than the Professional Version, and does not include any Toolboxes, except that some of the Signals and Systems capabilities are included, is nevertheless a quite powerful software package.

Please note that what is presented below is very basic, and intended primarily to indicate some of the capabilities of MATLAB applied to the design and analysis of analog filters. Please refer to *The Student Edition of MATLAB, Version 4, User's Guide*, Prentice Hall, 1995, or *The Student Edition of MATLAB, Version 5, User's Guide*, Prentice Hall, 1997 for details. The version 4 User's Guide includes a listing of MATLAB commands with brief instructions on their use, whereas the version 5 User's Guide does not. Both versions give tutorials, examples, and basic instruction. Also note that online help is available: within MATLAB, `help <Enter>` will list the command names that are available, and `help command_name <Enter>` will give a description of the command. Also online are demos and examples.

A.1 BASIC COMMANDS

Matrix operations:

$A = B*C$ Matrices B and C must have been previously defined and the dimensions must be compatible. The product A will be displayed on the screen.

$A = B*C;$ As above, but the product A will not be displayed on the screen.

$A = B/C;$ Solution to $A *C = B$.

$A = C\backslash B;$ Solution to $C*A = B$.

$A = \text{inv}(B);$ A will be the inverse of the square matrix B .

Vector operations:

$a = b'*c;$ Scalar a is the sum of the products of the elements of vectors b and c . The symbol $'$ denotes conjugate transpose ($.'$ denotes transposition without conjugation). This is the vector "dot" product. The assumption here is that both b and c are column vectors.

- $A = b * c'$; Matrix A is the cross, or outer, product of vectors b and c .
- $d = b .* c$; Vector d is the element-by-element product of vectors b and c , i.e., $d(1) = b(1) * c(1)$, etc.

A.2 SPECIALIZED COMMANDS

Analog filter design commands:

- $[z,p,k] = \text{butter}(n)$; Designs a Butterworth lowpass filter with normalized $\omega_c = 1$ for a given order n . Returns the zeros in vector z , the poles in vector p , and the gain in scalar k .
- $[z,p,k] = \text{cheb1ap}(n,Rp)$; Designs a Chebyshev Type I lowpass filter for given order n and passband ripple Rp .
- $[z,p,k] = \text{cheb2ap}(n,Rs)$; Designs a Chebyshev Type II lowpass filter for given order n and stopband ripple Rs .
- $[z,p,k] = \text{ellipap}(n,Rp,Rs)$; Designs an elliptic lowpass filter for given order n , passband ripple Rp , and stopband ripple Rs .

Analog filter frequency transformations: $lp2bp$, $lp2bs$, $lp2hp$ and $lp2lp$

These commands transform a normalized lowpass prototype filter design into a bandpass, bandstop, highpass, or another lowpass with different cutoff frequency, respectively.

Transfer function form changes:

- $p = \text{poly}(r)$; If r is a vector of poles or zeros, then p will be a vector of polynomial coefficients.
- $r = \text{roots}(p)$; If p is a vector of polynomial coefficients, then r will be a vector of poles or zeros.

Frequency response:

$w = 0:0.02:10;$ Establishes a vector of radian frequencies linearly spanning the range from 0 to 10 in 0.02 radians/sec steps.

$H = freqs(b,a,w);$ Given the vector b of transfer function numerator polynomial coefficients, the vector a of denominator coefficients, and the vector w of radian frequencies, H will be a vector of frequency response samples.

$HM = abs(H);$ The vector HM will be the magnitude frequency response.

$HMD = 20*log10(HM);$ The vector HMD will be the magnitude frequency response in dB.

$HP = angle(H);$ The vector HP will be the phase response.

Time-domain responses:

$[h,x,t] = impulse(b,a);$ Given the transfer function polynomial vectors, b for the numerator and a for the denominator, for a normalized analog filter transfer function, the impulse response vector h and the corresponding vector of time values t will be returned. The vector x is an internal state vector response.

$[r,x,t] = step(b,a);$ Given the transfer function polynomial vectors, b for the numerator and a for the denominator, for a normalized analog filter transfer function, the step response vector r and the corresponding vector of time values t will be returned. The vector x is an internal state vector response.

Graphing: See *plot*, *title*, *xlabel*, *ylabel*, *grid*, *legend*, *hold*, etc.

A.3 SUMMARY OF COMMANDS

Listed below are the MATLAB commands most relevant to the study of this book, with a very brief description of each. Refer to a MATLAB manual², or a MATLAB online *help* for additional information.

abs

Given a data array of complex values, such as samples of the frequency response of a transfer function, this command will produce a corresponding array of absolute values (magnitude) of the complex values. For example, if $MFR = abs(FR)$, the elements of the array MFR will be the absolute values of the elements of the complex array FR .

angle

Given a data array of complex values, such as samples of the frequency response of a transfer function, this command will produce a corresponding array of values that are the principal-value phase angles, in radians, of the complex values. For example, if $PFR = angle(FR)$, the elements of the array PFR will be the phase-angle values of the elements of the complex array FR .

axis

This command allows the user to specify axis ranges for the current plot. This allows plotting, for example, only part of a data file. For example, $axis([xmin xmax ymin ymax])$ will only plot that part of the data file for independent-variable values from $xmin$ to $xmax$, and the vertical axis will be from $ymin$ to $ymax$.

buttap

Given the order of an analog Butterworth lowpass filter, this command will return arrays of the filter zeros and poles, and also a scalar gain term, for a filter cutoff frequency of 1 radian/second. There are no finite-value zeros for an analog Butterworth lowpass filter, but the array of zeros is included to be consistent with the format of other filter design commands. For example, if $[BUTZ, BUTP, BUTK] = buttap(7)$, the array $BUTZ$ will be empty, the array $BUTP$ will have the seven complex pole values in it, and the scalar $BUTK$ will be unity.

²Refer to *The Student Edition of MATLAB, Version 4, User's Guide*, Prentice Hall, 1995, or an appropriate manual of the professional version of MATLAB.

cheb1ap

Given the order of an analog Chebyshev Type I lowpass filter and the passband ripple in dB , this command will return arrays of the filter zeros and poles, and also a scalar gain term, for a filter cutoff frequency of 1 radian/second. There are no finite-value zeros for an analog Chebyshev Type I lowpass filter, but the array of zeros is included to be consistent with the format of other filter design commands. For example, if $[CZ, CP, CK] = cheb1ap(8,1)$, the array CZ will be empty, the array CP will have the eight complex pole values in it, and the scalar CK will be such that the peak gain of the filter will be unity, for a filter with 1 dB of passband ripple.

cheb2ap

Given the order of an analog Chebyshev Type II lowpass filter and the minimum stopband attenuation in dB , this command will return arrays of the filter zeros and poles, and also a scalar gain term, for a filter stopband edge frequency of 1 radian/second. For example, if $[C2Z, C2P, C2K] = cheb2ap(8,60)$, the array $C2Z$ will have the eight imaginary-axis zeros in it, the array $C2P$ will have the eight complex pole values in it, and the scalar $C2K$ will be such that the peak gain of the filter will be unity, for a filter with 60 dB of stopband attenuation.

clc

This command will clear the command window.

close

This command will close (erase) the current figure window and return to the command window.

demo

This command will execute the MATLAB demonstration.

ellipap

Given the order of an analog elliptic lowpass filter, the ripple in the passband in dB , and the minimum stopband attenuation in dB , this command will return arrays of the filter zeros and poles, and also a scalar gain term, for a filter cutoff frequency of 1 radian/second. For example, if $[EZ, EP, EK] = ellipap(9,1.5,75)$, the array EZ will have the eight imaginary-axis zeros in it, the array EP will have the nine complex pole values in it, and the scalar EK will be such that the peak gain of the filter will be unity, for a filter with 1.5 dB of passband ripple and 75 dB of stopband attenuation.

freqs

Given vectors of filter transfer function numerator and denominator coefficients and radian frequency values, this command will return a vector of complex frequency response values. For example, if $FR = freqs(NUM, DEN, w)$,

where NUM contains the transfer function numerator polynomial coefficients, DEN contains the denominator coefficients, and w contains the radian frequency sample values, the array FR will contain samples of the frequency response, real and imaginary parts, for the frequencies specified in w . Note that there are variations for this command, e.g., it can choose its own frequency samples, etc.

grid

This command will provide grid lines for plots.

gtext

This command allows mouse placement of text on the current graph. For example, if the command *gtext('Note the first zero-crossing!')* is issued, then a cross will appear on the figure. Place the cross, by moving the mouse, where the extreme lower left of the text is to be placed, and click the mouse.

help

The command *help*, by itself, will list the primary help topics. The command *help topic*, will give help on that topic. The command *help command*, will give help on that specific command.

hold

This command will hold the current graph. Subsequent plotting will add to the existing graph without erasing it first. To return to the normal mode, where a new plot command will erase any existing graphs first, issue the following command: *hold off*.

