CHAPTER

Finite Impulse Response Filter Design

7

CHAPTER OUTLINE

OBJECTIVES:

This chapter introduces principles of the finite impulse response (FIR) filter design and investigates design methods such as the Fourier transform method, window method, frequency sampling method, and optimal design method. Then the chapter illustrates how to apply the designed FIR filters to solve real-world problems such as noise reduction and digital crossover for audio applications.

7.1 FINITE IMPULSE RESPONSE FILTER FORMAT

In this chapter, we describe techniques for designing *finite impulse response* (FIR) filters. An FIR filter is completely specified by the following input–output relationship:

$$\begin{aligned} y(n) &= \sum_{i=0}^{K} b_i x(n-i) \\ &= b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) + \cdots + b_K x(n-K) \end{aligned} \tag{7.1}$$

Digital Signal Processing. http://dx.doi.org/10.1016/B978-0-12-415893-1.00007-X

where b_i represents FIR filter coefficients and $K+1$ denotes the FIR filter length. Applying the z-transform on both sides on Equation (7.1) leads to

$$Y(z) = b_0X(z) + b_1z^{-1}X(z) + \cdots + b_Kz^{-K}X(z) \tag{7.2}$$

Factoring out $X(z)$ on the right-hand side of Equation (7.2) and then dividing $X(z)$ on both sides, we have the transfer function, which depicts the FIR filter, as

$$H(z) = \frac{Y(z)}{X(z)} = b_0 + b_1z^{-1} + \cdots + b_Kz^{-K} \tag{7.3}$$

The following example serves to illustrate the notations used in Equations (7.1) and (7.3) numerically.

EXAMPLE 7.1

Given the FIR filter

$$y(n) = 0.1x(n) + 0.25x(n-1) + 0.2x(n-2)$$

determine the transfer function, filter length, nonzero coefficients, and impulse response.

Solution:

Applying the z-transform on both sides of the difference equation yields

$$Y(z) = 0.1X(z) + 0.25X(z)z^{-1} + 0.2X(z)z^{-2}$$

Then the transfer function is found to be

$$H(z) = \frac{Y(z)}{X(z)} = 0.1 + 0.25z^{-1} + 0.2z^{-2}$$

The filter length is $K+1 = 3$, and the identified coefficients are

$$b_0 = 0.1,\ b_1 = 0.25 \quad \text{and} \quad b_2 = 0.2$$

Taking the inverse z-transform of the transfer function, we have

$$h(n) = 0.1\delta(n) + 0.25\delta(n-1) + 0.2\delta(n-2)$$

This FIR filter impulse response has only three terms.

The foregoing example is to help us understand the FIR filter format. We can conclude the following:

1. The transfer function in Equation (7.3) has a constant term, all the other terms have negative powers of z, and all the poles are at the origin on the z-plane. Hence, the stability of the filter is guaranteed. Its impulse response has only a finite number of terms.
2. The FIR filter operations involve only multiplying the filter inputs by their corresponding coefficients and accumulating them; the implementation of this filter type in real time is straightforward.

From the FIR filter format, the design objective is to obtain b_i coefficients for the FIR filter such that the magnitude frequency response of the FIR filter $H(z)$ will approximate the desired magnitude frequency response, such as that of a lowpass, highpass, bandpass, or bandstop filter. The following sections will introduce design methods to calculate the FIR filter coefficients.

7.2 FOURIER TRANSFORM DESIGN

We begin with an ideal lowpass filter with a normalized cutoff frequency Ω_c, whose magnitude frequency response in terms of the normalized digital frequency Ω is plotted in Figure 7.1 and is characterized by

$$H(e^{j\Omega}) = \begin{cases} 1, & 0 \le |\Omega| \le \Omega_c \\ 0, & \Omega_c \le |\Omega| \le \pi \end{cases} \tag{7.4}$$

Since the frequency response is periodic with a period of $\Omega = 2\pi$ radians, as we discussed in Chapter 6, we can extend the frequency response of the ideal filter $H(e^{j\Omega})$, as shown in Figure 7.2.

The periodic frequency response can be approximated using a complex Fourier series expansion (see Appendix B) in terms of the normalized digital frequency Ω, that is,

$$H(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} c_n e^{-j\omega_0 n\Omega} \tag{7.5}$$

and the Fourier coefficients are given by

$$c_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(e^{j\Omega})e^{j\omega_0 n\Omega}d\Omega \quad \text{for} \quad -\infty < n < \infty \tag{7.6}$$

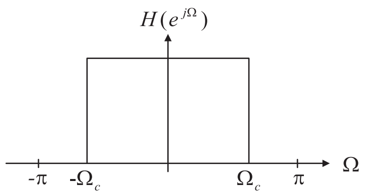

FIGURE 7.1

Frequency response of an ideal lowpass filter.

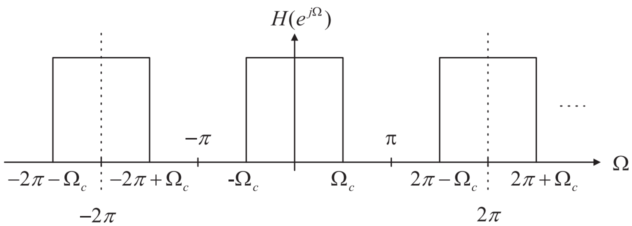

FIGURE 7.2

Periodicity of the ideal lowpass frequency response.

Notice that we obtain Equations (7.5) and (7.6) simply by treating the Fourier series expansion in the time domain with the time variable t replaced by the normalized digital frequency variable Ω. The fundamental frequency is easily found to be

$$\omega_0 = 2\pi/(\textit{period of waveform}) = 2\pi/2\pi = 1 \tag{7.7}$$

Substituting $\omega_0 = 1$ into Equation (7.6) and introducing $h(n) = c_n$, called the desired impulse response of the ideal filter, we obtain the Fourier transform design as

$$h(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(e^{j\Omega})e^{j\Omega n}d\Omega \quad \text{for} \quad -\infty < n < \infty \tag{7.8}$$

Now, let us look at the possible z-transfer function. If we substitute $e^{j\Omega} = z$ and $\omega_0 = 1$ back to Equation (7.5), we yield a z-transfer function in the following format:

$$\begin{aligned} H(z) &= \sum_{n=-\infty}^{\infty} h(n)z^{-n} \\ &\cdots + h(-2)z^2 + h(-1)z^1 + h(0) + h(1)z^{-1} + h(2)z^{-2} + \cdots \end{aligned} \tag{7.9}$$

This is a noncausal FIR filter. We will deal with this later in this section. Using the Fourier transform design shown in Equation (7.8), the desired impulse response approximation of the ideal lowpass filter is solved as

$$\text{For } n = 0 \quad h(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(e^{j\Omega})e^{j\Omega\times 0}d\Omega$$

$$= \frac{1}{2\pi}\int_{-\Omega_c}^{\Omega_c} 1 d\Omega = \frac{\Omega_c}{\pi}$$

$$\text{For } n \neq 0 \quad h(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(e^{j\Omega})e^{j\Omega n}d\Omega = \frac{1}{2\pi}\int_{-\Omega_c}^{\Omega_c} e^{j\Omega n}d\Omega$$

$$= \left.\frac{e^{jn\Omega}}{2\pi jn}\right|_{-\Omega_c}^{\Omega_c} = \frac{1}{\pi n}\,\frac{e^{jn\Omega_c} - e^{-jn\Omega_c}}{2j} = \frac{\sin(\Omega_c n)}{\pi n} \tag{7.10}$$

The desired impulse response $h(n)$ is plotted versus the sample number n in Figure 7.3.

Theoretically, $h(n)$ in Equation (7.10) exists for $-\infty < n < \infty$ and is symmetrical about $n = 0$; that is, $h(n) = h(-n)$. The amplitude of the impulse response sequence $h(n)$ becomes smaller when n increases in both directions. The FIR filter design must first be completed by truncating the infinite-length sequence $h(n)$ to achieve the $2M+1$ dominant coefficients using the coefficient symmetry, that is,

$$H(z) = h(M)z^M + \cdots + h(1)z^1 + h(0) + h(1)z^{-1} + \cdots + h(M)z^{-M}$$

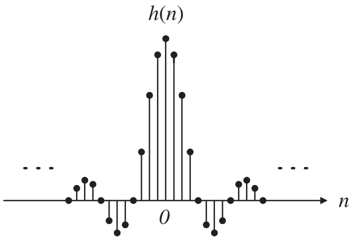

FIGURE 7.3

Impulse response of an ideal digital lowpass filter.

The obtained filter is a noncausal z-transfer function of the FIR filter, since the filter transfer function contains terms with the positive powers of z, which in turn means that the filter output depends on the future filter inputs. To remedy the noncausal z-transfer function, we delay the truncated impulse response $h(n)$ by M samples to yield the following causal FIR filter:

$$H(z) = b_0 + b_1 z^{-1} + \cdots + b_{2M}(2M)z^{-2M} \tag{7.11}$$

Table 7.1 Summary of Ideal Impulse Responses for Standard FIR Filters

Filter Type	Ideal Impulse Response $h(n)$ (noncausal FIR coefficients)
Lowpass:	$h(n) = \begin{cases} \dfrac{\Omega_c}{\pi} & \text{for} \quad n = 0 \\ \dfrac{\sin(\Omega_c n)}{n\pi} & \text{for} \quad n \neq 0 \end{cases} \quad -M \le n \le M$
Highpass:	$h(n) = \begin{cases} \dfrac{\pi - \Omega_c}{\pi} & \text{for} \quad n = 0 \\ -\dfrac{\sin(\Omega_c n)}{n\pi} & \text{for} \quad n \neq 0 \end{cases} \quad -M \le n \le M$
Bandpass:	$h(n) = \begin{cases} \dfrac{\Omega_H - \Omega_L}{\pi} & \text{for} \quad n = 0 \\ \dfrac{\sin(\Omega_H n)}{n\pi} - \dfrac{\sin(\Omega_L n)}{n\pi} & \text{for} \quad n \neq 0 \end{cases} \quad -M \le n \le M$
Bandstop:	$h(n) = \begin{cases} \dfrac{\pi - \Omega_H + \Omega_L}{\pi} & \text{for} \quad n = 0 \\ -\dfrac{\sin(\Omega_H n)}{n\pi} + \dfrac{\sin(\Omega_L n)}{n\pi} & \text{for} \quad n \neq 0 \end{cases} \quad -M \le n \le M$

Causal FIR filter coefficients: shifting $h(n)$ to the right by M samples.
Transfer function:
$H(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots b_{2M} z^{-2M}$
where $b_n = h(n - M), n = 0, 1, \cdots, 2M.$

where the delay operation is given by

$$b_n = h(n - M) \quad \text{for} \quad n = 0, 1, \cdots, 2M \tag{7.12}$$

Similarly, we can obtain the design equations for other types of FIR filters, such as highpass, bandpass, and bandstop, using their ideal frequency responses and Equation (7.8). The derivations are omitted here. Table 7.1 gives a summary of all the formulas for FIR filter coefficient calculations.

The following example illustrates the coefficient calculation for the lowpass FIR filter.

EXAMPLE 7.2

a. Calculate the filter coefficients for a 3-tap FIR lowpass filter with a cutoff frequency of 800 Hz and a sampling rate of 8,000 Hz using the Fourier transform method.
b. Determine the transfer function and difference equation of the designed FIR system.
c. Compute and plot the magnitude frequency response for $\Omega = 0, \pi/4, \pi/2, 3\pi/4,$ and π radians.

Solution:

a. Calculating the normalized cutoff frequency leads to

$$\Omega_c = 2\pi f_c T_s = 2\pi \times 800/8,000 = 0.2\pi \text{ radians}$$

Since $2M + 1 = 3$ in this case, using the equation in Table 7.1 results in

$$h(0) = \frac{\Omega_c}{\pi} \quad \text{for} \quad n = 0$$

$$h(n) = \frac{\sin(\Omega_c n)}{n\pi} = \frac{\sin(0.2\pi n)}{n\pi} \quad \text{for} \quad n \neq 1$$

The computed filter coefficients via the previous expression are listed as:

$$h(0) = \frac{0.2\pi}{\pi} = 0.2$$

$$h(1) = \frac{\sin[0.2\pi \times 1]}{1 \times \pi} = 0.1871$$

Using the symmetry leads to

$$h(-1) = h(1) = 0.1871$$

Thus delaying $h(n)$ by $M = 1$ sample using Equation (7.12) gives

$$b_0 = h(0 - 1) = h(-1) = 0.1871$$

$$b_1 = h(1 - 1) = h(0) = 0.2$$

$$b_2 = h(2 - 1) = h(1) = 0.1871$$

b. The transfer function is achieved as

$$H(z) = 0.1871 + 0.2z^{-1} + 0.1871z^{-2}$$

Using the technique described in Chapter 6, we have

$$\frac{Y(z)}{X(z)} = H(z) = 0.1871 + 0.2z^{-1} + 0.1871z^{-2}$$

Table 7.2 Frequency Response Calculation in Example 7.2

Ω radians	$f = \Omega f_s/(2\pi)$ **Hz**	**0.2 + 0.3742 cos Ω**	$\lvert H(e^{j\Omega})\rvert$	$\lvert H(e^{j\Omega})\rvert_{dB}$	$\angle H(e^{j\Omega})$ **degree**
0	0	0.5742	0.5742	−4.82	0
$\pi/4$	1000	0.4646	0.4646	−6.66	−45
$\pi/2$	2000	0.2	0.2	−14.0	−90
$3\pi/4$	3000	−0.0646	0.0646	−23.8	45
π	4000	−0.1742	0.1742	−15.2	0

Multiplying *X(z)* leads to

$$Y(z) = 0.1871X(z) + 0.2z^{-1}X(z) + 0.1871z^{-2}X(z)$$

Applying the inverse z-transform on both sides, the difference equation is yielded as

$$y(n) = 0.1871x(n) + 0.2x(n-1) + 0.1871x(n-2)$$

c. The magnitude frequency response and phase response can be obtained using the technique introduced in Chapter 6. Substituting $z = e^{j\Omega}$ into $H(z)$, it follows that

$$H(e^{j\Omega}) = 0.1871 + 0.2e^{-j\Omega} + 0.1871e^{-j2\Omega}$$

Factoring term $e^{-j\Omega}$ and using the Euler formula $e^{jx} + e^{-jx} = 2\cos(x)$, we achieve

$$\begin{aligned} H(e^{j\Omega}) &= e^{-j\Omega}(0.1871e^{j\Omega} + 0.2 + 0.1871e^{-j\Omega}) \\ &= e^{-j\Omega}(0.2 + 0.3742\cos(\Omega)) \end{aligned}$$

Then the magnitude frequency response and phase response are found to be

$$\left|H(e^{j\Omega})\right| = |0.2 + 0.3472\cos\Omega|$$

and

$$\angle H(e^{j\Omega}) = \begin{cases} -\Omega & \text{if} \quad 0.2 + 0.3472\cos\Omega > 0 \\ -\Omega + \pi & \text{if} \quad 0.2 + 0.3472\cos\Omega < 0 \end{cases}$$

Details of the magnitude calculations for several typical normalized frequencies are listed in Table 7.2.

Due to the symmetry of the coefficients, the obtained FIR filter has a linear phase response as shown in Figure 7.4. The sawtooth shape is produced by the contribution of the negative sign of the real magnitude term $0.2 + 0.3742\cos\Omega$ in the 3-tap filter frequency response, that is,

$$H(e^{j\Omega}) = e^{-j\Omega}(0.2 + 0.3742\cos\Omega)$$

In general, the FIR filter with symmetric coefficients has a linear phase response (linear function of Ω) as follows:

$$\angle H(e^{j\Omega}) = -M\Omega + \text{possible phase of } 180^\circ \tag{7.13}$$

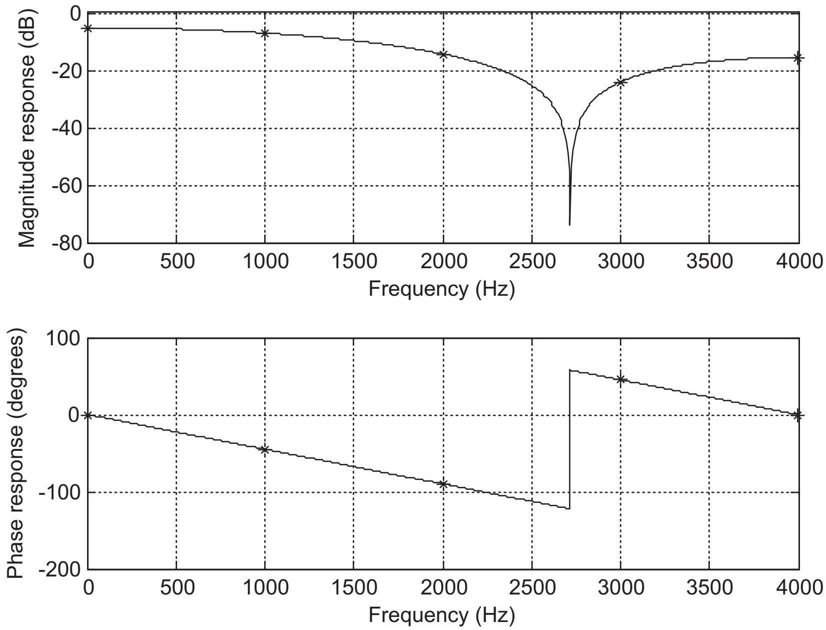

FIGURE 7.4

Magnitude and phase frequency responses in Example 7.2.

Next, we see that the 3-tap FIR filter does not give an acceptable magnitude frequency response.

To explore this response further, Figure 7.5 displays the magnitude and phase responses of 3-tap ($M = 1$) and 17-tap ($M = 8$) FIR lowpass filters with a normalized cutoff frequency $\Omega_c = 0.2\pi$ radians. The calculated coefficients for the 17-tap FIR lowpass filter are listed in Table 7.3.

We can make the following observations at this point:

1. The oscillations (ripples) exhibited in the passband (main lobe) and stopband (side lobes) of the magnitude frequency response constitute the *Gibbs effect*. The Gibbs oscillatory behavior originates from the abrupt truncation of the infinite impulse response in Equation (7.11). To remedy this problem, window functions will be used and will be discussed in the next section.
2. Using a larger number of the filter coefficients will produce the sharp roll-off characteristic of the transition band but may cause increased time delay and increase computational complexity for implementing the designed FIR filter.
3. The phase response is linear in the passband. This is consistent with Equation (7.13), which means that all frequency components of the filter input within the passband are subjected to the same time delay at the filter output. This is a requirement for applications in audio and speech filtering, where phase distortion needs to be avoided. Note that we impose a linear phase requirement, that is, the FIR coefficients are symmetric about the middle coefficient, and the FIR filter order is an odd number. If the design methods cannot produce the symmetric coefficients or generate anti-symmetric coefficients (Proakis and Manolakis, 1996), the resultant FIR filter does not have the

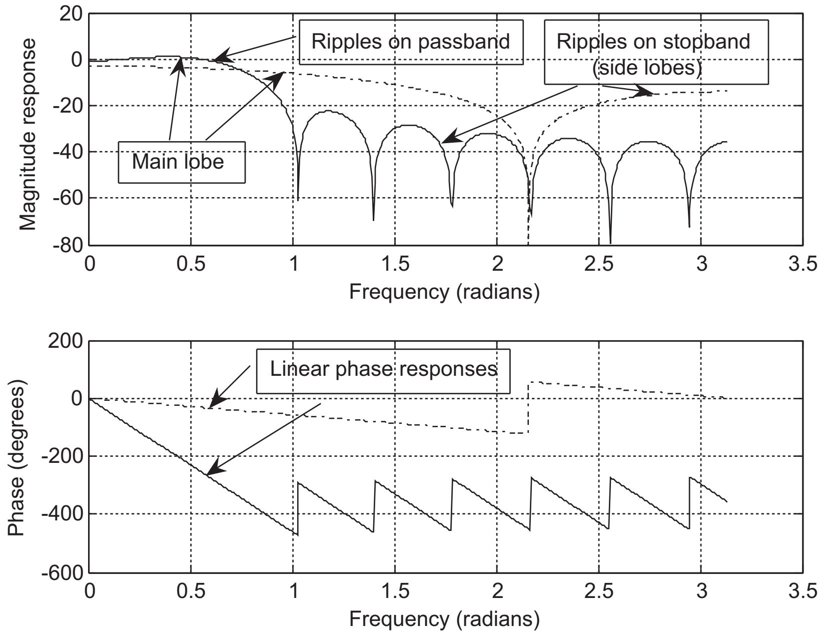

FIGURE 7.5

Magnitude and phase frequency responses of the lowpass FIR filters with 3 coefficients (dash-dotted line) and 17 coefficients (solid line).

linear phase property. (Linear phase even-order FIR filters and FIR filters using the anti-symmetry of coefficients are discussed in Proakis and Manolakis [1996].)

To further probe the linear phase property, we consider a sinusoidal sequence $x(n) = A \sin(n\Omega)$ as the FIR filter input, with the output expected to be

$$y(n) = A|H|\sin(n\Omega + \phi)$$

where $\phi = -M\Omega$. Substituting $\phi = -M\Omega$ into $y(n)$ leads to

$$y(n) = A|H|\sin[\Omega(n - M)]$$

Table 7.3 17-Tap FIR Lowpass Filter Coefficients in Example 7.2 ($M = 8$)

$b_0 = b_{16} = -0.0378$	$b_1 = b_{15} = -0.0432$	
$b_2 = b_{14} = -0.0312$	$b_3 = b_{13} = 0.0000$	
$b_4 = b_{12} = 0.0468$	$b_5 = b_{11} = 0.1009$	
$b_6 = b_{10} = 0.1514$	$b_7 = b_9 = 0.1871$	$b_8 = 0.2000$

This clearly indicates that within the passband, all frequency components passing through the FIR filter will have the same constant delay at the output, which equals M samples. Hence, phase distortion is avoided.

Figure 7.6 verifies the linear phase property using an FIR filter with 17 taps. Two sinusoids of the normalized digital frequencies 0.05π and 0.15π radians, respectively, are used as inputs. These two input signals are within the passband, so their magnitudes are not changed. As shown in Figure 7.6, beginning at the ninth sample the output matches the input, which is delayed by eight samples for each case.

What would happen if the filter phase were nonlinear? This can be illustrated using the following combined sinusoids as the filter input:

$$x(n) = x_1(n) + x_2(n) = \sin(0.05\pi n)u(n) - \frac{1}{3}\sin(0.15\pi n)u(n)$$

The original $x(n)$ is the top plot shown in Figure 7.7. If the linear phase response of a filter is considered, such as $\phi = -M\Omega_0$, where $M = 8$ in our illustration, we have the filtered output as

$$y_1(n) = \sin[0.05\pi(n-8)] - \frac{1}{3}\sin[0.15\pi(n-8)]$$

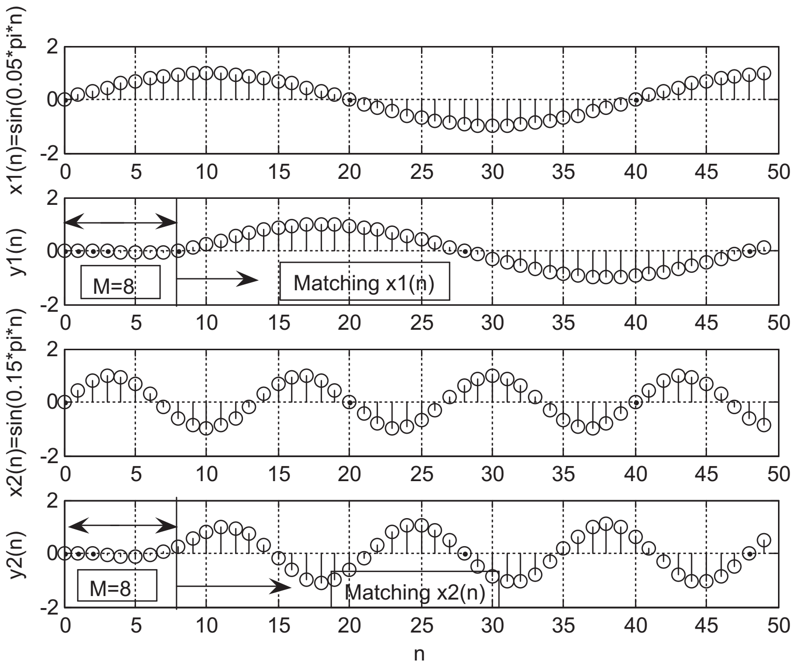

FIGURE 7.6

Illustration of FIR filter linear phase property (constant delay of eight samples).

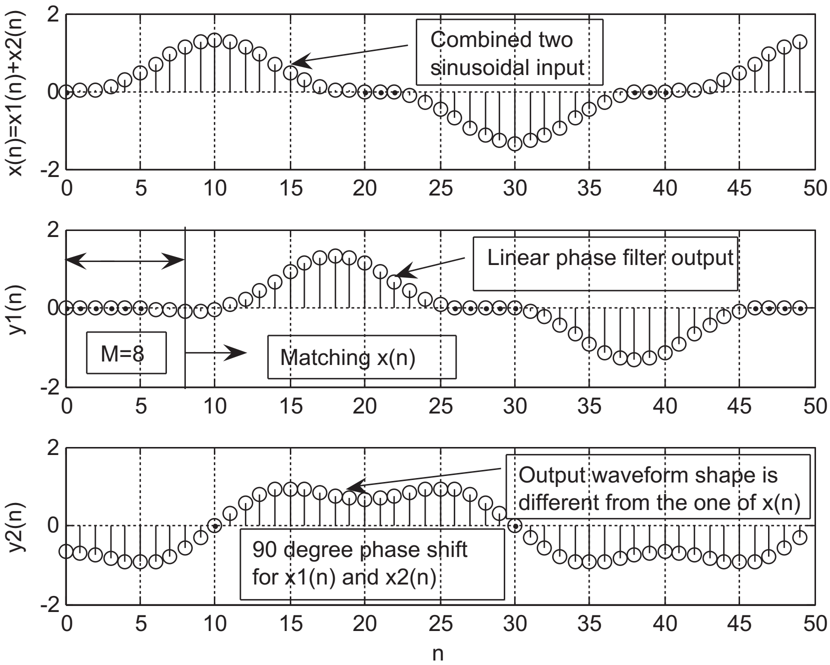

FIGURE 7.7

Comparison of linear and nonlinear phase responses.

