Chapter

2

Filter Fundamentals

Digital filters are often based upon common analog filter functions. Therefore, a certain amount of background material concerning analog filters is a necessary foundation for the study of digital filters. This chapter reviews the essentials of analog system theory and filter characterization. Some common analog filter types—Butterworth, Chebyshev, elliptical, and Bessel—are given more detailed treatment in subsequent chapters.

1.2 Systems

Within the context of signal processing, a *system* is something that accepts one or more input signals and operates upon them to produce one or more output signals. Filters, amplifiers, and digitizers are some of the systems used in various signal processing applications. When signals are represented as mathematical functions, it is convenient to represent systems as *operators* that operate upon input functions to produce output functions. Two alternative notations for representing a system H with input x and output y are given in Eqs. (2.1) and (2.2). Note that x and y can each be scalar valued or vector valued.

$$y = H[x] \tag{2.1}$$

$$y = H x \tag{2.2}$$

This book uses the notation of Eq. (2.1) as this is less likely to be confused with multiplication of x by a value H.

A system H can be represented pictorially in a flow diagram as shown in Fig. 2.1. For vector-valued x and y, the individual components are sometimes explicitly shown as in Fig. 2.2*a* or lumped together as shown in Fig. 2.2*b*. Sometimes, in order to emphasize their vector nature, the input and output are drawn as in Fig. 2.2*c*.

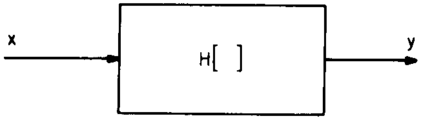


Figure 2.1 Pictorial representation of a system.

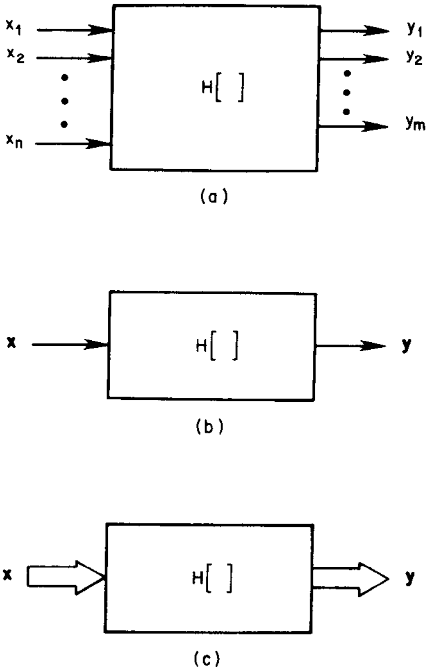


Figure 2.2 Pictorial representation of a system with multiple inputs and outputs.

In different presentations of system theory, the notational schemes used exhibit some variation. The more precise treatments (such as Chen 1984) use x or $x(\cdot)$ to denote a function of time defined over the interval $(-\infty, \infty)$. A function defined over a more restricted interval such as $[t_0, t_1)$ would be denoted as $x_{(t_0,t_1)}$. The notation $x(t)$ is reserved for denoting the value of x at time t. Less precise treatments (such as Schwartz and Friedland 1965) use $x(t)$ to denote both functions of time defined over $(-\infty, \infty)$ and the value of x at time t. When not evident from context, words of explanation must be included to indicate which particular meaning is intended. Using the less precise notational scheme, (2.1) could be rewritten as

$$y(t) = H[x(t)] \tag{2.3}$$

While it appears that the precise notation should be the more desirable, the relaxed conventions exemplified by (2.3) are widespread in the literature.

Linearity

If the relaxed system H is *homogeneous*, multiplying the input by a constant gain is equivalent to multiplying the output by the same constant gain, and

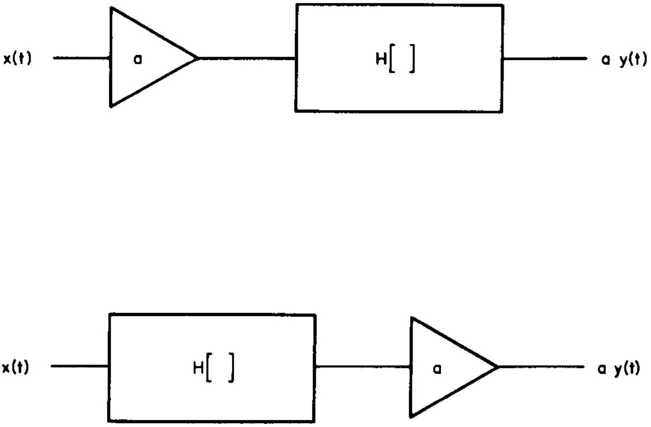


Figure 2.3 Homogeneous system.

the two configurations shown in Fig. 2.3 are equivalent. Mathematically stated, the relaxed system H is homogeneous if, for constant a,

$$H[ax] = a\,H[x] \tag{2.4}$$

If the relaxed system H is additive, the output produced for the sum of two input signals is equal to the sum of the outputs produced for each input individually, and the two configurations shown in Fig. 2.4 are equivalent. Mathematically stated, the relaxed system H is additive if

$$H[x_1 + x_2] = H[x_1] + H[x_2] \tag{2.5}$$

A system that is both homogeneous and additive is said to "exhibit *superposition*" or to "satisfy the principle of superposition." A system that exhibits superposition is called a *linear system.* Under certain restrictions, additivity implies homogeneity. Specifically, the fact that a system H is additive implies that

$$H[\alpha x] = \alpha\,H[x] \tag{2.6}$$

for any rational α. Any real number can be approximated with arbitrary precision by a rational number; therefore, additivity implies homogeneity for real a provided that

$$\lim_{\alpha \to a} H[\alpha x] = H[ax] \tag{2.7}$$

Time Invariance

The characteristics of a *time-invariant system* do not change over time. A system is said to be *relaxed* if it is not still responding to any previously

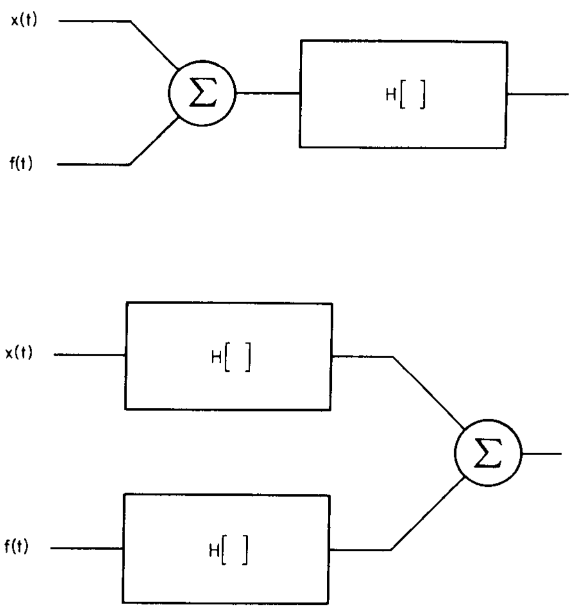


Figure 2.4 Additive system.

applied input. Given a relaxed system H such that

$$y(t) = H[x(t)] \tag{2.8}$$

then H is time invariant if and only if

$$y(t-\tau) = H[x(t-\tau)] \tag{2.9}$$

for any τ and any $x(t)$. A time-invariant system is also called a *fixed system* or *stationary system*. A system that is not time invariant is called a *time-varying system*, *variable system*, or *nonstationary system*.

Causality

In a *causal system*, the output at time t can depend only upon the input at times t and prior. Mathematically stated, a system H is causal if and only if

$$H[x_1(t)] = H[x_2(t)] \qquad \text{for } t \le t_0 \tag{2.10}$$

given that

$$x_1(t) = x_2(t) \qquad \text{for } t \le t_0$$

A *noncausal* or *anticipatory system* is one in which the present output depends upon future values of the input. Noncausal systems occur in theory, but they cannot exist in the real world. This is unfortunate, since we will often discover that some especially desirable frequency responses can be obtained only from noncausal systems. However, causal realizations can be created for noncausal systems in which the present output depends at most upon past, present, and a finite extent of future inputs. In such cases, a causal realization is obtained by simply delaying the output of the system for a finite interval until all the required inputs have entered the system and are available for determination of the output.

2.2 Characterization of Linear Systems

A linear system can be characterized by a differential equation, step response, impulse response, complex-frequency-domain system function, or a transfer function. The relationships among these various characterizations are given in Table 2.1.

