analog and digital FILTER DESIGN using C

LES THEDE

ANALOG AND DIGITAL FILTER DESIGN USING C

Les Thede

Department of Electrical Engineering
Ohio Northern University

Prentice Hall PTR
Upper Saddle River, New Jersey 07458

Library of Congress Cataloging-in-Publication Data

Thede, Les (Leslie D.)
Analog and digital filter design using C / Les Thede.
p. cm.
Includes bibliographical references and index.
ISBN 0-13-352627-5
1. Electrical filters—Design and construction. 2. C (Computer program language) I. Title
TK7872.F5T53 1996
621.3815'324—dc20 95-24511
CIP

Editorial/production supervision: *Kerry Reardon*
Cover design director: *Jerry Votta*
Cover design: *Talar Agasyan*
Manufacturing buyer: *Alexis R. Heydt*
Acquisitions editor: *Karen Gettman*

Prentice-Hall, Inc.
A Simon & Schuster Company
Upper Saddle River, New Jersey 07458

The publisher offers discounts on this book when ordered in bulk quantities. For more information, contact:

Corporate Sales Department
Prentice Hall PTR
One Lake Street
Upper Saddle River, NJ 07458

Phone: 800-382-3419
Fax: 201-236-7141
E-mail: corpsales@prenhall.com

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN 0-13-352627-5

Prentice-Hall International (UK) Limited, *London*
Prentice-Hall of Australia Pty. Limited, *Sydney*
Prentice-Hall Canada Inc., *Toronto*
Prentice-Hall Hispanoamericana, S.A., *Mexico*
Prentice-Hall of India Private Limited, *New Delhi*
Prentice-Hall of Japan, Inc., *Tokyo*
Simon & Schuster Asia Pte. Ltd., *Singapore*
Editora Prentice-Hall do Brasil, Ltda., *Rio de Janeiro*

This text is dedicated to
LeRoy L. Thede,
the first of three generations
of electrical engineers.
Thanks, Dad!

Preface

This book was intentionally written to be different from other filter design books in two important ways. First, the most common analog and digital filter design and implementation methods are ***covered in a no-nonsense manner***. All important derivations and descriptions are provided to allow the reader to apply them directly to his or her own filter design problem. Not only are the details of analog active and digital IIR and FIR filter design presented in an organized and direct manner, but implementation issues are discussed to alert the reader to potential pitfalls. The simulation of analog filters is made easier by the generation of PSpice circuit description files which includes R-C component values calculated directly from the filter coefficients. In addition, the testing of IIR and FIR filters designed for audio signals is enhanced by providing sample WAV and VOC sound files which can be filtered by using the digital filter design coefficients. Anyone with a sound card on their computer can then play the original and processed sound files for immediate evaluation.

The second difference between this book and others is that the important filter design and implementation techniques discussed in this text are ***supported by thoroughly tested C code***. No, the source code provided on the accompanying disk is not just a collection of fragmented functions, but rather a set of three organized programs to design and implement analog and digital filters. Not only are DOS executable programs provided on the disk, but also all of the source code is provided to allow additions or modifications to be made as desired. Virtually all of the included source code (except the graphics functions for frequency response display) is portable. Three graphics modules have been provided to support Microsoft, Borland, and MIX Software compilers on DOS platforms. However, readers using other platforms can still take advantage of the code in this text in three ways. They can forgo the frequency response display and still retain all of the design and implementation features of the software. Or they can link a graphics module for their system with the existing code for a complete system (requirements are presented in the text). And as a last option they can save the magnitude and phase responses to disk files (a standard feature) and display the data using other programs if desired. A basic knowledge of C programming is expected of the reader, but the code presented in the text is thoroughly discussed and well-documented. The text does assume the reader is familiar with the fundamental concepts of linear systems such as system transfer functions and frequency response although no prior knowledge of filter design is needed.