The linear phase effect is shown in the middle plot of Figure 7.7. We see that $y_1(n)$ is the eight-sample delayed version of $x(n)$. However, considering a unit gain filter with a phase delay of 90 degrees for all the frequency components, we obtain the filtered output as

$$y_2(n) = \sin(0.05\pi n - \pi/2) - \frac{1}{3}\sin(0.15\pi n - \pi/2)$$

where the first term has a phase shift of 10 samples (see $\sin[0.05\pi(n-10)]$), while the second term has a phase shift of 10/3 samples $\left(\text{see } \frac{1}{3}\sin\left[0.15\pi\left(n-\frac{10}{3}\right)\right]\right)$. Certainly, we do not have the linear phase feature. The signal $y_2(n)$ plotted in Figure 7.7 shows that the waveform shape is different from that of the original signal $x(n)$, and hence has significant phase distortion. This phase distortion is audible for audio applications and can be avoided by using an FIR filter, which has the linear phase feature.

We now have finished discussing the coefficient calculation for the FIR lowpass filter, which has a good linear phase property. To explain the calculation of filter coefficients for the other types of filters and examine the Gibbs effect, we look at another simple example.

EXAMPLE 7.3

a. Calculate the filter coefficients for a 5-tap FIR bandpass filter with a lower cutoff frequency of 2,000 Hz and an upper cutoff frequency of 2,400 Hz and a sampling rate of 8,000 Hz.

b. Determine the transfer function and plot the frequency responses with MATLAB.

Solution:

a. Calculating the normalized cutoff frequencies leads to

$$\Omega_L = 2\pi f_L/f_s = 2\pi \times 2,000/8,000 = 0.5\pi \text{ radians}$$

$$\Omega_H = 2\pi f_H/f_s = 2\pi \times 2,400/8,000 = 0.6\pi \text{ radians}$$

Since $2M+1 = 5$ in this case, using the equation in Table 7.1 yields

$$h(n) = \begin{cases} \dfrac{\Omega_H - \Omega_L}{\pi} & n = 0 \\ \dfrac{\sin(\Omega_H n)}{n\pi} - \dfrac{\sin(\Omega_L n)}{n\pi} & n \neq 0 \quad -2 \le n \le 2 \end{cases} \tag{7.14}$$

Calculations for noncausal FIR coefficients are listed as

$$h(0) = \frac{\Omega_H - \Omega_L}{\pi} = \frac{0.6\pi - 0.5\pi}{\pi} = 0.1$$

The other computed filter coefficients via Equation (7.14) are

$$h(1) = \frac{\sin[0.6\pi \times 1]}{1 \times \pi} - \frac{\sin[0.5\pi \times 1]}{1 \times \pi} = -0.01558$$

$$h(2) = \frac{\sin[0.6\pi \times 2]}{2 \times \pi} - \frac{\sin[0.5\pi \times 2]}{2 \times \pi} = -0.09355$$

Using symmetry leads to

$$h(-1) = h(1) = -0.01558$$

$$h(-2) = h(2) = -0.09355$$

Thus, delaying $h(n)$ by $M = 2$ samples gives

$$b_0 = b_4 = -0.09355$$

$$b_1 = b_3 = -0.01558, \quad \text{and} \quad b_2 = 0.1$$

b. The transfer function is achieved as

$$H(z) = -0.09355 - 0.01558z^{-1} + 0.1z^{-2} - 0.01558z^{-3} - 0.09355z^{-4}$$

To complete Example 7.3, the magnitude frequency response plotted in terms of $|H(e^{j\Omega})|_{dB} = 20\log_{10}|H(e^{j\Omega})|$ using MATLAB Program 7.1 is displayed in Figure 7.8.

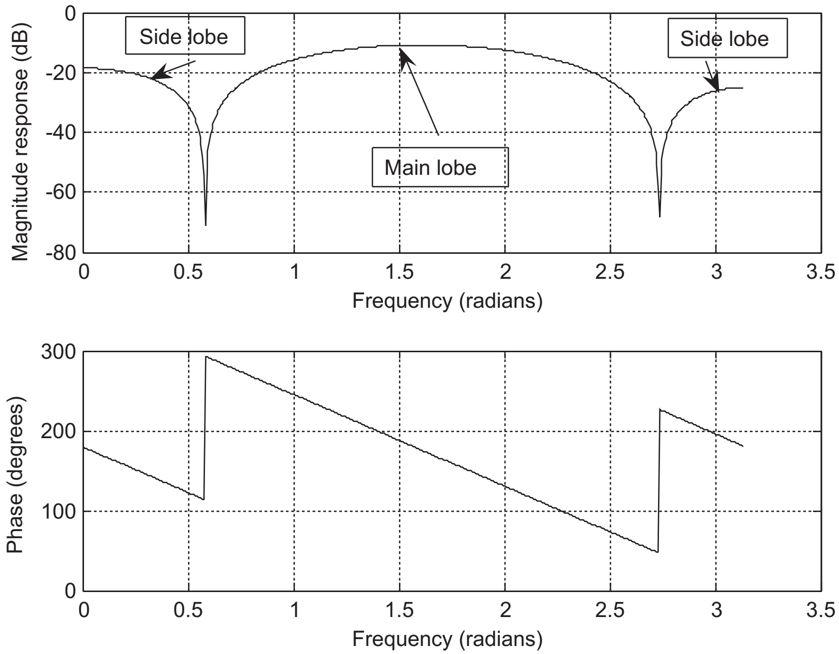

FIGURE 7.8

Frequency responses for Example 7.3.

Program 7.1. MATLAB program for Example 7.3.

```
% Example 7.3
% MATLAB program to plot frequency response
%
 [hz,w]=freqz([-0.09355 -0.01558 0.1 -0.01558 -0.09355], [1], 512);
 phi=180*unwrap(angle(hz))/pi;
 subplot(2,1,1), plot(w,20*log10(abs(hz))),grid;
 xlabel('Frequency (radians)');
 ylabel('Magnitude Response (dB)')
 subplot(2,1,2), plot(w, phi); grid;
 xlabel('Frequency (radians)');
 ylabel('Phase (degrees)');
```

To summarize Example 7.3, the magnitude frequency response demonstrates the Gibbs oscillatory behavior existing in the passband and stopband. The peak of the main lobe in the passband is dropped from 0 dB to approximately -10 dB, while for the stopband, the lower side lobe in the magnitude response plot swings approximately between –18 dB and –70 dB, and the upper side lobe swings between -25 dB and -68 dB. As we have pointed out, this is due to the abrupt truncation of the infinite impulse sequence $h(n)$. The oscillations can be reduced by increasing the number of coefficients and using a window function, which will be studied next.

7.3 WINDOW METHOD

In this section, the *window method* (Fourier transform design with window functions) is developed to remedy the undesirable Gibbs oscillations in the passband and stopband of the designed FIR filter. Recall that the Gibbs oscillations originate from the abrupt truncation of the infinite-length coefficient sequence. Then it is natural to seek a window function, which is symmetrical and can gradually weight the designed FIR coefficients down to zeros at both ends for the range $-M \le n \le M$. Applying the window sequence to the filter coefficients gives

$$h_w(n) = h(n) \cdot w(n)$$

where $w(n)$ designates the window function. Common window functions used in the FIR filter design are as follows:

1. Rectangular window:

$$w_{rec}(n) = 1, \; -M \le n \le M \tag{7.15}$$

2. Triangular (Bartlett) window:

$$w_{tri}(n) = 1 - \frac{|n|}{M}, \; -M \le n \le M \tag{7.16}$$

3. Hanning window:

$$w_{han}(n) = 0.5 + 0.5 \cos\left(\frac{n\pi}{M}\right), -M \le n \le M \tag{7.17}$$

4. Hamming window:

$$w_{ham}(n) = 0.54 + 0.46 \cos\left(\frac{n\pi}{M}\right), -M \le n \le M \tag{7.18}$$

5. Blackman window:

$$w_{black}(n) = 0.42 + 0.5 \cos\left(\frac{n\pi}{M}\right) + 0.08 \cos\left(\frac{2n\pi}{M}\right), \; -M \le n \le M \tag{7.19}$$

In addition, there is another popular window function, called the Kaiser window (detailed information can be found in Oppenheim, Shaffer, and Buck [1999]). As we expected, the rectangular window function has a constant value of 1 within the window, and hence only does truncation. For comparison, shapes of the other window functions from Equations (7.16) to (7.19) are plotted in Figure 7.9 for the case of $2M + 1 = 81$.

We apply the Hamming window function in Example 7.4.

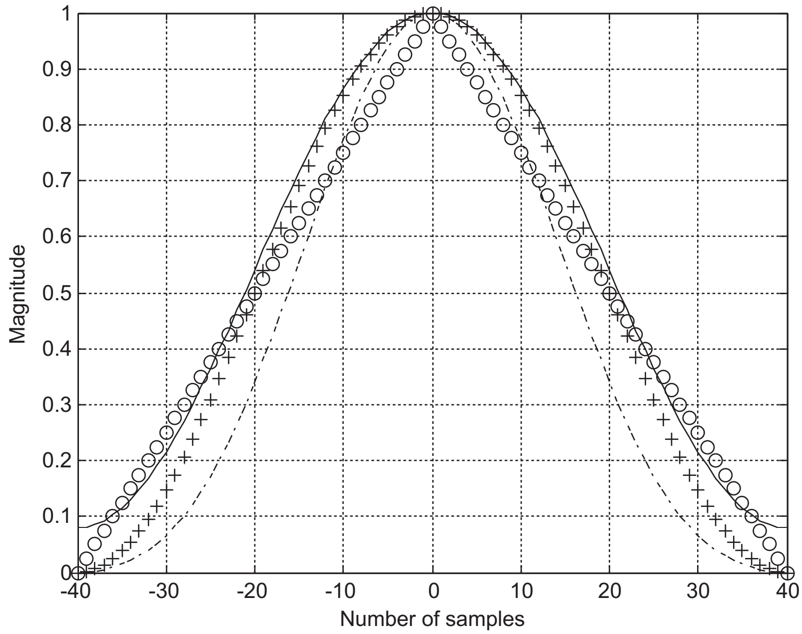

FIGURE 7.9

Shapes of window functions for the case of $2M + 1 = 81$. "o" line, triangular window; "+" line, Hanning window; solid line, Hamming window; dashed line, Blackman window.

EXAMPLE 7.4

Given the calculated filter coefficients

$$h(0) = 0.25,\ h(-1) = h(1) = 0.22508,\ h(-2) = h(2) = 0.15915,\ h(-3) = h(3) = 0.07503$$

a. apply the hamming window function to obtain windowed coefficients $h_w(n)$;
b. plot the impulse response $h(n)$ and windowed impulse response $h_w(n)$.

Solution:
a. Since $M = 3$, applying Equation (7.18) leads to the window sequence

$$w_{ham}(-3) = 0.54 + 0.46\cos\left(\frac{-3 \times \pi}{3}\right) = 0.08$$

$$w_{ham}(-2) = 0.54 + 0.46\cos\left(\frac{-2 \times \pi}{3}\right) = 0.31$$

$$w_{ham}(-1) = 0.54 + 0.46\cos\left(\frac{-1 \times \pi}{3}\right) = 0.77$$

$$w_{ham}(0) = 0.54 + 0.46\cos\left(\frac{0 \times \pi}{3}\right) = 1$$

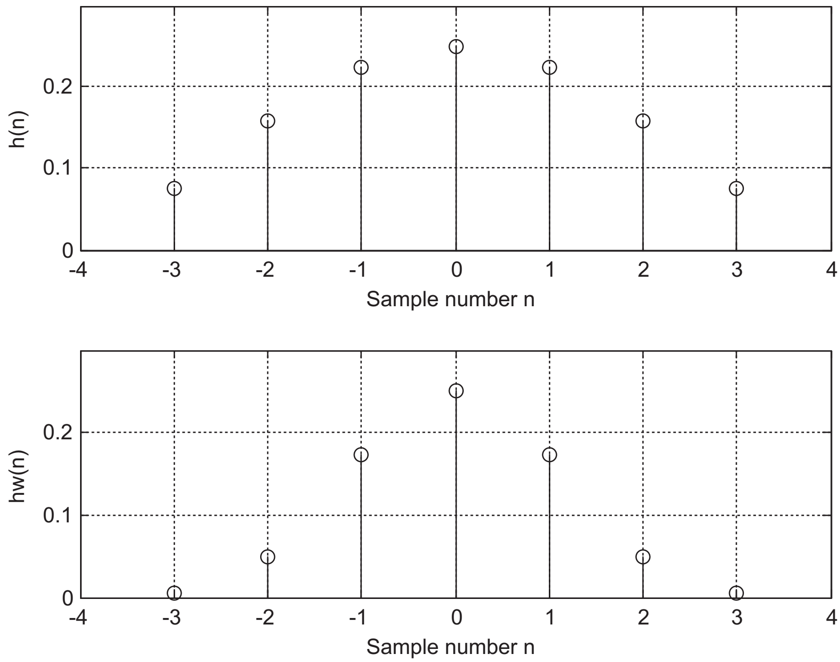

FIGURE 7.10

Plots of FIR noncausal coefficients and windowed FIR coefficients in Example 7.4.

$$w_{ham}(1) = 0.54 + 0.46 \cos\left(\frac{1 \times \pi}{3}\right) = 0.77$$

$$w_{ham}(2) = 0.54 + 0.46 \cos\left(\frac{2 \times \pi}{3}\right) = 0.31$$

$$w_{ham}(3) = 0.54 + 0.46 \cos\left(\frac{3 \times \pi}{3}\right) = 0.08$$

Applying the Hamming window function and its symmetric property to the filter coefficients, we get

$$h_w(0) = h(0) \cdot w_{ham}(0) = 0.25 \times 1 = 0.25$$

$$h_w(1) = h(1) \cdot w_{ham}(1) = 0.22508 \times 0.77 = 0.17331 = h_w(-1)$$

$$h_w(2) = h(2) \cdot w_{ham}(2) = 0.15915 \times 0.31 = 0.04934 = h_w(-2)$$

$$h_w(3) = h(3) \cdot w_{ham}(3) = 0.07503 \times 0.08 = 0.00600 = h_w(-3)$$

b. Noncausal impulse responses $h(n)$ and $h_w(n)$ are plotted in Figure 7.10.

We observe that the Hamming window does its job and weights the FIR filter coefficients to zero gradually at both ends. Hence, we can expect a reduced Gibbs effect in the magnitude frequency response.

Now lowpass FIR filter design via the window method can be achieved. The design procedure includes three steps. The first step is to obtain the truncated impulse response $h(n)$, where $-M \le n \le M$; then we multiply the obtained sequence $h(n)$ by the selected window data sequence to yield the windowed noncausal FIR filter coefficients $h_w(n)$; the final step is to delay the windowed noncausal sequence $h_w(n)$ by M samples to obtain the causal FIR filter coefficients, $b_n = h_w(n-M)$. The design procedure of the FIR filter via windowing is summarized as follows:

1. Obtain the FIR filter coefficients $h(n)$ via the Fourier transform method (Table 7.1).
2. Multiply the generated FIR filter coefficients by the selected window sequence

$$h_w(n) = h(n)w(n), \; n = -M, \cdots 0, 1, \cdots, M \tag{7.20}$$

   where $w(n)$ is chosen to be one of the window functions listed in Equations (7.15) to (7.19).
3. Delay the windowed impulse sequence $h_w(n)$ by M samples to get the windowed FIR filter coefficients

$$b_n = h_w(n-M), \quad \text{for} \quad n = 0, 1, \cdots, 2M \tag{7.21}$$

Let us study the following design examples.

EXAMPLE 7.5

a. Design a 3-tap FIR lowpass filter with a cutoff frequency of 800 Hz and a sampling rate of 8,000 Hz using the Hamming window function.
b. Determine the transfer function and difference equation of the designed FIR system.
c. Compute and plot the magnitude frequency response for $\Omega = 0, \pi/4, \pi/2, 3\pi/4,$ and π radians.

Solution:

a. The normalized cutoff frequency is calculated as

$$\Omega_c = 2\pi f_c T_s = 2\pi \times 800/8{,}000 = 0.2\pi \text{ radians}$$

Since $2M+1 = 3$ in this case, FIR coefficients obtained by using the equation in Table 7.1 are listed as

$$h(0) = 0.2 \quad \text{and} \quad h(-1) = h(1) = 0.1871$$

(see Example 7.2). Applying the Hamming window function defined in Equation (7.18), we have

$$w_{ham}(0) = 0.54 + 0.46 \cos\left(\frac{0\pi}{1}\right) = 1$$

$$w_{ham}(1) = 0.54 + 0.46 \cos\left(\frac{1 \times \pi}{1}\right) = 0.08$$

Using the symmetry of the window function gives

$$w_{ham}(-1) = w_{ham}(1) = 0.08$$

The windowed impulse response is calculated as

$$h_w(0) = h(0)w_{ham}(0) = 0.2 \times 1 = 0.2$$

$$h_w(1) = h(1)w_{ham}(1) = 0.1871 \times 0.08 = 0.01497$$

$$h_w(-1) = h(-1)w_{ham}(-1) = 0.1871 \times 0.08 = 0.01497$$

Thus delaying $h_w(n)$ by $M = 1$ sample gives

$$b_0 = b_2 = 0.01496 \quad \text{and} \quad b_1 = 0.2$$

b. The transfer function is

$$H(z) = 0.01497 + 0.2z^{-1} + 0.01497z^{-2}$$

Using the technique described in Chapter 6, we have

$$\frac{Y(z)}{X(z)} = H(z) = 0.01497 + 0.2z^{-1} + 0.01497z^{-2}$$

Multiplying X(z) leads to

$$Y(z) = 0.01497X(z) + 0.2z^{-1}X(z) + 0.01497z^{-2}X(z)$$

Applying the inverse z-transform on both sides, the difference equation is obtained as

$$y(n) = 0.01497x(n) + 0.2x(n-1) + 0.01497x(n-2)$$

c. The magnitude frequency response and phase response can be obtained using the technique introduced in Chapter 6. Substituting $z = e^{j\Omega}$ into $H(z)$, it follows that

$$\begin{aligned} H(e^{j\Omega}) &= 0.01497 + 0.2e^{-j\Omega} + 0.01497e^{-j2\Omega} \\ &= e^{-j\Omega}(0.01497e^{j\Omega} + 0.2 + 0.01497e^{-j\Omega}) \end{aligned}$$

Using Euler's formula leads to

$$H(e^{j\Omega}) = e^{-j\Omega}(0.2 + 0.02994 \cos \Omega)$$

Then the magnitude frequency response and phase response are found to be

$$\left|H(e^{j\Omega})\right| = |0.2 + 0.2994 \cos \Omega|$$

Table 7.4 Frequency Response Calculation in Example 7.5

Ω radians	$f = \Omega f_s/(2\pi)$ Hz	$0.2 + 0.02994 \cos \Omega$	$\left\|H(e^{j\Omega})\right\|$	$\left\|H(e^{j\Omega})\right\|_{dB}$ dB	$\angle H(e^{j\Omega})$ degrees
0	0	0.2299	0.2299	−12.77	0
$\pi/4$	1,000	0.1564	0.2212	−13.11	−45
$\pi/2$	2,000	0.2000	0.2000	−13.98	−90
$3\pi/4$	3,000	0.1788	0.1788	−14.95	−135
π	4,000	0.1701	0.1701	−15.39	−180

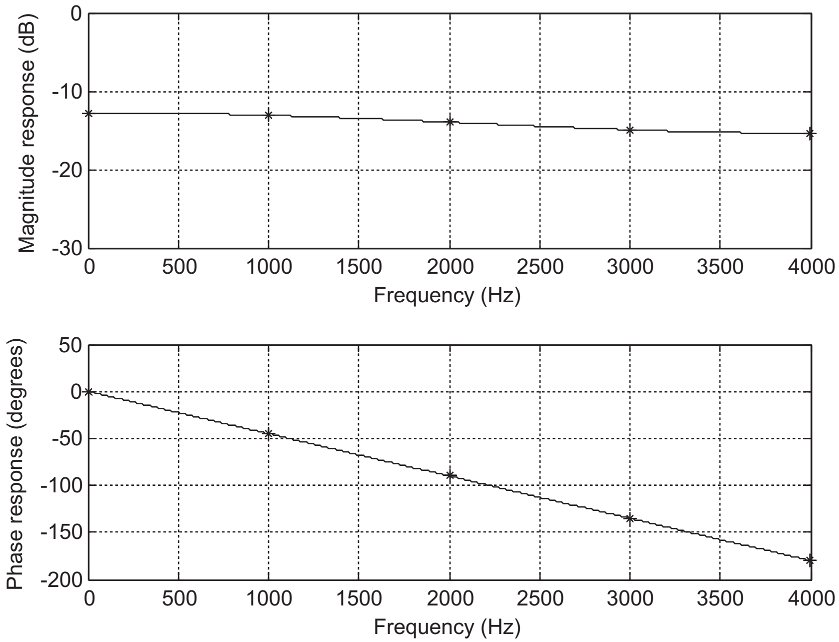

FIGURE 7.11

The frequency responses in Example 7.5.

and

$$\angle H(e^{j\Omega}) = \begin{cases} -\Omega & \text{if} \quad 0.2 + 0.02994 \cos\Omega > 0 \\ -\Omega + \pi & \text{if} \quad 0.2 + 0.02994 \cos\Omega < 0 \end{cases}$$

The calculation details of the magnitude response for several normalized values are listed in Table 7.4. Figure 7.11 shows the plots of the frequency responses.

EXAMPLE 7.6

a. Design a 5-tap FIR band reject (bandstop) filter with a lower cutoff frequency of 2,000 Hz, an upper cutoff frequency of 2,400 Hz, and a sampling rate of 8,000 Hz using the Hamming window method.

b. Determine the transfer function.

Solution:

a. Calculating the normalized cutoff frequencies leads to

$$\Omega_L = 2\pi f_L T = 2\pi \times 2{,}000/8{,}000 = 0.5\pi \text{ radians}$$

$$\Omega_H = 2\pi f_H T = 2\pi \times 2,400/8,000 = 0.6\pi \text{ radians}$$

Since $2M+1 = 5$ in this case, using the equation in Table 7.1 yields

$$h(n) = \begin{cases} \dfrac{\pi - \Omega_H + \Omega_L}{\pi} & n = 0 \\ -\dfrac{\sin(\Omega_H n)}{n\pi} + \dfrac{\sin(\Omega_L n)}{n\pi} & n \neq 0 \quad -2 \le n \le 2 \end{cases}$$

When $n = 0$, we have

$$h(0) = \frac{\pi - \Omega_H + \Omega_L}{\pi} = \frac{\pi - 0.6\pi + 0.5\pi}{\pi} = 0.9$$

The other computed filter coefficients for the previous expression are listed below:

$$h(1) = \frac{\sin[0.5\pi \times 1]}{1 \times \pi} - \frac{\sin[0.6\pi \times 1]}{1 \times \pi} = 0.01558$$

$$h(2) = \frac{\sin[0.5\pi \times 2]}{2 \times \pi} - \frac{\sin[0.6\pi \times 2]}{2 \times \pi} = 0.09355$$

Using symmetry leads to

$$h(-1) = h(1) = 0.01558$$

$$h(-2) = h(2) = 0.09355$$

Applying the Hamming window function in Equation (7.18), we have

$$w_{ham}(0) = 0.54 + 0.46 \cos\left(\frac{0 \times \pi}{2}\right) = 1.0$$

$$w_{ham}(1) = 0.54 + 0.46 \cos\left(\frac{1 \times \pi}{2}\right) = 0.54$$

$$w_{ham}(2) = 0.54 + 0.46 \cos\left(\frac{2 \times \pi}{2}\right) = 0.08$$

Using the symmetry of the window function gives

$$w_{ham}(-1) = w_{ham}(1) = 0.54$$

$$w_{ham}(-2) = w_{ham}(2) = 0.08$$

The windowed impulse response is calculated as

$$h_w(0) = h(0)w_{ham}(0) = 0.9 \times 1 = 0.9$$

$$h_w(1) = h(1)w_{ham}(1) = 0.01558 \times 0.54 = 0.00841$$

$$h_w(2) = h(2)w_{ham}(2) = 0.09355 \times 0.08 = 0.00748$$

$$h_w(-1) = h(-1)w_{ham}(-1) = 0.00841$$

$$h_w(-2) = h(-2)w_{ham}(-2) = 0.00748$$

Thus, delaying $h_w(n)$ by $M = 2$ samples gives

$$b_0 = b_4 = 0.00748, \; b_1 = b_3 = 0.00841, \quad \text{and} \quad b_2 = 0.9$$

b. The transfer function is

$$H(z) = 0.00748 + 0.00841z^{-1} + 0.9z^{-2} + 0.00841z^{-3} + 0.00748z^{-4}$$

The following design examples are demonstrated using MATLAB programs. The MATLAB function **firwd(N, Ftype, WnL, WnH, Wtype)** is listed in the "MATLAB Programs" section at the end of this chapter. Table 7.5 lists comments to show how the function is used.

