

Chapter 5

IIR Digital Filter Design



Filtering is an important aspect of signal processing. It allows desired frequency components of a signal to pass through the system without distortion and suppresses the undesired frequency components. One of the most important steps in the design of a digital filter is to obtain a realizable transfer function $H(z)$, satisfying the given frequency response specifications. In the case of the design of an IIR filter, it is required to confirm that $H(z)$ is stable. The most common technique used in designing IIR digital filters involves first designing an analog prototype lowpass filter and then transforming the prototype to a digital filter. In this chapter, the design of analog lowpass filters is first described. Second, frequency transformations for transforming analog lowpass filter into bandpass, bandstop, or highpass analog filters are considered. Next, the design of IIR filters is discussed and illustrated with numerical examples. Further, the design of IIR filters using MATLAB is demonstrated with a number of examples. Also, the design of IIR filters using graphical user interface MATLAB filter design SPTOOL is discussed and illustrated with examples. Finally, some application examples of IIR filters for audio processing are included.

5.1 Analog Lowpass Filter Design

A number of approximation techniques for the design of analog lowpass filters are well established [1–4]. The design of analog lowpass filter using Butterworth, Chebyshev I, Chebyshev II (inverse Chebyshev), and elliptic approximations is discussed in this section.

5.1.1 Filter Specifications

The specifications for an analog lowpass filter with tolerances are depicted in Fig. 5.1, where

Ω_p —passband edge frequency

Ω_s —stopband edge frequency

δ_p —peak ripple value in the passband

δ_s —peak ripple value in the stopband

Peak passband ripple in dB = $\alpha_p = -20 \log_{10}(1 - \delta_p)$ dB

Minimum stopband ripple in dB = $\alpha_s = -20 \log_{10}(\delta_s)$ dB

Peak ripple value in passband $\delta_p = 1 - 10^{-\alpha_p/20}$

Peak ripple value in stopband $\delta_s = 10^{-\alpha_s/20}$

5.1.2 Butterworth Analog Lowpass Filter

The magnitude-square response of an N th-order analog lowpass Butterworth filter is given by

$$|H_a(j\Omega)|^2 = \frac{1}{1 + (\Omega/\Omega_c)^{2N}} \tag{5.1}$$

Two parameters completely characterizing a Butterworth lowpass filter are Ω_c and N . These are determined from the specified band edges Ω_p and Ω_s , and peak

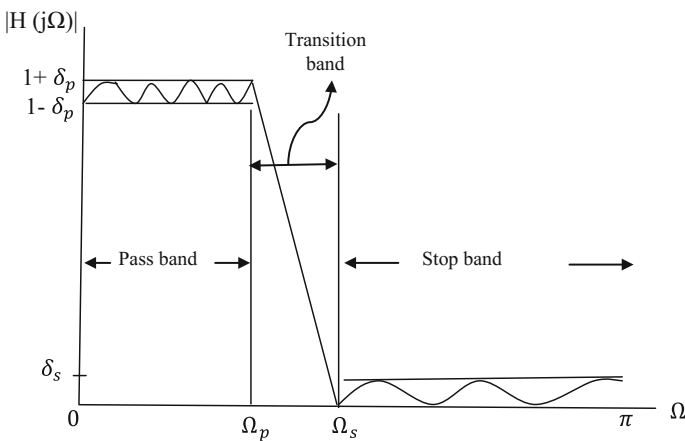


Fig. 5.1 Specifications of a lowpass analog filter

passband ripple α_p , and minimum stopband attenuation α_s . The first $(2N - 1)$ derivatives of $|H_a(j\Omega)|^2$ at $\Omega = 0$ are equal to zero. Thus, the Butterworth lowpass filter is said to have a maximally flat magnitude at $\Omega = 0$. The gain in dB is given by $10 \log_{10} |H_a(j\Omega)|^2$. At $\Omega = \Omega_c$, the gain is $10 \log_{10}(0.5) = -3$ dB; therefore, Ω_c is called the 3 dB cutoff frequency. The loss in dB in a Butterworth filter is given by

$$\alpha = 10 \log(1 + (\Omega/\Omega_c)^{2N}) \quad (5.2)$$

For $\Omega = \Omega_p$, the passband attenuation is given by

$$\alpha_p = 10 \log(1 + (\Omega_p/\Omega_c)^{2N}) \quad (5.3)$$

For $\Omega = \Omega_s$, the stopband attenuation is

$$\alpha_s = 10 \log(1 + (\Omega_s/\Omega_c)^{2N}) \quad (5.4)$$

Equations (5.3) and (5.4) can be rewritten as

$$(\Omega_p/\Omega_c)^{2N} = 10^{0.1\alpha_p} - 1 \quad (5.5)$$

$$(\Omega_s/\Omega_c)^{2N} = 10^{0.1\alpha_s} - 1 \quad (5.6)$$

From Eqs. (5.5) to (5.6), we obtain

$$(\Omega_s/\Omega_p) = \left(\frac{10^{0.1\alpha_s} - 1}{10^{0.1\alpha_p} - 1} \right)^{1/2N} \quad (5.7)$$

Equation (5.7) can be rewritten as

$$\log(\Omega_s/\Omega_p) = \frac{1}{2N} \log \left(\frac{10^{0.1\alpha_s} - 1}{10^{0.1\alpha_p} - 1} \right) \quad (5.8)$$

From Eq. (5.8), solving for N we get

$$N \geq \frac{\log \left(\frac{10^{0.1\alpha_s} - 1}{10^{0.1\alpha_p} - 1} \right)}{2 \log(\Omega_s/\Omega_p)} \quad (5.9)$$

Since the order N must be an integer, the value obtained is rounded up to the next higher integer. This value of N is used in either Eq. (5.5) or Eq. (5.6) to determine the 3-dB cutoff frequency Ω_c . In practice, Ω_c is determined by Eq. (5.6) that exactly satisfies stopband specification at Ω_c , while the passband specification is exceeded with a safe margin at Ω_p [2]. We know that $|H(j\Omega)|^2$ may be evaluated by letting $s = j\Omega$ in $H(s)H(-s)$, which may be expressed as

$$H(s)H(-s) = \frac{1}{1 + (-s^2/\Omega_c^2)^N} \quad (5.10)$$

If $\Omega_c = 1$, the magnitude response $|H_N(j\Omega)|$ is called the normalized magnitude response. Now, we have

$$1 + (-s^2)^N = \prod_{k=1}^{2N} (s - s_k) \quad (5.11)$$

where

$$s_k = \begin{cases} e^{j(2k-1)\pi/(2N)} & \text{for } n \text{ even} \\ e^{j(k-1)\pi/N} & \text{for } n \text{ odd} \end{cases} \quad (5.12)$$

Since $|s_k| = 1$, we can conclude that there are $2N$ poles placed on the unit circle in the s -plane. The normalized transfer function can be formed as

$$H_N(s) = \frac{1}{\prod_{l=1}^N (s - p_l)} \quad (5.13)$$

where p_l for $l = 1, 2, \dots, N$ are the left half s -plane poles. The complex poles occur in conjugate pairs.

For example, in the case of $N = 2$, from Eq. (5.12), we have

$$s_k = \cos\left(\frac{(2k-1)\pi}{4}\right) + j \sin\left(\frac{(2k-1)\pi}{4}\right), \quad k = 1, \dots, 2$$

The poles in the left half of the s -plane are

$$s_2 = -\frac{1}{\sqrt{2}} + \frac{j}{\sqrt{2}}; \quad s_3 = -\frac{1}{\sqrt{2}} - \frac{j}{\sqrt{2}}$$

Hence,

$$p_1 = -\frac{1}{\sqrt{2}} + \frac{j}{\sqrt{2}}; \quad p_2 = -\frac{1}{\sqrt{2}} - \frac{j}{\sqrt{2}}$$

and

$$H_N(s) = \frac{1}{s^2 + \sqrt{2}s + 1}$$

In the case of $N = 3$,

$$s_k = \cos\left(\frac{(k-1)\pi}{3}\right) + j \sin\left(\frac{(k-1)\pi}{3}\right), \quad k = 1, \dots, 2$$

The left half of s -plane poles are

$$s_3 = -\frac{1}{2} + \frac{j\sqrt{3}}{2}; \quad s_4 = -1; \quad s_5 = -\frac{1}{2} - \frac{j\sqrt{3}}{2}.$$

Hence

$$p_1 = -\frac{1}{2} + \frac{j\sqrt{3}}{2}; \quad p_2 = -1; \quad p_3 = -\frac{1}{2} - \frac{j\sqrt{3}}{2}.$$

and

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

The following MATLAB Program 5.1 can be used to obtain the Butterworth normalized transfer function for various values of N .

Program 5.1 Analog Butterworth lowpass filter normalized transfer function

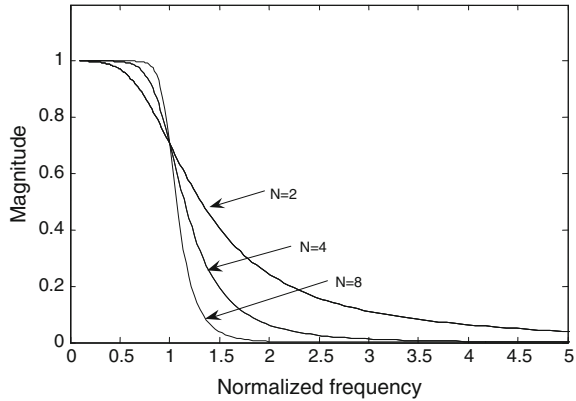
```
N = input('enter order of the filter');
[z,p,k] = buttap(N)% determines poles and zeros
disp('Poles are at');disp(p);
[num,den] = zp2tf(z,p,k);
%Print coefficients in powers of s
disp('Numerator polynomial');disp(num);
disp('Denominator polynomial');disp(den);
sos = zp2sos(z,p,k);%determines coefficients of second order sections
```

The normalized Butterworth polynomials generated from the above program for typical values of N are tabulated in Table 5.1.

Table 5.1 List of normalized Butterworth polynomials

N	Denominator of $H_N(s)$
1	$s + 1$
2	$s^2 + \sqrt{2}s + 1$
3	$(s + 1)(s^2 + s + 1)$
4	$(s^2 + 0.76537s + 1)(s^2 + 1.8477s + 1)$
5	$(s + 1)(s^2 + 0.61803s + 1)(s^2 + 1.61803s + 1)$
6	$(s^2 + 1.931855s + 1)(s^2 + \sqrt{2}s + 1)(s^2 + 0.51764s + 1)$
7	$(s + 1)(s^2 + 1.80194s + 1)(s^2 + 1.247s + 1)(s^2 + 0.445s + 1)$

Fig. 5.2 Magnitude response of typical Butterworth lowpass filter



The magnitude response of the normalized Butterworth lowpass filter for some typical values of N is shown in Fig. 5.2. From this figure, it can be seen that the response monotonically decreases in both the passband and the stopband as Ω increases. As the filter order N increases, the magnitude responses in both the passband and the stopband are improved with a corresponding decrease in the transition width. Since the normalized transfer function corresponds to $\Omega_c = 1$, the transfer function of the lowpass filter corresponding to the actual Ω_c can be obtained by replacing s by (s/Ω_c) in the normalized transfer function.

Example 5.1 Design a Butterworth analog lowpass filter with 1 dB passband ripple, passband edge frequency $\Omega_p = 2000\pi$ rad/s, stopband edge frequency $\Omega_s = 10,000\pi$ rad/s, and a minimum stopband ripple of 40 dB.

Solution Since $\alpha_s = 40$ dB, $\alpha_p = 1$ dB, $\Omega_p = 2000\pi$, and $\Omega_s = 10,000\pi$

$$\log\left(\frac{10^{0.1\alpha_s} - 1}{10^{0.1\alpha_p} - 1}\right) = \log\left(\frac{10^4 - 1}{10^{0.1} - 1}\right) = 4.5868.$$

Hence from Eq. (5.9),

$$N \geq \frac{\log\left(\frac{10^4 - 1}{10^{0.1} - 1}\right)}{2 \log(5/1)} = \frac{4.5868}{1.3979} = 3.2811$$

Since the order must be an integer, we choose $N = 4$.

The normalized lowpass Butterworth filter for $N = 4$ can be formulated as

$$H_N(s) = \frac{1}{(s^2 + 0.76537s + 1)(s^2 + 1.8477s + 1)}$$

From Eq. (5.6), we have

$$\Omega_c = \frac{\Omega_s}{(10^4 - 1)^{1/2N}} = \frac{10,000\pi}{(10^4 - 1)^{1/8}} = 9935$$

The transfer function for $\Omega_c = 9935$ can be obtained by replacing s by $\left(\frac{s}{\Omega_c}\right) = \left(\frac{s}{9935}\right)$ in $H_N(s)$

$$\begin{aligned} H_a(s) &= \frac{1}{\left(\frac{s}{9935}\right)^2 + 0.76537\left(\frac{s}{9935}\right) + 1} \times \frac{1}{\left(\frac{s}{9935}\right)^2 + 1.8477\left(\frac{s}{9935}\right) + 1} \\ &= \frac{9.7425 \times 10^{15}}{(s^2 + 7.604 \times 10^3 s + 9.8704225 \times 10^7)(s^2 + 1.8357 \times 10^4 s + 9.8704225 \times 10^7)} \end{aligned}$$

5.1.3 Chebyshev Analog Lowpass Filter

Type 1 Chebyshev lowpass filter

The magnitude-square response of an N th-order analog lowpass Type 1 Chebyshev filter is given by

$$|H(\Omega)|^2 = \frac{1}{1 + \varepsilon^2 T_N^2(\Omega/\Omega_p)} \quad (5.14)$$

where $T_N(\Omega)$ is the Chebyshev polynomial of order N

$$T_N(\Omega) = \begin{cases} \cos(N \cos^{-1} \Omega), & |\Omega| \leq 1 \\ \cosh(N \cosh^{-1} \Omega), & |\Omega| > 1 \end{cases} \quad (5.15)$$

The loss in dB in a Type 1 Chebyshev filter is given by

$$\alpha = 10 \log(1 + \varepsilon^2 T_N^2(\Omega/\Omega_p)) \quad (5.16)$$

For $\Omega = \Omega_p$, $T_N(\Omega) = 1$, and the passband attenuation is given by

$$\alpha_p = 10 \log(1 + \varepsilon^2) \quad (5.17)$$

From Eq. (5.17), ε can be obtained as

$$\varepsilon = \sqrt{10^{0.1\alpha_p} - 1} \quad (5.18)$$

For $\Omega = \Omega_s$, the stopband attenuation is

$$\alpha_s = 10 \log \left(1 + \epsilon^2 T_n^2 \left(\frac{\Omega_s}{\Omega_p} \right) \right) \quad (5.19)$$

Since $(\Omega_s/\Omega_p) > 1$, the above equation can be written as

$$\alpha_s = 10 \log [1 + \epsilon^2 \cosh^2 (N \cosh^{-1} (\Omega_s/\Omega_p))] \quad (5.20)$$

Substituting Eq. (5.18) for ϵ in the above equation and solving for N , we get

$$N \geq \frac{\cosh^{-1} \sqrt{\frac{10^{0.1\alpha_s} - 1}{10^{0.1\alpha_p} - 1}}}{\cosh^{-1} (\Omega_s/\Omega_p)} \quad (5.21)$$

We choose N to be the lowest integer satisfying (5.21). In determining N using the above equation, it is convenient to evaluate $\cosh^{-1}(x)$ by applying the identity $\cosh^{-1}(x) = \ln(x + \sqrt{x^2 - 1})$.

The poles of the normalized Type 1 Chebyshev filter transfer function lie on an ellipse in the s -plane and are given by [5]

$$x_k = -\sinh \left\{ \frac{1}{N} \sinh^{-1} \left(\frac{1}{\epsilon} \right) \right\} \cdot \sin \left\{ \frac{(2k-1)\pi}{2N} \right\}, \quad \text{for } k = 1, 2, \dots, N \quad (5.22)$$

$$y_k = \cosh \left\{ \frac{1}{N} \sinh^{-1} \left(\frac{1}{\epsilon} \right) \right\} \cdot \cos \left\{ \frac{(2k-1)\pi}{2N} \right\}, \quad \text{for } k = 1, 2, \dots, N \quad (5.23)$$

Also, the normalized transfer function is given by

$$H_N(s) = \frac{H_0}{\prod_k (s - p_k)} \quad (5.24)$$

where

$$\begin{aligned} p_k = & -\sinh \left\{ \frac{1}{N} \sinh^{-1} \left(\frac{1}{\epsilon} \right) \right\} \cdot \sin \left\{ \frac{(2k-1)\pi}{2N} \right\} \\ & + j \cosh \left\{ \frac{1}{N} \sinh^{-1} \left(\frac{1}{\epsilon} \right) \right\} \cdot \cos \left\{ \frac{(2k-1)\pi}{2N} \right\} \end{aligned} \quad (5.25a)$$

and

$$H_0 = \frac{1}{2^{N-1}} \frac{1}{\epsilon} \quad (5.25b)$$

As an illustration, consider the case of $N = 2$ with a passband ripple of 1 dB. From Eq. (5.18), we have

$$\frac{1}{\varepsilon} = \frac{1}{\sqrt{10^{0.1\alpha_p} - 1}} = 1.965227$$

Hence

$$\sinh^{-1}\left(\frac{1}{\varepsilon}\right) = \sinh^{-1}(1.965227) = 1.428$$

Therefore, from (5.25a), the poles of the normalized Chebyshev transfer function are given by

$$p_k = -\sinh(0.714)\sin\left\{\frac{(2k-1)\pi}{4}\right\} + j\cosh(0.714)\cos\left\{\frac{(2k-1)\pi}{4}\right\}, \quad k = 1, 2$$

Hence

$$p_1 = -0.54887 + j0.89513, \quad p_2 = -0.54887 - j0.89513$$

Also, from (5.25b), we have

$$H_0 = \frac{1}{2}(1.965227) = 0.98261$$

Thus for $N = 2$, with a passband ripple of 1 dB, the normalized Chebyshev transfer function is

$$H_N(s) = \frac{0.98261}{(s-p_1)(s-p_2)} = \frac{0.98261}{(s^2 + 1.098s + 1.103)}$$

Similarly for $N = 3$, for a passband ripple of 1 dB, we have

$$p_k = -\sinh(1.428/3)\sin\frac{(2k-1)\pi}{6} + j\cosh(1.428/3)\cos\frac{(2k-1)\pi}{6}, \quad k = 1, 2, 3$$

Thus,

$$p_1 = -0.24709 + j0.96600; \quad p_2 = -0.49417; \quad p_3 = -0.24709 - j0.966.$$

Also, from (5.25b),

$$H_0 = \frac{1}{4}(1.965227) = 0.49131$$

Hence, the normalized transfer function of Type 1 Chebyshev lowpass filter for $N = 3$ is given by

$$\begin{aligned} H_N(s) &= \frac{0.49131}{(s - p_1)(s - p_2)(s - p_3)} \\ &= \frac{0.49131}{(s^3 + 0.988s^2 + 1.238s + 0.49131)} \end{aligned}$$

The following MATLAB Program 5.2 can be used to form the Type 1 Chebyshev normalized transfer function for a given order and passband ripple.

Program 5.2 Analog Type 1 Chebyshev lowpass filter normalized transfer function

```
N = input('enter order of the filter');
Rp = input('enter passband ripple in dB');
[z,p,k] = cheblap(N,Rp)% determines poles and zeros
disp('Poles are at');disp(p);
[num,den] = zp2tf(z,p,k);
%Print coefficients in powers of s
disp('Numerator polynomial');disp(num);
disp('Denominator polynomial');disp(den);
```

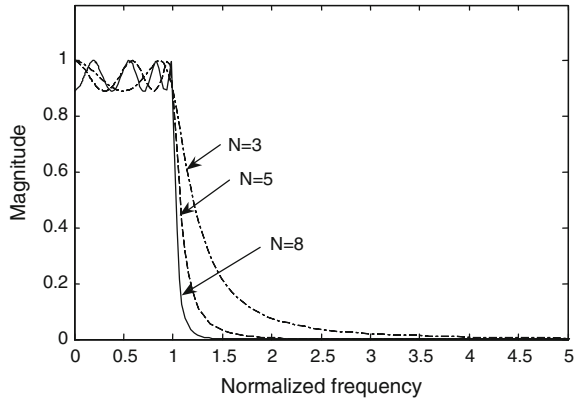
The normalized Type 1 Chebyshev polynomials generated from the above program for typical values of N and passband ripple of 1 dB are tabulated in Table 5.2.

The typical magnitude responses of a Type 1 Chebyshev lowpass filter for $N = 3, 5,$ and 8 with 1 dB passband ripple are shown in Fig. 5.3. From this figure, it

Table 5.2 List of normalized Type 1 Chebyshev transfer functions for passband ripple = 1 dB

N	Denominator of $H_N(s)$	H_0
1	$s + 1.9652$	1.9652
2	$s^2 + 1.0977s + 1.1025$	0.98261
3	$s^3 + 0.98834s^2 + 1.2384s + 0.49131$	0.49131
4	$s^4 + 0.95281s^3 + 1.4539s^2 + 0.74262s + 0.27563$	0.24565
5	$s^5 + 0.93682s^4 + 1.6888s^3 + 0.9744s^2 + 0.58053s + 0.12283$	0.12283

Fig. 5.3 Magnitude response of typical Type 1 Chebyshev lowpass filter with 1 dB passband ripple



is seen that Type 1 Chebyshev lowpass filter exhibits equiripple in the passband with a monotonic decrease in the stopband.

Example 5.2 Design a Type 1 Chebyshev analog lowpass filter for the specifications given in Example 5.1.