impulse

Given vectors of filter transfer function numerator and denominator coefficients and time values, this command will return a vector of impulse response values. For example, if $[IMP,x,t] = \text{impulse}(NUM,DEN,t)$, where NUM contains the transfer function numerator polynomial coefficients, DEN contains the denominator coefficients, and t contains the time sample values, the array IMP will contain samples of the impulse response, for the time values specified in t . Note that this command should only be used with filters that have a cutoff frequency (lowpass and highpass) or center frequency (bandpass and bandstop) at or near unity. Note also that the impulse that occurs at $t = 0$ for highpass and bandstop filters will not appear in the results; the results, while appearing continuous in a plot, are made up only of samples of the response; even if $t = 0$ is specifically in the t vector, it should be taken as 0^+ , missing the impulse. Note also that there are a number of variations for this command.

legend

This command will add a legend to the current graph. It is very useful when several plots have been overlaid: the legend will identify each individual plot.

log10

Given a data array, this command will produce a corresponding array of values that are the common logarithm, element-by-element, of the input data array values. For example, if $LOGX = \log_{10}(X)$, the elements of the array $LOGX$ will be the common logarithm of the elements of the array X . Note that the command \log is the natural logarithm.

lp2bp

Given vectors of analog lowpass prototype filter transfer function numerator and denominator coefficients for a cutoff frequency of 1 radian/second, the radian center frequency, and the radian frequency bandwidth, this command will return the filter coefficients of a bandpass filter. For example, if $[NUMBP, DENBP] = lp2bp(NUM, DEN, WO, BW)$, where NUM contains the lowpass transfer function numerator polynomial coefficients, DEN contains the denominator coefficients, WO is the radian center frequency, and BW is the radian bandwidth, the array $NUMBP$ will contain the bandpass transfer function numerator polynomial coefficients, and $DENBP$ will contain the bandpass denominator coefficients. Note that there are a number of variations for this command.

lp2bs

Given vectors of analog lowpass prototype filter transfer function numerator and denominator coefficients for a cutoff frequency of 1 radian/second, the radian center frequency, and the radian frequency bandwidth, this command will return the filter coefficients of a bandstop filter. For example, if $[NUMBS, DENBS] = lp2bs(NUM, DEN, WO, BW)$, where NUM contains the lowpass transfer function numerator polynomial coefficients, DEN contains the denominator coefficients, WO is the radian center frequency, and BW is the radian bandwidth, the array $NUMBS$ will contain the bandstop transfer function numerator polynomial coefficients, and $DENBS$ will contain the bandstop denominator coefficients. Note that there are a number of variations for this command.

lp2hp

Given vectors of analog lowpass prototype filter transfer function numerator and denominator coefficients for a cutoff frequency of 1 radian/second, and the desired radian cutoff frequency of a highpass filter, this command will return the filter coefficients of a highpass filter. For example, if $[NUMHP, DENHP] = lp2hp(NUM, DEN, WO)$, where NUM contains the lowpass transfer function numerator polynomial coefficients, DEN contains the denominator coefficients, and WO is the

desired highpass radian cutoff frequency, the array *NUMHP* will contain the highpass transfer function numerator polynomial coefficients, and *DENHP* will contain the highpass denominator coefficients. Note that there are a number of variations for this command.

lp2lp

Given vectors of analog lowpass prototype filter transfer function numerator and denominator coefficients for a cutoff frequency of 1 radian/second, and the desired new cutoff frequency, this command will return the filter coefficients of a lowpass filter with the desired cutoff frequency. For example, if $[NUMLP, DENLP] = lp2lp(NUM, DEN, WO)$, where *NUM* contains the prototype lowpass transfer function numerator polynomial coefficients, *DEN* contains the prototype denominator coefficients, and *WO* is the desired lowpass radian cutoff frequency, the array *NUMLP* will contain the new lowpass transfer function numerator polynomial coefficients, and *DENLP* will contain the new lowpass denominator coefficients. Note that there are a number of variations for this command.

pause

This command in an m-file will halt execution until a key is pressed.

plot

This command will produce a two-dimensional plot of one or more arrays of data. There are many features. See below for some examples.

poly

Given an array of polynomial roots, e.g., poles or zeros, this command will return corresponding polynomial coefficients. For example, if $DEN = poly(POLES)$, where *POLES* is an array of complex transfer function poles, then *DEN* will be an array of transfer function denominator polynomial coefficients.

print

This command will plot the current figure on a specified printer. There are many options with this command, but if the defaults are properly set up, the simple command *print* will produce the desired hardcopy graph.

roots

Given an array of polynomial coefficients, this command will return corresponding polynomial roots. For example, if $POLES = roots(DEN)$, where *DEN* is an array of transfer function denominator coefficients, then *POLES* will be an array of transfer function complex poles.

step

Given vectors of filter transfer function numerator and denominator coefficients and time values, this command will return a vector of step response values. For example, if $[STP,x,t] = \text{step}(NUM,DEN,t)$, where NUM contains the transfer function numerator polynomial coefficients, DEN contains the denominator coefficients, and t contains the time sample values, the array STP will contain samples of the step response, for the time values specified in t . Note that this command should only be used with filters that have a cutoff frequency (lowpass and highpass) or center frequency (bandpass and bandstop) at or near unity. Note also that the impulse that occurs at $t = 0$ for highpass and bandstop filters will not appear in the results; the results, while appearing continuous in a plot, are made up only of samples of the response; even if $t = 0$ is specifically in the t vector, it should be taken as 0^+ , missing the impulse. Note also that there are a number of variations for this command.

text

This command is similar to *gtext* except that it does not involve the mouse, and is therefore more useful within m-files. It allows placement of text on the current graph. For example, if the command $\text{text}(1.2,4.5, 'Note\ the\ first\ zero-crossing!')$ is issued, then the extreme lower left of the text will be placed at $x=1.2$, $y = 4.5$, in units of the current plot data.

title

This commands will print *text* as a title at the top of the current plot. For example, $\text{title}('This\ is\ the\ title\ of\ this\ plot.')$

unwrap

A command such as *angle* will produce only principal-value angles. The command *unwrap* will, in most cases, unwrap, or yield the actual, or complete angles. Given a data array of phase angles, such as samples of the phase response of a transfer function, this command will produce a corresponding array of unwrapped phase angle values, in radians. For example, if $UPFR = \text{unwrap}(PFR)$, the elements of the array $UPFR$ will be the unwrapped phase-angle values of the elements of the array PFR .

who, whos

The command *who* lists the variables currently in memory. The command *whos* not only lists the variables, but also their sizes and whether they have real or complex elements.

xlabel

This command will add *text* beneath the x-axis of the current two-dimensional graph. For example, $\text{xlabel}('Frequency\ in\ Hz')$.

ylabel

This command will add *text* beside the y-axis of the current two-dimensional graph. For example, *ylabel ('Magnitude in dB')*.

A.4 EXAMPLES**Example A.1**

The following is an example of an m-file that will generate two periods of a sinuswave and plot the result. The plot is shown in **Figure A.1**.

```

TT = 0:1:500;           % Generates an array of integers from 0 to 500.
XX = TT*(4*pi/500);    % Generates an array of real numbers from 0
                        % to 4*pi.
YY = sin(XX);
plot(XX,YY)
xlabel('Radian Values')
ylabel('Sinusoid Amplitude')
title('Plot of two periods of a sinusoid'), pause, close

```

Example A.2

The following is an example of an m-file that will design a sixth-order Butterworth filter, and plot the magnitude frequency response and the phase response. The resulting plots are shown in **Figures A.2** and **A.3**.

```

[z,p,k] = buttap(6);    % Designs the Butterworth filter.
NUM = poly(z);
DEN = poly(p);         % Converts the poles into polynomial
                        % coefficients.
w = 0:0.02:10;        % Generates the vector of radian frequencies.
H = freqs(NUM,DEN,w); % Computes the complex frequency response.
HM = 20*log10(abs(H)); % Computes the magnitude in dB.
HP = unwrap(angle(H)); % Computes the unwrapped phase response.
plot(w,HM)            % Plots the magnitude response.
grid
xlabel('Radian Frequency')
ylabel('Magnitude in dB')
title('Magnitude Response of a 6th-Order Butterworth Filter')
pause                % Waits for a keyboard response.
plot(w,HP)           % Erases the magnitude plot and plots the phase.
grid
xlabel('Radian Frequency')

```

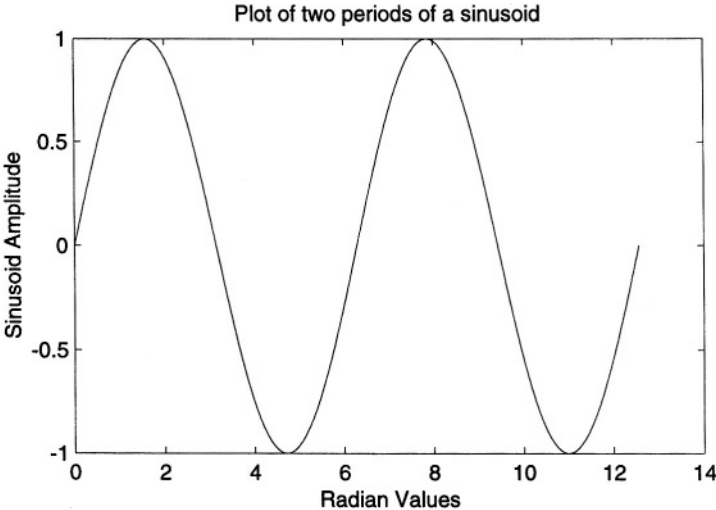


Figure A.1 Plot results for *Example A.1*.