Table 7.5 Illustration of the MATLAB Function for FIR Filter Design Using Window Methods

```
function B=firwd(N,Ftype,WnL,WnH,Wtype)
% B = firwd(N,Ftype,WnL,WnH,Wtype)
% FIR filter design using the window function method.
% Input parameters:
% N: the number of the FIR filter taps.
% Note: It must be odd number.
% Ftype: the filter type
%     1. Lowpass filter
%     2. Highpass filter
%     3. Bandpass filter
%     4. Band reject (Bandstop) filter
% WnL: lower cutoff frequency in radians. Set WnL=0 for the highpass filter.
% WnH: upper cutoff frequency in radians. Set WnH=0 for the lowpass filter.
% Wtype: window function type
%     1. Rectangular window
%     2. Triangular window
%     3. Hanning window
%     4. Hamming window
%     5. Blackman window
```

EXAMPLE 7.7

a. Design a lowpass FIR filter with 25 taps using the MATLAB program listed in the "MATLAB Programs" section at the end of this chapter. The cutoff frequency of the filter is 2,000 Hz, assuming a sampling frequency of 8,000 Hz. The rectangular window and Hamming window functions are used for each design.

b. Plot the frequency responses along with those obtained using the rectangular window and Hamming window for comparison.

c. List the FIR filter coefficients for each window design method.

Solution:

a. With a given sampling rate of 8,000 Hz, the normalized cutoff frequency can be found as

$$\Omega_c = \frac{2,000 \times 2\pi}{8,000} = 0.5\pi \text{ radians}$$

Now we are ready to design FIR filters via the MATLAB program. The function **firwd(N, Ftype, WnL, WnH, Wtype)** listed in the "MATLAB Programs" section at the end of this chapter, has five input parameters, which are described as follows:

- "N" is the number of specified filter coefficients (the number of filter taps).
- "Ftype" denotes the filter type, that is, input "1" for the lowpass filter design, input "2" for the highpass filter design, input "3" for the bandpass filter design, and input "4" for the band reject filter design.

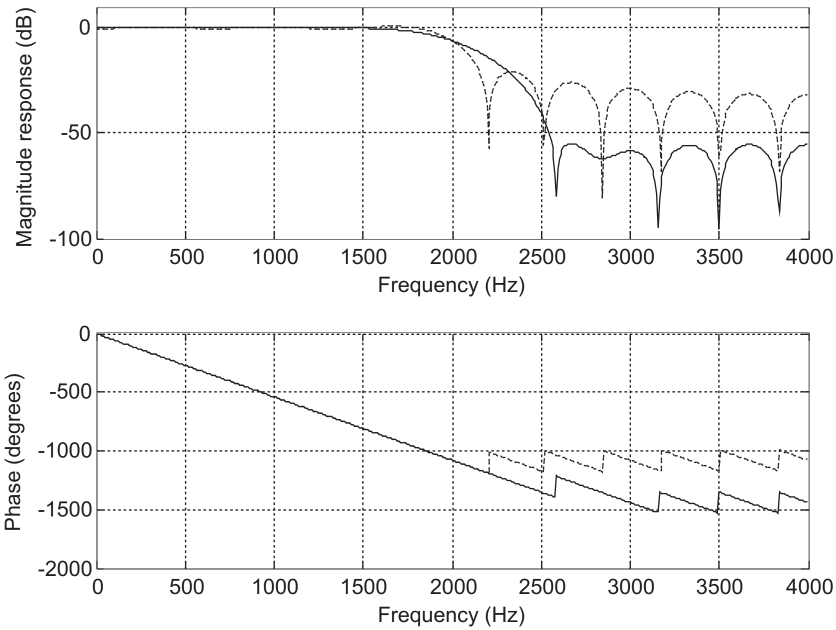

FIGURE 7.12

Frequency responses using the rectangular and Hamming windows.

- "WnL" and "WnH" are the lower and upper cutoff frequency inputs, respectively. Note that WnH = 0 when specifying WnL for the lowpass filter design, while WnL = 0 when specifying WnH for the highpass filter design.
- "Wtype" specifies the window data sequence to be used in the design, that is, input "1" for the rectangular window, input "2" for the triangular window, input "3" for the Hanning window, input "4" for the Hamming window, and input "5" for the Blackman window.

b. The following program (Program 7.2) is used to generate FIR filter coefficients using the rectangular window. Its frequency responses will be plotted together with the results of the FIR filter design obtained using the Hamming window, as shown in Program 7.3.

Program 7.2. MATLAB program for Example 7.7.

```
% Example 7.7
% MATLAB program to generate FIR coefficients
% using the rectangular window.
%
N=25; Ftype=1; WnL=0.5*pi; WnH=0; Wtype=1;
B=firwd(N,Ftype,WnL,WnH,Wtype);
```

Program 7.3. MATLAB program for Example 7.7.

```
%Figure 7.12
% MATLAB program to create Figure 7.12
%
N=25; Ftype=1; WnL=0.5*pi; WnH=0; Wtype=1;fs=8000;
%design using the rectangular window;
Brec=firwd(N,Ftype,WnL,WnH,Wtype);
N=25; Ftype=1; WnL=0.5*pi; WnH=0; Wtype=4;
%design using the Hamming window;
Bham=firwd(N,Ftype,WnL,WnH,Wtype);
[hrec,f]=freqz(Brec,1,512,fs);
[hham,f]=freqz(Bham,1,512,fs);
prec=180*unwrap(angle(hrec))/pi;
pham=180*unwrap(angle(hham))/pi;
subplot(2,1,1);
```

Table 7.6 FIR Filter Coefficients in Example 7.7 (rectangular and Hamming windows)

B: FIR Filter Coefficients (Rectangular Window)	Bham: FIR Filter Coefficients (Hamming Window)
$b_0 = b_{24} = 0.000000$	$b_0 = b_{24} = 0.000000$
$b_1 = b_{23} = -0.028937$	$b_1 = b_{23} = -0.002769$
$b_2 = b_{22} = 0.000000$	$b_2 = b_{22} = 0.000000$
$b_3 = b_{21} = 0.035368$	$b_3 = b_{21} = 0.007595$
$b_4 = b_{20} = 0.000000$	$b_4 = b_{20} = 0.000000$
$b_5 = b_{19} = -0.045473$	$b_5 = b_{19} = -0.019142$
$b_6 = b_{18} = 0.000000$	$b_6 = b_{18} = 0.000000$
$b_7 = b_{17} = 0.063662$	$b_7 = b_{17} = 0.041957$
$b_8 = b_{16} = 0.000000$	$b_8 = b_{16} = 0.000000$
$b_9 = b_{15} = -0.106103$	$b_9 = b_{15} = -0.091808$
$b_{10} = b_{14} = 0.000000$	$b_{10} = b_{14} = 0.000000$
$b_{11} = b_{13} = 0.318310$	$b_{11} = b_{13} = 0.313321$
$b_{12} = 0.500000$	$b_{12} = 0.500000$

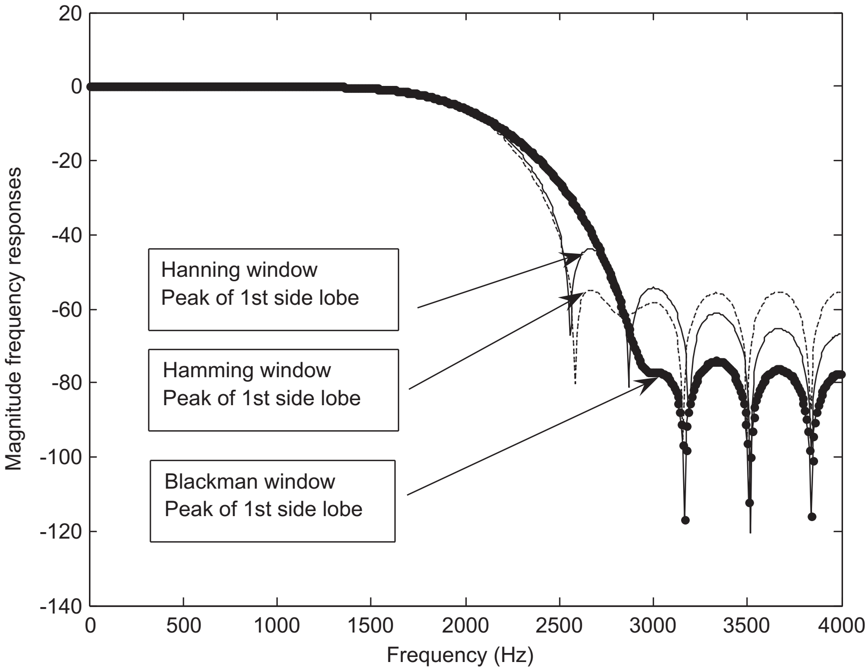

FIGURE 7.13

Comparisons of magnitude frequency responses for the Hanning, Hamming, and Blackman windows.

```
plot(f,20*log10(abs(hrec)),'-.',f,20*log10(abs(hham)));grid
axis([0 4000 -100 10]);
xlabel('Frequency (Hz)'); ylabel('Magnitude Response (dB)');
subplot(2,1,2);
plot(f,prec,'-.',f,pham);grid
xlabel('Frequency (Hz)'); ylabel('Phase (degrees)');
```

For comparison, the frequency responses achieved from the rectangular window and the Hamming window are plotted in Figure 7.12, where the dash-dotted line indicates the frequency response via the rectangular window, and the solid line indicates the frequency response via the Hamming window.

c. The FIR filter coefficients for both methods are listed in Table 7.6.

For comparison with other window functions, Figure 7.13 shows the magnitude frequency responses using the Hanning, Hamming, and Blackman windows, with 25 taps and a cutoff frequency of 2,000 Hz. The Blackman window offers the lowest side lobe, but with an increased width of the main lobe. The Hamming window and Hanning have a similar narrow width of the main lobe, but the Hamming window accommodates a lower side lobe than the Hanning window. Next, we will study how to choose a window in practice.

Applying the window to remedy the Gibbs effect will change the characteristics of the magnitude frequency response of the FIR filter, as the width of the main lobe becomes wider and more attenuation of the side lobes occurs.

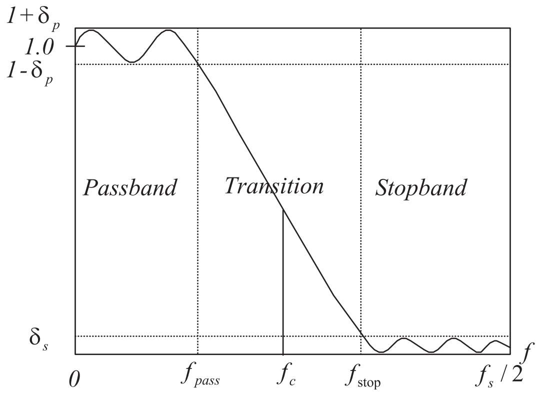

FIGURE 7.14

Lowpass filter frequency domain specifications.

Next, we illustrate the design for customer specifications in practice. Given the required stopband attenuation and passband ripple specifications shown in Figure 7.14, where the lowpass filter specifications are given for illustrative purposes, the appropriate window can be selected based on the performance of the window functions listed in Table 7.7. For example, the Hamming window offers a passband ripple of 0.0194 dB and stopband attenuation of 53 dB. With the selected Hamming window and the normalized transition band defined in Table 7.7,

$$\Delta f = \left|f_{stop} - f_{pass}\right| \big/ f_s \tag{7.22}$$

TABLE 7.7 FIR Filter Length Estimation Using Window Functions (normalized transition width $\Delta f = |f_{stop} - f_{pass}|/f_s$)

Window Type	Window Function $w(n), -M \le n \le M$	Window Length N	Passband Ripple (dB)	Stopband Attenuation (dB)
Rectangular	1	$N = 0.9/\Delta f$	0.7416	21
Hanning	$0.5 + 0.5\cos\left(\frac{\pi n}{M}\right)$	$N = 3.1/\Delta f$	0.0546	44
Hamming	$0.54 + 0.46\cos\left(\frac{\pi n}{M}\right)$	$N = 3.3/\Delta f$	0.0194	53
Blackman	$0.42 + 0.5\cos\left(\frac{n\pi}{M}\right) + 0.08\cos\left(\frac{2n\pi}{M}\right)$	$N = 5.5/\Delta f$	0.0017	74

the filter length using the Hamming window can be determined by

$$N = \frac{3.3}{\Delta f} \tag{7.23}$$

Note that the passband ripple is defined as

$$\delta_p dB = 20 \cdot \log_{10}(1 + \delta_p) \tag{7.24}$$

while the stopband attenuation is defined as

$$\delta_s dB = -20 \log_{10}(\delta_s) \tag{7.25}$$

The cutoff frequency used for the design will be chosen at the middle of the transition band, as illustrated for the lowpass filter case shown in Figure 7.14.

As a rule of thumb, the cutoff frequency used for design is determined by

$$f_c = (f_{pass} + f_{stop})/2 \tag{7.26}$$

Note that Equation (7.23) and formulas for other window lengths in Table 7.7 are empirically derived based on the normalized spectral transition width of each window function. The spectrum of each window function appears to be shaped like the lowpass filter magnitude frequency response with ripples in the passband and side lobes in the stopband. The passband frequency edge of the spectrum is the frequency where the magnitude just begins to drop below the passband ripple and where the stop frequency edge is at the peak of the first side lobe in the spectrum. With the passband ripple and stopband attenuation specified for a particular window, the normalized transition width of the window is in inverse proportion to the window length N multiplied by a constant. For example, the normalized spectral transition Δf for the Hamming window is $3.3/N$. Hence, matching the FIR filter transition width with the transition width of the window spectrum gives the filter length estimation listed in Table 7.7.

The following examples illustrate the determination of each filter length and cutoff frequency/frequencies for the design of lowpass, highpass, bandpass, and bandstop filters. Application of each designed filter to the processing of speech data is included, along with an illustration of filtering effects in both the time domain and frequency domain.

EXAMPLE 7.8

A lowpass FIR filter has the following specifications:

Passband 0–1,850 Hz
Stopband 2,150–4,000 Hz
Stopband attenuation 20 dB
Passband ripple 1 dB
Sampling rate 8,000 Hz

Determine the FIR filter length and the cutoff frequency to be used in the design equation.

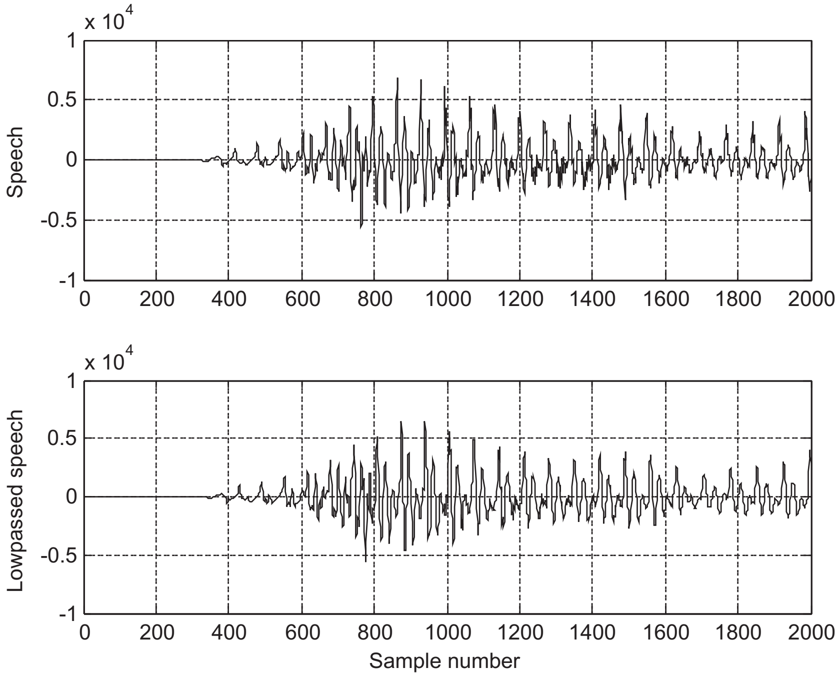

FIGURE 7.15A

Original speech and processed speech using the lowpass filter.

Solution:

The normalized transition band as defined in Equation (7.22) and Table 7.7 is given by

$$\Delta f = |2,150 - 1,850|/8,000 = 0.0375$$

Again, based on Table 7.7, selecting the rectangular window will result in a passband ripple of 0.74 dB and stopband attenuation of 21 dB. Thus, this window selection would satisfy the design requirement for a passband ripple of 1 dB and stopband attenuation of 20 dB. Next, we determine the length of the filter as

$$N = 0.9/\Delta f = 0.9/0.0375 = 24$$

We choose the odd number $N = 25$. The cutoff frequency is determined by $(1,850 + 2,150)/2 = 2,000$ Hz. Such a filter has been designed in Example 7.7, its filter coefficients are listed in Table 7.6, and its frequency responses can be found in Figure 7.12 (dashed lines).

Now we look at the time domain and frequency domain results from filtering a speech signal by using the lowpass filter we have just designed. Figure 7.15A shows the original speech and lowpass filtered speech. The spectral comparison is given in Figure 7.15B, where, as we can see, the frequency components beyond 2 kHz are filtered. The lowpass filtered speech would sound muffled.

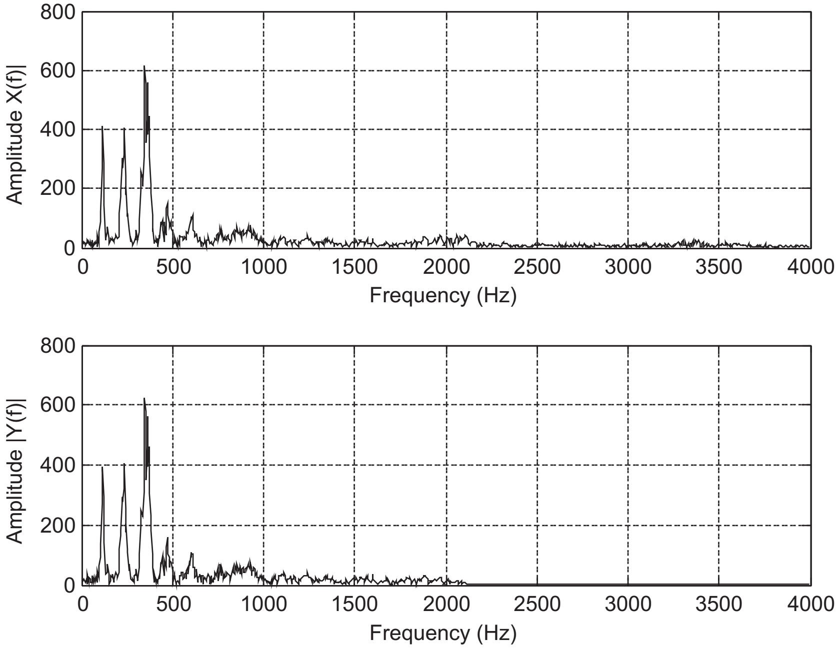

FIGURE 7.15B

Spectral plots of the original speech and processed speech by the lowpass filter.

We will continue to illustrate the determination of the filter length and cutoff frequency for other types of filters via the following examples.

EXAMPLE 7.9

Design a highpass FIR filter with the following specifications:

Stopband 0–1,500 Hz
Passband 2,500–4,000 Hz
Stopband attenuation 40 dB
Passband ripple 0.1 dB
Sampling rate 8,000 Hz

Solution:

Based on the specification, the Hanning window will do the job since it has a passband ripple of 0.0546 dB and stopband attenuation of 44 dB.

Then

$$\Delta f = |1,500 - 2,500|/8,000 = 0.125$$

Table 7.8 FIR Filter Coefficients in Example 7.9 (Hanning window)

Bhan: FIR Filter Coefficients (Hanning Window)	
$b_0 = b_{24} = 0.000000$	$b_1 = b_{23} = 0.000493$
$b_2 = b_{22} = 0.000000$	$b_3 = b_{21} = -0.005179$
$b_4 = b_{20} = 0.000000$	$b_5 = b_{19} = 0.016852$
$b_6 = b_{18} = 0.000000$	$b_7 = b_{17} = -0.040069$
$b_8 = b_{16} = 0.0000000$	$b_9 = b_{15} = 0.090565$
$b_{10} = b_{14} = 0.000000$	$b_{11} = b_{13} = -0.312887$
$b_{12} = 0.500000$	

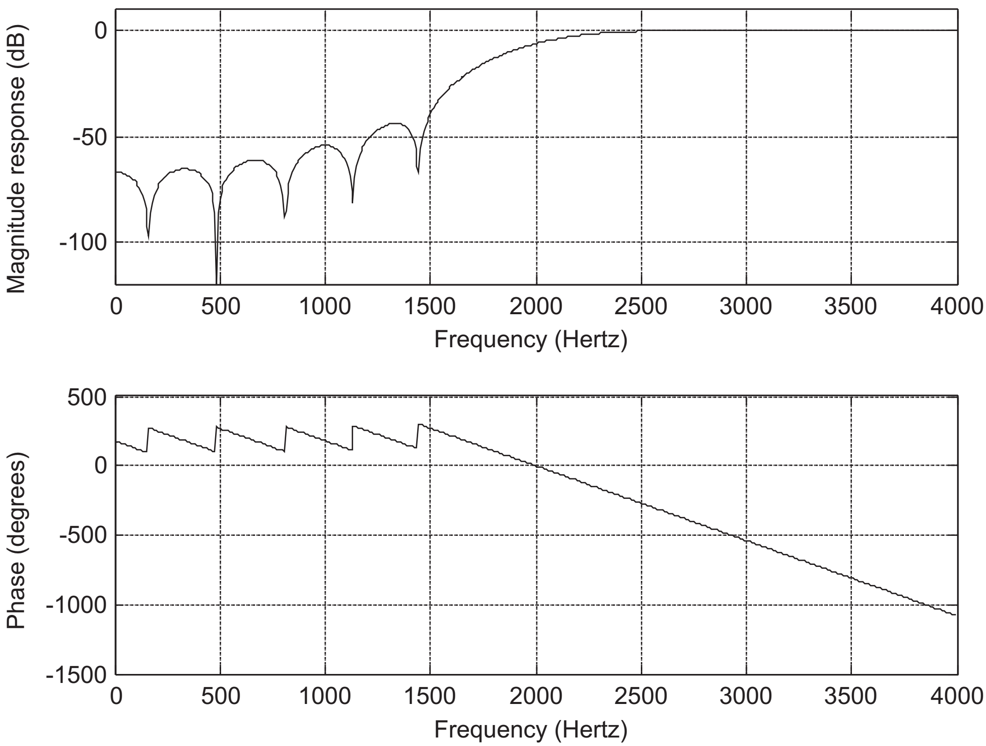

FIGURE 7.16

Frequency responses of the designed highpass filter using the Hanning window.

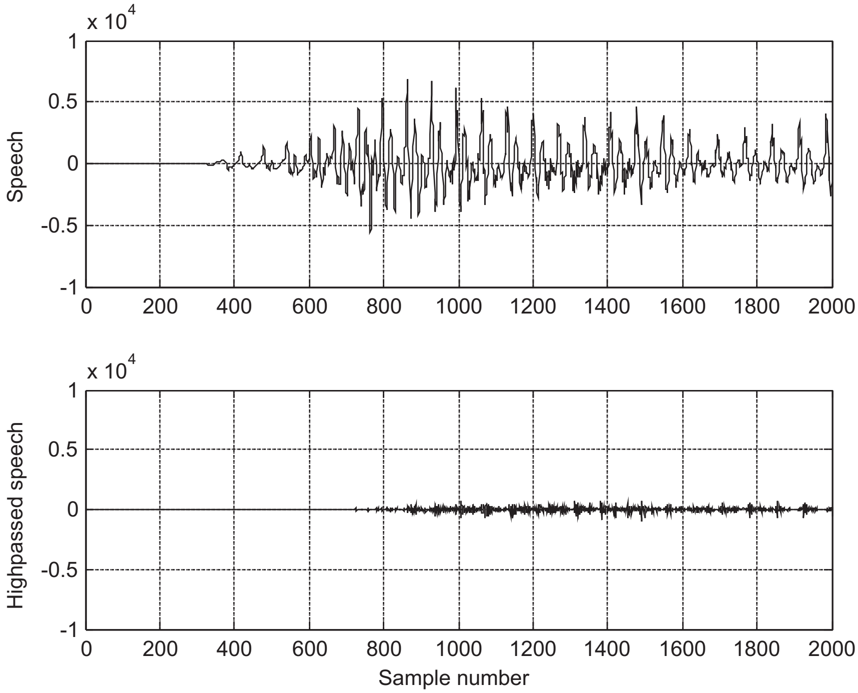

FIGURE 7.17A

Original speech and processed speech using the highpass filter.

$$N = 3.1/\Delta f = 24.2 \text{ (choose } N = 25)$$

Hence, we choose 25 filter coefficients using the Hanning window method. The cutoff frequency is $(1{,}500 + 2{,}500)/2 = 2{,}000$ Hz. The normalized cutoff frequency can be easily found as

$$\Omega_c = \frac{2{,}000 \times 2\pi}{8{,}000} = 0.5\pi \text{ radians}$$

Notice that $2M + 1 = 25$. The application program and design results are listed in Program 7.4 and Table 7.8.

Program 7.4. MATLAB program for Example 7.9

```
%Figure 7.16(Example 7.9)
% MATLAB program to create Figure 7.16
%
N=25; Ftype=2; WnL=0; WnH=0.5*pi; Wtype=3;fs=8000;
Bhan=firwd(N,Ftype,WnL,WnH,Wtype);
freqz(Bhan,1,512,fs);
axis([0 fs/2 -120 10]);
```

The corresponding frequency responses of the designed highpass FIR filter are displayed in Figure 7.16.

