1.1 Basic Concepts of Digital Signal Processing

Digital signal processing (DSP) technology and its advancements have dramatically impacted our modern society everywhere. Without DSP, we would not have digital/Internet audio or video; digital recording; CD, DVD, and MP3 players; digital cameras; digital and cellular telephones; digital satellite and TV; or wire and wireless networks. Medical instruments would be less efficient or unable to provide useful information for precise diagnoses if there were no digital electrocardiography (ECG) analyzers or digital x-rays and medical image systems. We would also live in many less efficient ways, since we would not be equipped with voice recognition systems, speech synthesis systems, and image and video editing systems. Without DSP, scientists, engineers, and technologists would have no powerful tools to analyze and visualize data and perform their design, and so on.

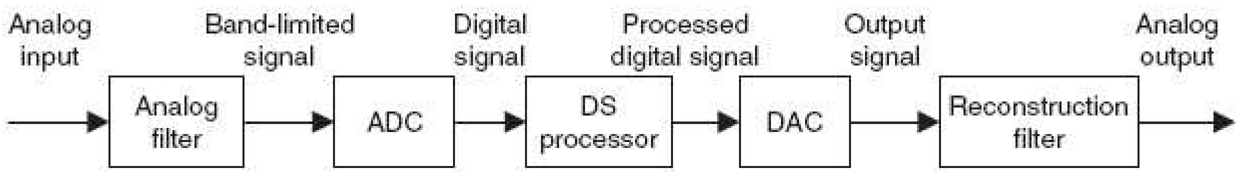


FIGURE 1.1 A digital signal processing scheme.

The concept of DSP is illustrated by the simplified block diagram in Fig. (1.1), which consists of an analog filter, an analog-to-digital conversion (ADC) unit, a digital signal (DS) processor, a digital-to-analog conversion (DAC) unit, and a reconstruction (anti-image) filter.

As shown in the diagram, the analog input signal, which is continuous in time and amplitude, is generally encountered in our real life. Examples of such analog signals include current, voltage, temperature, pressure, and light intensity.

Usually a transducer (sensor) is used to convert the non-electrical signal to the analog electrical signal (voltage). This analog signal is fed to an analog filter, which is applied to limit the frequency range of analog signals prior to the sampling process. The purpose of filtering is to significantly attenuate aliasing distortion.

The band-limited signal at the output of the analog filter is then sampled and converted via the ADC unit into the digital signal, which is discrete both in time and in amplitude.

The DS processor then accepts the digital signal and processes the digital data according to DSP rules such as lowpass, highpass, and bandpass digital filtering, or other algorithms for different applications. Notice that the DS processor unit is a special type of digital computer and can be a general-purpose digital computer, a microprocessor, or an advanced microcontroller; furthermore, DSP rules can be implemented using software in general.

With the DS processor and corresponding software, a processed digital output signal is generated. This signal behaves in a manner according to the specific algorithm used

The DAC unit converts the processed digital signal to an analog output signal. The signal is continuous in time and discrete in amplitude (usually a sample-and-hold signal). The final block in Fig. (1.1) is designated as a function to smooth the DAC output voltage levels back to the analog signal via a reconstruction (anti-image) filter for real-world applications.

1.2 Basic Digital Signal Processing

1.2.1 Digital Filtering

Consider the situation shown in Fig. (1.2), of a digitized noisy signal containing a useful low-frequency signal and noise that occupies all of the frequency range. After ADC, the digitized noisy signal x(n), where n is the sample number, can be enhanced using digital filtering. Since our useful signal contains the low-frequency component, the high frequency components above that of our useful signal are considered as noise, which can be removed by using a digital lowpass filter.

After processing the digitized noisy signal x(n), the digital lowpass filter produces a clean digital signal y(n). The cleaned signal y(n) is applied to another DSP algorithm for a different application or convert it to the analog signal via DAC and the reconstruction filter.

The digitized noisy signal and clean digital signal, respectively, are plotted in Fig. (1.3), where the top plot shows the digitized noisy signal, x(n), while the bottom plot demonstrates the clean digital signal ,y(n), obtained by applying the digital lowpass filter.

FIGURE 1.2 **The simple digital filtering block.**

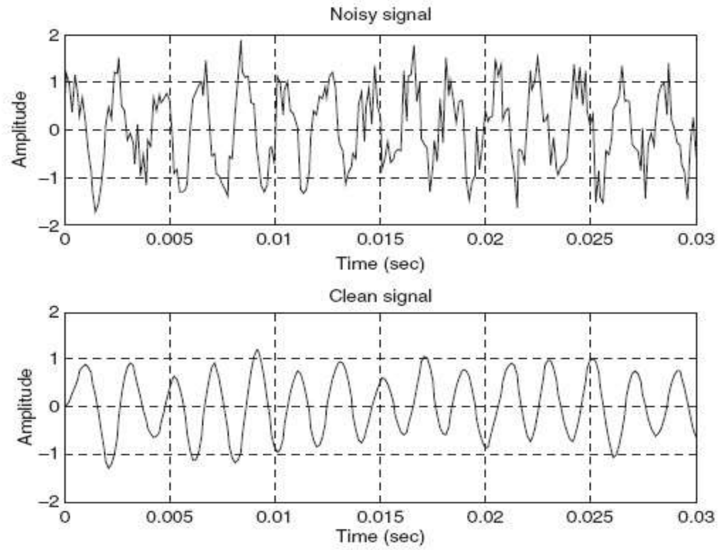


FIGURE 1.3 (*Top*) Digitized noisy signal. (*Bottom*) Clean digital signal using the digital lowpass filter.

1.2.2 Signal Frequency (Spectrum) Analysis

As shown in Figure 1.4, certain DSP applications often require that time domain information and the frequency content of the signal be analyzed.

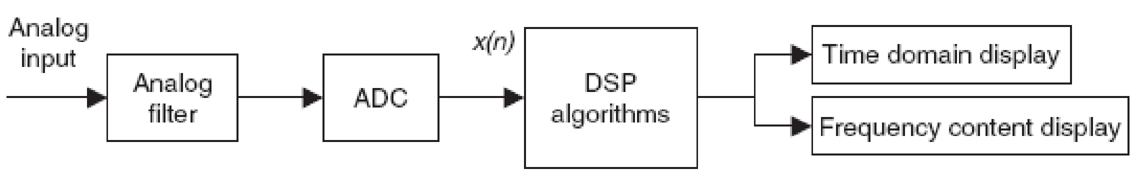


FIGURE 1.4 **Signal spectral analysis.**

Figure 1.5 shows a digitized audio signal and its calculated signal spectrum (frequency content), defined as the signal amplitude versus its corresponding frequency. It is also called fast Fourier transform (FFT).

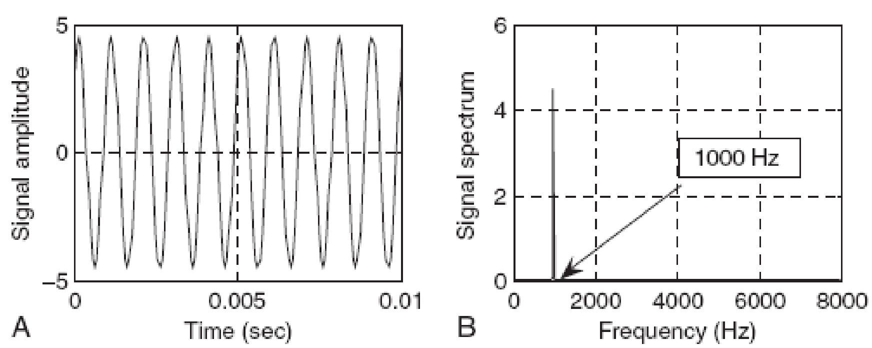


Figure 1.5 Audio signal and its spectrum

The plot in Figure 1.5 (a) is a time domain display of the recorded audio signal with frequency of 1,000 Hz sampled at 16,000 samples per second, while the frequency content display of plot (b) displays the calculated signal spectrum versus frequencies, in which the peak amplitude is clearly located at 1,000 Hz.

As another practical example, we often perform spectral estimation of a digitally recorded speech or audio (music) waveform using the FFT algorithm in order to investigate spectral frequency details of speech information. Figure 1.6 shows a speech signal produced by a human in the time domain and frequency content displays. The top plot shows the digital speech waveform versus its digitized sample number, while the bottom plot shows the frequency content information of speech for a range from 0 to 4,000 Hz. We can observe that there are about ten spectral peaks, called speech formants, in the range between 0 and 1,500 Hz. Those identified speech formants can be used for applications such as speech modeling, speech coding, and speech feature extraction for speech synthesis and recognition,

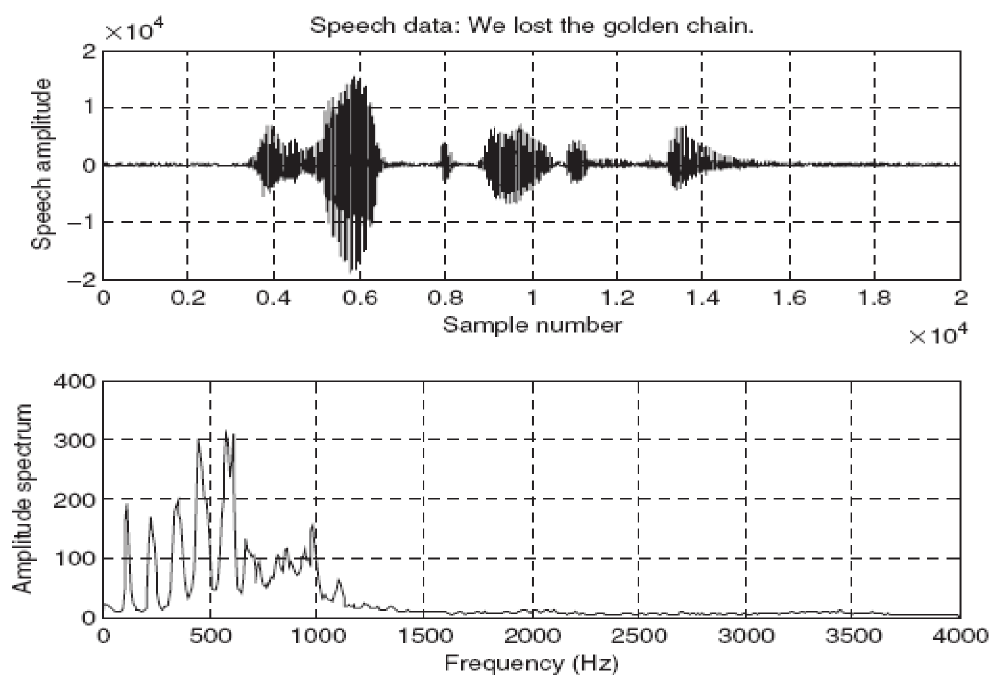


Figure 1.6 Speech sample and speech spectrum

1.3 Digital Signal Processing Applications

The list below by no means covers all DSP applications. Many more areas are increasingly being explored by engineers and scientists. Applications of DSP techniques will continue to have profound impacts and improve our lives.

1. ***Digital audio and speech***: Digital audio coding such as CD players, digital crossover, digital audio equalizers, digital stereo and surround sound, noise reduction systems, speech coding, data compression and encryption, speech synthesis and speech recognition.
2. ***Digital telephone***: Speech recognition, high-speed modems, echo cancellation, speech synthesizers, DTMF (dual-tone multi frequency) generation and detection, answering machines.
3. ***Automobile industry***: Active noise control systems, active suspension systems, digital audio and radio, digital controls.
4. ***Electronic communications***: Cellular phones, digital telecommunications, wireless LAN (local area networking), satellite communications.
5. ***Medical imaging equipment***: ECG analyzers, cardiac monitoring, medical imaging and image recognition, digital x-rays and image processing.
6. ***Multimedia***: Internet phones, audio, and video; hard disk drive electronics; digital pictures;

   digital cameras; text-to-voice and voice-to-text technologies

2.1 Sampling of Continuous Signal

Figure 2.1 shows an analog (continuous-time) signal (solid line) defined at every point over the time axis and amplitude axis. Hence, the analog signal contains an infinite number of points.

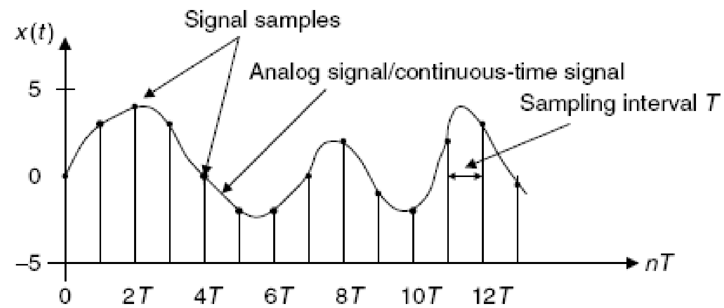


Figure 2.1 Display of the analog (continuous) signal and display of digital samples versus the sampling time instants.

It is impossible to digitize an infinite number of points. Furthermore, the infinite points are not appropriate to be processed by the digital signal (DS) processor or computer, since they require an infinite amount of memory and infinite amount of processing power for computations. Sampling can solve such a problem by taking samples at the fixed time interval, as shown in Figure 2.1 and Figure 2.2, where the time T represents the sampling interval or sampling period in seconds.

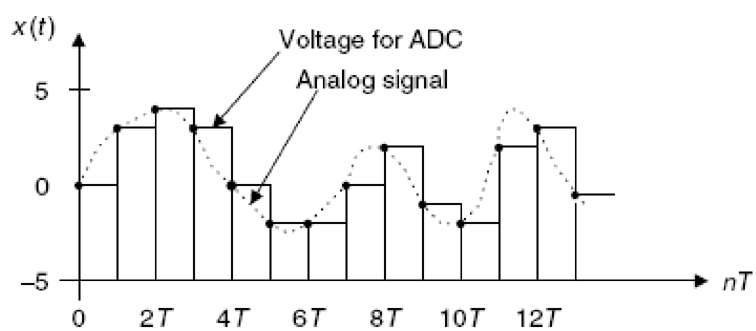


Figure 2.2 Sample-and-hold analog voltage for ADC.

As shown in Figure 2.2, each sample maintains its voltage level during the sampling interval T to give the ADC enough time to convert it. This process is called sample and hold.

For a given sampling interval T, which is defined as the time span between two sample points, the sampling rate is therefore given by:

$$f_s = \frac{1}{T_s} \qquad (2.1) \qquad \text{Samples per second (Hz)}$$

After the analog signal is sampled, we obtain the sampled signal whose amplitude values are taken at the sampling instants, thus the processor is able to handle the sample points. Next, we have to ensure that samples are collected at a rate high enough that the original analog signal can be reconstructed or recovered later.

In other words, we are looking for a minimum sampling rate to acquire a complete reconstruction of the analog signal from its sampled version.

If an analog signal is not appropriately sampled, aliasing will occur, which causes unwanted signals in the desired frequency band.

The sampling theorem guarantees that an analog signal can be in theory perfectly recovered as long as the sampling rate is at least twice as large as the highest-frequency component of the analog signal to be sampled. The condition is described as:

$$f_s \geq 2\, f_{max} \qquad (2.2)$$

Where, f_{max} is the maximum-frequency component of the analog signal to be sampled. For example, to sample a speech signal containing frequencies up to 4 kHz, the minimum sampling rate is chosen to be at least 8 kHz, or 8,000 samples per second; to sample an audio signal possessing frequencies up to 20 kHz, at least 40,000 samples per second, or 40 kHz, of the audio signal are required.

Figure 2.3 depicts the sampled signal $x_s(t)$ obtained by sampling the continuous signal x(t) at a sampling rate of f_s samples per second.

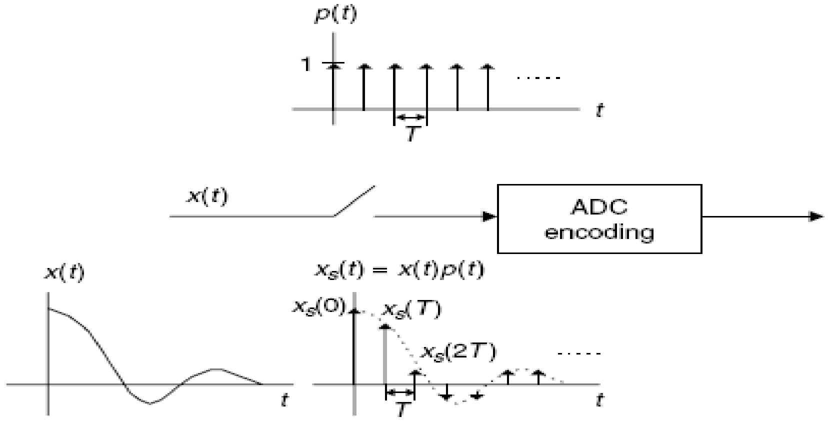


Figure 2.3 The simplified sampling process

Mathematically, this process can be written as the product of the continuous signal and the sampling pulses (pulse train):

$x_s(t) = x(t)\, p(t)$ (2.3)

Where, p(t) is the pulse train with a period $T = 1/ f_s$.

From the spectral analysis shown in Fig. 2.4, it is clear that the sampled signal spectrum consists of the scaled baseband spectrum centered at the origin and its replicas centered at the frequencies of $\pm nf_s$ (multiples of the sampling rate) for each of n = 1,2,3, ...In Figure 2.4, three possible sketches are classified. Given the original signal spectrum X(f) plotted in Figure 2.4(a), the sampled signal spectrum is plotted in Figure 2.4(b), where, the replicas have separations between them. In Fig. 2.4(c), the baseband spectrum and its replicas are just connected. In Fig. 2.4(d), the original spectrum and its replicas are overlapped; that is, there are many overlapping portions in the sampled signal spectrum.

If applying a lowpass reconstruction filter to obtain exact reconstruction of the original signal spectrum, equation (2.2) must be satisfied. This fundamental conclusion is well known as the **Shannon sampling theorem**, which is formally described below:

For a uniformly sampled DSP system, an analog signal can be perfectly recovered as long as the sampling rate is at least twice as large as the highest-frequency component of the analog signal to be sampled.

We summarize two key points here.

1. Sampling theorem establishes a minimum sampling rate for a given bandlimited analog signal with the highest-frequency component f_{max}. If the sampling rate satisfies equation (2.2), then the analog signal can be recovered via its sampled values using the lowpass filter, as described in Fig. 2.4(b).

2. Half of the sampling frequency (f_s / 2) is usually called the Nyquist frequency (Nyquist limit), or folding frequency. The sampling theorem indicates that a DSP system with a sampling rate of f_s can ideally sample an analog signal with its highest frequency up to half of the sampling rate without introducing spectral overlap (aliasing). Hence, the analog signal

can be perfectly recovered from its sampled version as described in Fig. 2.4 (c). Fig. 2.4(d) shows aliasing.

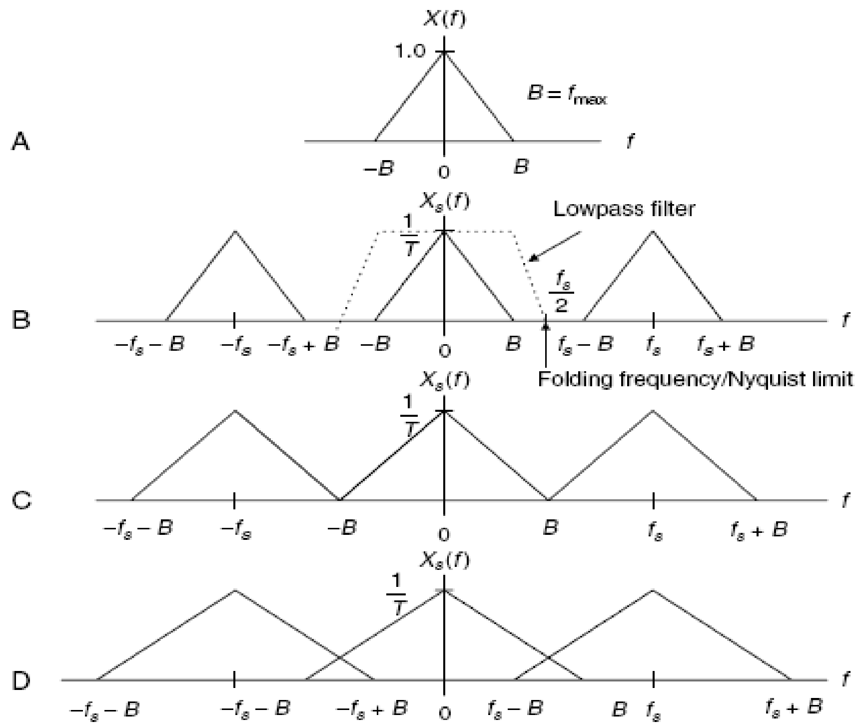


Fig. 2.4 plots of the sampled signal spectrum.

Example(1)

Suppose that an analog signal is given as

$$x(t) = 5\cos(2\pi \cdot 1000t), \text{ for } t \geq 0$$

and is sampled at the rate of 8,000 Hz.