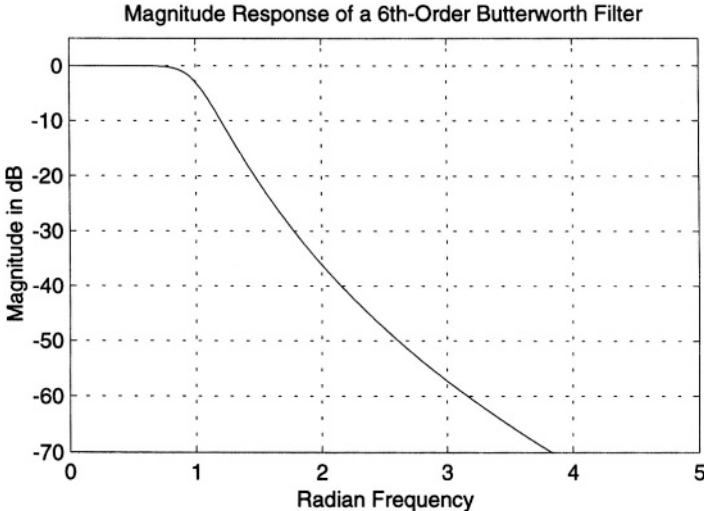


Figure A.2 Magnitude plot results for *Example A.2*.

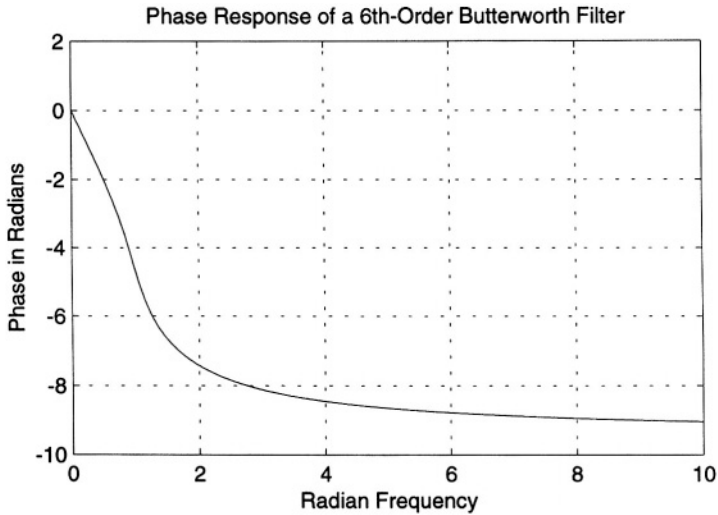


Figure A.3 Phase plot results for *Example A.2*.

```
ylabel('Phase in Radians')
title('Phase Response of a 6th-Order Butterworth Filter')
pause
close % Erases the figure.
```

Example A.3

The following is an example of an m-file that will design a fourth-order Butterworth filter and a fourth-order Chebyshev filter with 1 dB of ripple, and overlay-plot the magnitude frequency response of the two filter designs, both with a 3 dB cutoff frequency of unity. The plot is shown in **Figure A.4**.

```
[z,p,k] = buttap(4); % Designs the 4th-order Butterworth filter.
NUMB = poly(z);
DENB = poly(p); % Converts the poles into polynomial
% coefficients.
[z,p,k] = cheblap(4,1); % Designs the 4th-order Chebyshev filter with
% 1 dB of ripple.
```

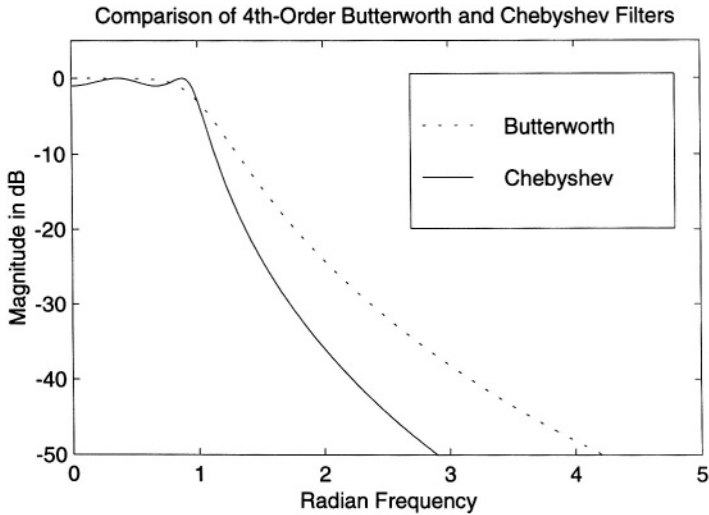


Figure A.4 Plot results for *Example A.3*.

```

ep = sqrt(10^0.1 - 1);      % Calculates the ripple factor epsilon.
WP = 1/cosh(0.25*acosh(1/ep)); % Calculates the desired wp frequency
                                % for a 3 dB frequency of 1.

NUM = poly(z);
DEN = poly(p);
[NUMC,DENC] = lp21p(NUM,DEN,WP); % Frequency scales for a
                                % 3 dB frequency of 1.

w = 0:0.02:10;             % Generates an array of frequencies from 0 to
                                % 10.

HB = freqs(NUMB,DENB,w);   % Obtains the Butterworth frequency
                                % response.
HC = freqs(NUMC,DENC,w).*k; % Obtains the Chebyshev frequency
                                % response.

HBM = 20*log10(abs(HB));    % Magnitude response in dB.
HCM = 10*log10(abs(HC));    % Magnitude response in dB.
plot(w,HBM,'-',w,HCM,'-')  % Plot both magnitude responses.
xlabel('Radian Frequency')
ylabel('Magnitude in dB')

```

```
title('Comparison of 4th-Order Butterworth and Chebyshev Filters')  
legend('Butterworth', 'Chebyshev')  
axis([0 5 -50 5])           % Set the axes ranges for more convenient  
                             % viewing.  
  
pause  
close
```

APPENDIX B

CONTENTS OF THE ACCOMPANYING DISK

The accompanying disk contains a number of MATLAB functions that extend basic MATLAB capabilities in terms of the design and analysis of analog filters. The disk also contains several m-files used in a number of examples in the book. They are included here as an instructional aid. These functions and m-files are intended to be used with MATLAB, version 5, *Student Edition*, and are not stand-alone. The MATLAB functions and m-files are organized below by the chapter in which they first appear.

B.1 CHAPTER 6 FUNCTIONS AND m-FILES

EXAMP6_1.m, *EXAMP6_2.m*, *EXAMP6_6.m*, *EXAMP6_8.m*, and *EXAMP6_9.m* are m-files that generate the data for *Examples 6.1, 6.2, 6.6, 6.8, and 6.9*, respectively.

ELLIPFS is a MATLAB function that computes the *Filter Selectivity* of an elliptic filter.

ELLIPINT is a MATLAB function that evaluates the elliptic integral of the first kind. The algorithm is an approximation to equation (6.4).

ELLIPOR is a MATLAB function that computes the minimum required order for a lowpass elliptic filter. The algorithm implements equation (6.40).

ELLIPPV is a MATLAB function that determines the frequencies of the passband peaks and valleys in the magnitude frequency response of an elliptic filter.

ELLIPZ is a MATLAB function that determines the poles and zeros of an elliptic filter.

ELLIPSF is a MATLAB function that computes the *Shaping Factor* of an elliptic filter.

ELLIPSN is a MATLAB function that returns the value of the Jacobi elliptic sine function, and is not restricted to a real argument.

ELLIPWC is a MATLAB function that determines the 3 dB cutoff frequency of an elliptic filter.

ELLIPWM is a MATLAB function that computes the finite-valued frequencies in the stopband of an elliptic filter where the attenuation is minimum.

ELLIPWS is a MATLAB function that computes ω_s for an elliptic filter.

ELLIPWZ is a MATLAB function that computes the finite-valued zero-transmission frequencies in the stopband of an elliptic filter.

B.2 CHAPTER 7 FUNCTIONS

BESSELDD is a MATLAB function that designs a Bessel filter for a normalized time delay of unity.

BESSELDE is a MATLAB function that designs a Bessel filter for a specified ω_p , A_p , and N .

BESSELS is a MATLAB function that computes the *Filter Selectivity* for a Bessel filter.

BESSELOR is a MATLAB function that computes the minimum required order for a Bessel filter to meet given specifications.

BESSELSF is a MATLAB function that computes the *Shaping Factor* for a Bessel filter.

B.3 CHAPTER 8 FUNCTIONS AND m-FILES

FIG8_1dat.m through *FIG8_7dat.m* are MATLAB m-files that generate the data for **Figures 8.1** through **8.7**. They illustrate transitional filters.