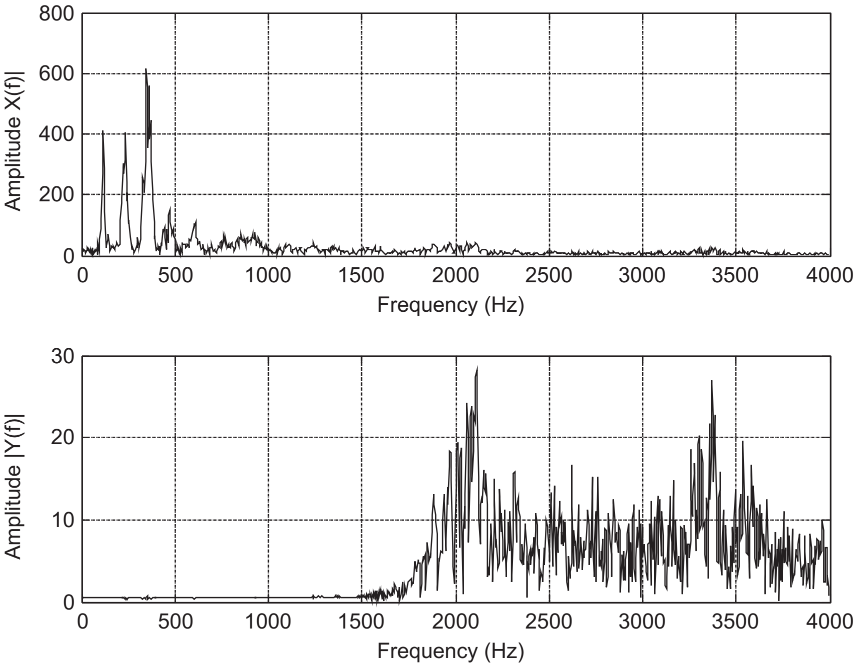

FIGURE 7.17B

Spectral comparison of the original speech and processed speech using the highpass filter.

Comparisons are given in Figure 7.17A, where the original speech and processed speech using the highpass filter are plotted, respectively. The high-frequency components of speech generally contain a small amount of energy. Figure 7.17B displays the spectral plots, where clearly the frequency components lower than 1.5 kHz are filtered. The processed speech would sound crisp.

EXAMPLE 7.10

Design a bandpass FIR filter with the following specifications:

Lower stopband 0–500 Hz
Passband 1,600–2,300 Hz
Upper stopband 3,500–4,000 Hz
Stopband attenuation 50 dB
Passband ripple 0.05 dB
Sampling rate 8,000 Hz

Solution:

$$\Delta f_1 = |1,600 - 500|/8,000 = 0.1375 \quad \text{and} \quad \Delta f_2 = |3,500 - 2,300|/8,000 = 0.15$$

$$N_1 = 3.3/0.1375 = 24 \quad \text{and} \quad N_2 = 3.3/0.15 = 22$$

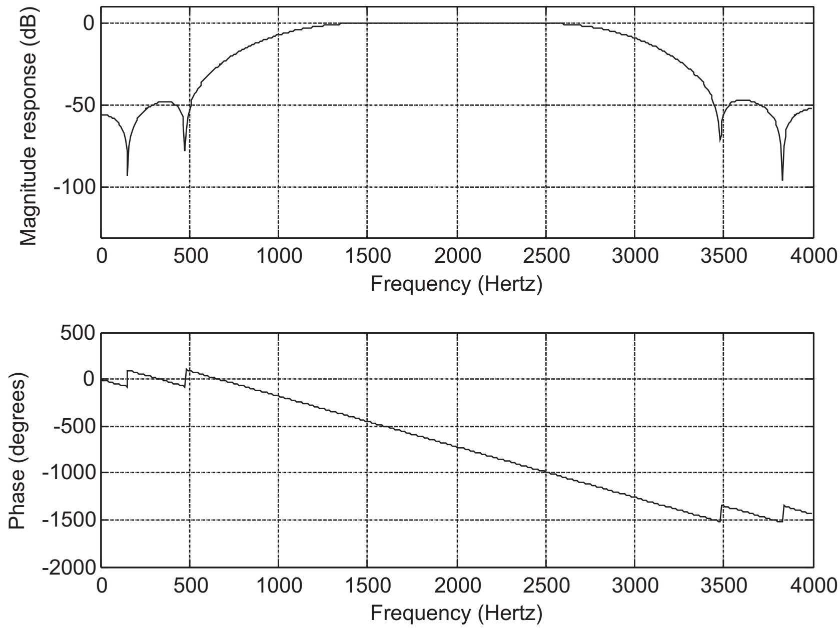

FIGURE 7.18

Frequency responses of the designed bandpass filter using the Hamming window.

We select $N = 25$ for the number of filter coefficients and using the Hamming window method, we next obtain

$$f_1 = (1,600 + 500)/2 = 1,050 \text{ Hz} \quad \text{and} \quad f_2 = (3,500 + 2,300)/2 = 2,900 \text{ Hz}$$

The normalized lower and upper cutoff frequencies are calculated as

$$\Omega_L = \frac{1,050 \times 2\pi}{8,000} = 0.2625\pi \text{ radians}$$

$$\Omega_H = \frac{2,900 \times 2\pi}{8,000} = 0.725\pi \text{ radians}$$

and $N = 2M + 1 = 25$. The design results are shown using the MATLAB program in Program 7.5.

Program 7.5. MATLAB program for Example 7.10.

```
%Figure 7.18(Example 7.10)
% MATLAB program to create Figure 7.18
%
N=25; Ftype=3; WnL=0.2625*pi; WnH=0.725*pi; Wtype=4;fs=8000;
Bham=firwd(N,Ftype,WnL,WnH,Wtype);
freqz(Bham,1,512,fs);
axis([0 fs/2 -130 10]);
```

Figure 7.18 depicts the frequency responses of the designed bandpass FIR filter. Table 7.9 lists the designed FIR filter coefficients.

Table 7.9 FIR Filter Coefficients in Example 7.10 (Hamming Window)

Bham: FIR Filter Coefficients (Hamming Window)	
$b_0 = b_{24} = 0.002680$	$b_1 = b_{23} = -0.001175$
$b_2 = b_{22} = -0.007353$	$b_3 = b_{21} = 0.000674$
$b_4 = b_{20} = -0.011063$	$b_5 = b_{19} = 0.004884$
$b_6 = b_{18} = 0.053382$	$b_7 = b_{17} = -0.003877$
$b_8 = b_{16} = 0.028520$	$b_9 = b_{15} = -0.008868$
$b_{10} = b_{14} = -0.296394$	$b_{11} = b_{13} = 0.008172$
$b_{12} = 0.462500$	

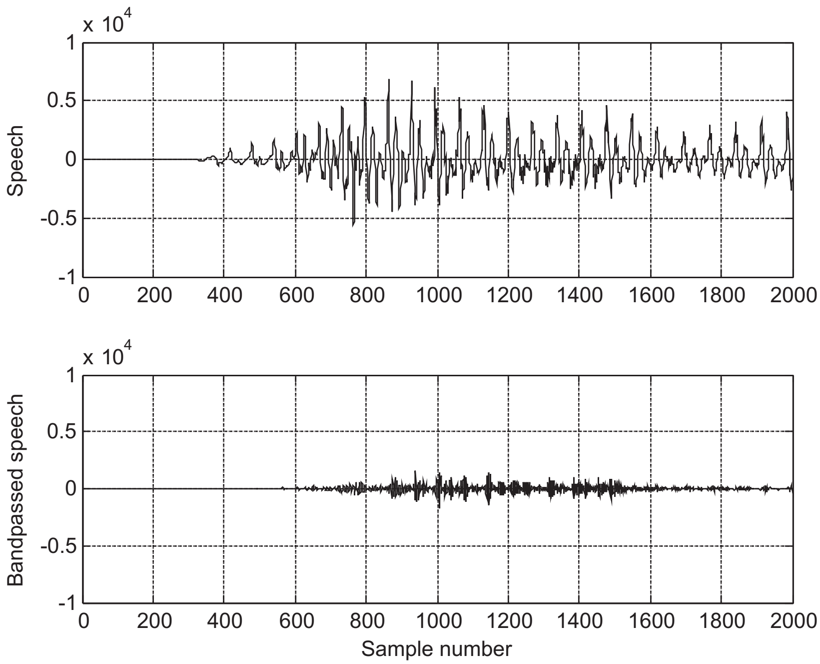

FIGURE 7.19A

Original speech and processed speech using the bandpass filter.

For comparison, the original speech and bandpass filtered speech are plotted in Figure 7.19A, where the bandpass frequency components contain a small portion of speech energy. Figure 7.19B shows a comparison indicating that low and high frequencies are removed by the bandpass filter.

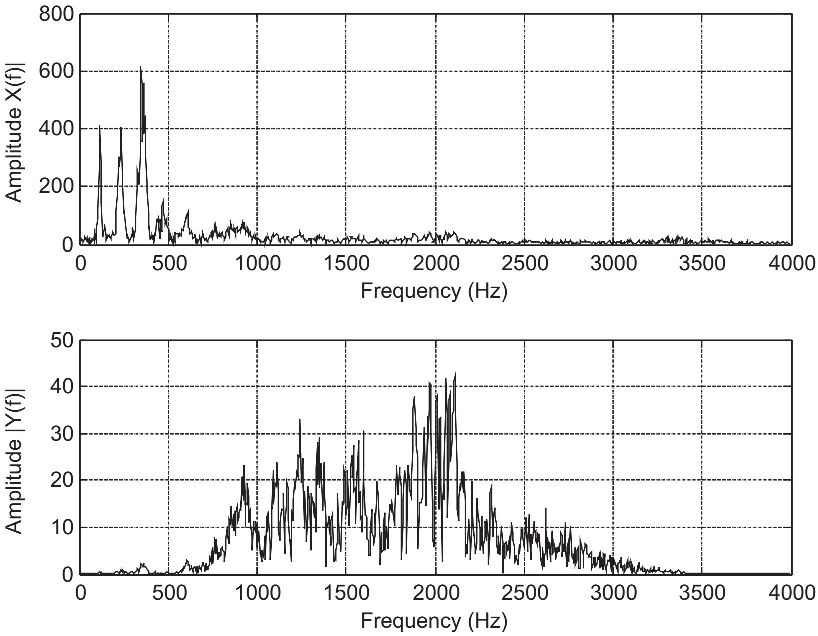

FIGURE 7.19B

Spectral comparison of the original speech and processed speech using the bandpass filter.

EXAMPLE 7.11

Design a bandstop FIR filter with the following specifications:

Lower cutoff frequency 1,250 Hz
Lower transition width 1,500 Hz
Upper cutoff frequency 2,850 Hz
Upper transition width 1,300 Hz
Stopband attenuation 60 dB
Passband ripple 0.02 dB
Sampling rate 8,000 Hz

Solution:

We can directly compute the normalized transition width:

$$\Delta f_1 = 1{,}500/8{,}000 = 0.1875 \quad \text{and} \quad \Delta f_2 = 1{,}300/8{,}000 = 0.1625$$

The filter lengths are determined using the Blackman windows as

$$N_1 = 5.5/0.1875 = 29.33, \quad \text{and} \quad N_2 = 5.5/0.1625 = 33.8$$

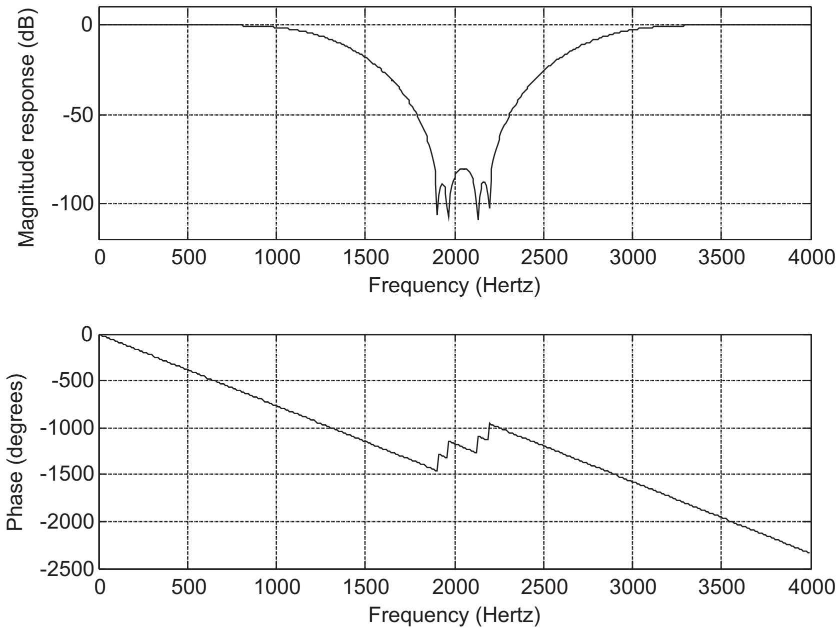

FIGURE 7.20

Frequency responses of the designed bandstop filter using the Blackman window.

We choose $N = 35$, an odd number. The normalized lower and upper cutoff frequencies are calculated as

$$\Omega_L = \frac{2\pi \times 1,250}{8,000} = 0.3125\pi \text{ radians}$$

$$\Omega_H = \frac{2\pi \times 2,850}{8,000} = 0.7125\pi \text{ radians}$$

and $N = 2M + 1 = 35$. Using MATLAB, the design results are demonstrated in Program 7.6.

Program 7.6. MATLAB program for Example 7.11.

```
%Figure 7.20 (Example 7.11)
% MATLAB program to create Figure 7.20
%
N=35; Ftype=4; WnL=0.3125*pi; WnH=0.7125*pi; Wtype=5;fs=8000;
Bblack=firwd(N,Ftype,WnL,WnH,Wtype);
freqz(Bblack,1,512,fs);
axis([0 fs/2 -120 10]);
```

Figure 7.20 shows the plot of the frequency responses of the designed bandstop filter. The designed filter coefficients are listed in Table 7.10.

Comparisons of filtering effects are illustrated in Figures 7.21A and 7.21B. In Figure 7.21A, the original speech and speech processed by the bandstop filter are plotted. The processed speech contains most of the energy of the original speech because most of the energy of the speech signal exists in the low-frequency band. Figure 7.21B verifies the filtering frequency effects. The requency components ranging from 2,000 Hz to 2,200 Hz have been completely removed.

Table 7.10 FIR Filter Coefficients in Example 7.11 (Blackman Window)

Bblack: FIR Filter Coefficients (Blackman Window)	
$b_0 = b_{34} = 0.000000$	$b_1 = b_{33} = 0.000059$
$b_2 = b_{32} = 0.000000$	$b_3 = b_{31} = 0.000696$
$b_4 = b_{30} = 0.001317$	$b_5 = b_{29} = -0.004351$
$b_6 = b_{28} = -0.002121$	$b_7 = b_{27} = 0.000000$
$b_8 = b_{26} = -0.004249$	$b_9 = b_{25} = 0.027891$
$b_{10} = b_{24} = 0.011476$	$b_{11} = b_{23} = -0.036062$
$b_{12} = b_{22} = 0.000000$	$b_{13} = b_{21} = -0.073630$
$b_{14} = b_{20} = -0.020893$	$b_{15} = b_{19} = 0.285306$
$b_{16} = b_{18} = 0.014486$	$b_{17} = 0.600000$

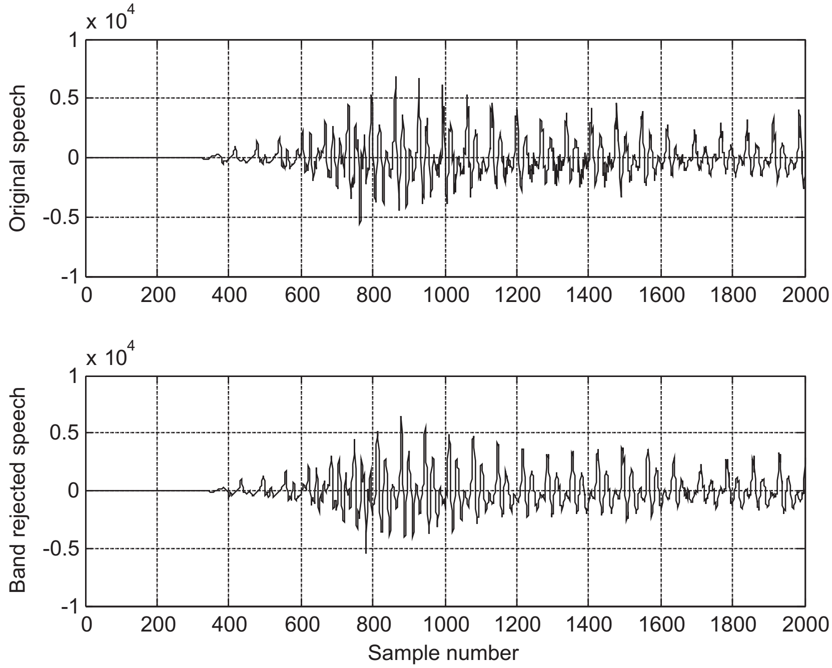

FIGURE 7.21A

Original speech and processed speech using the bandstop filter.

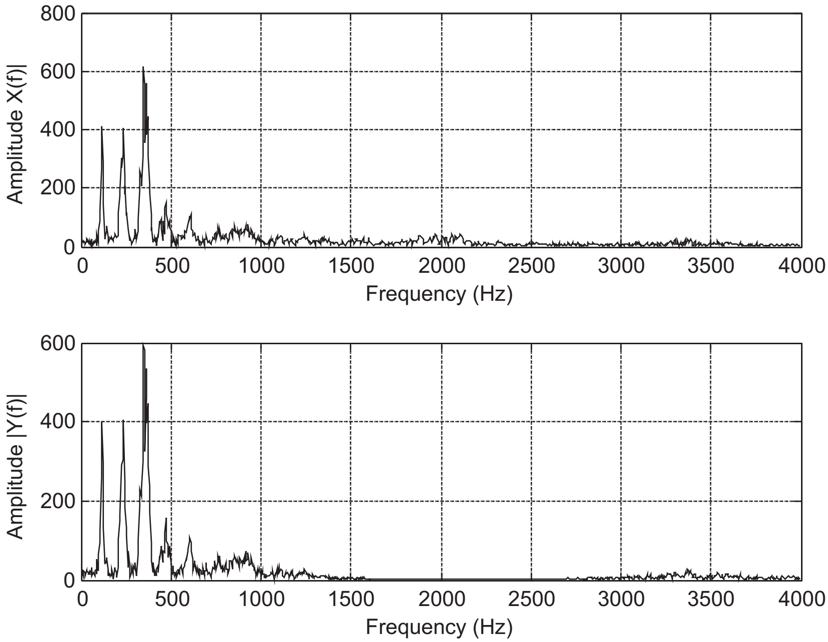

FIGURE 7.21B

Spectral comparison of the original speech and processed speech using the bandstop filter.

7.4 APPLICATIONS: NOISE REDUCTION AND TWO-BAND DIGITAL CROSSOVER

In this section, we will investigate noise reduction and digital crossover design using FIR filters.

7.4.1 Noise Reduction

One of the key digital signal processing (DSP) applications is noise reduction. In this application, a digital FIR filter removes noise in a signal that is contaminated by noise existing in a broad frequency range. For example, such noise often appears during the data acquisition process. In real-world applications, the desired signal usually occupies a certain frequency range. We can design a digital filter to remove frequency components other than the desired frequency range.

In a data acquisition system, we record a 500 Hz sine wave at a sampling rate of 8,000 samples per second. The signal is corrupted by broadband noise $v(n)$:

$$x(n) = 1.4141 \cdot \sin(2\pi \cdot 500n/8,000) + v(n)$$

The 500 Hz signal with noise and its spectrum are plotted in Figure 7.22, from which it is obvious that the digital sine wave contains noise. The spectrum is also displayed to give a better understanding of the noise frequency level. We can see that noise is broadband, existing from 0 Hz to the folding frequency of 4,000 Hz. Assuming that the desired signal has a frequency range of only 0 to 800 Hz, we can filter noise from 800 Hz and beyond. A lowpass filter would complete such a task. Then we develop the filter specifications:

Passband frequency range: 0 Hz to 800 Hz with the passband ripple less than 0.02 dB
Stopband frequency range: 1 kHz to 4 kHz with 50 dB attenuation

As we will see, lowpass filtering will remove noise in the range 1,000 Hz to 4,000 Hz , and hence the signal to noise power ratio will be improved.

Based on the specifications, we design an FIR filter with the Hamming window, a cutoff frequency of 900 Hz, and an estimated filter length of 133 taps. The enhanced signal is depicted in Figure 7.23, where the clean signal can be observed. The amplitude spectrum for the enhanced signal is also plotted. As shown in the spectral plot, the noise level is negligible between 1 and 4 kHz. Notice that since we use the higher-order FIR filter, the signal experiences a linear phase delay of 66 samples, as is expected. We also see some transient response effects. However, the transient response effects will end after the first 132 samples due to the length of the FIR filter. MATLAB implementation is given in Program 7.7.

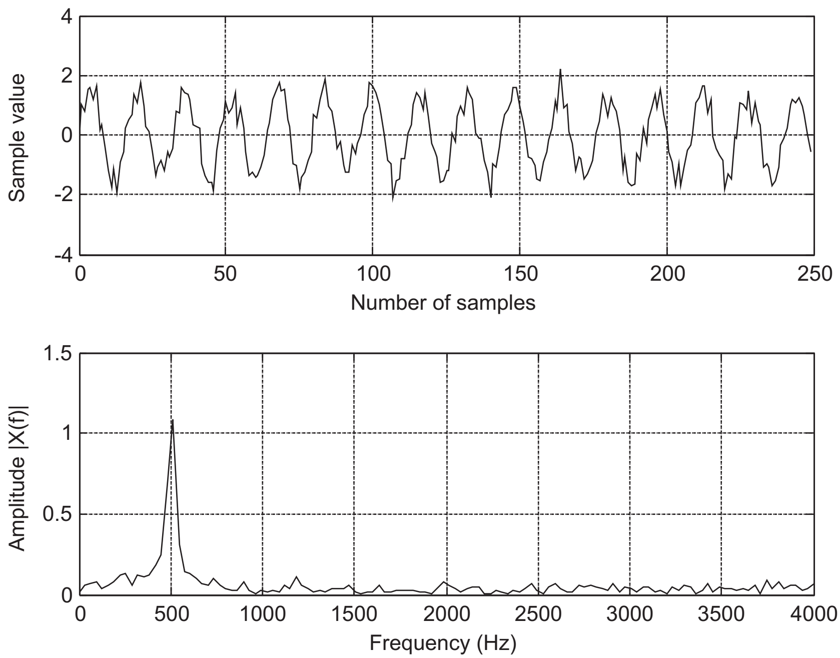

FIGURE 7.22

Signal with noise and its spectrum.

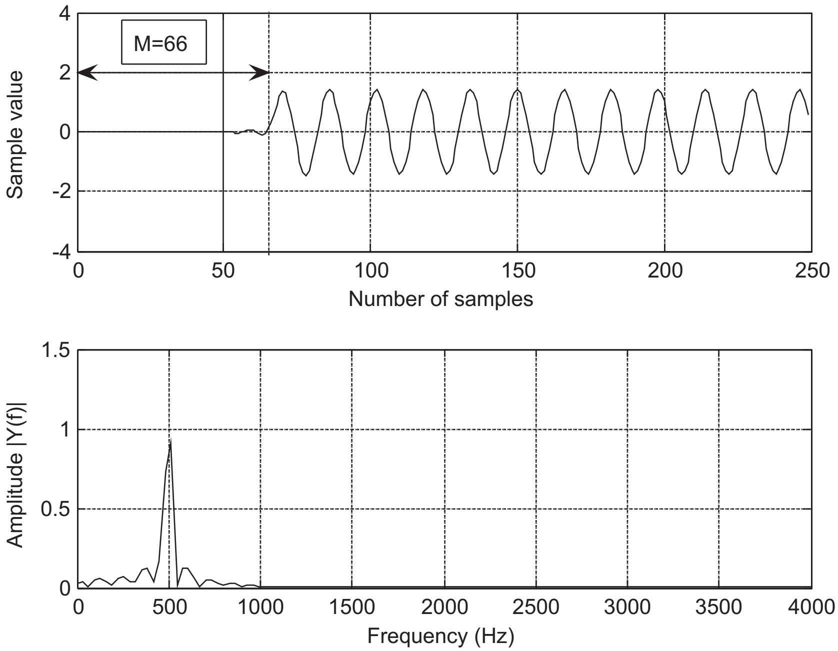

FIGURE 7.23

The clean signal and spectrum with noise removed.