Sketch the spectrum for the original signal. .a

Sketch the spectrum for the sampled signal from 0 to 20 kHz. .b

Solution:

$$5\cos(2\pi \times 1000t) = 5 \cdot \left(\frac{e^{j2\pi\times 1000t} + e^{-j2\pi\times 1000t}}{2}\right) = 2.5e^{j2\pi\times 1000t} + 2.5e^{-j2\pi\times 1000t},$$

The two-sided spectrum is plotted as shown in Fig. 2.5 (a). After the analog signal is sampled at the rate of 8,000 Hz, the sampled signal spectrum and its replicas centered at the frequencies ± nf_s, each with the scaled amplitude being 2.5/T, are as shown in Fig. 2.5(b)

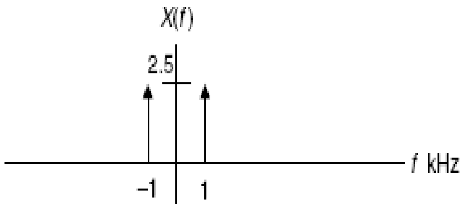


Fig. 2.5 (a)

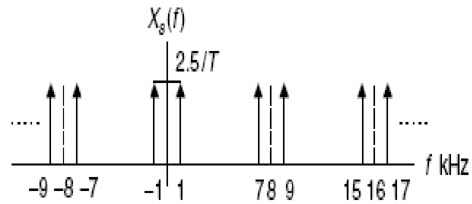


Fig. 2.5(b)

Notice that the spectrum of the sampled signal shown in Figure 2.5(b) contains the images of the original spectrum shown in Figure 2.5(a); that the images repeat at multiples of the sampling frequency f_s (for our example, 8 kHz, 16 kHz, 24 kHz, . . .); and that all images must be removed, since they convey no additional information.

2.2 Signal Reconstruction

Two simplified steps are involved, as described in Figure 2.6. First, the digitally processed data y(n) are converted to the ideal impulse train $y_s(t)$, in which each impulse has its amplitude proportional to digital output y(n), and two consecutive impulses are separated by a sampling period of T; second, the analog reconstruction filter is applied to the ideally recovered sampled signal $y_s(t)$ to obtain the recovered analog signal.

The following three cases are listed for recovery of the original signal spectrum:

Case 1: $f_s = 2f_{\max}$ Nyquist frequency is equal to the maximum frequency of the analog signal x(t), an ideal lowpass reconstruction filter is required to recover the analog signal spectrum. This is an impractical case.

Case 2: $f_s > 2f_{\max}$ In this case, there is a separation between the highest-frequency edge of the baseband spectrum and the lower edge of the first replica. Therefore, a practical lowpass reconstruction (anti-image) filter can be designed to reject all the images and achieve the original signal spectrum.

Case 3: $f_s < 2f_{\max}$ This is aliasing, where the recovered baseband spectrum suffers spectral distortion, that is, contains an aliasing noise spectrum; in time domain, the recovered analog signal may consist of the aliasing noise frequency or frequencies. Hence, the recovered analog signal is incurably distorted.

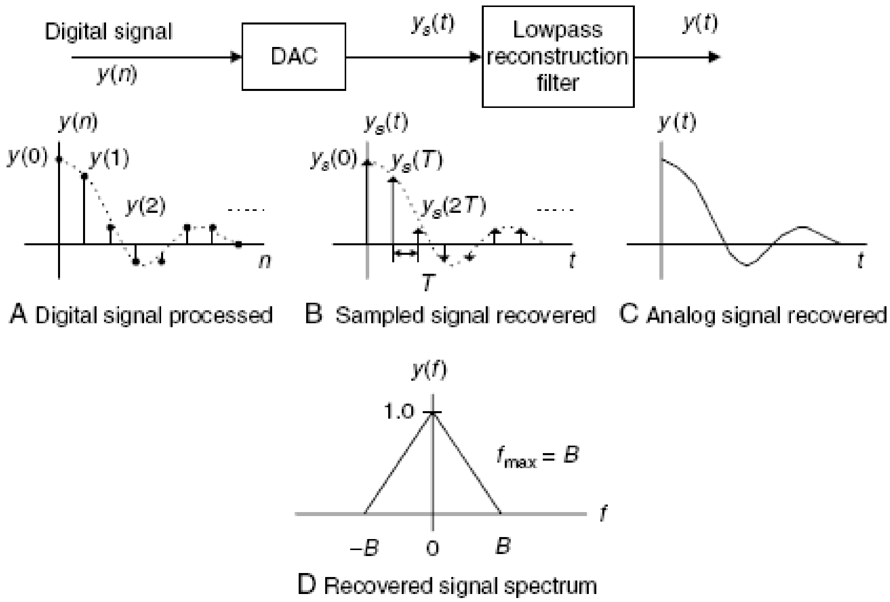


Fig. 2.6 Signal notations at reconstruction stage.

Example(2)

Assuming that an analog signal is given by

$$x(t) = 5\cos(2\pi \cdot 2000t) + 3\cos(2\pi \cdot 3000t), \text{ for } t \geq 0$$

and it is sampled at the rate of 8,000 Hz,

a. Sketch the spectrum of the sampled signal up to 20 kHz.
b. Sketch the recovered analog signal spectrum if an ideal lowpass filter with a cutoff frequency of 4 kHz is used to filter the sampled signal (y(n) = x(n) in this case) to recover the original signal.

Solution: Using Euler's identity, we get

$$x(t) = \frac{3}{2}e^{-j2\pi \cdot 3000t} + \frac{5}{2}e^{-j2\pi \cdot 2000t} + \frac{5}{2}e^{j2\pi \cdot 2000t} + \frac{3}{2}e^{j2\pi \cdot 3000t}.$$

The two-sided amplitude spectrum for the sinusoids is displayed in Figure 2.7 (a). The recovered spectrum is shown in Fig. 2.7 (b)

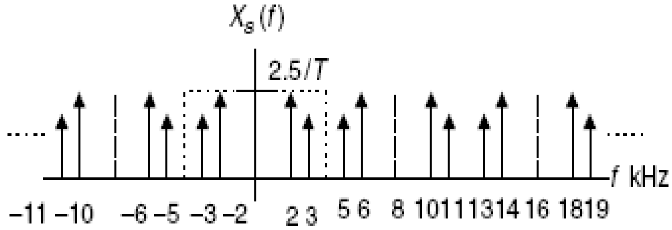


Fig. 2.7 (a)

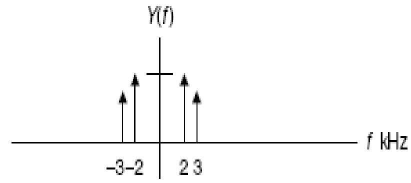


Fig. 2.7 (b)

2.2.3 Aliasing noise level

Given the DSP system shown in Fig. (2.8), where we can find the percentage of the aliasing noise level using the symmetry of the *Butterworth magnitude function* and its first replica. Then:-

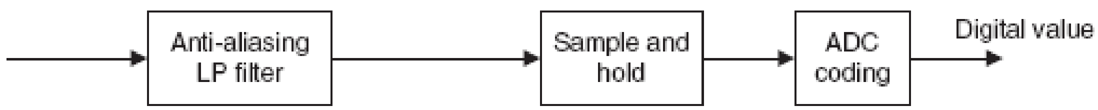


Fig. 2.8 DSP system with anti-aliasing filter

Aliasing noise level % = $$\frac{\sqrt{1+(f_a/f_c)^{2n}}}{\sqrt{1+\left(\frac{f_s-f_a}{f_c}\right)^{2n}}} \qquad 0 \le f \le f_c \qquad (2.4)$$

Where, n is the filter order, f_a is the aliasing frequency, f_c is the cutoff frequency, and f_s is the sampling frequency.

Example (3)

In a DSP system with anti-aliasing filter, if a sampling rate of 8,000 Hz is used and the anti-aliasing filter is a second-order Butterworth lowpass filter with a cutoff frequency of 3.4 kHz,

a. Determine the percentage of aliasing level at the cutoff frequency.

b. Determine the percentage of aliasing level at the frequency of 1,000 Hz.

Solution:

$$f_s = 8000,\ f_c = 3400, \text{ and } n = 2.$$

a. Since $f_a = f_c = 3400$ Hz, we compute

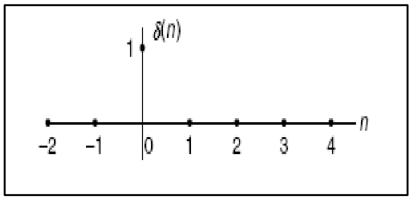
$$\text{aliasing noise level }\% = \frac{\sqrt{1+\left(\frac{3.4}{3.4}\right)^{2\times 2}}}{\sqrt{1+\left(\frac{8-3.4}{3.4}\right)^{2\times 2}}} = \frac{1.4142}{2.0858} = 67.8\%.$$

b. With $f_a = 1000$ Hz, we have

$$\text{aliasing noise level }\% = \frac{\sqrt{1+\left(\frac{1}{3.4}\right)^{2\times 2}}}{\sqrt{1+\left(\frac{8-1}{3.4}\right)^{2\times 2}}} = \frac{1.03007}{4.3551} = 23.05\%.$$

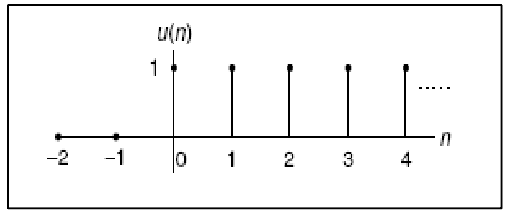
Lec. 3 Digital Signals and Systems

3.1 Digital Signals



1- Digital unit-impulse function

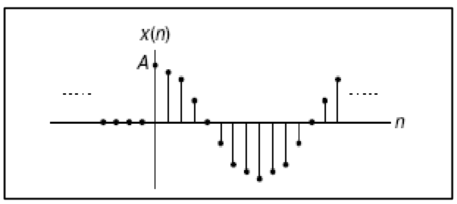
$$\delta(n) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$



2- Digital unit-step function

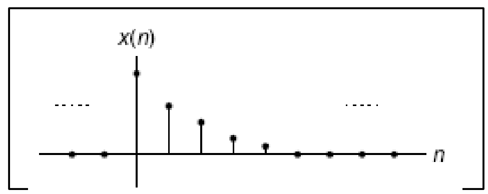
$$u(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$

3- Sinusoidal sequence



$x(n) = \cos wn$, $0 \leq n \leq \infty$

4- Exponential sequence



$x(n) = e^{-jwn}$, $0 \leq n \leq \infty$

Fig. (3.1) Some digital signals

3.2 Generation of Digital Signals

To develop the digital sequence from its analog signal function is by applying:

$$x(n) = x(t)|_{t=nT} = x(nT). \quad (3.1)$$

Example(1): assuming a DSP system with a sampling time interval of 125 microseconds, Convert each of the following analog signals x(t) to the digital signal x(n).

1. $x(t) = 10e^{-5000t}u(t)$
2. $x(t) = 10\sin(2000\pi t)u(t)$

Solution:

1. $x(n) = x(nT) = 10e^{-5000\times 0.000125n}u(nT) = 10e^{-0.625n}u(n).$
2. $x(n) = x(nT) = 10\sin(2000\pi \times 0.000125n)u(nT) = 10\sin(0.25\pi n)u(n).$

3.3 Power Signals:

Periodic signals are power signals because their energy per cycle is finite.

$$power = \frac{1}{T}\int_0^T |f(t)|^2 \, dt = \sum_{n=-\infty}^{\infty} |C_n|^2 = \varphi(\tau) \qquad (3.2)$$

Where:

$$C_n = \frac{1}{T}\int_0^T f(t)\, e^{-jnw_o t}\, dt \qquad , w_o = 2\pi f_o \qquad (3.3)$$

$$f(t) = \sum_{n=-\infty}^{\infty} C_n \, e^{jnw_o t} \qquad (3.4)$$

$$\varphi(\tau) = \frac{1}{T}\int_0^T f(t)\, f(t \pm \tau)\, dt \qquad (3.5)$$

3.4 Energy Signals:

Non-periodic signals are called an energy signals because their power → 0

$$energy = \int_{-\infty}^{\infty} |f(t)|^2 \, dt = \frac{1}{2\pi}\int_{-\infty}^{\infty} |F(W)|^2 \, dW = \lambda(\tau) \qquad (3.6)$$

Where:

$$F(W) = \int_{-\infty}^{\infty} f(t)\, e^{-jwt}\, dt \qquad (3.7)$$

$$\lambda(\tau) = \int_{-\infty}^{\infty} f(t)\, f(t \pm \tau)\, dt \qquad (3.8)$$

3.5 Classification of Systems

3.5.1 Linear System

Figure 3.2 illustrates that the system output due to the weighted sum inputs $\alpha\, x_1(n) \pm \beta\, x_2(n)$ is equal to the same weighted sum of the individual outputs obtained from their corresponding inputs, that is, $y(n) = \alpha\, y_1(n) \pm \beta\, y_2(n)$, where α and β are constants. Here, the principle of "superposition" is applied.

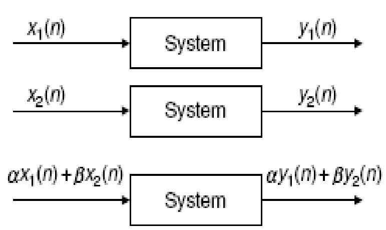


Fig. (3.2) Digital linear system

3.5.2 Time-Invariant System

A time-invariant system is illustrated in Figure 3.3. If the system is time invariant and $y_1(n)$ is the system output due to the input $x_1(n)$, then the shifted system input $x_1(n - n_0)$ will produce a shifted system output $y_1(n - n_0)$.

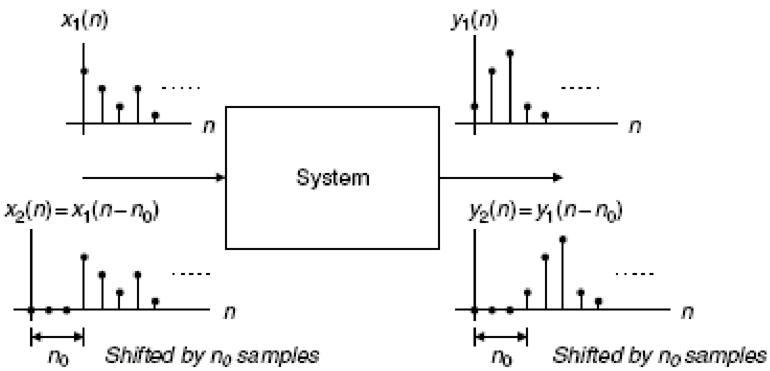


Fig. 3.3 Illustration of linear time-invariant system

Example 2: Given the linear systems:

a. $y(n) = 2x(n - 5)$

b. $y(n) = 2x(3n)$,

Determine whether each of the following systems is time invariant.

Solution:

Let the input and output be $x_1(n)$ and $y_1(n)$, respectively; then the system output is $y_1(n) =$ (a $2x_1(n - 5)$. Again, let $x_2(n) = x_1(n - n_0)$ be the shifted input and $y_2(n)$ be the output due to the shifted input. We determine the system output using the shifted input as

$$y_2(n) = 2x_2(n - 5) = 2x_1(n - n_0 - 5):$$

Meanwhile, shifting $y_1(n) = 2x_1(n - 5)$ by n_0 samples leads to

$$y_1(n - n_0) = 2x_1(n - 5 - n_0)$$

We can verify that $y_2(n) = y_1(n - n_0)$. Thus the shifted input of n_0 samples causes the system output to be shifted by the same n_0 samples, thus the system is *time invariant.*

Let the input and output be $x_1(n)$ and $y_1(n)$, respectively; then the system output is $y_1(n)$ (b $=2x_1(3n)$. Again, let the input and output be $x_2(n)$ and $y_2(n)$, where $x_2(n) = x_1(n - n_0)$, a shifted version, and the corresponding output is $y_2(n)$. We get the output due to the shifted input $x_2(n) = x_1(n - n_0)$ and note that $x_2(3n) = x_1(3n - n_0)$:

$$y_2(n) = 2x_2(3n) = 2x_1(3n - n_0):$$

On the other hand, if we shift $y_1(n)$ by n_0 samples, which replaces n in

$y_1(n) = 2x_1(3n)$ by $n - n_0$, it yield

$y_1(n - n_0) = 2x_1(3(n - n_0)) = 2x_1(3n - 3n_0)$:

Clearly, we know that $y_2(n) \neq y_1(n - n_0)$. Since the system output $y_2(n)$ using the input shifted by n_0 samples is not equal to the system output $y_1(n)$ shifted by the same n_0 samples, the system is *not time invariant.*

3.5.3 Causal System:

A causal system is one in which the output y(n) at time n depends only on the current input x(n) at time n, its past input sample values such as x(n − 1), x(n− 2), . . . : Otherwise, if a system output depends on the future input values, such as x(n + 1), x(n + 2), . . . , the system is noncausal. The noncausal system cannot be realized in real time.

Example 3: Given the following linear systems,

a. $y(n) = 0.5x(n) + 2.5x(n - 2)$, for $n \geq 0$

b. $y(n) = 0.25x(n - 1) + 0.5x(n + 1) - 0.4y(n - 1)$, for $n \geq 0$,

Determine whether each is causal.

Solution:

Since for $n \geq 0$, the output y(n) depends on the current input x(n) and its past value x(n − 2), (a the system is causal.

Since for $n \geq 0$, the output y(n) depends on the current input x(n) and its future value x(n + 2), (b the system is noncausal.

3.5.4. Stability:

A stable system is one for which every bounded input produces a bounded output (BIBO). The system is stable, if its transfer function vanishes after a sufficiently long time. For a stable system:

$$S = \sum_{k=-\infty}^{\infty} | h(k) | \quad \langle \infty \qquad (3.9)$$

Where *h(k)* = unit impulse response

3.6 Difference Equations and Impulse Responses

A causal, linear, time-invariant system can be described by a difference equation having the following general form:

$$y(n) + a_1 y(n-1) + \ldots + a_N y(n-N)$$

$$= b_0 x(n) + b_1 x(n-1) + \ldots + b_M x(n-M), \qquad (3.10)$$

Where $a_1, \ldots, a_N$ and $b_0, b_1, \ldots, b_M$ are the coefficients of the difference equation. Equation (3.10) can further be written as:

$$y(n) = -a_1 y(n-1) - \ldots - a_N y(n-N) + b_0 x(n) + b_1 x(n-1) + \ldots + b_M x(n-M)$$

$$y(n) = -\sum_{i=1}^{N} a_i y(n-i) + \sum_{j=0}^{M} b_j x(n-j). \quad (3.11)$$

Notice that y(n) is the current output, which depends on the past output samples y(n − 1), . . . , y(n − N), the current input sample x(n), and the past input samples, x(n−1), . . . , x(n − M).

Example4: Given a linear system described by the difference equation y(n) = x(n) + 0.5x(n − 1), Determine the nonzero system coefficients.

Solution: a. By comparing Equation (3.11), we have, $b_0 = 1$, and $b_1 = 0.5$

3.7 System Representation Using Its Impulse Response

A linear time-invariant system can be completely described by its unit-impulse response, which is defined as the system response due to the impulse input δ(n) with zero initial and y(n) = h(n).conditions, depicted in Figure 3.3. Here x(n) = δ(n)

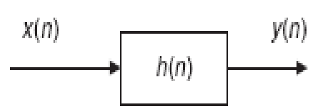


Fig. 3.4 Representation of a linear time-invariant system using the impulse response.

Example 5: Given the linear time-invariant system y(n) = 0.5x(n) + 0.25x(n − 1) with an initial condition x(−1) = 0

Determine the unit-impulse response h(n). .a

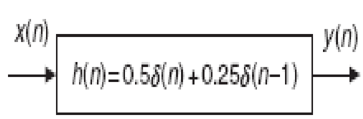
Draw the system block diagram. .b

Write the output using the obtained impulse response. .c

Solution:

a. h(n) = 0.5 δ(n) + 0.25 δ(n − 1) , where h(0)= 0.5, h(1) = 0.25 and h(n) = 0 elsewhere.

b.



c. $y(n) = h(0)\,x(n) + h(1)\,x(n-1)$

From this result, it is noted that if the difference equation without the past output terms, $y(n-1)$, $\ldots$, $y(n-N)$, that is, the corresponding coefficients $a_1, \ldots, a_N$, are zeros, the impulse response $h(n)$ has a finite number of terms. We call this a finite impulse response (FIR) system.

In general, we can express the output sequence of a linear time-invariant system from its impulse response and inputs as:

$$y(n) = \ldots + h(-1)\,x(n+1) + h(0)\,x(n) + h(1)\,x(n-1) + h(2)\,x(n-2) + \ldots \quad (3.12)$$

Equation (3.12) is called the ***digital convolution sum.***

Example 6: Given the difference equation

$$y(n) = 0.25\,y(n-1) + x(n) \text{ for } n \ge 0 \text{ and } y(-1) = 0,$$

Determine the unit-impulse response h(n). .a

Draw the system block diagram. .b

Write the output using the obtained impulse response. .c

For a step input $x(n) = u(n)$, verify and compare the output responses for the first three output .d samples using the difference equation and digital convolution sum (Equation 3.12).