Solution Since $\alpha_s = 40$ dB, $\alpha_p = 1$ dB, $\Omega_p = 2000\pi$, and $\Omega_s = 10000\pi$

$$\cosh^{-1} \sqrt{\frac{10^{0.1\alpha_s} - 1}{10^{0.1\alpha_p} - 1}} = \cosh^{-1} \sqrt{\frac{10^4 - 1}{10^{0.1} - 1}} = \cosh^{-1}(196.52)$$

$$\cosh^{-1}(\Omega_s/\Omega_p) = \cosh^{-1}(5) = 2.2924$$

$$N \geq \frac{\cosh^{-1} \sqrt{\frac{10^4 - 1}{10^{0.1} - 1}}}{\cosh^{-1}(5)} = 2.6059$$

Since the order of the filter must be an integer, we choose the next higher integer value 3 for N . The normalized Type 1 Chebyshev lowpass filter for $N = 3$ with a passband ripple of 1 dB is given from Table 5.2 as

$$H_N(s) = \frac{0.49131}{s^3 + 0.988s^2 + 1.238s + 0.49131}$$

The transfer function for $\Omega_p = 2000\pi$ is obtained by substituting $s = (s/\Omega_p) = (s/2000\pi)$ in $H_N(s)$

$$H_a(s) = \frac{0.49131}{\left(\frac{s}{2000\pi}\right)^3 + 0.988\left(\frac{s}{2000\pi}\right)^2 + 1.238\left(\frac{s}{2000\pi}\right) + 0.49131}$$

$$= \frac{1.2187 \times 10^{11}}{s^3 + 6.2099 \times 10^3 s^2 + 4.889 \times 10^7 s + 1.2187 \times 10^{11}}$$

Type 2 Chebyshev Filter

The squared-magnitude response of Type 2 Chebyshev lowpass filter, which is also known as the inverse Chebyshev filter, is given by

$$|H(\Omega)|^2 = \frac{1}{1 + \varepsilon^2 \left(\frac{T_N^2(\Omega_s/\Omega_p)}{T_N^2(\Omega_s/\Omega)} \right)} \quad (5.26)$$

The order N can be determined using Eq. (5.21). The Type 2 Chebyshev filter has both poles and zeros, and the zeros are on the $j\Omega$ axis. The normalized Type 2 Chebyshev lowpass filter, or the normalized inverse Chebyshev filter (normalized to $\Omega_s = 1$), may be formed as [4]

$$H_N(s) = \frac{H_0 \sum_k (s - z_k)}{\sum_k (s - p_k)}, \quad k = 1, 2, \dots, N \quad (5.27)$$

where

$$z_k = j \frac{1}{\cos \frac{(2k-1)\pi}{N}} \quad \text{for } k = 1, 2, \dots, N \quad (5.28a)$$

$$p_k = \frac{\sigma_k}{\sigma_k^2 + \Omega_k^2} + j \frac{\Omega_k}{\sigma_k^2 + \Omega_k^2} \quad \text{for } k = 1, 2, \dots, N \quad (5.28b)$$

$$\sigma_k = -\sinh \left\{ \frac{1}{N} \sinh^{-1} \left(\frac{1}{\delta_s} \right) \right\} \sin \frac{(2k-1)\pi}{2N} \quad \text{for } k = 1, 2, \dots, N \quad (5.28c)$$

$$\Omega_k = \cosh \left\{ \frac{1}{N} \sinh^{-1} \left(\frac{1}{\delta_s} \right) \right\} \cos \frac{(2k-1)\pi}{2N} \quad \text{for } k = 1, 2, \dots, N \quad (5.28d)$$

$$\delta_s = \frac{1}{\sqrt{10^{0.1\alpha_s} - 1}} \quad (5.28e)$$

$$H_0 = \frac{\prod_k (-z_k)}{\prod_k (-p_k)} \quad (5.28f)$$

For example, if we consider $N = 3$ with a stopband ripple of 40 dB, then from (5.28e),

$$\frac{1}{\delta_s} = \sqrt{10^{0.1\alpha_s} - 1} = \sqrt{10^4 - 1} = 99.995$$

Hence,

$$\sinh^{-1}\left(\frac{1}{\delta_s}\right) = 5.28829$$

Using (5.28c) and (5.28d), we have

$$\sigma_k = -\sinh(5.28829/3) \sin \frac{(2k-1)\pi}{6} \quad \text{for } k = 1, 2, 3$$

$$\Omega_k = \cosh(5.28829/3) \cos \frac{(2k-1)\pi}{6} \quad \text{for } k = 1, 2, 3$$

Hence,

$$\sigma_1 = -1.41927, \sigma_2 = -2.83854, \sigma_3 = -1.41927$$

$$\Omega_1 = -2.60387, \Omega_2 = -2.83854, \Omega_3 = 2.60387$$

Thus, from (5.28b), the poles are

$$p_1 = -0.16115 + j0.29593, p_2 = -0.3523, p_3 = -0.16115 + j0.29593$$

Also, using (5.28a), the zeros are given by

$$z_1 = -j(2/\sqrt{3}), z_2 = j(2/\sqrt{3})$$

Finally, from (5.28f),

$$H_0 = 0.03$$

Therefore, the normalized Type 2 Chebyshev lowpass filter for $N = 3$ with a stopband ripple of 40 dB is given by

$$\begin{aligned} H_N(s) &= \frac{0.03(s - z_1)(s - z_2)}{(s - p_1)(s - p_2)(s - p_3)} \\ &= \frac{0.03(s^2 + 1.3333)}{(s^3 + 0.6746s^2 + 0.22709s + 0.04)} \end{aligned}$$

The following MATLAB Program 5.3 can be used to form the Type 2 Chebyshev normalized transfer function for a given order and stopband ripple.

Program 5.3 Analog Type 2 Chebyshev lowpass filter normalized transfer function

```

N = input('enter order of the filter');
Rs = input('enter stopband attenuation in dB');
[z,p,k] = cheb2ap(N,Rs);% determines poles and zeros
disp('Poles are at');disp(p);
[num,den] = zp2tf(z,p,k);
%Print coefficients in powers of s
disp('Numerator polynomial');disp(num);
disp('Denominator polynomial');disp(den);
    
```

The normalized Type 2 Chebyshev transfer functions generated from the above program for typical values of N with a stopband ripple of 40 dB are tabulated in Table 5.3.

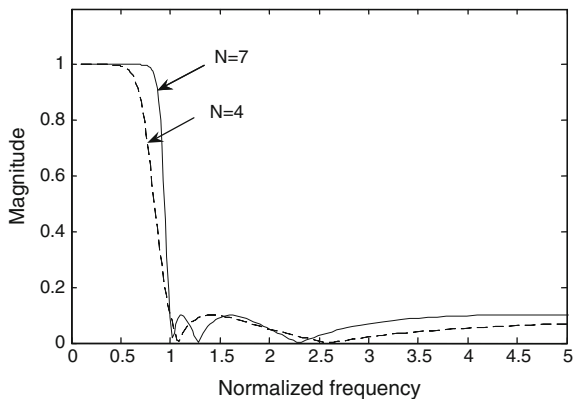
The typical magnitude response of a Type 2 Chebyshev lowpass filter for $N = 2$ and 7 with 20 dB stopband ripple is shown in Fig. 5.4. From this figure, it is seen that Type 2 Chebyshev lowpass filter exhibits monotonicity in the passband and equiripple in the stopband.

Example 5.3 Design a Type 2 Chebyshev lowpass filter for the specifications given in Example 5.1.

Table 5.3 List of normalized Type 2 Chebyshev transfer functions for stopband ripple = 40 dB

Order N	$H_N(s)$
1	$\frac{0.01}{s + 0.01}$
2	$\frac{0.01s^2 + 0.02}{s^2 + 0.199s + 0.02}$
3	$\frac{0.03s^2 + 0.04}{s^3 + 0.6746s^2 + 0.2271s + 0.04}$
4	$\frac{0.01s^4 + 0.08s^2 + 0.08}{s^4 + 1.35s^3 + 0.9139s^2 + 0.3653s + 0.08}$
5	$\frac{0.05s^4 + 0.2s^2 + 0.16}{s^5 + 2.1492s^4 + 2.3083s^3 + 1.5501s^2 + 0.6573s + 0.16}$
6	$\frac{0.01s^6 + 0.18s^4 + 0.48s^2 + 0.32}{s^6 + 3.0166s^5 + 4.5519s^4 + 4.3819s^3 + 2.8798s^2 + 1.2393s + 0.32}$

Fig. 5.4 Magnitude response of typical Type 2 Chebyshev lowpass filter with 20 dB stopband ripple



Solution The order N is chosen as 3, as in Example 5.2, since the equation for order finding is the same for both Type 1 and Type 2 Chebyshev filters. The normalized Type 2 Chebyshev lowpass filter for $N = 3$ with a stopband ripple of 40 dB has already been found earlier and is given by

$$H_N(s) = \frac{0.03(s^2 + 1.3333)}{(s^3 + 0.6746s^2 + 0.2271s + 0.04)}$$

For $\Omega_s = 10,000\pi$, the corresponding transfer function can be obtained by substituting $s = (s/\Omega_s) = (s/10,000\pi)$ in the above expression for $H_N(s)$. Thus, the required filter transfer function is

$$\begin{aligned} H_a(s) &= \frac{0.03\left(\frac{s}{10,000\pi}\right)^2 + 0.04}{\left(\frac{s}{10,000\pi}\right)^3 + 0.6746\left(\frac{s}{10,000\pi}\right)^2 + 0.22709\left(\frac{s}{10,000\pi}\right) + 0.04} \\ &= \frac{9.4252 \times 10^2 s^2 + 1.2403 \times 10^{12}}{s^3 + 2.1193 \times 10^4 s^2 + 2.2413 \times 10^8 s + 1.2403 \times 10^{12}} \end{aligned}$$

5.1.4 Elliptic Analog Lowpass Filter

The square-magnitude response of an elliptic lowpass filter is given by

$$|H_a(j\Omega)|^2 = \frac{1}{1 + \varepsilon^2 U_N(\Omega/\Omega_p)} \quad (5.29)$$

where $U_N(x)$ is the Jacobian elliptic function of order N and ε is a parameter related to the passband ripple. In an elliptic filter, a constant k , called the selectivity factor, representing the sharpness of the transition region is defined as

$$k = \frac{\Omega_p}{\Omega_s} \quad (5.30)$$

A large value of k represents a wide transition band, while a small value indicates a narrow transition band.

For a given set of Ω_p , Ω_s , α_p and α_s , the filter order can be estimated using the formula [5]

$$N \cong \frac{\log\left(16 \times \frac{10^{0.1\alpha_s} - 1}{10^{0.1\alpha_p} - 1}\right)}{\log_{10}(1/\rho)} \quad (5.31)$$

where ρ can be computed using

$$\rho_0 = \frac{1 - \sqrt{k'}}{2(1 + \sqrt{k'})} \tag{5.32}$$

$$k' = \sqrt{1 - k^2} \tag{5.33}$$

$$\rho = \rho_0 + 2(\rho_0)^5 + 15(\rho_0)^9 + 150(\rho_0)^{13} \tag{5.34}$$

The following MATLAB Program 5.4 can be used to form the elliptic normalized transfer function for given filter order, and passband ripple and stopband attenuation. The normalized passband edge frequency is set to 1.

Program 5.4 Analog elliptic lowpass filter normalized transfer function

```
N = input('enter order of the filter');
Rp = input('enter passband ripple in dB');
Rs = input('enter stopband attenuation in dB');
[z,p,k] = ellipap(N,Rp,Rs)% determines poles and zeros
disp('Poles are at');disp(p);
[num,den] = zp2tf(z,p,k);
%Print coefficients in powers of s
disp('Numerator polynomial');disp(num);
disp('Denominator polynomial');disp(den);
```

The normalized elliptic transfer functions generated from the above program for typical values of N and stopband ripple of 40 dB are tabulated in Table 5.4.

The magnitude response of a typical elliptic lowpass filter is shown in Fig. 5.5, from which it can be seen that it exhibits equiripple in both the pass and the stopbands.

For more details on elliptic filters, readers may refer to [2, 4, 6].

Example 5.4 Design an elliptic analog lowpass filter for the specifications given in Example 5.1.

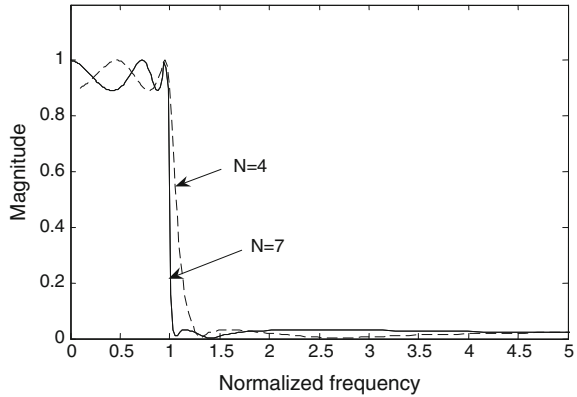
Solution

$$k = \frac{\Omega_p}{\Omega_s} = \frac{2000\pi}{10,000\pi} = 0.2$$

Table 5.4 List of normalized elliptic transfer functions for passband ripple = 1 dB and stopband ripple = 40 dB

Order N	$H_N(s)$
1	$\frac{1.9652}{s + 1.9652}$
2	$\frac{0.01s^2 + 0.9876}{s^2 + 1.0915s + 1.1081}$
3	$\frac{0.0692s^2 + 0.5265}{s^3 + 0.9782s^2 + 1.2434s + 0.5265}$
4	$\frac{0.01s^4 + 0.1502s^2 + 0.3220}{s^4 + 0.9391s^3 + 1.5137s^2 + 0.8037s + 0.3612}$
5	$\frac{0.0470s^4 + 0.2201s^2 + 0.2299}{s^5 + 0.9234s^4 + 1.8471s^3 + 1.1292s^2 + 0.7881s + 0.2299}$
6	$\frac{0.01s^6 + 0.1172s^4 + 0.28s^2 + 0.186}{s^6 + 0.9154s^5 + 2.2378s^4 + 1.4799s^3 + 1.4316s^2 + 0.5652s + 0.2087}$

Fig. 5.5 Magnitude response of typical elliptic lowpass filter with 1 dB passband ripple and 30 dB stopband ripple



and

$$k' = \sqrt{1 - k^2} = \sqrt{1 - 0.04} = 0.979796.$$

Substituting these values in Eqs. (5.32) and (5.34), we get

$$\rho_0 = 0.00255135,$$

$$\rho = 0.0025513525$$

and hence

$$N = \frac{\log\left(16 \times \frac{10^4 - 1}{10^{0.1} - 1}\right)}{\log_{10}\left(\frac{1}{0.0025513525}\right)} = 2.2331.$$

Choose $N = 3$. Then, for $N = 3$, a passband ripple of 1 dB, and a stopband ripple of 40 dB, the normalized elliptic transfer function is as given in Table 5.4. For $\Omega_p = 2000\pi$, the corresponding transfer function can be obtained by substituting $s = (s/\Omega_p) = (s/2000\pi)$ in the expression for $H_N(s)$. Thus, the required filter transfer function is

$$\begin{aligned} H_a(s) &= \frac{0.0692\left(\frac{s}{2000\pi}\right)^2 + 0.5265}{\left(\frac{s}{2000\pi}\right)^3 + 0.97825\left(\frac{s}{2000\pi}\right)^2 + 1.2434\left(\frac{s}{2000\pi}\right) + 0.5265} \\ &= \frac{4.348 \times 10^2 s^2 + 1.306 \times 10^{11}}{s^3 + 6.1465 \times 10^3 s^2 + 4.9087 \times 10^7 s + 1.306 \times 10^{11}} \end{aligned}$$

5.1.5 Bessel Filter

Bessel filter is a class of all-pole filters that provide linear phase response in the passband and characterized by the transfer function [5]

$$H_a(s) = \frac{1}{a_0 + a_1s + a_2s^2 + \cdots + a_{N-1}s^{N-1} + a_Ns^N} \quad (5.35)$$

where the coefficients a_N are given by

$$a_n = \frac{(2N - n)!}{2^{N-n}n!(N - n)!} \quad (5.36)$$

The magnitude responses of a third-order Bessel filter and Butterworth filter are shown in Fig. 5.6, and the phase responses of the same filters with the same order are shown in Fig. 5.7. From these figures, it is seen that the magnitude response of the Bessel filter is poorer than that of the Butterworth filter, whereas the phase response of the Bessel filter is more linear in the passband than that of the Butterworth filter.

5.1.6 Comparison of Various Types of Analog Filters

The magnitude response and phase response of the normalized Butterworth, Chebyshev Type 1, Chebyshev Type 2, and elliptic filters of the same order are compared with the following specifications:

filter order = 8, maximum passband ripple = 1 dB, and minimum stopband ripple = 35 dB.

Fig. 5.6 Magnitude responses of Bessel and Butterworth filters of order $N = 3$

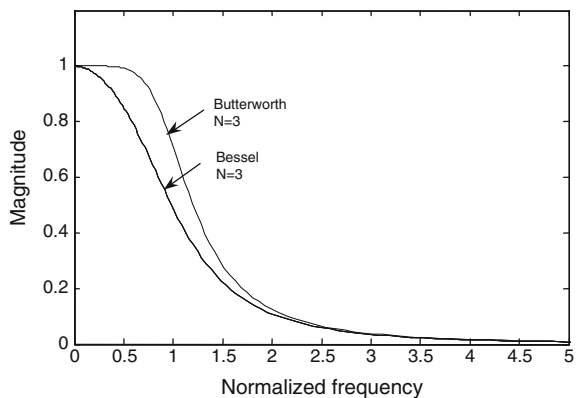
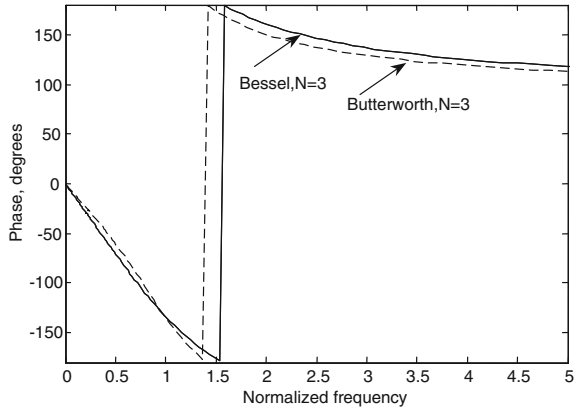


Fig. 5.7 Phase responses of Bessel and Butterworth filters of order $N = 3$



The following MATLAB program is used to generate the magnitude and phase responses for these specifications.

Program 5.5 Magnitude and phase responses of analog filters of order 8 with a passband ripple of 1 dB and a stopband ripple of 35 dB.

```
clear all;clc;
[z,p,k] = buttap(8);
[num1,den1] = zp2tf(z,p,k);[z,p,k] = cheb1ap(8,1);
[num2,den2] = zp2tf(z,p,k);[z,p,k] = cheb2ap(8,35);
[num3,den3] = zp2tf(z,p,k);[z,p,k] = ellipap(8,1,35);
[num4,den4] = zp2tf(z,p,k);
omega = [0:0.01:5];
h1 = freqs(num1,den1,omega);h2 = freqs(num2,den2,omega);
h3 = freqs(num3,den3,omega);h4 = freqs(num4,den4,omega);
ph1 = angle(h1);ph1 = unwrap(ph1);
ph2 = angle(h2);ph2 = unwrap(ph2);
ph3 = angle(h3);ph3 = unwrap(ph3);
ph4 = angle(h4);ph4 = unwrap(ph4);
Figure (1),plot(omega,20*log10(abs(h1)),'-');hold on
plot(omega,20*log10(abs(h2)),'-');hold on
plot(omega,20*log10(abs(h3)),'-');hold on
plot(omega,20*log10(abs(h4)),'-');
xlabel('Normalized frequency');ylabel('Gain,dB');axis([0 5-80 5]);
legend('Butterworth','Chebyshev Type 1','Chebyshev Type 2','Elliptic');hold off
Figure(2),plot(omega,ph1,'-');hold on
plot(omega,ph2,'-');hold on
plot(omega,ph3,'-');hold on
plot(omega,ph4,'-');
xlabel('Normalized frequency');ylabel('Phase,radians');axis([0 5 -8 0]);
legend('Butterworth','Chebyshev Type 1','Chebyshev Type 2','Elliptic');
```

The magnitude and phase responses for the above specifications are shown in Fig. 5.8. The magnitude response of Butterworth filter decreases monotonically in both the passband and the stopband with wider transition band. The magnitude response of the Chebyshev Type 1 exhibits ripples in the passband, whereas the Chebyshev Type 2 has approximately the same magnitude response to that of the Butterworth filter. The transition band of both the Type 1 and Type 2 Chebyshev filters is the same, but less than that of the Butterworth filter. The elliptic filter exhibits an equiripple magnitude response both in the passband and in the stopband with a transition width smaller than that of the Chebyshev Type 1 and Type 2 filters. But the phase response of the elliptic filter is more nonlinear in the passband than that of the phase response of the Butterworth and Chebyshev filters. If linear phase in the passband is the stringent requirement, then the Bessel filter is preferred, but with a poor magnitude response.

Another way of comparing the various filters is in terms of the order of the filter required to satisfy the same specifications. Consider a lowpass filter that meets the passband edge frequency of 450 Hz, stopband edge frequency of 550 Hz, passband ripple of 1 dB, and stopband ripple of 35 dB. The orders of the Butterworth, Chebyshev Type 1, Chebyshev Type 2, and elliptic filters are computed for the above specifications and listed in Table 5.5. From this table, we can see that elliptic filter can meet the specifications with the lowest filter order.

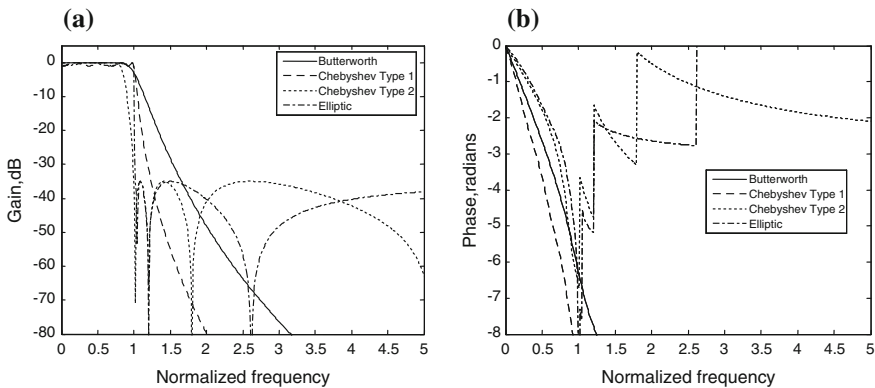


Fig. 5.8 Comparison of various types of analog lowpass filters **a** magnitude response and **b** phase response

Table 5.5 Comparison of orders of various types of filters

Filter	Order
Butterworth	24
Chebyshev Type 1	9
Chebyshev Type 2	9
Elliptic	5

5.1.7 Design of Analog Highpass, Bandpass, and Bandstop Filters

The analog highpass, bandpass, and bandstop filters can be designed using analog frequency transformations. In this design process, first, the analog prototype lowpass filter specifications are derived from the desired specifications of the analog filter using suitable analog-to-analog transformation. Next, by using the specifications so obtained, a prototype lowpass filter is designed. Finally, the transfer function of the desired analog filter is determined from the transfer function of the prototype analog lowpass transfer function using the appropriate analog-to-analog frequency transformation. The lowpass-to-lowpass, lowpass-to-highpass, lowpass-to-bandpass, and lowpass-to-bandstop analog transformations are considered next.