Impulse response

The *impulse response* of a system is the output response produced when a unit impulse $\delta(t)$ is applied to the input of a previously relaxed system. This is an especially convenient characterization of a linear system, since the response

TABLE 2.1 Relationships among Characterizations of Linear Systems

Starting with	Perform	To obtain
Time domain differential equation relating $x(t)$ and $y(t)$	Laplace transform	Complex-frequency-domain system function
	Compute $y(t)$ for $x(t)$ = unit impulse	Impulse response $h(t)$
	Compute $y(t)$ for $x(t)$ = unit step	Step response $a(t)$
Step response $a(t)$	Differentiate with respect to time	Impulse response $h(t)$
Impulse response $h(t)$	Integrate with respect to time	Step response $a(t)$
	Laplace transform	Transfer function $H(s)$
Complex-frequency-domain system function	Solve for $H(s) = \frac{Y(s)}{X(s)}$	Transfer function $H(s)$
Transfer function $H(s)$	Inverse Laplace transform	Impulse response $h(t)$

$y(t)$ to any continuous-time input signal $x(t)$ is given by

$$y(t) = \int_{-\infty}^{\infty} x(\tau)\, h(t, \tau)\, d\tau \tag{2.11}$$

where $h(t, \tau)$ denotes the system's response at time t to an impulse applied at time τ. The integral in (2.11) is sometimes referred to as the *superposition integral.* The particular notation used indicates that, in general, the system is time varying. For a time-invariant system, the impulse response at time t depends only upon the time delay from τ to t; and we can redefine the impulse response to be a function of a single variable and denote it as $h(t - \tau)$. Equation (2.11) then becomes

$$y(t) = \int_{-\infty}^{\infty} x(\tau)\, h(t - \tau)\, d\tau \tag{2.12}$$

Via the simple change of variables $\lambda = t - \tau$, Eq. (2.12) can be rewritten as

$$y(t) = \int_{-\infty}^{\infty} x(t - \lambda)\, h(\lambda)\, d\lambda \tag{2.13}$$

If we assume that the input is zero for $t < 0$, the lower limit of integration can be changed to zero; and if we further assume that the system is causal, the upper limit of integration can be changed to t, thus yielding

$$y(t) = \int_{0}^{t} x(\tau)\, h(t - \tau)\, d\tau = \int_{0}^{t} x(t - \lambda)\, h(\lambda)\, d\lambda \tag{2.14}$$

The integrals in (2.14) are known as *convolution integrals*, and the equation indicates that "$y(t)$ equals the *convolution* of $x(t)$ and $h(t)$." It is often more compact and convenient to denote this relationship as

$$y(t) = x(t) \otimes h(t) = h(t) \otimes x(t) \tag{2.15}$$

Various texts use different symbols, such as stars or asterisks, in place of $\otimes$ to indicate convolution. The asterisk is probably favored by most printers, but in some contexts its usage to indicate convolution could be confused with the complex conjugation operator. A typical system's impulse response is sketched in Fig. 2.5.

Step response

The *step response* of a system is the output signal produced when a unit step $u(t)$ is applied to the input of the previously relaxed system. Since the unit step is simply the time integration of a unit impulse, it can easily be shown that the step response of a system can be obtained by integrating the impulse response. A typical system's step response is shown in Fig. 2.6.

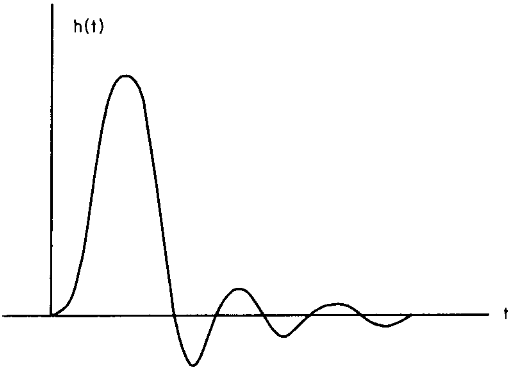


Figure 2.5 Impulse response of a typical system.

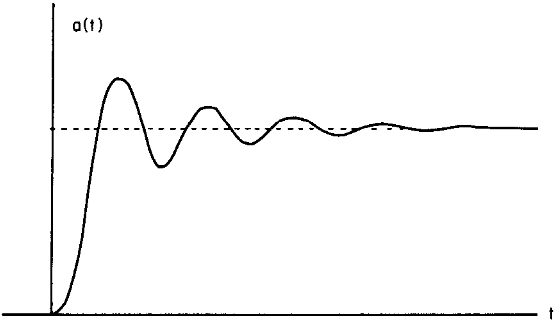


Figure 2.6 Step response of a typical system.

2.3 Laplace Transform

The *Laplace transform* is a technique that is useful for transforming differential equations into algebraic equations that can be more easily manipulated to obtain desired results.

In most communications applications, the functions of interest will usually (but not always) be functions of time. The Laplace transform of a time function $x(t)$ is usually denoted as $X(s)$ or $\mathscr{L}[x(t)]$ and is defined by

$$X(s) = \mathscr{L}[x(t)] = \int_{-\infty}^{\infty} x(t)\, e^{-st}\, dt \tag{2.16}$$

The complex variable s is usually referred to as *complex frequency* and is of the form $\sigma + j\omega$, where σ and ω are real variables sometimes referred to as *neper frequency* and *radian frequency*, respectively. The Laplace transform for a given function $x(t)$ is obtained by simply evaluating the given integral. Some mathematics texts (such as Spiegel 1965) denote the time function with an uppercase letter and the frequency function with a lowercase letter.

However, the use of lowercase for time functions is almost universal within the engineering literature.

If we transform both sides of a differential equation in t using the definition (2.16), we obtain an algebraic equation in s that can be solved for the desired quantity. The solved algebraic equation can then be transformed back into the time domain by using the inverse Laplace transform.

The inverse Laplace transform is defined by

$$x(t) = \mathscr{L}^{-1}[X(s)] = \frac{1}{2\pi j} \int_C X(s)\, e^{st}\, ds \tag{2.17}$$

where C is a contour of integration chosen so as to include all singularities of $X(s)$. The inverse Laplace transform for a given function $X(s)$ can be obtained by evaluating the given integral. However, this integration is often a major chore—when tractable, it will usually involve application of the residue theorem from the theory of complex variables. Fortunately, in most cases of practical interest, direct evaluation of (2.16) and (2.17) can be avoided by using some well-known transform pairs, as listed in Table 2.2, along with a number of transform properties presented in Sec. 2.4.

TABLE 2.2 Laplace Transform Pairs

Ref. no.	$x(t)$	$X(s)$
1	1	$\frac{1}{s}$
2	$u_1(t)$	$\frac{1}{s}$
3	$\delta(t)$	1
4	t	$\frac{1}{s^2}$
5	t^n	$\frac{n!}{s^{n+1}}$
6	$\sin \omega t$	$\frac{\omega}{s^2+\omega^2}$
7	$\cos \omega t$	$\frac{s}{s^2+\omega^2}$
8	e^{-at}	$\frac{1}{s+a}$
9	$e^{-at} \sin \omega t$	$\frac{\omega}{(s+a)^2+\omega^2}$
10	$e^{-at} \cos \omega t$	$\frac{s+a}{(s+a)^2+\omega^2}$

Example 2.1 Find the Laplace transform of $x(t) = e^{-\alpha t}$.

solution

$$X(s) = \int_0^\infty e^{-\alpha t}\, e^{-st}\, dt \tag{2.18}$$

$$= \int_0^\infty e^{-(\alpha + s)t}\, dt \tag{2.19}$$

$$= \frac{1}{s + \alpha} \tag{2.20}$$

Notice that this result agrees with entry 8 in Table 2.2.

Background

The Laplace transform defined by Eq. (2.16) is more precisely referred to as the *one-sided Laplace transform*, and it is the form generally used for the analysis of causal systems and signals. There is also a *two-sided transform* that is defined as

$$\mathscr{L}_{\text{II}}[x(t)] = \int_{-\infty}^{\infty} x(t)\, e^{-st}\, dt \tag{2.21}$$

The Laplace transform is named for the French mathematician Pierre Simon de Laplace (1749–1827).

2.4 Properties of the Laplace Transform

Some properties of the Laplace transform are listed in Table 2.3. These properties can be used in conjunction with the transform pairs presented in Table 2.2, to obtain most of the Laplace transforms that will ever be needed in practical engineering situations. Some of the entries in the table require further explanation, which is provided below.

