CHAPTER 6

SIGNAL PROCESSING

John Turnbull
Case Western Reserve University
Cleveland, Ohio

1 FREQUENCY-DOMAIN ANALYSIS OF LINEAR SYSTEMS **189**

2 BASIC ANALOG FILTERS **191**
2.1 Butterworth 193
2.2 Tchebyshev 194
2.3 Inverse Tchebyshev 195
2.4 Elliptical 195
2.5 Arbitrary Frequency Response Curve Fitting by Method of Least Squares 196
2.6 Circuit Prototypes for Pole and Zero Placement for Realization of Filters Designed from Rational Functions 197

3 BASIC DIGITAL FILTER **197**
3.1 z-Transforms 198
3.2 Design of FIR Filters 198
3.3 Design of IIR Filters 201
3.4 Design of Various Filters from Low-Pass Prototypes 203
3.5 Frequency-Domain Filtering 205

4 STABILITY AND PHASE ANALYSIS **206**
4.1 Stability Analysis 206
4.2 Phase Analysis 206
4.3 Comparison of FIR and IIR Filters 208

5 EXTRACTING SIGNAL FROM NOISE **208**

REFERENCES **208**

1 FREQUENCY-DOMAIN ANALYSIS OF LINEAR SYSTEMS

Signals are any carriers of information. Our objective in signal processing involves the encoding of information for the purpose of transmission of information or decoding the information at the receiving end of the transmission. Unfortunately, the signal is often corrupted by noise during our transmission, and hence it is our objective to extract the information from the noise. The standard method most commonly used for this involves filters that exploit some separation of the signal and noise in the frequency domain. To this end, it is useful to use frequency-domain tools such as the Fourier transform and the Laplace transform in designing and analyzing various filters. The Fourier transform of a function of a time is

$$\mathcal{F}\{f(t)\} = F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{j\omega t}\, dt \qquad j^2 = -1 \tag{1}$$

For continuous systems, the transfer characteristics of a filter system is a function that gives information of the gain versus frequency. The Laplace transform for a given time-domain function is

$$\mathcal{L}\{f(t)\} = F(s) = \int_{0}^{\infty} f(t) e^{-st}\, dt \tag{2}$$

The steady-state Laplace transform (i.e., neglecting transients) for the derivative and integral of a given function is

Copyrighted Material
Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

$$\mathfrak{L}\left\{\frac{df(t)}{dt}\right\} = sF(s) \qquad \mathfrak{L}\left\{\int f(t)\,dt\right\} = \frac{F(s)}{s} \tag{3}$$

By convention, functions in the time domain use t as the independent variable and functions in the Laplace domain use s as the independent variable. For this reason, the Laplace domain is commonly called the *S-domain.* The Fourier transform and the Laplace transform are similar but different in two respects: (1) The Fourier transform integrates the signal over all time while the Laplace transform integrates for positive times only and (2) the exponent of the kernel in the Laplace transform is complex with both real and imaginary components while the exponent in the Fourier transform has imaginary component and no real component. By using Euler's identity, $e^{j\omega} = \cos(\omega) + j\,\sin(\omega)$, we see that

$$\begin{aligned}\mathfrak{F}\{f(t)\} = F(\omega) &= \int_{-\infty}^{\infty} f(t)e^{j\omega}\,dt \\ &= \int_{-\infty}^{\infty} f(t)\,\cos(\omega t)\,dt + j\int_{-\infty}^{\infty} f(t)\,\sin(\omega t)\,dt \\ &= \langle\cos(\omega t)\rangle + j\langle\sin(\omega t)\rangle.\end{aligned} \tag{4}$$

We approximate the Fourier transform from discrete samples $f(k) \leftrightarrow F(k)$, where $0 \le k \le N$ and $F(k) = \alpha_k + j\beta_k$:

$$\alpha_k = \sum_{i=0}^{N-1} f(i)\,\cos\left(2\pi\,\frac{i\cdot k}{N}\right) \qquad \beta_k = \sum_{i=0}^{N-1} f(i)\,\sin\left(2\pi\,\frac{i\cdot k}{N}\right) \tag{5}$$

However, if the number of points we transform is a composite number and not a prime number, we can restructure our calculations to eliminate some of the calculations. Furthermore, of the factors that are themselves composite, we can further factor and eliminate more calculations. For this reason, the most efficient vector sizes are those that are highly composite. As an example, consider the simple case of transforming four points. We can express Eq. (5) for this case in matrix form as

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \rho & \rho^2 & \rho^3 \\ 1 & \rho^2 & \rho^4 & \rho^6 \\ 1 & \rho^3 & \rho^6 & \rho^9 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

where ρ is the principal root of 1. The nth principal root of 1 is $\cos(2\pi/n) + j\,\sin(2\pi/n)$. We can then factor the matrix into two sparse matrices:

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & \rho^2 & 0 & 0 \\ 0 & 0 & 1 & \rho \\ 0 & 0 & 1 & \rho^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & \rho^2 & 0 \\ 0 & 1 & 0 & \rho^2 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

Although it is possible to implement this efficient algorithm—known as the fast Fourier transform (FFT)—for any vector size that is a composite number, it is most commonly implemented for vector sizes in which all factors are 2. The effort in evaluating Eq. (5) increases with the square of the number of points to transform, or $O(n^2)$. In contrast, the effort for the FFT of order 2^p increases proportionately to $O(n\,\log(n))$. Finally, we can use the FFT to estimate the power spectral density (PSD) of a given discrete signal by computing the square of the magnitude of the FFT. The following algorithm describes the method for computing the Fast Fourier Transform.[1]

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

FAST FOURIER TRANSFORM 1 *Given a vector* $\{x_1, \ldots, x_n\}$ *of complex numbers, where n is some integer and there exists some integer p such that* $n = 2^p$. *This algorithm outputs the Fourier transform overwriting the input vector* **x**.

```
k ← 1
For i = 1 To n
    If k > i
        swap(x_i, x_k)
    End If
    m = n/2
    While m ≥ 1 And k > m
        k ← k − m
        m ← m/2
    End While
    k ← k + m
Next i
M_max ← 1
While n > M_max
    i_c ← 2·M_max
    θ ← π/M_max
    wp ← −2 sin²(θ/2) − j·sin(θ)
    w ← 1
    For m = 1 To M_max
        For i = m To n By i_c
            k ← i + M_max
            xtemp ← x_i − w·x_k
            x_i ← x_i + w·x_k
            x_k ← xtemp
        Next i
        w ← w·(wp + 1)
    Next m
    M_max = i_c
End While
```