The construction of this text is unique in that the filter design and implementation techniques are developed by following the evolution of the C code for the three programs in the book. By following the development of

FILTER, the filter design program, throughout the text, the reader is introduced to the fundamental steps in analog and digital filter design. In addition, methods of frequency response determination and display are presented. The filter design problem is presented in ever-increasing detail until practically each entry in the project outline becomes a C function. The two filter implementation programs are developed in a similar manner. ANALOG aids in the implementation of analog active filters by determining electronic component values from the coefficients generated by FILTER and by generating PSpice circuit description files. DIGITAL illustrates the implementation of digital filters by allowing the user to digitally filter sound files using the coefficients generated by FILTER. The reader can then play the original and filtered sound files to hear the effects of the filter.

CHAPTER CONTENTS

Chapter 1 introduces the reader to the filter design problem. An overview of the FILTER design project is presented with additional details of the project developed as they are needed in succeeding chapters. Chapter 2 develops the normalized transfer functions for the Butterworth, Chebyshev, inverse Chebyshev, and elliptic approximation cases. The FILTER project outline is enhanced to include the necessary C functions to determine the order and coefficients for these approximations. Chapter 3 describes the conversion of the normalized lowpass filter to an unnormalized lowpass, highpass, bandpass, or bandstop filter. By the end of the third chapter, a complete analog filter design can be performed.

Chapter 4 introduces the reader to the calculation of the frequency response of the analog filters designed in the previous chapters. In addition to the C code which calculates the frequency response, the reader is also introduced to the graphics functions necessary to display the frequency response. Techniques for compiling and linking the source files for the FILTER project are also discussed. In Chapter 5, the implementation of analog filters is considered using popular techniques in active filter design with discussion of real-world considerations. The ANALOG active filter implementation program is developed to determine the RC coefficients necessary to implement active filters. A PSpice circuit description file is generated to enable the filter developer to analyze the circuit. Chapter 5 completes the discussion of analog filters in this book.

Chapter 6 begins the discussion of discrete-time systems and digital filter design in this book. Several key features of discrete-time systems, including the notion of analog-to-digital conversion, Nyquist sampling theorem, the z-transform, and discrete-time system diagrams, are reviewed.

Similarities and differences between discrete-time and continuous-time systems are discussed. In Chapter 7, digital IIR (recursive) filters are designed. Three methods of designing IIR filters are considered with C code developed for the predominant bilinear transformation method. In addition, the frequency response calculations and related C code for the IIR filter are developed. Chapter 8 considers digital FIR (nonrecursive) filters using a variety of window methods and the Parks-McClellan optimization routine. The special techniques necessary for FIR frequency response calculation are discussed before developing the C code for the FIR filter design portion of the FILTER project. The implementation of real-time and nonreal-time digital FIR and IIR filters is discussed in Chapter 9. Implementation issues such as which type of digital filter to use, accuracy of quantized samples, fixed or floating point processing, and finite register length computation are discussed. Popular sound file formats are introduced and the C code necessary to process these sound files is generated. Users may then use the DIGITAL program to process sound files using the filter coefficients determined by FILTER. The reader can then hear the effects of filtering by replaying the original and processed sound files on a sound card.

ACKNOWLEDGMENTS

I would not have been able to complete this book without the help and support of a number of people. First, I thank the reviewers of this text who provided many helpful comments, both in the initial and final stages of development. These include Malcolm Slaney, Randy Crane, Paul Embree, John O'Donnell, Dave Retterer, and Dave Bogner.

I also thank the staff at Prentice Hall who have provided me with help and guidance throughout the publication process. These include Senior Editor Karen Gettman and her administrative assistant Barbara Alfieri.

I thank the faculty of the Department of Electrical Engineering at Ohio Northern University for their encouragement and support throughout this hectic and time-consuming process.

And, finally, I thank my wife Diane for all of her encouragement and support, and for the many hours of proofreading a text that made no sense to her!