Lowpass to Lowpass:

Let $\Omega_p = 1$ and $\hat{\Omega}_p$ be the passband edge frequencies of the normalized prototype low pass filter and the desired lowpass filter, as shown in Fig. 5.9. The transformation from the prototype lowpass to the required lowpass must convert $\hat{\Omega} = 0$ to $\Omega = 0$ and $\hat{\Omega} = \pm\infty$ to $\Omega = \pm\infty$. The transformation such as $s = k\hat{s}$ or $\Omega = k\hat{\Omega}$ achieves the above transformation for any positive value of k . If k is chosen to be $(1/\hat{\Omega}_p)$, then $\hat{\Omega}_p$ gets transformed to $\Omega_p = 1$, and $\hat{\Omega}_s$ to $\Omega_s = \hat{\Omega}_s / \hat{\Omega}_p$. Since $\Omega_p = 1$ is the passband edge frequency for the normalized Type I Chebyshev and elliptic lowpass filters, we have the design equations for these filters as

$$\Omega_p = 1, \Omega_s = \hat{\Omega}_s / \hat{\Omega}_p. \tag{5.37a}$$

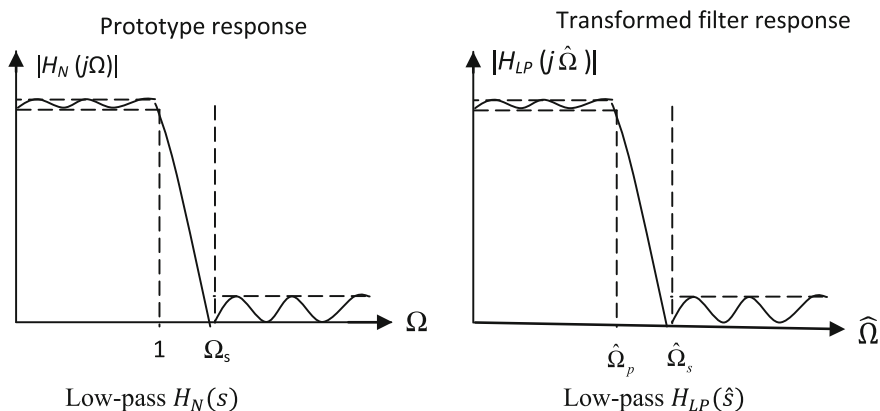


Fig. 5.9 Lowpass-to-lowpass frequency transformation

Also, the transfer function $H_{LP}(\hat{s})$ for these filters is related to the corresponding normalized transfer function $H_N(s)$ by

$$H_{LP}(\hat{s}) = H_N(s)_{s=\frac{\hat{s}}{\hat{\Omega}_p}} \tag{5.37b}$$

However, in the case of a Butterworth filter, since $\Omega = 1$ corresponds to the cutoff frequency of the filter, the transfer function $H_{LP}(\hat{s})$ for the Butterworth filter is related to the normalized lowpass Butterworth transfer function $H_N(s)$ by

$$H_{LP}(\hat{s}) = H_N(s)_{s=\hat{s}/\hat{\Omega}_c} \tag{5.37c}$$

where $\hat{\Omega}_c$ is the cutoff frequency of the desired Butterworth filter and is given by Eq. (5.5). For similar reasons, the transfer function $H_{LP}(\hat{s})$ for the Type 2 Chebyshev filter is related to the normalized transfer function $H_N(s)$ by

$$H_{LP}(\hat{s}) = H_N(s)_{s=\hat{s}/\hat{\Omega}_c} \tag{5.37d}$$

Lowpass to Highpass:

Let the passband edge frequencies of the prototype lowpass and the desired highpass filters be $\Omega_p = 1$ and $\hat{\Omega}_p$, as shown in Fig. 5.10. The transformation from prototype lowpass to the desired highpass must transform $\hat{\Omega} = 0$ to $\Omega = \infty$ and $\hat{\Omega} = \infty$ to $\Omega = 0$. The transformation such as $s = k/\hat{s}$ or $\hat{\Omega} = \infty$ achieves the above transformation for any positive value of k . By transforming $\hat{\Omega}_p$ to $\Omega_p = 1$, the constant k can be determined as $k = \hat{\Omega}_p$. Thus, design equations are

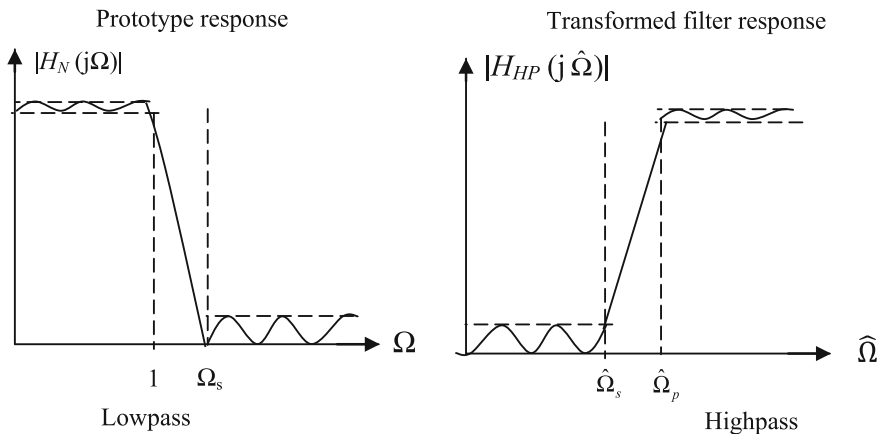


Fig. 5.10 Lowpass-to-highpass frequency transformation

$$\Omega_p = 1, \Omega_s = \hat{\Omega}_p / \hat{\Omega}_s, \tag{5.38a}$$

and the desired transfer function $H_{HP}(\hat{s})$ is related to the normalized lowpass transfer function $H_N(s)$ by

$$H_{HP}(\hat{s}) = H_N(s)|_{s=\hat{\Omega}_p/\hat{s}} \tag{5.38b}$$

Equations (5.38a) and (5.38b) hold for all filters except for Butterworth and Type 2 Chebyshev filter. For Butterworth

$$H_{LP}(\hat{s}) = H_N(s)_{s=\hat{s}/\hat{\Omega}_c} \tag{5.38c}$$

$$H_{HP}(\hat{s}) = H_N(s)|_{s=\hat{\Omega}_p/\hat{s}} \tag{5.38d}$$

For Type 2 Chebyshev filter, the design equations are

$$\Omega_p = \hat{\Omega}_s / \hat{\Omega}_p, \Omega_s = 1 \tag{5.39a}$$

and

$$H_{HP}(\hat{s}) = H_N(s)_{s=\hat{\Omega}_s/\hat{s}} \tag{5.39b}$$

Lowpass to Bandpass

The prototype lowpass and the desired bandpass filters are shown in Fig. 5.11. In this figure, $\hat{\Omega}_{p1}$ is the lower passband edge frequency, $\hat{\Omega}_{p2}$ the upper passband edge frequency, $\hat{\Omega}_{s1}$ the lower stopband edge frequency, and $\hat{\Omega}_{s2}$ the upper stopband

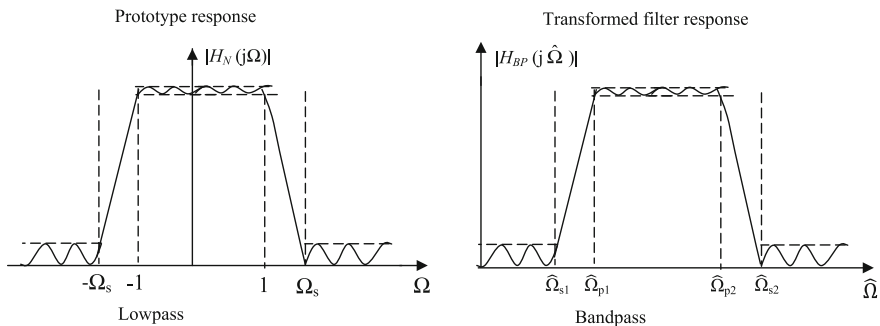


Fig. 5.11 Lowpass-to-bandpass frequency transformation

edge frequency of the desired bandpass filter. Let us denote by B_p , the bandwidth of the passband, and by $\hat{\Omega}_{mp}$ the geometric mean between the passband edge frequencies of the bandpass filter, i.e.,

$$B_p = \hat{\Omega}_{p2} - \hat{\Omega}_{p1} \quad (5.40a)$$

$$\hat{\Omega}_{mp} = \sqrt{\hat{\Omega}_{p1}\hat{\Omega}_{p2}} \quad (5.40b)$$

Now, consider the transformation

$$s = \frac{(\hat{s}^2 + \hat{\Omega}_{mp}^2)}{B_p \hat{s}} \quad (5.41)$$

As a consequence of this transformation, it is seen that $\hat{\Omega} = 0$, $\hat{\Omega}_{p1}$, $\hat{\Omega}_{mp}$, $\hat{\Omega}_{p2}$, and ∞ transform to the frequencies $\Omega = -\infty$, -1 , 0 , $+1$, and ∞ , respectively, for the normalized lowpass filter. Also, the transformation (5.41) transforms the frequencies $\hat{\Omega}_{s1}$ and $\hat{\Omega}_{s2}$ to Ω'_s and Ω''_s , respectively, where

$$\Omega'_s = \frac{\hat{\Omega}_{s1}^2 - \hat{\Omega}_{p1}\hat{\Omega}_{p2}}{(\hat{\Omega}_{p2} - \hat{\Omega}_{p1})\hat{\Omega}_{s1}} = A_1 \text{ (say)} \quad (5.42)$$

and

$$\Omega''_s = \frac{\hat{\Omega}_{s2}^2 - \hat{\Omega}_{p1}\hat{\Omega}_{p2}}{(\hat{\Omega}_{p2} - \hat{\Omega}_{p1})\hat{\Omega}_{s2}} = A_2 \text{ (say)} \quad (5.43)$$

In order to satisfy the stopband requirements and to have symmetry of the stopband edges in the lowpass filter, we choose Ω_s to be the $\min\{|A_1|, |A_2|\}$. Thus, the spectral transformation (5.41) leads to the following design equations for the normalized lowpass filter (except in the case of the Type 2 Chebyshev filter)

$$\Omega_p = 1, \Omega_s = \min\{|A_1|, |A_2|\} \quad (5.44a)$$

where A_1 and A_2 are given by (5.42) and (5.43), respectively, and the desired highpass transfer function $H_{BP}(\hat{s})$ can be obtained from the normalized lowpass transfer function $H_N(s)$ using (5.41). In the case of the Type 2 Chebyshev filter, the equation corresponding to (5.44a) is

$$\Omega_p = \max\{1/|A_1|, 1/|A_2|\}, \Omega_s = 1 \quad (5.44b)$$

Lowpass to Bandstop

The prototype lowpass and the desired bandstop filters are shown in Fig. 5.12. In this figure, $\hat{\Omega}_{p1}$ is the lower passband edge frequency, $\hat{\Omega}_{p2}$ the upper passband edge frequency, $\hat{\Omega}_{s1}$ the lower stopband edge frequency, and $\hat{\Omega}_{s2}$ the upper stopband edge frequency of the desired bandstop filter. Let us now consider the transformation

$$s = \frac{k\hat{s}}{\left(\hat{s}^2 + \hat{\Omega}_{ms}^2\right)} \tag{5.45}$$

where $\hat{\Omega}_{ms}$ is the geometric mean between the stopband edge frequencies of the bandstop filter, i.e.,

$$\hat{\Omega}_{ms} = \sqrt{\hat{\Omega}_{s1}\hat{\Omega}_{s2}} \tag{5.46}$$

As a consequence of this transformation, it is seen that $\hat{\Omega} = 0$ and ∞ transform to the frequency $\hat{\Omega} = 0$ for the normalized lowpass filter. Now, we transform the lower stopband edge frequency $\hat{\Omega}_{s1}$ to the stopband edge frequency Ω_s of the normalized lowpass filter; hence,

$$\Omega_s = \frac{k}{\hat{\Omega}_{s2} - \hat{\Omega}_{s1}} = \frac{k}{B_S} \tag{5.47a}$$

where $B_S = (\hat{\Omega}_{s2} - \hat{\Omega}_{s1})$ is the bandwidth of the stopband. Also, the upper stopband edge frequency $\hat{\Omega}_{s2}$ is transformed to

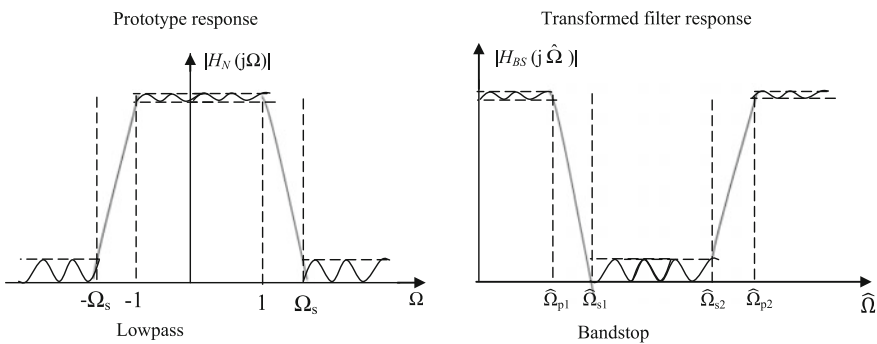


Fig. 5.12 Lowpass-to-bandstop frequency transformation

$$-\frac{k}{\hat{\Omega}_{s2} - \hat{\Omega}_{s1}} = -\frac{k}{B_S} = -\Omega_S \quad (5.47b)$$

Hence, the constant k is given by

$$k = B_S \Omega_S = (\hat{\Omega}_{s2} - \hat{\Omega}_{s1}) \Omega_S \quad (5.47c)$$

As a consequence, the passband edge frequencies $\hat{\Omega}_{p1}$ and $\hat{\Omega}_{p2}$ are transformed to

$$\Omega'_p = \frac{(\hat{\Omega}_{s2} - \hat{\Omega}_{s1}) \hat{\Omega}_{p1}}{\hat{\Omega}_{s1} \hat{\Omega}_{s2} - \hat{\Omega}_{p1}^2} \Omega_S = \frac{1}{A_1} \Omega_S \quad (5.48a)$$

and

$$\Omega''_p = \frac{(\hat{\Omega}_{s2} - \hat{\Omega}_{s1}) \hat{\Omega}_{p2}}{\hat{\Omega}_{s1} \hat{\Omega}_{s2} - \hat{\Omega}_{p2}^2} \Omega_S = \frac{1}{A_2} \Omega_S \quad (5.48b)$$

In order to satisfy the passband requirement as well as to satisfy the symmetry requirement of the passband edge of the normalized lowpass filter, we have to choose the higher of $|\Omega'_p|$ and $|\Omega''_p|$ as Ω_p . Since for the normalized filter (except for the case of Type 2 Chebyshev filter), $\Omega_p = 1$, we have to choose Ω_s to be the lower of $\{|A_1|, |A_2|\}$. Hence, the design equations for the normalized lowpass filter (except for the Type 2 Chebyshev) are

$$\Omega_p = 1, \Omega_s = \min\{|A_1|, |A_2|\} \quad (5.49a)$$

where

$$A_1 = \frac{\hat{\Omega}_{s1} \hat{\Omega}_{s2} - \hat{\Omega}_{p1}^2}{(\hat{\Omega}_{s2} - \hat{\Omega}_{s1}) \hat{\Omega}_{p1}}, A_2 = \frac{\hat{\Omega}_{s1} \hat{\Omega}_{s2} - \hat{\Omega}_{p2}^2}{(\hat{\Omega}_{s2} - \hat{\Omega}_{s1}) \hat{\Omega}_{p2}} \quad (5.49b)$$

and the transfer function of the required bandstop filter is

$$H_{BS}(\hat{s}) = H_N(s)_s = \frac{(\hat{\Omega}_{s2} - \hat{\Omega}_{s1}) \Omega_S \hat{s}}{\hat{s}^2 + \hat{\Omega}_{s1} \hat{\Omega}_{s2}} \quad (5.49c)$$

For the Type 2 Chebyshev filter, Eq. (5.49a) would be replaced by

$$\Omega_p = \max\{1/|A_1|, 1/|A_2|\}, \Omega_s = 1 \tag{5.50}$$

For further details on analog frequency transformations, readers may refer to [7].

5.2 Design of Digital Filters from Analog Filters

5.2.1 Digital Filter Specifications

The digital filter frequency response specifications are often in the form of a tolerance scheme. The specifications for a low pass filter are depicted in Fig. 5.13.

The following parameters are usually used as the specifications.

- ω_p —passband edge frequency
- ω_s —stopband edge frequency
- δ_p —peak ripple value in the passband
- δ_s —peak ripple value in the stopband

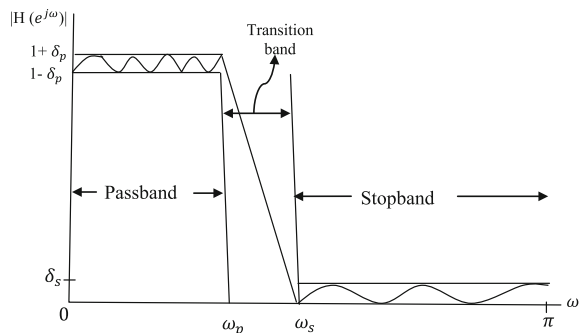
Generally, the passband edge frequency (f_p), the stopband edge frequency (f_s), and the sampling frequency (F_T) are represented in Hz. But, the digital filter design methods require normalized angular edge frequencies in radians. The normalized angular edge frequencies ω_p and ω_s can be obtained using the following relations

$$\omega_p = \frac{2\pi f_p}{F_T} = 2\pi f_p T \tag{5.51a}$$

$$\omega_s = \frac{2\pi f_s}{F_T} = 2\pi f_s T \tag{5.51b}$$

where T is the sampling period.

Fig. 5.13 Specifications of a digital lowpass filter



5.2.2 Design of Digital Filters Using Impulse-Invariant Method

In this method, the impulse response of an analog filter is uniformly sampled to obtain the impulse response of the digital filter, and hence, this method is called the impulse-invariant method. The process of designing an IIR filter using this method is as follows:

- Step 1 Design an analog filter to meet the given frequency specifications. Let $H_a(s)$ be the transfer function of the designed analog filter. We assume for simplicity that $H_a(s)$ has only simple poles. In such a case, the transfer function of the analog filter can be expressed in partial fraction form as

$$H_a(s) = \sum_{k=1}^N \frac{A_k}{s - p_k} \quad (5.52)$$

where A_k is the residue of $H(s)$ at the pole p_k .

- Step 2 Calculate the impulse response $h(t)$ of this analog filter by applying the inverse Laplace transformation on $H(s)$. Hence,

$$h_a(t) = \sum_{k=1}^N A_k e^{p_k t} u_a(t) \quad (5.53)$$

- Step 3 Sample the impulse response of the analog filter with a sampling period T . Then, the sampled impulse response $h(n)$ can be expressed as

$$\begin{aligned} h(n) &= h_a(t)|_{t=nT} \\ &= \sum_{k=1}^N (A_k e^{p_k t})^n u(n) \end{aligned} \quad (5.54)$$

- Step 4 Apply the z -transform on the sampled impulse response obtained in Step 3, to form the transfer function of the digital filter, i.e., $H(z) = Z[h(n)]$. Thus, the transfer function $H(z)$ for the impulse-invariant method is given by

$$H(z) = \sum_{k=1}^N \frac{A_k}{1 - e^{p_k T} z^{-1}} \quad (5.55)$$

This impulse-invariant method can be extended for the case when the poles are not simple [8].

Example 5.5 Design a third-order Butterworth digital filter using impulse-invariant technique. Assume a sampling period of $T = 1$ s.

Solution For $N = 3$, the transfer function of a normalized Butterworth filter is given by

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

$$H(s) = \frac{1}{s+1} + \frac{-0.5+0.288j}{s+0.5+j0.866} + \frac{-0.5-0.288j}{s+0.5-j0.866}$$

Hence, from (5.55), we have

$$H(z) = \frac{1}{1-e^{-1}z^{-1}} + \frac{-0.5+j0.288}{1-e^{-0.5}e^{-j0.866}z^{-1}} + \frac{-0.5-j0.288}{1-e^{-0.5}e^{j0.866}z^{-1}}$$

$$= \frac{1}{1-0.368z^{-1}} + \frac{-1+0.66z^{-1}}{1-0.786z^{-1}+0.368z^{-2}}$$

Example 5.6 Design a Butterworth filter using the impulse-invariant method for the following specifications:

$$0.8 \leq |H(e^{j\omega})| \leq 1 \quad 0 \leq \omega \leq 0.2\pi$$

$$|H(e^{j\omega})| \leq 0.2 \quad 0.6\pi \leq \omega \leq \pi$$

Solution From (5.1), the magnitude-squared function of the Butterworth filter is

$$|H_a(j\Omega)|^2 = \frac{1}{1+(\Omega/\Omega_c)^{2N}}$$

Substituting the requirements in the above magnitude function, we get

$$1 + \left(\frac{0.2\pi}{\Omega_c}\right)^{2N} = \left(\frac{1}{0.8}\right)^2$$

$$1 + \left(\frac{0.6\pi}{\Omega_c}\right)^{2N} = \left(\frac{1}{0.2}\right)^2$$

The solution of the above two equations leads to

$$N = \frac{\log \frac{24}{0.5625}}{2 \log 3} = \frac{1.6301}{0.9542} = 1.71$$

Approximating to the nearest higher value, we have $N = 2$. Substituting $N = 2$ in

$$1 + \left(\frac{0.2\pi}{\Omega_c}\right)^{2N} = \left(\frac{1}{0.8}\right)^2$$

we get $\Omega_c = 0.231\pi$. Also, for $N = 2$ the transfer function of the normalized Butterworth filter is

$$H_N(s) = \frac{1}{s^2 + \sqrt{2}s + 1}$$

Hence, from (5.37c),

$$\begin{aligned} H_a(s) &= H_N(s)_{s=s/\Omega_c} \\ &= \frac{0.5266}{s^2 + 1.03s + 0.5266} \\ &= \frac{0.516j}{s + 0.51 + j0.51} - \frac{0.516j}{s + 0.51 - j0.51} \\ H(z) &= \frac{0.516j}{1 - e^{-0.51T}e^{-j0.51T}z^{-1}} - \frac{0.516j}{1 - e^{-0.51T}e^{j0.51T}z^{-1}} \end{aligned}$$

Since $T = 1$, we have

$$H(z) = \frac{0.3019z^{-1}}{1 - 1.048z^{-1} + 0.36z^{-2}}$$

Disadvantage of Impulse-Invariant Method

The frequency responses of the digital and analog filters are related by

$$H(e^{j\omega}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} H_a\left(j\frac{\omega + 2\pi k}{T}\right) \quad (5.56)$$

From Eq. (5.56), it is evident that the frequency response of the digital filter is not identical to that of the analog filter due to aliasing in the sampling process. If the analog filter is band-limited with

$$H_a\left(j\frac{\omega}{T}\right) = 0 \quad \left|\frac{\omega}{T}\right| = |\Omega| \geq \pi/T \quad (5.57)$$

then the digital filter frequency response is of the form

$$H(e^{j\omega}) = \frac{1}{T} H_a\left(j\frac{\omega}{T}\right) \quad |\omega| \leq \pi \quad (5.58)$$

In the above expression, if T is small, the gain of the filter becomes very large. This can be avoided by introducing a multiplication factor T in the impulse-invariant transformation. In such a case, the transformation would be

$$h(n) = T h_a(nT) \quad (5.59)$$

and $H(z)$ would be

$$H(z) = T \sum_{k=1}^N \frac{A_k}{1 - e^{p_k T} z^{-1}} \quad (5.60)$$

Also, the frequency response is

$$H(e^{j\omega}) = \frac{1}{T} H_a\left(j\frac{\omega}{T}\right) \quad |\omega| \leq \pi \quad (5.61)$$

Hence, the impulse-invariant method is appropriate only for band-limited filters, i.e., lowpass and bandpass filters, but not suitable for highpass or bandstop filters where additional band limiting is required to avoid aliasing. Thus, there is a need for another mapping method such as bilinear transformation technique which avoids aliasing.

5.2.3 Design of Digital Filters Using Bilinear Transformation

In order to avoid the aliasing problem mentioned in the case of the impulse-invariant method, we use the bilinear transformation, which is a one-to-one mapping from the s -plane to the z -plane; that is, it maps a point in the s -plane to a unique point in the z -plane and vice versa. This is the method that is mostly in designing an IIR digital filter from an analog filter. This approach is based on the trapezoidal rule, and for details, one could refer to [8]. Consider the bilinear transformation given by

$$s = \frac{(z-1)}{(z+1)} \quad (5.62)$$

Then, a transfer function $H_a(s)$ in the analog domain is transformed in the digital domain as

$$H(z) = H_a(s) \Big|_{s=\frac{z-1}{z+1}} \quad (5.63)$$

Also, from (5.62) we have

$$z = \frac{1+s}{1-s} \quad (5.64)$$

We now study the mapping properties of the bilinear transformation. Consider a point $s = -\sigma + j\Omega$ in the left half of the s -plane. Then, from (5.64), we get

$$|z| = \left| \frac{1 - \sigma + j\Omega}{1 + \sigma - j\Omega} \right| > 1 \quad (5.65)$$

Hence, the left half of the s -plane maps into the interior of the unit circle in the z -plane (see Fig. 5.14). Similarly, it can be shown that the right half of the s -plane maps into the exterior of the unit circle in the z -plane. For a point z on the unit circle, $z = e^{j\omega}$, we have from (5.62)

$$s = \frac{e^{j\omega} - 1}{e^{j\omega} + 1} = j \tan \frac{\omega}{2} \quad (5.66)$$

Thus

$$\Omega = \tan \frac{\omega}{2} \quad (5.67)$$

or

$$\omega = 2 \tan^{-1} \Omega \quad (5.68)$$

showing that the positive and negative imaginary axes of the s -plane are mapped respectively into the upper and lower halves of the unit circle in the z -plane. We

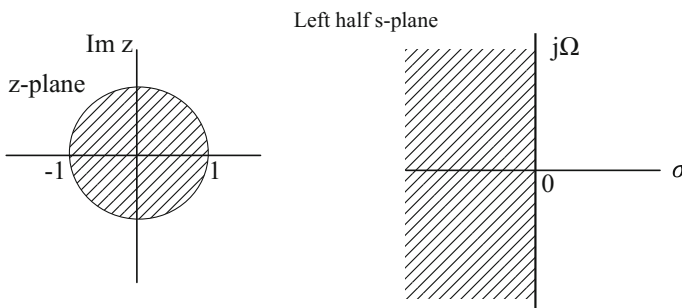


Fig. 5.14 Mapping of the s -plane into the z -plane by the bilinear transformation

thus see that the bilinear transformation avoids the problem of aliasing encountered in the impulse-invariant method, since it maps the entire imaginary axis in the s -plane onto the unit circle in the z -plane. Further, in view of the mapping, this transformation converts a stable analog filter into a stable digital filter.

Warping effect

The price paid, however, is in the introduction of a distortion in the frequency axis due to the nonlinear relation between Ω and ω , exhibited particularly at higher frequencies, as shown in Fig. 5.15. This behavior is called the warping effect. This can be corrected by ‘prewarping’ the analog filter specifications. The procedure to be followed is as follows:

- Step 1 From the digital filter specifications, prewarp the critical frequencies, such as the cutoff frequency, passband edge, stopband edge using Eq. (5.67).
- Step 2 From these new critical frequencies, obtain the transfer function $H_a(s)$ of the analog filter using the methods already described.