2 BASIC ANALOG FILTERS

Linear filters apply frequency-specific gains to a signal. This is often done to enhance desired portions of the spectrum while attenuating or eliminating other portions. Four common filters are low pass, high pass, bandpass, and band reject. The objective of an ideal low-pass filter is to eliminate a range of undesired high frequencies from a signal and leave the remaining portion undistorted. To this end, an ideal low-pass filter will have a gain of 1 for all frequencies less than some desired cutoff frequency f_c and a gain of 0 for all frequencies greater than f_c, as seen in Fig. 1. There are various rational functions that approximate this ideal. But because of the discontinuity in the ideal low-pass response, all realizations of this ideal with be an approximation. The various approximation functions generally trade off between three characteristics: passband ripple, stop-band ripple, and the transition width, shown in Fig. 2. Four common rational function approximations for low-pass filters are the Butterworth, the Tchebyshev Types I and II, and the elliptical filter. By convention, we use $H(s)$ as the transfer function from which we determine the frequency response, where

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

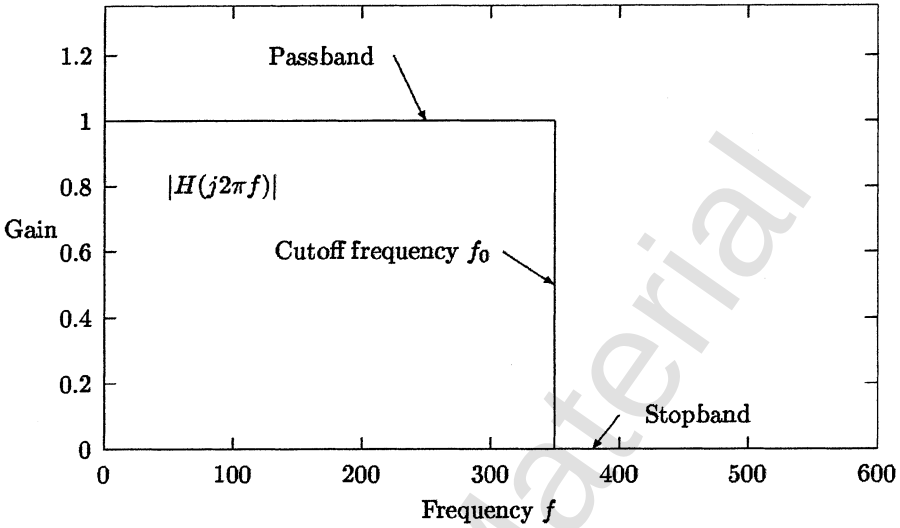


Figure 1 Frequency response for an ideal low-pass filter.

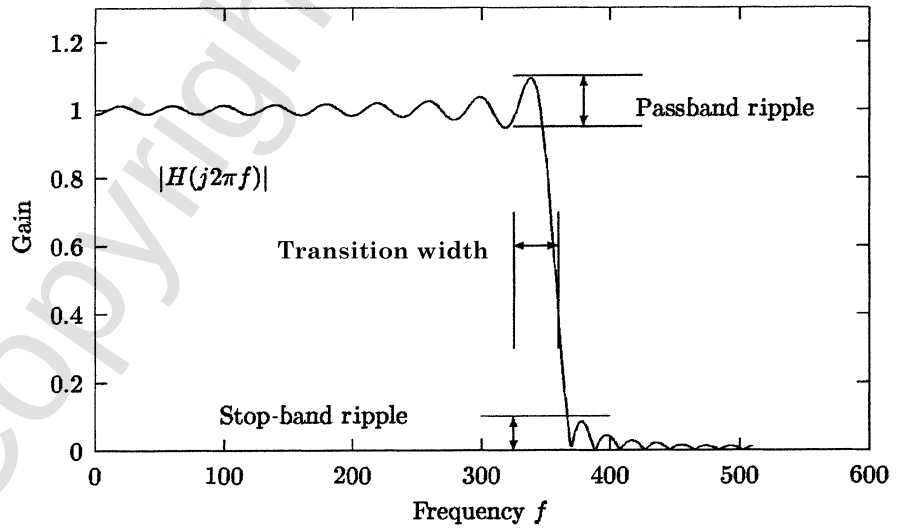


Figure 2 Typical frequency response for an approximate low-pass filter.

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

$$H(s) = \frac{V_{\text{output}}(s)}{V_{\text{input}}(s)} \tag{6}$$

is the output-to-input gain of a standard two-port system.

2.1 Butterworth

The Butterworth filter has a smooth passband region (frequencies less than f_c hertz) and a smooth stop band (frequencies greater than f_c) and a comparatively wide transition region as shown in Figs. 3 and 4. Let S_c be the cutoff frequency; then a low-pass rational function approximation is as follows:

$$|H(s)|^2 = \frac{1}{1 + (s/s_c)^{2N}} \tag{7}$$

where $2\pi f_c = s_c$. In factored form,

$$H(s) = \frac{1}{(s - p_0)(s - p_1) \cdots (s - p_{N-1})} \tag{8}$$

where $p_i = \alpha_i + j\beta_i$ and

$$\begin{aligned} \alpha_i &= 2\pi f_c \cos\left(\frac{\pi}{2N}(N + 2i + 1)\right) \qquad i = 0, \ldots, N - 1 \\ \beta_i &= 2\pi f_c \sin\left(\frac{\pi}{2N}(N + 2i + 1)\right) \qquad i = 0, \ldots, N - 1 \end{aligned} \tag{9}$$

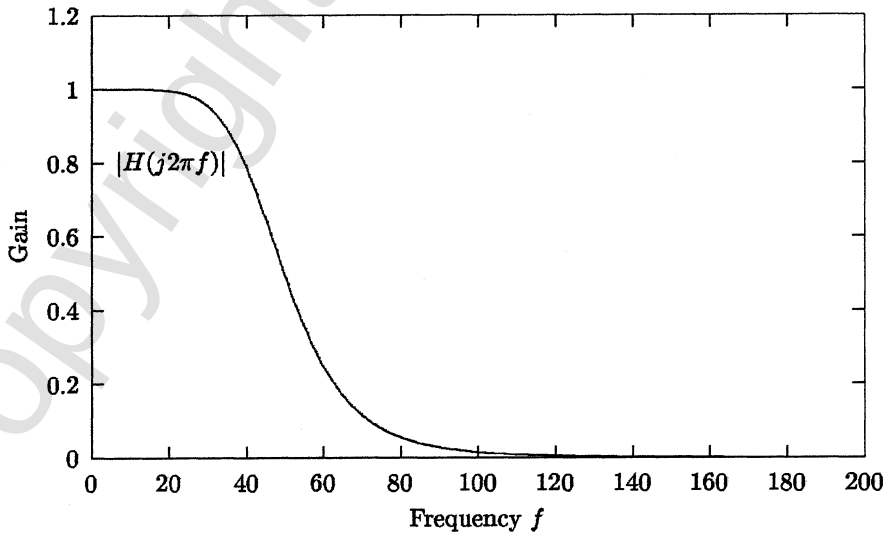


Figure 3 Frequency response to a third-order Butterworth filter.