TRADEMARKS

Table of Contents

Chapter 1

Introduction to Filters and Filter Design Software

Everyone has probably come in contact with one type of filter or another in their lifetime. Maybe it was a coffee filter used to separate the grounds from the liquid, or perhaps an oil filter to remove contaminants from the oil of an engine. Anyone working in an office often filters the unimportant work from the important. In essence then the act of filtering is the act of separating desired items from undesired items. Of course when we discuss filters in this text, we are not talking about coffee, oil or paperwork, but rather electronic signals. The electronic filters we will be designing will separate the desirable signal frequencies from the undesirable.

There are many types of electronic filters and many ways that they can be classified. A filter's frequency selectivity is probably the most common method of classification. A filter can have a lowpass, highpass, bandpass, or bandstop response, where each name indicates how a band of frequencies is affected. For example, a lowpass filter would pass low frequencies with little attenuation (reduction in amplitude), while high frequencies would be significantly reduced. A bandstop filter would severely attenuate a middle band of frequencies while passing frequencies above and below the attenuated frequencies. Filter selectivity will be the focus of the first section in this chapter.

Filters can also be described by the method used to approximate the ideal filter. Some approximation methods emphasize low distortion in the passband of the filter while others stress the ability of the filter to attenuate the signals in the stopband. Each approximation method has visible characteristics which distinguish it from the others. Most notably, the absence or presence of ripple (variations) in the passband and stopband clearly set one approximation method apart from another. Filter approximation methods will be discussed in further detail in the second section.

Another means of classifying filters is by the implementation method used. Some filters will be built to filter analog signals using individual components mounted on circuit boards, while other filters might simply be part of a larger digital system which has other functions as well. Several implementation methods will be described in the third section of this chapter as well as the differences between analog and digital signals. However, it should be noted that digital filter implementation will be considered in detail starting in Chapter 6, while the first five chapters concentrate on filter approximation theory and analog filter implementation.

In the final sections of this chapter we will begin discussing the development of a C program to design analog and digital filters. In these sections, we will develop an approach to be used throughout the remainder of this text. A spiral technique will be employed to allow us to study the framework of the project in ever greater detail. This framework will not only include the primary functions to be employed within the program, but also the definition of the primary data variables used. By the end of this chapter we will have the framework of the main filter design program and the ability to enter and store the key filter specifications. We will even test FILTER, the filter design software included with this text.

1.1 Filter Selectivity

As indicated earlier, a filter's primary purpose is to differentiate between different bands of frequencies, and therefore frequency selectivity is the most common method of classifying filters. Names such as lowpass, highpass, bandpass, and bandstop are used to categorize filters, but it takes more than a name to completely describe a filter. In most cases a precise set of specifications is required in order to allow the proper design of a filter. There are two primary sets of specifications necessary to completely define a filter's response, and each of these can be provided in different ways.

The frequency specifications used to describe the passband(s) and stopband(s) could be provided in Hertz (Hz) or in radians/second (rad/sec). We will use the frequency variable f measured in Hertz as filter input and output specifications because it is a slightly more common way of discussing frequency. However, the frequency variable ω measured in radians/second will also be used as the internal variable of choice as well as for unnormalized frequency responses since most calculations will use radians/second.

The other major filter specifications are the gain characteristics of the passband(s) and stopband(s) of the filter response. A filter's gain is simply the ratio of the output signal level to the input signal level. If the filter's gain is greater than 1, then the output signal is larger than the input signal, while if the gain is less than 1, the output is smaller than the input. In most filter applications, the gain response in the stopband is very small. For this reason, the gain is typically converted to decibels (dB) as indicated in Equation 1.1. For example, a filter's passband gain response could be specified as 0.707 or –3.0103 dB, while the stopband gain might be specified as 0.0001 or –80.0 dB.

$$\text{gain}_{\text{dB}} = 20 \cdot \log(\text{gain}) \tag{1.1}$$

As we can see, the values in dB are more manageable for very small gains. Some filter designers prefer to use attenuation (or loss) values instead of gain values. Attenuation is simply the inverse of gain. For example, a filter with a gain of 1/2 at a particular frequency would have an attenuation of 2 at that frequency. If we express attenuation in dB we will find that it is simply the negative of the gain in dB as indicated in Equation 1.2. Gain values expressed in dB will be the standard quantities used as filter specifications, although the term attenuation (or loss) will be used occasionally when appropriate.