Program 7.7. MATLAB program for the application of noise filtering.

```
close all; clear all
fs=8000;                                % Sampling rate
T=1/fs;                                 % Sampling period
v=sqrt(0.1)*randn(1,250);               % Generate Gaussian random noise
n=0:1:249;                              % Indexes
x=sqrt(2)*sin(2*pi*500*n*T)+v;          % Generate 500-Hz sinusoid plus noise
subplot(2,1,1);plot(n,x);
xlabel('Number of samples');ylabel('Sample value');grid;
N=length(x);
f=[0:N/2]*fs/N;
Axk=2*abs(fft(x))/N;Axk(1)=Axk(1)/2;    % Calculate single side spectrum for x(n)
subplot(2,1,2); plot(f,Axk(1:N/2+1));
xlabel('Frequency (Hz)'); ylabel('Amplitude |X(f)| ');grid;
figure
Wnc=2*pi*900/fs;                        % Determine the normalized digital cutoff
                                        % frequency
B=firwd(133,1,Wnc,0,4);                 % Design FIR filter
y=filter(B,1,x);                        % Perform digital filtering
Ayk=2*abs(fft(y))/N;Ayk(1)=Ayk(1)/2;    % Single-side spectrum of the filtered data
subplot(2,1,1); plot(n,y);
```

```
xlabel('Number of samples');ylabel('Sample value');grid;
subplot(2,1,2);plot(f,Ayk(1:N/2+1)); axis([0 fs/2 0 1.5]);
xlabel('Frequency (Hz)'); ylabel('Amplitude |Y(f)| ');grid;
```

7.4.2 Speech Noise Reduction

In a speech recording system, we digitally record speech in a noisy environment at a sampling rate of 8,000 Hz. Assuming the recorded speech contains information within 1,800 Hz, we can design a lowpass filter to remove the noise between 1,800 Hz and the Nyquist limit (the folding frequency of 4,000 Hz). The filter specifications are listed below:

Filter type: lowpass FIR filter
Passband frequency range: 0–1,800 Hz
Passband ripple: 0.02 dB
Stopband frequency range: 2,000–4,000 Hz
Stopband attenuation: 50 dB

According to these specifications, we can determine the following parameters for filter design:

Window type = Hamming window
Number of filter taps = 133
Lowpass cutoff frequency = 1,900 Hz

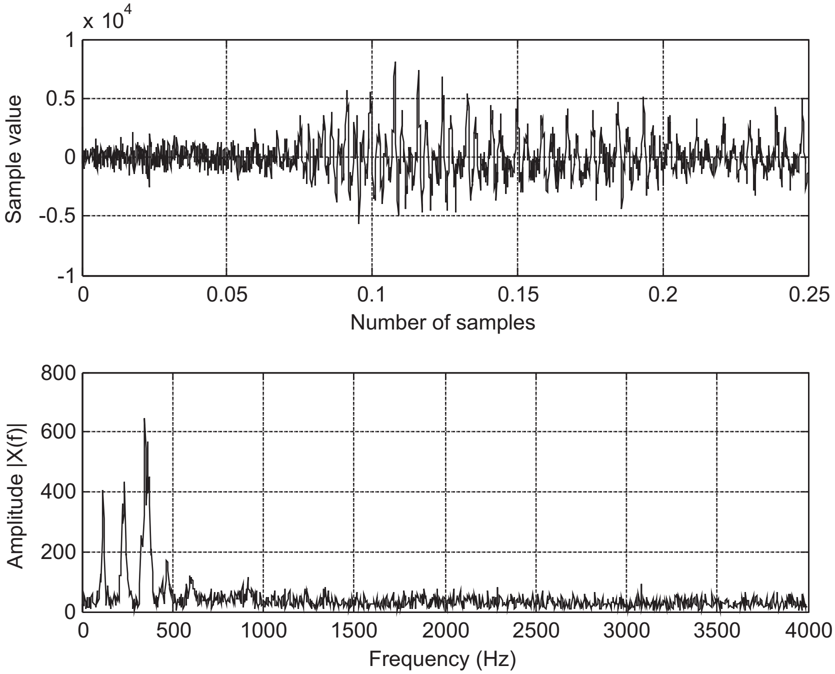

FIGURE 7.24A

Noisy speech and its spectrum.

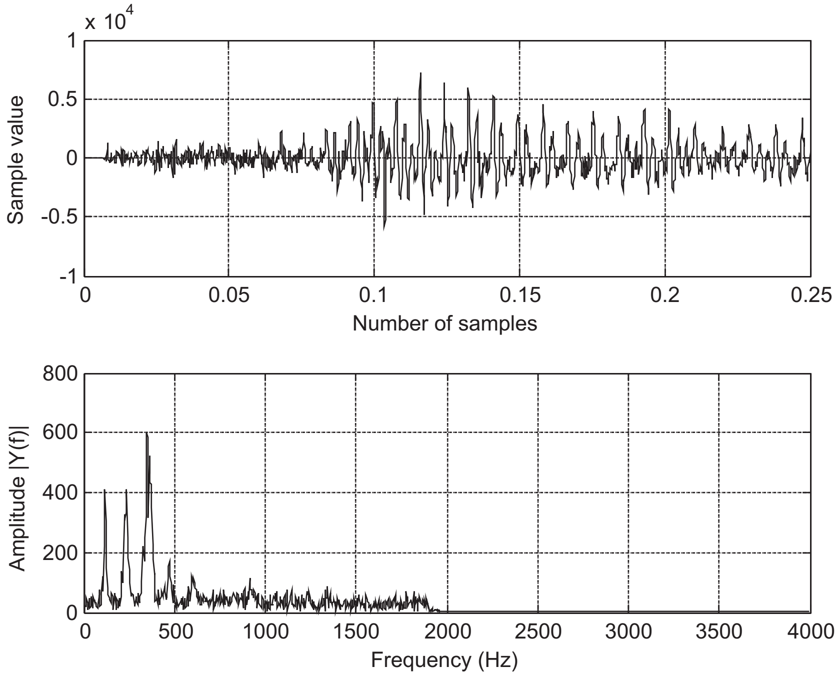

FIGURE 7.24B

Enhanced speech and its spectrum.

Figure 7.24A shows the plots of the recorded noisy speech and its spectrum. As we can see in the noisy spectrum, the noise level is high and broadband. After applying the designed lowpass filter, we plot the filtered speech and its spectrum shown in Figure 7.24B, where the clean speech is clearly identified, while the spectrum shows that the noise components above 2 kHz have been completely removed.

7.4.3 Noise Reduction in Vibration Signals

In a data acquisition system for vibration analysis, a vibration signal is captured using an accelerometer sensor in the noisy environment. The sampling rate is 1,000 Hz. The captured signal is significantly corrupted by a broadband noise. Vibration analysis requires the first dominant frequency component in the range from 35 to 50 Hz to be retrieved. We list the filter specifications below:

Filter type = bandpass FIR filter
Passband frequency range = 35–50 Hz
Passband ripple = 0.02 dB
Stopband frequency ranges = 0–15 and 70–500 Hz
Stopband attenuation: 50 dB

According to these specifications, we can determine the following parameters for the filter design:

Window type = Hamming window
Number of filter taps = 167
Low cutoff frequency = 25 Hz
High cutoff frequency = 65 Hz

Figure 7.25 displays the plots of the recorded noisy vibration signal and its spectrum. Figure 7.26 shows the retrieved vibration signal with noise reduction by a bandpass filter.

7.4.4 Two-Band Digital Crossover

In audio systems, there is often a situation where the application requires the entire audible range of frequencies, but this is beyond the capability of any single speaker driver. So, we combine several drivers, such as the speaker cone and horns, each covering a different frequency range, to reproduce the full audio frequency range.

A typical two-band digital crossover can be designed as shown in Figure 7.27. There are two speaker drivers. The woofer responds to low frequencies, and the tweeter responds to high frequencies. The incoming digital audio signal is split into two bands by using a lowpass filter and a highpass filter in parallel. We then amplify the separated audio signals and send them to their respective corresponding speaker drivers. Hence, the objective is to design the lowpass filter and the highpass filter so

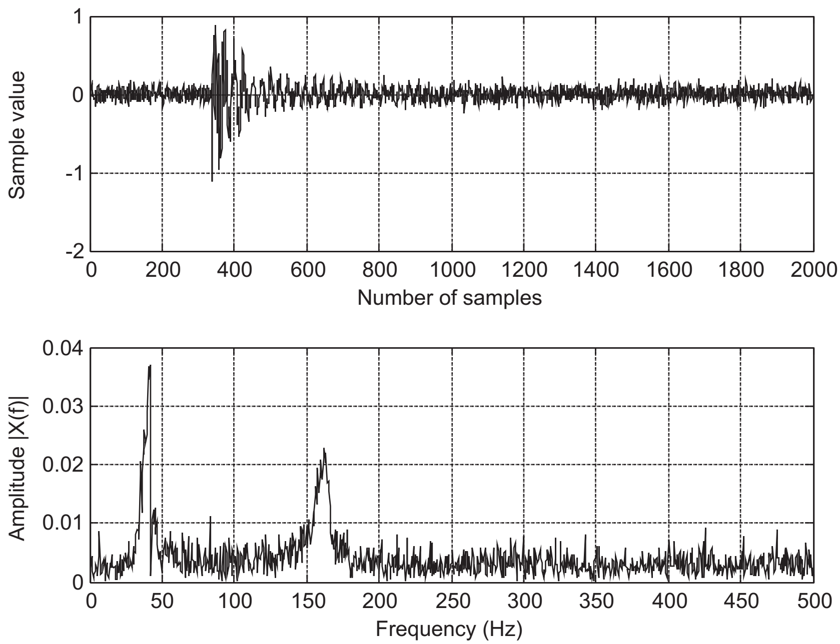

FIGURE 7.25

Noisy vibration signal and its spectrum.

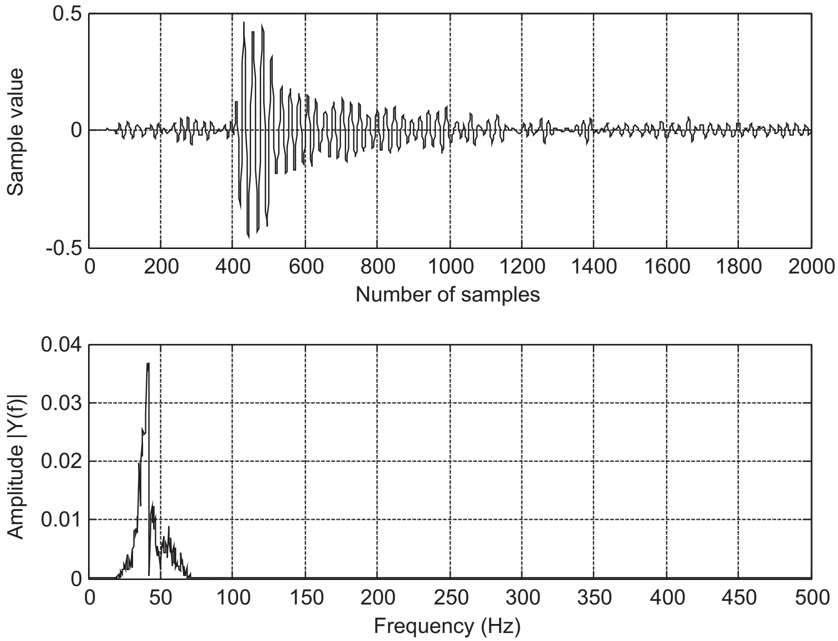

FIGURE 7.26

Retrieved vibration signal and its spectrum.

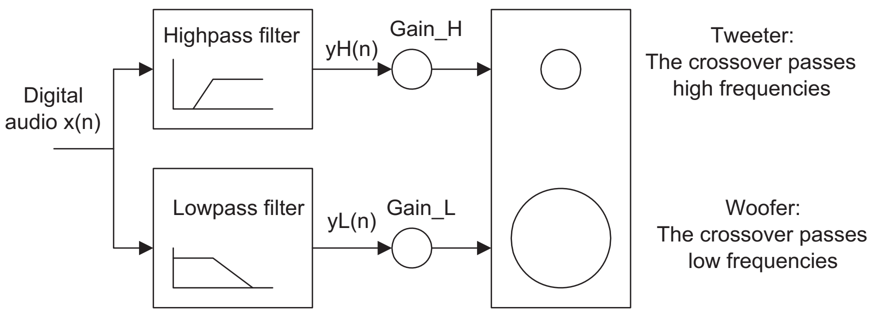

FIGURE 7.27

Two-band digital crossover.

that their combined frequency response is flat, while keeping transitions as sharp as possible to prevent audio signal distortion in the transition frequency range. Although traditional crossover systems are designed using active circuits (analog systems) or passive circuits, the digital crossover system provides a cost-effective solution with programmability, flexibility, and high quality.

A crossover system has the following specifications:

Sampling rate = 44,100 Hz
Crossover frequency = 1,000 Hz (cutoff frequency)
Transition band = 600 Hz to 1,400 Hz
Lowpass filter = passband frequency range from 0 to 600 Hz with a ripple of 0.02 dB and stopband edge at 1,400 Hz with an attenuation of 50 dB
Highpass filter = passband frequency range from 1.4 to 44.1 kHz with ripple of 0.02 dB and stopband edge at 600 Hz with an attenuation of 50 dB

In the design of this crossover system, one possibility is to use an FIR filter, since it provides a linear phase for the audio system. However, an infinite impulse response (IIR) filter (which will be discussed in the next chapter) is a possible alternative. Based on the transition band of 800 Hz and the passband ripple and stopband attenuation requirements, the Hamming window is chosen for both lowpass and highpass filters. We can determine the number of filter taps as 183, each with a cutoff frequency of 1,000 Hz.

The frequency responses for the designed lowpass and highpass filters are given in Figure 7.28A; the lowpass filter, highpass filter, and combined responses appear in Figure 7.28B. As we can see, the crossover frequency for both filters is at 1,000 Hz, and the combined frequency response is perfectly flat. The impulse responses (filter coefficients) for the lowpass and highpass filters are plotted in Figure 7.28C.

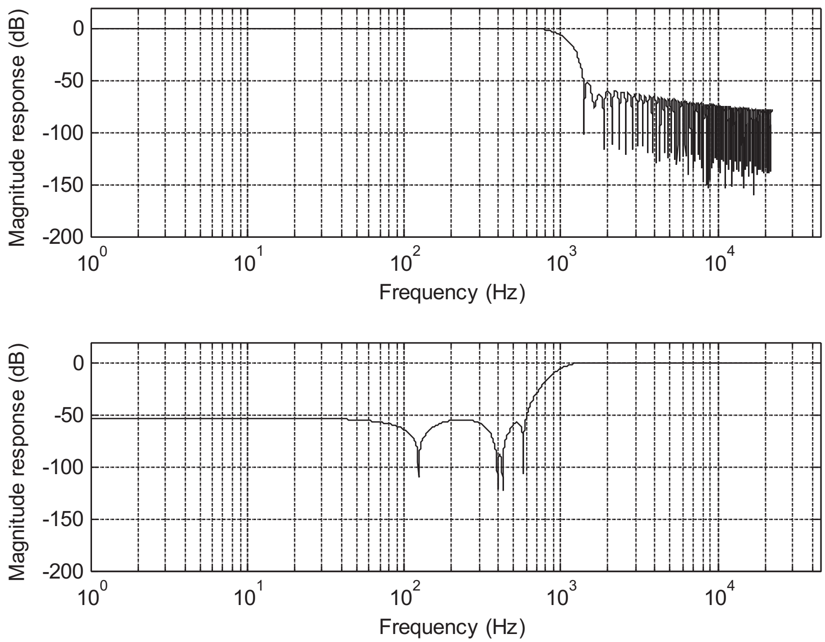

FIGURE 7.28A

Magnitude frequency responses for lowpass filter and highpass filter.

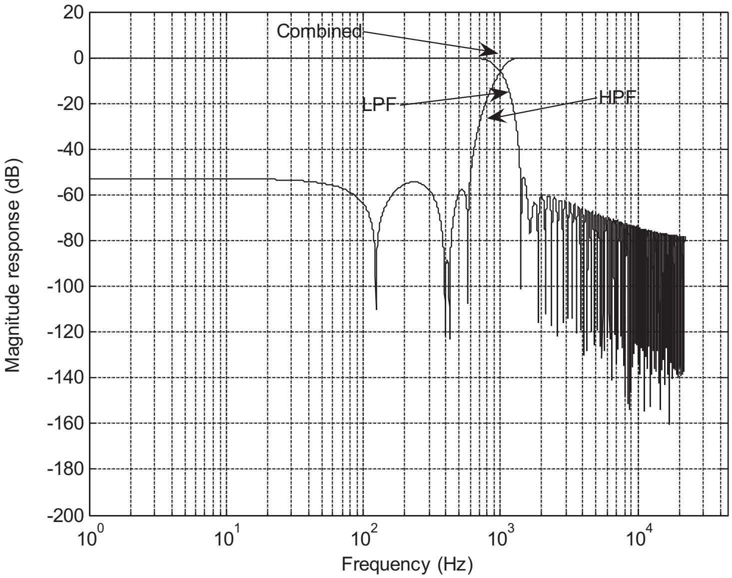

FIGURE 7.28B

Magnitude frequency responses for both the lowpass and highpass filters, and the combined magnitude frequency response for the digital audio crossover system.

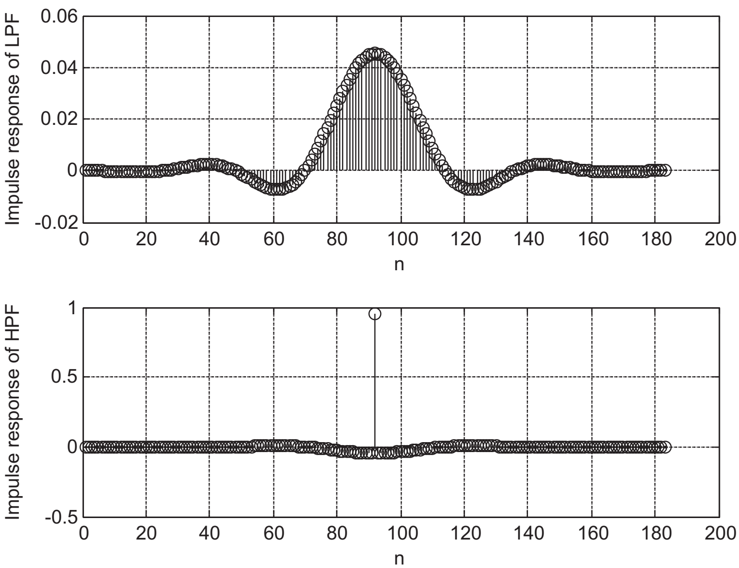

FIGURE 7.28C

Impulse responses of both the FIR lowpass filter and the FIR highpass filter for the digital audio crossover system.

7.5 FREQUENCY SAMPLING DESIGN METHOD

In addition to methods of Fourier transform design and Fourier transform with windowing discussed in the previous section, *frequency sampling* is another alternative. The key feature of frequency sampling is that the filter coefficients can be calculated based on the specified magnitudes of the desired filter frequency response uniformly in the frequency domain. Hence, it has design flexibility.

To begin development, we let $h(n)$, for $n = 0, 1, \cdots, N-1$, be the causal impulse response (FIR filter coefficients) that approximates the FIR filter, and we let $H(k)$, for $k = 0, 1, \cdots, N-1$, represent the corresponding discrete Fourier transform (DFT) coefficients. We obtain $H(k)$ by sampling the desired frequency filter response $H(k) = H(e^{j\Omega_k})$ at equally spaced instants in frequency domain, as shown in Figure 7.29.

Then, according to the definition of the inverse DFT (IDFT), we can calculate the FIR coefficients:

$$h(n) = \frac{1}{N}\sum_{k=0}^{N-1} H(k) W_N^{-kn}, \quad \text{for} \quad n = 0, 1, \cdots, N-1 \tag{7.27}$$

where

$$W_N = e^{-j\frac{2\pi}{N}} = \cos\left(\frac{2\pi}{N}\right) - j\sin\left(\frac{2\pi}{N}\right)$$

We assume that the FIR filter has linear phase and the number of taps is $N = 2M + 1$. Equation (7.27) can be significantly simplified as

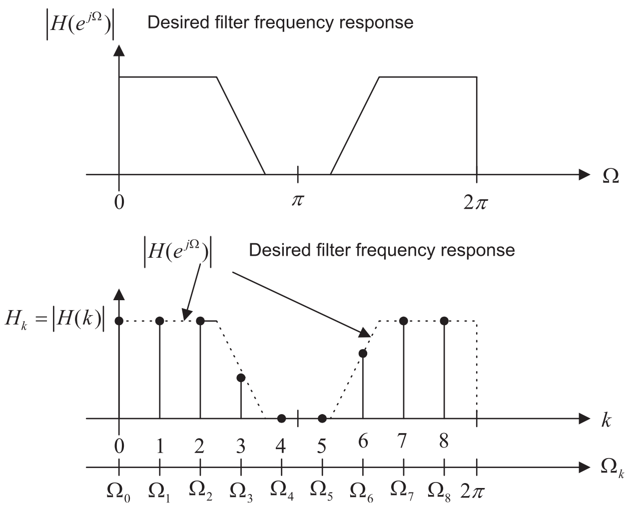

FIGURE 7.29

Desired filter frequency response and sampled frequency response.

$$h(n) = \frac{1}{2M+1}\left\{H_0 + 2\sum_{k=1}^{M} H_k \cos\left(\frac{2\pi k(n-M)}{2M+1}\right)\right\}, \quad \text{for} \quad n = 0, 1, \cdots, 2M \tag{7.28}$$

where H_k, for $k = 0, 1, \cdots, 2M$, represents the magnitude values specifying the desired filter frequency response sampled at $\Omega_k = \frac{2\pi k}{(2M+1)}$. The derivation is detailed in Appendix E. The design procedure is therefore simply summarized as follows:

1. Given the filter length of $2M+1$, specify the magnitude frequency response for the normalized frequency range from 0 to π:

$$H_k \text{ at } \Omega_k = \frac{2\pi k}{(2M+1)} \quad \text{for} \quad k = 0, 1, \cdots, M \tag{7.29}$$

2. Calculate the FIR filter coefficients:

$$b_n = h(n) = \frac{1}{2M+1}\left\{H_0 + 2\sum_{k=1}^{M} H_k \cos\left(\frac{2\pi k(n-M)}{2M+1}\right)\right\} \quad \text{for} \quad n = 0, 1, \cdots, M \tag{7.30}$$

3. Use symmetry (linear phase requirement) to determine the rest of coefficients:

$$h(n) = h(2M-n) \quad \text{for} \quad n = M+1, \cdots, 2M \tag{7.31}$$

Example 7.12 illustrates the design procedure.

EXAMPLE 7.12

Design a linear phase lowpass FIR filter with 7 taps and a cutoff frequency of $\Omega_c = 0.3\pi$ radians using the frequency sampling method.

Solution:

Since $N = 2M+1 = 7$ and $M = 3$, the sampled frequencies are given by

$$\Omega_k = \frac{2\pi}{7}k \text{ radians}, \ k = 0, 1, 2, 3$$

Next we specify the magnitude values H_k at the specified frequencies as follows:

$$\text{for } \Omega_0 = 0 \text{ radians}, \ H_0 = 1.0$$

$$\text{for } \Omega_1 = \frac{2}{7}\pi \text{ radians}, \ H_1 = 1.0$$

$$\text{for } \Omega_2 = \frac{4}{7}\pi \text{ radians}, \ H_2 = 0.0$$

$$\text{for } \Omega_3 = \frac{6}{7}\pi \text{ radians}, \ H_3 = 0.0$$

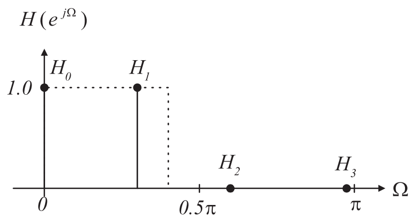

FIGURE 7.30

Sampled values of the frequency response in Example 7.12.

Figure 7.30 shows the specifications.

Using Equation (7.30), we achieve

$$h(n) = \frac{1}{7}\left\{1 + 2\sum_{k=1}^{3} H_k \cos\left[2\pi k(n-3)/7\right]\right\},\ n = 0, 1, \cdots, 3.$$

$$= \frac{1}{7}\{1 + 2\cos\left[2\pi(n-3)/7\right]\}$$

Thus, computing the FIR filter coefficients yields

$$h(0) = \frac{1}{7}\{1 + 2\cos(-6\pi/7)\} = -0.11456$$

$$h(1) = \frac{1}{7}\{1 + 2\cos(-4\pi/7)\} = 0.07928$$

$$h(2) = \frac{1}{7}\{1 + 2\cos(-2\pi/7)\} = 0.32100$$

$$h(3) = \frac{1}{7}\{1 + 2\cos(-0\times\pi/7)\} = 0.42857$$

By symmetry, we obtain the rest of the coefficients as follows:

$$h(4) = h(2) = 0.32100$$

$$h(5) = h(1) = 0.07928$$

$$h(6) = h(0) = -0.11456$$

The following two examples are devoted to illustrating the FIR filter design using the frequency sampling method. A MATLAB program, **firfs(N, Hk),** is provided in the "MATLAB Programs" section at the end of this chapter (see its usage in Table 7.11) to implement the design in Equation (7.30) with the input parameters of $N = 2M + 1$

Table 7.11 Illustrative Usage for MATLAB Function **firfs(N, Hk)**

```
function B=firfs(N,Hk)
% B=firfs(N,Hk)
% FIR filter design using the frequency sampling method.
% Input parameters:
% N: the number of filter coefficients.
%     note: N must be odd number.
% Hk: sampled frequency response for k=0,1,2,...,M=(N-1)/2.
% Output:
% B: FIR filter coefficients.
```

(number of taps) and a vector **Hk** containing the specified magnitude values $H_k, k = 0, 1, \cdots, M$. Finally, the MATLAB function will return the calculated FIR filter coefficients.

EXAMPLE 7.13

a. Design a linear phase lowpass FIR filter with 25 coefficients using the frequency sampling method. Let the cutoff frequency be 2,000 Hz and assume a sampling frequency of 8,000 Hz.
b. Plot the frequency responses.
c. List the FIR filter coefficients.

Solution:

a. The normalized cutoff frequency for the lowpass filter is $\Omega_c = \omega T = 2\pi 2000/8000 = 0.5\pi$ radians, $N = 2M + 1 = 25$, and the specified values of the sampled frequency response are chosen to be

$$H_k = [1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0]$$

MATLAB Program 7.8 produces the design results.