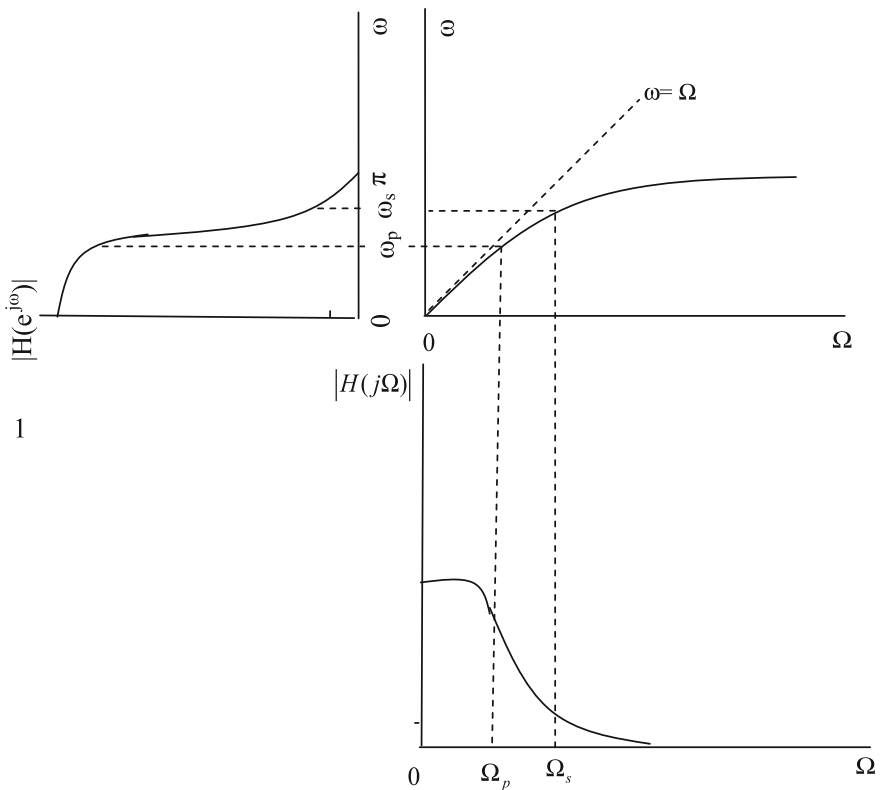


Fig. 5.15 Warping effect due to bilinear transformation

Step 3 Use the bilinear transformation given by Eq. (5.62) to obtain the corresponding digital transfer function $H(z)$.

Example 5.7 Design a low pass Butterworth IIR digital filter using bilinear transformation for the following specifications:

3 dB cutoff frequency $\omega_c = 0.2\pi$ and $|H(e^{j\omega})| \leq 0.317$ for $0.4\pi \leq \omega \leq \pi$.

Solution From Eq. (5.1), the magnitude-squared function of the Butterworth filter is

$$|H_a(j\Omega)|^2 = \frac{1}{1 + (\Omega/\Omega_c)^{2N}}$$

As bilinear transformation is used and $\omega_c = 0.2\pi$, prewarping of the cutoff frequency yields

$$\Omega_c = \tan\left(\frac{0.2\pi}{2}\right) = \tan(0.1\pi) = 0.325$$

From the magnitude response specification, we obtain

$$1 + \left(\frac{\tan(0.4\pi/2)}{\tan(0.2\pi/2)}\right)^{2N} = \left(\frac{1}{0.317}\right)^2$$

Solving the above equation gives $N = 2$. Hence,

$$H_a(s) = \frac{\Omega_c^2}{s^2 + \sqrt{2}\Omega_c s + \Omega_c^2}$$

Thus,

$$H_a(s) = \frac{0.10563}{s^2 + 0.46s + 0.10563}$$

The digital transfer function $H(z)$ is now obtained by using Eq. (5.62) in the above transfer function $H_a(s)$.

$$H(z) = \frac{0.068(z+1)^2}{z^2 - 1.142z + 0.413}$$

Example 5.8 Consider the following analog transfer function

$$H_a(s) = \frac{s^2 - 3s + 3}{s^2 + 3s + 3}$$

- (i) Is it possible to obtain the corresponding digital transfer function using the impulse-invariant method?
- (ii) Is it possible to obtain the corresponding digital transfer function using bilinear transformation?

Solution

- (i) $H_a(s) = \frac{s^2 - 3s + 3}{s^2 + 3s + 3}$ represents an allpass system.

According to the impulse-invariant design, using Eq. (5.56),

$$H(e^{j\omega}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} H_a\left(j\frac{\omega + 2\pi k}{T}\right)$$

The aliasing terms will destroy the allpass nature of the continuous time filter. Therefore, one cannot design a corresponding digital system using the impulse-invariant method.

- (ii) The bilinear transformation only warps the frequency axis. The magnitude response is not affected; therefore, an allpass filter will map to an allpass filter. Thus, one can design a corresponding digital system using the bilinear transformation.

$$H(z) = H_a(s)|_{s=(z-1)/(z+1)}$$

Example 5.9 Design a low pass IIR digital filter using the bilinear transformation for the following specifications:

$$0.9 \leq |H(e^{j\omega})| \leq 1, \quad 0 \leq \omega \leq 0.2\pi$$

$$|H(e^{j\omega})| \leq 0.25, \quad 0.3\pi \leq \omega \leq \pi$$

Solution Prewarping the critical frequencies, we have the passband and stopband edge frequencies of the analog filter to be

$$\Omega_p = \tan\left(\frac{0.2\pi}{2}\right) = 0.325$$

$$\Omega_s = \tan\left(\frac{0.3\pi}{2}\right) = 0.51$$

Since

$$|H_a(j\Omega)|^2 = \frac{1}{1 + (\Omega/\Omega_c)^{2N}}$$

we have

$$1 + \left(\frac{\Omega_p}{\Omega_c}\right)^{2N} = 1 + \left(\frac{0.325}{\Omega_c}\right)^{2N} = \left(\frac{1}{0.9}\right)^2$$

and

$$1 + \left(\frac{\Omega_s}{\Omega_c}\right)^{2N} = 1 + \left(\frac{0.51}{\Omega_c}\right)^{2N} = \left(\frac{1}{0.25}\right)^2$$

Solving the above two equations, we get $N = 4.6$; hence, we choose $N = 5$. Using this value of N , we can calculate Ω_c to be $\Omega_c = 0.398$. Hence, we have

$$\begin{aligned} H_a(s) &= \frac{\Omega_c^5}{(s + \Omega_c)(s^2 + 0.61803\Omega_c s + \Omega_c^2)(s^2 + 1.61803\Omega_c s + \Omega_c^2)} \\ &= \frac{0.01}{(s + 0.398)(s^2 + 0.246s + 0.1584)(s^2 + 0.644s + 0.1584)} \end{aligned}$$

Now substituting for s using Eq. (5.62), we get the required digital transfer function to be

$$H(z) = \frac{0.01(z+1)}{(1.398z - 0.602)(1.404z^2 - 1.683z + 0.9124)(1.802z^2 - 1.683z + 0.5144)}$$

Example 5.10 Design a lowpass digital filter with 3 dB cutoff frequency at 50 Hz and attenuation of at least 10 dB for frequency larger than 100 Hz. Assume a suitable sampling frequency.

Solution Assume the sampling frequency as 500 Hz. Then,

$$\omega_c = \frac{2\pi f_c}{F_T} = \frac{2\pi \times 50}{500} = 0.2\pi$$

$$\omega_s = \frac{2\pi f_s}{F_T} = \frac{2\pi \times 100}{500} = 0.4\pi$$

Prewarping of the above-normalized frequencies yields

$$\Omega_c = \tan\left(\frac{0.2\pi}{2}\right) = \tan(0.1\pi) = 0.325$$

$$\Omega_s = \tan\left(\frac{0.4\pi}{2}\right) = \tan(0.2\pi) = 0.727$$

Substituting these values in $(\Omega_s/\Omega_c)^{2N} = 10^{0.1\alpha_s} - 1$ and solving for N , we get

$$N = \frac{\log(10^1 - 1)}{2 \log(0.727/0.325)} = \frac{0.9542}{0.6993} = 1.3643.$$

Hence, the order of the Butterworth filter is 2. The normalized lowpass Butterworth filter for $N = 2$ is given by

$$H_N(s) = \frac{1}{s^2 + \sqrt{2}s + 1}$$

The transfer function $H_c(s)$ corresponding to $\Omega_c = 0.325$ is obtained by substituting

$$s = \frac{s}{\Omega_c} = \frac{s}{0.325}$$

in the expression for $H_N(s)$; hence,

$$H_a(s) = \frac{0.1056}{s^2 + 0.4595s + 0.1056}$$

The digital transfer function $H(z)$ of the desired filter is now obtained by using the bilinear transformation (5.52) in the above expression:

$$H(z) = H_c(s) \Big|_{s=\frac{z-1}{z+1}}$$

$$H(z) = \frac{0.1056z^2 + 0.2112z + 0.1056}{1.5651z^2 - 1.7888z + 0.646}$$

Example 5.11 Design a lowpass Butterworth IIR filter for the following specifications:

Passband edge frequency: 1000 Hz

Stopband edge frequency: 3000 Hz

Passband ripple: 2 dB

Stopband ripple: 20 dB

Assume a suitable sampling frequency and use the bilinear transformation.

Solution Assuming the sampling frequency as 8 kHz, the normalized angular band edge frequencies are given by

$$\omega_p = \frac{2\pi f_p}{F_T} = \frac{2\pi \times 1000}{8000} = 0.25\pi$$

$$\omega_s = \frac{2\pi f_s}{F_T} = \frac{2\pi \times 3000}{8000} = 0.75\pi$$

By prewarping these frequencies, we get

$$\hat{\Omega}_p = \tan(\omega_p/2) = 0.4142; \hat{\Omega}_s = \tan(\omega_s/2) = 2.4142.$$

For the prototype analog lowpass filter, we get

$$\Omega_p = 1, \Omega_s = \hat{\Omega}_s/\hat{\Omega}_p = 2.4142/0.41422 = 5.8286, \alpha_p = 2 \text{ dB}, \alpha_s = 20 \text{ dB}$$

Using these values, the order of the filter is computed using Eq. (5.9) as

$$N \geq \frac{\log\left(\frac{10^2-1}{10^{0.2}-1}\right)}{2\log\left(\frac{5.8286}{1}\right)} = 1.4556$$

Hence, we choose $N = 2$. The normalized lowpass Butterworth filter for $N = 2$ is given by

$$H_N(s) = \frac{1}{s^2 + \sqrt{2}s + 1}$$

Substituting the values of $\hat{\Omega}_s$ and N in Eq. (5.6), we obtain

$$\left(2.4142/\hat{\Omega}_c\right)^4 = 10^2 - 1$$

Solving for $\hat{\Omega}_c$, we get $\hat{\Omega}_c = 0.7654$. The transfer function corresponding to $\hat{\Omega}_c = 0.7654$ is obtained by substituting $s = \left(s/\hat{\Omega}_c\right) = (s/0.7654)$ in $H_N(s)$; hence,

$$H_a(s) = \frac{0.5858}{s^2 + 1.0824s + 0.5858}$$

The digital transfer function $H(z)$ of the desired filter is now obtained as

$$H(z) = H_a(s)\Big|_{s=\frac{z-1}{z+1}}$$

$$H(z) = \frac{0.2195z^2 + 0.439z + 0.2195}{z^2 - 0.31047z + 0.1887}$$

Example 5.12 Design a lowpass Chebyshev Type 1 IIR filter for the following specifications:

Passband edge frequency: 1 kHz

Stopband edge frequency: 3 kHz

Sampling frequency: 10 kHz

Passband ripple: 1 dB

Stopband ripple: 40 dB

Solution The normalized angular band edge frequencies are given by

$$\omega_p = \frac{2\pi f_p}{F_T} = \frac{2\pi \times 1000}{10000} = 0.2\pi$$

$$\omega_s = \frac{2\pi f_s}{F_T} = \frac{2\pi \times 3000}{10000} = 0.6\pi$$

By prewarping these frequencies, we get

$$\hat{\Omega}_p = \tan(\omega_p/2) = 0.32492; \hat{\Omega}_s = \tan(\omega_s/2) = 1.3764.$$

For the prototype analog lowpass filter

$$\Omega_p = 1, \Omega_s = \hat{\Omega}_s/\hat{\Omega}_p = 1.3764/0.32492 = 4.236, \alpha_p = 1 \text{ dB}, \alpha_s = 40 \text{ dB}$$

Hence from (5.21), we have

$$N \geq \frac{\cosh^{-1} \sqrt{\frac{10^{\alpha_s} - 1}{10^{\alpha_p} - 1}}}{\cosh^{-1}(4.236)} = 2.45$$

Hence, we choose $N = 3$. For $N = 3$, from Table 5.2, the normalized transfer function is given by

$$H_N(s) = \frac{0.49131}{s^3 + 0.988s^2 + 1.238s + 0.49131}$$

The transfer function corresponding to $\hat{\Omega}_p = 0.32492$ is obtained by substituting $s = (s/\hat{\Omega}_p) = (s/0.32492)$ in $H_N(s)$; hence,

$$H_a(s) = \frac{0.016849}{s^3 + 0.32099s^2 + 0.13068s + 0.016849}$$

The digital transfer function $H_{LP}(z)$ of the desired lowpass filter is now obtained as

$$H_{LP}(z) = H_a(s) \Big|_{s=\frac{z-1}{z+1}}$$

$$H_{LP}(z) = \frac{0.011474z^3 + 0.034421z^2 + 0.034421z + 0.011474}{z^3 - 2.178z^2 + 1.7698z + 0.53976}$$

Example 5.13 Design a lowpass Chebyshev Type 2 IIR digital filter for the specifications given in Example 5.12.

Solution The order of the filter required is the same as in Example 5.12, i.e., $N = 3$. For, $N = 3$, from Table 5.3, the normalized transfer function is given by

$$H_N(s) = \frac{0.03(s^2 + 1.3333)}{(s^3 + 0.6746s^2 + 0.22709s + 0.04)}$$

for which the stopband edge is at $\Omega_s = 1$. The transfer function corresponding to the stopband edge $\hat{\Omega}_s = 1.3764$ is obtained by substituting $s = \left(s/\hat{\Omega}_s\right) = (s/1.3764)$ in the expression for $H_N(s)$; hence,

$$H_a(s) = \frac{0.041292s^2 + 0.10430}{s^3 + 0.92852s^2 + 0.43022s + 0.10430}$$

The digital transfer function $H_{LP}(z)$ of the desired lowpass filter is now obtained as

$$H(z) = H_a(s) \Big|_{s=\frac{z-1}{z+1}}$$

$$H_{LP}(z) = \frac{0.059111z^3 + 0.11028z^2 + 0.11028z + 0.059111}{z^3 - 1.2933z^2 + 0.7934z - 0.16134}$$

Example 5.14 Design an elliptic lowpass IIR filter for the following specifications:

Passband edge frequency: 800 Hz

Stopband edge frequency: 1600 Hz

Sampling frequency: 4 kHz

Passband ripple: 1 dB

Stopband ripple: 40 dB

Solution The normalized angular bandedge frequencies are given by

$$\omega_p = \frac{2\pi \times 800}{4000} = 0.4\pi$$

$$\omega_s = \frac{2\pi \times 1600}{4000} = 0.8\pi$$

Prewarping these frequencies, we get

$$\hat{\Omega}_p = \tan(\omega_p/2) = 0.72654; \hat{\Omega}_s = \tan(\omega_s/2) = 3.0777.$$

For the prototype analog lowpass filter

$$\Omega_p = 1, \Omega_s = \hat{\Omega}_s/\hat{\Omega}_p = 4.236, \quad \alpha_p = 1 \text{ dB}, \quad \alpha_s = 40 \text{ dB}$$

$$k = \frac{\Omega_p}{\Omega_s} = \frac{0.72654}{3.0777} = 0.23607$$

$$k' = \sqrt{1 - k^2} = 0.97174$$

$$\rho_0 = \frac{1 - \sqrt{k'}}{2(1 + \sqrt{k'})} = 0.0035837$$

$$\rho = \rho_0 + 2(\rho_0)^5 + 15(\rho_0)^9 + 150(\rho_0)^{13} = 0.0035837$$

$$N \cong \frac{\log\left(16 \times \frac{10^{0.1z_s} - 1}{10^{0.1z_p} - 1}\right)}{\log_{10}(1/\rho)} = 2.3678$$

Hence, we choose $N = 3$. For $N = 3$, from Table 5.4, the normalized transfer function is

$$H_N(s) = \frac{0.0692s^2 + 0.5265}{s^3 + 0.9782s^2 + 1.2434s + 0.5265}$$

The transfer function corresponding to the passband edge $\hat{\Omega}_p = 0.72654$ is obtained by substituting $s = (s/\hat{\Omega}_p) = (s/0.72654)$ in the expression for $H_N(s)$; hence,

$$H_a(s) = \frac{0.050277s^2 + 0.10430}{s^3 + 0.7107s^2 + 0.65634s + 0.20192}$$

The digital transfer function $H_{LP}(z)$ of the desired lowpass filter is now obtained as

$$H_{LP}(z) = H_a(s)\Big|_{s=\frac{z-1}{z+1}}$$

$$H_{LP}(z) = \frac{0.09817z^3 + 0.21622z^2 + 0.21622z + 0.09817}{z^3 - 0.95313z^2 + 0.87143z - 0.2895}$$

Example 5.15 Design a Butterworth IIR digital highpass filter for the following specifications:

Passband edge frequency: 40 Hz

Stopband edge frequency: 25 Hz

Sampling frequency: 100 Hz

Passband ripple: 1 dB

Stopband ripple: 20 dB

Solution The normalized angular bandedge frequencies are

$$\omega_s = \frac{2\pi \times 25}{100} = 0.5\pi$$

$$\omega_p = \frac{2\pi \times 40}{100} = 0.8\pi$$

Prewarping these frequencies, we get

$$\hat{\Omega}_p = \tan(\omega_p/2) = 3.0777$$

$$\hat{\Omega}_s = \tan(\omega_s/2) = 1.0$$

For the prototype analog lowpass filter, we have

$$\Omega_p = 1, \Omega_s = \hat{\Omega}_p/\hat{\Omega}_s = 3.0777, \alpha_p = 1 \text{ dB}, \alpha_s = 20 \text{ dB}$$

Substituting these values in Eq. (5.9), the order of the filter is given by

$$N \geq \frac{\log\left(\frac{10^2-1}{10^{0.1}-1}\right)}{2\log\left(\frac{3.0777}{1}\right)} = 2.6447$$

Hence, we choose $N = 3$. From Table 5.1, the third-order normalized Butterworth lowpass filter transfer function is given by

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

Substituting the values of Ω_s and N in Eq. (5.6), we obtain

$$\left(\frac{3.0777}{\Omega_c}\right)^6 = 10^2 - 1$$

Solving for Ω_c , we get $\Omega_c = 1.4309$.

The analog transfer function of the lowpass filter is obtained from the above transfer function by substituting $s = \frac{s}{\Omega_c} = \frac{s}{1.4309}$; hence,

$$H_{LP}(s) = \frac{2.93}{s^3 + 2.8619s^2 + 4.0952s + 2.93}$$

From the above transfer function, the analog transfer function of the highpass filter can be obtained by substituting $s = \frac{\hat{\Omega}_p}{s} = \frac{3.0777}{s}$

$$H_{HP}(s) = \frac{s^3}{s^3 + 4.3017s^2 + 9.2521s + 9.9499}$$

The digital transfer function of the required highpass filter is obtained by using the bilinear transformation:

$$H_{HP}(z) = H_{HP}(s)|_{s=\frac{z-1}{z+1}}$$

Thus,

$$H_{HP}(z) = \frac{0.0408z^3 - 0.1224z^2 + 0.1224z - 0.0408}{z^3 + 1.2978z^2 + 0.7875z + 0.1632}$$

Example 5.16 Design a Type 1 Chebyshev IIR digital highpass filter for the following specifications:

Passband edge frequency: 700 Hz
 Stopband edge frequency: 500 Hz
 Sampling frequency: 2 kHz
 Passband ripple: 1 dB
 Stopband ripple: 40 dB

Solution Normalized angular bandedge frequencies are

$$\omega_s = \frac{2\pi \times 500}{2000} = 0.5\pi; \quad \omega_p = \frac{2\pi \times 700}{2000} = 0.7\pi$$

Prewarping these frequencies, we get

$$\hat{\Omega}_p = \tan(\omega_p/2) = 1.9626105, \hat{\Omega}_s = \tan(\omega_s/2) = 1$$

For the prototype analog lowpass filter

$$\Omega_p = 1, \Omega_s = \hat{\Omega}_p/\hat{\Omega}_s = 1.9626105, \alpha_p = 1 \text{ dB}, \alpha_s = 40 \text{ dB}$$

Substituting these values in Eq. (5.21), the order of the filter is given by

$$N \geq \frac{\cosh^{-1} \sqrt{\frac{10^4 - 1}{10^{0.1} - 1}}}{\cosh^{-1}(1.9626)} = 4.6127$$

Hence, we choose $N = 5$. From (5.18), we have

$$\frac{1}{\epsilon} = \frac{1}{\sqrt{10^{0.1z_p} - 1}} = 1.965227$$

$$\sinh^{-1}\left(\frac{1}{\epsilon}\right) = \sinh^{-1}(1.965227) = 1.428$$

Using Eqs. (5.24) and (5.25a, 5.25b), the normalized transfer function is given by

$$H_N(s) = \frac{H_0}{\prod_k (s - p_k)}$$

where

$$p_k = -\sinh\left\{\frac{1}{N} \sinh^{-1}\left(\frac{1}{\epsilon}\right)\right\} \cdot \sin\left\{\frac{(2k-1)\pi}{2N}\right\} \\ + j \cosh\left\{\frac{1}{N} \sinh^{-1}\left(\frac{1}{\epsilon}\right)\right\} \cdot \cos\left\{\frac{(2k-1)\pi}{2N}\right\}$$

and

$$H_0 = \frac{1}{2^{N-1}} \frac{1}{\epsilon}$$

Substituting $N = 5$ and $k = 1, 2, 3, 4, 5$ in the above equations, we get

$$p_{1,5} = -0.08946 \pm j0.99014, \quad p_{2,4} = -0.23421 \pm j0.61194, \quad p_3 = -0.2895$$

and

$$H_0 = 0.12283$$

Hence,

$$H_N(s) = \frac{0.12283}{s^5 + 0.93682s^4 + 1.6888s^3 + 0.9744s^2 + 0.58053s + 0.12283}$$

The analog transfer function of the highpass filter can be obtained from the above transfer function by substituting $s = (\hat{\Omega}_p/s) = (1.9626105/s)$;

$$H_{HP}(s) = \frac{0.12283s^5}{s^5 + 9.2762s^4 + 30.557s^3 + 103.94s^2 + 113.16s + 237.07}$$

The digital transfer function of the highpass filter can be obtained by using bilinear transformation:

$$H_{HP}(z) = H_{HP}(s)|_{s=\frac{z-1}{z+1}}$$

Thus,

$$H_{HP}(z) = \frac{0.0020202z^5 - 0.010101z^4 + 0.020202z^3 - 0.020202z^2 + 0.010101z - 0.0020202}{z^5 + 3.1624z^4 + 4.7607z^3 + 4.0528z^2 + 1.9344z + 0.41529}$$

Example 5.17 Using bilinear transformation, design a digital bandpass Butterworth filter with the following specifications:

Lower passband edge frequency: 200 Hz
 Upper passband edge frequency: 400 Hz
 Lower stopband edge frequency: 100 Hz
 Upper stopband edge frequency: 500 Hz
 Passband ripple: 2 dB
 Stopband ripple: 20 dB

Assume a suitable sampling frequency.