GAUSDE is a MATLAB function that designs a Gaussian filter for given ω_p , A_p , and N .

HALPDE is a MATLAB function that designs a Halpern filter for given ω_p , A_p , and N .

LEGENDE is a MATLAB function that designs a Legendre, 1st-associated Legendre, or a 2nd-associated Legendre, filter for given ω_p , A_p , and N .

PAPOULD is a MATLAB function that designs a Papoulis filter for given ω_p , A_p , and N .

ULTRADE is a MATLAB function that designs ultraspherical filters for given ω_p , A_p , and N .

APPENDIX C

THE MATLAB m-FILE EXAMP6_1.m

```
%
% File Name: EXAMP6_1.m
%
% This MATLAB m-file demonstrates that a Chebyshev filter is a special case of
% an elliptic filter, and that in this case, the elliptic sine form and the elliptic
% cosine form of (6.2) are identical. All three filter responses, in this example,
% are third-order Chebyshev with 3 dB of passband ripple.
%
%
% This is Example 6.1.
%
%
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
%
%
% Enter Parameters:
%     kappa = 3;
%     AC = 1000;
%     tau1 = 0;
% Note that tau2 = 0 is not necessary, since sin and cos are used in place of sn and
% en, respectively.
%
% Generate an array of radian frequency values going from 0 to 5 radians/second:
%     nn = 0:1:500;
%     w = nn/100;
%
% Produce the array of values for the upper limit of the elliptic integral of the first
% kind for the sn case and the cn case, respectively:
```

```

    asin1 = asin(w);
    acos1 = acos(w);
%
% Evaluate the elliptic integral of the first kind for the sn case and the cn case,
% respectively:
    invsn = ellipint(asin1,tau1,AC);
    invcn = ellipint(acos1,tau1,AC);
%
% Compute the argument for sn and cn, respectively:
    argsn = invsn*kappa;
    argcn = invcn*kappa;
%
% Compute sn and cn, respectively:
    sn = sin(argsn);
    cn = cos(argcn);
%
% Compute the squares (real parts are used since sn and cn are, in general, complex,
% but it can be shown that the imaginary parts, ideally zero, are insignificantly
% small):
    snr = real(sn);
    cnr = real(cn);
    snsq = snr.*snr;
    cnsq = cnr.*cnr;
%
% Adjust values, prior to taking the logarithm, to prevent problems with the
% logarithm of zero (or near zero):
    N1 = 501;
    for ii=1:N1
        if(snsq(ii) > 10^30)
            snsq(ii) = 10^30;
        end
        if(cnsq(ii) > 10^30)
            cnsq(ii) = 10^30;
        end
    end
%
% Form the magnitude-squared frequency response denominators:
    dens = 1 + snsq;
    denc = 1 + cnsq;
%
% Form the magnitude-squared frequency responses:
    HMmags = 1 ./ dens;

```



```
HMmagc = 1 ./ denc;
%
% Form the magnitude-squared frequency responses in dB:
HMdBs = 10*log10(HMmags);
HMdBc = 10*log10(HMmagc);
%
%
% Design a third-order Chebyshev filter:
[zz,pp,kk] = cheblap(3,10*log10(2));
NUM = kk*poly(zz);
DEN = poly(pp);
%
% Form the magnitude-squared frequency response of the third-order Chebyshev
% filter in dB:
freq = freqs(NUM,DEN,w);
frem = abs(freq);
HMdBf = 20*log10(frem);
%
% Plot and compare the three responses:
plot(w,HMdBs,'g',w,HMdBc,'r',w,HMdBf,'b')
grid
xlabel('Radian Frequency')
ylabel('Magnitude in dB')
title('sn form in green, cn form in red, Chebyshev in blue')
pause
close
%
%
% End of Example 6.1
%
```

APPENDIX D

THE MATLAB m-FILE EXAMP6_2.m

```
%  
% File Name: EXAMP6_2.m  
%  
% This MATLAB m-file demonstrates that setting up the parameters of  
% sn(kappa*sn^(-1) (w/wp,wp/ws),epsp*epss)  
% yielding a real result is critical, and that arbitrarily chosen values will yield,  
% in general, a complex result, and are therefore not realizable. When set up  
% properly, the result in this example is a third-order elliptic magnitude-  
% squared response with 3 dB of passband ripple, a stopband minimum atten-  
% uation of 40 dB, and a stopband frequency edge of 1.9789 radians/sec.  
%  
%  
%  
% This is Example 6.2.  
%  
%  
%  
%%%%%%%%%%%%%%  
%  
  
%  
% STEP ONE: Setup Proper Parameters and Compute Results  
%  
% Enter parameters:  
% ws = 1.9789;  
% epss = 0.0100005;  
% kappa = 2.7906;  
% AC = 10000;  
% tau1 = epss;  
% tau2 = 1 / ws;  
%
```

```

% Generate an array of radian frequency values going from 0 to 5 radians/second:
    nn = 0:1:500;
    w = nn/100;
%
% Produce the array of values for the upper limit of the elliptic integral of the first
% kind:
    asin1 = asin(w);
%
% Evaluate the elliptic integral of the first kind:
    invsn = ellipint(asin1,tau2,AC);
%
% Compute the argument for sn:
    argsn = invsn*kappa;
%
% Compute sn:
    sn = ellipsn(argsn,tau1);
%
% Compute the squares of the magnitudes (real and imaginary parts are computed
% also for analysis):
    snr = real(sn);
    sni = imag(sn);
    snm = abs(sn);
    snsq = snm.*snm;
%
% Adjust values, prior to taking the logarithm, to prevent problems with the
% logarithm of zero (or near zero):
    N1 = 501;
    for ii = 1:N1
        if(snsq(ii) > 10^30)
            snsq(ii) = 10^30;
        end
    end
%
% Form the magnitude-squared frequency response denominator:
    den = 1 + snsq;
%
% Form the magnitude-squared frequency response:
    HMmag = 1 ./den;
%
% Form the magnitude-squared frequency response in dB:
    HMdB = 10*log10(HMmag);
%

```

```

%
%
% STEP TWO: Setup Arbitrary Parameters 1 and Compute Results
%
% Enter parameters:
    ws2 = 2.5;
    eps2 = 0.01;
    kappa2 = 3;
    AC2 = 1000;
    tau1 = eps2;
    tau2 = 1 / ws2;
%
% Evaluate the elliptic integral of the first kind:
    invsn2 = ellipint(asin1,tau2,AC2);
%
% Compute the argument for sn:
    argsn2 = invsn2*kappa2;
%
% Compute sn:
    sn2 = ellipn(argsn2,tau1);
%
% Compute the squares(real and imaginary parts are computed also for analysis):
    snr2 = real(sn2);
    sni2 = imag(sn2);
    sns2 = sn2.*sn2;
%
% Form the squared frequency response denominator:
    den2 = 1 + sns2;
%
% Form the magnitude-squared frequency response:
    HMmag2 = abs(1 ./ den2);
%
% Adjust values, prior to taking the logarithm, to prevent problems with the
% logarithm of zero (or near zero):
    N1 = 501;
    for ii = 1:N1
        if(HMmag2(ii) < 10^(-30))
            HMmag2(ii) = 10^(-30);
        end
    end
% Form the magnitude-squared frequency response in dB:
    HMdB2 = 10*log10(HMmag2);

```

```

%
%
%
% STEP THREE: Setup Arbitrary Parameters 2 and Compute Results
%
% Enter parameters:
%   ws3 = 1.5;
%   epss3 = 0.008;
%   kappa3 = 3;
%   tau1 = epss3;
%   tau2 = 1 / ws3;
%
% Evaluate the elliptic integral of the first kind:
%   invsn3 = ellipint(asin1,tau2,AC2);
%
% Compute the argument for sn:
%   argsn3 = invsn3*kappa3;
%
% Compute sn:
%   sn3 = ellipfn(argsn3,tau1);
%
% Compute the squares(real and imaginary parts are computed also for analysis):
%   snr3 = real(sn3);
%   sni3 = imag(sn3);
%   sns3 = sn3.*sn3;
%
% Form the squared frequency response denominator:
%   den3 = 1 + sns3;
%
% Form the magnitude-squared frequency response:
%   HMmag3 = abs(1 ./ den3);
%
% Adjust values, prior to taking the logarithm, to prevent problems with the
% logarithm of zero (or near zero):
%   N1 = 501;
%   for ii=1:N1
%       if(HMmag3(ii) < 10^(-30))
%           HMmag3(ii) = 10^(-30);
%       end
%   end
%
% Form the magnitude-squared frequency response in dB:
%   HMdB3 = 10*log10(HMmag3);