Solution:

a. Let $x(n) = \delta(n)$, then $h(n) = 0.25\,h(n-1) + \delta(n)$

To solve for h(n), we evaluate

$$h(0) = 0.25\,h(-1) + \delta(0) = 0.25\,(0) + 1 = 1$$

$$h(1) = 0.25\,h(0) + \delta(1) = 0.25\,(1) + 0 = 0.25$$

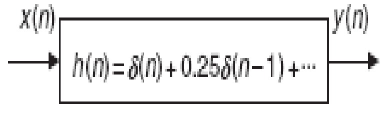
$$h(2) = 0.25\,h(1) + \delta(2) = 0.25\,(0.5) + 0 = 0.0625$$

$\ldots$

With the calculated results, we can predict the impulse response as:

$$h(n) = (0.25)^n u(n) = \delta(n) + 0.25\,\delta(n-1) + 0.0625\,\delta(n-2) + \ldots \ldots$$

b. The system block diagram is given below



c. The output sequence is a sum of infinite terms expressed as

$$y(n) = h(0)\,x(n) + h(1)\,x(n-1) + h(2)x(n-2) + \ldots$$

$$= x(n) + 0.25x(n-1) + 0.0625x(n-2) + \ldots$$

d. From the difference equation and using the zero-initial condition, we have

$$
\begin{aligned}
&y(n) = 0.25y(n-1) + x(n) \text{ for } n \geq 0 \text{ and } y(-1) = 0 \\
&n = 0, y(0) = 0.25y(-1) + x(0) = u(0) = 1 \\
&n = 1, y(1) = 0.25y(0) + x(1) = 0.25 \times u(0) + u(1) = 1.25 \\
&n = 2, y(2) = 0.25y(1) + x(2) = 0.25 \times 1.25 + u(2) = 1.3125 \\
&\ldots
\end{aligned}
$$

Applying the convolution sum in Equation (3.12) yields:

$$y(n) = x(n) + 0.25x(n-1) + 0.0625x(n-2) + \ldots$$

$$
\begin{aligned}
n = 0,\ y(0) &= x(0) + 0.25x(-1) + 0.0625x(-2) + \ldots \\
&= u(0) + 0.25 \times u(-1) + 0.125 \times u(-2) + \ldots = 1 \\
n = 1,\ y(1) &= x(1) + 0.25x(0) + 0.0625x(-1) + \ldots \\
&= u(1) + 0.25 \times u(0) + 0.125 \times u(-1) + \ldots = 1.25 \\
n = 2,\ y(2) &= x(2) + 0.25x(1) + 0.0625x(0) + \ldots \\
&= u(2) + 0.25 \times u(1) + 0.0625 \times u(0) + \ldots = 1.3125 \\
&\ldots
\end{aligned}
$$

Notice that this impulse response h(n) contains an infinite number of terms in its duration due to the past output term y(n − 1). Such a system as described in the preceding example is called an infinite impulse response (IIR) system.

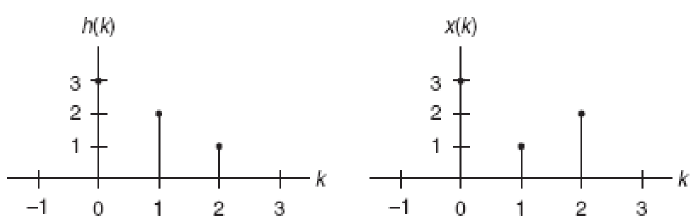
3.8 Digital Convolution

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k)$$

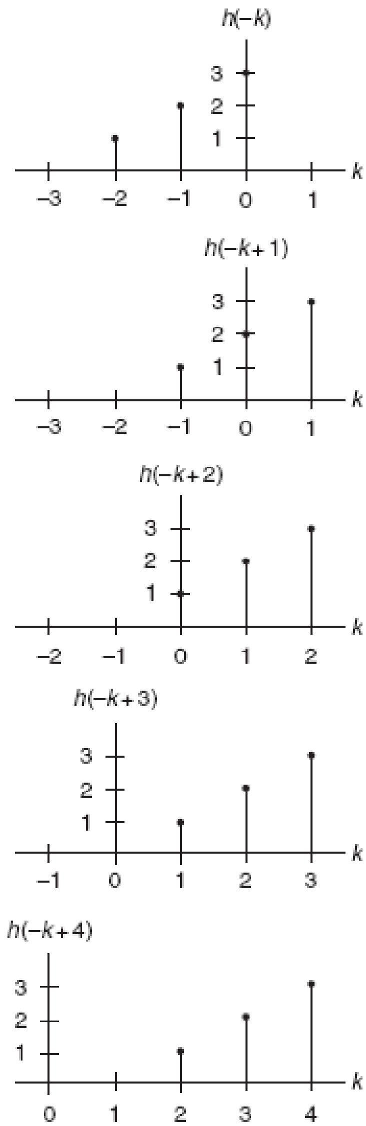
$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \tag{3.13}$$

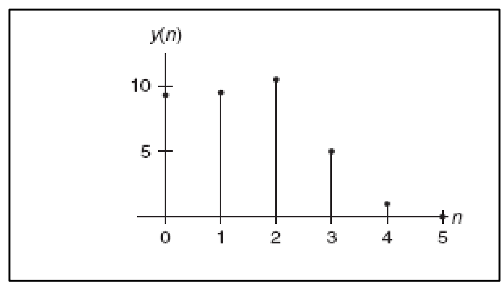
$N = N_1 + N_2 - 1$. Where N_1 = number of samples of x(n), N_2 = number of samples of h(n), and N = total number of samples.

3.8.1 Graphical method:



Example7: Find $y(n) = x(n) \otimes h(n)$ using graphical method





$$n = 0,\ y(0) = x(0)h(0) + x(1)h(-1) + x(2)h(-2) = 3\times 3 + 1\times 0 + 2\times 0 = 9,$$

$$n = 1,\ y(1) = x(0)h(1) + x(1)h(0) + x(2)h(-1) = 3\times 2 + 1\times 3 + 2\times 0 = 9,$$

$$n = 2,\ y(2) = x(0)h(2) + x(1)h(1) + x(2)h(0) = 3\times 1 + 1\times 2 + 2\times 3 = 11,$$

$$n = 3,\ y(3) = x(0)h(3) + x(1)h(2) + x(2)h(1) = 3\times 0 + 1\times 1 + 2\times 2 = 5.$$

$$n = 4,\ y(4) = x(0)h(4) + x(1)h(3) + x(2)h(2) = 3\times 0 + 1\times 0 + 2\times 1 = 2,$$

$$n \geq 5,\ y(n) = x(0)h(n) + x(1)h(n-1) + x(2)h(n-2) = 3\times 0 + 1\times 0 + 2\times 0 = 0.$$

3.8.2 Table lookup method

y(0) = 9

y(1) = 9

y(2) = 11

y(3) = 5

y(4) = 2

	3	2	1
3	9	6	3
1	3	2	1
2	6	4	2

3.8.3 Matrix by Vector method

Example7: If x(n) = [0.5 0.5 0.5], and h(n) = [3 2 1]

$$\begin{bmatrix} 0.5 & 0 & 0 \\ 0.5 & 0.5 & 0 \\ 0.5 & 0.5 & 0.5 \\ 0 & 0.5 & 0.5 \\ 0 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 2.5 \\ 3 \\ 1.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \end{bmatrix}$$

3.8.4 Linear convolution and circular convolution

***Linear convolution*:**

$$x_1(n) \otimes x_2(n) = \sum_{k=-\infty}^{\infty} x_1(n-k)\, x_2(k) = \sum_{k=-\infty}^{\infty} x_1(k)\, x_2(n-k) \qquad (3.14)$$

***Circular convolution*:**

$$x_1(n) \otimes_N x_2(n) = \sum_{k=0}^{N-1} x_1((n-k) \bmod N)\, x_2(k) = \sum_{k=0}^{N-1} x_1(k)\, x_2((n-k) \bmod N) \qquad (3.15)$$

If both $x_1(n)$ and $x_2(n)$ are of *finite length* N_1 and N_2 and defined on [0 N_1-1], and [0 N_2-1] respectively, the value of N needed so that circular and linear convolution are the same on [0 N-1] is : $N \geq N_1 + N_2 - 1$

Example 8: If x(n) = [1 2 3 2], and h(n) = [1 1 2]. Find y(n) such that linear and circular convolution are the same.

Solution:

N = 4 + 3 – 1 = 6

Then x(n) = [1 2 3 2 0 0] and h(n) = [1 1 2 0 0 0] x(n) is arranged in clockwise direction (italic numbers),while h(n) is arranged in the opposite clockwise direction (bold numbers). Each time, <u>only</u> h(n) will be shifted with the ***clockwise direction*** to find y(n). *<u>Note</u>*: the reference point is * and, the arrows represent multiplication process. Finally, addition process is performed.

2	1	1*
0	0	1*
2	3	2
0	0	0

y(0) = 1(1) = 1

0	2	1*
0	0	1*
2	3	2
0	0	1

y(1)= 1(1) + 2(1) = 3

0	0	2*
0	0	1*
2	3	2
0	1	1

y(2)= 2(1)+2(1)+3(1)=7

0	0	0*
0	0	1*
2	3	2
1	1	2

y(3)= 2(2)+3(1)+2(1)=9

1	0	0*
0	0	1*
2	3	2
1	2	0

y(4)= 3(2)+2(1)= 8

1	1	0*
0	0	1*
2	3	2
2	0	0

y(5)= 2(2)= 4

Using table lookup method:

	1	1	2	y(0)= 1
1	1	1	2	y(1)= 3
2	2	2	4	y(2)= 7
3	3	3	6	y(3)= 9
2	2	2	4	y(4)= 8
				y(5)= 4

<u>Example(9):</u> Use graphical method to find circular convolution $x_1(n) \otimes_N x_2(n)$, if N = 4, $x_1(n)$= [1 2 2 0] and $x_2(n)$= [0 1 2 3]

Solution: Applying eq. (3.15), then

$$y(n) = \sum_{k=0}^{3} x_1(k)\; x_2((n-k) \bmod 4)$$

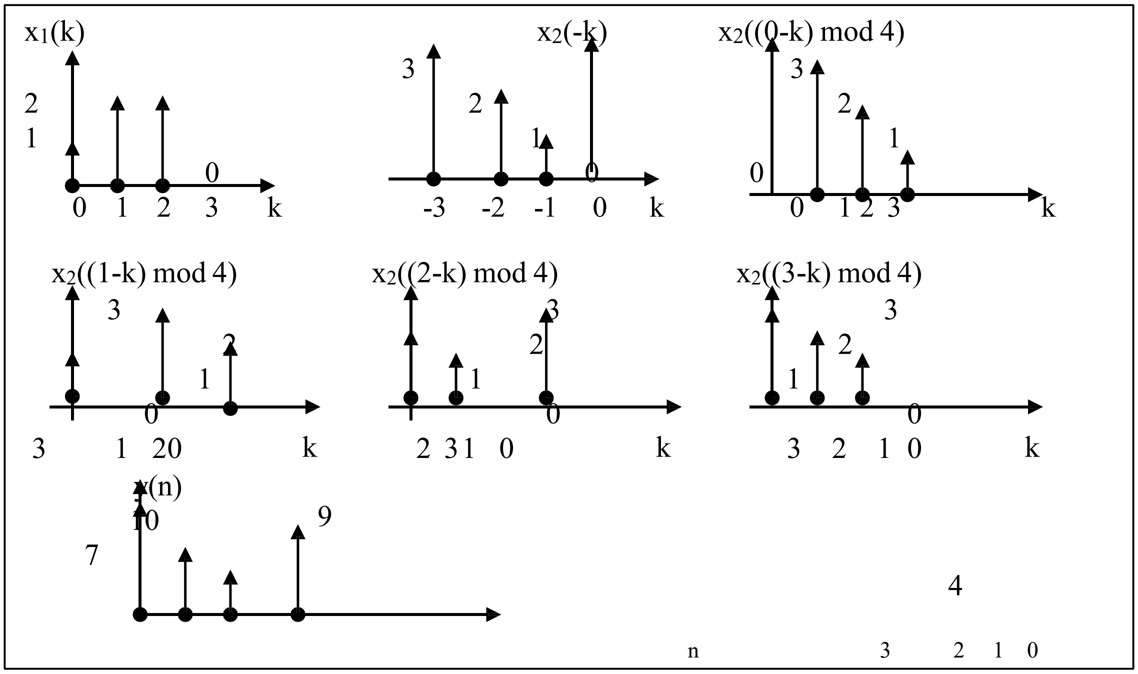
$$y(0) = \sum_{k=0}^{3} x_1(k)\; x_2((-k) \bmod 4)$$

y(0) = $x_1(0)$ $x_2(-0 \bullet 4)$ + $x_1(1)$ $x_2(-1 \bullet 4)$ + $x_1(2)$ $x_2(-2 \bullet 4)$ + $x_1(3)$ $x_2(-3 \bullet 4)$

● = mod addition

$y(0) = x_1(0)\,x_2(0) + x_1(1)\,x_2(3) + x_1(2)\,x_2(2) + x_1(3)\,x_2(1) = 1(0) + 2(3) + 2(2) + 0\,(1) = 10$

And so on



Deconvolution: .9

Iterative approach .1

Using equation (3.14) and assuming causal system (started at k =0), then:

then $x(0) = y(0) / h(0)$ $y(0) = x(0)\,h(0)$,

$y(1) = h(1)\,x(0) + h(0)\,x(1)$, then $x(1) = (\,y(1) – h(1)\,x(0)\,) / h(0)$

Polynomial Approach: .2

A long division process is applied between two polynomials. For causal system, the remainder is always *zero*.

If $y(n) = [\,12\ 10\ 14\ 6]$ and $h(n) = [\,4\ 2\,]$

Then $y = 12 + 10\,x + 14\,x^2 + 6\,x^3$, and $h = 4 + 2\,x$. Applying long division, we

Then $x(n) = [\,3\ 1\ 3\,]$ obtain i/p = $3 + x + 3\,x^2$.

Graphical method .3

$$\begin{bmatrix}4\\2\\ \\ \end{bmatrix} b_0 \times \begin{bmatrix}12\\10\\14\\6\end{bmatrix} \qquad \begin{bmatrix}4\\2\\ \\ \end{bmatrix}\begin{bmatrix}3\\ \times \\ \\ \end{bmatrix} b_1 \times \begin{bmatrix}12\\10\\14\\6\end{bmatrix} \qquad \begin{bmatrix}4\\2\\0\\ \end{bmatrix}\begin{bmatrix}3 \times\\1 \times\\ \\ \end{bmatrix} b_2 \times \begin{bmatrix}12\\10\\14\\6\end{bmatrix} \qquad \begin{bmatrix}4\\2\\0\\0\end{bmatrix}\begin{bmatrix}3 \times\\1 \times\\3 \times\\ \end{bmatrix} b_3 \times \begin{bmatrix}12\\10\\14\\6\end{bmatrix}$$

$4\,b_0 = 12$	$4\,b_1 + 2(3) = 10$	$4\,b_2 + 2(1) + 0\,(3) = 14$	$4\,b_3 + 6 + 0 + 0 = 6$
$b_0 = 3$	$b_1 = 1$	$b_2 = 3$	$b_3 = 0$

So, $x(n) = [\,3\quad 1\quad 3\,]$

9.6 Finite Impulse Response (FIR) filter

In many cases a linear phase c/cs is required throughout the pass-band of the filter to preserve the shape of a given signal within the pass-band. Assume a LP filter with:

$$H(e^{jW}) = \begin{cases} e^{-jW\alpha} & |W| < W_o \\ 0 & W_o < |W| < \pi \\ \text{periodic} & \text{for all other } W \end{cases} \tag{9.32}$$

$$Y(e^{jW}) = X(e^{jW}) \,.\, H(e^{jW}) = X(e^{jW}) \,.\, e^{-jW\alpha} \tag{9.33 a}$$

$$Y(Z) = X(Z) \,.\, Z^{-\alpha} \tag{9.33 b}$$

$$y(n) = x(n - \alpha) \tag{9.34}$$

The linear phase filter did not alter the shape of the original signal, simply translated it by an amount α, as shown in Fig. (9.9)

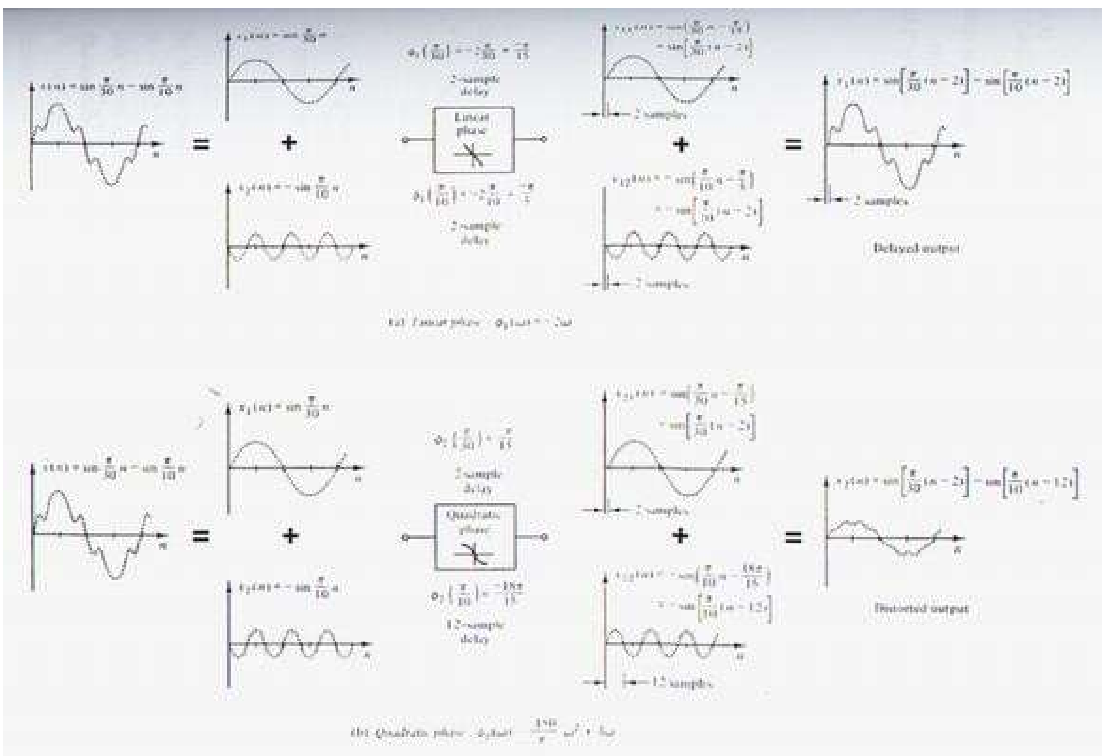


Fig.(9.9) The effect of (a) linear phase and (b) nonlinear phase c/cs on steady state outputs with identical magnitude frequency response curves

A causal IIR filter can not produce a linear phase c/cs and that only special forms of FIR filters can give linear phase.

The necessary conditions for linear phase:

1. h(n) have finite duration (for causal FIR filter, h(n) begins at zero and ends at N-1)

Lec. 9 – Part 2

9.6 Finite Impulse Response (FIR) filter

In many cases a linear phase c/cs is required throughout the pass-band of the filter to preserve the shape of a given signal within the pass-band. Assume a LP filter with:

$$H(e^{jW}) = \begin{cases} e^{-jW\alpha} & |W| < W_o \\ 0 & W_o < |W| < \pi \\ \text{for all other } W & \text{periodic} \end{cases} \tag{9.32}$$

$$Y(e^{jW}) = X(e^{jW}) \, . \, H(e^{jW}) = X(e^{jW}) \, . \, e^{-jW\alpha} \tag{9.33 a}$$

$$Y(Z) = X(Z) \, . \, Z^{-\alpha} \tag{9.33 b}$$

$$y(n) = x(n - \alpha) \tag{9.34}$$

The linear phase filter did not alter the shape of the original signal, simply translated it by an amount α, as shown in Fig. (9.9)

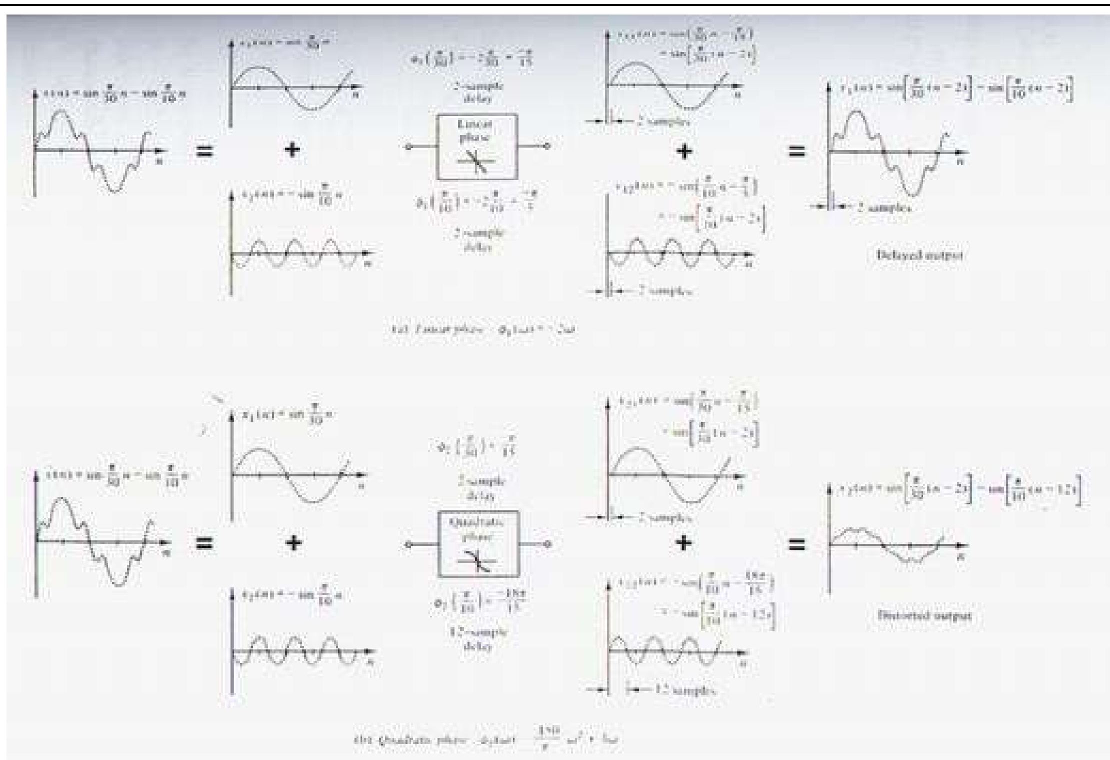


Fig.(9.9) The effect of (a) linear phase and (b) nonlinear phase c/cs on steady state outputs with identical magnitude frequency response curves

A causal IIR filter can not produce a linear phase c/cs and that only special forms of FIR filters can give linear phase.