Program 7.8. MATLAB program for Example 7.13.

```
%Figure 7.31(Example 7.13)
% MATLAB program to create Figure 7.31
fs=8000;  % Sampling frequency
H1=[1 1 1 1 1 1 1 0 0 0 0 0 0];   % Magnitude specifications
B1=firfs(25,H1);                  % Design filter
[h1,f]=freqz(B1,1,512,fs);        % Calculate magnitude frequency response
H2=[1 1 1 1 1 1 1 0.5 0 0 0 0 0]; % Magnitude specifications
B2=firfs(25,H2);                  % Design filter
[h2,f]=freqz(B2,1,512,fs);        % Calculate magnitude frequency response
p1=180*unwrap(angle(h1))/pi;
p2=180*unwrap(angle(h2))/pi
subplot(2,1,1); plot(f,20*log10(abs(h1)),'-.',f,20*log10(abs(h2)));grid
axis([0 fs/2 -80 10]);
xlabel('Frequency (Hz)'); ylabel('Magnitude Response (dB)');
subplot(2,1,2); plot(f,p1,'-.',f,p2);grid
xlabel('Frequency (Hz)'); ylabel('Phase (degrees)');
```

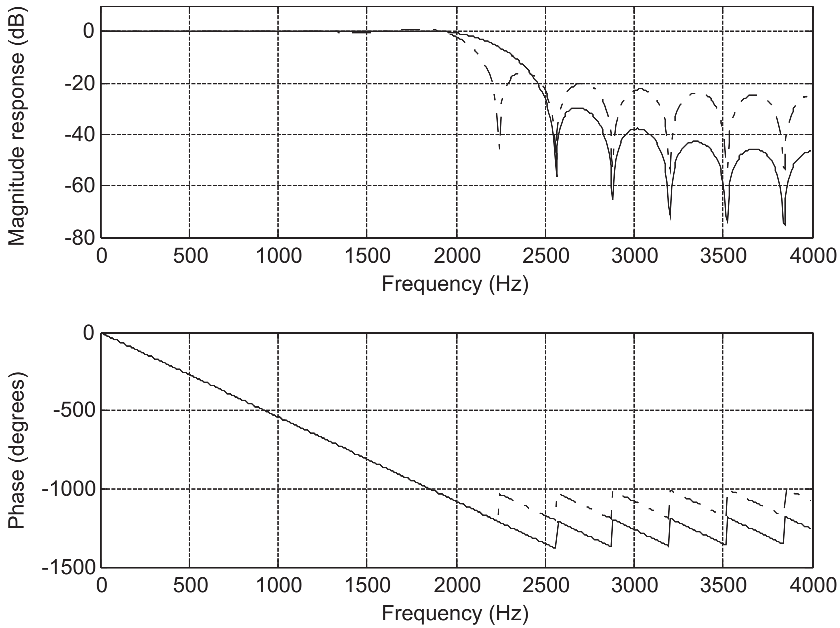

FIGURE 7.31

Frequency responses using the frequency sampling method in Example 7.13.

Table 7.12 FIR Filter Coefficients in Example 7.13 (frequency sampling method)

B1: FIR Filter Coefficients	B2: FIR Filter Coefficients
$b_0 = b_{24} = 0.027436$	$b_0 = b_{24} = 0.001939$
$b_1 = b_{23} = -0.031376$	$b_1 = b_{23} = 0.003676$
$b_2 = b_{22} = -0.024721$	$b_2 = b_{22} = -0.012361$
$b_3 = b_{21} = 0.037326$	$b_3 = b_{21} = -0.002359$
$b_4 = b_{20} = 0.022823$	$b_4 = b_{20} = 0.025335$
$b_5 = b_{19} = -0.046973$	$b_5 = b_{19} = -0.008229$
$b_6 = b_{18} = -0.021511$	$b_6 = b_{18} = -0.038542$
$b_7 = b_{17} = 0.064721$	$b_7 = b_{17} = 0.032361$
$b_8 = b_{16} = 0.020649$	$b_8 = b_{16} = 0.049808$
$b_9 = b_{15} = -0.106734$	$b_9 = b_{15} = -0.085301$
$b_{10} = b_{14} = -0.020159$	$b_{10} = b_{14} = -0.057350$
$b_{11} = b_{13} = 0.318519$	$b_{11} = b_{13} = 0.311024$
$b_{12} = 0.520000$	$b_{12} = 0.560000$

b. The magnitude frequency response plotted using the dash-dotted line is displayed in Figure 7.31, where it is observed that oscillations occur in the passband and stopband of the designed FIR filter. This is due to the abrupt change of the specification in the transition band (between the passband and the stopband). To reduce this ripple effect, the modified specification with a smooth transition band, H_k, $k = 0, 1, \cdots, 13$, is used:

$$H_k = [1\ 1\ 1\ 1\ 1\ 1\ 1\ 0.5\ 0\ 0\ 0\ 0\ 0\ 0]$$

The improved magnitude frequency response is shown in Figure 7.31 via the solid line.
c. The calculated FIR coefficients for both filters are listed in Table 7.12.

EXAMPLE 7.14

a. Design a linear phase bandpass FIR filter with 25 coefficients using the frequency sampling method. Let the lower and upper cutoff frequencies be 1,000 Hz and 3,000 Hz, respectively, and assume a sampling frequency of 8,000 Hz.
b. List the FIR filter coefficients.
c. Plot the frequency responses.

Solution:
a. First we calculate the normalized lower and upper cutoff frequencies for the bandpass filter; that is, $\Omega_L = 2\pi \times 1,000/8,000 = 0.25\pi$ radians and $\Omega_H = 2\pi \times 3,000/8,000 = 0.75\pi$ radians, respectively. The sampled values of the bandpass frequency response are specified by the following vector:

$$H_k = [0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0]$$

For comparison, a second specification of H_k with a smooth transition band is used:

$$H_k = [0\ 0\ 0\ 0.5\ 1\ 1\ 1\ 1\ 1\ 0.5\ 0\ 0\ 0]$$

b. The MATLAB list is shown in Program 7.9. The generated FIR coefficients are listed in Table 7.13.

Program 7.9 MATLAB program for Example 7.14.

```
% Figure 7.32 (Example 7.14)
% MATLAB program to create Figure 7.32
%
fs=8000;
H1=[0 0 0 0 1 1 1 1 1 0 0 0 0];     % Magnitude specifications
B1=firfs(25,H1);                    % Design filter
[h1,w]=freqz(B1,1,512);             % Calculate magnitude frequency response
H2=[0 0 0 0.5 1 1 1 1 1 0.5 0 0 0]; % Magnitude spectrum
B2=firfs(25,H2);                    % Design filter
[h2,w]=freqz(B2,1,512);             % Calculate magnitude frequency response
p1=180*unwrap(angle(h1)')/pi;
p2=180*unwrap(angle(h2)')/pi
subplot(2,1,1); plot(f,20*log10(abs(h1)),'-.',f,20*log10(abs(h2)));grid
axis([0 fs/2 -100 10]);
xlabel('Frequency (Hz)'); ylabel('Magnitude Response (dB)');
subplot(2,1,2); plot(f,p1,'-.',f,p2);grid
xlabel('Frequency (Hz)'); ylabel('Phase (degrees)');
```

Table 7.13 FIR Filter Coefficients in Example 7.14 (frequency sampling method)

B1: FIR Filter Coefficients	B2: FIR Filter Coefficients
$b_0 = b_{24} = 0.055573$	$b_0 = b_{24} = 0.001351$
$b_1 = b_{23} = -0.030514$	$b_1 = b_{23} = -0.008802$
$b_2 = b_{22} = 0.000000$	$b_2 = b_{22} = -0.020000$
$b_3 = b_{21} = -0.027846$	$b_3 = b_{21} = 0.009718$
$b_4 = b_{20} = -0.078966$	$b_4 = b_{20} = -0.011064$
$b_5 = b_{19} = 0.042044$	$b_5 = b_{19} = 0.023792$
$b_6 = b_{18} = 0.063868$	$b_6 = b_{18} = 0.077806$
$b_7 = b_{17} = 0.000000$	$b_7 = b_{17} = -0.020000$
$b_8 = b_{16} = 0.094541$	$b_8 = b_{16} = 0.017665$
$b_9 = b_{15} = -0.038728$	$b_9 = b_{15} = -0.029173$
$b_{10} = b_{14} = -0.303529$	$b_{10} = b_{14} = -0.308513$
$b_{11} = b_{13} = 0.023558$	$b_{11} = b_{13} = 0.027220$
$b_{12} = 0.400000$	$b_{12} = 0.480000$

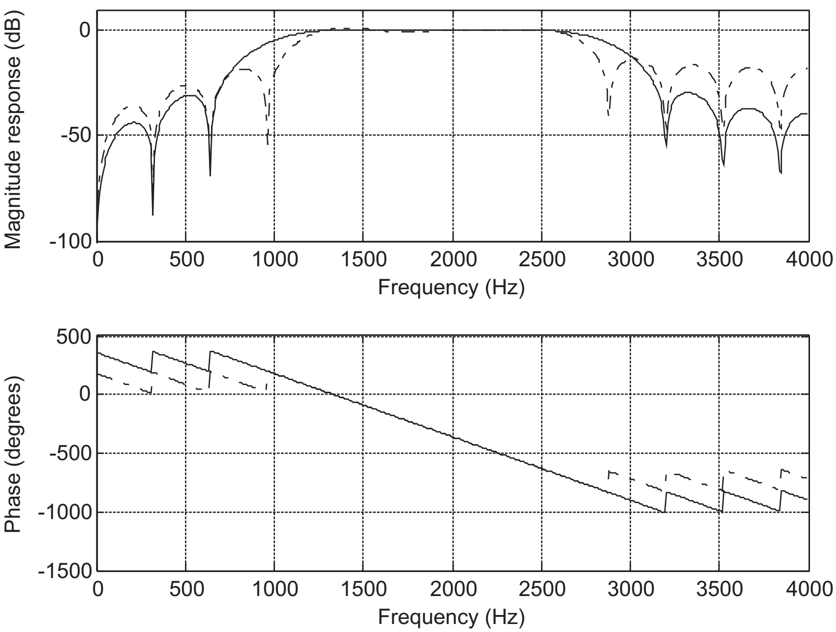

FIGURE 7.32

Frequency responses using the frequency sampling method in Example 7.14.

c. Similar to the preceding example, Figure 7.32 shows the frequency responses. Focusing on the magnitude frequency responses depicted in Figure 7.32, the dash-dotted line indicates the magnitude frequency response obtained without specifying the smooth transition band, while the solid line indicates the magnitude frequency response achieved with the specification of the smooth transition band, resulting in the reduced ripple effect.

Observations can be made from examining Examples 7.13 and 7.14. First, the oscillations (Gibbs behavior) in the passband and stopband can be reduced at the expense of increasing the width of the main lobe. Second, we can modify the specification of the magnitude frequency response with a smooth transition band to reduce the oscillations and thus improve the performance of the FIR filter. Third, the magnitude values H_k, $k = 0, 1, \cdots, M$ in general can be arbitrarily specified. This indicates that the frequency sampling method is more flexible and can be used to design the FIR filter with an arbitrary specification of the magnitude frequency response.

7.6 OPTIMAL DESIGN METHOD

This section introduces the Parks–McClellan algorithm, which is one of the most popular optimal design method used in the industry due to its efficiency and flexibility. The FIR filter design using the Parks–McClellan algorithm is developed based on the idea of minimizing the maximum approximation error between a Chebyshev polynomial and the desired filter magnitude frequency response. The details of this design development are beyond the scope of this text and can be found in Ambardar (1999) and Porat (1997). We will outline the design criteria and notation and then focus on the design procedure.

Given an ideal magnitude response $H_d(e^{j\omega T})$, the approximation error $E(\omega)$ is defined as

$$E(\omega) = W(\omega)[H(e^{j\omega T}) - H_d(e^{j\omega T})] \tag{7.32}$$

where $H(e^{j\omega T})$ is the frequency response of the linear phase FIR filter to be designed, and $W(\omega)$ is the weight function for emphasizing certain frequency bands over others during the optimization process. The goal is to minimize the error over the set of FIR coefficients:

$$\min(\max|E(\omega)|) \tag{7.33}$$

With the help of the Remez exchange algorithm, which is also beyond the scope of this book, we can obtain the best FIR filter whose magnitude response has an equiripple approximation to the ideal magnitude response. The achieved filters are optimal in the sense that the algorithms minimize the maximum error between the desired frequency response and actual frequency response. These are often called *minimax filters*.

Next, we establish the notation that will be used in the design procedure. Figure 7.33 shows the characteristics of the FIR filter designed by the Parks–McClellan and Remez exchange algorithms. As illustrated in the top graph of Figure 7.33, the passband frequency response and stopband frequency response have equiripples. δ_p is used to specify the magnitude ripple in the passband, while δ_s

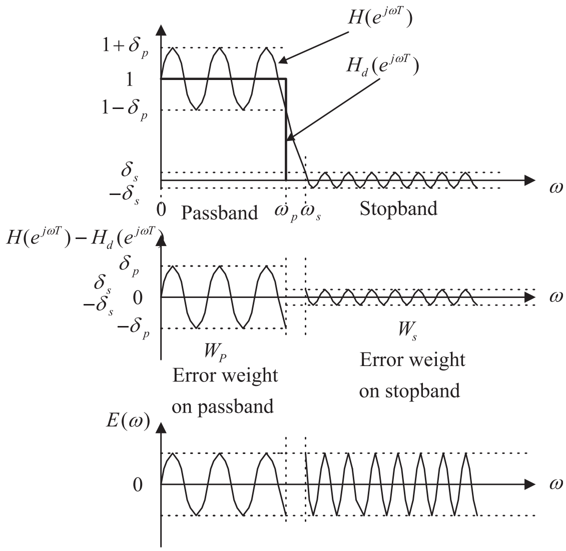

FIGURE 7.33

(Top) Magnitude frequency response of an ideal lowpass filter and a typical lowpass filter designed using the Parks–McClellan algorithm. (Middle) Error between the ideal and practical responses. (Bottom) Weighted error between the ideal and practical responses.

specifies the stopband magnitude attenuation. In terms of dB value specification, we have $\delta_p dB = 20 \times \log_{10}(1+\delta_p)$ and $\delta_s dB = 20 \times \log_{10}\delta_s$.

The middle graph in Figure 7.33 describes the error between the ideal frequency response and the actual frequency response. In general, the error magnitudes in the passband and stopband are different. This makes optimization unbalanced, since the optimization process involves an entire band. When the error magnitude in a band dominates the other(s), the optimization process may deemphasize the contribution due to a small magnitude error. To make the error magnitudes balanced, a weight function can be introduced. The idea is to weight a band with a bigger magnitude error with a small weight factor and to weight a band with a smaller magnitude error with a big weight factor. We use a weight factor W_p for weighting the passband error and W_s for weighting the stopband error. The bottom graph in Figure 7.33 shows the weighted error, and clearly, the error magnitudes on both bands are at the same level. Selection of the weighting factors is further illustrated in the following design procedure.

Optimal FIR Filter Design Procedure for the Parks–McClellan Algorithm

1. Specify the band edge frequencies such as passband and stopband frequencies, passband ripple, stopband attenuation, filter order, and sampling frequency of the DSP system.
2. Normalize band edge frequencies to the Nyquist limit (folding frequency $= f_s/2$) and specify the ideal magnitudes.

3. Calculate the absolute values of the passband ripple and stopband attenuation if they are given in terms of dB values:

$$\delta_p = 10^{\left(\frac{\delta_p dB}{20}\right)} - 1 \tag{7.34}$$

$$\delta_s = 10^{\left(\frac{\delta_s dB}{20}\right)} \tag{7.35}$$

Then calculate the ratio and put it into fraction form:

$$\frac{\delta_p}{\delta_s} = fraction\ form = \frac{numerator}{denominator} = \frac{W_s}{W_p} \tag{7.36}$$

Next, set the error weight factors for passband and stopband, respectively:

$$W_s = numerator$$

$$W_p = denominator \tag{7.37}$$

4. Apply the Remez algorithm to calculate filter coefficients.
5. If the specifications are not met, then increase the filter order and repeat steps 1 to 4.

The following two examples are given to illustrate the design procedure.

EXAMPLE 7.15

Design a lowpass filter with the following specifications:

DSP system sampling rate = 8,000 Hz
Passband = 0–800 Hz
Stopband = 1,000–4,000 Hz
Passband ripple = 1 dB
Stopband attenuation = 40 dB
Filter order = 53

Solution:

From the specifications, we have two bands: a lowpass band and a stopband. We perform normalization and specify ideal magnitudes as follows:

Folding frequency:	$f_s/2 = 8,000/2 = 4,000$ Hz	
For 0 Hz:	$0/4,000 = 0$	magnitude: 1
For 800 Hz:	$800/4,000 = 0.2$	magnitude: 1
For 1,000 Hz:	$1,000/4,000 = 0.25$	magnitude: 0
For 4,000 Hz:	$4,000/4,000 = 1$	magnitude: 0

Next, let us determine the weights:

$$\delta_p = 10^{\left(\frac{1}{20}\right)} - 1 = 0.1220$$

$$\delta_s = 10^{\left(\frac{-40}{20}\right)} = 0.01$$

Then, applying Equation (7.36) gives

$$\frac{\delta_p}{\delta_s} = 12.2 \approx \frac{12}{1} = \frac{W_s}{W_p}$$

Hence, we have

$$W_s = 12 \text{ and } W_p = 1$$

We apply the **remez()** routine provided by MATLAB in Program 7.10. The filter coefficients are listed in Table 7.14. Figure 7.34 shows the frequency responses.

Program 7.10. MATLAB program for Example 7.15.

```
%Figure 7.34 (Example 7.15)
% MATLAB program to create Figure 7.34
%
fs=8000;
f=[ 0 0.2 0.25 1]; % Edge frequencies
m=[ 1 1 0 0] ; % Ideal magnitudes
w=[ 1 12 ]; % Error weight factors
b=remez(53,f,m,w); % (53+1)Parks-McClellan algorithm and Remez exchange
format long
freqz(b,1,512,fs) % Plot the frequency response
axis([0 fs/2 -80 10]);
```

Clearly, the stopband attenuation is satisfied. We plot the details for the filter passband in Figure 7.35.

Table 7.14 FIR Filter Coefficients in Example 7.15

B: FIR Filter Coefficients (optimal design method)	
$b_0 = b_{53} = -0.006075$	$b_1 = b_{52} = -0.00197$
$b_2 = b_{51} = 0.001277$	$b_3 = b_{50} = 0.006937$
$b_4 = b_{49} = 0.013488$	$b_5 = b_{48} = 0.018457$
$b_6 = b_{47} = 0.019347$	$b_7 = b_{46} = 0.014812$
$b_8 = b_{45} = 0.005568$	$b_9 = b_{44} = -0.005438$
$b_{10} = b_{43} = -0.013893$	$b_{11} = b_{42} = -0.015887$
$b_{12} = b_{41} = -0.009723$	$b_{13} = b_{40} = 0.002789$
$b_{14} = b_{39} = 0.016564$	$b_{15} = b_{38} = 0.024947$
$b_{16} = b_{37} = 0.022523$	$b_{17} = b_{36} = 0.007886$
$b_{18} = b_{35} = -0.014825$	$b_{19} = b_{34} = -0.036522$
$b_{20} = b_{33} = -0.045964$	$b_{21} = b_{32} = -0.033866$
$b_{22} = b_{31} = 0.003120$	$b_{23} = b_{30} = 0.060244$
$b_{24} = b_{29} = 0.125252$	$b_{25} = b_{28} = 0.181826$
$b_{26} = b_{27} = 0.214670$	

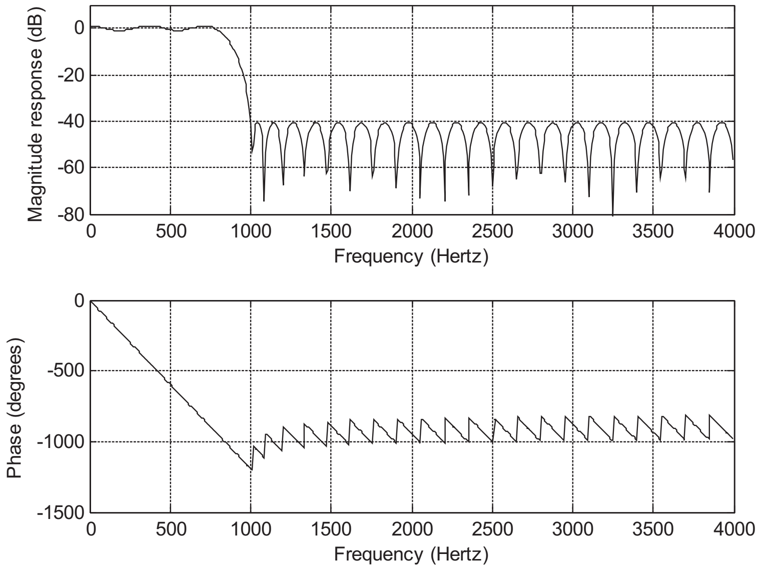

FIGURE 7.34

Frequency and phase responses for Example 7.15.

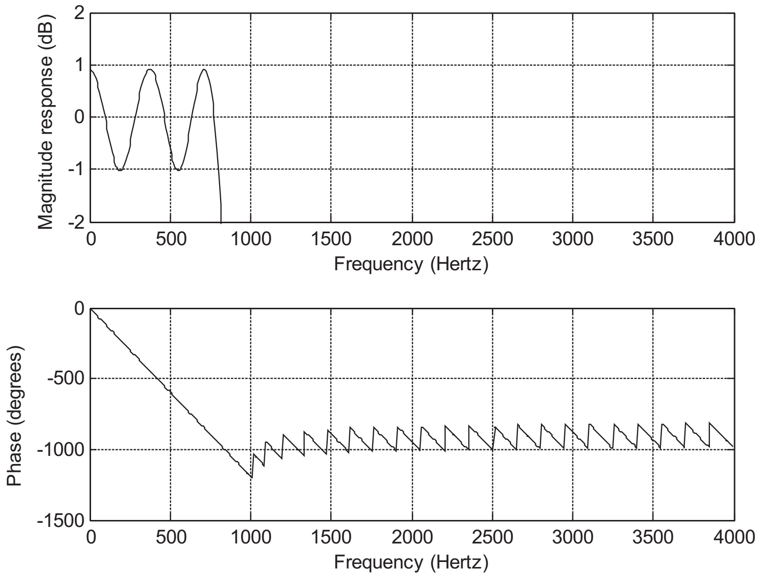

FIGURE 7.35

Frequency response details for passband in Example 7.15.

As shown in Figure 7.35, the ripples in the passband are between -1 and 1 dB. Hence, all the specifications are met. Note that if the specification is not satisfied, we will increase the order until the stopband attenuation and passband ripple are met.

The next example illustrates bandpass filter design.

EXAMPLE 7.16

Design a bandpass filter with the following specifications:

DSP system sampling rate = 8,000 Hz
Passband = 1,000–1,600 Hz
Stopband = 0–600 Hz and 2,000–4,000 Hz
Passband ripple = 1 dB
Stopband attenuation = 30 dB
Filter order = 25

Solution:
From the specifications, we have three bands: a passband, a lower stopband, and an upper stopband. We perform normalization and specify ideal magnitudes as follows:

Folding frequency:	$f_s/2 = 8{,}000/2 = 4{,}000$ Hz	
For 0 Hz:	$0/4{,}000 = 0$	magnitude: 0
For 600 Hz:	$600/4{,}000 = 0.15$	magnitude: 0
For 1,000 Hz:	$1{,}000/4{,}000 = 0.25$	magnitude: 1
For 1,600 Hz:	$1{,}600/4{,}000 = 0.4$	magnitude: 1
For 2,000 Hz:	$2{,}000/4{,}000 = 0.5$	magnitude: 0
For 4,000 Hz:	$4{,}000/4{,}000 = 1$	magnitude: 0

Next, let us determine the weights:

$$\delta_p = 10^{\left(\frac{1}{20}\right)} - 1 = 0.1220$$

$$\delta_s = 10^{\left(\frac{-30}{20}\right)} = 0.0316$$

Then applying Equation (7.36), we get

$$\frac{\delta_p}{\delta_s} = 3.86 \approx \frac{39}{10} = \frac{W_s}{W_p}$$

Hence, we have

$$W_s = 39 \text{ and } W_p = 10$$

We apply the **Remez()** routine provided by MATLAB and check performance in Program 7.11. Table 7.15 lists the filter coefficients. The frequency responses are depicted in Figure 7.36.