Solution Assuming the sampling frequency to be 2000 Hz, the normalized angular bandedge frequencies are given by

$$\omega_{p1} = 0.2\pi, \omega_{p2} = 0.4\pi, \omega_{s1} = 0.1\pi, \omega_{s2} = 0.5\pi$$

The prewarped analog frequencies are given by

$$\begin{aligned}\hat{\Omega}_{p1} &= \tan(\omega_{p1}/2) = \tan(\pi/10) = 0.325 \\ \hat{\Omega}_{p2} &= \tan(\omega_{p2}/2) = \tan(\pi/5) = 0.7265 \\ \hat{\Omega}_{s1} &= \tan(\omega_{s1}/2) = \tan(\pi/20) = 0.1584 \\ \hat{\Omega}_{s2} &= \tan(\omega_{s2}/2) = \tan(\pi/4) = 1\end{aligned}$$

We now obtain the corresponding specifications for the normalized analog lowpass filter using the lowpass-to-bandpass transformation. From Eqs. (5.42) to (5.43), we have

$$A_1 = \frac{\hat{\Omega}_{s1}^2 - \hat{\Omega}_{p1}\hat{\Omega}_{p2}}{(\hat{\Omega}_{p2} - \hat{\Omega}_{p1})\hat{\Omega}_{s1}} = 1.90258$$

and

$$A_2 = \frac{\hat{\Omega}_{s2}^2 - \hat{\Omega}_{p1}\hat{\Omega}_{p2}}{(\hat{\Omega}_{p2} - \hat{\Omega}_{p1})\hat{\Omega}_{s2}} = -3.318$$

Now using (5.44a), we get the specifications for the normalized analog lowpass filter to be

$$\begin{aligned}\Omega_p &= 1, \Omega_s = \min\{|A_1|, |A_2|\} = 1.90258, \\ \alpha_p &= 2 \text{ dB}, \alpha_s = 20 \text{ dB}\end{aligned}$$

Substituting these values in Eq. (5.9), the order of the filter is given by

$$N \geq \frac{\log\left(\frac{10^2-1}{10^{0.2}-1}\right)}{2 \log(1.90258)} = 3.9889$$

We choose $N = 4$. The transfer function of the fourth-order normalized Butterworth lowpass filter is given by

$$H_N(s) = \frac{1}{s^4 + 2.6131s^3 + 3.4142s^2 + 2.6131s + 1}$$

Substituting the values of Ω_s and N in Eq. (5.6), we obtain

$$\left(\frac{1.90258}{\Omega_c}\right)^8 = 10^2 - 1$$

Solving for Ω_c , we get $\Omega_c = 1.0712$. The analog transfer function of the lowpass filter is obtained from the above transfer function by substituting $s = \frac{s}{\Omega_c} = \frac{s}{1.0712}$

$$H_{LP}(s) = \frac{1.3169}{s^4 + 2.7993s^3 + 3.9180s^2 + 3.2124s + 1.3169}$$

To arrive at the analog transfer function of the bandpass filter, we use in the above expression the lowpass-to-bandpass transformation given by (5.41), namely

$$s = \frac{s^2 + \hat{\Omega}_{p1}\hat{\Omega}_{p2}}{(\hat{\Omega}_{p2} - \hat{\Omega}_{p1})s} = \frac{s^2 + 0.236}{0.402s}$$

to obtain

$$\begin{aligned}H_{BP}(s) &= \frac{0.0344s^4}{s^8 + 1.1253s^7 + 1.5772s^6 + 1.0054s^5 + 0.6674s^4 + 0.2373s^3 + 0.0878s^2 + 0.0148s + 0.0031}\end{aligned}$$

The digital bandpass filter is now obtained by using the bilinear transformation in the above expression. Thus,

$$H_{BP}(z) = \frac{z^8 - 0.0241z^6 + 0.0361z^4 - 0.0241z^2 + 0.0060}{z^8 - 3.8703z^7 + 7.9661z^6 - 10.6337z^5 + 10.0678z^4 - 6.8080z^3 + 3.3529z^2 - 1.002z + 0.1666}$$

Example 5.18 Using bilinear transformation, design a digital bandpass Chebyshev Type 1 filter with the following specifications:

Lower passband edge frequency: 200 Hz

Upper passband edge frequency: 400 Hz

Lower stopband edge frequency: 100 Hz

Upper stopband edge frequency: 500 Hz

Passband ripple: 1 dB

Stopband ripple: 10 dB

Assume a suitable sampling frequency.

Solution The prewarped analog frequencies, as well as the values of A_1 and A_2 , are the same as for the above example. Hence, for the prototype analog lowpass filter, the specifications are

$$\Omega_p = 1, \Omega_s = \min\{|A_1|, |A_2|\} = 1.90258, \alpha_p = 1 \text{ dB}, \alpha_s = 10 \text{ dB}$$

Substituting these values in Eq. (5.21), the order of the filter is given as

$$N \geq \frac{\cosh^{-1} \sqrt{\frac{10^1 - 1}{10^{0.1} - 1}}}{\cosh^{-1}(1.90258)} = 1.9544$$

We choose $N = 2$. From (5.18), we have

$$\frac{1}{\varepsilon} = \frac{1}{\sqrt{10^{0.1\alpha_p} - 1}} = 1.965227$$

$$\sinh^{-1}\left(\frac{1}{\varepsilon}\right) = \sinh^{-1}(1.965227) = 1.428$$

Using Eqs. (5.25a) and (5.25b), the poles of the normalized lowpass transfer function are given by

$$p_k = -\sinh(0.714)\sin\left\{\frac{(2k-1)\pi}{4}\right\} + j \cosh(0.714)\cos\left\{\frac{(2k-1)\pi}{4}\right\}, \quad k = 1, 2$$

and

$$H_0 = \frac{1}{2} \frac{1}{\varepsilon} = 0.9826$$

Hence,

$$p_{1,2} = -0.54887 \pm j0.89513$$

Thus for $N = 2$, with a passband ripple of 1 dB, the normalized transfer function is

$$H_N(s) = \frac{0.9826}{s^2 + 1.0977s + 1.1025}$$

To arrive at the analog transfer function of the bandpass filter, we use in the above expression the lowpass-to-bandpass transformation given by Eq. (5.41), namely

$$s = \frac{s^2 + \hat{\Omega}_{p1}\hat{\Omega}_{p2}}{(\hat{\Omega}_{p2} - \hat{\Omega}_{p1})s} = \frac{s^2 + 0.236}{0.402s}$$

to obtain

$$H_{BP}(z) = \frac{0.1584s^2}{(s^4 + 0.4407s^3 + 0.6497s^2 + 0.1040s + 0.0557)}$$

We now use the bilinear transformation in the above to obtain the required digital bandpass filter transfer function as

$$H_{BP}(z) = \frac{0.0704z^4 - 0.1408z^2 + 0.0704}{z^4 - 1.9779z^3 + 2.2375z^2 - 1.3793z + 0.5158}$$

Example 5.19 Using bilinear transformation, design a digital bandstop Butterworth filter with the following specifications:

- Lower passband edge frequency: 35 Hz
- Upper passband edge frequency: 215 Hz
- Lower stopband edge frequency: 100 Hz
- Upper stopband edge frequency: 150 Hz
- Passband ripple: 3 dB
- Stopband ripple: 15 dB

Assume a suitable sampling frequency.

Solution Assuming a sampling frequency of 500 Hz, the normalized angular bandedge frequencies are given by

$$\omega_{p1} = 0.14\pi, \omega_{p2} = 0.86\pi, \omega_{s1} = 0.4\pi, \omega_{s2} = 0.6\pi$$

The prewarped analog frequencies are given by

$$\hat{\Omega}_{p1} = \tan(\omega_{p1}/2) = \tan(0.14\pi/2) = 0.2235$$

$$\hat{\Omega}_{p2} = \tan(\omega_{p2}/2) = \tan(0.86\pi/2) = 4.4737$$

$$\hat{\Omega}_{s1} = \tan(\omega_{s1}/2) = \tan(0.4\pi/2) = 0.7265$$

$$\hat{\Omega}_{s2} = \tan\left(\frac{\omega_{s2}}{2}\right) = \tan\left(\frac{0.6\pi}{2}\right) = 1.3764$$

We now obtain the corresponding specifications for the normalized analog lowpass filter using the lowpass-to-bandstop transformation. From Eq. (5.49b), we have

$$A_1 = \frac{\hat{\Omega}_{s1}\hat{\Omega}_{s2} - \hat{\Omega}_{p1}^2}{(\hat{\Omega}_{s2} - \hat{\Omega}_{s1})\hat{\Omega}_{p1}} = 6.5403, A_2 = \frac{\hat{\Omega}_{s1}\hat{\Omega}_{s2} - \hat{\Omega}_{p2}^2}{(\hat{\Omega}_{s2} - \hat{\Omega}_{s1})\hat{\Omega}_{p2}} = -6.5397$$

Now using (5.49a), we get the specifications for the normalized analog lowpass filter to be

$$\Omega_p = 1, \Omega_s = \min\{|A_1|, |A_2|\}, \alpha_p = 3 \text{ dB}, \alpha_s = 15 \text{ dB}$$

Substituting these values in Eq. (5.9), the order of the filter is given by

$$N \geq \frac{\log\left(\frac{10^{1.5}-1}{10^{0.3}-1}\right)}{2\log(6.5397)} = 0.9125$$

We choose $N = 1$. The transfer function of the first-order normalized Butterworth lowpass filter is

$$H_N(s) = \frac{1}{(s+1)}$$

Substituting the values of Ω_s and N in Eq. (5.6), we obtain

$$\left(\frac{6.5397}{\Omega_c}\right)^2 = 10^{1.5} - 1$$

Solving for Ω_c , we get $\Omega_c = 1.1818$. The analog transfer function of the lowpass filter is obtained from $H_N(s)$ by substituting $s = \frac{s}{\Omega_c} = \frac{s}{1.1818}$

$$H_{LP}(s) = \frac{1.1818}{s + 1.1818}$$

To arrive at the analog transfer function of the bandstop filter, we use in the above expression the lowpass-to-bandstop transformation given by (5.49c), namely

$$s = \frac{(\hat{\Omega}_{s2} - \hat{\Omega}_{s1})\Omega_s s}{s^2 + \hat{\Omega}_{s1}\hat{\Omega}_{s2}} = \frac{(0.6499)(6.5397)s}{s^2 + 1} = \frac{4.25s}{s^2 + 1}$$

to obtain

$$H_{BS}(s) = \frac{s^2 + 1}{s^2 + 3.5964s + 1}$$

The transfer function of the required digital bandstop filter is now obtained by using the bilinear transformation:

$$\begin{aligned} H_{BS}(z) &= H_{BS}(s) \Big|_{s=\frac{z-1}{z+1}} \\ &= \frac{0.3574z^2 + 0.3574}{z^2 - 0.2853} \end{aligned}$$

Example 5.20 Design an elliptic IIR digital highpass filter with the specifications given in Example 5.16.

Solution Normalized angular bandedge frequencies are given as

$$\omega_s = \frac{2\pi \times 500}{2000} = 0.5\pi; \quad \omega_p = \frac{2\pi \times 700}{2000} = 0.7\pi$$

Prewarping these frequencies, we get

$$\hat{\Omega}_p = \tan\left(\frac{\omega_p}{2}\right) = 1.9626105, \quad \hat{\Omega}_{s1} = \tan\left(\frac{\omega_s}{2}\right) = 1$$

For the prototype analog lowpass filter

$$\Omega_p = 1, \quad \Omega_s = \hat{\Omega}_p / \hat{\Omega}_s = 1.9626105, \quad \alpha_p = 1 \text{ dB}, \quad \alpha_s = 40 \text{ dB}$$

From (5.30) and (5.32) to (5.34), we get

$$k = \frac{\Omega_p}{\Omega_s} = \frac{1}{1.9626105} = 0.5095; \quad k' = \sqrt{1 - k^2} = 0.8605;$$

$$\rho_0 = \frac{1 - \sqrt{k'}}{2(1 + \sqrt{k'})} = 0.0188$$

$$\rho = \rho_0 + 2(\rho_0)^5 + 15(\rho_0)^9 + 150(\rho_0)^{13} = 0.0188$$

Substituting these values in Eq. (5.31), the order of the filter is given by

$$N \cong \frac{\log\left(16 \times \frac{10^{0.1N_s} - 1}{10^{0.1N_p} - 1}\right)}{\log_{10}(1/\rho)} = \frac{\log_{10}\left(16 \times \frac{10^4 - 1}{10^{0.1} - 1}\right)}{\log_{10}(1/0.0188)} = 3.3554$$

Let us choose $N = 4$. Then from Table 5.4, we have

$$H_N(s) = \frac{0.01s^4 + 0.1502s^2 + 0.3220}{s^4 + 0.9391s^3 + 1.5137s^2 + 0.8037s + 0.3612}$$

To arrive at the analog transfer function of the highpass filter, the variable s in the above-normalized transfer function is to be replaced by $(\hat{\Omega}_p/s) = (1.9626105/s)$

$$H_{HP}(s) = \frac{0.322s^4 + 0.5785s^2 + 0.1484}{0.3612s^4 + 1.5774s^3 + 5.8305s^2 + 7.0993s + 14.8367}$$

Then, the required highpass filter in the digital domain is given by

$$H_{HP}(z) = H_{HP}(s)|_{s=\frac{z-1}{z+1}}$$

$$H_{HP}(z) = \frac{0.035z^4 - 0.0234z^3 + 0.0561z^2 - 0.0234z + 0.035}{z^4 + 2.321z^3 + 2.6772z^2 + 1.5774z + 0.4158}$$

5.3 Design of Digital Filters Using Digital-to-Digital Transformations

In the design of analog filters, we start with designing a normalized lowpass filter, and then through an appropriate frequency transformation of the lowpass filter, the filter for the given magnitude response specifications is obtained. We can adopt a similar procedure by first designing a digital lowpass filter and then applying frequency transformation $z \rightarrow g(\hat{z})$ in the discrete domain to obtain highpass, bandpass,

bandstop, or another lowpass filter. The transformation function $g(z)$ has to satisfy certain conditions in order to produce the desired magnitude specifications.

- (i) The transformation function $g(\hat{z})$ should be a rational function of \hat{z} .
- (ii) The transformation $z \rightarrow g(\hat{z})$ should map the interior of the unit circle in the z -plane into the interior of the unit circle in the \hat{z} -plane, the exterior to the exterior, and the unit circle in the z -plane into the unit circle in the \hat{z} -plane. Hence, the transformed filter resulting from a stable filter will remain stable.

Table 5.6 shows a set of transformations that can be used to for this purpose, and interested readers may refer to the work of Constantinides [9] for details. For illustration, we consider the cases of lowpass-to-lowpass and lowpass-to-highpass transformations.

Lowpass-to-lowpass transformation

Consider the transformation function

$$z = \frac{\hat{z} - b}{1 - b\hat{z}} = g(\hat{z}) \tag{5.69}$$

where b is real. Then,

$$e^{j\omega} = \frac{e^{j\hat{\omega}} - b}{1 - be^{j\hat{\omega}}}$$

Table 5.6 Digital-to-digital transformations

Filter type	Spectral transformation	Design parameters
Lowpass	$z = \frac{\hat{z} - b}{1 - b\hat{z}}$	$b = \frac{\sin\left(\frac{\omega_p - \hat{\omega}_p}{2}\right)}{\sin\left(\frac{\omega_p + \hat{\omega}_p}{2}\right)}$ $\hat{\omega}_p$ = new passband edge frequency
Highpass	$z = \frac{\hat{z} + b}{1 + b\hat{z}}$	$b = \frac{\cos\left(\frac{\omega_p + \hat{\omega}_p}{2}\right)}{\cos\left(\frac{\omega_p - \hat{\omega}_p}{2}\right)}$ $\hat{\omega}_p$ = new passband edge frequency
Bandpass	$z = \frac{\hat{z}^2 - \frac{2a}{b+1}\hat{z} + \frac{b-1}{b+1}}{\frac{b-1}{b+1}\hat{z}^2 - \frac{2a}{b+1}\hat{z} + 1}$	$a = \frac{\cos\left(\frac{\hat{\omega}_{p2} + \hat{\omega}_{p1}}{2}\right)}{\cos\left(\frac{\hat{\omega}_{p2} - \hat{\omega}_{p1}}{2}\right)}$ $b = \cot\left(\frac{\hat{\omega}_{p2} - \hat{\omega}_{p1}}{2}\right) \tan\left(\frac{\omega_p}{2}\right)$ $\hat{\omega}_{p2}, \hat{\omega}_{p1}$ = desired upper and lower passband edge frequencies
Bandstop	$z = \frac{\hat{z}^2 - \frac{2a}{1+b}\hat{z} + \frac{1-b}{1+b}}{\frac{1-b}{1+b}\hat{z}^2 - \frac{2a}{1+b}\hat{z} + 1}$	$a = \frac{\cos\left(\frac{\hat{\omega}_{s2} + \hat{\omega}_{s1}}{2}\right)}{\cos\left(\frac{\hat{\omega}_{s2} - \hat{\omega}_{s1}}{2}\right)}$ $b = \tan\left(\frac{\hat{\omega}_{s2} - \hat{\omega}_{s1}}{2}\right) \tan\left(\frac{\omega_s}{2}\right)$ $\hat{\omega}_{s2}, \hat{\omega}_{s1}$ = desired upper and lower stopband edge frequencies

Note ω_p is the passband edge frequency of the normalized lowpass filter $H_N(z)$

Let the passband edge of the original lowpass filter be ω_p and that of the desired lowpass filter be $\hat{\omega}_p$. It can easily be seen that $g(\hat{z})$ maps $\hat{\omega} = 0$ into $\omega = 0$, and $\hat{\omega} = \pm\pi$ into $\omega = \pm\pi$. We now choose the value of b so that $g(\hat{z})$ maps the frequency $\hat{\omega}_p$ to ω_p ; then, we will have the required mapping function. Hence, we should have

$$e^{j\omega_p} = \frac{e^{j\hat{\omega}_p} - b}{1 - be^{j\hat{\omega}_p}}$$

Hence,

$$b = \frac{e^{-j(\omega_p - \hat{\omega}_p)/2} - e^{j(\omega_p - \hat{\omega}_p)/2}}{e^{-j(\omega_p + \hat{\omega}_p)/2} - e^{j(\omega_p + \hat{\omega}_p)/2}} = \frac{\sin(\omega_p - \hat{\omega}_p)/2}{\sin(\omega_p + \hat{\omega}_p)/2} \quad (5.70)$$

Thus, the transformation (5.69) with the value of b given by (5.70) will transform a digital lowpass filter with passband edge at ω_p into another digital lowpass filter with its passband edge at $\hat{\omega}_p$.

Lowpass-to-highpass transformation

Consider the transformation function

$$z = -\frac{\hat{z} + b}{1 + b\hat{z}} = g(\hat{z}) \quad (5.71)$$

where b is real. Then,

$$e^{j\omega} = -\frac{e^{j\hat{\omega}} + b}{1 + be^{j\hat{\omega}}}$$

Let the passband edge of the original lowpass filter be ω_p and that of the desired highpass filter be $\hat{\omega}_p$. It can easily be seen that $g(\hat{z})$ maps $\hat{\omega} = 0$ into $\omega = \pm\pi$, and $\hat{\omega} = \pm\pi$ into $\omega = 0$. We now choose the value of b so that $g(\hat{z})$ maps the frequency $\hat{\omega}_p$ to $-\omega_p$; then, we will have the required mapping function. Hence, we should have

$$e^{-j\omega_p} = -\frac{e^{j\hat{\omega}_p} + b}{1 + be^{j\hat{\omega}_p}}$$

Hence,

$$b = \frac{e^{-j(\omega_p + \hat{\omega}_p)/2} + e^{j(\omega_p + \hat{\omega}_p)/2}}{e^{-j(\omega_p - \hat{\omega}_p)/2} + e^{j(\omega_p - \hat{\omega}_p)/2}} = \frac{\cos(\omega_p + \hat{\omega}_p)/2}{\cos(\omega_p - \hat{\omega}_p)/2} \quad (5.72)$$

Thus, the transformation (5.71) with the value of b given by (5.72) will transform a digital lowpass filter with passband edge at ω_p into a digital highpass filter with its passband edge at $\hat{\omega}_p$.

Similarly, the other transformations given in Table 5.6 can be established [Con70].

These transformations can easily be applied to obtain highpass, bandpass, band reject, or another lowpass filter as follows:

- Step 1 Find the normalized transfer function $H_N(z)$ of a lowpass filter using an approximation technique
- Step 2 Obtain the passband edge ω_p in $H_N(z)$
- Step 3 Find the function $H(z)$ from $H_N(z)$ using the appropriate transformation from Table 5.6.

An important aspect of the filters designed using the above transformations is that the passband edge of the lowpass or the highpass filter can be varied by varying the single parameter b . Similarly, in the case of bandpass or bandstop filters, both the lower and upper passband edges can be varied by varying two parameters, namely a and b [9, 10].

Example 5.21 Consider the second-order lowpass digital filter of Example 5.7 with -3 dB cutoff frequency of 0.2π . Redesign this lowpass filter by applying the lowpass-to-lowpass digital transformation so that the -3 dB cutoff frequency moves from 0.2π to 0.3π .

Solution Since $\omega_p = 0.2\pi$ and $\hat{\omega}_p = 0.3\pi$, we obtain

$$b = \frac{\sin[(\omega_p - \hat{\omega}_p)/2]}{\sin[(\omega_p + \hat{\omega}_p)/2]} = \frac{\sin[(0.2\pi - 0.3\pi)/2]}{\sin[(0.2\pi + 0.3\pi)/2]} = -0.2212$$

From the solution of Example 5.7, the digital transfer function with -3 dB cutoff frequency at 0.2π is

$$H(z) = \frac{0.068(z+1)^2}{z^2 - 1.142z + 0.413}$$

Hence, the desired low pass transfer function with -3 dB cutoff frequency at 0.3π is given by

$$\begin{aligned} H(z) &= \frac{0.068(z+1)^2}{z^2 - 1.142z + 0.413} \Bigg|_{z = \frac{z+0.2212}{1+0.2212z}} \\ &= \frac{0.1321(z+1)^2}{z^2 - 0.7467z + 0.2727} \end{aligned}$$

Example 5.22 Consider the design of a highpass filter by applying lowpass-to-highpass digital transformation to the second-order lowpass digital filter of Example 5.11. The desired passband edge frequency is 0.5π .

Solution To apply the digital lowpass-to-highpass transformation shown in Table 5.6, b is first computed as

$$b = \frac{\cos[(\omega_p + \hat{\omega}_p)/2]}{\cos[(\omega_p - \hat{\omega}_p)/2]} = \frac{-\cos[(0.25\pi + 0.5\pi)/2]}{\cos[(0.25\pi - 0.5\pi)/2]} = -0.4142$$

From the solution of Example 5.11, the digital transfer function with passband edge frequency at 0.25π is

$$H(z) = \frac{0.2195z^2 + 0.439z + 0.2195}{z^2 - 0.31047z + 0.1887}$$

Hence, the desired highpass transfer function with passband edge frequency at 0.5π is given by

$$\begin{aligned} H(z) &= \left. \frac{0.2195z^2 + 0.439z + 0.2195}{z^2 - 0.31047z + 0.1887} \right|_{z = \frac{0.4142-z}{1-0.4142z}} \\ &= \frac{0.4857z^2 - 0.9714z + 0.4857}{z^2 - 0.6871z + 0.2564} \end{aligned}$$

5.4 Design of IIR Digital Filters Using MATLAB

Various types of M-files are included in the signal processing toolbox of MATLAB software for the design of IIR digital filters. The use of these M-files is illustrated by the following examples:

Example 5.23 An IIR digital lowpass filter is required to meet the following specifications:

Passband ripple	≤ 0.5 dB
Passband edge	1.2 kHz
Stopband attenuation	≥ 40 dB
Stopband edge	2 kHz
Sample rate	8 kHz

Design a (i) digital Butterworth filter, (ii) Type 1 Chebyshev digital filter, (iii) Type 2 Chebyshev digital filter, and (iv) digital elliptic filter.

Solution The following MATLAB program is used to design the required filters.

Program 5.6 Butterworth, Chebyshev, and elliptic IIR lowpass filter design

```

flag = input('enter 1 for BWF, 2 for Type 1 CSF, 3 for Type 2 CSF, 4 for
Ellip = ');
%BWT stands for Butterworth filter, CSF for Chebyshev filter, and Ellip for
%Elliptic filter
Wp = input('Normalized passband edge = ');
Ws = input('Normalized stopband edge = ');
Rp = input('Passband ripple in dB = ');
Rs = input('Minimum stopband attenuation in dB = ');
if flag ==1
[N,Wn] = buttord(Wp,Ws,Rp,Rs)
[b,a] = butter(N,Wn);
end
if flag ==2
[N,Wn] = cheb1ord(Wp,Ws,Rp,Rs)
[b,a] = cheby1(N,Rp,Wn);
end
if flag ==3
[N,Wn] = cheb2ord(Wp,Ws,Rp,Rs)
[b,a] = cheby2(N,Rs,Wn);
end
if flag ==4
[N,Wn] = ellipord(Wp,Ws,Rp,Rs)
[b,a] = ellip(N,Rp,Rs,Wn);
end
[h,omega] = freqz(b,a,256);
plot (omega/pi,20*log10(abs(h)));grid;
xlabel('\omega/\pi'); ylabel('Gain, dB');

```

The magnitude responses of the designed lowpass filters are shown in Fig. 5.16.