```

```

%
%
%
% STEP FOUR: Design a Third-Order Elliptic Filter:
%
    [zz,pp,kk] = ellipap(3,10*log10(2),40);
    NUM = kk*poly(zz);
    DEN = poly(pp);
%
% Form the magnitude-squared frequency response of the third-order elliptic filter
% in dB:
    freq = freqs(NUM,DEN,w);
    frem = abs(freq);
    HMdBf = 20*log10(frem);
%
% Plot and compare the four magnitude responses:
    plot(w,HMdB,'g',w,HMdB2,'r',w,HMdB3,'y',w,HMdBf,'b')
    grid
    xlabel('Radian Frequency')
    ylabel('Magnitude in dB')
    title('proper sn in green, arbitrary 1 in red, arbitrary 2 in yellow, elliptic in
          blue')
    pause
    close
%
% Scale snr and sni for plotting purposes:
    snr40 = snr/40;
    sni40 = sni/40;
%
% Plot and compare the real and imaginary responses:
    plot(w,snr40,'m',w,sni40,'c',w,snr2,'g',w,sni2,'r',w,snr3,'y',w,sni3,'b')
    grid
    xlabel('Radian Frequency')
    ylabel('Amplitude')
    title('Prop/40: magenta (r), cyan (i). Arb1: green (r), red (i). Arb2: yellow (r),
          blue (i)')
    pause
    close
%
%
% End of Example 6.2
%
```

APPENDIX E

THE MATLAB m-FILE EXAMP6_6.m

```
%
% File Name: EXAMP6_6.m
%
% This MATLAB m-file computes Filter Selectivity and
% Shaping Factor for an elliptic lowpass filter for the
% following parameters:
%   N = 3, 4, ..., 10
%   a = 6 dB
%   b = 60 dB
%   Ap = 1 dB
%   As = 80 dB
%   wc = 1
%
%
%
% This algorithm computes the data for Example 6.6.
%
%
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
%
%
%
% Enter Parameters:
%   a = 6;
%   b = 60;
%   Ap = 1;
%   As = 80;
%   epsp = sqrt(10^(Ap/10) - 1);
%   epss = 1/sqrt(10^(As/10) - 1);
%   tau1 = epsp*epss;
```

```

    mod1 = tau1^2;
    [X1,E] = ellipke(mod1);
%
% The main processing loop:
    for N=3:10
        % Initially assume a normalized wp:
        ws = ellipws(N,Ap,As,1);
        wc = ellipwc(N,Ap,As,1,ws);
        tau2 = 1/ws;
        [X2,E] = ellipke(tau2^2);
        q = 0;
        if(2*round(N/2) == N)
            q = 1;
        end
        % Scale for wc = 1:
        wp = 1/wc;
        ws = ws/wc;
        wc = 1;
        % Compute Filter Selectivity:
        FS(N-2) = ellipfs(N,Ap,As,wp,wc,ws);
        % Compute the Shaping Factor:
        SF(N-2) = ellipsf(N,a,b,Ap,As,wp,ws);
    end
%
%
% End of Example 6.6
%
```


REFERENCES

- Aiello, G.L., and P.M. Angelo (1974). "Transitional Legendre-Thomson Filters," *IEEE Trans. Cir. Sys.*, vol. CAS-21, no. 1, pp. 159-162.
- Alexander, S.T. (1986). *Adaptive Signal Processing: Theory and Applications*, Springer-Verlag, New York.
- Ambaradar, A. (1995). *Analog and Digital Signal Processing*, PWS Publishing Co., Boston.
- Ariga, M., and S. Masamitsu (1970). "An Extremum Approach to Constant-Delay Transfer Functions Providing Large Amplitude Bandwidth," *IEEE Trans. Cir. Theory*, vol. CT-17, no. 1, pp. 121-125.
- Attikiouzel, J. and D.T. Phuc (1978). "On Transitional Ultraspherical-Ultraspherical Filters," *Proceed. IEEE*, vol. 66, no. 6 pp. 703-706.
- Bennett, B.J. (1988). "A New Filter Synthesis Technique - The Hourglass," *IEEE Trans Cir Sys.*, vol. 35, no. 12, pp. 1469-1477
- Bigelow, S.J. (1994). *Understanding Telephone Electronics*, 3rd ed., Sams, A Division of Prentice-Hall, Indianapolis, IN.
- Blake, G.G. (1974) [1928]. *History of Radio Telegraphy and Telephony*, Arno Press, New York (originally published by Chapman & Hall, London, 1928).
- Bode, H.W. (1945). *Network Analysis and Feedback Amplifier Design*, Van Nostrand, New York.
- Brittain, J.E. (1992). "George A. Campbell and the Electric Wave Filter," (Scanning the Past), *Proceed. IEEE*, vol. 80, no. 5, p. 782.
- Brown, R.G., and P.Y.C. Hwang (1992). *Introduction to Random Signals and Applied Kalman Filtering*, 2nd ed., John Wiley & Sons, New York.

- Brune, O. (1931). "Synthesis of a Finite Two-Terminal Network Whose Driving-Point Impedance is a Prescribed Function of Frequency," *J. Math. Phys.*, vol. 10, pp. 191-236.
- Brune, O. (1932). "Note on Bartlett's Bisection Theorem for 4-terminal Electrical Networks," *Phil. Mag.*, vol. 14, no. 93, pp. 806-811.
- Bunker, W.M. (1970). "Symmetrical Equal-Ripple Delay and Symmetrical Equal-Ripple Phase Filters," *IEEE Trans, Cir. Theory*, vol. CT-17, no. 3, pp. 455-458.
- Burchnall, J.L. (1951). "The Bessel Polynomials," *Can. J. Math.*, vol. 3, pp. 62-68.
- Butterworth, S. (1930). "On the Theory of Filter Amplifiers," *Wireless Engineer*, vol. 7, pp. 536-541.
- Byrd, P.F., and M.D. Friedman (1954). *Handbook of Elliptic Integrals for Engineers and Physicists*, Springer-Verlag, New York.
- Calahan, D.A. (1964). *Modern Network Synthesis, Volume 1: Approximation*, Hayden Book Co., New York.
- Campbell, G.A. (1911). "Cisoidal Oscillations," *Trans. AIEE*, vol. 30, pp. 873-909.
- Campbell, G.A. (1922). "Direct Capacity Measurement," *Bell System Tech. J.*, vol. 1, no. 1, pp. 18-38.
- Candy, J.V. (1986). *Signal Processing: The Model-Based Approach*, McGraw-Hill New York.
- Carslaw, H.S. (1950). *An Introduction to the Theory of Fourier's Series and Integrals*, 3rd rev. ed., Dover, New York.
- Cauer, W. (1931). *Siebshaltungen*, V.D.I. Verlag, G.m.b.H., Berlin.
- Cauer, W. (1939). "Ausgangsseitig leerlaufende Filter," *ENT*, vol. 16, no. 6, pp. 161-163.

References

- Cauer, W. (1958). *Synthesis of Linear Communication Networks* (translated from the German by G.E. Knausenberger and J.N. Warfield), McGraw-Hill, New York.
- Chebyshev, P.L. (1899). "Theorie des Mecanismes Connus sous le Nom de Parallelogrammes," *Oeuvres*, vol. 1, St. Petersburg.
- Chen, W.H. (1964). *Linear Network Design and Synthesis*, McGraw-Hill, New York.
- Chui, C.K., and G. Chen (1991). *Kalman Filtering, with Real-Time Applications*, 2nd ed., Springer-Verlag, New York.
- Cunningham, E.P. (1992). *Digital Filtering: An Introduction*, Houghton Mifflin, Boston.
- Darlington, S. (1939). "Synthesis of Reactance 4-poles Which Produce Prescribed Insertion Loss Characteristics," *J. Math. Phys.*, vol. 18, pp. 257-353.
- Darlington, S. (1984). "A History of Network Synthesis and Filter Theory for Circuits Composed of Resistors, Inductors, and Capacitors," *IEEE Trans. Cir. Sys.*, vol. CAS-31, no. 1, pp. 3-13.
- Davenport, W.B., Jr., and W.L. Root (1987) [1958]. *An Introduction to the Theory of Random Signals and Noise*, IEEE Press, New York.
- Dishal, M. (1959). "Gaussian-Response Filter Design," *Electrical Communication*, vol. 36, no. 1, pp. 3-26.
- Fettweis, A. (1995). "Fifty Years Since Wilhelm Cauer's Death," *IEEE Trans. Cir. Sys. I*, vol. 42, no. 4, pp. 193-194.
- Gardner, W.A. (1986). *Introduction to Random Processes, With Applications to Signals and Systems*, Macmillan, New York.
- Gradshteyn, I.S., and I.M. Ryzhik (1980). *Table of Integrals, Series, and Products*, Academic Press, San Diego, CA.
- Grosswald, E. (1951). "On Some Algebraic Properties of the Bessel Polynomials," *Trans. Am. Math. Soc.*, pp. 197-210.