$$\text{attn}_{\text{dB}} = 20 \cdot \log(\text{gain}^{-1}) = -20 \cdot \log(\text{gain}) = -\text{gain}_{\text{dB}} \tag{1.2}$$

1.1.1 Lowpass Filters

Figure 1.1 shows a typical lowpass filter's response using frequency and gain specifications which are necessary for precision filter design. The frequency range of the filter specification has been divided into three areas. The passband extends from zero frequency (DC) to the passband edge frequency f_{pass}, and the stopband extends from the stopband edge frequency f_{stop} to infinity. (We will see later in this text, that digital filters have a finite upper frequency limit. We will discuss that issue at the appropriate time.) These two bands are separated by the transition band which extends from

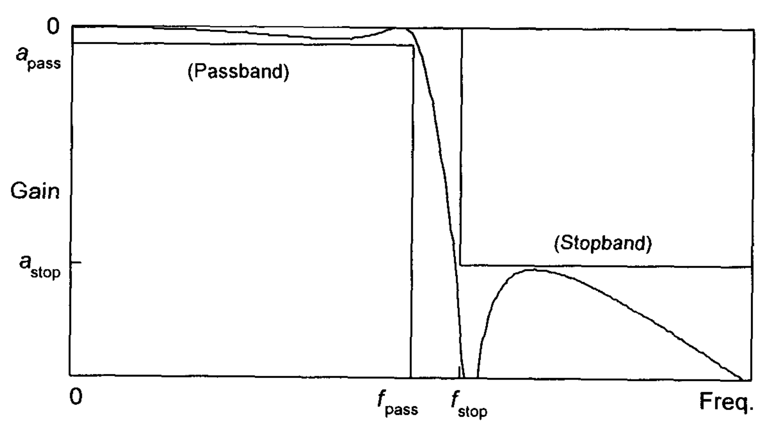

Figure 1.1 Lowpass filter specification.

f_{pass} to f_{stop}. The filter response within the passband is allowed to vary between 0 dB and the passband gain a_{pass}, while the gain in the stopband can vary between the stopband gain a_{stop} and negative infinity. (The 0 dB gain in the passband relates to a gain of 1.0, while the gain of negative infinity in the stopband relates to a gain of 0.0.) A lowpass filter's selectivity can now be specified with only four parameters: the passband gain a_{pass}, the stopband gain a_{stop}, the passband edge frequency f_{pass} and the stopband edge frequency f_{stop}.

Lowpass filters are used whenever it is important to limit the high frequency content of a signal. For example, if an old audio tape has a lot of high-frequency "hiss", a lowpass filter with a passband edge frequency of 8 kHz could be used to eliminate much of the hiss. Of course, it also eliminates high frequencies which were intended to be reproduced. We should remember that any filter can differentiate only between bands of frequencies, not between information and noise.

1.1.2 Highpass Filters

A highpass filter can be specified as shown in Figure 1.2. Note that in this case the passband extends from f_{pass} to infinity (for analog filters) and is located at a higher frequency than the stopband which extends from zero to f_{stop}. The transition band still separates the passband and stopband. The passband gain is still specified as a_{pass} and the stopband gain is still specified as a_{stop}.