Example 5.24 An IIR digital highpass filter is required to meet the following specifications:

Passband ripple	≤ 1 dB
Passband edge	800 Hz
Stopband attenuation	≥ 60 dB
Stopband edge	400 Hz
Sample rate	2000 Hz

Design (i) a digital Butterworth filter, (ii) Type 1 Chebyshev digital filter, (iii) Type 2 Chebyshev digital filter, and (iv) digital elliptic filter

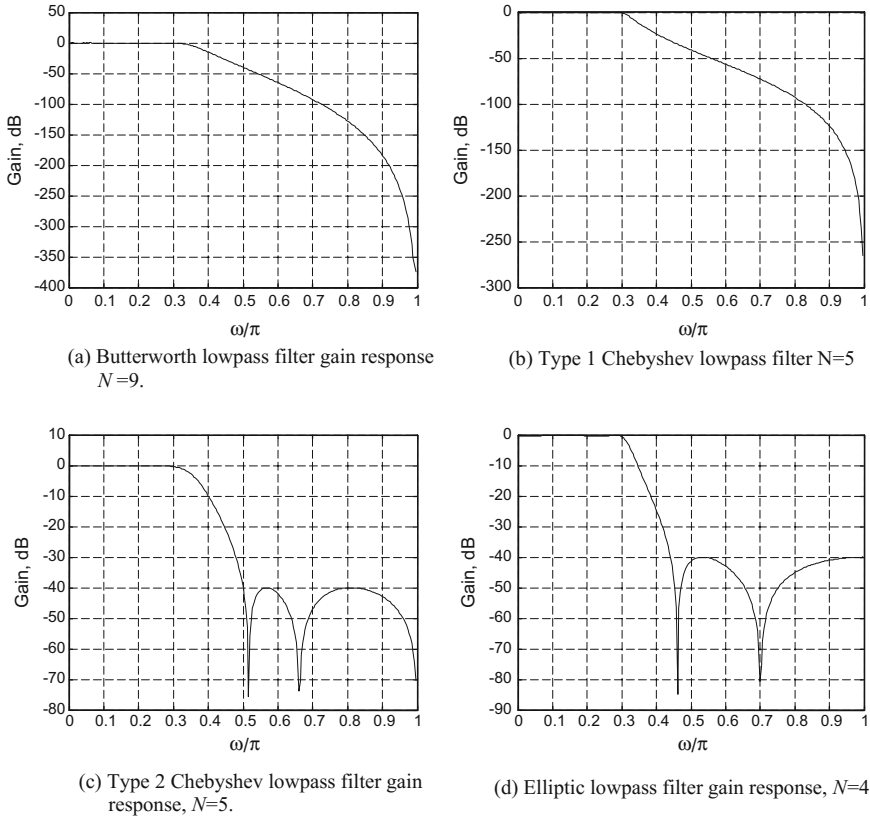


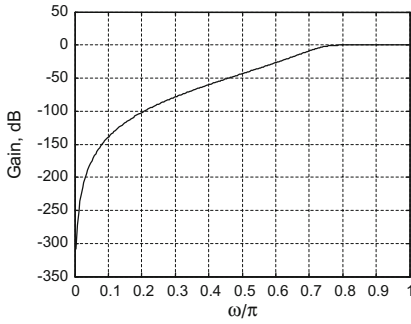
Fig. 5.16 Lowpass filter magnitude responses

Solution Program 5.6 can be used to design highpass filters with the following MATLAB functions for determining the coefficients b and a .

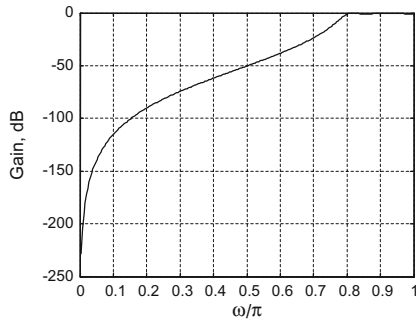
$$\begin{aligned}
 [b,a] &= \text{butter}(N,Wn, 'high'); & [b,a] &= \text{cheby1}(N,Rp,Wn, 'high'); \\
 [b,a] &= \text{cheby2}(N,Rs,Wn, 'high'); & [b,a] &= \text{ellip}(N,Rp,Rs,Wn, 'high');
 \end{aligned}$$

The gain responses and filter orders for the Butterworth, Type 1 and Type 2 Chebyshev, and elliptic filters are shown in Fig. 5.17a–d, respectively.

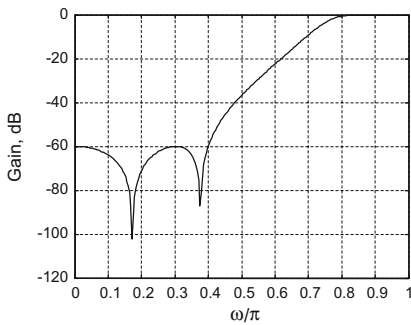
Example 5.25 Design Butterworth, Type 1 Chebyshev bandpass, Type 2 Chebyshev, and elliptic bandpass digital filters satisfying the following specifications:



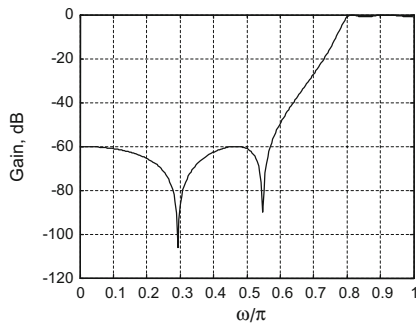
(a) Butterworth highpass filter gain response, $N=6$.



(b) Type 1 Chebyshev highpass filter gain response, $N=4$.



(c) Type 2 Chebyshev highpass filter gain response, $N=4$.



(d) Elliptic highpass filter gain response, $N=4$.

Fig. 5.17 Highpass filter magnitude responses

- Lower passband edge $\omega_{p1} = 0.4\pi$ rad
- Upper passband edge $\omega_{p2} = 0.6\pi$ rad
- Lower stopband edge $\omega_{s1} = 0.25\pi$ rad
- Upper stopband edge $\omega_{s2} = 0.7\pi$ rad
- Passband ripple = 0.5 dB
- Stopband attenuation = 45 dB

Solution The following MATLAB program is used to design the desired filters.

Program 5.7 Butterworth, Chebyshev, and elliptic IIR bandpass Filters Design

```

flag = input('enter 1 for BWF, 2 for Type 1 CSF, 3 for Type 2 CSF, 4 for
Ellip = ');
%BWT stands for Butterworth filter, CSF for Chebyshev filter, and Ellip for
%Elliptic filter
Wp1 = input('Normalized lower passband edge = ');
Wp2 = input('Normalized upper passband edge = ');
Ws1 = input('Normalized lower stopband edge = ');
Ws2 = input('Normalized upper stopband edge = ');
Rp = input('Passband ripple in dB = ');
Rs = input('Minimum stopband attenuation in dB = ');
if flag ==1
[N,Wn] = buttord([Wp1 Wp2],[Ws1 Ws2],Rp,Rs);
[b,a] = butter(N,Wn);
end
if flag ==2
[N,Wn] = cheb1ord([Wp1 Wp2],[Ws1 Ws2],Rp,Rs);
[b,a] = cheby1(N,Rp,Wn);
end
if flag ==3
[N,Wn] = cheb2ord([Wp1 Wp2],[Ws1 Ws2],Rp,Rs);
[b,a] = cheby2(N,Rs,Wn);
end
if flag ==4
[N,Wn] = ellipord([Wp1 Wp2],[Ws1 Ws2],Rp,Rs);
[b,a] = ellip(N,Rp,Rs,Wn);
end
[h,omega] = freqz(b,a,256);
plot (omega/pi,20*log10(abs(h)));
grid;
xlabel('\omega/\pi');
ylabel('Gain, dB');

```

The gain responses and filter orders for the designed filters are shown in Fig. 5.18.

Example 5.26 Design Butterworth, Type 1 Chebyshev bandstop, Type 2 Chebyshev, and elliptic bandstop digital filter satisfying the following specifications:

Lower passband edge $\omega_{p1} = 0.1\pi$ rad

Lower stopband edge $\omega_{s1} = 0.2\pi$ rad

Upper passband edge $\omega_{p2} = 0.5\pi$ rad

Upper stopband edge $\omega_{s2} = 0.4\pi$ rad

Passband ripple 1 dB

Stopband attenuation 40 dB

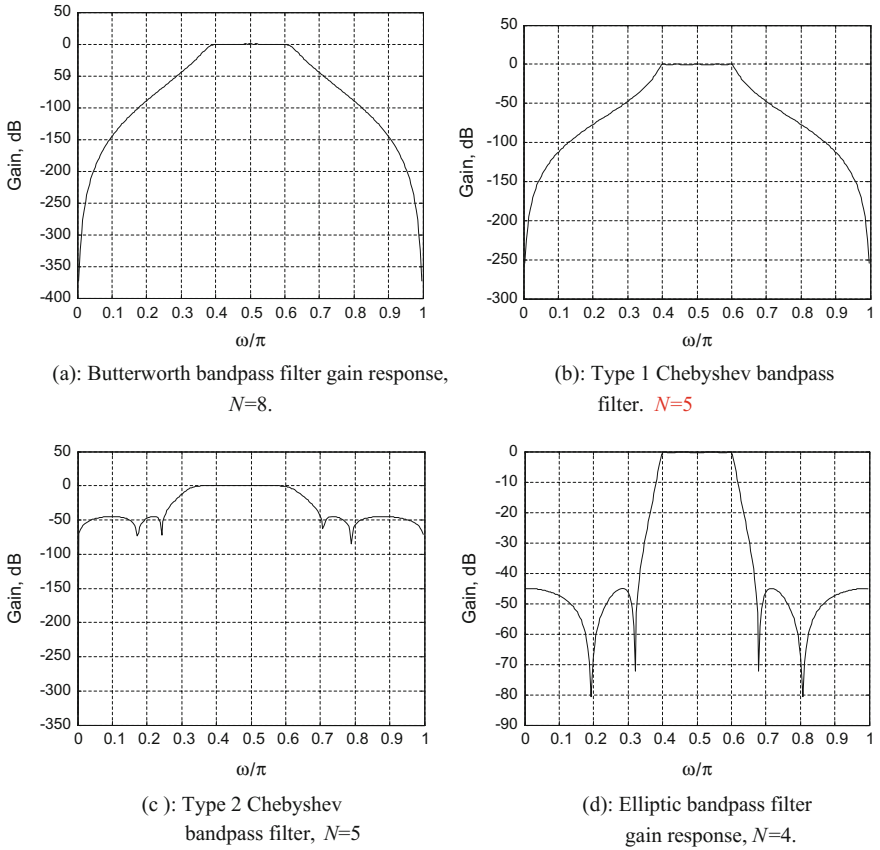


Fig. 5.18 Bandpass filter magnitude responses

Solution Program 5.7 can be used to design bandstop filters with the following MATLAB functions for determining the coefficients b and a .

```
[b,a] = butter(N,Wn,'stop');           [b,a] = cheby1(N,Rp,Wn,'stop');
[b,a] = cheby2(N,Rs,Wn,'stop');       [b,a] = ellip(N,Rp,Rs,Wn,'stop');
```

The gain responses and filter orders for the designed filters are shown in Fig. 5.19.

Example 5.27 Design a filter using digital-to-digital transformation as required in Example 5.21.

Solution To design the desired lowpass filter, the MATLAB command 'iirlp2lp' can be used for digital lowpass-to-lowpass transformation. The following MATLAB program is used to design the desired filter.

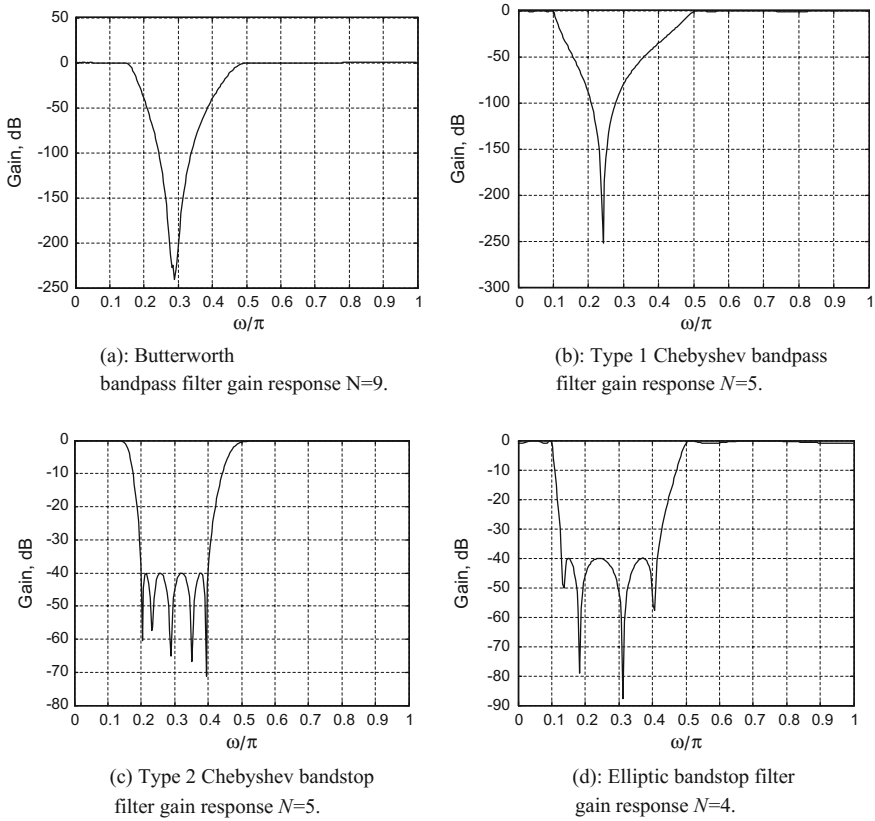


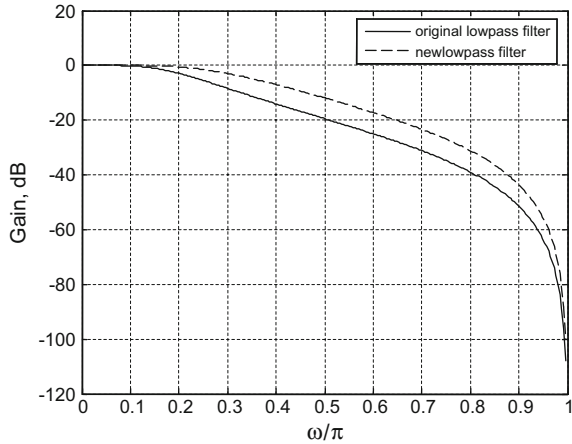
Fig. 5.19 Bandstop filter magnitude responses

Program 5.8 Digital lowpass-to-lowpass transformation

```
clear all;
b1 = 0.068*[1 2 1]; % numerator coefficients of original LPF
a1 = [1-1.142 0.413]; % denominator coefficients of original LPF
[num,den,anum,aden] = iirlp2lp(b1,a1,0.2,0.3); % coefficients of new LPF
[h1,omega] = freqz(b1,a1,256);
plot(omega/pi,20*log10(abs(h1)));
hold on
[h2,omega] = freqz(num,den,256);
plot(omega/pi,20*log10(abs(h2)),'-');
xlabel('\omega/\pi'); ylabel('Gain, dB');
legend('original lowpass filter','newlowpass filter');grid;
```

The magnitude responses of the original filter and the new transformed filter are shown in Fig. 5.20. From this figure, it is observed that the requirements of the

Fig. 5.20 Magnitude responses of the original and new lowpass filters



original and the transformed filters are fulfilled. The coefficients of the lowpass filter obtained using the above program are equal to those obtained in Example 5.21.

Example 5.28 Design a digital highpass filter using digital-to-digital transformation as required in Example 5.13.

Solution The following MATLAB program is used to design the desired filter.

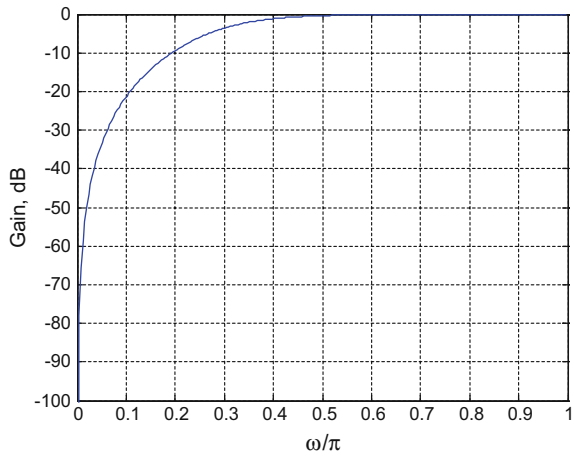
Program 5.9 Digital lowpass-to-highpass transformation

```
clear all;
b1 = [0.2195 0.439 0.2195]; % numerator coefficients of prototype lowpass filter
HN (z)
a1 = [1 -0.31047 0.1887]; % denominator coefficients of prototype lowpass filter
HN (z)
[num,den,anum,aden] = iirlp2hp(b1,a1,0.25,0.5); % coefficients of desired highpass
filter H(z)
[h,omega] = freqz(num,den,256);
plot (omega/pi,20*log10(abs(h)));
xlabel('\omega/\pi'); ylabel('Gain, dB');grid;
```

The magnitude response of the desired filter is shown in Fig. 5.21.

From this figure, it is observed that the requirements of the transformed highpass filter are fulfilled. The coefficients of the highpass filter obtained using the above program are equal to those obtained in Example 5.22.

Fig. 5.21 Magnitude response of the highpass filter



5.5 Design of IIR Filters Using MATLAB GUI Filter Designer SPTOOL

With the aid of MATLAB GUI filter designer SPTOOL, the filter satisfying the specifications can be designed using the following procedure [11]:

- Step 1 Access the MATLAB’s GUI filter designer SPTOOL for the design of both FIR and IIR filters.
From MATLAB, type the following: `sptool`
- Step 2 From the startup window `startup.spt`, select a new design and enter the specifications of the filter. Then, the MATLAB’s filter designer SPTOOL window with the characteristics of the designed filter is displayed.
- Step 3 When finished, access the startup window again. Select `→ Edit → Name`. Change name (enter new variable name).
- Step 4 Select `File → Export → Export to workspace` the new variable name
- Step 5 Access MATLAB’s workspace and type the following commands:
 - `new variable name.tf.num;`
 - `round (new variable name.tf.num*2^15).`

Example 5.29 Design an IIR lowpass digital filter using the bilinear transformation for the following specifications using (i) Butterworth, (ii) Chebyshev Type 1, and (iii) elliptic approximations:

Passband ripple	≤ 1 dB
Passband edge	4 kHz
Stopband attenuation	≥ 40 dB
Stopband edge	6 kHz
Sample rate	24 kHz

Solution It can be designed by following the above stepwise procedure. After the execution of Steps 1 and 2, the SPTOOL in MATLAB7.0 for Butterworth lowpass filter, the filter characteristics displayed by the window is shown in Fig. 5.22.

From Fig. 5.22, the filter order obtained with MATLAB GUI filter designer SPTOOL is 10. The execution of Steps 3, 4, and 5 will display the designed filter coefficients and coded coefficients. Similarly, execution of the SPTOOL in MATLAB7.0 for Chebyshev Type 1 filter displays the filter characteristics as shown in Fig. 5.23. The order of the filter found to be 6. The execution of the SPTOOL for elliptic filter displays the filter characteristics as shown in Fig. 5.24. The order of the elliptic filter is found to be 4.

Example 5.30 Design a bandstop IIR elliptic digital filter operating at sampling frequency of 2 kHz with the passband edges at 300 and 750 Hz, stopband edges at 450 and 650 Hz, peak passband ripple of 0.5 dB, and minimum stopband attenuation of 30 dB. Use Bilinear transformation method to obtain the transfer function $H(z)$.

Solution Following the stepwise procedure used in the above example and execution of the SPTOOL in MATLAB7.0 for elliptic bandstop filter, the filter characteristics are shown in Fig. 5.25. The order of the designed filter is observed to be 6.

5.6 Design of Specialized Digital Filters by Pole-Zero Placement

There are certain specialized filters often used in digital signal processing applications in addition to the filters designed in the previous sections. These specialized filters can be directly designed based on placement of poles and zeros.

5.6.1 Notch Filter

The notch filter removes a single frequency f_0 , called the notch frequency. The magnitude of the notch filter at f_0 can be made zero by placing a zero on the unit circle with $\omega_0 = (2\pi f_0)/F_T$ corresponding to the notch frequency, where F_T is the sampling frequency. The bandwidth B_w of the notch filter can be controlled by placing pole at the same angle with the pole radius $r < 1$. The poles and zeros should occur in complex conjugate pairs. As such, the transfer function of a second-order notch filter can be formed as