- Guillemin, E.A. (1957). *Synthesis of Passive Networks: Theory and Methods Appropriate to the Realization and Approximation Problems*, John Wiley & Sons, New York.
- Halpern, P.H. (1969). "Optimum Monotonic Low-Pass Filters," *IEEE Trans. Cir. Theory*, vol. CT-16, no. 2, pp. 240-242.
- Halpern, P.H. (1976). "Monotonic Step Response Filters with Maximum Asymptotic Cutoff," *IEEE Trans. Cir. Sys.*, vol. CAS-23, no. 6, pp. 380-383.
- Hamming, R.W. (1989). *Digital Filters*, 3rd ed., Prentice-Hall, Englewood Cliffs, NJ.
- Hansell, G.E. (1969). *Filter Design and Evaluation*, Van Nostrand Reinhold, New York.
- Haykin, S. (1989). *Modern Filters*, Macmillan, New York.
- Haykin, S. (1991). *Adaptive Filter Theory*, 2nd ed., Prentice-Hall, Englewood Cliffs, NJ.
- Henderson, K.W., and W.H. Kautz (1958). "Transient Responses of Conventional Filters," *IRE Trans. Cir. Theory*, vol. CT-5, no. 4, pp. 333-347.
- Honig, M.L., and D.G. Messerschmitt (1984). *Adaptive Filters: Structures, Algorithms, and Applications*, Kluwer Academic, Boston.
- Huelsman, L.P. (1993). *Active and Passive Analog Filter Design: An Introduction*, McGraw-Hill, New York.
- Johnson, D.E., and J.R. Johnson (1966). "Low-Pass Filters using Ultraspherical Polynomials," *IEEE Trans. Cir. Theory*, vol. CT-13, no. 4, pp. 364-369.
- Krall, H.H., and O. Frink (1949). "A New Class of Orthogonal Polynomials: The Bessel Polynomials," *Trans. Am. Math. Soc.*, vol. 65, pp. 100-115.
- Ku, Y.H., and M. Drubin (1962). "Network Synthesis using Legendre and Hermite Polynomials," *J. Franklin Inst.*, vol. 273, no. 2, pp. 138-157.
- Lam, H.Y-F. (1979). *Analog and Digital Filters: Design and Realization*, Prentice-Hall, Englewood Cliffs, NJ.

References

- Lathi, B.P. (1992). *Linear Systems and Signals*, Berkeley-Cambridge Press, Carmichael, CA.
- Lawden, D.F. (1989). *Elliptic Functions and Applications*, Springer-Verlag, New York.
- Lindquist, C.S. (1977). *Active Network Design with Signal Filtering Applications*, Steward & Sons, Long Beach, CA.
- Loy, N.J. (1988). *An Engineer's Guide to FIR Digital Filters*, Prentice-Hall, Englewood Cliffs, NJ.
- Lutovac, M.D. and D.M. Rabrenovic (1992). "A Simplified Design of Some Cauer Filters without Jacobian Elliptic Functions," *IEEE Trans. Cir. Sys. II*, vol. 39, no. 9, pp. 666-671.
- Macnee, A.B. (1963). "Chebyshev Approximation of a Constant Group Delay," *IEEE Trans. Cir. Theory*, vol. CT-10, no. 2, pp. 284-285.
- Norton, E.L. (1937). "Constant Resistance Networks with Applications to Filter Groups," *Bell Sys. Tech. J.*, vol. 16, pp. 178-193.
- Oppenheim, A.V., and R.W. Schaffer (1975). *Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ.
- Oppenheim, A.V., and R.W. Schaffer (1989). *Discrete-Time Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ.
- Oppenheim, A.V., A.S. Willsky, with I.T. Young (1983). *Signals and Systems*, Prentice-Hall, Englewood Cliffs, NJ.
- Paley, R.E.A.C., and N. Wiener (1933). "Notes on the Theory and Application of Fourier Transforms. I-II," *Trans. Amer. Math. Soc.*, vol. 35, pp. 348-355.
- Paley, R.E.A.C., and N. Wiener (1934). *Fourier Transforms in the Complex Domain*, American Mathematical Society Colloquium Publications, vol. 19, American Mathematical Society, New York.
- Papoulis, A. (1958). "Optimum Filters with Monotonic Response," *Proceed. IRE*, vol. 46, no. 3, pp. 606-609.

- Papoulis, A. (1959). "On Monotonic Response Filters," *Proceed. IRE*, vol. 47, pp. 332, 333.
- Papoulis, A. (1962). *The Fourier Integral and its Applications*, McGraw-Hill, New York.
- Papoulis, A. (1977). *Signal Analysis*, McGraw-Hill, New York.
- Parks, T.W., and C.S. Burrus (1987). *Digital Filter Design*, John Wiley & Sons, New York.
- Peless, Y., and T. Murakami (1957). "Analysis and Synthesis of Transitional Butterworth-Thomson Filters and Bandpass Amplifiers," *RCA Review*, vol. 18, no. 1, pp. 60-94.
- Poularikas, A.D., editor (1996). *The Transforms and Applications Handbook*, CRC Press, Boca Raton, FL.
- Proakis, J.G., and D.G. Manolakis (1988). *Introduction to Digital Signal Processing*, Macmillan, New York.
- Proakis, J.G., and M. Salehi (1994). *Communication Systems Engineering*, Prentice-Hall, Englewood Cliffs, NJ.
- Rabrenovic, D.M. and M.D. Lutovac (1992). "A Quasi-Elliptic Function," *IEEE Trans. Cir. Sys. I*, vol. 39, no. 3, pp. 247-249.
- Scanlan, J.O. (1965). "Transfer Functions with Elliptic Distribution of Poles at Equal Frequency Spacings," *IEEE Trans. Cir. Theory*, vol. CT-12, no. 2, pp. 260-266.
- Selby, S.M., editor (1970). *CRC Standard Mathematical Tables*, The Chemical Rubber Co.
- Sheahan, D.F., and R.A. Johnson, editors (1977). *Modern Crystal & Mechanical Filters*, IEEE Press, New York.
- Siebert, W.M. (1986). *Circuits, Signals, and Systems*, The MIT Press, Cambridge, MA.
- Stephenson, F.W. (1985). *RC Active Filter Design Handbook*, John Wiley & Sons, New York.

References

- Storch, L. (1954). "Synthesis of Constant-Time Delay Ladder Networks Using Bessel Polynomials," *Proceed. IRE*, vol. 42, pp. 1666-1675.
- Storer, J.E. (1957). *Passive Network Synthesis*, McGraw-Hill, New York.
- Stremler, F.G. (1990). *Introduction to Communication Systems*, 3rd ed., Addison-Wesley, New York.
- Terrell, T.J. (1988). *Introduction to Digital Filters*, 2nd ed., John Wiley & Sons, New York.
- Therrien, C.W. (1992). *Discrete Random Signals and Statistical Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ.
- Thomson, W.E. (1949). "Delay Networks Having Maximally Flat Frequency Characteristics," *Proceed. IEE*, part 3, vol. 96, pp.487-490.
- Thomson, W.E. (1959). "Maximally-Flat Delay Networks," *IRE Trans. Cir. Theory*, vol. CT-6, no. 2, p. 235.
- Van Valkenburg, M.E. (1960). *Introduction to Modern Network Synthesis*, John Wiley & Sons, New York.
- Van Valkenburg, M.E. (1974). *Network Analysis*, 3rd ed., Prentice-Hall, Englewood Cliffs, NJ.
- Van Valkenburg, M.E. (1982). *Analog Filter Design*, Holt, Rinehart and Winston, Fort Worth, TX.
- Weinberg, L. (1962). *Network Analysis and Synthesis*, McGraw-Hill, New York.
- Widrow, B., and S.D. Stearns (1985). *Adaptive Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ.
- Williams, C.S. (1986). *Designing Digital Filters*, Prentice-Hall, Englewood Cliffs, NJ.
- Young, I.T., and L.J. van Vliet (1995). "Recursive Implementation of the Gaussian Filter," *Signal Process.*, vol. 44, no. 2, pp. 139-151.

INDEX

A

- Active filters, 359-390
 - advantages and disadvantages
 - compared to passive filters, 359
 - Delyiannis second-order all-pass circuit, 375, 376
 - examples of active filter
 - implementation, 378-384
 - first-order active filter stages, 360-363
 - Friend bandpass circuit, 373, 374
 - highpass notch circuit, 370-373
 - lowpass notch circuit, 366-368
 - Sallen and Key highpass circuit, 368-370
 - Sallen and Key lowpass circuit, 364-366
 - second-order active filter stages, 364-376
 - summary of active filter
 - implementation, 377
- Adaptive filters, 1, 6
- Aiello and Angelo, 233
- Alexander, 6
- All-pass transfer functions, 78-80, 259
- Ambardar, 42
- Amplitude modulation (AM), 2, 54, 84-86
- Analog filter design theorem, 58-66, 107, 108, 110, 111, 246, 247
 - Bessel filters, 215
 - Butterworth filters, 115
 - Chebyshev Type I filters, 132, 139, 140, 151
 - Chebyshev Type II filters, 156, 163, 174
 - elliptic filters, 177, 178, 200

- examples, 58, 59, 62-66
- summary, 61

- Analog filters, 1, 7
- Analytic signal, 216
- Ariga and Masamitsu, 16
- Attenuation function, 93
- Attikiouzel and Phuc, 16
- Autocorrelation, 60, 107, 266