The necessary conditions for linear phase:

1. h(n) have finite duration (for causal FIR filter, h(n) begins at zero and ends at N-1)

$$, n = 0, 1, \ldots, N-1 \quad h(n) = h(N-1-n) \qquad (9.35)$$

2. Symmetric about its mid-point (see Fig. (9.10))

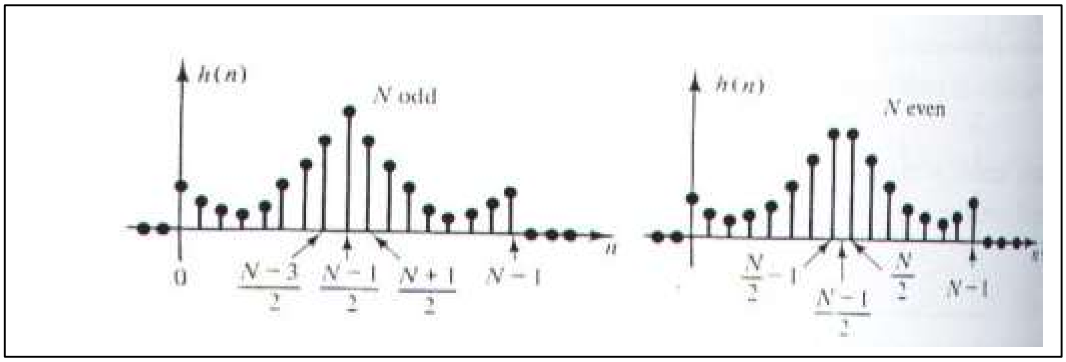


Fig. (9.10) General shapes of h(n) that give linear phase for odd and even N.

If h(n) is as given in the above conditions, we now show that $H(e^{jW})$ has linear phase. For N even:

$$H(e^{jW}) = \sum_{n=0}^{N-1} h(n)\, e^{-jWn} = \sum_{n=-\infty}^{\infty} h(n)\, e^{-jWn} \quad \text{(Finite duration)} \qquad (9.36)$$

$$H(e^{jW}) = \sum_{n=N/2}^{N-1} h(n)\, e^{-jWn} = \sum_{n=0}^{(N/2)-1} h(n)\, e^{-jWn} \quad = H_2(e^{jW}) + H_1(e^{jW}) \qquad (9.37)$$

Let m = N-1-n

$$H_2(e^{jW}) = \sum_{m=(\frac{N}{2})-1}^{0} h(N-1-m)\, e^{-jW(N-1-m)} = \sum_{m=0}^{(\frac{N}{2})-1} h(m)\, e^{-jW(N-1-m)} \qquad (9.38)$$

$$\therefore H(e^{jW}) = \sum_{m=0}^{(\frac{N}{2})-1} h(n)\, e^{-jWn} = \sum_{n=0}^{(\frac{N}{2})-1} h(m)\, e^{-jW(N-1-m)} \qquad (9.39)$$

$$H(e^{jW}) = \sum_{n=0}^{(\frac{N}{2})-1} h(n)\, e^{-jW(\frac{N-1}{2})} \left\{ e^{-jW(n-\frac{N-1}{2})} + e^{-jW(N-1-n-\frac{N-1}{2})} \right\} \qquad (9.40)$$

$$H(e^{jW}) = \sum_{n=0}^{(\frac{N}{2})-1} 2\, h(n)\, e^{-jW(\frac{N-1}{2})} \left\{ \cos\left[W\left(n - \frac{N-1}{2}\right)\right] \right\} \qquad (9.41)$$

For N even:

$$H(e^{jW}) = e^{-jW\left(\frac{N-1}{2}\right)} \sum_{n=0}^{\left(\frac{N}{2}\right)-1} 2\,h(n)\left\{\cos\left[W\left(n-\frac{N-1}{2}\right)\right]\right\} \qquad (9.42)$$

Linear phase *magnitude*

For N odd:

$$H(e^{jW}) = e^{-jW\left(\frac{N-1}{2}\right)}\left\{h\left(\frac{N-1}{2}\right) + \sum_{n=0}^{(N-3)/2} 2\,h(n)\left\{\cos\left[W\left(n-\frac{N-1}{2}\right)\right]\right\}\right\} \qquad (9.43)$$

For N odd, the slope of – α = – (N–1) /2 causes a delay in the output of (N–1)/2 , which is an integer number of samples, whereas for N even, the slope causes a non-integer delay. The non-integer delay will cause the values of the sequence to be changed, which, in some cases, may be undesirable.

9.7 Design of FIR filters using Windows

If $h_d(n)$ represents the impulse response of a desired IIR filter, then an FIR filter with impulse response h(n) can be obtained as follows:

$$h(n) = h_d(n)\,.\,w(n)$$

$$h(n) = \begin{cases} h_d(n) & N_1 \le n \le N_2 \\ 0 & otherwise \end{cases}$$

$$w(n) = \begin{cases} 1 & N_1 \le n \le N_2 \\ 0 & otherwise \end{cases} \quad , \textit{window function} \qquad (9.44)$$

$$H(e^{jW}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_d(e^{j\theta})\,W(e^{j(W-\theta)})\,d\theta = H_d(e^{jW}) \otimes W(e^{jW})$$

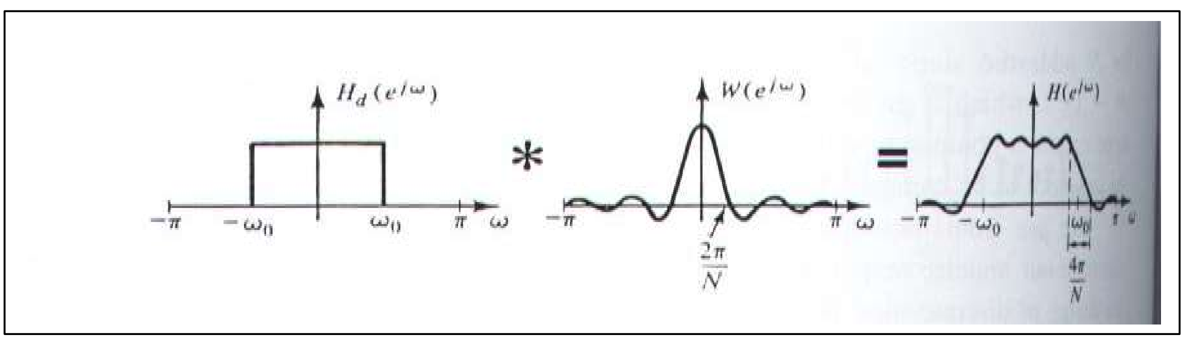


Fig. (9.11) Frequency response obtained by rectangularly windowing ideal LP impulse response.

As shown in Fig.(9.11), the convolution produces a smeared version of ideal LP frequency response $H_d(e^{jW})$. In general, the wider the main lobe of $W(e^{jW})$, the more spreading, whereas the narrower the main lobe (larger N), the closer $|H(e^{jW})|$ comes to $|H_d(e^{jW})|$.

Some of the most commonly used windows are:

Lec. 5 **Z - Transform**

5.1 Definition of Z.T

The z-transform is a very important tool in describing and analyzing digital systems. It also offers the techniques for digital filter design and frequency analysis of digital signals.
The z-transform of a *causal* sequence x(n), designated by X(z) or Z(x(n)), is defined as:

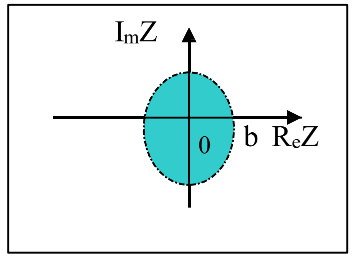
$$X(z) = Z(x(n)) = \sum_{n=0}^{\infty} x(n)z^{-n}$$
$$= x(0)z^{-0} + x(1)z^{-1} + x(2)z^{-2} + \dots \qquad (5.1)$$

Where, z is the complex variable. Here, the summation taken from n = 0 to n = ∞ is according to the fact that for most situations, the digital signal x(n) is the ***causal*** sequence, that is, x(n) = 0 for n ≤ 0. For non-causal system, the summation starts at n = -∞. Thus, the definition in Equation (5.1) is referred to as a one-sided z-transform or a unilateral transform. The region of convergence is defined based on the particular sequence x(n) being applied. The z-transforms for common sequences are summarized below:

Line No.	$x(n)$, $n \geq 0$	z-Transform $X(z)$	Region of Convergence
1	$x(n)$	$\sum_{n=0}^{\infty} x(n)z^{-n}$	
2	$\delta(n)$	1	$\|z\| > 0$
3	$au(n)$	$\frac{az}{z-1}$	$\|z\| > 1$
4	$nu(n)$	$\frac{z}{(z-1)^2}$	$\|z\| > 1$
5	$n^2u(n)$	$\frac{z(z+1)}{(z-1)^3}$	$\|z\| > 1$
6	$a^n u(n)$	$\frac{z}{z-a}$	$\|z\| > \|a\|$
7	$e^{-na}u(n)$	$\frac{z}{(z-e^{-a})}$	$\|z\| > e^{-a}$
8	$na^n u(n)$	$\frac{az}{(z-a)^2}$	$\|z\| > \|a\|$
9	$\sin(an)u(n)$	$\frac{z\sin(a)}{z^2-2z\cos(a)+1}$	$\|z\| > 1$
10	$\cos(an)u(n)$	$\frac{z[z-\cos(a)]}{z^2-2z\cos(a)+1}$	$\|z\| > 1$
11	$a^n\sin(bn)u(n)$	$\frac{[a\sin(b)]z}{z^2-[2a\cos(b)]z+a^2}$	$\|z\| > \|a\|$
12	$a^n\cos(bn)u(n)$	$\frac{z[z-a\cos(b)]}{z^2-[2a\cos(b)]z+a^{-2}}$	$\|z\| > \|a\|$
13	$e^{-an}\sin(bn)u(n)$	$\frac{[e^{-a}\sin(b)]z}{z^2-[2e^{-a}\cos(b)]z+e^{-2a}}$	$\|z\| > e^{-a}$
14	$e^{-an}\cos(bn)u(n)$	$\frac{z[z-e^{-a}\cos(b)]}{z^2-[2e^{-a}\cos(b)]z+e^{-2a}}$	$\|z\| > e^{-a}$
15	$2\|A\|\|P\|^n\cos(n\theta+\phi)u(n)$ where P and A are complex constants defined by $P=\|P\|\angle\theta, A=\|A\|\angle\phi$	$\frac{Az}{z-P}+\frac{A^*z}{z-P^*}$	

<u>Example(1)</u>: Find Z.T including region of convergence of x(n) = - b^n u(- n - 1)

Solution: the system is non- causal



$$X(Z)=\sum_{n=-\infty}^{-1}-b^n u(-n-1)Z^{-n}=-\sum_{n=-\infty}^{\infty}(b/Z)^n$$

Let $m = -n$

$$X(Z)=-\sum_{m=0}^{\infty}(Z/b)^m=1-\sum_{m=1}^{\infty}(Z/b)^m$$

By using $\sum_{m=0}^{\infty} x^m = 1 + x + x^2 + x^3 + ... = \frac{1}{x-1}$, $|x| \langle 1$

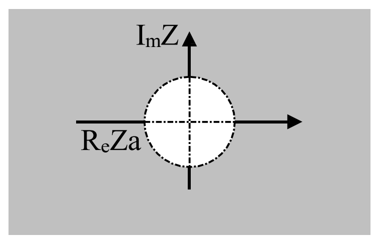
$$X(Z)=1-\frac{1}{1-(Z/b)}=\frac{Z}{Z-b}, \quad |Z| \langle\, |b| \quad or \quad |Z/b| \langle 1$$

The region of convergence (ROC) is <u>inside</u> the unit circle only.

<u>Example(2):</u> Find Z.T including region of convergence of x(n) = a^n u(n)

$$X(Z)=\sum_{n=0}^{\infty}a^n Z^{-n}=\sum_{n=0}^{\infty}(aZ^{-1})^n=\frac{1}{1-aZ^{-1}}=\frac{Z}{Z-a}, \quad |aZ^{-1}| \langle 1$$

Or $|Z| > |a|$



The region of convergence (ROC) is <u>outside</u> the unit circle only.

<u>5.2 Properties of Z.T:</u>

<u>5.2.1 Linearity</u>: The z-transform is a linear transformation, which implies

$$Z\,(a\;x_1(n) \pm b\;x_2(n)\,) = a\;X_1(Z) \pm b\;X_2(Z) \qquad (5.2)$$

Where *a* and *b* are constants

<u>5.2.2 Shift theorem (without initial conditions)</u>: Given X(z), the z-transform of a sequence x(n), the z-transform of x(n - m), the time-shifted sequence, is given by;

$$Z\{x(n-m)\}=Z^{-m}X(Z) \qquad (5.3)$$

5.2.3 Convolution: Given two sequences $x_1(n)$ and $x_2(n)$, their convolution can be determined as follows:

$$x(n) = x_1(n) \otimes x_2 = \sum_{k=-\infty}^{\infty} x_1(k)\, x_2(n-k) = \sum_{k=-\infty}^{\infty} x_1(n-k)\, x_2(k) \qquad (5.4)$$

Where ⊗ designates the linear convolution. In z-transform domain, we have

$$X(Z) = X_1(Z)\,.\,X_2(Z) \qquad (5.5)$$

5.2.4 Multiplication by exponential:

$$Z\{\, a^n\, x(n)\,\} = X(Z)\,\Big|_{Z\to\frac{Z}{a}} \qquad (5.6.a)$$

$$Z\{e^{\pm an} x(n)\} = X(Z)\,\Big|_{Z\to e^{\mp a} Z} \qquad (5.6.b)$$

5.2.5 Initial and final value theorems:

$$\textit{initial value theorem} \quad \lim_{n\to 0} x(n) = \lim_{Z\to\infty} X(Z) = x(0) \qquad (5.7.a)$$

$$\lim_{n\to\infty} x(n) = \lim_{Z\to 1} Z^{-1}(Z-1)\, X(Z) \qquad \textit{final value theorem} \qquad (5.7.b)$$

5.2.6 Multiplication by n:

$$Z\{\, n\, x(n)\,\} = -Z \frac{d}{dZ} X(Z) \qquad (5.8)$$

Example(3) : Find $Z\{(n-2)\, a^{(n-2)} \cos[\, w(n-2)]\, u(n-2)$.

The solution is:

$$= Z^{-2} Z\{\; n\, a^n \cos\; wn\, u(n)\;\}$$

$$= Z^{-2}(-Z)\frac{d}{dZ} Z\{\; a^n \cos\; wn\, u(n)\;\}$$

$$= -Z^{-1} \frac{d}{dZ} \frac{Z^2 - Z\cos wd}{Z^2 - 2\,Z\cos\, w + 1}\,\Big|_{Z\to\frac{Z}{a}}$$

5.3 Inverse of Z.T

$$x(n) = Z^{-1}\{X(Z)\} \tag{5.9}$$

The inverse z-transform may be obtained by the following methods:

Using properties. .1

Partial fraction expansion method. .2

Residue method. .3

4. Power series expansion (the solution is obtained by applying long division because the denominator can't be analyzed. It is not accurate method compared with the above three methods)

Example(4): Find x(n), using properties , if

$$X(z) = \frac{10z}{z^2 - z + 1}$$

Solution:

Since $X(z) = \dfrac{10z}{z^2 - z + 1} = \left(\dfrac{10}{\sin(a)}\right)\dfrac{\sin(a)z}{z^2 - 2z\cos(a) + 1}$,
by coefficient matching, we have

$$-2\cos(a) = -1.$$

Hence, $\cos(a) = 0.5$, and $a = 60°$. Substituting $a = 60°$ into the sine function leads to

$$\sin(a) = \sin(60°) = 0.866.$$

Finally, we have

$$x(n) = \frac{10}{\sin(a)} Z^{-1}\left(\frac{\sin(a)z}{z^2 - 2z\cos(a) + 1}\right) = \frac{10}{0.866}\sin(n \cdot 60^0)$$

$$= 11.547\sin(n \cdot 60^0).$$

Example(5): Find x(n) using partial fraction method , if:

$$X(z) = \frac{1}{(1 - z^{-1})(1 - 0.5z^{-1})}.$$

Solution:

Eliminating the negative power of z by multiplying the numerator and denominator by z^2 yields

$$X(z) = \frac{z^2}{z^2(1-z^{-1})(1-0.5z^{-1})}.$$
$$= \frac{z^2}{(z-1)(z-0.5)}$$

Dividing both sides by z leads to

$$\frac{X(z)}{z} = \frac{z}{(z-1)(z-0.5)}.$$

Again, we write

$$\frac{X(z)}{z} = \frac{A}{(z-1)} + \frac{B}{(z-0.5)}.$$

$$A = (z-1)\frac{X(z)}{z}\bigg|_{z=1} = \frac{z}{(z-0.5)}\bigg|_{z=1} = 2,$$

$$B = (z-0.5)\frac{X(z)}{z}\bigg|_{z=0.5} = \frac{z}{(z-1)}\bigg|_{z=0.5} = -1.$$

Thus

$$\frac{X(z)}{z} = \frac{2}{(z-1)} + \frac{-1}{(z-0.5)}.$$

Multiplying z on both sides gives

$$X(z) = \frac{2z}{(z-1)} + \frac{-z}{(z-0.5)}.$$

$$x(n) = 2u(n) - (0.5)^n u(n).$$

Example(6) : Find x(n) using the residue theorem, if

$$X(Z) = \frac{2Z}{(Z-1)^2\,(Z-2)\,(Z-3)}$$

The residue theorem is:

x(n) = ∑ residues of $X(Z)\,Z^{n-1}$ at the poles of $X(Z)\,Z^{n-1} = a_{-1} + b_{-1} + c_{-1} + \ldots$ (5.10)

$$a_{-1} = \frac{1}{(m-1)!}\lim_{Z\to a}\frac{d^{m-1}}{dZ^{m-1}}\{(Z-a)^m X(Z) Z^{n-1}\},\quad m \text{ is the order of the pole} \qquad (5.11)$$

Solution:

$$\Rightarrow a = \frac{1}{1!}\lim_{Z\to 1}\frac{d}{dZ}\frac{2Z\, Z^{n-1}}{(Z-2)(Z-3)} = n + \frac{3}{2}$$

$$\Rightarrow b = \frac{1}{0!}\lim_{Z\to 2}\frac{2Z^{n}}{(Z-1)^2(Z-3)} = -2(2)^n$$

$$\Rightarrow c = \frac{1}{0!}\lim_{Z\to 3}\frac{2Z^{n}}{(Z-1)^2(Z-2)} = \frac{1}{2}(3)^n$$

$$x(n) = a + b + c = n + \frac{3}{2} - 2(2)^n + \frac{1}{2}(3)^n$$

5.4 Solution of linear constant coefficient difference equation using Z.T

$$Z\{x(n-m)\} = Z^{-m}\{X(Z) + \sum_{k=-m}^{-1} x(k)\, Z^{-k}\} \qquad (5.12)$$

Example(7) : Solve y(n) – (3/2) y(n – 1) + (1/2) y(n – 2) = (1/4)^n, y(-1) = 4, y(-2) = 10 for n ≥ 0

Solution:

$$Y(Z) - \frac{3}{2}\{Y(Z)\,.\,Z^{-1} + y(-1)\} + \frac{1}{2}\{Z^{-2}\,Y(Z) + Z^{-1}\,y(-1) + y(-2)\} = \frac{Z}{Z-\frac{1}{4}}$$

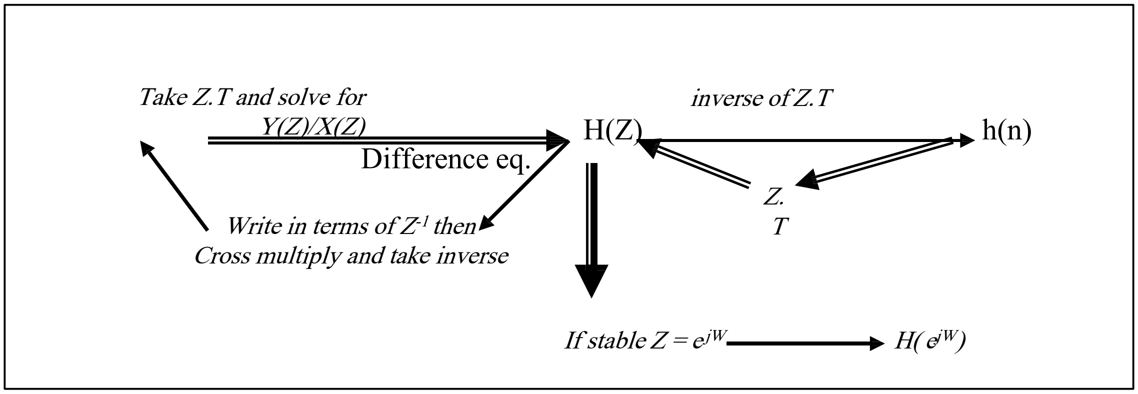
$$Y(Z)\{1 - \frac{3}{2}Z^{-1} + \frac{1}{2}Z^{-2}\} = \frac{Z}{Z-\frac{1}{4}} + 1 - 2Z^{-1}$$

$$Y(Z) = \frac{Z(2Z^2 - \frac{9}{4}Z + \frac{1}{2})}{(Z-\frac{1}{2})(Z-\frac{1}{4})(Z-1)}$$

$$Y(Z) = \frac{Z}{Z-\frac{1}{2}} + \frac{(1/3)\,Z}{Z-\frac{1}{4}} + \frac{(2/3)\,Z}{Z-1}$$

$$y(n) = \{(\frac{1}{2})^n + \frac{1}{3}(\frac{1}{4})^n + \frac{2}{3}\}\,u(n)$$

5.5 Relations between system representations:



Continuous time system	Discrete time system
Differential equation	Difference equation
H_a (S)	H(Z)
H(jΩ)	$H(e^{jW})$
$H_a(t) = L^{-1}$ { H_a (S)}	$h(n) = Z^{-1}$ { H(Z) }

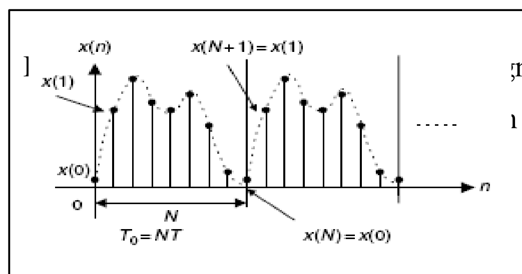
Discrete Fourier Transform .1

In time domain, representation of digital signals describes the signal amplitude versus the sampling time instant or the sample number. However, in some applications, signal frequency content is very useful than as digital signal samples.