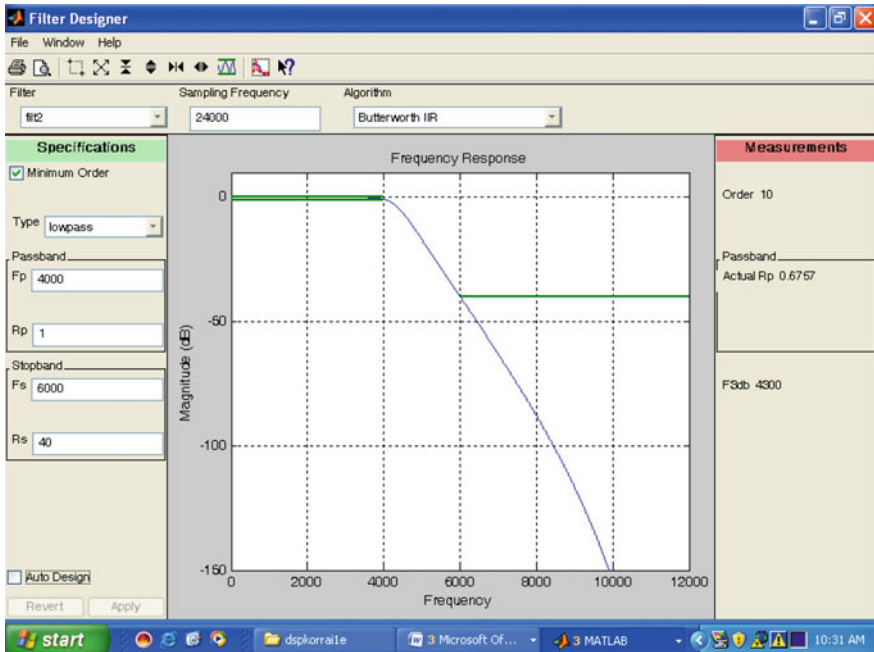


Fig. 5.22 Magnitude response of Butterworth lowpass filter

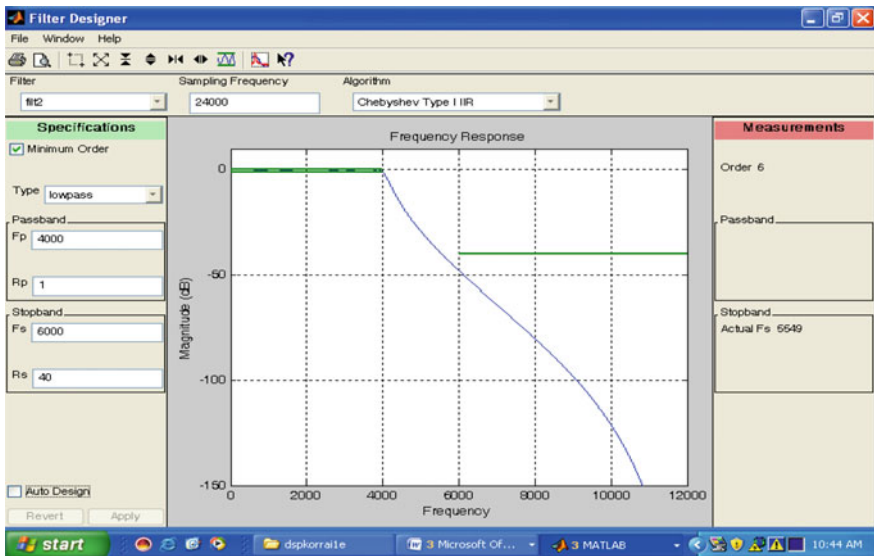


Fig. 5.23 Magnitude response of Chebyshev Type 1 lowpass filter

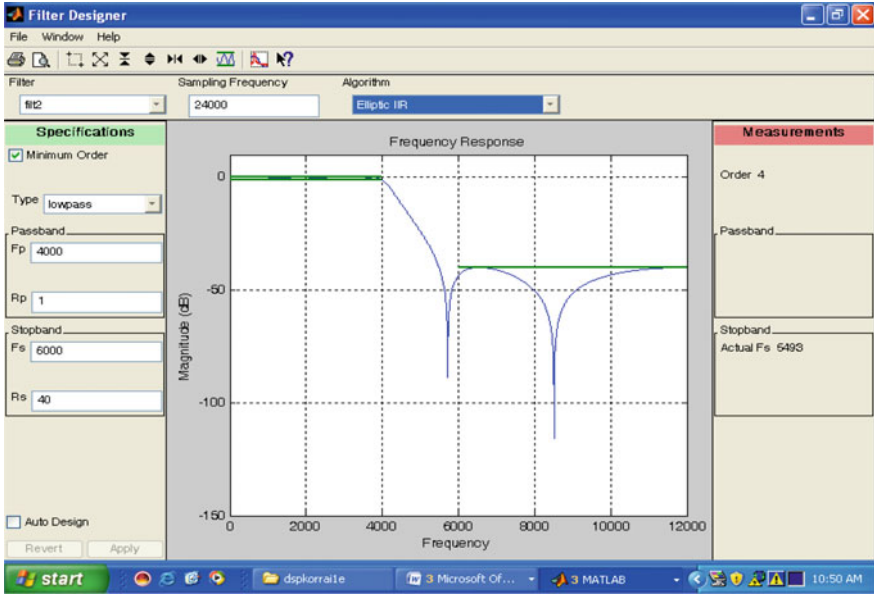


Fig. 5.24 Magnitude response of elliptic lowpass filter

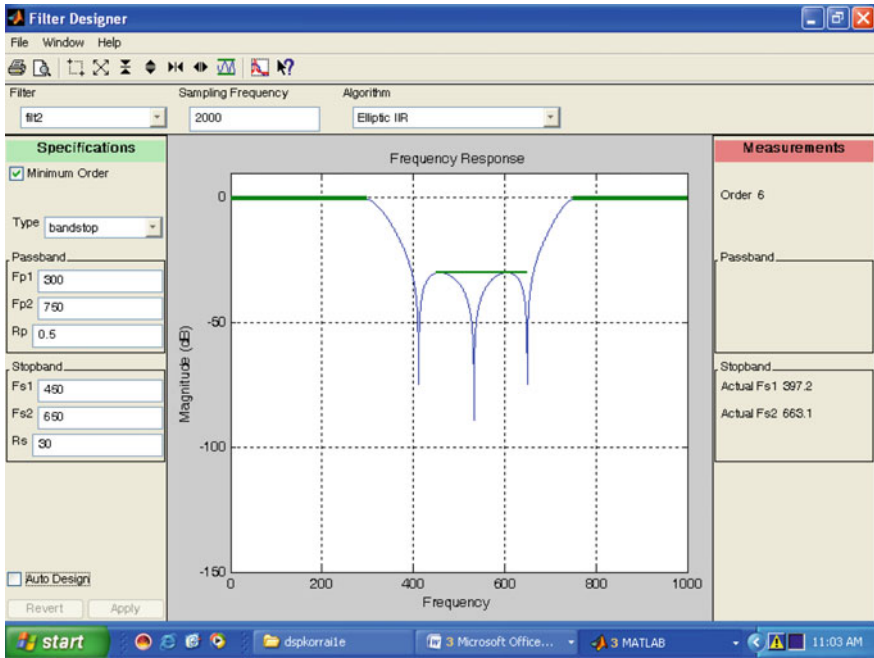


Fig. 5.25 Magnitude response of elliptic bandstop filter

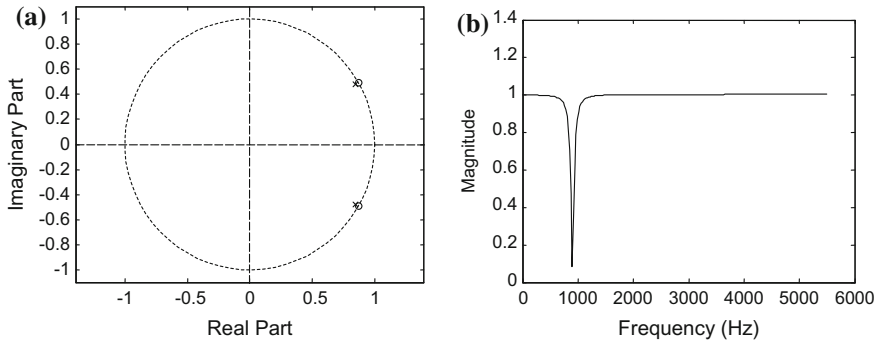


Fig. 5.26 a Pole-zero plot and b magnitude response of notch filter of Example 5.31

$$H(z) = \frac{b_0(z - e^{j\omega_0})(z - e^{-j\omega_0})}{(z - r e^{j\omega_0})(z - r e^{-j\omega_0})} \tag{5.73}$$

which can be rewritten as

$$H(z) = \frac{b_0\{z^2 - (2 \cos \omega_0)z + 1\}}{z^2 - (2r \cos \omega_0)z + r^2} \tag{5.74}$$

To ensure that the passband gain is unity, the gain factor b_0 is to be chosen so that $|H(1)| = 1$. Hence, b_0 is given by

$$b_0 = \frac{|1 - (2 \cos \omega_0) + r^2|}{|2 - 2 \cos \omega_0|} \tag{5.75}$$

If $B_w \ll F_T$, the pole radius r can be approximated [12] as

$$r = 1 - (\pi B_w)/F_T \tag{5.76}$$

The following example illustrates the design of a notch filter using MATLAB.

Example 5.31 Design a digital notch filter with notch frequency at 900 Hz, bandwidth of 100 Hz, and the sampling frequency of 11,025 Hz.

Solution The following MATLAB Program 5.10 is used to design the desired notch filter and the pole-zero plot, and the magnitude response of the filter obtained from the program is shown in Fig. 5.26a, b, respectively.

Program 5.10 Design of a notch filter

```

clear;clc;
FT = 11025;
f0 = 900;% Notch frequency
Bw = 100;% Bandwidth
% Compute filter coefficients
W0 = 2*pi*f0/FT;
r = 1 - (Bw*pi/ FT);% pole radius
b0 = abs(1-2*r*cos(W0) + r^2)/abs(2-2*cos(W0));% gain
b = b0*[1-2*cos(W0) 1];% Numerator polynomial coefficients of transfer function
a = [1-2*r*cos(W0) r^2];% denominator polynomial coefficients of transfer function
% pole-zero plot
[z,p,k] = tf2zp(b,a);
figure(1),zplane(z,p)
% Plot magnitude response
N = 240
[H,f] = freqz (b,a,N, FT);
A = abs(H);
figure(2),plot (f,A)
xlabel('Frequency (Hz)');ylabel('Magnitude');

```

5.6.2 Comb Filter

Comb filters have a wide range of practical applications such as suppression of interference in LORAN navigation systems [13] and separation of solar and lunar spectral components in ionospheric measurements [14]. Comb filter is a filter with multiple passbands and stopbands with periodic frequency response with periodicity of $(2\pi/N)$ where N is an integer. An N th-order comb filter can be designed by placing N zeros equally spaced on the unit circle and N poles equally spaced around a circle of radius $r < 1$, but close to the unit circle. Thus, the poles correspond to the N roots of r^N . Hence, the transfer function of an N th-order comb notch filter is given by

$$H(z) = \frac{b_0(z^N - 1)}{z^N - r^N} \quad (5.77)$$

And the transfer function for comb peaking filter is given by

$$H(z) = \frac{b_0(z^N + 1)}{z^N - r^N} \quad (5.78)$$

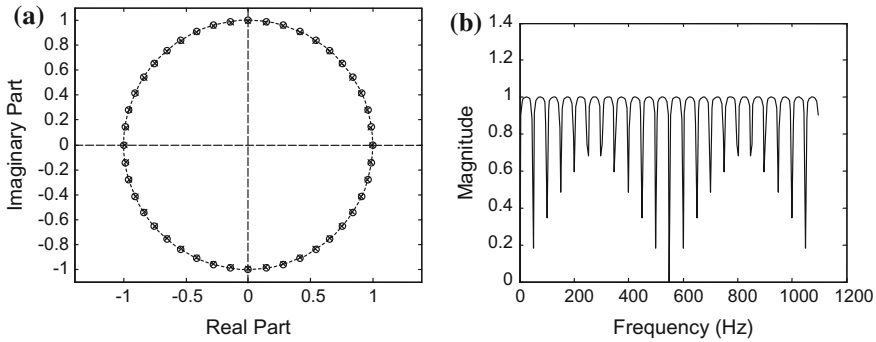


Fig. 5.27 a Pole-zero plot and b magnitude response of comb notching filter of Example 5.32

The gain constant b_0 is to be chosen so that the passband gain is unity at $f = \frac{f_0}{2}$, where $F_T = Nf_0$.

Hence, $b_0 = \frac{1+r^N}{2}$ for comb notching filter and $b_0 = \frac{1-r^N}{2}$ for comb peaking filter. The bandwidth B_w is related to the Q -factor of the filter by

$$B_w = \frac{2\pi f_0}{Q} \tag{5.79}$$

The following MATLAB command can be used to design a comb notching filter or a comb peaking filter

$$[b, a] = \text{ircomb}(N, B_w, \text{Type})$$

where b and a are the coefficients of the numerator and denominator polynomials of the transfer function of the comb filter, N is the order of the comb filter, B_w is the bandwidth of the comb filter, and *Type* specifies the notch or peak. The following example illustrates the design of comb notching filter using MATLAB.

Example 5.32 Design a comb notch filter to suppress 50 Hz hum of overhead fluorescent lights in biomedical measurements. Choose the sampling frequency of 2200 Hz and the Q -factor of the filter as 35.

Solution For this design, $f_0 = 50$, $F_T = 2200$. Hence, using Eqs. (5.79) and (5.76), we have

$$B_w = \left(\frac{100\pi}{35}\right) = 8.9760, \quad r = 1 - \frac{8.9760}{2200} = 0.9872$$

The following MATLAB Program 5.11 is used to design the desired notch filter and the pole-zero plot, and the magnitude response of the filter obtained from the program is shown in Fig. 5.27a, b, respectively.

Program 5.11 Design of a comb notch filter

```

clear;clc;
FT = 2200;
f0 = 50;% Notch frequency
Bw = 8.9760;,% Bandwidth
N = FT/f0;
[b,a] = iircomb(N,Bw/Fs,'notch');
[z,p,k] = tf2zp(b,a);
Figure (1),
zplane(z,p)
% Plot magnitude response
N = 240
[H,f] = freqz (b,a,N, FT);
A = abs(H);
figure(2)
plot (f,A)
xlabel('Frequency (Hz)');ylabel('Magnitude');

```

5.7 Some Examples of IIR Filters for Audio Processing Applications

5.7.1 *Suppression of Power Supply Hum in Audio Signals*

By and large most of the audio processing systems are affected by the interference caused by the power supply hum at 50 Hz or 60 Hz. This can be avoided by using a second-order notch filter whose transfer function is given by

$$H(z) = \frac{b_0z^2 + b_1z + b_2}{a_0z^2 + a_1z + a_2} \quad (5.80)$$

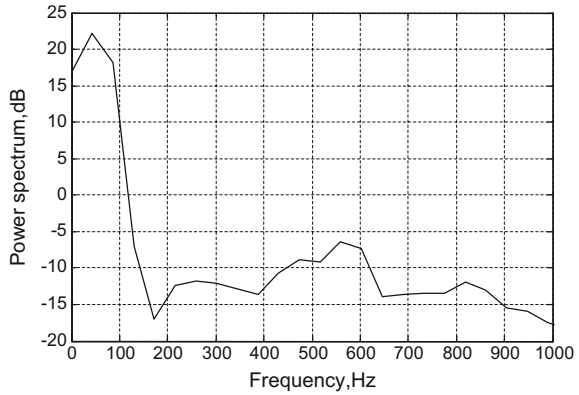
For example, consider a speech signal from the sound file 'DT.wav' [15]. The following MATLAB code is used to read the speech signal from the sound file, to add sinusoidal interference at 50 Hz to it, and to plot the power spectra of the interference added signal (x_n) as shown in Fig. 5.28.

```

[x, FT] = wavread('DT.wav');%Reads the wav file to obtain speech signal x and
sampling frequency FT
for i = 1:size(x)
xn(i) = x(i) + 2*sin(2*pi*50*i/ FT); % adds power supply hum at 50 Hz to the
speech signal
end
wavwrite(xn, FT, 'DTn50.wav');

```

Fig. 5.28 Power spectrum of the speech signal corrupted with power supply hum



```
[pxx,f] = psd(xn,256, F_T);% power spectrum of the speech signal corrupted with
power supply hum plot(f,10*log10(pxx));grid; xlabel('Frequency,Hz');ylabel
('Power spectrum,dB');
```

In Fig. 5.28, the peak at 50 Hz with large magnitude of the power spectrum is due to the power supply hum at 50 Hz. To suppress the power supply hum, a digital IIR second-order notch filter of the form given in Eq. (5.73) is designed for the notch frequency $f_0 = 50$ Hz and bandwidth $B_w = 100$ Hz. As such, the transfer function given by Eq. (5.80) becomes

$$H(z) = \frac{1.9716z^2 - 3.9415z + 1.9716}{z^2 - 1.9427z + 0.9438} \tag{5.81}$$

The magnitude response of the notch filter described by the above transfer function is shown in Fig. 5.29.

To recover the original speech signal, the corrupted signal is passed through the designed notch filter, and the power spectrum of the recovered signal is shown in

Fig. 5.29 Magnitude response of IIR digital notch filter

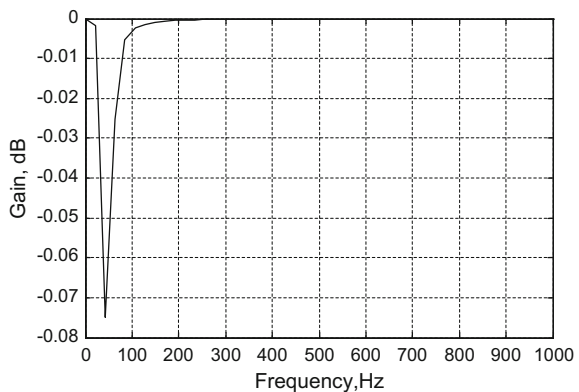


Fig. 5.30 Power spectrum of the recovered speech signal

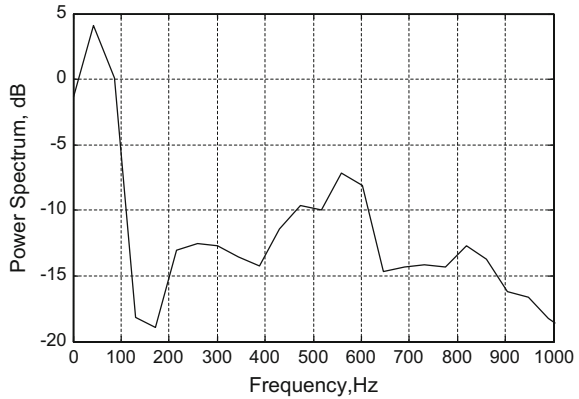


Fig. 5.30. From this figure, it can be observed that the peak at 50 Hz with large magnitude is suppressed and the speech signal is recovered. Also when the recovered signal is connected to a loudspeaker, its audio quality is observed to be the same, as that of the original speech signal.

5.7.2 Generation of Artificial Reverberations

The recorded sounds in a studio are unnatural to the listener, compared to the recorded sounds in a closed room. Digital filtering can be used to generate artificial reverberations, and by adding these reverberations to the studio recorded sounds as shown in Fig. 5.31, one can arrive at a pleasant-sounding reverberation.

An artificial reverberation generator, in general, is an interconnection scheme consisting of parallel connection of IIR filters in cascade with allpass reverberators as shown in Fig. 5.32.

The structures for IIR filters and allpass reverberators are shown in Fig. 5.33a, b, respectively.

Artificial reverberations can be generated by choosing different delays $d_i, i = 1, \dots, K + L$, and the multiplier constants $a_i, i = 1, \dots, K + L + 1$, and $b_i, i = 1, \dots, K$. For $K = 4$ and $L = 2$, the reverberation generator shown in

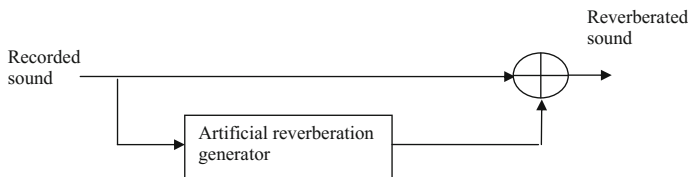


Fig. 5.31 Reverberated sound generation scheme

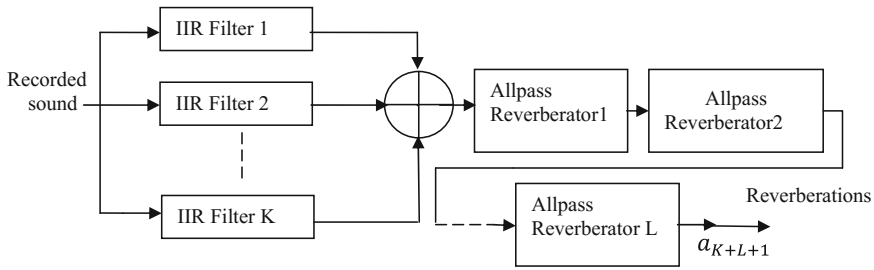


Fig. 5.32 Artificial reverberation generator

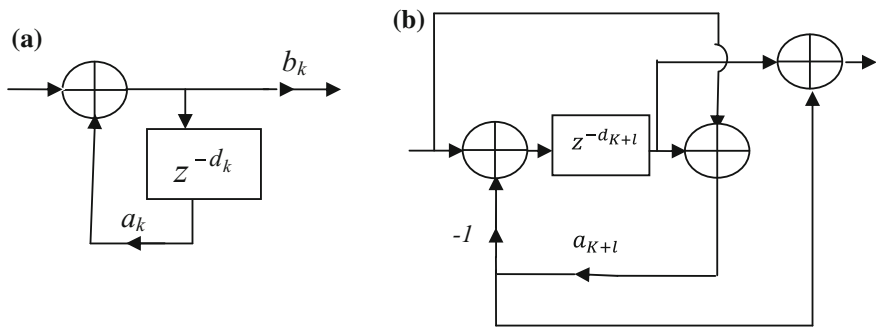


Fig. 5.33 **a** Structure of IIR filter ($k = 1, 2, \dots, K$) and **b** structure of allpass reverberator ($l = 1, 2, \dots, L$)

Fig. 5.31 corresponds to the Schroeder reverberator with four IIR filters and two allpass reverberators with delays given by

$$d_1 = 800; d_2 = 900; d_3 = 650; d_4 = 700; d_5 = 670; d_6 = 990;$$

and multiplier constants

$$a_1 = 0.8; a_2 = 0.4; a_3 = 0.2; a_4 = 0.1; a_5 = 0.7; a_6 = 0.9; a_7 = 0.6;$$

$$b_1 = 0.9; b_2 = 0.8; b_3 = 0.9; b_4 = 1.$$

The reverberated sound generation scheme shown in Fig. 5.31 is implemented on music sound from sound file ‘utopia.wav’[website4]. The original music sound and the reverberated music sound waveforms are shown in Fig. 5.34a, b, respectively. The reverberated music sound is found to be more pleasant to hear than the original.

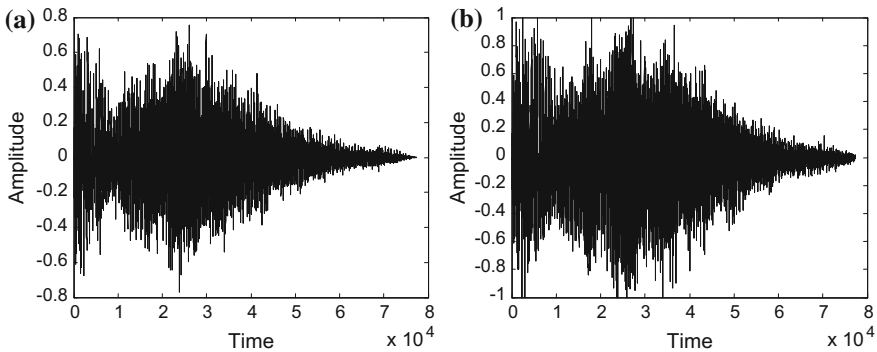


Fig. 5.34 a Original music sound and b reverberated

5.7.3 Audio Peaking Equalizers

With the availability of low-cost DSPs and recurrent usage of digital sounds, the need for audio equalizers has become crucial. By using a peaking equalizer filter section, a boost or cut is obtained in the vicinity of the center frequency. Peaking equalizer filter section is commonly known as parametric equalizer section, in which the gain is outlying from the boost or cut, so that a number of such sections can be arranged in series. Figure 5.35 shows a typical peaking equalizer comprised of cascaded IIR second-order filters.

The transfer function of an IIR second-order peaking filter is given by Robert [16]

$$H(z) = \frac{b_0z^2 + b_1z + b_2}{a_0z^2 + a_1z + a_2} \tag{5.82}$$

where

$$b_0 = 1 + \alpha\sqrt{K}, b_1 = -2\cos\omega_0, b_2 = 1 - \alpha\sqrt{K}$$

$$a_0 = 1 + \frac{\alpha}{\sqrt{K}}, a_1 = -2\cos\omega_0, a_2 = 1 - \frac{\alpha}{\sqrt{K}}$$

$$\alpha = [\sin \omega_0] \sinh\left(\frac{\ln 2}{2} B_w \frac{\omega_0}{\sin \omega_0}\right)$$

$$\omega_0 = \frac{2\pi f_0}{F_T}, B_w = \frac{2\pi b_w}{F_T}, K = 10^{(G/20)}$$

f_0 = Peak frequency in Hz

b_w = Bandwidth in Hz

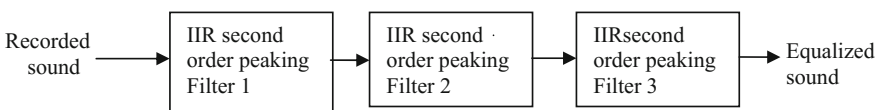


Fig. 5.35 Typical peaking equalizer comprised of cascaded IIR second-order filters

F_T = Sampling frequency

K = Gain at the peak frequency f_0

G = Peak gain in dB

B_w = Bandwidth in octaves given by $\omega_+ = \omega_- 2^{B_w}$, ω_+ and ω_- being the upper and lower edge frequencies, where the gain in dB is $G/2$.

As an example, consider the design of a peaking equalizer (Fig. 5.35) satisfying the following specifications:

Peaking Filter 1: $f_0 = 1600$ Hz, $b_w = 800$ Hz, $F_T = 44,100$ Hz, $G = 20$ dB

Peaking Filter 2: $f_0 = 2400$ Hz, $b_w = 800$ Hz, $F_T = 44,100$ Hz, $G = 20$ dB

Peaking Filter 3: $f_0 = 3200$ Hz, $b_w = 800$ Hz, $F_T = 44,100$ Hz, $G = 20$ dB

The magnitude and phase responses of each of the three peaking equalizers are shown in Fig. 5.36.

The music sound from the sound file ‘original.wav’ [17] is applied to the peaking equalizer of Fig. 5.35 with the magnitude response and phase response as shown in Fig. 5.36. The original sound signal and the equalized sound signal are shown in Fig. 5.37a, b, respectively.