Time shifting

Consider the function $f(t)$ shown in Fig. 2.7*a*. The function has nonzero values for $t < 0$, but since the one-sided Laplace transform integrates only over positive time, these values for $t < 0$ have no impact on the evaluation of the transform. If we now shift $f(t)$ to the right by τ units as shown in Fig. 2.7*b*, some of the nonzero values from the left of the origin will be moved to the right of the origin, where they will be included in the evaluation of the transform. The Laplace transform's properties with regard to a time-shift right must be stated in such a way that these previously unincluded values will not be included in the transform of the shifted function either. This can be easily accomplished through multiplying the shifted function $f(t - \tau)$ by a shifted unit step function $u_1(t - \tau)$ as shown in Fig. 2.7*c*. Thus we have

$$\mathscr{L}[u_1(t - \tau) f(t - \tau)] = e^{-\tau s}\, F(s) \qquad a > 0 \tag{2.22}$$

TABLE 2.3 Properties of the Laplace Transform

Property	Time function	Transform
1. Homogeneity	$a\,f(t)$	$a\,F(s)$
2. Additivity	$f(t) + g(t)$	$F(s) + G(s)$
3. Linearity	$a\,f(t) + b\,g(t)$	$a\,F(s) + b\,G(s)$
4. First derivative	$\frac{d}{dt} f(t)$	$s\,F(s) - f(0)$
5. Second derivative	$\frac{d^2}{dt^2} f(t)$	$s\,F(s) - s\,f(0) - \frac{d}{dt} f(0)$
6. kth derivative	$\frac{d^{(k)}}{dt^k} f(t)$	$s^k\,F(s) = \sum_{n=0}^{k-1} s^{k-1-n} f^{(n)}(0)$
7. Integration	$\int_{-\infty}^{t} f(\tau)\,d\tau$	$\frac{F(s)}{s} + \frac{1}{s}\left(\int_{-\infty}^{t} f(\tau)\,d\tau\right)_{t=0}$
	$\int_{0}^{t} f(\tau)\,d\tau$	$\frac{F(s)}{s}$
8. Frequency shift	$e^{-at} f(t)$	$X(s + a)$
9. Time shift right	$u_1(t - \tau) f(t - \tau)$	$e^{-\tau s} F(s) \qquad a > 0$
10. Time shift left	$f(t + r), f(t) = 0 \qquad \text{for } 0 < t < \tau$	$e^{\tau s} F(s)$
11. Convolution	$y(t) = \int_0^t h(t - \tau)\,x(\tau)\,d\tau$	$Y(s) = H(s)\,X(s)$
12. Multiplication	$f(t)\,g(t)$	$\frac{1}{2\pi j}\int_{c-j\infty}^{c+j\infty} F(s - r)\,G(r)\,dr$ $\sigma_g < c < \sigma - \sigma_f$

Notes: $f^{(k)}(t)$ denotes the kth derivative of $f(t)$. $f^{(0)}(t) = f(t)$.

Consider now the case when $f(t)$ is shifted to the right. Such a shift will move a portion of $f(t)$ from positive time, where it is included in the transform evaluation, into negative time, where it will not be included in the transform evaluation. The Laplace transform's properties with regard to a time shift left must be stated in such a way that all included values from the unshifted function will likewise be included in the transform of the shifted function. This can be accomplished by requiring that the original function be equal to zero for all values of t from zero to τ, if a shift to the left by τ units is to be made. Thus for a shift left by τ units

$$\mathscr{L}[f(t + \tau)] = F(s)\,e^{\tau s} \qquad \text{if } f(t) = 0 \qquad \text{for } 0 < t < \tau \tag{2.23}$$

Multiplication

Consider the product of two time functions $f(t)$ and $g(t)$. The transform of the product will equal the complex convolution of $F(s)$ and $G(s)$ in the frequency

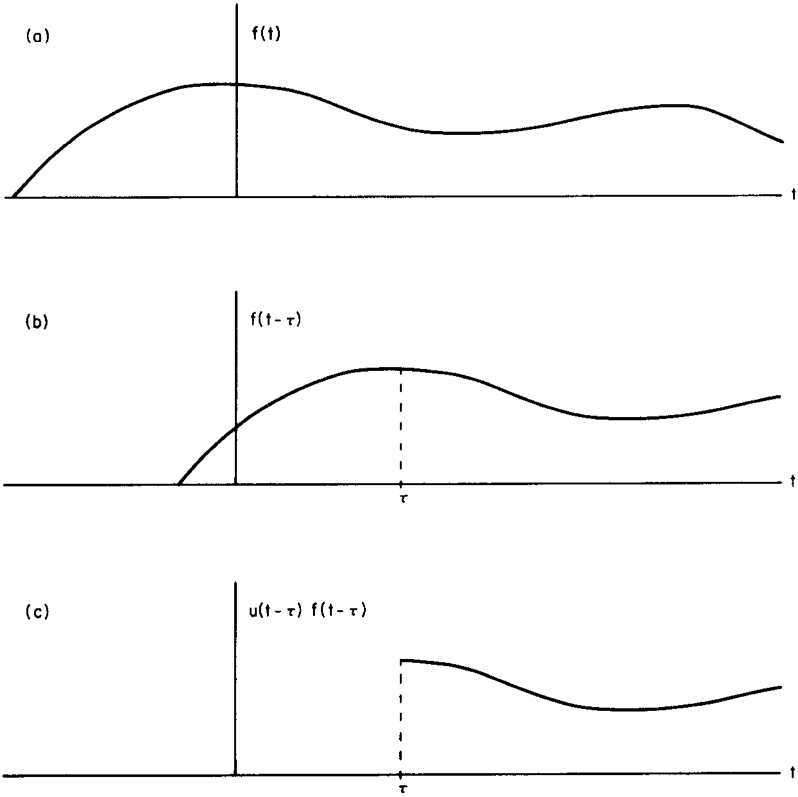


Figure 2.7 Signals for explanation of the Laplace transform's "time-shift-right" property.

domain.

$$\mathscr{L}[f(t)\,g(t)] = \frac{1}{2\pi j}\int_{c-j\infty}^{c+j\infty} F(s-r)\,G(r)\,dr \qquad \sigma_g < c < \sigma - \sigma_f \tag{2.24}$$

2.5 Transfer Functions

The *transfer function* $H(s)$ of a system is equal to the Laplace transform of the output signal divided by the Laplace transform of the input signal:

$$H(s) = \frac{Y(s)}{X(s)} = \frac{\mathscr{L}[y(t)]}{\mathscr{L}[x(t)]} \tag{2.25}$$

It can be shown that the transfer function is also equal to the Laplace transform of the system's impulse response:

$$H(s) = \mathscr{L}[h(t)] \tag{2.26}$$

Therefore,

$$y(t) = \mathscr{L}^{-1}\{H(s)\mathscr{L}[x(t)]\} \tag{2.27}$$

Equation (2.27) presents an alternative to the convolution defined by Eq. (2.14) for obtaining a system's response $y(t)$ to any input $x(t)$, given the impulse response $h(t)$. Simply perform the following steps:

1. Compute $H(s)$ as the Laplace transform of $h(t)$.
2. Compute $X(s)$ as the Laplace transform of $x(t)$.
3. Compute $Y(s)$ as the product of $H(s)$ and $X(s)$.
4. Compute $y(t)$ as the inverse Laplace transform of $Y(s)$. (The Heaviside expansion presented in Sec. 2.6 is a convenient technique for performing the inverse transform operation.)

A transfer function defined as in (2.25) can be put into the form

$$H(s) = \frac{P(s)}{Q(s)} \tag{2.28}$$

where $P(s)$ and $Q(s)$ are polynomials in s. For $H(s)$ to be stable and realizable in the form of a lumped-parameter network, it can be shown (Van Valkenburg 1974) that all of the coefficients in the polynomials $P(s)$ and $Q(s)$ must be real. Furthermore, all of the coefficients in $Q(s)$ must be positive. The polynomial $Q(s)$ must have a nonzero term for each degree of s from the highest to the lowest, unless all even-degree terms or all odd-degree terms are missing. If $H(s)$ is a voltage ratio or current ratio (that is, the input and output are either both voltages or both currents), the maximum degree of s in $P(s)$ cannot exceed the maximum degree of s in $Q(s)$. If $H(s)$ is a transfer impedance (that is, the input is a current and the output is a voltage) or a transfer admittance (that is, the input is a voltage and the output is a current), then the maximum degree of s in $P(s)$ can exceed the maximum degree of s in $Q(s)$ by at most 1. Note that these are only upper limits on the degree of s in $P(s)$; in either case, the maximum degree of s in $P(s)$ may be as small as zero. Also note that these are necessary but not sufficient conditions for $H(s)$ to be a valid transfer function. A candidate $H(s)$ satisfying all of these conditions may still not be realizable as a lumped-parameter network.