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

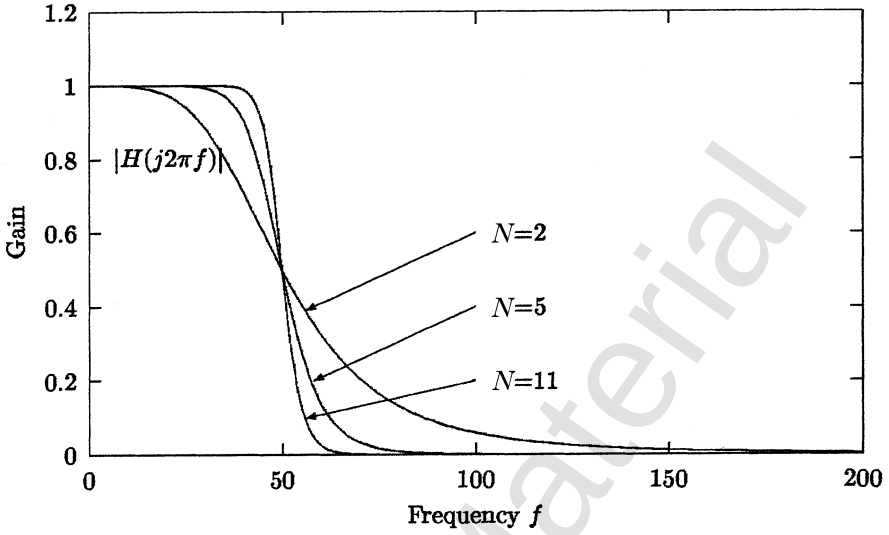


Figure 4 Frequency response to Butterworth filters of order 2, 5, and 11.

2.2 Tchebyshev

Unlike the Butterworth rational function, the Tchebyshev (Type I) rational function permits some ripple to occur in the passband (frequencies less than f_c in the low-pass filter), in exchange for a sharper transition region compared to a Butterworth filter of equal order N, as seen in Fig. 5:

$$|H(s)|^2 = \frac{1}{1 + \epsilon^2 T_N^2 (s/s_c)} \tag{10}$$

Where T_N is the Tchebyshev polynomial defined as

$$T_N(s) = \cos[n \cos^{-1}(s)] \tag{11}$$

The pole placements $p_i = \alpha_i + j\beta_i$ for the Tchebyshev Type I filter are

$$\begin{aligned} \alpha_i &= 2\pi f_c \sinh(v_0) \cos\left(\frac{\pi}{2N}(N + 2i + 1)\right) \qquad i = 0, \ldots, N-1 \\ \beta_i &= 2\pi f_c \cosh(v_0) \sin\left(\frac{\pi}{2N}(N + 2i + 1)\right) \qquad i = 0, \ldots, N-1 \end{aligned} \tag{12}$$

where

$$v_0 = \frac{1}{N} \sinh^{-1}\left(\frac{1}{\epsilon}\right) \tag{13}$$

$$\epsilon = \sqrt{\frac{1}{(1-r)^2} - 1} \qquad 0 < r < 1 \tag{14}$$

where r is this amplitude of the ripple in proportion to the gain of the passband.

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

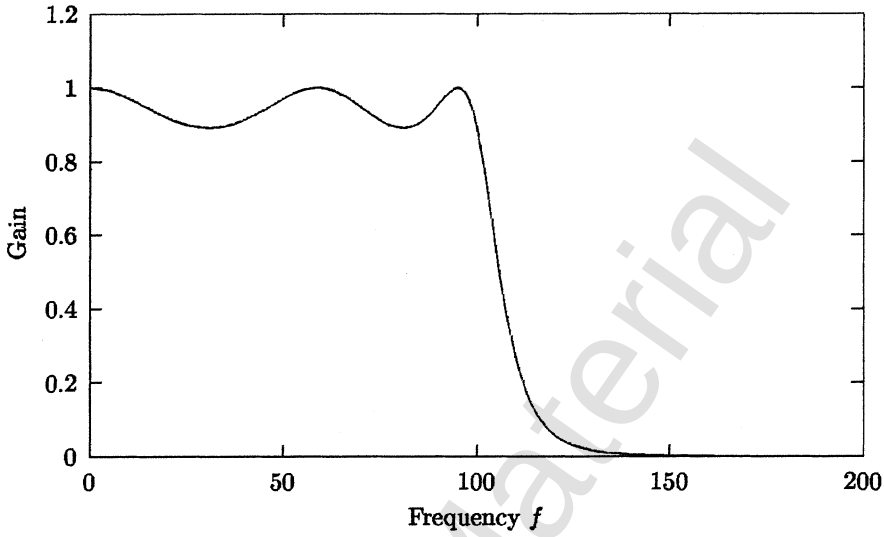


Figure 5 Frequency response to a fifth-order Tchebyshev filter.

2.3 Inverse Tchebyshev

The inverse Tchebyshev (or Tchebyshev Type II) filter has a smooth passband, ripple in the stop band (frequencies less than f_c for the low-pass filter), and a sharper transition region compared to the Butterworth function of equal order N, as seen in Fig. 6:

$$|H(s)|^2 = \frac{\epsilon^2 T_N^2 (s_c/s)}{1 + \epsilon^2 T_N^2 (s_c/s)} \tag{15}$$

The zero placements $\zeta_i = \alpha_i + \beta_i$ for the Tchebyshev type II filter are

$$\zeta_i = \frac{1}{\sin(i\pi/2N)} \qquad 0 \le i \le N - 1 \tag{16}$$

The pole placements $p_i = \alpha_i + j\beta_i$ for the Tchebyshev Type II filter are simply the reciprocals of the pole placements computed for the Tchebyshev Type I filter.

2.4 Elliptical

The Elliptical filter has ripple in both the passband and the stop band but, in exchange, has the narrowest transition region for equal filter order N, as seen in Fig. 7. The derivation and implementation for the determination of the poles and zeros involve the Jacobian elliptic function:

$$f(t, k) = \int_0^t \frac{dx}{\sqrt{1 - k^2 \sin^2(x)}} \tag{17}$$

The method is beyond the scope of this chapter. The interested reader is referred to Refs. 2 and 3.