Program 7.11. MATLAB program for Example 7.16.

```
%Figure 7.36(Example 7.16)
% MATLAB program to create Figure 7.36
```

Table 7.15 FIR Filter Coefficients in Example 7.16

B: FIR Filter Coefficients (optimal design method)	
$b_0 = b_{25} = -0.022715$	$b_1 = b_{24} = -0.012753$
$b_2 = b_{23} = 0.005310$	$b_3 = b_{22} = 0.009627$
$b_4 = b_{21} = -0.004246$	$b_5 = b_{20} = 0.006211$
$b_6 = b_{19} = 0.057515$	$b_7 = b_{18} = 0.076593$
$b_8 = b_{17} = -0.015655$	$b_9 = b_{16} = -0.156828$
$b_{10} = b_{15} = -0.170369$	$b_{11} = b_{14} = 0.009447$
$b_{12} = b_{13} = 0.211453$	

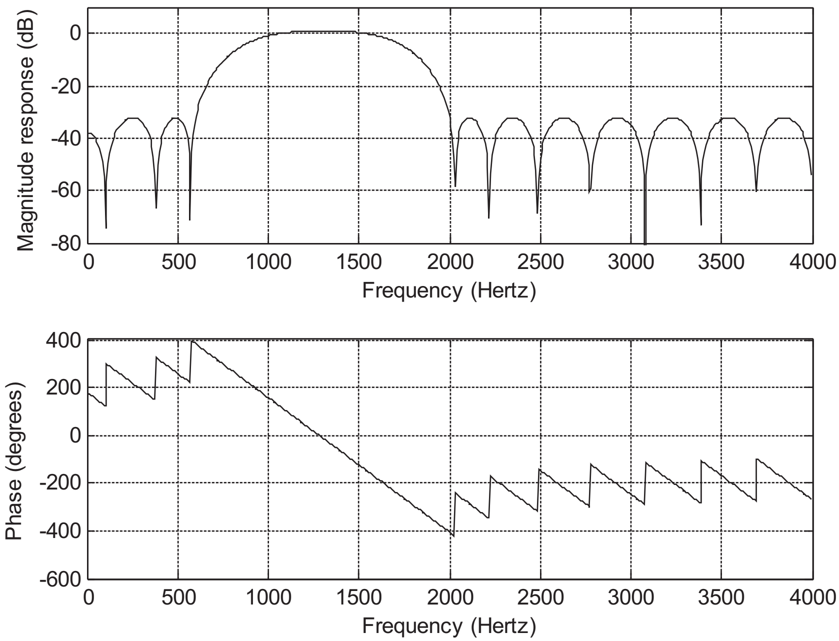

FIGURE 7.36

Frequency and phase responses for Example 7.16.

```
%
fs=8000;
f=[ 0 0.15 0.25 0.4 0.5 1];   % Edge frequencies
m=[ 0 0 1 1 0 0];             % Ideal magnitudes
w=[ 39 10 39 ];               % Error weight factors
format long
```

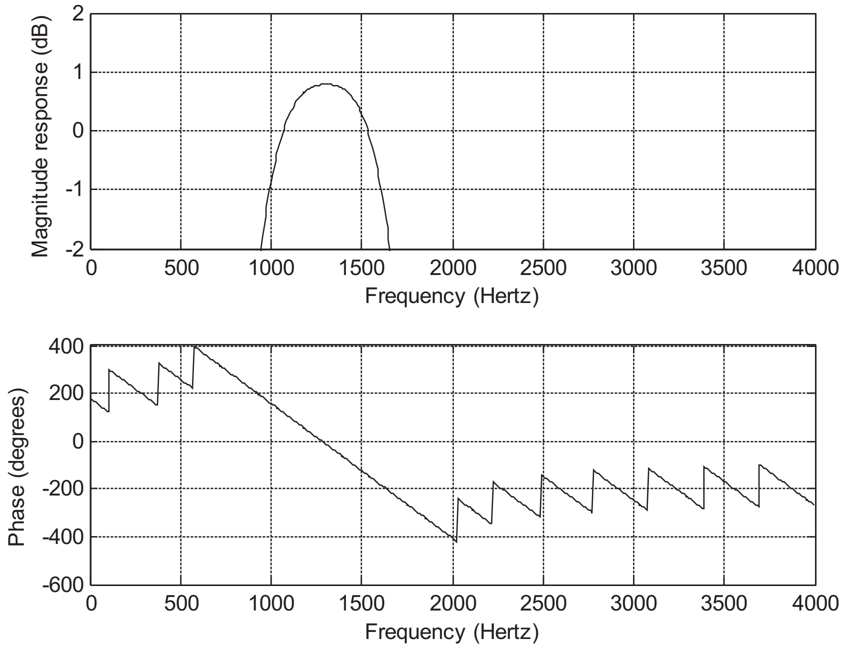

FIGURE 7.37

Frequency response details for passband in Example 7.16.

```
b=remez(25,f,m,w) % (25+1) taps Parks-McClellan algorithm and Remez exchange
freqz(b,1,512,fs); % Plot the frequency response
axis([0 fs/2 -80 10])
```

Clearly, the stopband attenuation is satisfied. We also check the details for the passband as shown in Figure 7.37.

As shown in Figure 7.37, the ripples in the passband between 1,000 Hz and 1,600 Hz are between -1 and 1 dB. Hence, all specifications are satisfied.

EXAMPLE 7.17

Now we show how the Remez exchange algorithm in Equation (7.32) is processed using a linear phase 3-tap FIR filter represented as follows:

$$H(z) = b_0 + b_1 z^{-1} + b_0 z^{-2}$$

The ideal frequency response specifications are shown in Figure 7.38(a), where the filter gain increases linearly from a gain of 0.5 at $\Omega = 0$ radians to a gain of 1 at $\Omega = \pi/4$ radians. The band between $\Omega = \pi/4$ radians and $\Omega = \pi/2$ radians is a transition band. Finally, the filter gain decreases linearly from the gain 0.75 at $\Omega = \pi/2$ radians to the gain of 0 at $\Omega = \pi$ radians.

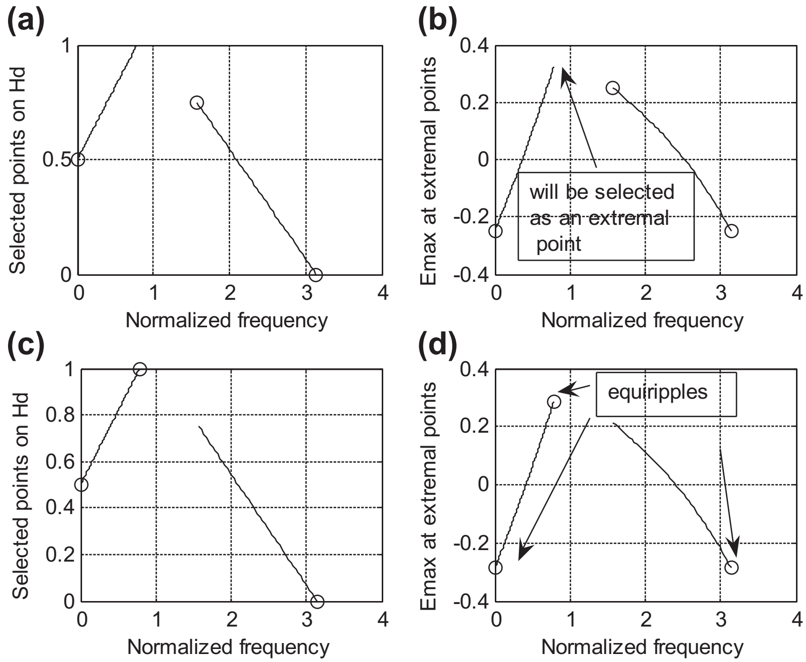

FIGURE 7.38

Determining the 3-tap FIR filter coefficients using the Remez algorithm in Example 7.17.

For simplicity, we set all the weight factors to 1, that is, $W(\Omega) = 1$. Equation (7.32) is simplified to

$$E(\Omega) = H(e^{j\Omega}) - H_d(e^{j\Omega})$$

Substituting $z = e^{j\Omega}$ into the transfer function $H(z)$ gives

$$H(e^{j\Omega}) = b_0 + b_1 e^{-j\Omega} + b_0 e^{-j2\Omega}$$

After simplification using Euler's identity $e^{j\Omega} + e^{-j\Omega} = 2\cos\Omega$, the filter frequency response is given by

$$H(e^{j\Omega}) = e^{j\Omega}(b_1 + 2b_0 \cos\Omega)$$

Disregarding the linear phase shift term $e^{j\Omega}$ for the time being, we have a Chebyshev real magnitude function (there are a few other types as well):

$$H(e^{j\Omega}) = b_1 + 2b_0 \cos\Omega$$

The *alternation theorem* (Ambardar, 1999; Porat, 1997) must be used. The alternation theorem states that given a Chebyshev polynomial $H(e^{j\Omega})$ to approximate the ideal magnitude response $H_d(e^{j\Omega})$, we can find at least $M+2$ (where $M = 1$ for our case) frequencies $\Omega_0, \Omega_1, \ldots \Omega_{M+1}$, called the extremal frequencies, so that signs of the error at the extremal frequencies alternate and the absolute error value at each extremal point reaches the maximum absolute error, that is,

$$E(\Omega_k) = -E(\Omega_{k+1}) \quad \text{for} \quad \Omega_0,\ \Omega_1, \ldots \Omega_{M+1}$$

and

$$|E(\Omega_k)| = E_{\max}$$

But the alternation theorem does not tell us how to execute the algorithm. The Remez exchange algorithm actually is employed to solve this problem. The equations and steps (Ambardar, 1999; Porat, 1997) are briefly summarized for our illustrative example:

1. Given an order of $N = 2M+1$, choose the initial extremal frequencies:
   $\Omega_0,\ \Omega_1,\ \ldots,\ \Omega_{M+1}$ (can be uniformly distributed first)
2. Solve the following equation to satisfy the alternate theorem:

$$-(-1)^k E = W(\Omega_k)(H_d(e^{j\Omega_k}) - H(e^{j\Omega_k})) \quad \text{for} \quad \Omega_0,\ \Omega_1,\ \ldots,\ \Omega_{M+1}$$

   Note that since $H(e^{j\Omega}) = b_1 + 2b_0 \cos\Omega$, for example, the solution will include solving for three unknowns: b_0,b_1, and $E_{\max}$.
3. Determine the extremal points including band edges (can be more than $M+2$ points), and retain $M+2$ extremal points with the largest error values $E_{\max}$.
4. Output the coefficients, if the extremal frequencies are not changed; otherwise, go to step 2 using the new set of extremal frequencies.

Now let us apply the Remez exchange algorithm.
First Iteration

1. We use uniformly distributed extremal points $\Omega_0 = 0$, $\Omega_1 = \pi/2$, $\Omega_2 = \pi$ whose ideal magnitudes are marked by the symbol "o" in Figure 7.38(a).
2. The alternation theorem requires $-(-1)^k E = H_d(e^{j\Omega}) - (b_1 + 2b_0 \cos\Omega)$.
   Applying extremal points yields the following three simultaneous equations with three unknowns: b_0,b_1, and E:

$$\begin{cases} -E = 0.5 - b_1 - 2b_0 \\ E = 0.75 - b_1 \\ -E = 0 - b_1 + 2b_0 \end{cases}$$

   We solve these three equations to get

$$b_0 = 0.125,\ \ b_1 = 0.5,\ E = 0.25, H(e^{j\Omega}) = 0.5 + 0.25\cos\Omega$$

3. We then determine the extremal points, including at the band edge, with their error values from Figure 7.38(b) using the following error function:

$$E(\Omega) = H_d(e^{j\Omega}) - 0.5 - 0.25\cos\Omega$$

   These extremal points are marked by symbol "o" and their error values are listed in Table 7.16.
4. Since the band edge $\Omega = \pi/4$ has an larger error than the others, it must be chosen as the extremal frequency. After deleting the extremal point at $\Omega = \pi/2$, a new set of extremal points are found according the largest error values as

$$\Omega_0 = 0$$

$$\Omega_1 = \pi/4$$

Table 7.16 Extremal Points and Band Edges with Their Error Values for the First Iteration.

Ω	0	$\pi/4$	$\pi/2$	π
E_{max}	−0.25	0.323	0.25	−0.25

Table 7.17 Error Values at Extremal Frequencies and Band Edge

Ω	0	$\pi/4$	$\pi/2$	π
E_{max}	−0.287	0.287	0.213	−0.287

$$\Omega_2 = \pi$$

The ideal magnitudes at these three extremal points are given in Figure 7.38(c), that is, 0.5, 1, 0. Now let us examine the second iteration.

Second Iteration

Applying the alternation theorem at the new set of extremal points, we have

$$\begin{cases} -E = 0.5 - b_1 - 2b_0 \\ E = 1 - b_1 - 1.4142b_0 \\ -E = 0 - b_1 + 2b_0 \end{cases}$$

Solving these three simultaneous equations leads to

$$b_0 = 0.125, \ b_1 = 0.537, \ E = 0.287, \text{ and } H(e^{j\Omega}) = 0.537 + 0.25\cos\Omega$$

The error values at the extremal points and band edge are listed in Table 7.17 and shown in Figure 7.38(d), where the determined extremal points are marked by the symbol "o".

Since the extremal points have the same maximum error value of 0.287, they are $\Omega_0 = 0, \Omega_1 = \pi/4$, and $\Omega_2 = \pi$, which is unchanged. We stop the iteration and output the filter transfer function as

$$H(z) = 0.125 + 0.537z^{-1} + 0.125z^{-2}$$

As shown in Figure 7.37(d), we obtain the equiripples of error at the extemal points $\Omega_0 = 0$, $\Omega_1 = \pi/4$, and $\Omega_2 = \pi$; their signs are alternating, and the maximum absolute error of 0.287 is obtained at each point. It takes two iterations to determine the coefficients for this simplified example.

As we mentioned, the Parks–McClellan algorithm is one of the most popular filter design methods in industry due to its flexibility and performance. However, there are two disadvantages. The filter length has to be estimated by the empirical method. Once the frequency edges, magnitudes, and weighting factors are specified, the Remez exchange algorithm cannot control the actual ripple obtained from the design. We may often need to try a longer length of filter or different weight factors to remedy situations where the ripple is unacceptable.

7.7 REALIZATION STRUCTURES OF FINITE IMPULSE RESPONSE FILTERS

Using the direct-form I realization (discussed in Chapter 6), we will obtain a special realization form, called the *transversal form*. Using the linear phase property will produce a linear phase realization structure.

7.7.1 Transversal Form

Given the transfer function of the FIR filter in Equation (7.38),

$$H(z) = b_0 + b_1 z^{-1} + \cdots + b_K z^{-K} \tag{7.38}$$

we obtain the difference equation as

$$y(n) = b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) + \cdots + b_K x(n-K)$$

Realization of such a transfer function is the transversal form, displayed in Figure 7.39.

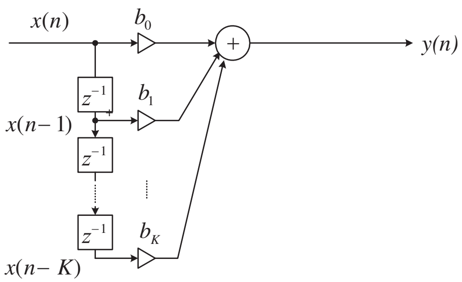

FIGURE 7.39

FIR filter realization (transversal form).

EXAMPLE 7.18

Given an FIR filter transfer function

$$H(z) = 1 + 1.2z^{-1} + 0.36z^{-2}$$

perform the FIR filter realization.

Solution:

From the transfer function, we can identify that

$$b_0 = 1,\ b_1 = 1.2, \text{ and } b_2 = 0.36$$

Using Figure 7.39, we find the FIR realization to be as displayed in Figure 7.40.

We determine the DSP equation for implementation to be

$$y(n) = x(n) + 1.2x(n-1) + 0.36x(n-2)$$

Program 7.12 shows the MATLAB implementation.

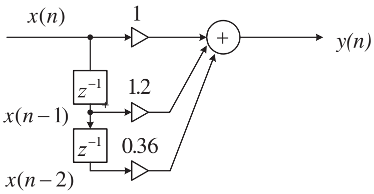

FIGURE 7.40

FIR filter realization for Example 7.18.

Program 7.12. MATLAB program for Example 7.18.

```
% Sample MATLAB code
sample =1:1:10;            % Input test array
x=[ 0 0 0];                % Input buffer [x(n) x(n-1) ...]
y=[0];                     % Output buffer [y(n) y(n-1) ... ]
b=[1.0 1.2 0.36];          % FIR filter coefficients [b0 b1 ...]
KK=length(b);
for n=1:1:length(sample)   % Loop processing
for k=KK:-1:2              % Shift input by one sample
x(k)=x(k-1);
end
x(1)=sample(n);            % Get new sample
y(1)=0;                    % Perform FIR filtering
for k=1:1:KK
y(1)=y(1)+b(k)*x(k);
end
out(n)=y(1); %send filtered sample to the output array
end
out
```

7.7.2 Linear Phase Form

We illustrate the linear phase structure using the following simple example.

Consider the following transfer function with 5 taps obtained from the design:

$$H(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + b_1 z^{-3} + b_0 z^{-4} \tag{7.39}$$

We can see that the coefficients are symmetrical and the difference equation is

$$y(n) = b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) + b_1 x(n-3) + b_0 x(n-4)$$

This DSP equation can further be combined to yield

$$y(n) = b_0(x(n) + x(n-4)) + b_1(x(n-1) + x(n-3)) + b_2 x(n-2)$$

Then we obtain the realization structure in a linear phase form as shown in Figure 7.41.

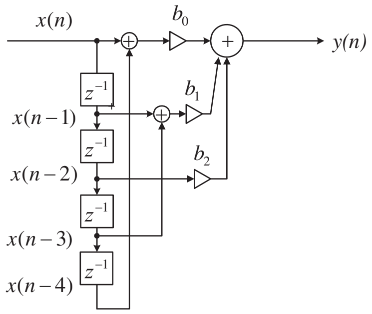

FIGURE 7.41

Linear phase FIR filter realization.

7.8 COEFFICIENT ACCURACY EFFECTS ON FINITE IMPULSE RESPONSE FILTERS

In practical applications, the filter coefficients achieved through high-level software such as MATLAB must be quantized using finite word length. This may have two effects. First, the locations of zeros are changed; second, due to the location change of zeros, the filter frequency response will change correspondingly. In practice, there are two types of digital signal (DS) processors: *fixed-point processors* and *floating-point processors*. The fixed-point DS processor uses integer arithmetic, and the floating-point processor employs floating-point arithmetic. Such effects of filter coefficient quantization will be covered in Chapter 9.

In this section, we will study the effects of FIR filter coefficient quantization in general, since during practical filter realization, obtaining filter coefficients with infinite precision is impossible. Filter coefficients are usually truncated or rounded off for the application. Assume that the FIR filter transfer function with infinite precision is given by

$$H(z) = \sum_{n=0}^{K} b_n z^{-n} = b_0 + b_1 z^{-1} + \cdots + b_{2M} z^{-K} \tag{7.40}$$

where each filter coefficient b_n has infinite precision. Now let the quantized FIR filter transfer function be

$$H^q(z) = \sum_{n=0}^{K} b_n^q z^{-n} = b_0^q + b_1^q z^{-1} + \cdots + b_K^q z^{-K} \tag{7.41}$$

where each filter coefficient b_n^q is quantized (rounded off) using the specified number of bits. Then the error of the magnitude frequency response can be bounded as

$$\begin{aligned} \left|H(e^{j\Omega}) - H^q(e^{j\Omega})\right| &= \sum_{n=0}^{K} \left|\left(b_n - b_n^q\right)e^{j\Omega}\right| \\ &< \sum_{n=0}^{K} \left|b_n - b_n^q\right| \langle (K+1)\cdot 2^{-B} \end{aligned} \tag{7.42}$$

where B is the number of bits used to encode each magnitude of the filter coefficient.

EXAMPLE 7.19

In Example 7.7, a lowpass FIR filter with 25 taps using a Hamming window was designed, and FIR filter coefficients are listed below for comparison in Table 7.18. One sign bit is used, and 7 bits are used for fractional parts, since all FIR filter coefficients are less than 1. We will multiply each filter coefficient by a scale factor of 2^7 and round off each scaled magnitude to an integer whose magnitude can be encoded using 7 bits. When the coefficient integer is scaled back, the coefficient with finite precision (quantized filter coefficient) using 8 bits, including the sign bit, will be achieved.

To understand quantization, we take a look at one of the infinite precision coefficients $Bham(3) = 0.00759455135346$, for illustration. The quantization using 7 magnitude bits is

$$0.00759455135346 \times 2^7 = 0.9721 = 1$$

Then the quantized filter coefficient is obtained as

$$BhamQ(3) = 1/2^7 = 0.0078125$$

Since the poles for both FIR filters always reside at the origin, we need to examine only their zeros. The z-plane zero plots for both FIR filters are shown in Figure 7.42A, where the circles are zeros from the FIR filter with infinite precision, while the crosses are zeros from the FIR filter with the quantized coefficients.

Most importantly, Figure 7.42B shows the difference of the frequency responses for both filters obtained using Program 7.13. In the figure, the solid line represents the frequency response with infinite filter coefficient precision, and the dot-dashed line indicates the frequency response with finite filter coefficients. It is observed that the stopband performance is degraded due to the filter coefficient quantization. The degradation in the passband is not severe.

TABLE 7.18 FIR Filter Coefficients and Their Quantized Filter Coefficients in Example 7.19 (Hamming window)

Bham: FIR Filter Coefficients	BhamQ: FIR Filter Coefficients
$b_0 = b_{24} = 0.00000000000000$	$b_0 = b_{24} = 0.0000000$
$b_1 = b_{23} = -0.00276854711076$	$b_1 = b_{23} = -0.0000000$
$b_2 = b_{22} = 0.00000000000000$	$b_2 = b_{22} = 0.0000000$
$b_3 = b_{21} = 0.00759455135346$	$b_3 = b_{21} = 0.0078125$
$b_4 = b_{20} = 0.00000000000000$	$b_4 = b_{20} = 0.0000000$
$b_5 = b_{19} = -0.01914148493949$	$b_5 = b_{19} = -0.0156250$
$b_6 = b_{18} = 0.00000000000000$	$b_6 = b_{18} = 0.0000000$
$b_7 = b_{17} = 0.04195685650042$	$b_7 = b_{17} = 0.0390625$
$b_8 = b_{16} = 0.00000000000000$	$b_8 = b_{16} = 0.0000000$
$b_9 = b_{15} = -0.09180790496577$	$b_9 = b_{15} = -0.0859375$
$b_{10} = b_{14} = 0.00000000000000$	$b_{10} = b_{14} = 0.0000000$
$b_{11} = b_{13} = 0.31332065886015$	$b_{11} = b_{13} = 0.3125000$
$b_{12} = 0.50000000000000$	$b_{12} = 0.5000000$

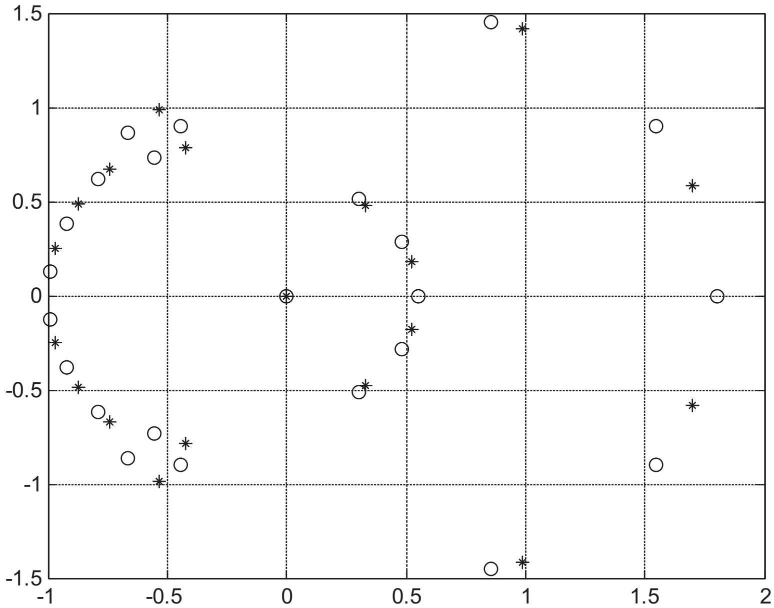

FIGURE 7.42A

The z-plane zero plots for both FIR filters. The circles are zeros for infinite precision; the crosses are zeros for round-off coefficients.

Program 7.13. MATLAB program for Example 7.19.

```
fs=8000;
[hham,f]=freqz(Bham,1,512,fs);
[hhamQ,f]=freqz(BhamQ,1,512,fs);
p=180*unwrap(angle(hham))/pi;
pQ=180*unwrap(angle(hhamQ))/pi
subplot(2,1,1); plot(f,20*log10(abs(hham)),f,20*log10(abs(hhamQ)),':');grid
axis([0 4000 -100 10]);
xlabel('Frequency (Hz)'); ylabel('Magnitude Response (dB)');
subplot(2,1,2); plot(f,p,f,pQ,':');grid
```

Using Equation (7.42), the error of the magnitude frequency response due to quantization is bounded by

$$\left|H(e^{j\Omega}) - H^{q}(e^{j\Omega})\right| < 25/256 = 0.0977$$

This can be easily verified at the stopband of the magnitude frequency response for the worst condition as follows:

$$\left|H(e^{j\Omega}) - H^{q}(e^{j\Omega})\right| = \left|10^{-100/20} - 10^{-30/20}\right| = 0.032 < 0.0977$$

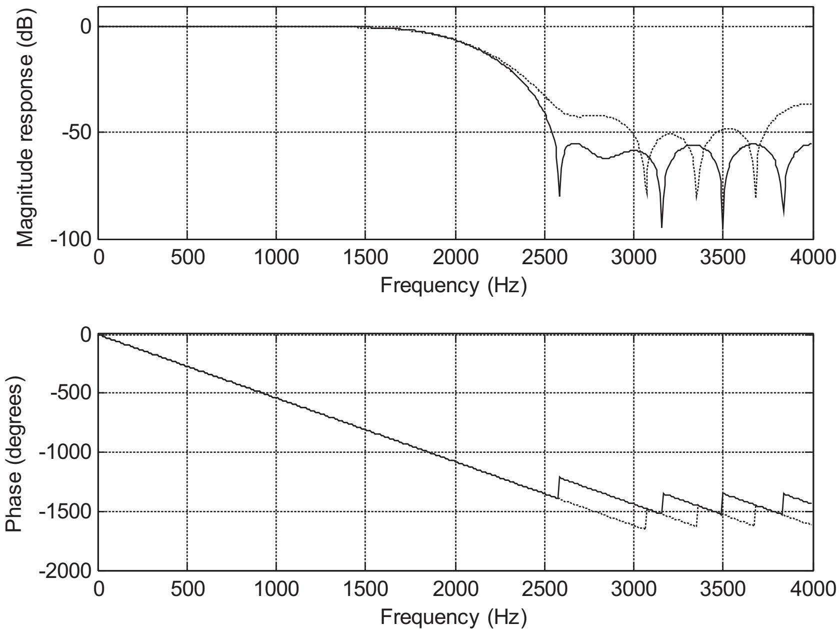

FIGURE 7.42B

Frequency responses. The solid line indicates the FIR filter with infinite precision; the dashed line indicates the FIR filter with the round-off coefficients.