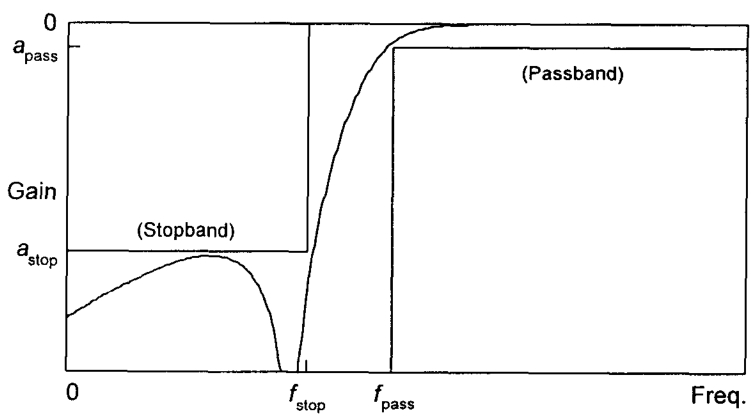

Figure 1.2 Highpass filter specification.

Highpass filters find a use when it is important to eliminate low frequencies from a signal. For example, when turntables are used to play LP records (some readers may remember those black vinyl disks which would warp in a car's back window), turntable rumble can sometimes occur, producing distracting low-frequency signals. A highpass filter set to a passband edge frequency of 100 Hz could help to eliminate this distracting signal.

1.1.3 Bandpass Filters

The filter specification for a bandpass filter shown in Figure 1.3 requires a bit more description. A bandpass filter will pass a band of frequencies while attenuating frequencies which are above or below that band. In this case the passband exists between the lower passband edge frequency f_{pass1} and the upper passband edge frequency f_{pass2}. A bandpass filter has two stopbands. The lower stopband extends from zero to f_{stop1}, while the upper stopband extends from f_{stop2} to infinity (for analog filters). Within the passband, there is a single passband gain parameter a_{pass}. However, individual parameters for the lower stopband gain a_{stop1} and the upper stopband gain a_{stop2} could be used if necessary.

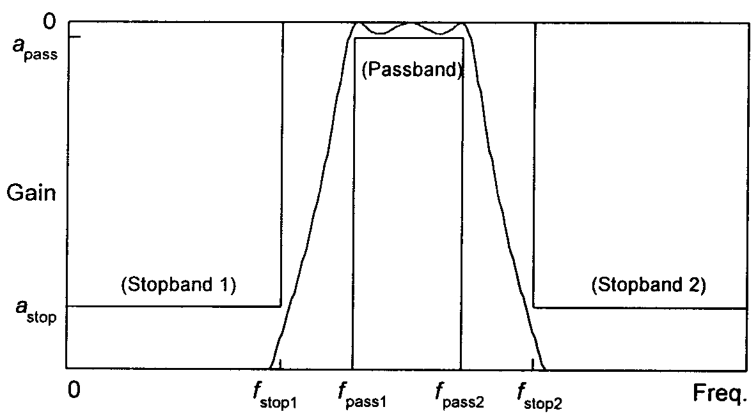

Figure 1.3 Bandpass filter specification.

A good example for the application of a bandpass filter is the processing of voice signals. The normal human voice has a frequency content located primarily in the range of 300 – 3000 Hz. Therefore, the frequency response for any system designed to pass primarily voice signals should contain the input signal to that frequency range. In this case, f_{pass1} would be 300 Hz and f_{pass2} would be 3000 Hz. The stopband edge frequencies would be selected by how fast we would want the signal response to roll off above and below the passband.

1.1.4 Bandstop Filters

The final type of filter to be discussed in this section is the bandstop filter as shown in Figure 1.4. In this case the band of frequencies being rejected is located between the two passbands. The stopband exists between the lower stopband edge frequency f_{stop1} and the upper stopband edge frequency f_{stop2}. The bandstop filter has two passbands. The lower passband extends from zero to f_{pass1}, while the upper passband extends from f_{pass2} to infinity (for analog filters). Within the stopband, the single stopband gain parameter a_{stop} is used. However, individual gain parameters for the lower and upper passbands, a_{pass1} and a_{pass2}, respectively, could be used if necessary.