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

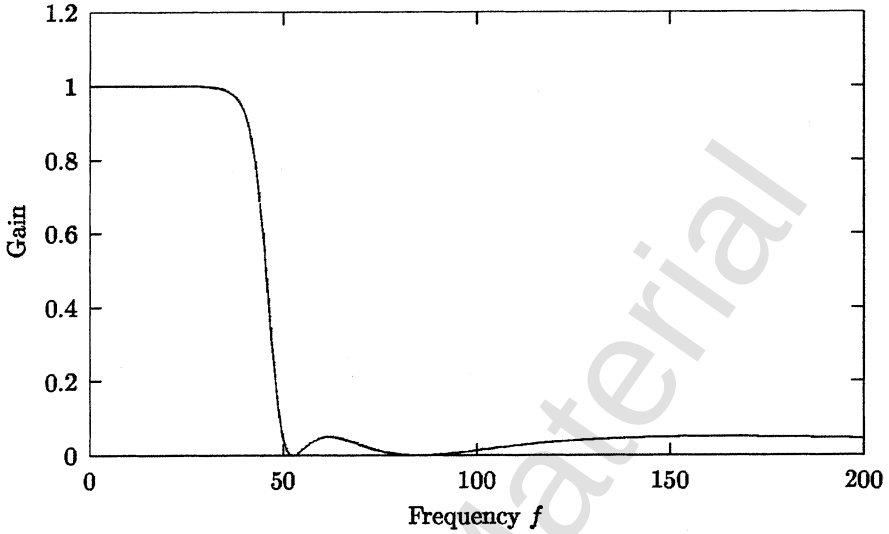


Figure 6 Frequency response to a fifth-order inverse Tchebyshev filter.

2.5 Arbitrary Frequency Response Curve Fitting by Method of Least Squares

It is possible to design a filter to approximate a desired frequency response $F(\omega)$ by the method of least squares. Consider a transfer function in factored form as

$$H(s) = G \frac{\prod_{i=i}^{M} s - \zeta_i}{\prod_{k=1}^{N} s - p_k} \tag{18}$$

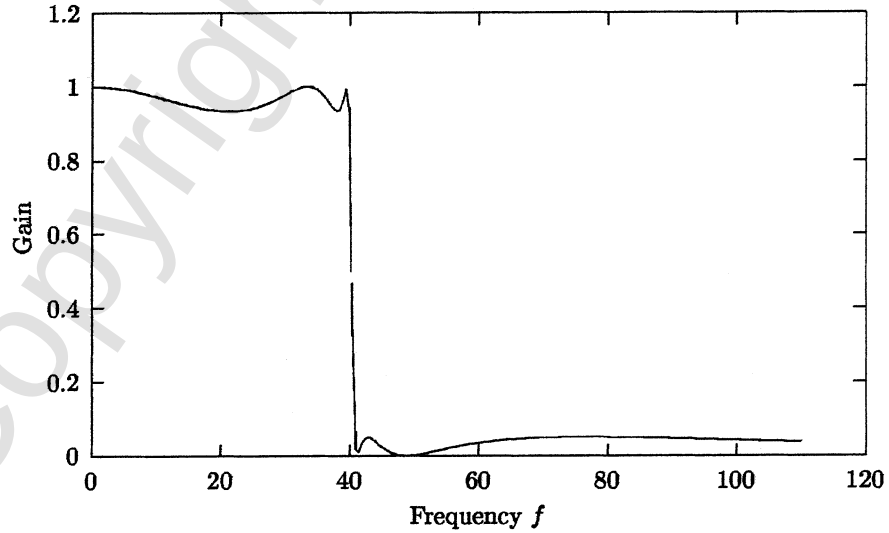


Figure 7 Frequency response to a fifth-order elliptical filter.

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

Minimize

$$\chi^2(G;\boldsymbol{\zeta};\mathbf{p}) = \int_D [|H(j\omega)| - F(\omega)]^2 \, d\omega \tag{19}$$

subject to

$$\Re\mathrm{e}(p_k) < 0 \qquad \text{for all } k = 1, \ldots, N \tag{20}$$

Unfortunately, this results in a nonlinear system of equations. Furthermore, the topography of this objective function is generally complicated with many local minima, making standard gradient-descent methods unfeasible. It is usually best to use finite-impulse-response (FIR) filtering or filtering in the frequency domain for these types of problems. If an infinite-impulse-response (IIR) filter is desired, the reader is referred to Prony's method, which linearizes this system and finds an approximate optimal solution.[4]

2.6 Circuit Prototypes for Pole and Zero Placement for Realization of Filters Designed from Rational Functions

The voltage–current relation for a resistor (R), inductor (L), and capacitor (C) is

$$v_r = iR \qquad v_l = L\frac{di}{dt} \qquad v_c = \frac{1}{C}\int_{-\infty}^{t} i \, dr \tag{21}$$

These relationships are represented in the S-domain as

$$V_r(s) = RI(s) \qquad V_l(s) = sLI(s) \qquad V_c = \frac{I(s)}{sC} \tag{22}$$

Thus, the general transfer function for any linear circuit involving standard passive, active, and reactive devices is a rational function, that is, a ratio of polynomials:

$$H(s) = \frac{\sum_{i=0}^{M-1} a_i s^i}{\sum_{k=0}^{N-1} b_k s^k} = \frac{V_0}{V_i} = -\frac{Z_2}{Z_1} \tag{23}$$

Thus one can construct any arbitrary transfer function through a serial placement of this building block circuit prototype shown in Fig. 8. Table 1 gives circuit elements for Z_1 and Z_2 to construct the basic prototype circuits.

3 BASIC DIGITAL FILTER

Basic linear digital filters are of two types: those that have a finite response to an impulse, or FIR, and those that have an infinite response to an impulse (IIR). The general form of a linear digital filter is

$$y_k = b_1 y_{k-1} + b_2 y_{k-2} + \cdots + b_{k-m} y_{k-m} + a_0 x_k + a_1 x_{k-1} + \cdots + a_n x_{k-n} \tag{24}$$

where $k - i$ represents the ith delay. That is, the kth output from a linear digital filter is some linear combination of previous inputs and outputs. The filter will have a finite response to an impulse if $b_1 = b_2 = \cdots = b_m = 0$; otherwise, the filter is of type IIR.

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

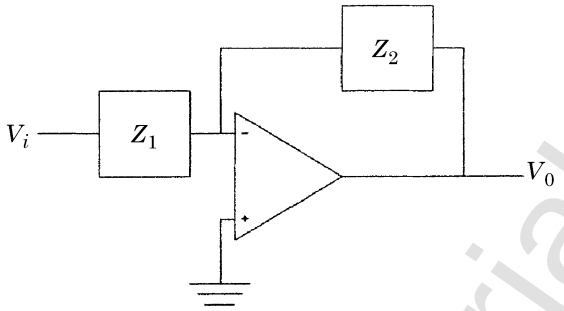


Figure 8 Prototype circuit element for construction of analog filter.

3.1 z-Transforms

The z-transform is used to analyze the frequency response and stability of a system of difference equations in much the same way that the Laplace transform is used to analyze the frequency response and stability of a system of differential equations. The z-transform of Eq. (24) is

$$H(z) = \frac{a_0 + a_1 z^{-1} + \cdots + a_M z^{-M}}{1 - b_1 z^{-1} - \cdots b_N z^{-N}} = \frac{\sum_0^M a_j z^{-j}}{1 - \sum_{j=1}^N b_j z^{-j}} \tag{25}$$

We determine the frequency response by $|H(e^{j\omega})|$, where $0 \leq \omega \leq 2\pi$ and corresponds to the scaled frequencies of our sampled system from 0 Hz up to the sampled frequency. The function $e^{j\omega}$ is a periodic signal. For $\pi < \omega \leq 2\pi$, $e^{j\omega} = e^{j(\pi-\omega)}$, and for $2\pi < \omega \leq 4\pi$, $e^{j\omega} = e^{j(\omega-2\pi)}$. Thus frequencies greater than π, corresponding to the Nyquist frequency, or one-half of the sampling frequency assume an identical characteristic to an analogous frequency less than π. This phenomenon is called *aliasing* and is illustrated in Fig. 9.