The algorithm transforming the time domain signal samples to the frequency domain components is known as the discrete Fourier transform, or DFT. The DFT also establishes a relationship between the time domain representation and the frequency domain representation. Therefore, we can apply the DFT to perform frequency analysis of a time domain sequence. In addition, the DFT is widely used in many other areas, including spectral analysis, acoustics, imaging/ video, audio, instrumentation, and communications systems.

Fourier Series Coefficients of Periodic Digital Signals .2

To estimate the spectrum of a periodic digital signal x(n), sampled at a rate of f_s Hz with where there are N samples within the duration of the the fundamental period $T_0 = NT$, fundamental period and $T = 1/f_s$ is the sampling period. Fig. 6.1 shows periodic digital signal.



$$c_k = \frac{1}{T_0}\int_0^{T_0} x(t)\, e^{-jkW_0 t}\, dt, \quad -\infty \le k \le \infty \qquad (6.1)$$

Fig. 6.1 periodic digital signal

Where, k is the number of harmonics corresponding to the harmonic frequency of kf_0 and $W_0 = 2\pi / T_0$ and $f_0 = 1/T_0$ are the fundamental frequency in radians per second and the fundamental frequency in Hz, respectively. To apply Equation (6.1), we substitute $T_0 = NT$, $W_0 = 2\pi / T_0$ and approximate the integration over one period using a summation by substituting $dt = T$ and $t = nT$. We obtain:

$$c_k = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\, e^{-j\frac{2\pi kn}{N}} \quad -\infty \le k \le \infty, \qquad (6.2)$$

Since the coefficients c_k are obtained from the Fourier series expansion in the complex form, the resultant spectrum c_k will have two sides. Therefore, the two-sided line amplitude spectrum $|c_k|$ is periodic, as shown in Fig. 6.2.

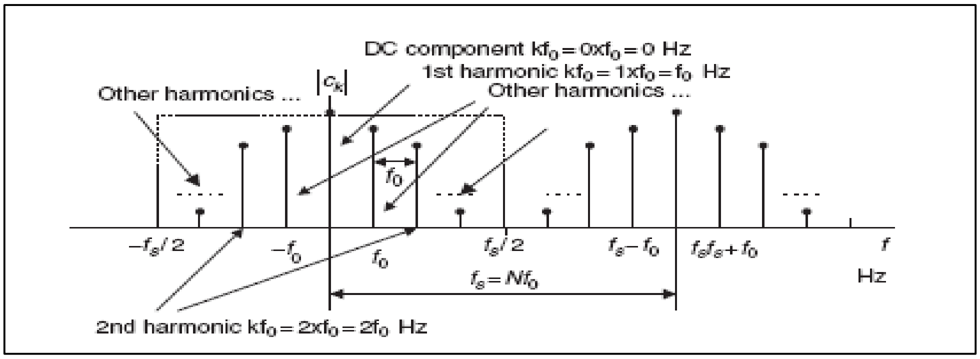


Fig. 6.2 Amplitude Spectrum of periodic Digital signal

As displayed in Figure 6.3 we note the following points:

a. Only the line spectral portion between the frequency $-f_s/2$ and frequency $f_s/2$ (folding frequency) represents the frequency information of the periodic signal.

b. The spectrum is periodic for every Nf_0 Hz.

c. For the kth harmonic, the frequency is $f = kf_0$ Hz. f_0 is called the frequency resolution.

Example(1):

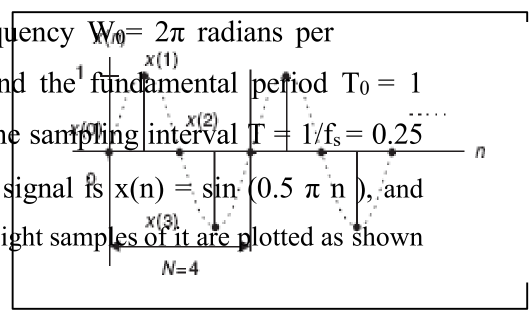
The periodic signal $x(t) = \sin(2\pi t)$ is sampled using the rate $f_s = 4$ Hz.

a. Compute the spectrum c_k using the samples in one period.

b. Plot the two-sided amplitude spectrum $|c_k|$ over the range from −2 to 2 Hz.

Solution:

The fundamental frequency $W_0 = 2\pi$ radians per second and $f_0 = 1$, and the fundamental period $T_0 = 1$ second. Since using the sampling interval $T = 1/f_s = 0.25$ second. The sampled signal is $x(n) = \sin(0.5\pi n)$, and the first eight samples of it are plotted as shown



Choosing one period, N = 4, we have x(0) = 0; x(1) = 1; x(2) = 0; and x(3) = −1. Using Eq. (6.2),

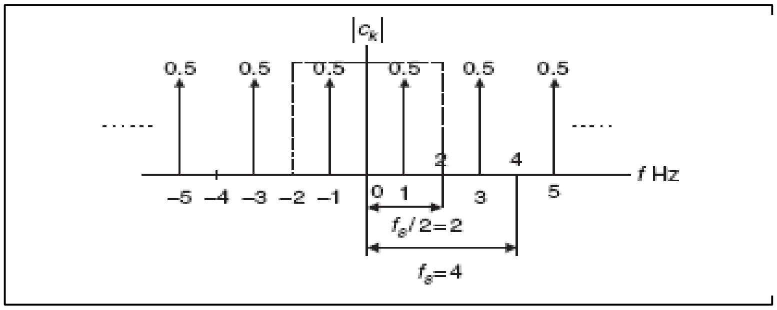
$$c_0 = \frac{1}{4}\sum_{n=0}^{3} x(n) = \frac{1}{4}(x(0) + x(1) + x(2) + x(3)) = \frac{1}{4}(0 + 1 + 0 - 1) = 0$$

$$c_1 = \frac{1}{4}\sum_{n=0}^{3} x(n)e^{-j2\pi\times 1n/4} = \frac{1}{4}\left(x(0) + x(1)e^{-j\pi/2} + x(2)e^{-j\pi} + x(3)e^{-j3\pi/2}\right)$$

$$= \frac{1}{4}(x(0) - jx(1) - x(2) + jx(3) = 0 - j(1) - 0 + j(-1)) = -j0.5.$$

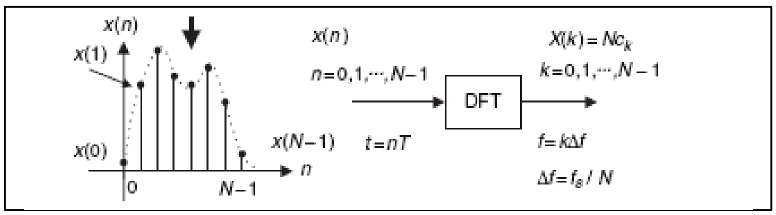
Similarly $c_3 = j\,0.5$. Using periodicity, it follows that $c_{-1} = c_1 = j0{:}5$, and $c_{-2} = c_2 = 0$.

b. The amplitude spectrum for the digital signal is sketched below:



6.3 Discrete Fourier Transform Formulas

Given a sequence $x(n)$, $0 \le n \le N-1$, its **DFT** is defined as:



$$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{-j\frac{2\pi kn}{N}} = \sum_{n=0}^{N-1} x(n)\, W_N^{kn}, \quad \text{for } k = 0, 1, \ldots, N-1 \tag{6.3}$$

Where the factor W_N (called the twiddle factor in some textbooks) is defined as

$$W_N = e^{-j\frac{2\pi}{N}} = \cos\left(\frac{2\pi}{N}\right) - j\sin\left(\frac{2\pi}{N}\right) \tag{6.4}$$

The **inverse DFT** is given by:

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)\, e^{j\frac{2\pi kn}{N}} = \frac{1}{N}\sum_{k=0}^{N-1} X(k)\, W_N^{-kn}, \quad \text{for } n = 0, 1, \ldots, N-1 \tag{6.5}$$

We can use MATLAB functions **fft()** and **ifft()** to compute the DFT coefficients and the inverse DFT.

Example (2): Given a sequence $x(n)$ for $0 \le n \le 3$, where $x(0) = 1$, $x(1) = 2$, $x(2) = 3$, and $x(3) = 4$. Evaluate its DFT $X(k)$.

Solution:

Since $N = 4$, $W_4 = e^{-j\pi/2}$, then using:

$$X(k) = \sum_{n=0}^{3} x(n) W_4^{kn} = \sum_{n=0}^{3} x(n) e^{-j\frac{\pi kn}{2}}.$$

For K= 0, X(0) = 10. Similarly, X(1) = −2 + j 2 , X(2) = −2, X(3) = −2 −j 2

Let us verify the result using the MATLAB function **fft():**

X = fft([1 2 3 4])

− 2.0000 − 2.0000i− 2.0000X = 10.0000   −2.0000+ 2.0000i

Mapping the frequency bin k to its corresponding frequency is as follows:

$$\omega = \frac{k\omega_s}{N} \text{ (radians per second)}, \tag{6.6}$$

Since $w_s = 2\pi f_s$, then:

$$f = \frac{kf_s}{N} \text{ (Hz)}, \tag{6.7}$$

We can define the frequency resolution as the frequency step between two consecutive DFT coefficients to measure how fine the frequency domain presentation is and achieve

$$\Delta\omega = \frac{\omega_s}{N} \text{ (radians per second)}, \tag{6.8}$$

$$\Delta f = \frac{f_s}{N} \text{ (Hz)}. \tag{6.9}$$

Example (3): In example (2), If the sampling rate is 10 Hz,

a. Determine the sampling period, time index, and sampling time instant for a digital sample x(3) in time domain.

b. Determine the frequency resolution, frequency bin number, and mapped frequency for each of the DFT coefficients X(1) and X(3) in frequency domain.

Solution:

a. In time domain, we have the sampling period calculated as

$$T = 1/f_s = 1/10 = 0.1 \text{ second}.$$

For data $x(3)$, the time index is $n = 3$ and the sampling time instant is determined by

$$t = nT = 3 \cdot 0.1 = 0.3 \text{ second}.$$

b. In frequency domain, since the total number of DFT coefficients is four, the frequency resolution is determined by

$$\Delta f = \frac{f_s}{N} = \frac{10}{4} = 2.5 \text{ Hz}.$$

The frequency bin number for $X(1)$ should be $k = 1$ and its corresponding frequency is determined by

$$f = \frac{kf_s}{N} = \frac{1 \times 10}{4} = 2.5 \text{ Hz}.$$

Similarly, for $X(3)$ and $k = 3$,

$$f = \frac{kf_s}{N} = \frac{3 \times 10}{4} = 7.5 \text{ Hz}.$$

6.4 Amplitude Spectrum and Power Spectrum

One of the DFT applications is transformation of a finite-length digital signal x(n) into the spectrum in frequency domain. Fig. 6.3 demonstrates such an application, where A_k and P_k are the computed amplitude spectrum and the power spectrum, respectively, using the DFT coefficients X(k).

First, we achieve the digital sequence x(n) by sampling the analog signal x(t) and truncating the sampled signal with a data window with a length $T_0 = NT$, where T is the sampling period and N the number of data points. The time for data window is $T_0 = NT$**.**

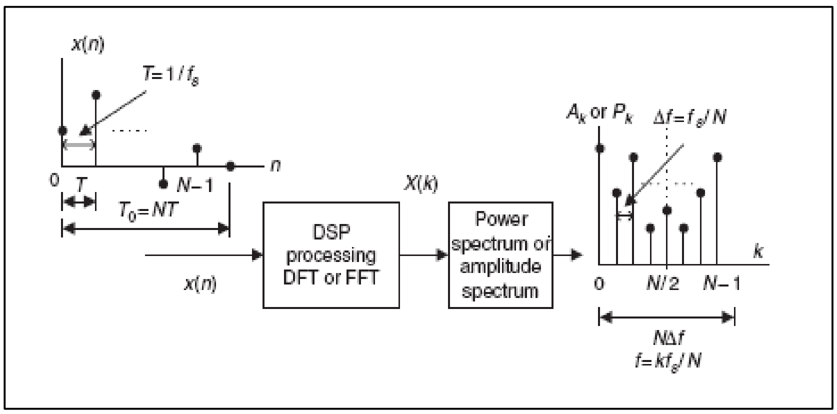


Fig. 6.3 Applications of DFT/ FFT

Next, we apply the DFT to the obtained sequence, x(n), to get the N DFT coefficients

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}, \text{ for } k = 0, 1, 2, \ldots, N-1. \tag{6.10}$$

We define the amplitude spectrum as:

$$A_k = \frac{1}{N}|X(k)| = \frac{1}{N}\sqrt{(\mathrm{Real}[X(k)])^2 + (\mathrm{Imag}[X(k)])^2},$$
$$k = 0, 1, 2, \ldots, N-1. \tag{6.11}$$

Keeping original DC term at k = 0, a one-sided amplitude spectrum for equation (6.11) is:

$$\bar{A}_k = \begin{cases} \frac{1}{N}|X(0)|, & k = 0 \\ \frac{2}{N}|X(k)|, & k = 1, \ldots, N/2 \end{cases} \tag{6.12}$$

Correspondingly, the phase spectrum is given by:

$$\varphi_k = \tan^{-1}\left(\frac{\mathrm{Imag}[X(k)]}{\mathrm{Real}[X(k)]}\right), \; k = 0, 1, 2, \ldots, N-1. \tag{6.13}$$

Besides the amplitude spectrum, the power spectrum is also used. The DFT power spectrum is defined as:

$$P_k = \frac{1}{N^2}|X(k)|^2 = \frac{1}{N^2}\left\{(\mathrm{Real}[X(k)])^2 + (\mathrm{Imag}[X(k)])^2\right\},$$
$$k = 0, 1, 2, \ldots, N-1. \qquad (6.14)$$

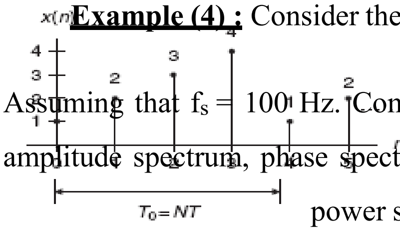
Similarly, for a one-sided power spectrum, we get:

$$\bar{P}_k = \begin{cases} \frac{1}{N^2}|X(0)|^2 & k = 0 \\ \frac{2}{N^2}|X(k)|^2 & k = 0, 1, \ldots, N/2 \end{cases}$$

and $$f = \frac{kf_s}{N}. \qquad (6.15)$$

The frequency resolution is defined in equation (6.9). It follows that better frequency resolution can be achieved by using a longer data sequence.

Example (4) : Consider the sequence:



Assuming that f_s = 100 Hz. Compute the amplitude spectrum, phase spectrum, and power spectrum.

Solution:

Since N = 4, DFT coefficients are: X(0) = 10, X(1) = −2 + j 2 , X(2) = −2, X(3) = −2 −j 2

For $k = 0$, $f = k \cdot f_s / N = 0 \times 100/4 = 0\,\mathrm{Hz}$,

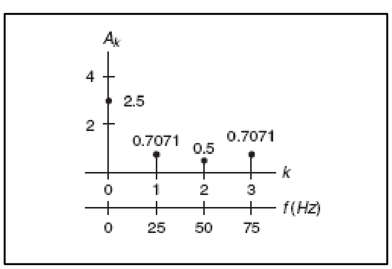
$$A_0 = \frac{1}{4}|X(0)| = 2.5,\ \varphi_0 = \tan^{-1}\left(\frac{\mathrm{Imag}[X(0)]}{\mathrm{Real}([X(0)]}\right) = 0^0,$$

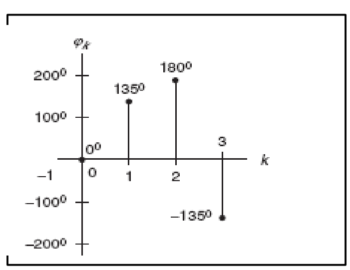
$$P_0 = \frac{1}{4^2}|X(0)|^2 = 6.25.$$

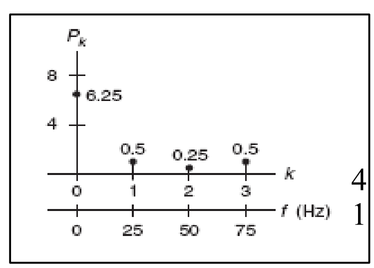
Similarly:

K	*f*	A_K	Φ_K *in degree*	P_K
1	25	0.7071	135	0.5
2	50	0.5	180	0.25
3	75	0.7071	-135	0.5

Thus, the sketches for the amplitude spectrum, phase spectrum, and power spectrum are given in the below Figures:



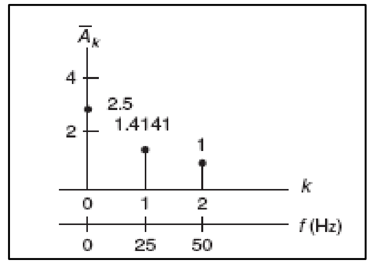




We can easily find the one-sided amplitude spectrum and one-sided power spectrum as:

$\bar{A}_0 = 2.5,\ \bar{A}_1 = 1.4141,\ \bar{A}_2 = 1$ and
$\bar{P}_0 = 6.25,\ \bar{P}_1 = 2,\ \bar{P}_2 = 1.$

We plot the one-sided amplitude spectrum for comparison:



Note that in the one-sided amplitude spectrum, the negative-indexed frequency components are added back to the corresponding positive-indexed frequency components; thus each amplitude value other than the DC term is doubled. It represents the frequency components up to the folding frequency.

7.1 Spectral Estimation Using Window Functions

Consider the pure 1-Hz sine wave with 32 samples shown in Fig. 7.1. As shown in the figure, if we use a window size of N =16 samples, which is a multiple of the two waveform cycles, the second window repeats with continuity. However, when the window size is chosen to be 18 samples, which is not a multiple of the waveform cycles (2.25 cycles), the second window repeats the first window with discontinuity. *It is this discontinuity that produces harmonic frequencies that are not present in the original signal (spectral leakage).* Fig.7.2 shows the spectral plots for both cases using the DFT/FFT directly.