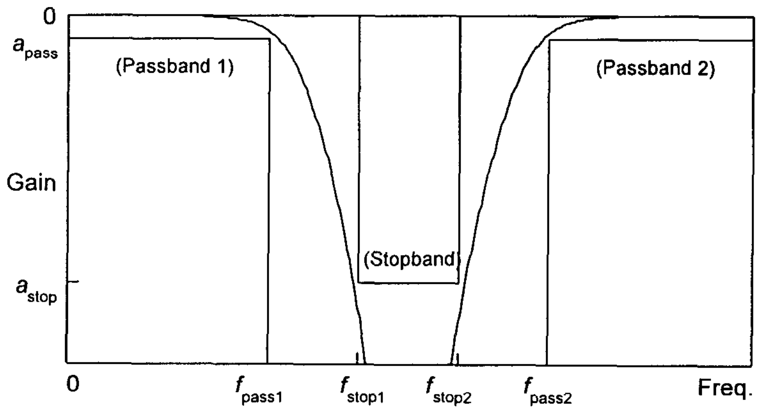

Figure 1.4 Bandstop filter specification.

An excellent example of a bandstop application would be a 60 Hz notch filter used in sensitive measurement equipment. Most electronic measurement equipment today runs from an AC power source using a 60 Hz input frequency. However, it is not uncommon for some of the 60 Hz signal to make its way into the sensitive measurement areas of the equipment. In order to eliminate this troublesome frequency, a bandstop filter (sometimes called a notch filter in these applications) could be used with f_{stop1} set to 58 Hz and f_{stop2} set to 62 Hz. The passband edge frequencies could be adjusted based on the other technical requirements of the filter.

1.2 FILTER APPROXIMATION

The response of an ideal lowpass filter is shown in Figure 1.5, where all frequencies from 0 to f_o are passed with a gain of 1, and all frequencies above

f_o are completely attenuated (gain = 0). This type of filter response is physically unattainable. Practical filter responses which can be attained are also shown. As a filter's response becomes closer and closer to the ideal, the cost of the filter (time delay, number of elements, dollars, etc.) will increase. These practical responses are referred to as approximations to the ideal. There are a variety of ways to approximate an ideal response based on different criteria. For example, some designs may emphasize the need for minimum distortion of the signals in the passband and would be willing to trade-off stopband attenuation for that feature. Other designs may need the fastest transition from passband to stopband and will allow more distortion in the passband to accomplish that aim. It is this engineering trade-off which makes the design of filters so interesting.

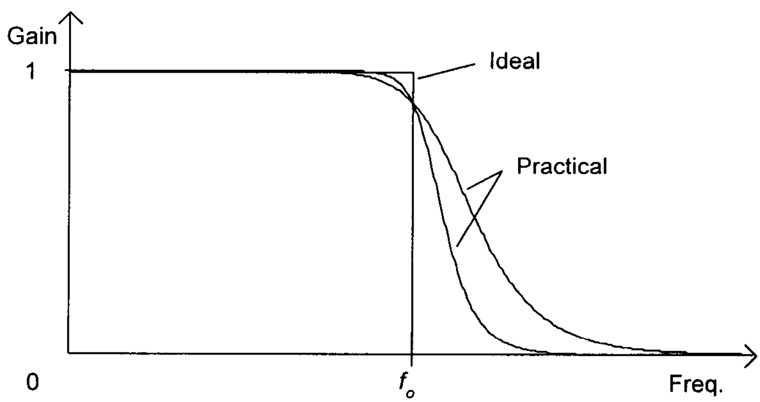

Figure 1.5 Practical and ideal filter responses.

We will be discussing the primary approximation functions used in filter design today which can be classified both by name and the presence of ripple or variation in the signal bands. Elliptic or Cauer filter approximations provide the fastest transition between passband and stopband of any which are studied in this text. An illustration of the magnitude response of an elliptic filter is shown in Figure 1.1, where we can see that ripple exists in both the passband and stopband. What is not shown in that figure is the phase distortion that the elliptic filter generates. If the filter is to be used with audio signals, this phase distortion must usually be corrected. However, in other applications, for example the transmission of data, the elliptic filter is a popular choice because of its excellent selectivity characteristics. The