Example 2.2 Consider the following alleged transfer functions:

$$H_1(s) = \frac{s^2 - 2s + 1}{s^3 - 3s^2 + 3s + 1} \tag{2.29}$$

$$H_2(s) = \frac{s^4 + 2s^3 + 2s^2 - 3s + 1}{s^3 + 3s^2 + 3s + 2} \tag{2.30}$$

$$H_3(s) = \frac{s^2 - 2s + 1}{s^3 + 3s^2 + 1} \tag{2.31}$$

TABLE 2.4 System Characterizations Obtained from the Transfer Function

Starting with	Perform	To obtain
Transfer function $H(s)$	Compute roots of $H(s)$ denominator	Pole locations
	Compute roots of $H(s)$ numerator	Zero locations
	Compute $\lvert H(j\omega)\rvert$ over all ω	Magnitude response $A(\omega)$
	Compute $\arg[H(j\omega)]$ over all ω	Phase response $\theta(\omega)$
Phase response $\theta(\omega)$	Divide by ω	Phase delay $\tau_p(\omega)$
	Differentiate with respect to ω	Group delay $\tau_g(\omega)$

Equation (2.29) is not acceptable because the coefficient of s^2 in the denominator is negative. If Eq. (2.30) is intended as a voltage- or current-transfer ratio, it is not acceptable because the degree of the numerator exceeds the degree of the denominator. However, if Eq. (2.30) represents a transfer impedance or transfer admittance, it may be valid since the degree of the numerator exceeds the degree of the denominator by just 1. Equation (2.31) is not acceptable because the term for s is missing from the denominator.

A system's transfer function can be manipulated to provide a number of useful characterizations of the system's behavior. These characterizations are listed in Table 2.4 and examined in more detail in subsequent sections.

Some authors, such as Van Valkenburg (1974), use the term "network function" in place of "transfer function."

2.6 Heaviside Expansion

The Heaviside expansion provides a straightforward computational method for obtaining the inverse Laplace transform of certain types of complex-frequency functions. The function to be inverse-transformed must be expressed as a ratio of polynomials in s, where the order of the denominator polynomial exceeds the order of the numerator polynomial. If

$$H(s) = K_0 \frac{P(s)}{Q(s)} \tag{2.32}$$

where $Q(s) = \prod_{k=1}^{n} (s - s_k)^{m_k} = (s - s_1)^{m_1}(s - s_2)^{m_2} \cdots (s - s_n)^{m_n}$ (2.33)

then inverse transformation via the Heaviside expansion yields

$$\mathscr{L}^{-1}[H(s)] = K_0 \sum_{r=1}^{n} \sum_{k=1}^{m_r} [K_{rk} t^{m_r - k} \exp(s_r t)] \tag{2.34}$$

where $K_{rk} = \dfrac{1}{(k-1)!(m_r - k)!} \dfrac{d^{k-1}}{ds^{k-1}} \left[\dfrac{(s - s_r)m_r P(s)}{Q(s)} \right]_{s = s_r}$ (2.35)

A method for computing the derivative in (2.35) can be found in Section 1.4.

Simple pole case

The complexity of the expansion is significantly reduced for the case of $Q(s)$ having no repeated roots. The denominator of (2.32) is then given by

$$Q(s) = \prod_{k=1}^{n} (s - s_k) = (s - s_1)(s - s_2) \cdots (s - s_n) \qquad s_1 \neq s_2 \neq s_3 \neq \cdots s_n \quad (2.36)$$

Inverse transformation via the Heaviside expansion then yields

$$\mathscr{L}^{-1}[H(s)] = K_0 \sum_{r=1}^{n} K_r \, e^{s_r t} \quad (2.37)$$

where $K_r = \left[\dfrac{(s - s_r)P(s)}{Q(s)} \right]_{s = s_r}$ (2.38)

The Heaviside expansion is named for Oliver Heaviside (1850–1925), an English physicist and electrical engineer who was the nephew of Charles Wheatstone (as in Wheatstone bridge).

2.7 Poles and Zeros

As pointed out previously, the transfer function for a realizable linear time-invariant system can always be expressed as a ratio of polynomials in s:

$$H(s) = \frac{P(s)}{Q(s)} \quad (2.39)$$

The numerator and denominator can each be factored to yield

$$H(s) = H_0 \frac{(s - z_1)(s - z_2)(s - z_3) \cdots (s - z_m)}{(s - p_1)(s - p_2)(s - p_3) \cdots (s - p_n)} \quad (2.40)$$

Where the roots $z_1, z_2, \ldots, z_m$ of the numerator are called *zeros* of the transfer function, and the roots $p_1, p_2, \ldots, p_n$ of the denominator are called *poles* of the transfer function. Together, poles and zeros can be collectively referred to as *critical frequencies*. Each factor $(s - z_i)$ is called a *zero factor*, and each factor $(s - p_j)$ is called a *pole factor*. A repeated zero appearing n times is called either an *nth-order zero* or a *zero of multiplicity n*. Likewise, a repeated pole appearing n times is called either an *nth-order pole* or a *pole of multiplicity n*. Nonrepeated poles or zeros are sometimes described as *simple* or *distinct* to emphasize their nonrepeated nature.

Example 2.3 Consider the transfer function given by

$$H(s) = \frac{s^3 + 5s^2 + 8s + 4}{s^3 + 13s^2 + 59s + 87} \quad (2.41)$$

The numerator and denominator can be factored to yield

$$H(s) = \frac{(s+2)^2(s+1)}{(s+5+2j)(s+5-2j)(s+3)} \tag{2.42}$$

Examination of (2.42) reveals that

$s = -1$ is a simple zero

$s = -2$ is a second-order zero

$s = -5 + 2j$ is a simple pole

$s = -5 - 2j$ is a simple pole

$s = -3$ is a simple pole

A system's poles and zeros can be depicted graphically as locations in a complex plane as shown in Fig. 2.8. In mathematics, the complex plane itself is called the *gaussian plane*, while a plot depicting complex values as points in the plane is called an *Argand diagram* or a *Wessel-Argand-Gaussian diagram*. In the 1798 transactions of the Danish academy, Caspar Wessel (1745–1818) published a technique for graphical representation of complex numbers, and Jean Robert Argand published a similar technique in 1806. Geometric interpretation of complex numbers played a central role in the doctoral thesis of Gauss.

Pole locations can provide convenient indications of a system's behavior as indicated in Table 2.5. Furthermore, poles and zeros possess the following properties that can sometimes be used to expedite the analysis of a system:

1. For real $H(s)$, complex or imaginary poles and zeros will each occur in complex conjugate pairs that are symmetric about the σ axis.

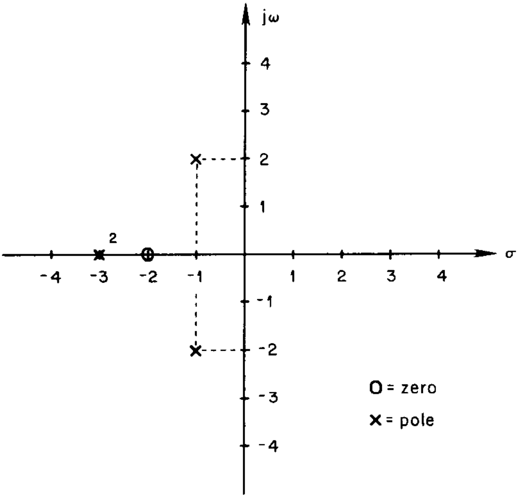


Figure 2.8 Plot of pole and zero locations.

TABLE 2.5 Impact of Pole Locations upon System Behavior

Pole type	Corresponding natural response component	Corresponding description of system behavior
Single real, negative	Decaying exponential	Stable
Single real, positive	Divergent exponential	Divergent instability
Real pair, negative, unequal	Decaying exponential	Overdamped (stable)
Real pair, negative equal	Decaying exponential	Critically damped (stable)
Complex conjugate pair with negative real parts	Exponentially decaying sinusoid	Underdamped (stable)
Complex conjugate pair with zero real parts	Sinusoid	Undamped (marginally stable)
Complex conjugate pair with positive real parts	Exponentially saturating sinusoid	Oscillatory instability

2. For $H(s)$ having even symmetry, the poles and zeros will exhibit symmetry about the $j\omega$ axis.
3. For nonnegative $H(s)$, any zeros on the $j\omega$ axis will occur in pairs.