3.2 Design of FIR Filters

It is possible to use the Fourier transform to determine the coefficients to an FIR filter. However, the Fourier coefficients are generally complex numbers, and when working with real signals, it is desirable to have a real coefficients in our filter. To do this, we apply Euler's identity and observe from Eq. (4) that the coefficients will be real if the inner products with

Table 1 Circuit Elements for the Construction of Basic Transfer Function Prototypes

Single pole	$Z_1 \leftarrow$ resistor
	$Z_2 \leftarrow RC$ in parallel
Single zero	$Z_1 \leftarrow RC$ in parallel
	$Z_2 \leftarrow$ resistor
Complex-conjugate pole pair	$Z_1 \leftarrow LRC$ in series
	$Z_2 \leftarrow$ capacitor
Complex-conjugate zero pair	$Z_1 \leftarrow$ capacitor
	$Z_2 \leftarrow LRC$ series

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

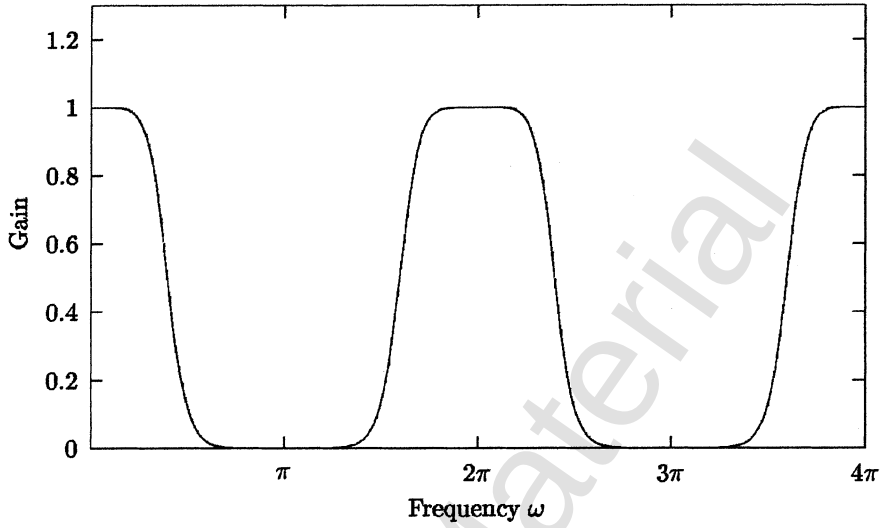


Figure 9 Demonstration of aliasing from digital filter response.

the sine function are all zero. We can artificially construct our desired frequency response so that this will be so. To do this, suppose $F(\omega)$ is the desired frequency response where $0 \leq \omega \leq \pi$:

Step 1. Augment the domain of the function over $0 \leq \omega \leq 2\pi$.

Step 2. Augment the function values from $\pi \leq \omega \leq 2\pi$ as $F(\omega) = F(2\pi - \omega)$.

Step 3. Compute the discrete cosine Transform [α_i coefficients from Eq. (5)] over the range $0 \leq \omega \leq 2\pi$.

For a discrete system with a desired FIR filter of length N:

Step 1. Augment the domain of the function over $1 \leq i \leq 2N$.

Step 2. Augment the function values from $N + 1 \leq i \leq 2N$ as $F(i) = F(2N - i)$.

Step 3. Construct the FIR filter with $2N$ points from the discrete cosine transform of the desired frequency response.

Step 4. Keep the first N coefficients and truncate the remaining coefficients.

The discrete cosine Transform of an ideal low-pass filter is a sinc function, defined as

$$\text{sinc}(x) = \begin{cases} \dfrac{\sin(\pi x)}{\pi x} & \text{if } x \neq 0 \\ 1 & \text{if } x = 0 \end{cases} \tag{26}$$

Therefore, the coefficients to a low-pass FIR filter with cutoff frequency f_c and length $2N - 1$ are determined using Eq. (27), where ω_c is the desired cutoff frequency divided by the Nyquist frequency:

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

$$h(i) = \omega_c \operatorname{sinc}\left[\omega_c\left(i - \left\lfloor\frac{N}{2}\right\rfloor\right)\right] \qquad 1 \le i \le N \tag{27}$$

Where $\lfloor x \rfloor$ is the greatest integer less than or equal to x and ω_c is the desired cutoff frequency divided by the Nyquist frequency.

Windowing

This FIR filter will have ripple in the passband and in the stop band. It is possible to suppress these ripples and smooth the frequency response, but the trade-off will be an increased transition width. The method for suppressing these ripples is with the application of a windowing function. There is a large class of windowing functions that allow the designer to determine how he or she wishes to trade off the transition width and how much ripple is to be tolerated. The design of an FIR filter with windowing involves the use of Eq. (26) for the determination of the FIR coefficients followed by the component-by-component product of the coefficients with the windowing values, that is, $h'(i) = h(i)\ \text{win}(i)$. Below, is a list of common windows[5]:

Rectangular:

$$\text{win}(i) = 1 \qquad 0 \le i \le N - 1 \tag{28}$$

Bartlett:

$$\text{win}(i) = \begin{cases} \dfrac{2i}{N-1} & 0 \le i \le \dfrac{N-1}{2} \\ 2 - \dfrac{2i}{N-1} & \dfrac{N-1}{2} \le i \le N-1 \end{cases} \tag{29}$$

Hanning:

$$\text{win}(i) = \tfrac{1}{2}\left[1 - \cos\left(\frac{2\pi i}{N-1}\right)\right] \qquad 0 \le i \le N - 1 \tag{30}$$

Hamming:

$$\text{win}(i) = 0.54 - 0.46\cos\left(\frac{2\pi i}{N-1}\right) \qquad 0 \le i \le N - 1 \tag{31}$$

Blackman:

$$\text{win}(i) = 0.42 - 0.5\cos\left(\frac{2\pi i}{N-1}\right) + 0.08\cos\left(\frac{4\pi i}{N-1}\right) \qquad 0 \le i \le N - 1 \tag{32}$$

Table 2 gives a list of several common windowing functions together with their characteristics.[5]

Figure 10 demonstrates the effect of a Hanning window.