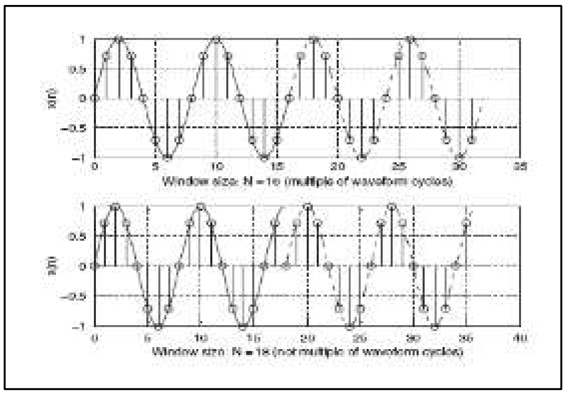


Fig. 7.1 Sampling a 1-Hz sine wave using (top) 16 samples per cycle and (bottom) 18 samples per cycle.

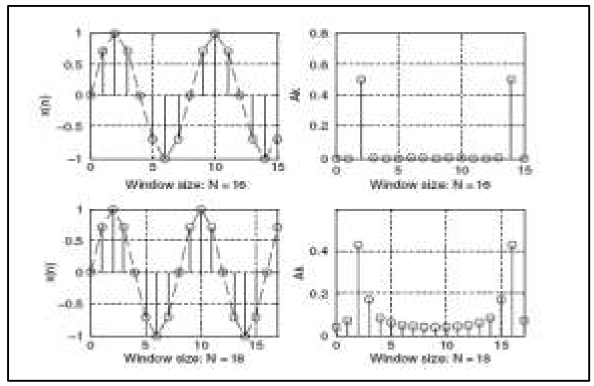
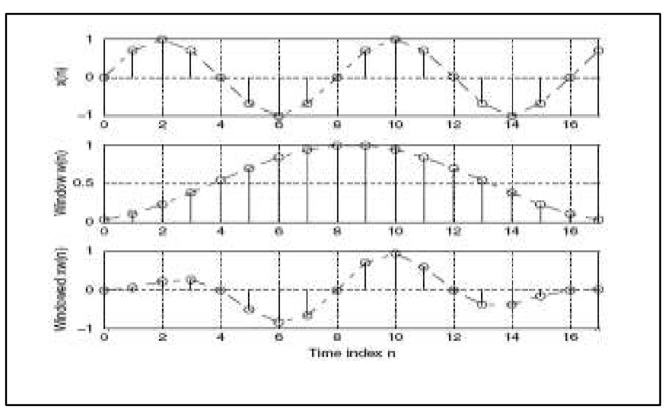


Fig. 7.2 Signal samples and spectra without and with spectral leakage.

The amount of spectral leakage shown in the second plot is due to amplitude discontinuity in time domain. The bigger the discontinuity, the more is the leakage. To reduce the effect of spectral leakage, a window function can be used whose amplitude tapers smoothly and gradually toward zero at both ends. Applying the window function w(n) to a data sequence x(n) to obtain a windowed sequence $x_w(n)$ is better illustrated in Fig. 7.3 using :

$$x_w(n) = x(n)w(n), \text{ for } n = 0, 1, \ldots, N-1. \tag{7.1}$$

Fig. 7.3 Illustration of the window operation.



<u>The common window functions are listed as follows:</u>

The rectangular window (no window function):

$$w_R(n) = 1 \qquad 0 \le n \le N-1 \tag{7.2}$$

The triangular window:

$$w_{tri}(n) = 1 - \frac{|2n - N + 1|}{N-1},\ 0 \le n \le N-1 \tag{7.3}$$

The Hamming window:

$$w_{hm}(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right),\ 0 \le n \le N-1 \tag{7.4}$$

The Hanning window:

$$w_{hn}(n) = 0.5 - 0.5\cos\left(\frac{2\pi n}{N-1}\right),\ 0 \le n \le N-1 \tag{7.5}$$

Plots for each window function for a size of 20 samples are shown in Figure 7.4.

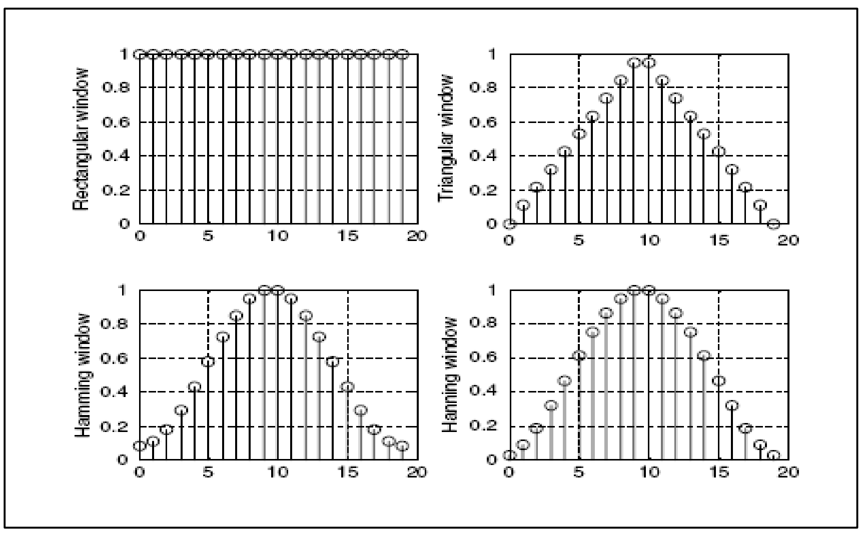


Fig. 7.4 Plots of window sequences

<u>Example (1):</u> Considering the sequence x(0) = 1, x(1) = 2, x(2) = 3, and x(3) = 4, and given f_s = 100 Hz, T = 0.01 seconds, compute the amplitude spectrum, phase spectrum, and power spectrum

Using the triangular window function. .a

Using the Hamming window function. .b

Solution:

Since N = 4, from the triangular window function given in equation (7.3), we have: (a

$w_{tri}(0) = 0$, $w_{tri}(1) = 0.6667$, $w_{tri}(2) = 0.6667$, and $w_{tri}(3) = 0$.

Now, applying eq. (7.1), we have:

$x_w(0) = x(0)\ w_{tri}(0) = 0$. Similarly $x_w(1) = 1.3334$, $x_w(2) = 2$, and $x_w(3) = 0$

Applying DFT equation (6.3) to $x_w(n)$ for K=0, 1, 2, and 3 , we have:

X(0)= 3.3334, X(1) = −2 − j1.3334 , X(2) = 0.6666, and X(3) = − 2 + j 1.3334

Δf= 1 / NT = 25 Hz

Applying equations (6.11), (6.13), and (6.14):

$$A_0 = \frac{1}{4}|X(0)| = 0.8334,\ \varphi_0 = \tan^{-1}\left(\frac{0}{3.3334}\right) = 0^0,$$

$$P_0 = \frac{1}{4^2}|X(0)|^2 = 0.6954$$

K	A_K	Φ_K in degree	P_K
1	0.6009	− 146.31	0.3611
2	0.1667	0	0.0278
3	0.6009	146.31	0.3611

b. Since N = 4, from the Hamming window function given in eq. (7.4), we have: $w_{hm}(0) = 0.08$, $w_{hm}(1) = 0.77$, $w_{hm}(2) = 0.77$, and $w_{hm}(3) = 0.08$. The windowed sequence is computed using eq. (7.1) as:

$x_w(0) = x(0)\ w_{hm}(0) = 0.08$, $x_w(1) = 1.54$, $x_w(2) = 2.31$, and $x_w(3) = 0.32$

Applying DFT equation (6.3) to $x_w(n)$ for K=0, 1, 2, and 3 , we have:

X(0)= 4.25, X(1) = −2.23 − j1.22 , X(2) = 0.53, and X(3) = − 2.23 + j 1.22

Δf= 1 / NT = 25 Hz

Applying equations (6.11), (6.13), and (6.14):

$$A_0 = \frac{1}{4}|X(0)| = 1.0625,\ \varphi_0 = \tan^{-1}\left(\frac{0}{4.25}\right) = 0^0,$$

$$P_0 = \frac{1}{4^2}|X(0)|^2 = 1.1289$$

K	A_K	Φ_K in degree	P_K
1	0.6355	−151.32	0.4308
2	0.1325	0	0.0176
3	0.6355	151.32	0.4308

7.2 Application to Speech Spectral Estimation

The following plots show the comparisons of amplitude spectral estimation for speech data with 2,001 samples and a sampling rate of 8,000 Hz using the rectangular window (no window) function and the Hamming window function. As demonstrated in Fig. 7.5 (two-sided spectrum) and Fig. 7.6 (one-sided spectrum), there is little difference between the amplitude spectrum using the Hamming window function and the spectrum without using the window function. This is due to the fact that when the data length of the sequence (e.g., 2,001 samples) increases, the frequency resolution will be improved and spectral leakage will become less significant. However, when data length is short, reduction of spectral leakage using a window function will come to be prominent.

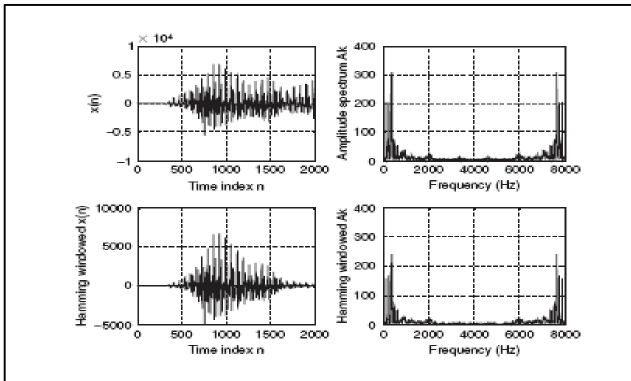


Fig. 7.5 Comparison of a spectrum without using a window function and a spectrum using the Hamming window for speech data.

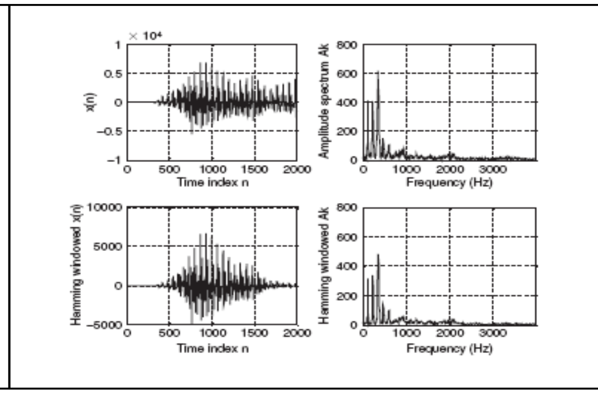


Fig. 7.6 Comparison of a one-sided spectrum without using a window function and a one-sided spectrum using the Hamming window for speech data.

7.3 Fast Fourier Transform

FFT is a very efficient algorithm in computing DFT coefficients and can reduce a very large amount of computational complexity (multiplications).

Consider the digital sequence x(n) consisting of 2^m samples, where m is a positive integer—the number of samples of the digital sequence x(n) is a power of 2, N = 2, 4, 8, 16, etc. If x(n) does not contain 2^m samples, then *we simply append it with zeros* until the number of the appended sequence is equal to an integer of a power of 2 data points.

The number of points $N = 2^m$, where the stages $m = \log_2 N$.

In this section, we focus on two formats. One is called the decimation in- frequency algorithm, while the other is the decimation-in-time algorithm. They are referred to as the radix-2 FFT algorithms.

7.3.1 Method of Decimation-in-Frequency (Reduced DIF FFT)

Beginning with the definition of DFT :

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \text{ for } k = 0,\ 1, \ldots,\ N-1, \tag{7.6}$$

Where, $W_N = e^{-j\pi / N}$ is the twiddle factor, and N = 0, 2, 4, 8, 16, …..Equation (7.6) can be expanded as:

$$X(k) = x(0) + x(1) W_N^k + \ldots + x(N-1) W_N^{k(N-1)}. \tag{7.7}$$

If we split equation (7.7):

$$X(k) = x(0) + x(1) W_N^k + \ldots + x\left(\frac{N}{2} - 1\right) W_N^{k(N/2-1)} + x\left(\frac{N}{2}\right) W^{kN/2} + \ldots + x(N-1) W_N^{k(N-1)} \tag{7.8}$$

Then we can rewrite as a sum of the following two parts:

$$X(k) = \sum_{n=0}^{(N/2)-1} x(n) W_N^{kn} + \sum_{n=N/2}^{N-1} x(n) W_N^{kn}. \tag{7.9}$$

Modifying the second term in Equation (7.9) yields:

$$X(k) = \sum_{n=0}^{(N/2)-1} x(n) W_N^{kn} + W_N^{(N/2)k} \sum_{n=0}^{(N/2)-1} x\left(n + \frac{N}{2}\right) W_N^{kn}. \tag{7.10}$$

Recall $W_N^{N/2} = e^{-j\frac{2\pi(N/2)}{N}} = e^{-j\pi} = -1$; then we have

$$X(k) = \sum_{n=0}^{(N/2)-1} \left(x(n) + (-1)^k x\left(n + \frac{N}{2}\right)\right) W_N^{kn}. \tag{7.11}$$

Now letting k = 2m as an even number achieves:

$$X(2m) = \sum_{n=0}^{(N/2)-1} \left(x(n) + x\left(n + \frac{N}{2}\right)\right) W_N^{2mn}, \tag{7.12}$$

While substituting k = 2m + 1 as an odd number yields:

$$X(2m+1) = \sum_{n=0}^{(N/2)-1} \left(x(n) - x\left(n + \frac{N}{2}\right) \right) W_N^n W_N^{2mn}. \tag{7.13}$$

Using the fact that $W_N^2 = e^{-j\frac{2\pi \times 2}{N}} = e^{-j\frac{2\pi}{(N/2)}} = W_{N/2}$, it follows that

$$X(2m) = \sum_{n=0}^{(N/2)-1} a(n) W_{N/2}^{mn} = DFT\{a(n)\ with\ (N/2)\ \text{points}\} \tag{7.14}$$

$$X(2m+1) = \sum_{n=0}^{(N/2)-1} b(n) W_N^n W_{N/2}^{mn} = DFT\{b(n) W_N^n\ with\ (N/2)\ \text{points}\}, \tag{7.15}$$

Where, a(n) and b(n) are introduced and expressed as:

$$a(n) = x(n) + x\left(n + \frac{N}{2}\right), \text{ for } n = 0,1 \ldots, \frac{N}{2} - 1$$

$$b(n) = x(n) - x\left(n + \frac{N}{2}\right), \text{ for } n = 0,1, \ldots, \frac{N}{2} - 1. \tag{7.16}$$

$$DFT\{x(n)\ with\ N\ \text{points}\} = \begin{cases} DFT\{a(n)\ with\ (N/2)\ \text{points}\} \\ DFT\{b(n) W_N^n\ with\ (N/2)\ \text{points}\} \end{cases} \tag{7.17}$$

Figure 7.7(a) illustrates the block diagram of N-point DIF FFT. Fig. 7.7(b) illustrates **reduced** DIF FFT computation for the eight-point DFT, where there are 12 complex multiplications as compared with the eight-point DFT with 64 complex multiplications. For a data length of N, the number of complex multiplications for DFT and FFT, respectively, are determined by:

Complex multiplications of DFT = N^2, and (7.18a)

Complex multiplications of FFT (***With Reduction***) = $(N / 2)\log_2(N)$ (7.18b)

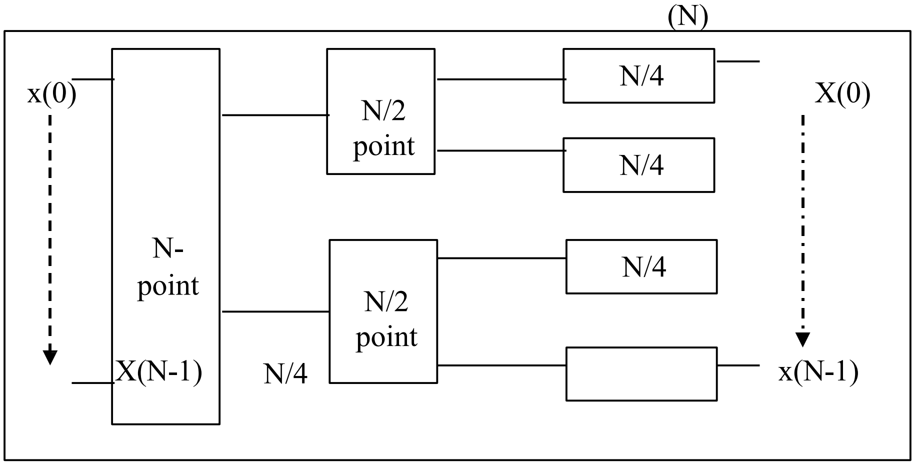


Fig. 7.7(a) Block diagram of DIF FFT

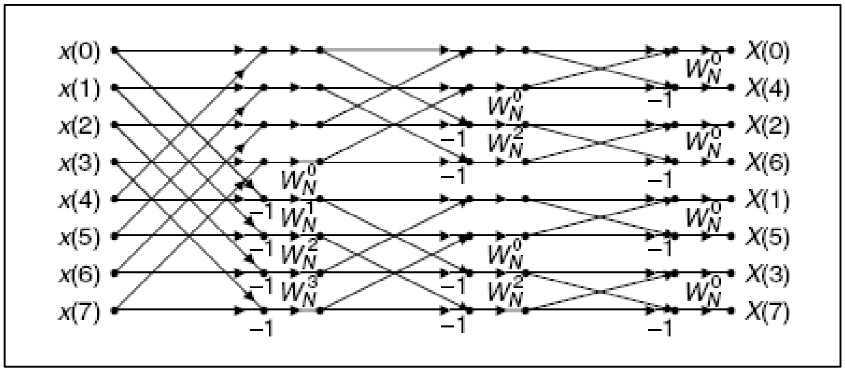
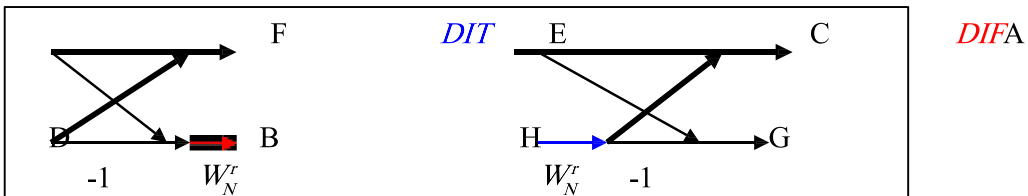


Fig. 7.7(b) The eight-point FFT (total twelve multiplications).

Reduced DIF FFT

Note: The input sequence is in normal order index and the output frequency bin number is in reversal bits order. The *Butterfly structure* for DIF FFT and DIT FFT is shown below:



C = A+B, D = (A-B) W_N^r ***(7.19a)***, F= E + W_N^rG, H= E - W_N^rG ***(7.19b)***

The inverse FFT is defined as:

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k) W_N^{-kn} = \frac{1}{N}\sum_{k=0}^{N-1} X(k)\tilde{W}_N^{kn}, \text{ for } k = 0,\ 1, \ldots,\ N-1. \tag{7.20}$$

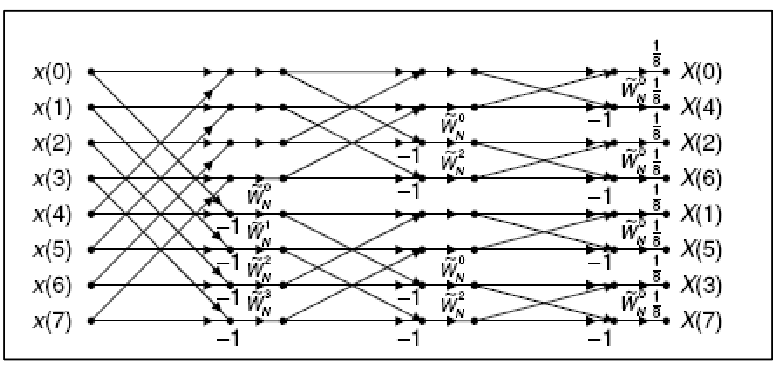


Fig. 7.8 Block diagram for the inverse of eight-point FFT.