In many situations, it is necessary to determine the poles of a given transfer function. For some systems, such as Chebyshev filters or Butterworth filters, explicit expressions have been found for evaluation of pole locations. For other systems, such as Bessel filters, the poles must be found by numerically solving for the roots of the transfer function's denominator polynomial. Several root-finding algorithms appear in the literature, but I have found the *Laguerre method* to be the most useful for approximating pole locations. The approximate roots can be subjected to small-step iterative refinement or polishing as needed.

Algorithm 2.1 Laguerre method for approximating one root of a polynomial $P(z)$

Step 1. Set z equal to an initial guess for the value of a root. Typically, z is set to zero so that the smallest root will tend to be found first.

Step 2. Evaluate the polynomial $P(z)$ and its first two derivatives $P'(z)$ and $P''(z)$ at the current value of z.

Step 3. If $P(z)$ evaluates to zero or to within some predefined epsilon of zero, exit with the current value of z as the root. Otherwise, continue on to step 4.

Step 4. Compute a correction term Δz, using

$$\Delta z = \frac{N}{F \pm \sqrt{(N-1)(NG - G^2)}}$$

where $F \triangleq P'(z)/P(z)$, $G \triangleq F^2 - P''(z)/P(z)$, and the sign in the denominator is taken so as to minimize the magnitude of the correction (or, equivalently, so as to maximize the denominator).

Step 5. If the correction term Δz has a magnitude smaller than some specified fraction of the magnitude of z, then take z as the value of the root and terminate the algorithm.

Step 6. If the algorithm has been running for a while (let's say six iterations) and the correction value has gotten bigger since the previous iteration, then take z as the value of the root and terminate the algorithm.

Step 7. If the algorithm was not terminated in step 3, 5, or 6, then subtract Δz from z and go back to step 2.

A C routine **laguerreMethod( )** that implements Algorithm 2.1 is provided in Listing 2.1.

2.8 Magnitude, Phase, and Delay Responses

A system's *steady-state response* $H(j\omega)$ can be determined by evaluating the transfer function $H(s)$ at $s = j\omega$:

$$H(j\omega) = |H(j\omega)|\, e^{j\theta(\omega)} = H(s)\big|_{s=j\omega} \tag{2.43}$$

The *magnitude response* is simply the magnitude of $H(j\omega)$:

$$|H(j\omega)| = (\{\mathrm{Re}[H(j\omega)]\}^2 + \{\mathrm{Im}[H(j\omega)]\}^2)^{1/2} \tag{2.44}$$

It can be shown that

$$|H(j\omega)|^2 = H(s)H(-s)\big|_{s=j\omega} \tag{2.45}$$

If $H(s)$ is available in factored form as given by

$$H(s) = H_0 \frac{(s-z_1)(s-z_2)(s-z_3)\cdots(s-z_m)}{(s-p_1)(s-p_2)(s-p_3)\cdots(s-p_n)} \tag{2.46}$$

then the magnitude response can be obtained by replacing each factor with its absolute value evaluated at $s = j\omega$:

$$|H(j\omega)| = H_0 \frac{|j\omega - z_1| \cdot |j\omega - z_2| \cdot |j\omega - z_3| \cdot \cdots \cdot |j\omega - z_m|}{|j\omega - p_1| \cdot |j\omega - p_2| \cdot |j\omega - p_3| \cdot \cdots \cdot |j\omega - p_n|} \tag{2.47}$$

The *phase response* $\theta(\omega)$ is given by

$$\theta(\omega) = \tan^{-1}\left\{\frac{\mathrm{Im}[H(j\omega)]}{\mathrm{Re}[H(j\omega)]}\right\} \tag{2.48}$$

Phase delay

The *phase delay* $\tau_p(\omega)$ of a system is defined as

$$\tau_p(\omega) = \frac{-\theta(\omega)}{\omega} \tag{2.49}$$

where $\theta(\omega)$ is the phase response defined in Eq. (2.48). When evaluated at any specific frequency ω_1, Eq. (2.49) will yield the time delay experienced by a sinusoid of frequency ω passing through the system. Some authors define $\tau_p(\omega)$ without the minus sign shown on the right-hand side of (2.49). As illustrated in Fig. 2.9, the phase delay at a frequency ω_1 is equal to the negative slope of a secant drawn from the origin to the phase response curve at ω_1.

Group delay

The *group delay* $\tau_g(\omega)$ of a system is defined as

$$\tau_g(\omega) = \frac{-d}{dt}\,\theta(\omega) \tag{2.50}$$

where $\theta(\omega)$ is the phase response defined in (2.48). In the case of a modulated carrier passing through the system, the modulation envelope will be delayed by an amount that is in general not equal to the delay $\tau_p(\omega)$ experienced by the carrier. If the system exhibits constant group delay over the entire bandwidth of the modulated signal, then the envelope will be delayed by an amount equal to τ_g. If the group delay is not constant over the entire bandwidth of the signal, the envelope will be distorted. As shown in Fig. 2.10, the group delay at a frequency ω_1 is equal to the negative slope of a tangent to the phase response at ω_1.

Assuming that the phase response of a system is sufficiently smooth, it can be approximated as

$$\theta(\omega + \omega_c) = \tau_p\omega_c + \tau_g\omega_c \tag{2.51}$$

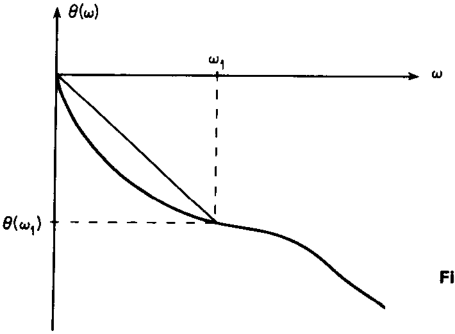


Figure 2.9 Phase delay.

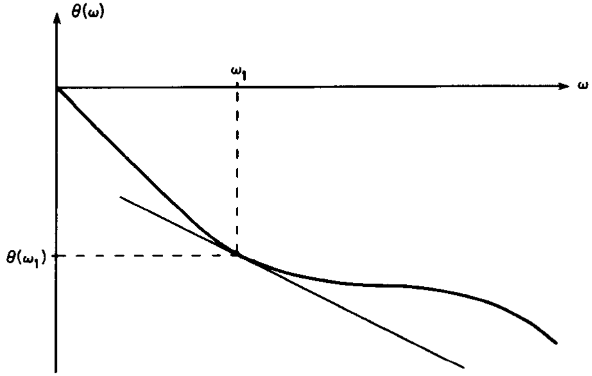


Figure 2.10 Group delay.

If an input signal $x(t) = a(t) \cos \omega_c t$ is applied to a system for which (2.51) holds, the output response will be given by

$$y(t) = Ka(t - \tau_g) \cos[\omega_c(t - \tau_p)] \tag{2.52}$$

Since the envelope $a(t)$ is delayed by τ_g, the group delay is also called the *envelope delay*. Likewise, since the carrier is delayed by τ_p, the phase delay is also called the *carrier delay*.

2.9 Filter Fundamentals

Ideal filters would have rectangular magnitude responses as shown in Fig. 2.11. The desired frequencies are passed with no attenuation, while the undesired frequencies are completely blocked. If such filters could be implemented, they would enjoy widespread use. Unfortunately, ideal filters are noncausal and therefore not realizable. However, there are practical filter designs that approximate the ideal filter characteristics and which are realizable. Each of the major types—Butterworth, Chebyshev, and Bessel—optimizes a different aspect of the approximation.