FIR High-Pass and Bandpass Design

The design of a high-pass filter is simply $1 - H(z)$. In the time domain, this is

$$h(i) = \omega_c \operatorname{sinc}\left[\omega_c\left(i - \left\lfloor\frac{N}{2}\right\rfloor\right)\right] - \omega_c\delta\left(\left\lfloor\frac{N}{2}\right\rfloor\right) \qquad 1 \le i \le N \tag{33}$$

where

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

Table 2 Comparison of Characteristics for Commonly Used Windowing Functions

Window Name	Minimum Stop-Band Attenuation (dB)
Rectangular	−21
Bartlett	−25
Hanning	−44
Hamming	−53
Blackman	−74

$$\delta(i) = \begin{cases} 1 & i = 0 \\ 0 & i \neq 0 \end{cases} \tag{34}$$

We construct a bandpass by filtering the data with a high pass-filter, then filtering the output with a low-pass filter. Or we can combine the two filters together into a single filter by convolving the coefficients.

3.3 Design of IIR Filters

The common strategy in designing IIR filters is as follows:

Step 1. Design a rational function in the *S*-domain in factored form that best approximates the desired frequency response characteristics (using Butterworth, Tchebyshev, elliptical, etc.).

Step 2. Transform the poles and zeros into the *z*-domain

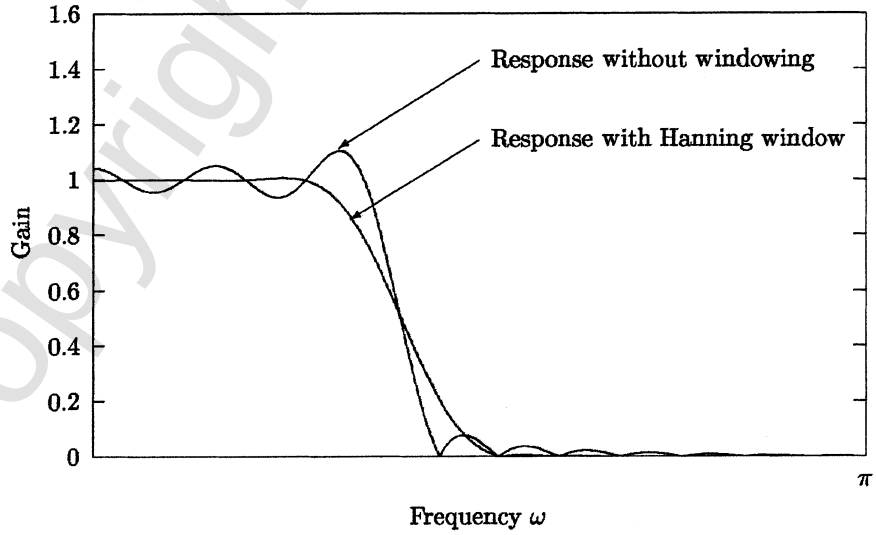


Figure 10 Comparison of FIR filter with Hanning window and with windowing.

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

Step 3. Reconstruct the rational function.

Step 4. Realize the difference equation by inverse z-transform of the rational polynomials.

There are different S- to z-transforms. The two most common are the impulse-invariant and the bilinear transformation. The impulse-invariant transformation is

$$z = e^{sT} \tag{35}$$

where T is the sampling period. This method is usually not used because it can cause aliasing. The bilinear transformation avoids this but distorts the mapping in other ways for which we need to compensate. The bilinear transformation is

$$z = \frac{2/T + s}{2/T - s} \tag{36}$$

The S- to z-transformation is not exact and always involves various trade-offs. Because of this, the actual placement of the cutoff in a designed digital filter is misplaced and the error increases for cutoff frequencies closer to the Nyquist frequency. To compensate for this effect, we apply Eq. (37) in the design of our filter:

$$f_c' = \frac{1}{\pi T} \tan(\pi f_c T) \tag{37}$$

This process is called "prewarping."

Example IIR Design

For a digital system with a sampling rate of 100 Hz, a third-order low-pass Tchebyshev filter with a cutoff frequency at 15 Hz and with 20% ripple in the passband is designed as follows:

Step 1. Compute the Nyquist frequency $f_N = 100 \text{ Hz}/2 = 50 \text{ Hz}$.

Step 2. Compute f' using Eq. (37) to prewarp the cutoff frequency:

$$f' = 16.22 \text{ Hz}$$

Step 3. Compute v_0 using Eqs. (13) and (14):

$$\epsilon = 0.75 \qquad v_0 = 0.3662$$

Step 4. Determine the poles and zeros of the analog system using Eq. (12):

$$p_1 = -19.0789 + j94.2364$$

$$p_2 = -38.1578$$

$$p_3 = -19.0789 - j94.2364$$

Step 5. Map the poles from the S-domain into the z-domain using the bilinear transformation equation (36) and $T = (1/\text{sampling rate}) = 0.01$ s:

$$pz_1 = 0.5407 + j0.6627$$

$$pz_2 = 0.6796$$

$$pz_3 = 0.5407 - j0.6627$$

Step 6. Map the zeros from the S-domain into the z-domain also using Eq. (36). Although we may be tempted to conclude that there are no zeros in the S-domain, since our

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

numerator is constant, we note that $H(s) \to 0$ as $s \to \infty$. Thus mapping the bilinear transformation for this case yields

$$\lim_{s\to\infty} \frac{2/T + s}{2/T - s} = -1$$

And since this is a third-order system,

$$z_1 = -1 \qquad z_2 = -1 \qquad z_3 = -1$$

Step 7. Expand the numerator and denominator polynomials:

$$\frac{(z - \zeta_1)(z - \zeta_2)(z - \zeta_3)}{(z - pz_1)(z - pz_2)(z - pz_3)} = \frac{1 + 3z^{-1} + 3z^{-2} + z^{-3}}{-1 + 1.7611z^{-1} - 1.466z^{-2} + 0.4972z^{-3}}$$

Step 8. Normalize the filter so that the gain in the passband will be 1. In this case, we know that the gain should be 1 at $\omega = 0$. Hence, we evaluate $|H(e^{j\omega})|$ for $\omega = 0$: $|H(e^0)| = 38.4$. The normalized transfer function is then

$$H(z) = \frac{0.026 + 0.078z^{-1} + 0.078z^{-2} + 0.026z^{-3}}{-1 + 1.7611z^{-1} - 1.466z^{-2} + 0.4972z^{-3}}$$

Step 9. Realize the difference equation from inverse z-transformation of the derived transfer function:

$$y_n = 1.7611y_{n-1} - 1.466y_{n-2} + 0.4972y_{n-3} + 0.026x_n + 0.078x_{n-1} + 0.078x_{n-2} + 0.026x_{n-3}$$