Reduced DIF IFFT

, and the sum is W_N multiplied by a factor of $\tilde{W}_N = W_N^{-1}$ 1/N. Hence, the is changed to beThe twiddle factor inverse FFT block diagram is achieved as shown in Fig. 7.8

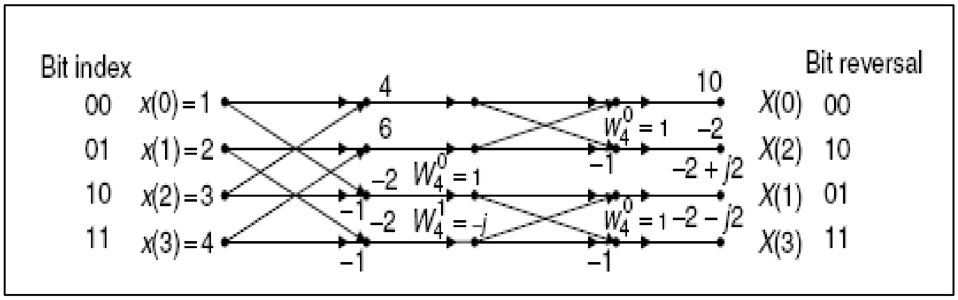
Example (2): Given a sequence x(n) for 0 ≤ n ≤ 3, where x(0) = 1, x(1) = 2, x(2) = 3, and x(3) = 4,

Evaluate its DFT X(k) using the decimation-in-frequency FFT method. .a

Determine the number of complex multiplications. .b

Solution:

$$W_4^0 = e^{-j\frac{2\pi}{4}(0)} = 1 \quad and \quad W_4^1 = e^{-j\frac{2\pi}{4}(1)} = -j$$



b) The number of complex multiplications is four, which can also be determined from eq. (7.18b), where N=4

7.3.2 Method of Decimation-in-Time (Reduced DIT FFT):

In this method, we split the input sequence x(n) into the even indexed x(2m) and x(2m + 1), each with N data points. Then Equation (7.6) becomes:

$$X(k) = \sum_{m=0}^{(N/2)-1} x(2m)W_N^{2mk} + \sum_{m=0}^{(N/2)-1} x(2m+1)W_N^k W_N^{2mk},$$
$$\text{for } k = 0, 1, \ldots, N-1. \qquad (7.21)$$

Using $W_N^2 = W_{N/2}$, it follows that:

$$X(k) = \sum_{m=0}^{(N/2)-1} x(2m)W_{N/2}^{mk} + W_N^k \sum_{m=0}^{(N/2)-1} x(2m+1)W_{N/2}^{mk},$$
$$\text{for } k = 0, 1, \ldots, N-1. \qquad (7.22)$$

Define new functions as:

$$G(k) = \sum_{m=0}^{(N/2)-1} x(2m)W_{N/2}^{mk} = DFT\{x(2m)\ with\ (N/2)\ \text{points}\}$$

$$H(k) = \sum_{m=0}^{(N/2)-1} x(2m+1)W_{N/2}^{mk} = DFT\{x(2m+1)\ with\ (N/2)\ \text{points}\}.$$

(7.23)

Note that:

$$G(k) = G\left(k + \frac{N}{2}\right), \text{ for } k = 0, 1, \ldots, \frac{N}{2} - 1$$

$$H(k) = H\left(k + \frac{N}{2}\right), \text{ for } k = 0, 1, \ldots, \frac{N}{2} - 1. \tag{7.24}$$

Substituting Equations (7.24) into Equation (7.22) yields the first half frequency bins

$$X(k) = G(k) + W_N^k H(k), \text{ for } k = 0, 1, \ldots, \frac{N}{2} - 1. \tag{7.25}$$

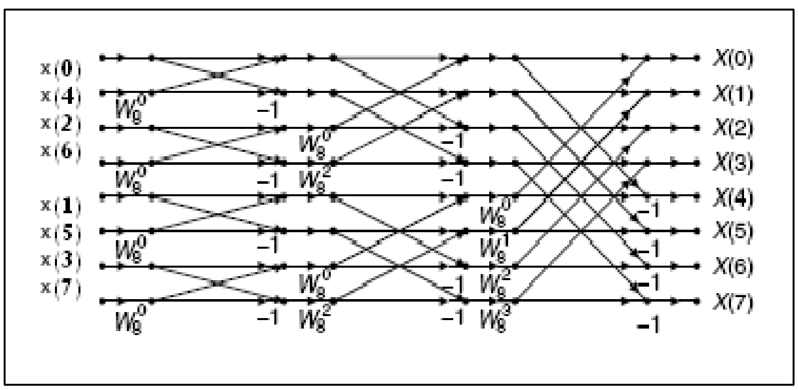
Considering the following fact and using Equations (7.24):

$$W_N^{(N/2+k)} = -W_N^k. \tag{7.26}$$

Then the second half of frequency bins can be computed as follows:

$$X\left(\frac{N}{2} + k\right) = G(k) - W_N^k H(k), \text{ for } k = 0, 1, \ldots, \frac{N}{2} - 1. \tag{7.27}$$

The block diagram for the eight-point DIT FFT algorithm is illustrated in Fig.. 7.9



complex (twelve decimation-in-timealgorithm usingFFTeight-pointFig.7.9 The multiplications). Reduced DIT FFT

The index for each input sequence element can be achieved by bit reversal of the frequency index in a sequential order. Similar to the method of decimation-in-frequency, after we change in Fig. 7.9 $\tilde{W}_N$ and multiply the output sequence by a factor of 1/N, we derive the inverse FFT W_N to block diagram for the eight-point inverse FFT in Fig. 7.10.

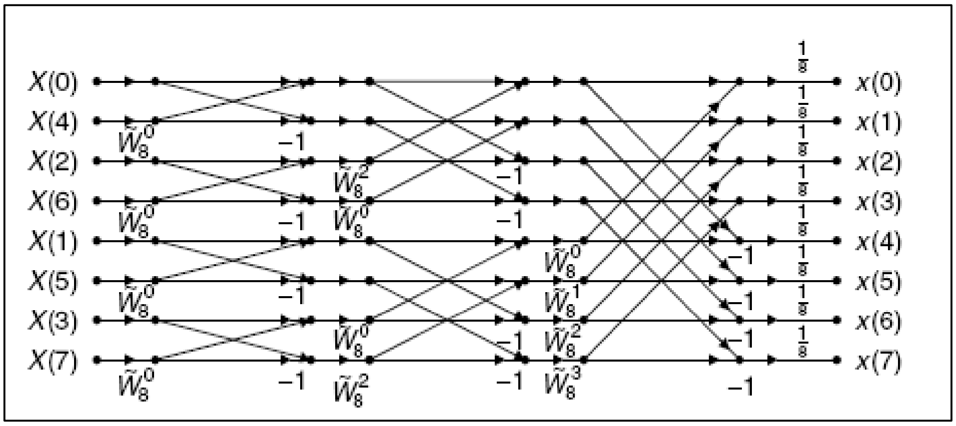
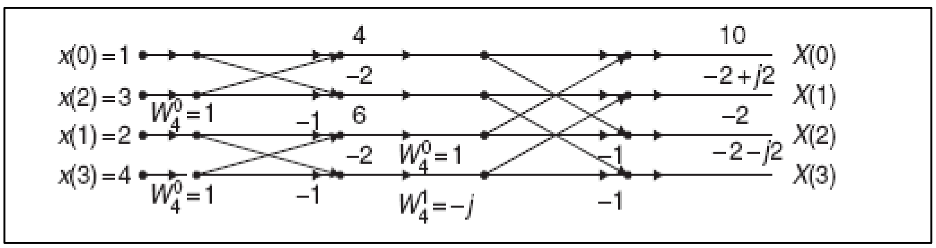


Fig. 7.10 The eight-point IFFT using decimation-in-time (Reduced method).

Example(3): Given a sequence x(n) for $0 \le n \le 3$, where x(0) = 1, x(1) = 2, x(2) = 3, and x(3) = 4. Evaluate its DFT X(k) using the decimation-in-time FFT method.

Solution:



bit indexed Bit reversal

H.W Find DFT of the following sequence [1 -1 -1 -1 1 1 1 -1], using:

Reduced DIT FFT (a

Reduced DIF FFT (b

$2 - j(\sqrt{2} - 2)\ 4 \quad 2 + j(\sqrt{2} - 2)\ 2 - j2 \quad 2 + j(\sqrt{2} + 2) - \quad Ans: [0$

$] \quad 2 - j(\sqrt{2} + 2) - \quad 2 + j2$

7.4 Properties of DFT for real x(n):

$$X(K) = X^*(N - K)$$

$$\mathrm{Re}\{X(K)\} = \mathrm{Re}\{X(N - K)\}$$

$$\mathrm{Im}\{X(K)\} = -\mathrm{Im}\{X(N - K)\}$$

— $$x(n) = \frac{1}{N} FFT[X^*(K)]^* \quad (7.26)$$

$K \rightarrow$	$W_K = K.\,2\pi / N \rightarrow$	$\Omega_K = K.\,2\pi / NT$
frequency index	*digital frequency* (*rad*)	*analog frequency* (*rad*/sec)

* means complex conjugate

For N even:

$$x(n)=\frac{X(0)}{N}+\frac{2}{N}\sum_{K=1}^{(N/2)-1}\left\{X_R(K)\cos\left(\frac{2\pi}{N}nK\right)-X_I(K)\sin\left(\frac{2\pi}{N}nK\right)\right\}+\frac{X(N/2)}{N}\cos\pi n$$

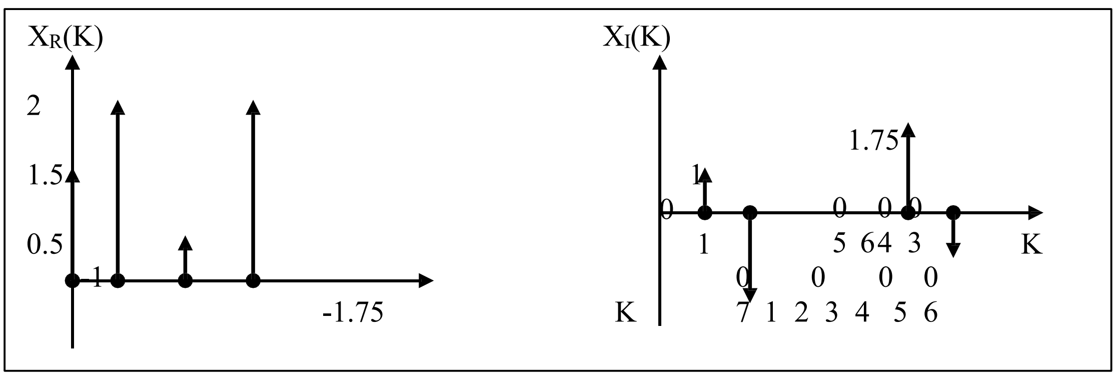
(7.27a)

For N odd:

$$x(n)=\frac{X(0)}{N}+\frac{2}{N}\sum_{K=1}^{(N-1)/2}\left\{X_R(K)\cos\left(\frac{2\pi}{N}nK\right)-X_I(K)\sin\left(\frac{2\pi}{N}nK\right)\right\} \quad (7.27b)$$

x(n)	X(K)
Real	Real part is even, imaginary part is odd
Real and even	Real and even
Real and odd	Imaginary and odd

Example (4): Find x(n) for $X_R(K)$ and $X_I(K)$, then find $x_a(t)$ if T = 0.1 sec.



N=8 , then using eq.(7.27a):

$$x(n)=\frac{1.5}{8}+\frac{2}{8}\left\{-(1)\sin\left(\frac{2\pi}{8}1n\right)+(2)\cos\left(\frac{2\pi}{8}2n\right)-(-1.75)\sin\left(\frac{2\pi}{8}2n\right)\right\}+\frac{0.5}{8}\cos\pi n$$

$$for\ K=1,2,\ldots\ldots\frac{N}{2}-1=3$$

$$n=\frac{t}{T},\quad t=nT,\quad \Omega_k=W_k/T=2\pi K/NT$$

$$x_a(t)=\frac{1.5}{8}+\frac{2}{8}\left\{-(1)\sin\left(\frac{2\pi}{0.1}\frac{1}{8}t\right)+(2)\cos\left(\frac{2\pi}{0.1}\frac{2}{8}t\right)-(-1.75)\sin\left(\frac{2\pi}{0.1}\frac{2}{8}t\right)\right\}+\frac{0.5}{8}\cos\pi n$$

$$for\ T=0.1\,\text{sec}.$$

$$x_a(t)=0.1875-0.25\sin 2.5\pi t+0.5\cos 5\pi t+0.4375\sin 5\pi t+0.0625\cos 10\pi t$$

7.5 DFT and Fourier transform relations:

The Fourier transform $X(e^{jW})$ of an x(n) is given for all W:

$$X(e^{jW}) = \sum_{-\infty}^{\infty} x(n)\, e^{-jWn} = \sum_{n=0}^{N-1} x(n)\, e^{-jWn}, \quad n = 0,1,2,...N-1 \tag{7.28}$$

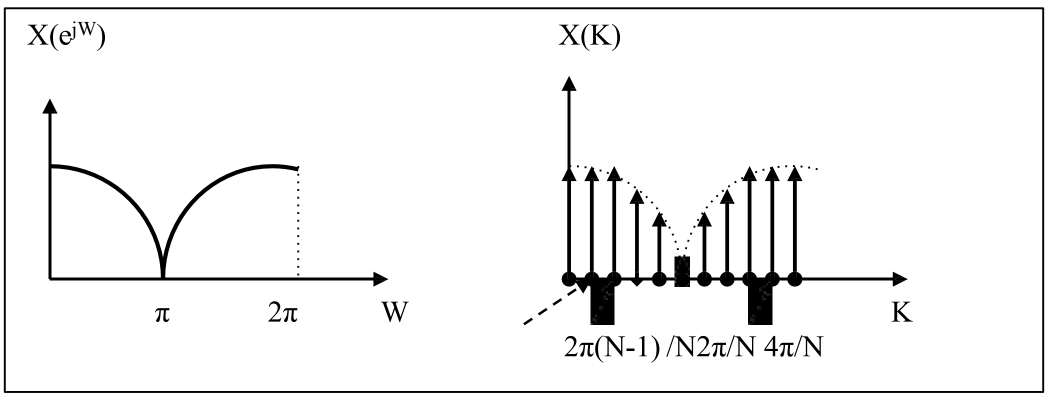
From eq. (7.28), $X(e^{jW})$ is a ***continuous*** function of W.

The DFT (N-point) of an x(n) is given by:

$$X(K) = \sum_{n=0}^{N-1} x(n)\, e^{-j2\pi Kn/N} \qquad K = 0,1,2....N-1, \tag{7.29}$$

Comparing eq.(7.28) and eq.(7.29), the DFT of x(n) is the sampled version of the Fourier transform sequence as shown below

$$X(K) = X(e^{jW})\Big|_{W=(2\pi K/N)}, \quad K = 0,1,2,...N-1 \tag{7.30}$$



1. Introduction:

Let us review analog filter design using lowpass prototype transformation. This method converts the analog lowpass filter with a cutoff frequency of 1 radian per second, called the lowpass prototype, into practical analog lowpass, highpass, bandpass, and bandstop filters with their frequency specifications.

2. Butterworth Filters

8.2.1 Butterworth low-pass filter (LPF)

A typical frequency response for a Butterworth low-pass filter of order n is shown in Fig. 8.1.

$$|H_n(j\Omega)|^2 = \frac{1}{1+(\Omega/\Omega_c)^{2n}} \qquad (8.1)$$

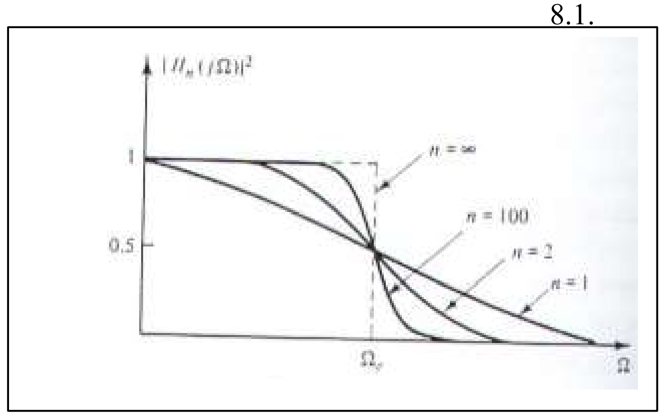


Fig.8.1 Butterworth LPF c/cs

Properties:

$$|H_n(j\Omega)|^2_{\Omega=0} = 1 \quad \text{for all } n$$

$$|H_n(j\Omega)|^2_{\Omega=\Omega_c} = \frac{1}{2} \quad \text{for all finite } n$$

$$|H_n(j\Omega)|_{\Omega=\Omega_c} = 0.707 \quad (-3.0103\ dB)$$

$|H_n(j\Omega)|^2$ is monotonically decreasing function of Ω, it is also called maximally flat at the origin since all derivatives exist and are zero. As n → ∞ , we get ideal response.

The *normalized* LP Butterworth is obtained when:

$\Omega_c = 1$ rad / sec.

Substituting S = j Ω in eq. (8.1), and rearrange to get the LP Butterworth poles, then:

$$S = (-1)^{[(n+1)/2n]}$$

For n odd, $S_K = 1 \angle k\pi / n$, $k = 0, 1, 2, \ldots, 2n-1$ (8.2a)

For n even, $S_K = 1 \angle (k\pi/n) + (\pi/2n)$, $k = 0, 1, 2, \ldots, 2n-1$ (8.2.b)

For stable and causal filter:

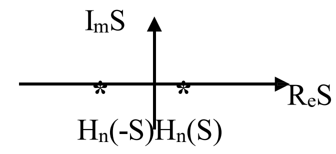
$$H_n(S) = \frac{1}{\prod_{LHP\ poles}(S - S_k)} = \frac{1}{B_n(S)} \qquad (8.3)$$

$B_n(S)$: Butterworth polynomial of order n (see Table (1)).

LHP: Left half plane.

Example(1): Find the transfer function $H_1(S)$ for the normalized Butterworth filter of order one.

Solution: applying eq.(8.2a), where n=1, k = 0,1



$$S_0 = 1 \angle 0 = H_n(-S)$$

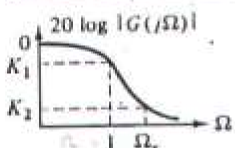
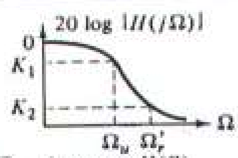
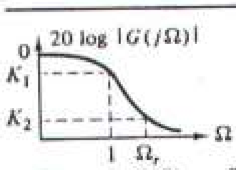
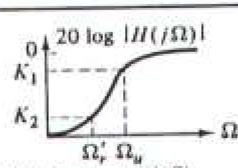
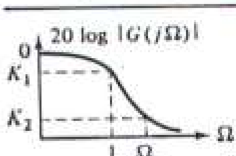
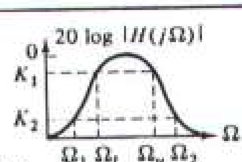
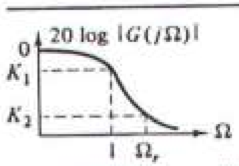
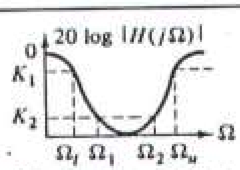
$S_1 = 1 \angle \pi = H_n(S)$. Using eq. (8.3) and taking **LHP poles** S_1:

$$H_1(S) = \frac{1}{S+1} = \frac{1}{S-(-1)}$$

8.2.2 Analog- to analog transformation

To obtain Butterworth filters with cutoff frequencies other than 1 rad /sec. It is convenient to use 1 rad /sec. Butterworth filters as prototypes and apply analog-to-analog transformation (see Table (2)). *The transformational method is not limited in its application to Butterworth filters.*

TABLE 2 ANALOG-TO-ANALOG THANSFUHMATIUN

Prototype response	Transformed filter response	Design equations
 Low-pass $G(S)$	$S \to S/\Omega_u$  Low-pass $H(S)$	Forward: $\Omega_r' = \Omega_r\Omega_u$ Backward: $\Omega_r = \Omega_r'/\Omega_u$
 Low-pass $G(S)$	$S \to \Omega_u/S$  High-pass $H(S)$	Forward: $\Omega_r' = \Omega_u/\Omega_r$ Backward: $\Omega_r = \Omega_u/\Omega_r'$
 Low-pass $G(S)$	$S \to \dfrac{S^2 + \Omega_l\Omega_u}{S(\Omega_u - \Omega_l)}$  Bandpass $H(S)$	Forward: $\Omega_{av} = (\Omega_u - \Omega_l)/2$ $\Omega_1 = (\Omega_r^2\Omega_{av}^2 + \Omega_l\Omega_u)^{1/2} - \Omega_{av}\Omega_r$ $\Omega_2 = (\Omega_r^2\Omega_{av}^2 + \Omega_l\Omega_u)^{1/2} + \Omega_{av}\Omega_r$ Backward: $\Omega_r = \min\{\lvert A\rvert, \lvert B\rvert\}$ $A = (-\Omega_1^2 + \Omega_l\Omega_u)/[\Omega_1(\Omega_u - \Omega_l)]$ $B = (+\Omega_2^2 - \Omega_l\Omega_u)/[\Omega_2(\Omega_u - \Omega_l)]$
 Low-pass $G(S)$	$S \to \dfrac{S(\Omega_u - \Omega_l)}{S^2 + \Omega_l\Omega_u}$  Bandstop $H(S)$	Forward: $\Omega_{av} = (\Omega_u - \Omega_l)/2$ $\Omega_1 = [(\Omega_{av}/\Omega_r)^2 + \Omega_l\Omega_u]^{1/2} - \Omega_{av}/\Omega_r$ $\Omega_2 = [(\Omega_{av}/\Omega_r)^2 + \Omega_l\Omega_u]^{1/2} + \Omega_{av}/\Omega_r$ Backward: $\Omega_r = \min\{\lvert A\rvert, \lvert B\rvert\}$ $A = \Omega_1(\Omega_u - \Omega_l)/[-\Omega_1^2 + \Omega_l\Omega_u]$ $B = \Omega_2(\Omega_u - \Omega_l)/[\pm\Omega_2^2 \mp \Omega_l\Omega_u]$

8.2.3 Design Equations of Butterworth Filters:

A **Butterworth LPF** Filter of order n is given by the following equation:

$$n = \left\lceil \frac{\log_{10} \{ (10^{-0.1 k_1} - 1) / (10^{-0.1 k_2} - 1) \}}{2 \log_{10} (1 / \Omega_r)} \right\rceil \qquad (8.4)$$

Here, $1/\Omega_r = \Omega_u / \Omega'_r$, see Table (2).