Magnitude response features of lowpass filters

The magnitude response of a practical lowpass filter will usually have one of the four general shapes shown in Figs. 2.12 through 2.15. In all four cases the filter characteristics divide the spectrum into three general regions as shown. The *pass band* extends from direct current up to the cutoff frequency ω_c. The *transition band* extends from ω_c up to the beginning of the stop band at ω_1, and the *stop band* extends upward from ω_1 to infinity. The cutoff frequency ω_c is the frequency at which the amplitude response falls to a specified fraction (usually -3 dB, sometimes -1 dB) of the peak pass-band values. Defining the

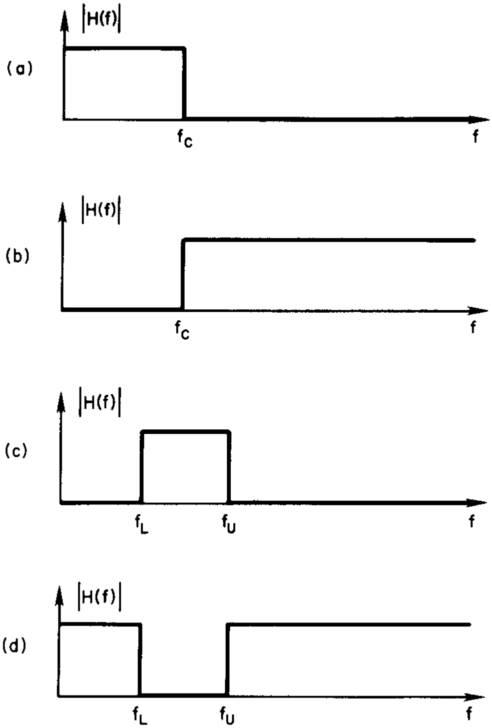


Figure 2.11 Ideal filter responses: (*a*) lowpass, (*b*) highpass, (*c*) bandpass, and (*d*) bandstop.

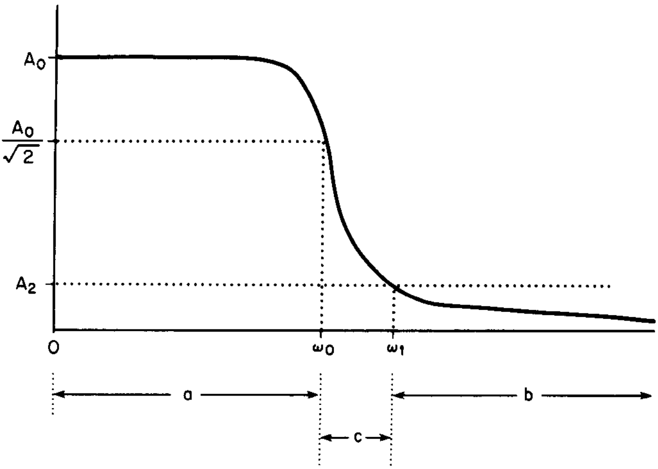


Figure 2.12 Monotonic magnitude response of a practical lowpass filter: (*a*) pass band, (*b*) stop band, and (*c*) transition band.

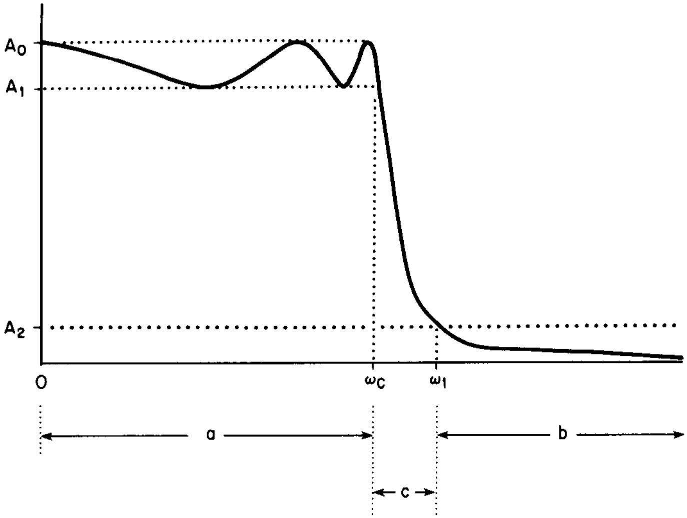


Figure 2.13 Magnitude response of a practical lowpass filter with ripples in the pass band: (*a*) pass band, (*b*) stop band, and (*c*) transition band.

frequency ω_1 which marks the beginning of the stop band is not quite so straightforward. In Fig. 2.12 or 2.13 there really isn't any particular feature that indicates just where ω_1 should be located. The usual approach involves specifying a *minimum stop-band loss* α_2 (or conversely a maximum stop-band amplitude A_2) and then defining ω_1 as the lowest frequency at which the loss

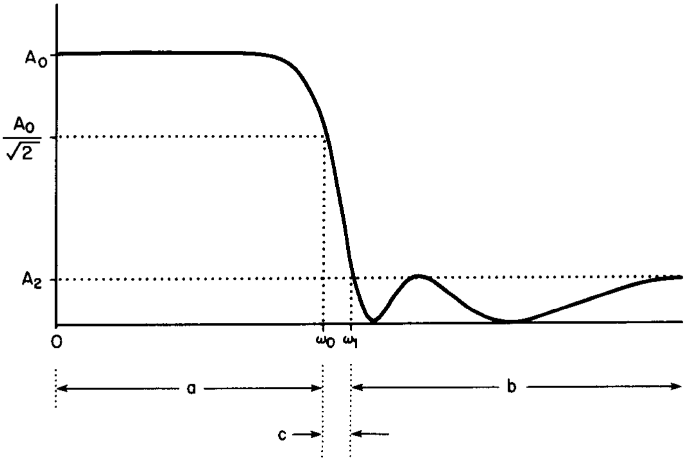


Figure 2.14 Magnitude response of a practical lowpass filter with ripples in the stop band: (*a*) pass band, (*b*) stop band, and (*c*) transition band.

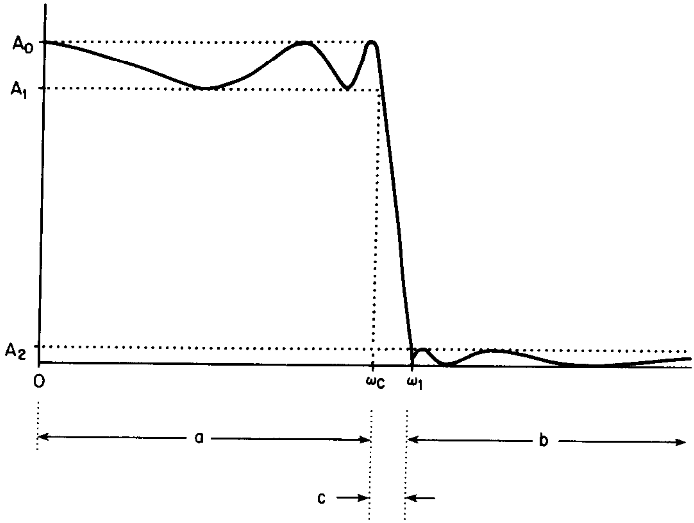


Figure 2.15 Magnitude response of a practical lowpass filter with ripples in the pass band and stop band: (*a*) pass band, (*b*) stop band, and (*c*) transition band.

exceeds and subsequently continues to exceed α_2. The width W_T of the transition band is equal to $\omega_c - \omega_1$. The quantity W_T/ω_c is sometimes called the *normalized transition width*. In the case of response shapes like those shown in Figs. 2.14 and 2.15, the minimum stop-band loss is clearly defined by the peaks of the stop-band ripples.

Scaling of lowpass filter responses

In plots of practical filter responses, the frequency axes are almost universally plotted on logarithmic scales. Magnitude response curves for lowpass filters are scaled so that the cutoff frequency occurs at a convenient frequency such as 1 rad/s (radian per second), 1 Hz, or 1 kHz. A single set of such normalized curves can then be denormalized to fit any particular cutoff requirement.

Transfer functions. For common filter types such as Butterworth, Chebyshev, and Bessel, transfer functions are usually presented in a scaled form such that $\omega_c = 1$. Given such a response normalized for $\omega_c = 1$, we can scale the transfer function to yield the corresponding response for $\omega_c = \alpha$. If the normalized response for $\omega_c = 1$ is given by

$$H_N(s) = \frac{K \prod_{i=1}^{m} (s - z_i)}{\prod_{i=1}^{n} (s - p_i)}$$

then the corresponding response for $\omega_c = \alpha$ is given by

$$H_\alpha(s) = \frac{K \prod_{i=1}^{m} (s - \alpha z_i)}{\alpha^{(m-n)} \prod_{i=1}^{n} (s - \alpha p_i)}$$

Magnitude scaling. The vertical axes of a filter's magnitude response can be presented in several different forms. In theoretical presentations, the magnitude response is often plotted on a linear scale. In practical design situations it is convenient to work with plots of attenuation in decibels using a high-resolution linear scale in the pass band and a lower-resolution linear scale in the stop band. This allows details of the pass-band response to be shown as well as large attenuation values deep into the stop band. In nearly all cases, the data are normalized to present a 0-dB attenuation at the peak of the pass band.