In practical situations, a similar procedure can be used to analyze the effects of filter coefficient quantization to make sure that the designed filter meets the requirements.

7.9 SUMMARY OF FIR DESIGN PROCEDURES AND SELECTION OF FIR FILTER DESIGN METHODS IN PRACTICE

In this section, we first summarize the design procedures of the window design, frequency sampling design, and optimal design methods, and then discuss the selection of the particular filter for typical applications.

The window method (Fourier transform design using windows):

1. Given the filter frequency specifications, determine the filter order (odd number used in this book) and the cutoff frequency/frequencies using Table 7.7 and Equation (7.26).
2. Compute the impulse sequence $h(n)$ via the Fourier transform method using the appropriate equations (in Table 7.1).
3. Multiply the generated FIR filter coefficients $h(n)$ in step 2 by the selected window sequence using Equation (7.20) to obtain the windowed impulse sequence $h_w(n)$.

4. Delay the windowed impulse sequence $h_w(n)$ by M samples to get the causal windowed FIR filter coefficients $b_n = h_w(n-M)$ using Equation (7.21).
5. Output the transfer function and plot the frequency responses.
6. If the frequency specifications are satisfied, output the difference equation. If the frequency specifications are not satisfied, increase the filter order and repeat beginning with step 2.

The frequency sampling method:

1. Given the filter frequency specifications, choose the filter order (odd number used in the book), and specify the equally spaced magnitudes of the frequency response for the normalized frequency range from 0 to π using Equation (7.29).
2. Calculate FIR filter coefficients using Equation (7.30).
3. Use the symmetry in Equation (7.31) and the linear phase requirement to determine the rest of the coefficients.
4. Output the transfer function and plot the frequency responses.
5. If the frequency specifications are satisfied, output the difference equation. If the frequency specifications are not satisfied, increase the filter order and repeat beginning with step 2.

The optimal design method (Parks–McClellan algorithm):

1. Given the band edge frequencies, choose the filter order , normalize each band edge frequency to the Nyquist limit (folding frequency $= f_s/2$), and specify the ideal magnitudes.
2. Calculate the absolute values of the passband ripple and stopband attenuation, if they are given in terms of dB values, using Equations (7.34) and (7.35).
3. Determine the error weight factors for the passband and stopband, respectively, using Equations (7.36) and (7.37).
4. Apply the Remez algorithm to calculate filter coefficients.

Table 7.19 Comparisons of Three Design Methods

Design Method	Window	Frequency Sampling	Optimal
Filter type	1. Lowpass, highpass, bandpass, bandstop. 2. Formulas are not valid for arbitrary frequency selectivity.	1. Any type of filter 2. The formula is valid for arbitrary frequency selectivity.	1. Any type of filter 2. Valid for arbitrary frequency selectivity
Linear phase	Yes	Yes	Yes
Ripple and stopband specifications	Used for determining the filter order and cutoff frequency/-cies	Need to be checked after each design trial	Used in the algorithm; need to be checked after each design trial
Algorithm complexity for coefficients	Moderate: 1. Impulse sequence calculation 2. Window function weighting	Simple: Single equation	Complicated: 1. Parks–McClellan algorithm 2. Remez exchange algorithm
Minimal design tool	Calculator	Calculator	Software

5. Output the transfer function and check the frequency responses.
6. If the frequency specifications are satisfied, output the difference equation. If the frequency specifications are not satisfied, increase the filter order and repeat beginning with step 4.

Table 7.19 shows the comparisons for the window, frequency sampling, and optimal methods. The table can be used as a selection guide for each design method in this book.

Example 7.20 describes the possible selection of the design method by a DSP engineer to solve a real-world problem.

EXAMPLE 7.20

Determine the appropriate FIR filter design method for each of the following DSP applications.

a. A DSP engineer implements a digital two-band crossover system as described in Section 7.4.4 in this book. He selects the FIR filters to satisfy the following specifications:

Sampling rate = 44,100 Hz
Crossover frequency = 1,000 Hz (cutoff frequency)
Transition band = 600 Hz to 1,400 Hz
Lowpass filter = passband frequency range from 0 to 600 Hz with a ripple of 0.02 dB and stopband edge at 1,400 Hz with an attenuation of 50 dB.
Highpass filter = passband frequency range from 1.4 to 44.1 kHz with a ripple of 0.02 dB and stopband edge at 600 Hz with an attenuation of 50 dB.

The engineer does not have the software routine for the Remez algorithm.

b. An audio engineer tries to equalize a speech signal sampled at 8,000 Hz using a linear phase FIR filter based on the magnitude specifications in Figure 7.43. The engineer does not have the software routine for the Remez algorithm.

Solution:

a. The window design method is the first choice, since this formula is expressed in terms of the cutoff frequency (crossover frequency), the filter order is based on the transient band, and the filter types are standard lowpass and highpass. The ripple and stopband specifications can be satisfied by selecting the Hamming window. The optimal design method will also do the job if the **remez()** algorithm is available. But there exists a challenge to satisfy the combined unity gains at the crossover frequency of 1,000 Hz.

b. Since the magnitude frequency response is not a standard filter type such as lowpass, highpass, bandpass, or bandstop, and the **remez()** algorithm is not available, the first choice should be the frequency sampling method.

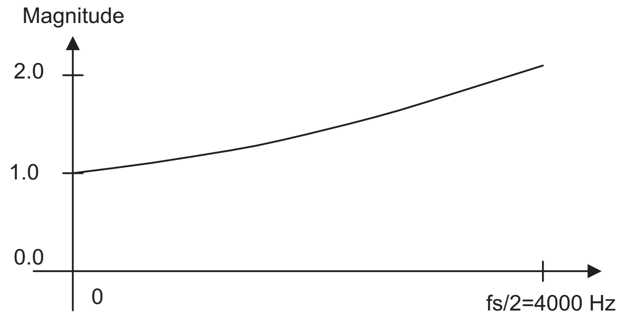

FIGURE 7.43

Magnitude frequency response in Example 7.20(b).

7.10 SUMMARY

1. The Fourier transform method is used to compute noncausal FIR filter coefficients, including those of lowpass, highpass, bandpass, and bandstop filters.
2. Converting noncausal FIR filter coefficients to causal FIR filter coefficients only introduces linear phase, which is a good property for audio applications. The linear phase filter output has the same amount of delay for all the input signals whose frequency components are within the passband.
3. The causal FIR filter using the Fourier transform method generates ripple oscillations (Gibbs effect) in the passband and stopband in its filter magnitude frequency response due to abrupt truncation of the FIR filter coefficient sequence.
4. To reduce the oscillation effect, the window method is introduced to tap down the coefficient values towards both ends. A substantial improvement of the magnitude frequency response is achieved.
5. Real-life DSP applications such as noise reduction systems and two-band digital audio crossover systems were investigated.
6. Frequency sampling design is feasible for an FIR filter with an arbitrary magnitude response specification.
7. An optimal design method, the Parks–McClellan algorithm using the Remez exchange algorithm, offers flexibility for filter specifications. The Remez exchange algorithm was explained using a simplified example.
8. Realization structures of FIR filters have special forms, such as the transversal form and the linear phase form.
9. The effect of quantizing FIR filter coefficients for implementation changes the zero locations of the FIR filter. More effects on the stopband in the magnitude and phase responses are observed.
10. Guidelines for selecting an appropriate design method in practice were summarized with consideration of the filter type, linear phase, ripple and stopband specifications, algorithm complexity, and design tools.

7.11 MATLAB PROGRAMS

Program 7.14 enables one to design FIR filters via the window method using window functions such as the rectangular window, triangular window, Hanning window, Hamming window, and Blackman window. Filter types of the design include lowpass, highpass, bandpass, and bandstop.

Program 7.14. MATLAB function for FIR filter design using the window method.

```
function B=firwd(N,Ftype,WnL,WnH,Wtype)
% B = firwd(N,Ftype,WnL,WnH,Wtype)
% FIR filter design using the window function method.
% Input parameters:
% N: the number of the FIR filter taps.
%  Note: It must be odd number.
```

```
% Ftype: the filter type
%      1. Lowpass filter
%      2. Highpass filter
%      3. Bandpass filter
%      4. Bandstop filter
% WnL: lower cutoff frequency in radians. Set WnL=0 for the highpass filter.
% WnH: upper cutoff frequency in radians. Set WnL=0 for the lowpass filter.
% Wtype: window function type
%      1. Rectangular window
%      2. Triangular window
%      3. Hanning window
%      4. Hamming window
%      5. Balckman window
% Output:
% B: FIR filter coefficients.
       M=(N-1)/2;
       hH=sin(WnH*[-M:1:-1])./([-M:1:-1]*pi);
       hH(M+1)=WnH/pi;
       hH(M+2:1:N)=hH(M:-1:1);
       hL=sin(WnL*[-M:1:-1])./([-M:1:-1]*pi);
       hL(M+1)=WnL/pi;
       hL(M+2:1:N)=hL(M:-1:1);
       if Ftype == 1
        h(1:N)=hL(1:N);
       end
       if Ftype == 2
        h(1:N)=-hH(1:N);
        h(M+1)=1+h(M+1);
end
if Ftype ==3
  h(1:N)=hH(1:N)-hL(1:N);
end
if Ftype == 4
  h(1:N)=hL(1:N)-hH(1:N);
  h(M+1)=1+h(M+1);
end
% Window functions
      if Wtype ==1
      w(1:N)=ones(1,N);
  end
  if Wtype ==2
  w=1-abs([-M:1:M])/M;
  end
  if Wtype ==3
  w= 0.5+0.5*cos([-M:1:M]*pi/M);
  end
  if Wtype ==4
  w=0.54+0.46*cos([-M:1:M]*pi/M);
```

```
end
if Wtype ==5
w=0.42+0.5*cos([-M:1:M]*pi/M)+0.08*cos(2*[-M:1:M]*pi/M);
end
B=h .* w
```

Program 7.15 enables one to design FIR filters using the frequency sampling method. Note that values of the frequency response, which correspond to the equally spaced DFT frequency components, must be specified for design. Besides the lowpass, highpass, bandpass and bandstop filter designs, the method can be used to design FIR filters with an arbitrarily specified magnitude frequency response.

Program 7.15. MATLAB function for FIR filter design using the frequency sampling method.

```
function B=firfs(N,Hk)
% B=firfs(N,Hk)
% FIR filter design using the frequency sampling method.
% Input parameters:
% N: the number of filter coefficients.
%   note: N must be odd number.
% Hk: sampled frequency reponse for k=0,1,2,...,M=(N-1)/2.
% Output:
% B: FIR filter coefficients.
      M=(N-1)/2;
      for n=1:1:N
      B(n)=(1/N)*(Hk(1)+...
      2*sum(Hk(2:1:M+1)...
      .*cos(2*pi*([1:1:M])*(n-1-M)/N)));
end
```

7.12 PROBLEMS

7.1. Design a 3-tap FIR lowpass filter with a cutoff frequency of 1,500 Hz and a sampling rate of 8,000 Hz using a

a. rectangular window function

b. Hamming window function

Determine the transfer function and difference equation of the designed FIR system, and compute and plot the magnitude frequency response for $\Omega = 0, \pi/4, \pi/2, 3\pi/4,$ and π radians.

7.2. Design a 3-tap FIR highpass filter with a cutoff frequency of 1,600 Hz and a sampling rate of 8,000 Hz using a

a. rectangular window function

b. Hamming window function

Determine the transfer function and difference equation of the designed FIR system, and compute and plot the magnitude frequency response for $\Omega = 0, \pi/4, \pi/2, 3\pi/4,$ and π radians.

7.3. Design a 5-tap FIR lowpass filter with a cutoff frequency of 100 Hz and a sampling rate of 1,000 Hz using a

a. rectangular window function

b. Hamming window function

Determine the transfer function and difference equation of the designed FIR system, and compute and plot the magnitude frequency response for $\Omega = 0, \pi/4, \pi/2, 3\pi/4,$ and π radians.

7.4. Design a 5-tap FIR highpass filter with a cutoff frequency of 250 Hz and a sampling rate of 1,000 Hz using a

a. rectangular window function

b. Hamming window function

Determine the transfer function and difference equation of the designed FIR system, and compute and plot the magnitude frequency response for $\Omega = 0, \pi/4, \pi/2, 3\pi/4,$ and π radians.

7.5. Design a 5-tap FIR bandpass filter with a lower cutoff frequency of 1,600 Hz, an upper cut-off frequency of 1,800 Hz and a sampling rate of 8,000 Hz using a

a. rectangular window function

b. Hamming window function

Determine the transfer function and difference equation of the designed FIR system, and compute and plot the magnitude frequency response for $\Omega = 0, \pi/4, \pi/2, 3\pi/4,$ and π radians.

7.6. Design a 5-tap FIR band reject filter with a lower cutoff frequency of 1,600 Hz, an upper cutoff frequency of 1,800 Hz, and a sampling rate of 8,000 Hz using a

a. rectangular window function

b. Hamming window function

Determine the transfer function and difference equation of the designed FIR system, and compute and plot the magnitude frequency response for $\Omega = 0, \pi/4, \pi/2, 3\pi/4,$ and π radians.

7.7. Consider an FIR lowpass filter design with the following specifications:

Passband = 0–800 Hz

Stopband = 1,200–4,000 Hz

Passband ripple = 0.1 dB

Stopband attenuation = 40 dB

Sampling rate = 8,000 Hz

Determine the following:

a. window method

b. length of the FIR filter

c. cutoff frequency for the design equation

7.8. Consider an FIR highpass filter design with the following specifications:

Stopband = 0–1,500 Hz

Passband = 2,000–4,000 Hz

Passband ripple = 0.02 dB

Stopband attenuation = 60 dB

Sampling rate = 8,000 Hz

Determine the following:

a. window method

b. length of the FIR filter

c. cutoff frequency for the design equation

7.9. Consider an FIR bandpass filter design with the following specifications:

Lower cutoff frequency = 1,500 Hz

Lower transition width = 600 Hz

Upper cutoff frequency = 2,300 Hz

Upper transition width = 600 Hz

Passband ripple = 0.1 dB

Stopband attenuation = 50 dB

Sampling rate: 8,000 Hz

Determine the following:

a. window method

b. length of the FIR filter

c. cutoff frequencies for the design equation

7.10. Consider an FIR bandstop filter design with the following specifications:

Lower passband = 0–1,200 Hz

Stopband = 1,600–2,000 Hz

Upper passband = 2,400–4,000 Hz

Passband ripple = 0.05 dB

Stopband attenuation = 60 dB

Sampling rate = 8,000 Hz

Determine the following:

a. window method

b. length of the FIR filter

c. cutoff frequencies for the design equation

7.11. Given an FIR system

$$H(z) = 0.25 - 0.5z^{-1} + 0.25z^{-2}$$

realize $H(z)$ using each of the following specified methods:

a. transversal form (write the difference equation for implementation)

b. linear phase form (write the difference equation for implementation)

7.12. Given an FIR filter transfer function

$$H(z) = 0.2 + 0.5z^{-1} - 0.3z^{-2} + 0.5z^{-3} + 0.2z^{-4}$$

perform the linear phase FIR filter realization, and write the difference equation for implementation.

7.13. Determine the transfer function for a 3-tap FIR lowpass filter with a cutoff frequency of 150 Hz and a sampling rate of 1,000 Hz using the frequency sampling method.

7.14. Determine the transfer function for a 3-tap FIR highpass filter with a cutoff frequency of 250 Hz and a sampling rate of 1,000 Hz using the frequency sampling method.

7.15. Determine the transfer function for a 5-tap FIR lowpass filter with a cutoff frequency of 2,000 Hz and a sampling rate of 8,000 Hz using the frequency sampling method.

7.16. Determine the transfer function for a 5-tap FIR highpass filter with a cutoff frequency of 3,000 Hz and a sampling rate of 8,000 Hz using the frequency sampling method.

7.17. Given the following specifications, determine the transfer function:

- 7-tap FIR bandpass filter
- lower cutoff frequency of 1,500 Hz and upper cutoff frequency of 3,000 Hz
- sampling rate of 8,000 Hz
- frequency sampling design method

7.18. Given the following specifications, determine the transfer function:

- 7-tap FIR bandstop filter
- lower cutoff frequency of 1,500 Hz and upper cutoff frequency of 3,000 Hz
- sampling rate of 8,000 Hz
- frequency sampling design method

7.19. A lowpass FIR filter to be designed has the following specifications:

Design method: Parks–McClellan algorithm

Sampling rate = 1,000 Hz

Passband = 0–200 Hz

Stopband = 300–500 Hz

Passband ripple = 1 dB

Stopband attenuation = 40 dB

Determine the error weights W_p and W_s for the passband and stopband in the Parks–McClellan algorithm.

7.20. A bandpass FIR filter to be designed has the following specifications:

Design method: Parks–McClellan algorithm

Sampling rate = 1,000 Hz

Passband = 200–250 Hz

Lower stopband = 0–150 Hz

Upper stopband = 300–500 Hz

Passband ripple = 1 dB

Stopband attenuation = 30 dB

Determine the error weights W_p and W_s for the passband and stopband in the Parks–McClellan algorithm.

7.21. A highpass FIR filter to be designed has the following specifications:

Design method: Parks–McClellan algorithm

Sampling rate = 1,000 Hz

Passband = 350–500 Hz

Stopband = 0–250 Hz

Passband ripple = 1 dB

Stopband attenuation = 60 dB

Determine the error weights W_p and W_s for the passband and stopband in the Parks–McClellan algorithm.

7.22. A bandstop FIR filter to be designed has the following specifications:

Design method: Parks–McClellan algorithm

Sampling rate = 1,000 Hz

Stopband = 250−350 Hz

Lower passband = 0−200 Hz

Upper passband = 400−500 Hz

Passband ripple = 1 dB

Stopband attenuation = 25 dB

Determine the error weights W_p and W_s for the passband and stopband in the Parks−McClellan algorithm.

7.23. In a speech recording system with a sampling rate of 10,000 Hz, the speech is corrupted by broadband random noise. To remove the random noise while preserve speech information, the following specifications are given:

Speech frequency range = 0−3,000 Hz

Stopband range = 4,000−5,000 Hz

Passband ripple = 0.1 dB

Stopband attenuation = 45 dB

FIR filter with Hamming window

Determine the FIR filter length (number of taps) and the cutoff frequency; use MATLAB to design the filter; and plot the frequency response.

7.24. Consider the speech equalizer shown in Figure 7.44 to compensate for midrange frequency loss of hearing that has the following specifications:

Sampling rate = 8,000 Hz

Bandpass FIR filter with Hamming window

Frequency range to be emphasized = 1,500−2,000 Hz

Lower stopband = 0−1,000 Hz

Upper stopband = 2,500−4,000 Hz

Passband ripple = 0.1 dB

Stopband attenuation = 45 dB

Determine the filter length and the lower and upper cutoff frequencies.

FIGURE 7.44

Speech equalizer in Problem 7.24.

7.25. A digital crossover can be designed as shown in Figure 7.45.

Consider the following audio specifications:

Sampling rate = 44,100 Hz

Crossover frequency = 2,000 Hz

Transition band range = 1,600 Hz

Passband ripple = 0.1 dB

Stopband attenuation = 50 dB

Filter type = FIR

Determine the following for each filter in Figure 7.45:

a. window function

b. filter length

c. cutoff frequency

Use MALAB to design both filters and plot frequency responses for both filters.

7.12.1 Computer Problems with MATLAB

Use the MATLAB programs provided in Section 7.11 to design the following FIR filters.

7.26. Design a 41-tap lowpass FIR filter whose cutoff frequency is 1,600 Hz using the following window functions. Assume that the sampling frequency is 8,000 Hz.

a. rectangular window function

b. triangular window function

c. Hanning window function

d. Hamming window function

e. Blackman window function

List the FIR filter coefficients and plot the frequency responses for each case.

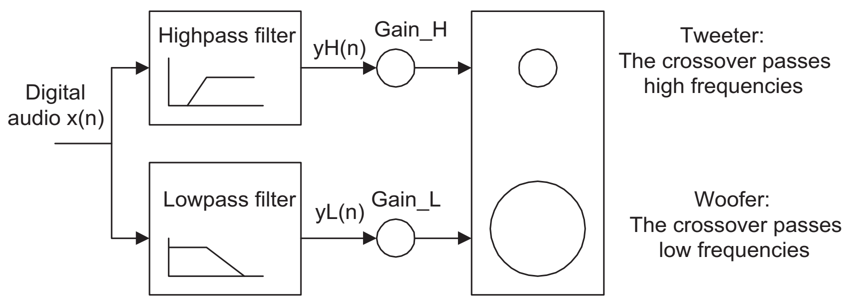

FIGURE 7.45

Two-band crossover in Problem 7.25.

7.27. Design a lowpass FIR filter whose cutoff frequency is 1,000 Hz using the Hamming window function for the following specified filter length. Assume that the sampling frequency is 8,000 Hz.

a. 21 filter coefficients

b. 31 filter coefficients

c. 41 filter coefficients

List the FIR filter coefficients for each design and compare the magnitude frequency responses.

7.28. Design a 31-tap highpass FIR filter whose cutoff frequency is 2,500 Hz using the following window functions. Assume that the sampling frequency is 8,000 Hz.

a. Hanning window function

b. Hamming window function

c. Blackman window function

List the FIR filter coefficients and plot the frequency responses for each design.

7.29. Design a 41-tap bandpass FIR filter with lower and upper cutoff frequencies of 2,500 Hz and 3,000 Hz, respectively, using the following window functions. Assume a sampling frequency of 8,000 Hz.

a. Hanning window function

b. Blackman window function.

List the FIR filter coefficients and plot the frequency responses for each design.

7.30. Design a 41-tap band reject FIR filter with cutoff frequencies of 2,500 Hz and 3,000 Hz, respectively, using the Hamming window function. Assume a sampling frequency of 8,000 Hz. List the FIR filter coefficients and plot the frequency responses.

7.31. Use the frequency sampling method to design a linear phase lowpass FIR filter with 17 coefficients. Let the cutoff frequency be 2,000 Hz and assume a sampling frequency of 8,000 Hz. List the FIR filter coefficients and plot the frequency responses.

7.32. Use the frequency sampling method to design a linear phase bandpass FIR filter with 21 coefficients. Let the lower and upper cutoff frequencies be 2,000 Hz and 2,500 Hz, respectively, and assume a sampling frequency of 8,000 Hz. List the FIR filter coefficients and plot the frequency responses.

7.33. Given an input data sequence

$$x(n) = 1.2 \cdot \sin(2\pi(1,000)n/8,000) - 1.5 \cdot \cos(2\pi(2,800)n/8,000)$$

with a sampling frequency of 8,000 Hz, use the designed FIR filter with a Hamming window in Problem 7.26 to filter 400 data points of $x(n)$, and plot the 400 samples of the input and output data.

7.34. Design a lowpass FIR filter with the following specifications:

Design method: Parks−McClellan algorithm

Sampling rate = 8,000 Hz

Passband = 0−1,200 Hz

Stopband = 1,500−4,000 Hz

Passband ripple = 1 dB

Stopband attenuation = 40 dB

List the filter coefficients and plot the frequency responses.

7.35. Design a bandpass FIR filter with the following specifications:

Design method: Parks−McClellan algorithm

Sampling rate = 8,000 Hz

Passband = 1,200−1,600 Hz

Lower stopband = 0−800 Hz

Upper stopband = 2,000−4,000 Hz

Passband ripple = 1 dB

Stopband attenuation = 40 dB

List the filter coefficients and plot the frequency responses.

7.12.2 MATLAB Projects

7.36. Speech enhancement:

Digitally recorded speech in a noisy environment can be enhanced using a lowpass filter if the recorded speech with a sampling rate of 8,000 Hz contains the desired frequency components lower than 1,600 Hz. Design a lowpass filter to remove the high frequency noise above 1,600 Hz with the following filter specifications: passband frequency range: 0 − 1,600 Hz; passband ripple: 0.02 dB; stopband frequency range: 1,800−4,000 Hz; stopband attenuation: 50 dB.

Use the designed lowpass filter to filter the noisy speech and adopt the following code to simulate the noisy speech:

```
    load speech.dat
t=[0:length(speech)-1]*T;
th=mean(speech.*speech)/4; %Noise power =(1/4) speech power
noise=sqrt(th)*randn([1,length(speech)]); %Generate Gaussian noise
nspeech=speech+noise; % Generate noisy speech
```

In this project, plot the speech samples and spectra for both noisy speech and the enhanced speech and use the MATLAB **sound()** function to evaluate the sound quality. For example, to hear the noisy speech, use the following:

$$\text{sound}(\text{nspeech}/\max(\text{abs}(\text{nspeech})), 8000);$$

7.37. Digital crossover system:

Design a two-band digital crossover system with the following specifications:

Sampling rate = 44,100 Hz

Crossover frequency = 1,200 Hz (cutoff frequency)

Transition band = 800−1,600 Hz

Lowpass filter: passband frequency range from 0 to 800 Hz with a ripple of 0.02 dB and stopband edge at 1,600 Hz with the attenuation of 50 dB

Highpass filter: passband frequency range from 1.6 to 44.1 kHz with a ripple of 0.02 dB and stopband edge at 800 Hz with the attenuation of 50 dB

In this project, plot the magnitude frequency responses for both lowpass and highpass filters. Use the following MATLAB code to read stereo audio data ("No9seg.wav"):

$$[\text{x fs Nbits}] = \text{wavread}('\text{No9seg.wav}');$$

Process the given stereo audio segment. Listen to and describe the sound effects of the processed audio in the following sequences:

Channel 1: original, lowband, and highband

Channel 2: original, lowband, and highband

Stereo (both channels): original, lowband, and highband