3.4 Design of Various Filters from Low-Pass Prototypes

The procedure for designing a high-pass, bandpass, or band-reject IIR filter is as follows: First, design, by pole–zero placement, a low-pass filter (Butterworth, Tchebyshev, etc.) with an arbitrary cutoff frequency (though for practical considerations, it is best to choose a value midway between 0 Hz and Nyquist), transform the poles and zeros according to the following formulas, reconstitute a new transfer function from the transformed poles and zeros, then realize the digital filter by taking the inverse z-transform of the new transfer function. The formulas with an example are given below, where ω_L and ω_H are the low- and high-frequency cutoffs, respectively, normalized between 0 and π, that is, $\omega_L = 2\pi f_L/\text{sample rate}$, where f_L is the cutoff frequency in hertz; ϕ_L is the normalized cutoff frequency of the low-pass prototype.[5]

High Pass

$$z' = -\frac{1 + AZ}{Z + A} \tag{38}$$

where Z is the pole or zero to be transformed and z' is the transformed pole or zero and

$$A = -\frac{\cos[(\omega_H + \phi_L)/2]}{\cos[(\omega_H - \phi_L)/2]} \tag{39}$$

Bandpass

The bandpass filter has two transitions: a rising edge and a falling edge. For this reason, we need twice as many coefficients for the same approximate transition width as the prototype filter. Thus the order of these polynomials will be twice the order of polynomials in the

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

prototype. Each pole from the prototype will transform into a pair of poles (z_1' and z_2'). Likewise, each zero will transform into a pair of zeros:

$$z_1' = \frac{(A + AZ) + \sqrt{(A + AZ)^2 - 4(Z + B)(BZ + 1)}}{2(Z + B)} \tag{40}$$

$$z_2' = \frac{(A + AZ) - \sqrt{(A + AZ)^2 - 4(Z + B)(BZ + 1)}}{2(Z + B)} \tag{41}$$

$$A = \frac{2CD}{D + 1} \tag{42}$$

$$B = \frac{D - 1}{D + 1} \tag{43}$$

$$C = \frac{\cos[(\omega_H + \omega_L)/2]}{\cos[(\omega_H - \omega_L)/2]} \tag{44}$$

$$D = \cot\left(\frac{\omega_H - \omega_L}{2}\right) \tan\frac{\phi_L}{2} \tag{45}$$

Band Reject

$$z_1' = \frac{(AZ - A) + \sqrt{(AZ - A)^2 - 4(Z - B)(BZ - 1)}}{2(Z - B)} \tag{46}$$

$$z_2' = \frac{(AZ - A) - \sqrt{(AZ - A)^2 - 4(Z - B)(BZ - 1)}}{2(Z - B)} \tag{47}$$

$$C = \frac{\cos[(\omega_H + \omega_L)/2]}{\cos[(\omega_H - \omega_L)/2]} \tag{48}$$

$$D = \tan\left(\frac{\omega_H - \omega_L}{2}\right) \tan\frac{\phi_L}{2} \tag{49}$$

As an example, design a Tchebyshev (type I) bandpass filter for a system sampled at 100 Hz with a low cutoff frequency of 20 Hz and a high cutoff frequency of 35 Hz.

Step 1. We can use the poles and zeros designed in Section 3.3 as the low-pass prototype.

Step 2. Convert the low and high cutoff frequencies to values normalized between 0 and π, where π corresponds to the Nyquist frequency:

$$\omega_L = 2\pi \cdot \tfrac{20}{100} = 1.2566 \qquad \omega_H = 2\pi \cdot \tfrac{35}{100} = 2.1991 \tag{50}$$

Step 3. Map the poles from the prototype using Eqs. (40) and (41):

$$p_1 \rightarrow \begin{cases} -0.6095 - j0.9105 \\ 0.2940 + j1.0722 \end{cases} \tag{51}$$

$$p_2 \rightarrow \begin{cases} -0.2302 - j1.2527 \\ -0.2302 + j1.2527 \end{cases} \tag{52}$$

$$p_3 \rightarrow \begin{cases} -0.6095 + j0.9105 \\ 0.2940 - j1.0722 \end{cases} \tag{53}$$

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

Then map the zeros from the prototype using Eqs. (40) and (41):

$$\zeta_1 \rightarrow \begin{cases} -1 \\ 1 \end{cases} \tag{54}$$

$$\zeta_2 \rightarrow \begin{cases} -1 \\ 1 \end{cases} \tag{55}$$

$$\zeta_3 \rightarrow \begin{cases} -1 \\ 1 \end{cases} \tag{56}$$

Step 4. Expand the numerator and denominator polynomials:

$$\frac{(z - \zeta_1) \cdots (z - \zeta_6)}{(z - pz_1) \cdots (z - pz_6)} = \frac{1 - 3z^{-2} + 3z^{-4} - z^{-6}}{(1 + 1.091z^{-1} + 3.632z^{-2} + 2.616z^{-3} + 4.642z^{-4} + 1.982z^{-5} + 2.407z^{-6}} \tag{57}$$

Step 5. Normalize the transfer function so that it will have unity gain in the passband. For this, we estimate

$$M = \max_{0 \le \omega \le \pi} |H(e^{j\omega})] \tag{58}$$

Then compute

$$H_{\text{normalized}}(z) = \frac{1}{M} H(z) \tag{59}$$

For this example, $M = 14.45$.

Step 6. Realize the difference equation from inverse z-transformation of the derived transfer function:

$$\begin{aligned} y_n = {} & -1.091y_{n-1} - 3.632y_{n-2} - 2.616y_{n-3} - 4.642y_{n-4} - 1.982y_{n-5} \\ & -2.407y_{n-6} + 0.0692 * x_n - 0.2076x_{n-2} + 0.2076x_{n-4} - 0.0692x_{n-6} \end{aligned} \tag{60}$$

3.5 Frequency-Domain Filtering

It is possible to filter the data in the frequency domain. The method involves the use of the Fourier transform. We Fourier transform the data, multiply by the desired frequency response, then inverse Fourier transform the data. This is similar to the FIR filters discussed earlier. Deriving the FIR coefficients by performing a discrete cosine transform (DCT) of the desired frequency response and then convolving the coefficients with the data is equivalent to filtering the data in the frequency domain. One difference, however, is that the frequency domain filtering is generally done on blocks of data and not on streaming data, as is done in the time domain, which can be of concern when processing highly nonstationary data with abrupt transients. The inverse Fourier transform is

$$\mathfrak{F}^{-1}\{F(\omega)\} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(\omega) e^{-j\omega t}\, d\omega \tag{61}$$

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

4 STABILITY AND PHASE ANALYSIS

4.1 Stability Analysis

Consider a transfer function

$$H(s) = \frac{1}{s - p} \tag{62}$$

where the pole p is a complex number $\alpha + j\beta$. The inverse Laplace transform of this function is $e^{\alpha t}[\cos(\beta t) + j\ \sin(\beta t)]$. This function is bounded as $t \to \infty$ if and only if $\alpha < 0$. From this, we can determine the stability of a function by inspecting the real components of all poles of a given transfer function. The procedure for a rational function (a ratio of polynomials) would be to factor the polynomials in the denominator and inspect to ensure that the real components to all of the poles are less than zero. Suppose

$$\begin{aligned} H(s) &= \frac{a_0 + a_1 s + \cdots + a_m s^m}{b_0 + b_1 s + \cdots + b_n s^2} \\ &= \frac{(s - \zeta_0)(s - \zeta_1) \cdots (s - \zeta_m)}{(s - p_0)(s - p_1) \cdots (s - p_n)} \end{aligned} \tag{63}$$

In a similar way, by inspection of the S-to-z transformation $z = e^s$, we see that the entire left half of the plane in the S-domain maps inside the unit circle in the z-domain. For this reason, we analyze the stability of systems in the z-domain by inspecting the poles of the transfer function. The system is stable if the norm of all poles is less than 1.