Phase response. The phase response is plotted as a phase angle in degrees or radians versus frequency. By adding or subtracting the appropriate number of full-cycle offsets (that is, 2π rad or 360°), the phase response can be presented either as a single curve extending over several full cycles (Fig. 2.16) or as an equivalent set of curves, each extending over a single cycle (Fig. 2.17). Phase calculations will usually yield results confined to a single 2π cycle. Listing 2.2 contains a C function, **unwrapPhase( )**, that can be used to convert such data into the multicycle form of Fig. 2.16.

Step response. Normalized step response plots are obtained by computing the step response from the normalized transfer function. The inherent scaling of the time axis will thus depend upon the transient characteristics of the normalized filter. The amplitude axis scaling is not dependent upon normal-

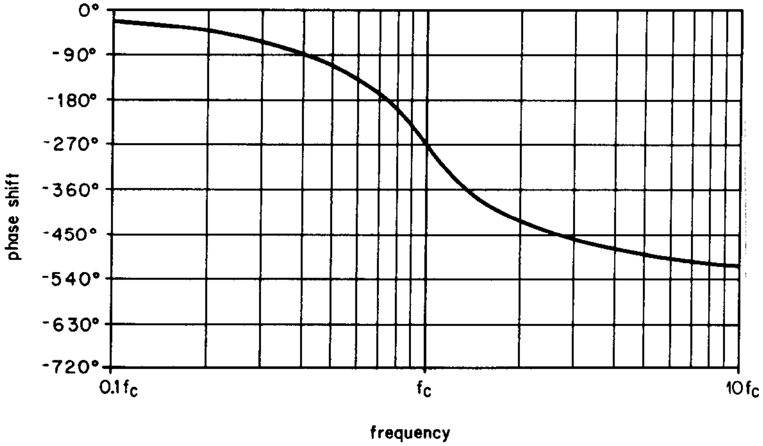


Figure 2.16 Phase response extending over multiple cycles.

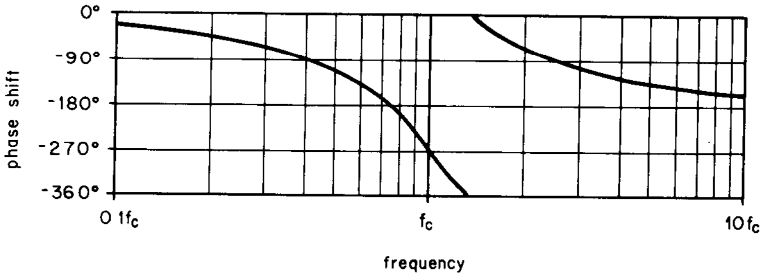


Figure 2.17 Phase response confined to a single-cycle range.

ization. The usual lowpass presentation will require that the response be denormalized by dividing the frequency axis by some form of the cutoff frequency.

Impulse response. Normalized impulse response plots are obtained by computing the impulse response from the normalized-transfer function. Since an impulse response will always have an area of unity, both the time axis and the amplitude axis will exhibit inherent scaling that depends upon the transient characteristics of the normalized filter. The usual lowpass presentation will require that the response be denormalized by multiplying the amplitude by some form of the cutoff frequency and dividing the time axis by the same factor.

Highpass filters

Highpass filters are usually designed via transformation of lowpass designs. Normalized lowpass-transfer functions can be converted into corresponding highpass-transfer functions by simply replacing each occurrence of s with $1/s$. This will cause the magnitude response to be "flipped" around a line at f_c as shown in Fig. 2.18. (Note that this flip works only when the frequency is plotted on a logarithmic scale.) Rather than actually trying to draw a flipped response curve, it is much simpler to take the reciprocals of all the important frequencies for the highpass filter in question and then read the appropriate response directly from the lowpass curves.

Bandpass filters

Bandpass filters are classified as wide band or narrow band based upon the relative width of their pass bands. Different methods are used for obtaining the transfer function for each type.

Wide-band bandpass filters. Wide-band bandpass filters can be realized by cascading a lowpass filter and a highpass filter. This approach will be acceptable as long as the bandpass filters used exhibit relatively sharp

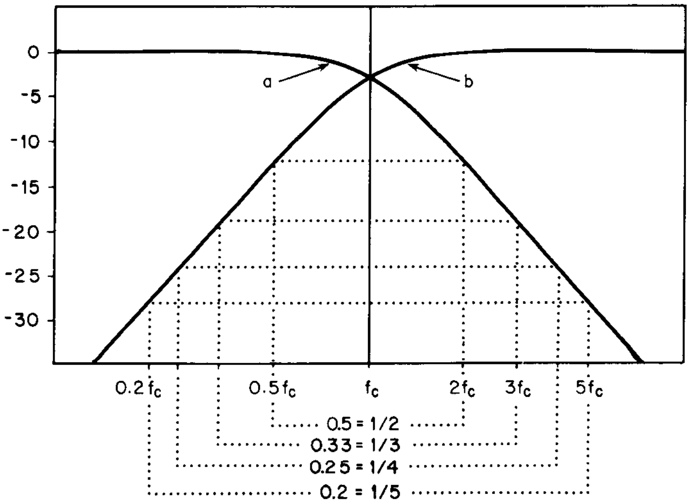


Figure 2.18 Relationship between lowpass and highpass magnitude responses: (*a*) lowpass response and (*b*) highpass response.

transitions from the pass band to cutoff. Relatively narrow bandwidths and/or gradual rolloffs that begin within the pass band can cause a significant center-band loss as shown in Fig. 2.19. In situations where such losses are unacceptable, other bandpass filter realizations must be used. A general rule of thumb is to use narrow-band techniques for pass bands that are an octave or smaller.

Narrow-band bandpass filters. A normalized lowpass filter can be converted into a normalized narrow-band bandpass filter by substituting $[s-(1/s)]$ for s in

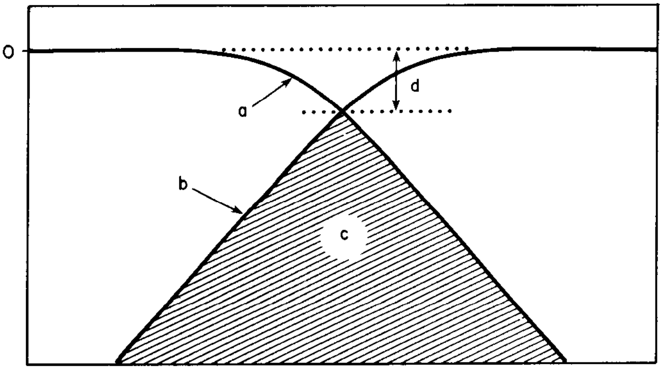


Figure 2.19 Center-band loss in a bandpass filter realized by cascading lowpass and highpass filters: (*a*) lowpass response, (*b*) highpass response, (*c*) pass band of BPF, and (*d*) center-band loss.

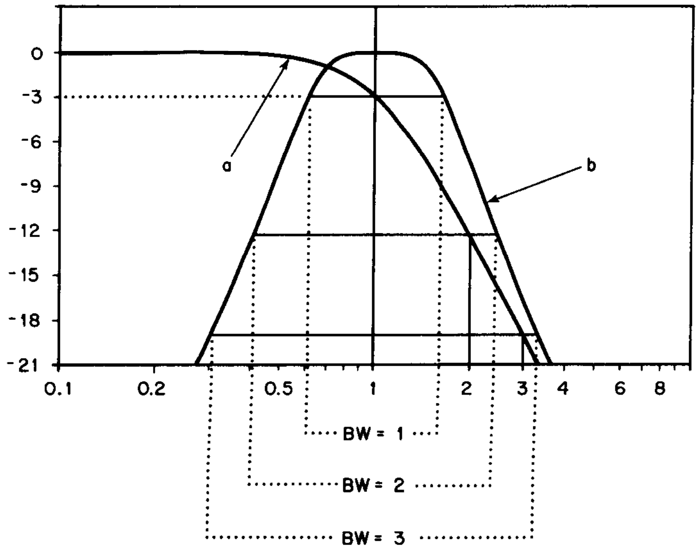


Figure 2.20 Relationship between lowpass and bandpass magnitude responses: (*a*) normalized lowpass response and (*b*) normalized bandpass response.