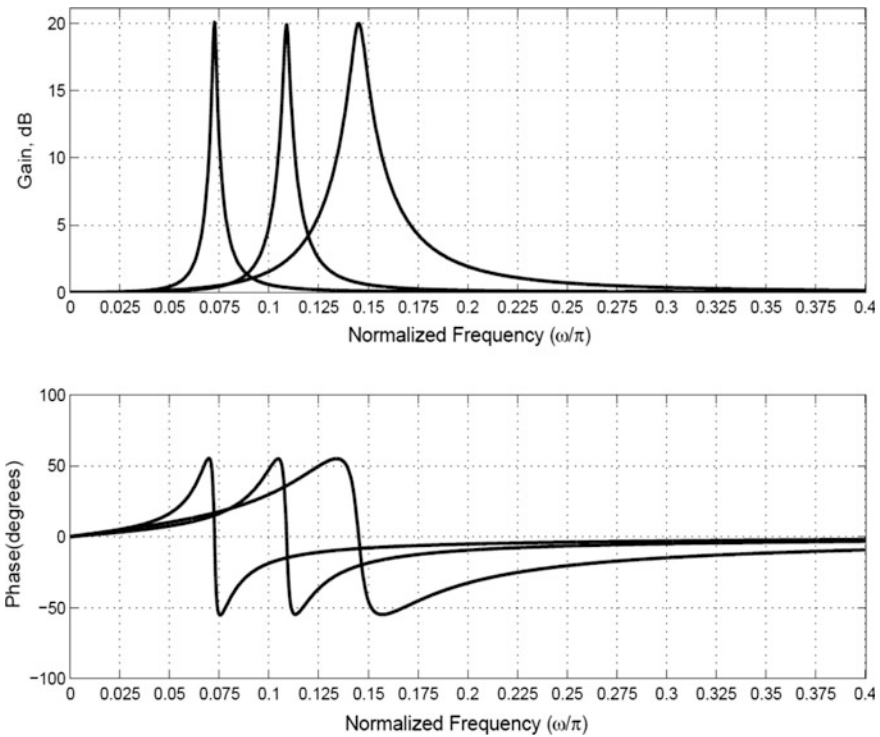


Fig. 5.36 Magnitude and phase responses of the three second-order IIR peaking equalizers

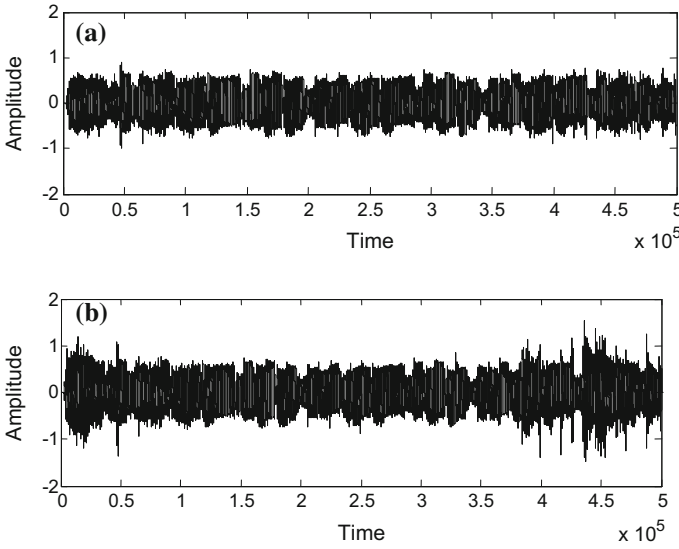


Fig. 5.37 **a** Original sound and **b** sound obtained after passing through a peaking equalizer comprised of three cascaded IIR second-order filters

5.7.4 Generation and Detection of DTMF Tones

Dual-Tone Multifrequency Tone Generator

The DTMF tone generator can be developed using two IIR digital filters in parallel.

The DTMF generator for key ‘5’ is depicted in Fig. 5.38.

Dual-Tone Multifrequency Tone Detection Using the Modified Goertzel Algorithm

Based on the specified frequencies of each DTMF tone and the modified Goertzel algorithm, the stepwise procedure for DTMF tone detection is as follows [18]:

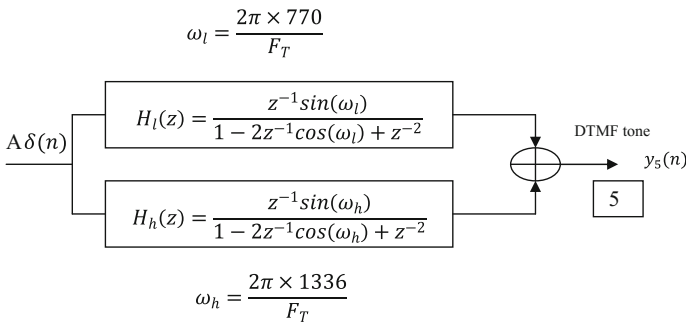


Fig. 5.38 Digital DTMF tone generator for the key ‘5’

- Step 1 For every digitized DTMF tone received, two nonzero frequency components are found from the following seven: 697, 770, 852, 941, 1209, 1336, and 1477 Hz.
- Step 2 Apply the modified Goertzel algorithm to compute seven spectral values, which correspond to the seven frequencies mentioned in Step 1. The single-sided amplitude spectrum is computed using the following expression:

$$A_k = \frac{2\sqrt{|X(k)|^2}}{N} \quad (5.83)$$

- Step 3 Determine the Key by using two nonzero spectral components corresponding to the key is pressed.
- Step 4 Determine the frequency bin number (frequency index) based on the sampling rate f_s and the data size of N using the following relation:

$$k = \frac{f}{F_T} \times N(\text{round off to integer}) \quad (5.84)$$

Since the telephone industry has preset F_T the sampling frequency to 8 kHz and the DTMFs to 697, 770, 852, 941, 1209, 1336, and 1477, the filter length must be large enough to find the desired k value that corresponds to the DTMF frequencies. Therefore, there is a trade-off to be considered between the computation burden and better resolution. For this application report, the filter length, N , was chosen as 105, which is the smallest value that can fulfill DTMF detection. Table 5.7 shows the calculated k values for $N = 105$.

Table 5.7 DTMFs and their frequency bins

DTMF f (Hz)	Frequency bin $k = \frac{f}{F_T} \times N$ (round off to an integer)
697	9
770	10
852	11
941	12
1209	16
1336	18
1477	19

Now, compute the frequency bin k for each DTMF frequency with $f_s = 8000$ Hz and $N = 105$ as tabulated in Table 5.7.

The DTMF detector block diagram is shown in Fig. 5.39.

Step 5. Add all seven spectral values and divide the sum by 4 to obtain the threshold value

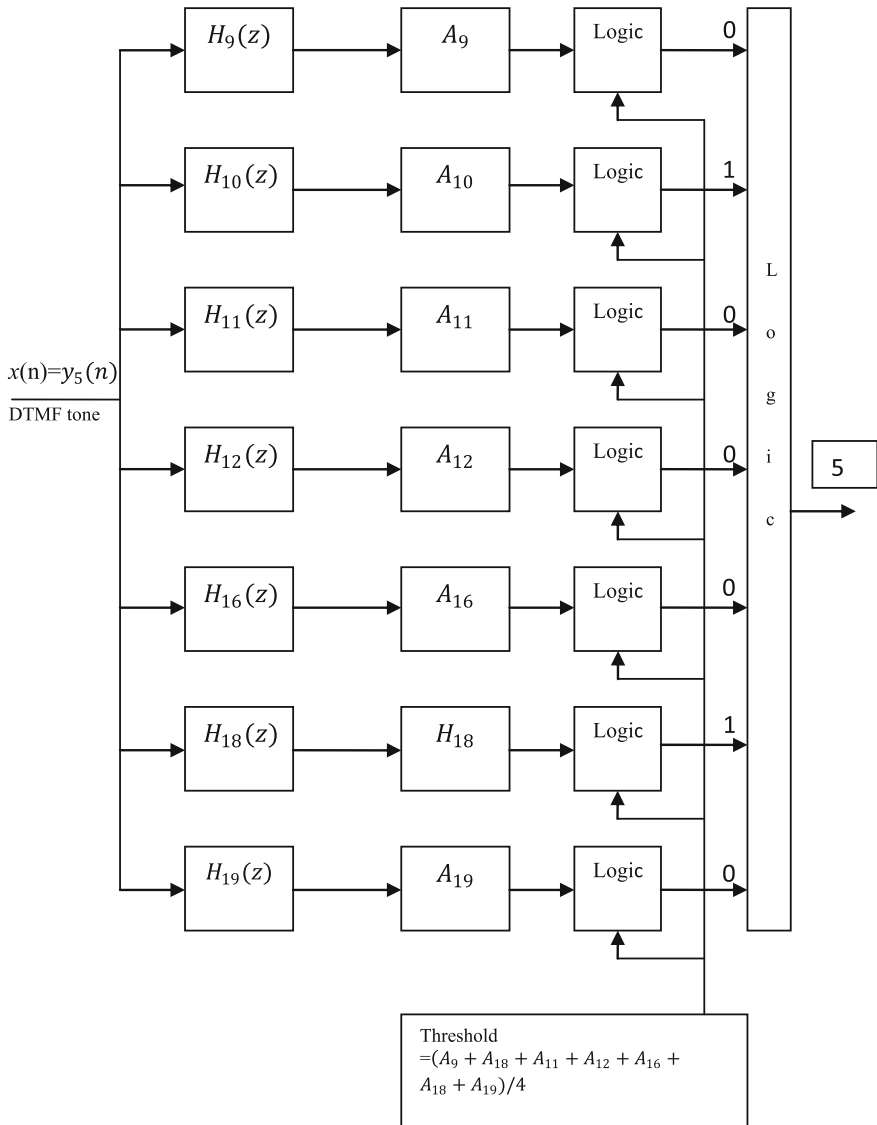
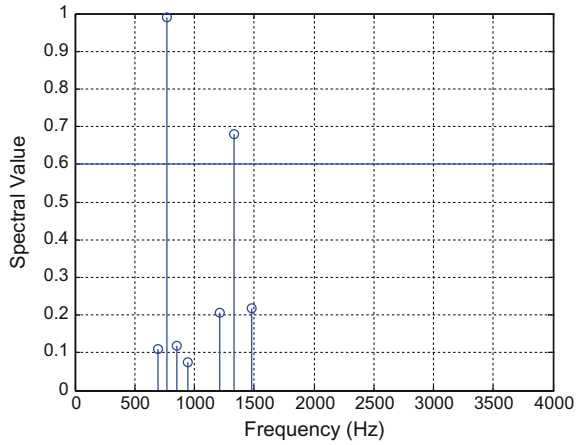


Fig. 5.39 DTMF tone detector using the Goertzel algorithm

Fig. 5.40 Spectral values and threshold for key 5



The logic operation outputs logic 1 for the spectrum value greater than the threshold value; otherwise, the logic operation outputs logic 0. The last-stage logic operation decodes the key information based on the 7-bit binary pattern.

The MATLAB simulation for decoding key 5 is shown in Program 5.12. The input is generated as shown in Fig. 5.38. After filtering, the calculated spectral values and the threshold value for decoding key 5 are displayed in Fig. 5.40, where only two spectral values corresponding to the frequencies of 770 and 1336 Hz are above the threshold and are encoded as logic 1. According to the key information in Fig. 5.39, the final logic operation decodes the key as 5.

Program 5.12 DTMF Detection Using Goertzel Algorithm

```

close all;clear all;
N=105;fs=8000; t=[0:1:N-1]/fs; % Sampling rate and time vector
x=zeros(1,length(t));x(1)=1; % Generate the impulse functionn
yDTMF=[];
%Generation of tones
f = [697 770 852 941 1209 1336 1477];
lf=[697;770;852;941];hf=[1209;1336;1477];
y1f1=filter([0 sin(2*pi*1f(1)/FT)],[1 -2*cos(2*pi*1f(1)/FT) 1],x);
y1f2=filter([0 sin(2*pi*1f(2)/FT)],[1 -2*cos(2*pi*1f(2)/FT) 1],x);
y1f3=filter([0 sin(2*pi*1f(3)/FT)],[1 -2*cos(2*pi*1f(3)/FT) 1],x);
y1f4=filter([0 sin(2*pi*1f(4)/FT)],[1 -2*cos(2*pi*1f(4)/FT) 1],x);
yh1f=filter([0 sin(2*pi*hf(1)/FT)],[1 -2*cos(2*pi*hf(1)/FT) 1],x);
yh2f=filter([0 sin(2*pi*hf(2)/FT)],[1 -2*cos(2*pi*hf(2)/FT) 1],x);
yh3f=filter([0 sin(2*pi*hf(3)/FT)],[1 -2*cos(2*pi*hf(3)/fs) 1],x);
key = input('enter key=') FT
if key==1 yDTMF=y1f1+yh1f; end if key==2 yDTMF=y1f1+yh2f; end
if key==3 yDTMF=y1f1+yh3f; end if key==4 yDTMF=y1f2+yh1f; end
if key==5 yDTMF=y1f2+yh2f; end if key==6 yDTMF=y1f2+yh3f; end
if key==7 yDTMF=y1f3+yh1f; end if key==8 yDTMF=y1f3+yh2f; end
if key==9 yDTMF=y1f3+yh3f; end
if key==10 % '*'
    yDTMF=y1f4+yh1f; end
if key==11 yDTMF=y1f4+yh2f; end
if key==12 % '#'
    yDTMF=y1f4+yh3f; end
yDTMF=[yDTMF 0]; % DTMF signal appended with a zero
% DTMF detector (use Goertzel algorithm)
a9=[1 -2*cos(2*pi*9/N) 1];a10=[1 -2*cos(2*pi*10/N) 1];
a11=[1 -2*cos(2*pi*11/N) 1];a12=[1 -2*cos(2*pi*12/N) 1];
a16=[1 -2*cos(2*pi*16/N) 1];a18=[1 -2*cos(2*pi*18/N) 1];
a19=[1 -2*cos(2*pi*19/N) 1];y9=filter(1,a9,yDTMF);
y10=filter(1,a10,yDTMF);y11=filter(1,a11,yDTMF);
y12=filter(1,a12,yDTMF);y16=filter(1,a16,yDTMF);
y18=filter(1,a18,yDTMF);y19=filter(1,a19,yDTMF);
% Determine the absolute magnitude of DFT coefficients
m(1)=sqrt(y9(105)^2+y9(104)^2-2*cos(2*pi*9/105)*y9(105)*y9(104));
m(2)=sqrt(y10(105)^2+y10(104)^2-2*cos(2*pi*10/105)*y10(105)*y10(104));
m(3)=sqrt(y11(105)^2+y11(104)^2-2*cos(2*pi*11/105)*y11(105)*y11(104));
m(4)=sqrt(y12(105)^2+y12(104)^2- 2*cos(2*pi*12/105)*y12(105)*y12(104));
m(5)=sqrt(y16(105)^2+y16(104)^2- 2*cos(2*pi*16/105)*y16(105)*y16(104));
m(6)=sqrt(y18(105)^2+y18(104)^2- 2*cos(2*pi*18/105)*y18(105)*y18(104));
m(7)=sqrt(y19(105)^2+y19(104)^2- 2*cos(2*pi*19/105)*y19(105)*y19(104));
m=2*m/105;th=sum(m)/4;% threshold
f1=[0 FT/2];th=[ th th];stem(f,m);grid;hold; plot(f1,th);
% xlabel('Frequency (Hz)'); ylabel(' (b) Spectral values');
m=round(m); % Round to the binary pattern
if m==[ 1 0 0 0 1 0 0] disp('Detected Key 1'); end
if m== [ 1 0 0 0 0 1 0] disp('Detected Key 2'); end
if m== [ 1 0 0 0 0 0 1] disp('Detected Key 3'); end
if m== [ 0 1 0 0 1 0 0] disp('Detected Key 4'); end
if m== [ 0 1 0 0 0 1 0] disp('Detected Key 5'); end
if m==[ 0 1 0 0 0 0 1] disp('Detected Key 6'); end
if m== [ 0 0 1 0 1 0 0] disp('Detected Key 7'); end
if m== [ 0 0 1 0 0 1 0] disp('Detected Key 8'); end
if m==[ 0 0 1 0 0 0 1] disp('Detected Key 9'); end
if m== [ 0 0 0 1 1 0 0] disp('Detected Key *'); end
if m== [ 0 0 0 1 0 1 0] disp('Detected Key 0'); end

```


5.8 Problems

1. For the following specifications, design a lowpass IIR digital Butterworth filter using the impulse-invariant method.

$$0.8 \leq |H(e^{j\omega})| \leq 1 \quad \text{for } 0 \leq \omega \leq 0.3\pi$$

$$|H(e^{j\omega})| \leq 0.4 \quad \text{for } 0.6\pi \leq \omega \leq \pi$$

2. Using the bilinear transformation, design a lowpass IIR digital Butterworth filter with -3 dB cutoff at 150 Hz and stopband attenuation of 20 dB or greater at 600 Hz. The sampling frequency is 6000 Hz.
3. Design a digital Butterworth highpass filter to meet the following specifications:

Passband edge frequency: 1000 Hz
 Stopband edge frequency: 400 Hz
 Passband ripple: 3 dB
 Stopband ripple: 10 dB

Assume a suitable sampling frequency.

4. Design a Butterworth IIR digital bandpass filter for the following specifications:

Lower passband edge frequency: 500 Hz
 Upper passband edge frequency: 600 Hz
 Lower stopband edge frequency: 100 Hz
 Upper stopband edge frequency: 1000 Hz
 Passband ripple: 2 dB
 Stopband ripple: 10 dB

Assume 4000 Hz as the sampling frequency. Use bilinear transformation.

5. Design a Chebyshev IIR digital lowpass filter for the following specifications:

Passband cutoff frequency: 400 Hz
 Stopband cutoff frequency: 600 Hz
 Passband ripple: 1 dB
 Stopband ripple: 10 dB

Assume a suitable sampling frequency. Use bilinear transformation.

6. Design a Chebyshev IIR digital highpass filter for the following specifications:

3 dB cutoff frequency: 2000 Hz
 Stopband cutoff frequency: 500 Hz
 Stopband ripple: 10 dB

Assume a suitable sampling frequency. Use bilinear transformation.

7. Using bilinear transformation, design a digital Chebyshev Type 1 Bandpass filter with the following specifications:

Lower passband edge frequency: 200 Hz
 Upper passband edge frequency: 400 Hz
 Lower stopband edge frequency: 100 Hz
 Upper stopband edge frequency: 500 Hz
 Passband ripple: 3 dB
 Stopband ripple: 15 dB

Assume a suitable sampling frequency.

8. Using bilinear transformation, design a digital bandstop Chebyshev Type 1 filter with the following specifications:

Lower passband edge frequency: 35 Hz
 Upper passband edge frequency: 215 Hz
 Lower stopband edge frequency: 100 Hz
 Upper stopband edge frequency: 150 Hz
 Passband ripple: 2 dB
 Stopband ripple: 20 dB

Assume a suitable sampling frequency.

9. Using bilinear transformation, design a digital bandstop elliptic filter with the following specifications:

Lower passband edge frequency: 800 Hz
 Upper passband edge frequency: 2000 Hz
 Lower stopband edge frequency: 1200 Hz
 Upper stopband edge frequency: 1300 Hz
 Passband ripple: 1 dB
 Stopband ripple: 40 dB

Assume a suitable sampling frequency.

10. A third-order lowpass IIR digital filter with passband edge frequency at 0.25π has a transfer function

$$H(z) = \frac{0.0662272z^3 + 0.1987z^2 + 0.1987z + 0.0662272}{z^3 - 0.9356142z^2 + 0.5671268z - 0.1015911}$$

Design a lowpass filter with passband edge frequency at 0.375π by transforming the above transfer function using lowpass-to-lowpass digital-to-digital transformation.

5.9 MATLAB Exercises

1. Write a MATLAB program using the M-file *impinvar* to design a Type 1 Chebyshev IIR digital lowpass filter using the impulse-invariant method for the specifications given in Example 5.9.

2. Write MATLAB code to design a Type 1 Chebyshev bandstop filter using bilinear transformation with the following specifications:

Lower passband edge: 0.3333π

Upper passband edge: 0.75π

Lower stopband edge: 0.45π

Upper stopband edge: 0.75π

Passband ripple: 1 dB

Stopband ripple: 40 dB

3. Write a MATLAB program to design a highpass Butterworth filter using digital-to-digital transformation satisfying the following specifications:

Passband edge frequency: 0.5π

Stopband edge frequency: 0.4π

Passband ripple: 2 dB

Stopband ripple: 20 dB

4. A third-order lowpass IIR digital filter with passband edge frequency at 0.25π has a transfer function

$$H(z) = \frac{0.0662272z^3 + 0.1987z^2 + 0.1987z + 0.0662272}{z^3 - 0.9356142z^2 + 0.5671268z - 0.1015911}$$

Write MATLAB code to design a highpass filter with passband edge frequency at 0.45π by transforming the above transfer function using lowpass-to-highpass digital-to-digital transformation. Show the magnitude responses of the lowpass and highpass filters on the same plot.

5. A first-order lowpass filter with passband edge frequency at 0.1667π has a transfer function as

$$H(z) = \frac{0.5z + 0.5}{z - 0.302}$$

Write MATLAB code to design a bandpass filter with lower passband edge frequency at 0.25π and upper passband edge frequency at 0.75π by transforming the above transfer function using lowpass-to-highpass digital transformation. Show the magnitude responses of the lowpass and bandpass filters.

6. Write a MATLAB program to suppress a sinusoidal interference of 1750 Hz from an audio signal using a second-order IIR digital notch filter. Consider the audio signal 'DT.wav' included in CD, and corrupt it by a sinusoidal signal of 1750 Hz. Implement the notch filter on it and comment on the results.
7. Write a MATLAB program to generate artificial reverberations using the scheme (Fig. 5.31) with six IIR filters and four allpass reverberators as shown in Fig. 5.32, and with structures for IIR filters and allpass reverberators as shown in Fig. 5.32. Implement it with suitable delays and multiplier constants on the music sound 'utopia.wav' included in the CD and comment on the result.

8. Write a MATLAB program for peaking equalizer consisting of three second-order IIR filters in cascade with different center frequencies, bandwidths, and DB gains for the filters. Implement it on the music sound ‘original.wav,’ and comment on the result.

References

1. G.C. Temes, S.K. Mitra (eds.), *Modern Filter Theory and Design* (Wiley, New York, 1973)
2. G.C. Temes, J.W. LaPatra, *Introduction to Circuit Synthesis and Design* (McGraw Hill, New York, 1977)
3. J. Vlach, *Computerized Approximation and Synthesis of Linear Networks* (Wiley, New York, 1969)
4. A. Antoniou, in *Digital Filters Analysis, Design, and Applications* (McGraw Hill Book Co., 2006)
5. R. Raut, M.N.S. Swamy, *Modern Analog Filter Analysis and Design: A Practical Approach* (Springer, VCH, 2010)
6. T.W. Parks, C.S. Burrus, *Digital Filter Design* (Wiley, New York, NY, 1987)
7. C.T. Chen, in *Digital Signal Processing, Spectral Computation and Filter Design* (Oxford University Press, New York, Oxford, 2001)
8. P.V.A. Mohan, V. Ramachandran, M.N.S. Swamy, *Switched Capacitor Filters: Theory, Analysis and Design* (Prentice Hall, London, 1995)
9. A.G. Constantinides, Proc. IEEE **117**, 1585–1590, 170
10. R.E. Crochier, P. Penfield Jr., On the efficient design of bandpass digital filter structures. IEEE Trans. ASSP **23**, 380–381 (1975)
11. R. Chassaing, in *DSP Applications Using C and the TMS320C6x DSK* (Wiley Interscience Publication, 2002)
12. E.C. Ifeachor, B.W. Jervis, in *Digital Signal Processing—A Practical Approach* (Prentice Hall, 2002)
13. L.B. Jackson, *Digital Filters and Signal Processing* (Kluwer Academic, Norwell, M.A., 1996)
14. P.A. Bernhardt, D.A. Antoniadis, A.V. Da Rosa, Lunar perturbations in columnar electron content and their interpretation in terms of dynamo electrostatic fields. J. Geophys. Res. (1976)
15. www.harmony-central.com
16. B.-J. Robert, in *The Equivalence of Various Methods of Computing Biquad Coefficients for Audio Parametric Equalizers*. Audio Engineering Society Conference, November 1994, paper # 3906
17. Audio Special Effects—Equalization and Artificial Reverberation, <http://www.nd.edu>
18. L. Tan, J. Jiang, in *Digital Signal Processing Fundamentals and Applications* (Elsevier, Academic Press 2nd, USA, edition, 2013)