B

- Bennett, 16
- Bessel, 15, 215
- Bessel filters, 7, 15, 16, 215-232, 233-237, 238, 266
 - Bessel filter design formula, 220
 - Bessel polynomials, 221
 - comparison with other filters, 228-230
 - determination of order, 224, 225, 231
 - filter selectivity, 223, 224, 231
 - magnitude frequency response, 223, 224
 - phase response, phase delay, and
 - group delay, 225-228, 232
 - pole locations, 225
 - shaping factor, 223, 224, 231
 - time-domain response, 228, 229, 232
- Bigelow, 9
- Black, 15
- Blake, 15
- Bode, 15, 67
- Bode plots, 67-70, 115
- Brittain, 15
- Brown and Hwang, 7
- Brune, 15
- Bunker, 16

- Burchnell, 15, 215
- Butterworth, 15, 113
- Butterworth filters, 7, 15, 113-130, 233-237, 242, 246, 252, 254, 257, 260-265, 266, 268, 269
- definition, 115
 - determination of order, 117-119
 - filter selectivity, 115
 - magnitude frequency response, 115, 116
 - phase response, phase delay, and group delay, 122-124
 - pole locations, 119-122
 - Butterworth polynomials, 121, 122
 - shaping factor, 116, 117
 - time-domain response, 125, 126
- Byrd and Friedman, 182
- C**
- Calahan, 182
- Campbell, 15
- Candy, 7
- Cauchy integral formula, 35, 36
- Cauer, 15, 177
- Cauer filters, 7, 15, 177
- Chebyshev rational functions, 198-200
- Chebyshev Type I filters, 7, 15, 131-153, 238, 242, 246, 252, 253, 257, 267, 268, 269
- Chebyshev polynomials, 138-140
 - comparison with Butterworth filters, 148, 149
 - cutoff frequency, 3 dB, 135
 - definition, 132
 - determination of order, 137
 - equiripple, 134
 - filter selectivity, 136
 - magnitude frequency response, 132, 135
 - passband rippling, 132-134
 - phase response, phase delay, and group delay, 143-146
 - pole locations, 140-143
 - response peaks and valleys, 132-134
 - ripple factor, 132, 134
 - shaping factor, 136
 - time-domain response, 146, 147
- Chebyshev Type II filters, 7, 15, 155-176
- 268, 269
 - comparison with Butterworth filters, 159, 170, 171, 173, 174
 - comparison with Chebyshev Type I filters, 170-174
 - cutoff frequency, 158
 - definition, 156
 - determination of order, 161
 - equiripple stopband, 158
 - filter selectivity, 160, 161, 171, 173
 - impulse response, 169-172, 175, 176
 - inverse Chebyshev polynomials, 162, 163
 - magnitude frequency response, 156, 159
 - phase response, phase delay, and group delay, 166-169
 - poles and zeros, 163-165
 - ripple factor, 156, 157
 - shaping factor, 160, 161, 171, 173
 - step response, 169-172, 175, 176
 - stopband peaks and valleys, 157, 158
 - stopband rippling, 156-158
- Chen, 54
- Chui and Chen, 7
- Constant time delay, 215, 230
- Convolution, 3, 27, 59, 77, 81, 107
- Crossover network, 13, 14
- Cunningham, 7
- D**
- Darlington, 15
- Davenport and Root, 6
- Delyiannis second-order all-pass circuit, 375, 376
- Dishal, 16, 236
- Digital filters, 7
- E**
- Elliptic filters, 7, 15, 16, 177-214
- Chebyshev rational functions, 198-200
 - comparison with other filters, 206-208

cutoff frequency, 3 dB, 193
 definition, 177, 178
 determination of order, 197, 198
 filter selectivity, 196
 magnitude frequency response, 194, 195
 minimum attenuation frequencies, 194
 passband rippling, 189
 passband peaks and valleys, 192
 phase response, phase delay, and group delay, 202-205
 poles and zeros, 200-202
 shaping factor, 196, 197
 stopband edge frequency, 195
 stopband rippling, 189
 time-domain response, 205, 206
 transmission zeros, 193
 Elliptic integral, complete, 178, 183
 Elliptic integral, inverse, 183
 Elliptic integrals, 182-188
 Envelope delay, 85

F

Fettweis, 15
 Filter selectivity, 52, 53, 106, 107, 115, 136, 160, 171, 196, 215, 223, 224, 231, 247, 251, 268
 Filters
 active, 7
 adaptive, 6, 7
 analog, 1, 7, 8
 anti-aliasing, 6, 11, 104, 152
 applications of, 8-14
 bandpass, 2, 3, 9, 13, 28-32, 103
 bandstop, 29-31, 103
 Bessel, 7, 15, 16, 215-232, 233-237
 brickwall, 25
 Butterworth, 7, 15, 113-130
 Cauer, 7, 15
 Chebyshev Type I, 7, 15, 131-153
 Chebyshev Type II, 7, 15, 155-176
 classes of, 4-8
 definition, 1
 digital, 7
 elliptic, 7, 15, 177-214
 examples, 1-4, 8-14

frequency selective, 5, 6
 Gaussian, 16, 107, 108, 129, 235-241
 Halpern, 255-260
 highpass, 9, 13, 28-30, 103
 historical perspective, 15, 16
 ideal, 25-34
 imposed constraints, 23, 54-57
 inverse Chebyshev, 15, 155
 Kalman, 6
 Legendre, 16, 238-246
 linear phase, 26, 40
 lowpass, 4, 8, 9, 11, 13, 24-28, 94-96, 102, 103, 215
 notch, 9
 Papoulis, 249-256
 passive, 7
 phase-compensated, 259-265
 raised cosine, 32-34, 46, 50, 103, 106
 realizability, 23
 statistical, 6
 transitional, 233-237
 ultraspherical, 245-251
 Wiener, 6
 First-order transfer functions, 66-72
 Fourier transforms, 24, 25, 54, 55
 Frequency band definitions, 51
 Frequency modulation (FM), 11, 13, 20
 Frequency scaling, 98-102
 effect on:
 magnitude response, 100, 101
 phase response, 100, 101
 phase delay, 101
 group delay, 101
 impulse response, 102
 step response, 102
 Frequency transformation circuit operations, 345-348
 Frequency transformations, 271-325
 lowpass-to-lowpass transformation, 271-274
 poles and zeros, 272
 magnitude frequency response and phase response, 272
 determination of minimum order,

- 272, 273
 - filter selectivity, 273
 - shaping factor, 273
 - phase delay and group delay, 273
 - time-domain response, 273, 274
 - lowpass-to-highpass transformation, 274-287
 - poles and zeros, 275, 276
 - magnitude frequency response and phase response, 276, 277
 - determination of minimum order, 277
 - filter selectivity, 277, 278
 - shaping factor, 278
 - phase delay and group delay, 279, 280
 - time-domain response, 280-282
 - detailed example (*Example 9.6*), 282-287
 - lowpass-to-bandpass transformation, 287-303
 - poles and zeros, 288-292
 - magnitude frequency response and phase response, 293, 294
 - determination of minimum order, 294, 295
 - filter selectivity, 295, 296
 - shaping factor, 296
 - phase delay and group delay, 296-298
 - time-domain response, 298
 - detailed example (*Example 9.15*), 298-303
 - lowpass-to-bandstop transformations, 303-319
 - poles and zeros, 304-309
 - magnitude frequency response and phase response, 309-311
 - determination of minimum order, 311
 - filter selectivity, 311, 312
 - shaping factor, 312
 - phase delay and group delay, 313, 314
 - time-domain response, 314
 - detailed example (*Example 9.23*), 314-319
 - Friend bandpass circuit, 373, 374
- G**
- Gardner, 6
 - Gaussian filters, 235-241, 266, 267
 - definition, 235
 - design method, 235-238
 - impulse response, 238, 241
 - group delay, 238, 240
 - magnitude frequency response, 238, 239
 - phase delay, 238, 240
 - phase response, 238, 239
 - step response, 238, 241
 - Grosswald, 15, 215
 - Group delay, 81-88, 110
 - Bessel filters, 225-228, 232
 - Butterworth filters, 123, 124
 - Chebyshev Type I filters, 145, 146
 - Chebyshev Type II filters, 168, 169
 - elliptic filters, 202-205
 - examples, 86-88
 - Gaussian filters, 238, 240
 - Halpern filters, 257, 259
 - illustration, 84-86
 - Legendre filters, 242, 243, 245
 - Papoulis filters, 253, 255
 - phase-compensated filters, 263
 - summary, 88
 - transitional filters, 234, 236
 - ultraspherical filters, 247, 250
 - Guillemin, 54
- H**
- Halpern, 16, 255
 - Halpern filters, 255-260, 268, 269
 - definition, 255, 256
 - design, 256, 257, 258
 - group delay, 257, 259
 - impulse response, 257, 260
 - Jacobi polynomials, 256
 - magnitude frequency response, 257, 258
 - phase delay, 257, 259
 - phase response, 257, 258