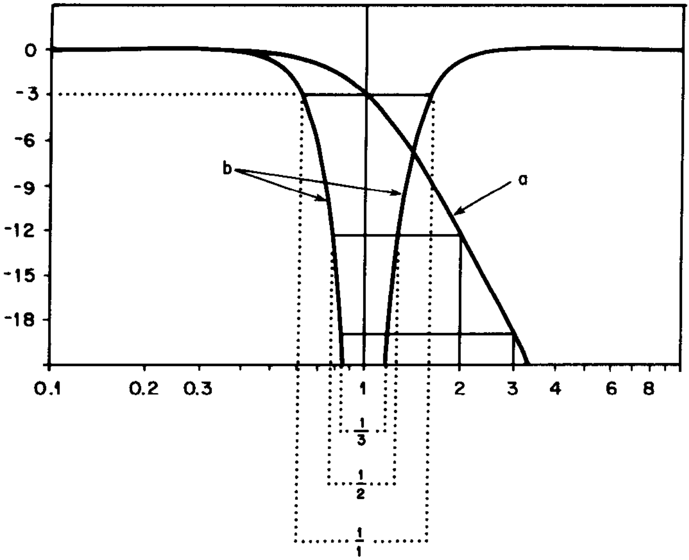


Figure 2.21 Relationship between lowpass and band-stop magnitude responses: (*a*) normalized lowpass response and (*b*) normalized band-stop response.

the lowpass-transfer function. The center frequency of the resulting bandpass filter will be at the cutoff frequency of the original lowpass filter, and the pass band will be symmetric about the center frequency when plotted on a logarithmic frequency scale. At any particular attenuation level, the bandwidth of the bandpass filter will equal the frequency at which the lowpass filter exhibits the same attenuation (see Fig. 2.20). This particular bandpass transformation preserves the magnitude response shape of the lowpass prototype but distorts the transient responses.

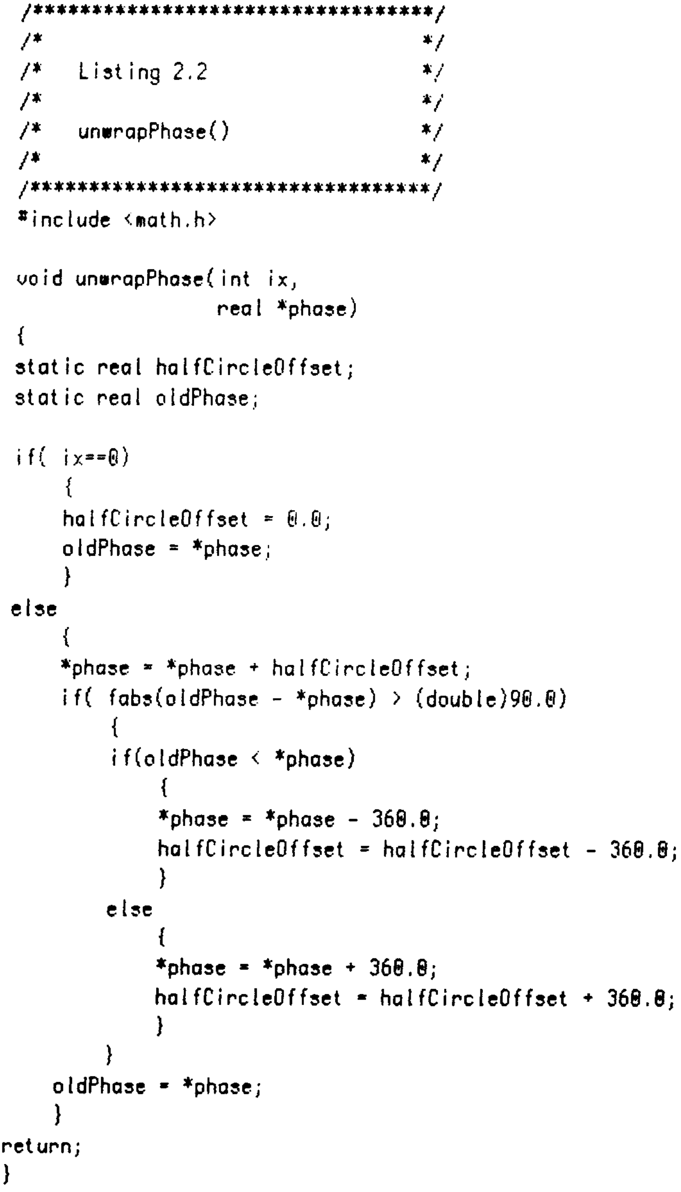
Bandstop filters. A normalized lowpass filter can be converted into a normalized bandstop filter by substituting $s/(s^2 - 1)$ for s in the lowpass-transfer function. The center frequency of the resulting bandstop filter will be at the cutoff frequency of the original lowpass filter, and the stop band will be symmetrical about the center frequency when plotted on a logarithmic *frequency scale. At any particular attenuation level, the width of the stop band will be equal to the reciprocal of the frequency at which the lowpass filter exhibits the same attenuation (see Fig. 2.21).*

Listing 2.1 laguerreMethod( )

```
/***********************************************/
/*                                             */
/*  Listing 2.1                                */
/*                                             */
/*  laguerreMethod()                           */
/*                                             */
/***********************************************/
#include "globDefs.h"
#include "protos.h"
extern FILE *fptr;

int laguerreMethod(
                    int order,
                    struct complex coef[],
                    struct complex *zz,
                    real epsilon,
                    real epsilon2,
                    int maxIterations)
{
int iteration, j;
struct complex d2P_dz2, dP_dz, P, f, g, fSqrd, radical, cwork;
struct complex z, fPlusRad, fMinusRad, deltaZ;
real error, magZ, oldMagZ, fwork;
double dd1, dd2;

z = *zz;
oldMagZ = cAbs(z);

for( iteration=1; iteration<=maxIterations; iteration++)
    {
    d2P_dz2 = cmplx(0.0, 0.0);
    dP_dz = cmplx(0.0, 0.0);
    P = coef[order];
    error = cAbs(P);
    magZ = cAbs(z);


for( j=order-1; j>=0; j--)
    {
    d2P_dz2 = cAdd(dP_dz, cMult(z, d2P_dz2));
    dP_dz = cAdd( P, cMult(dP_dz,z));
    cwork = cMult(P,z);
    P = cAdd( coef[j], cMult(P,z));
    error = cAbs(P) + magZ * error;
    }
error = epsilon2 * error;
d2P_dz2 = sMult(2.0, d2P_dz2);
```

```
     if( cAbs(P) < error)
         {
         *zz = z;
         return 1;
         }
     f = cDiv( dP_dz,P);
     fSqrd = cMult( f, f);
     g = cSub( fSqrd, cDiv( d2P_dz2,P));
     radical = cSub( sMult( (real)order, g), fSqrd);
     fwork = (real)(order-1);
     radical = cSqrt( sMult(fwork, radical));
     fPlusRad = cAdd(f, radical);
     fMinusRad = cSub( f, radical);
     if( (cAbs(fPlusRad)) > (cAbs(fMinusRad)) )
         {
         deltaZ = cDiv( cmplx( (real)order, 0.0), fPlusRad);
         }
     else
         {
         deltaZ = cDiv( cmplx( (real)order, 0.0), fMinusRad);
         }
     z = cSub(z,deltaZ);
     if( (iteration > 6)  && (cAbs(deltaZ) > oldMagZ) )
         {
         *zz = z;
         return 2;
         }
     if( cAbs(deltaZ) < ( epsilon * cAbs(z)))
         {
         *zz = z;
         return 3;
         }
     }
fprintf(fptr,"Laguerre method failed to converge \n");
return -1;
}
```

Listing 2.2 unwrapPhase( )



```
/**********************************/
/*                                */
/*   Listing 2.2                  */
/*                                */
/*   unwrapPhase()                */
/*                                */
/**********************************/
#include <math.h>

void unwrapPhase( int ix,
                  real *phase)
{
static real halfCircleOffset;
static real oldPhase;

if( ix==0)
    {
    halfCircleOffset = 0.0;
    oldPhase = *phase;
    }
else
    {
    *phase = *phase + halfCircleOffset;
    if( fabs(oldPhase - *phase) > (double)90.0)
        {
        if(oldPhase < *phase)
            {
            *phase = *phase - 360.0;
            halfCircleOffset = halfCircleOffset - 360.0;
            }
        else
            {
            *phase = *phase + 360.0;
            halfCircleOffset = halfCircleOffset + 360.0;
            }
        }
    oldPhase = *phase;
    }
return;
}
```