4.2 Phase Analysis

While processing the data in real time, our filters must act on the signal history. For this reason, there will always be some delay in the output of our process. Worse, certain filters will delay some frequency components by more or less than other frequency components. This results in a phase distortion of the filter. For a certain class of FIR filters, it is possible to design filters that shift each frequency component by a time delay in proportion to the frequency. In this way, all frequency components are shifted by an equal time delay. Though it is possible to design certain non-real-time, noncausal IIR filters that are phase shift distortionless, in general, IIR filters will produce some phase shift distortion. We can determine the actual phase shift for each frequency component by computing

Table 3 Comparison of FIR and IIR Characteristics

	FIR	IIR
Run time efficiency	Less efficient; requires high-order filter.	Higher efficiency; usually possible to achieve a desired design specification in fewer computations
Stability	Always stable	Stable if all poles are inside the unit circle
Phase shift distortion	Can be designed to be phase shift distortionless	Generally distorts phase
Ease of design	Simpler design process, usually involving Fourier transforms or solving linear systems	Design is more complex, involving special functions or solving nonlinear systems

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

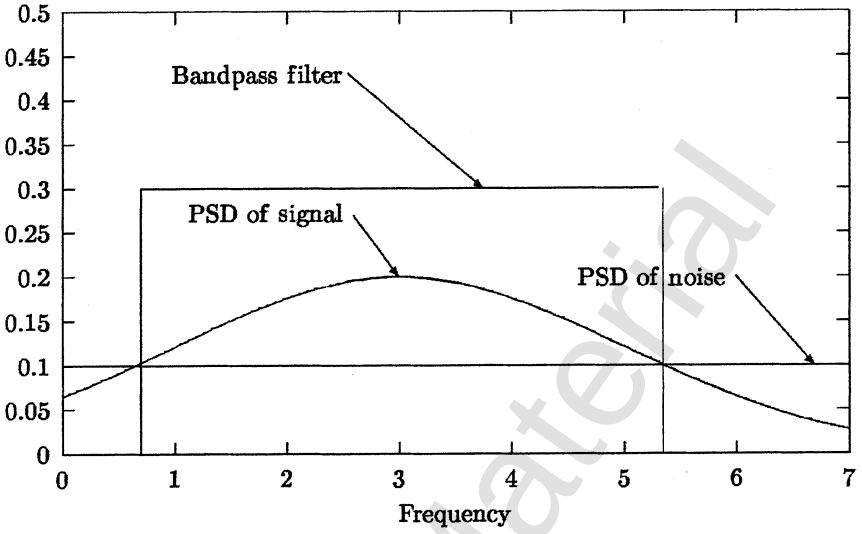


Figure 11 Use of bandpass filter for discriminating signal from noise.

$$\arg H(j\omega) = \angle H_{\text{real}} + jH_{\text{imag}} = \tan^{-1}\frac{H_{\text{imag}}}{H_{\text{real}}}$$

The arctan will produce the principal value of the phase shift, not necessarily the cumulative phase shift, since the arctan function produces principal value $-\pi \leq \tan^{-1}(\phi) \leq \pi$. It is possible to recover the accumulated phase shift by factoring the rational function into its binomial parts, then expressing this in exponential form as a summation. One can then

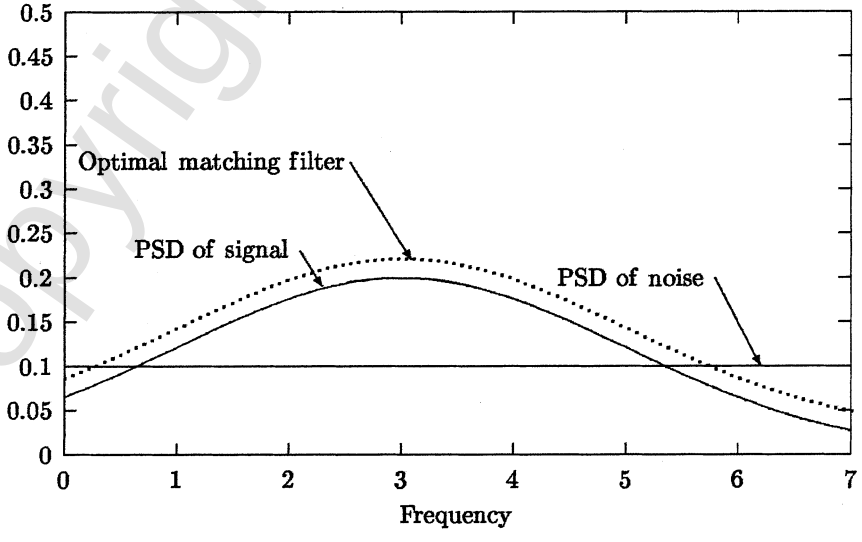


Figure 12 Improved matching filter for better discrimination of signal from noise.

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com

determine the principal angle on each part and the accumulated phase shift by summing the parts.

4.3 Comparison of FIR and IIR Filters

There are various factors when deciding on a particular filter for a given application. Table 3 summarizes these.

5 EXTRACTING SIGNAL FROM NOISE

The PSD of white noise is uniformly distributed over all frequencies. Therefore, it is possible to detect the PSD signature of a signal corrupted by white noise by inspecting spectral components that rise above some baseline. From this we can design a matching filter to optimally extract the signal from the noise. Figures 11 and 12 illustrate this procedure.

REFERENCES

1. W. Press, B. Flannery, A. Teukolsky, and W. Vetterline, *Numerical Recipes in C,* Cambridge University Press, Cambridge, UK, 1988.
2. T. Parks and C. Burrus, *Digital Filter Design,* Wiley, New York, 1987.
3. M. Vlček and R. Unbehauen, "Analytical Solutions for Design of IIR Equiripple Filters," *IEEE Transactions on Acoustics, Speech, and Signal Processing* **37**(10) (October 1989).
4. S. Marple, *Digital Spectral Analysis with Applications,* Prentice-Hall, Englewood Cliffs, NJ, 1987.
5. A. Oppenheim and R. Schafer, *Digital Signal Processing,* Prentice-Hall, Englewood Cliffs, NJ, 1975.

Copyrighted Material

Copyright © 2006 John Wiley & Sons Retrieved from: www.knovel.com