- step response, 257, 260
 - Hamming, 7
 - Hansell, 343, 344
 - Haykin, 6, 7
 - Henderson and Kautz, 15
 - Hilbert transform relations, 88-98, 110, 111, 216
 - examples, 91, 92, 94-97
 - summary, 98
 - Honig and Messerschmitt, 6
 - Huelsman, 9
- I**
- Ideal filters, 25-34, 102, 103
 - Ideal transmission, 79, 215, 216
 - Impedance scaling, 348, 349
 - Imposed constraints, 54-57
 - Butterworth filters, 115
 - Chebyshev Type I filters, 132
 - Chebyshev Type II filters, 156, 163
 - elliptic filters, 178, 200
 - summary, 57
 - Impulse response, 25, 29-34, 69, 73, 77, 89, 102, 103, 266
 - Bessel filters, 228
 - Butterworth filters, 125
 - Chebyshev Type I filters, 146, 147
 - Chebyshev Type II filters, 169, 170
 - elliptic filters, 205, 206
 - Gaussian filters, 238, 241
 - Halpern filters, 257, 260
 - Legendre filters, 243, 245
 - Papoulis filters, 254, 255
 - phase-compensated filters, 264
 - transitional filters, 234, 237
 - ultraspherical filters, 248, 250
 - Inverse Chebyshev polynomials, 162, 163
 - Inverse Chebyshev response, 155
- J**
- Jacobian elliptic cosine function, 179, 183
 - Jacobian elliptic cosine, inverse, 183
 - Jacobian elliptic functions, 182-188
 - periodicity, 184
 - Jacobian elliptic sine function, 178, 183
 - inverse, 183, 187
 - modulus, 178, 183
- Johnson and Johnson, 16, 245
- K**
- Kalman filters, 6
 - Krall and Frink, 15, 215, 221
 - Ku and Drubin, 16, 238
- L**
- Lam, 92, 95
 - Laplace transforms, 24, 25, 107
 - Lawden, 179, 182, 184, 185, 186
 - Legendre filters, 16, 238-246, 267, 268
 - definition, 239
 - design, 244
 - impulse response, 243, 245
 - group delay, 242, 243, 245
 - Legendre polynomials, 239, 240, 250, 251, 267
 - m-th associated Legendre filters, 239, 240
 - magnitude frequency response, 242, 243
 - modified m-th associated, 242
 - phase delay, 242, 244
 - phase response, 242, 244
 - step response, 243, 246
 - Lindquist, 9, 52, 53, 67, 94, 114, 233, 236, 245, 249
 - Linear phase, 79, 83, 95, 216
 - Loss function, 93
 - Loy, 7
 - Lutovac and Rabrenovic, 16
- M**
- Maclaurin series, 114, 221, 236
 - Macnee, 16
 - Marconi, 15
 - MATLAB, 1, 16, 20, 103-111, Appendix A
 - Maximally-flat group delay, 217-223
 - Minimum phase transfer functions, 77, 78
 - Modern filters, 1, 6
- N**
- Noise

- additive, 3, 4
 - example, 9
 - white, 4, 60, 107, 266
- Norton, 15
- O**
- Oppenheim and Schaffer, 6, 7, 11
- Oppenheim, Willsky and Young, 104
- Overshoot, 27, 103
- P**
- Paley and Wiener, 34, 35
- Paley-Wiener theorem, 34-40, 104, 105, 267
- example, 39
 - summary, 40
- Papoulis, 16, 35, 36, 38, 90, 92, 94, 95, 249
- Papoulis filters, 249-256, 255, 268, 269
- definition, 249, 250
 - design, 251, 252, 254
 - impulse response, 254, 255
 - group delay, 253, 254
 - magnitude frequency response, 252, 253
 - phase delay, 253, 255
 - phase response, 253, 254
 - step response, 254, 256
- Parks and Burrus, 7
- Passband definition, 51
- Passive filters, 329-357
- Bessel filter implementation table, 342
 - Butterworth filter implementation table, 336
 - Chebyshev Type I filter implementation tables, 337, 339-341
 - continued-fraction ladder implementation, 331-335
 - elliptic filter implementation tables, 343, 344
 - examples of passive filter implementation, 350-355
 - frequency-transformation circuit operations, 345-348
 - impedance scaling, 348, 349
 - summary of passive filter design, 329, 330, 349, 350
- Peless and Murakami, 233
- Phase-compensated filters, 259-265, 269
- group delay, 263
 - impulse response, 264
 - magnitude frequency response, 261, 262
 - phase delay, 264
 - phase response, 261, 262
 - step response, 265
- Phase delay, 81-88, 110
- Bessel filters, 225, 227, 232
 - Butterworth filters, 123, 124
 - Chebyshev Type I filters, 145, 146
 - Chebyshev Type II filters, 168
 - elliptic filters, 202-204
 - examples, 86-88
 - Gaussian filters, 238, 240
 - Halpern filters, 257, 259
 - illustration, 84-86
 - Legendre filters, 242, 244
 - Papoulis filters, 253, 254
 - phase-compensated filters, 264
 - summary, 88
 - transitional filters, 234, 236
 - ultraspherical filters, 247, 249
- Phase function, 93
- Power spectral density, 4, 5, 60, 266
- Proakis and Manolakis, 7
- Proakis and Salehi, 216
- Prototype filter, 271
- Q**
- Quadrantal symmetry, 58, 120
- R**
- Rabrenovic and Lutovac, 16
- Rational transfer function, 55
- Region of convergence, 58
- Rise time, 27
- Root locus, 108
- S**
- Sallen and Key highpass circuit, 368-370

Sallen and Key lowpass circuit, 364-366
 Scanlan, 16
 Schwartz inequality, 48
 Second-order transfer functions, 70-76
 Selby, 240, 246
 Settling time, 27
 Shaping factor, 52, 53, 106, 107, 116,
 117, 136, 160, 171, 196, 197, 215,
 223, 224, 231, 238, 242, 247, 252,
 255, 256, 257, 269
 Sheahan and Johnson, 7, 330
 Siebert, 41, 67, 115
 Single side-band modulation, 216
 Signal-to-noise ratio, 4, 17
 Sinc function, 25
 Statistical filters, 1, 6
 Step response, 27, 28, 69, 73, 75, 77, 81,
 102, 103, 108, 109
 Bessel filters, 228, 229
 Butterworth filters, 125, 126
 Chebyshev Type I filters, 146, 147
 Chebyshev Type II filters, 169, 170
 elliptic filters, 205, 206
 Gaussian filters, 238, 241
 Halpern filters, 257, 260
 Legendre filters, 243, 246
 Papoulis filters, 254, 256
 phase-compensated filters, 265
 transitional filters, 234, 237
 ultraspherical filters, 248, 251
 Stephenson, 15, 343, 344
 Stereo-multiplexed signals, 11-13, 20
 Stopband definition, 51
 Storch, 15, 215, 221
 Storer, 15
 Stremler, 2, 13, 32, 54, 60, 84, 90, 216

T

Telephone tone decoder, 9
 Terrell, 7
 Therrien, 6
 Thomson, 15, 215
 Time-bandwidth products, 40-51, 105,
 106
 examples, 42-44, 46, 47, 49, 50
 summary, 50, 51

Transition band definition, 51
 Transitional filters, 233-237, 266
 design method, 234
 group delay, 234, 236
 impulse response, 234, 237
 magnitude frequency response, 234,
 235
 phase delay, 234, 236
 phase response, 234, 235
 step response, 234, 237

U

Ultraspherical filters, 245-251, 268
 Butterworth, special case of, 247,
 267
 Chebyshev Type I, special case of,
 247, 267
 definition of, 246
 design, 247, 248
 Gegenbauer polynomials, 245
 group delay, 247, 250
 impulse response, 248, 250
 Jacobi polynomials, 245
 Legendre, special case of, 247, 268
 magnitude frequency response, 247,
 248
 phase delay, 247, 249
 phase response, 247, 249
 step response, 248, 251
 ultraspherical polynomials, 245

V

Van Valkenburg, 15, 67, 221, 331, 364

W

Wagner, 15
 Weinberg, 15, 54, 331, 336, 337, 339-342
 Widrow and Stearns, 6, 7
 Wiener, 35
 Wiener filters, 6
 Williams, 7

Y

Young and van Vliet, 16

ABOUT THE ACCOMPANYING DISK

MATLAB® Examples

A valuable relationship between analog filter theory and analysis and modern digital signal processing is made by the application of MATLAB to both the design and analysis of analog filters. Throughout the book, computer-oriented problems are assigned. The disk that accompanies this book contains MATLAB functions and m-files written specifically for this book. The MATLAB functions on the disk extend basic MATLAB capabilities in terms of the design and analysis of analog filters. The m-files are used in a number of examples in the book. They are included on the disk as an instructional aid.

The diskette is distributed by Kluwer Academic Publishers with absolutely no support and no warranty from Kluwer Academic Publishers. Use or reproduction of the information on the disk for commercial gain is strictly prohibited. Kluwer Academic Publishers shall not be liable for damage in connection with, or arising out of, the furnishing, performance or use of the disk.

Copyright©2001 Kluwer Academic Publishers
All Rights Reserved

MATLAB® is a registered trademark of The Math Works, Inc.