Where, k_1, k_2, Ω_u, and Ω'_r are the pass-band gain and stop-band attenuation with their relative frequencies respectively(see Table (2)).

To satisfy our requirement at Ω_u exactly, then:

$$\Omega_c = \Omega_u / (10^{-0.1 k_1} - 1)^{1/2n} \qquad (8.5a)$$

To satisfy our requirement at Ω'_r exactly, then:

$$\Omega_c = \Omega'_r / (10^{-0.1 k_2} - 1)^{1/2n} \qquad (8.5b)$$

Ω_c is the cutoff frequency at – 3dB

Example (2): design an analog Butterworth LPF that has a – 2 dB butter cutoff frequency of 20 rad/sec. and at least 10 dB of attenuation at 30 rad/sec**.**

Solution: Applying eq. (8.4), where k_1= -2 dB, k_2 = -10 dB, Ω_u = 20 rad/sec., and Ω'_r = 30 rad/sec

$$n = \left\lceil \frac{\log_{10} \{ (10^{0.2} - 1) / (10^{1} - 1) \}}{2 \log_{10} (20 / 30)} \right\rceil = \lceil 3.3709 \rceil = 4$$

To satisfy our requirement at Ω_u exactly, then:

$$\Omega_c = 20 / (10^{0.2} - 1)^{1/8} = 21.3836 \ rad/\sec$$

From Table (1) of *normalized* Butterworth LPF (Ω_c =1 rad/ sec) with n = 4 :

$$H_4(S) = \frac{1}{(S^2 + 0.76536 S + 1) (S^2 + 1.84776 S + 1)}$$

Using Table (2) and applying LP → LP transformation, S→ S / 21.3836, and rearranging:

$$H(S) = \frac{0.20921 \times 10^6}{(S^2 + 16.3686 S + 457.394) (S^2 + 39.5176 S + 457.394)}$$

For **Butterworth HPF**:

1- Put $1/\Omega_r = \Omega'_r / \Omega_u$ in equation (8.4), and find its order n .(see Table(2))

2- Use Table (1) to find the normalized Butterworth LPF equation with order n.

3- Apply LP → HP transformation, $S \to \Omega_c / S$, and rearrange the equation obtained in step 2.

For **Butterworth BPF**:

1- Calculate $\Omega_r = \min \{ |A|, |B| \}$ using equations given in Table (2). Find the filter order using eq.(8.4)

2- Use Table (1) to find the normalized Butterworth LPF equation with order n.

3- Apply LP → BP transformation, $S \to \frac{S^2 + \Omega_l \Omega_u}{S(\Omega_u - \Omega_l)}$, and rearrange the equation obtained in step 2

For **Butterworth BSF:**

Refer to Table (2) to see the variables.

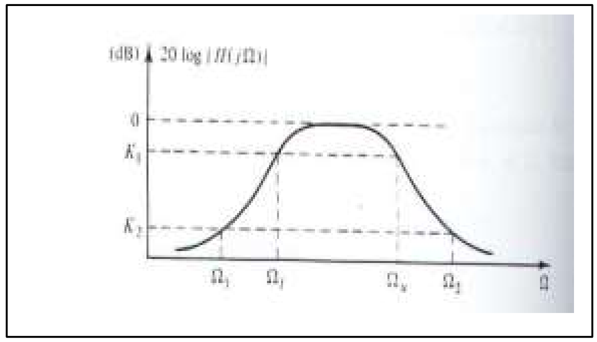


Fig. 8.2 Butterworth BPF

Example (3): Design an analog Butterworth BPF with the following c/cs:

A – 3.0103 dB upper and lower cutoff frequencies of 50 Hz and 20 KHz.

A stop-band attenuation of at least 20 dB at 20 Hz and 45 kHz.

Solution:

$\Omega_1 = 2\pi(20) = 125.663$ rad / sec.

$\Omega_2 = 2\pi(45 \times 10^3) = 2.82743 \times 10^5$ rad / sec.

$\Omega_u = 2\pi(20 \times 10^3) = 1.25663 \times 10^5$ rad / sec.

$\Omega_l = 2\pi(50) = 314.159$ rad / sec

Calculate $\Omega_r = \min \{ |A|, |B| \} = \min(|2.5053|, |2.2545|) = 2.2545$ by using equations given in Table (2) . Apply eq. (8.4) to find:

$$n = \lceil 2.829 \rceil = 3$$

From Table (1) of *normalized* Butterworth LPF ($\Omega_c = 1$ rad/ sec) with n = 3:

$$H_3(S) = \frac{1}{S^3 + 2S^2 + 2S + 1}$$

Apply LP → BP transformation by substituting $S \to \frac{S^2 + \Omega_l \Omega_u}{S(\Omega_u - \Omega_l)} = \frac{S^2 + 3.94784 \times 10^7}{S(1.25349 \times 10^5)}$, in the above equation and rearrange it to obtain H_{BPF} (as H.W)

8.3 Chebyshev Filters:

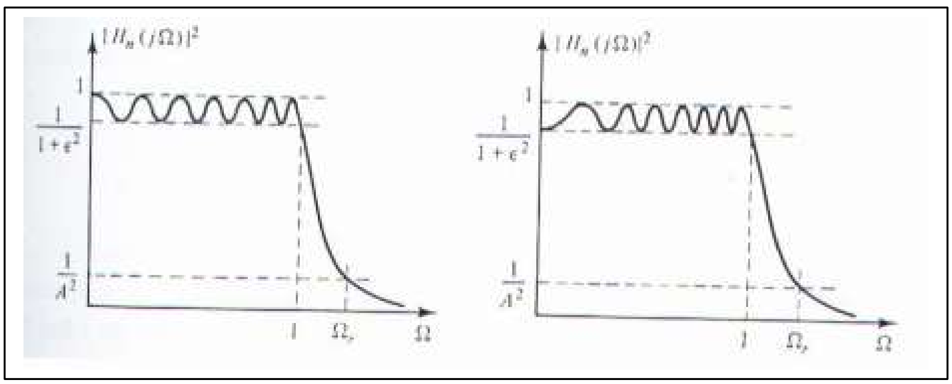
There are two types of Chebyshev Filters:

1- One containing a ripple in the pass-band (type 1).

2- One containing a ripple in the stop-band (type 2).

$$|H_n(j\Omega)|^2 = \frac{1}{1+\varepsilon^2 T_n^2(\Omega)} \tag{8.6}$$

$T_n(\Omega)$ is the nth order Chebyshev polynomial where $T_0(x)=1$, and $T_1(x)=x$ as listed in Table (3). ε^2 is a parameter chosen to provide the proper pass-band ripple. Fig. (8.3) shows *normalized* Chebyshev Filters of both types.



n even ***n odd***

Fig.(8.3) Normalized Chebyshev filters of type 1 for (n odd), and (n even)

8.3.1 Design Equations of Chebyshev Filters:

$$n = \frac{\log_{10}\left[g+\sqrt{g^2-1}\right]}{\log_{10}\left[\Omega_r+\sqrt{\Omega_r^2-1}\right]} \tag{8.7}$$

$$= \textit{stop band attenuation (dB)} = 20\log_{10}[1/A] \tag{8.8a}$$

$$g = \left[(A^2-1)/\varepsilon^2\right]^{1/2} \tag{8.8b}$$

$$|H_n(S)| = \frac{K}{\prod\limits_{\substack{LPF\\poles}}(S-S_K)} = \frac{K}{V_n(S)}$$

$$K = V_n(0) = b_0 \quad n\ odd$$
$$K = V_n(0)/\sqrt{1+\varepsilon^2} \quad n\ even \tag{8.9}$$

Table (4) gives $V_n(S)$ for n =1 to n =10 and ε corresponding to 0.5, 1, 2, and 3 dB ripples.

Table (5) gives the zeros {poles of $H_n(S)$ } for the same n and ε.

8.3.2 Design steps of Chebeshev LPF, HPF, BPF, and BSF :

1. Use the *backward design equations* from Table (2) to obtain normalized LPF requirements (Ω_r).
2. Calculate A using eq. (8.8a)
3. Calculate g from eq. (8.8b), then apply eq.(8.7) to find the order n.
4. Use Table (4) and Table (5) to find the Chebeshev Filter equation with order n.
5. Apply LP → LP or HP or BP or BS transformation (Table (2)) and rearrange the equation obtained in step 4.

Example (4): Design a Chebshev filter to satisfy the following specifications:

1. Acceptable pass-band ripple of 2dB
2. Cutoff frequency of 40 rad/sec.
3. stop-band attenuation of 20 dB or more at 52 rad/sec.

Solution: From Table (2)

$\Omega_r = \Omega'_r / \Omega_u = 52/40 = 1.3$ rad/sec.

$= -20 \; 20 \log_{10} [1/A] \; 1722$

$A = 10$, using ε = 2 dB = 0.76478 (see Table (4) and Table (5))

Applying eq. (8.8b), then $g = 13.01$

$$n = \left\lceil \frac{\log_{10}\left[13.01 + \sqrt{(13.01)^2 - 1}\right]}{\log_{10}\left[1.3 + \sqrt{(1.3)^2 - 1}\right]} \right\rceil = \lceil 4.3 \rceil = 5 \quad \text{, n odd}$$

From Table (4) with n = 5 and ε = 2 dB = 0.76478

$$H_5(S) = \frac{0.08172}{S^5 + 0.70646\,S^4 + 1.499\,S^3 + 0.6934\,S^2 + 0.459349\,S + 0.08172}$$

Using poles from Table (5):

$$H_5(S) = \frac{0.08172}{(S + 0.218303)(S^2 + 0.134922\,S + 0.95215)(S^2 + 0.35323\,S + 0.393115)}$$

Using Table (2) and applying LP → LP transformation, S→ S / 40, and rearranging the above equation:

$$H_{LPF}(S) = \frac{8.366 \times 10^6}{(S + 8.73212)\,(S^2 + 5.3969\,S + 1523.44)\,(S^2 + 14.1292\,S + 628.984)}$$

Notes:

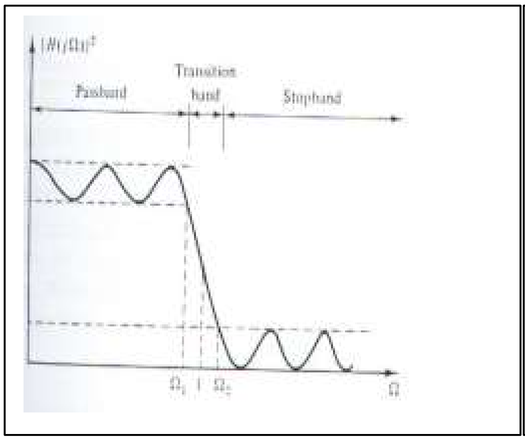
1. Butterworth or maximally flat amplitude; as the order (n) is increased the response becomes flatter in the pass-band and the attenuation is greater in the stop-band.

Chebshev Filter has a sharper cutoff; i.e., a narrower transition band (best amplitude response) than a Butterworth filter of the same order (n) .2

Chebshev Filter provides poorest phase response (most nonlinear). The Butterworth filter compromise between amplitude and phase (this is one of the reasons for its widespread popularity). .3

<u>8.4 Elliptic Filters:</u>

A LP elliptic filter provides a smaller transition width and is optimum in the sense that no other filter of the same order has a narrower transition width for a given pass-band ripple and stop-band attenuation.



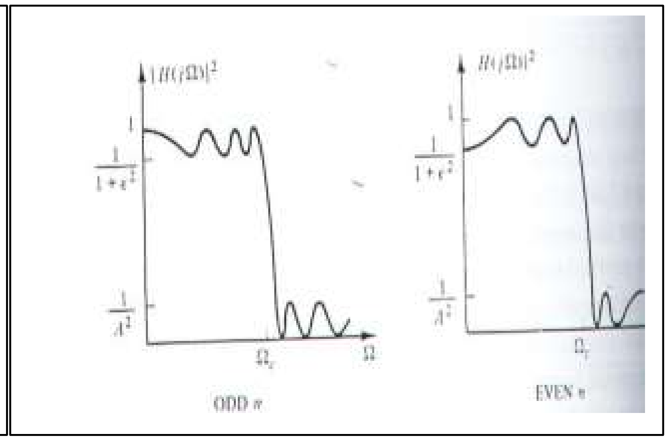


Fig. 8.4(b) elliptic LP filters typesFig. 8.4(a) normalized elliptic LPF

Fig. 8.4 (a) shows a normalized elliptic LPF and Fig. 8.4 (b) shows elliptic LP filters of type 1 (n odd), and type 2 (n even).

<u>**Design steps of Elliptic LPF, HPF, BPF, and BSF**</u> : using Table (6) **.1**

Locate k_1= acceptable pass-band ripple (dB) , and k_2 = stop-band attenuation (dB). .1

Calculate Ω_r using Table(2), pp.55. .2

At Ω_r column, take a value *less than* Ω_r. .3

The filter order (n) is the far left of that row, and the coefficients for the filter are found in all rows corresponding to that (n). .4

According to (n), the normalized elliptic LPF equations are: .5

$$H_n(S) = \frac{H_o}{(S+S_o)} \prod_{i=1}^{(n-1)/2} \frac{S^2+A_{0i}}{S^2+B_{1i}S+B_{oi}} \qquad \text{, n odd} \qquad (8.10\text{ a})$$

$$H_n(S) = H_0 \prod_{i=1}^{n/2} \frac{S^2 + A_{0i}}{S^2 + B_{1i}S + B_{0i}} \qquad (8.10\text{ b}) \qquad \text{, n even}$$

6- Apply LP → LP or HP or BP or BS transformation (Table (2)) and rearrange the equation obtained in step 5.

Notes:

For ***normalized*** elliptic filter, $\Omega_0 = (\Omega_2 \Omega_1)^{0.5} = 1$ = geometric mean, and $\Omega_r = \Omega_2 / \Omega_1$, then $\Omega_1 = (\Omega_r)^{-0.5}$, and $\Omega_2 = (\Omega_r)^{0.5}$

For ***not normalized*** elliptic filter, $\Omega_0 = (\Omega'_1 \Omega'_2)^{0.5}$, $\Omega_1 = \Omega'_1 / \Omega_0$ and $\Omega_2 = \Omega'_2 / \Omega_0$, where $\Omega_0 = \Omega_1$. Then $\Omega_2 / \Omega_1 = \Omega'_2 / \Omega'_1 = \Omega_r$

n (elliptic) ≤ n (chebeshev) ≤ n (Butterworth)

Example (5) : Find the transfer function for an elliptic LPF with − 2 dB cutoff value at 10000 rad/sec., and a stop-band attenuation of 40 dB for all Ω past 14400 rad/sec.

Solution:

$\Omega_0 = (\Omega'_1 \Omega'_2)^{0.5} = \{(14400)(10000)\}^{0.5} = 12000$

$\Omega_1 = \Omega'_1 / \Omega_0 = 10000/12000 = 5/6$ and $\Omega_2 = \Omega'_2 / \Omega_0 = 14400/12000 = 6/5$

$\Omega_2 / \Omega_1 = \Omega'_2 / \Omega'_1 = \Omega_r = 1.44$, $k_1 = -2$ dB, and $k_2 = -40$ dB. From Table (6), n = 4

Applying eq. (8.10 b), Where:

$H_0 = 0.01$, $A_{01} = 7.25202$, $B_{01} = 0.212344$, and $B_{11} = 0.467290$, i = 1

$A_{02} = 1.57676$, $B_{02} = 0.677934$, and $B_{12} = 0.127954$ i = 2

$$H_4(S) = \frac{0.01(S^2 + 7.25202)(S^2 + 1.57676)}{(S^2 + 0.467290\,S + 0.212344)(S^2 + 0.127954\,S + 0.677934)}$$

Apply LP → LP transformation (Table (2)), where Ω_0 = geometric mean = 12000. Substituting S → S / 12000 in the above equation:

$$H_{LPF}(S) = \frac{0.01(S^2 + 1.04429\times10^9)(S^2 + 2.27053\times10^8)}{(S^2 + 5607.48\,S + 30577536)(S^2 + 1535.448\,S + 97622497)}$$

9.1 Introduction:

A discrete time filter takes a discrete time input sequence x(n) and produces a discrete time output sequence y(n).

A special class of a discrete time shift-invariant system can be characterized by a unit sample response h(n), a system function H(Z), or difference equation.

$$\sum_{k=0}^{M} a_k \, y(n-k) = \sum_{k=0}^{N} b_k \, x(n-k) \qquad (9.1)$$

$$H(Z) = \frac{\sum_{k=0}^{M} b_k Z^{-k}}{\sum_{k=0}^{N} a_k Z^{-k}} \qquad (9.2)$$

$$H(e^{jW}) = \frac{\sum_{k=0}^{M} b_k e^{-jWk}}{\sum_{k=0}^{N} a_k e^{-jWk}} \qquad Z = e^{jW}, \qquad (9.3)$$

A filter may be required to have a given frequency response, or specific response to an impulse, step, or ramp, or simulate a continuous analog system. The simulation of analog filter is shown in Fig. (9.1).

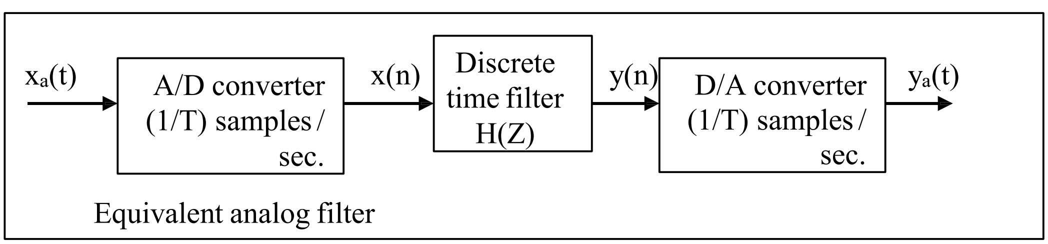


Fig. (9.1) Equivalent analog filter

A/D converter consists of sampler, quantizer, and coder.

D/A converter consists of decoder, sample and hold, and low-pass filter.

Definitions .1

.1 If unit sample response h(n) is of finite duration, the system is said to be a finite impulse response (FIR) system. Eq. (9.1) represents FIR system if $a_0 \neq 0$ and $a_k = 0$ for k=1, 2,..N.

2. If unit sample response h(n) is of infinite duration, the system is said to be an infinite impulse response (IIR) system.

3. IIR filter is usually implemented by recursive realization (is one in which the present value of the output depends on both the input present and or past values), i.e., *with feedback.*

4. FIR filter is usually implemented by either a nonrecursive realization (*without feedback*) or an FFT realization.

9.1.2 A comparison between FIR and IIR filters:

FIR	IIR
1- Finite impulse response h(n) $n_1 \le n \le n_2$	1- Infinite impulse response h(n) $n_1 \le n \le \infty$
2-Complex requires large number of computations	2- Simple, does not require large number of computations
3- Due to large number of computations, it requires large memory	3- Dose not require large memory
4- Always stable because its poles lie at the origin	4- Stable only if its poles lie inside the unit circle of the Z-plane
5- Linear phase characteristics	5- nonlinear phase characteristics

9.2 Infinite Impulse Response (IIR) filter format

An IIR filter is described using the difference equation (9.1) as:

$$y(n) = b_0 x(n) + b_1 x(n-1) + \cdots + b_M x(n-M) - a_1 y(n-1) - \cdots - a_N y(n-N). \quad (9.4)$$

The IIR filter transfer function given in eq.(9.2) as:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}}{1 + a_1 z^{-1} + \cdots + a_N z^{-N}}, \quad (9.5)$$

Example (1): Given the following IIR filter:

$y(n) = 0.2\, x(n) + 0.4\, x(n-1) + 0.5\, y(n-1)$,

Determine the transfer function, nonzero coefficients, and impulse response.

Solution:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{0.2 + 0.4z^{-1}}{1 - 0.5z^{-1}}.$$

$b_0 = 0.2$, $b_1 = 0.4$, and $a_1 = -0.5$.

Using the inverse z-transform and shift theorem, we obtain the impulse response as

$$h(n) = 0.2(0.5)^n u(n) + 0.4(0.5)^{n-1} u(n-1).$$

9.3 Techniques for designing H(Z) for IIR filter:

9.3.1 Design by using numerical solutions of differential equations:

A continuous time linear filter is specified by the following difference equation:

$$\sum_{k=0}^{N} c_k \frac{d^k y_a(t)}{dt^k} = \sum_{k=0}^{M} d_k \frac{d^k x_a(t)}{dt^k} \qquad (9.6)$$

$$H_a(S) = \frac{\sum_{k=0}^{M} d_k S^k}{\sum_{k=0}^{N} c_k S^k} \qquad (9.7)$$

Approximate the derivates using *first backward* differences:

$$\nabla^{(1)}[y(n)] = [y(n) - y(n-1)] / T \qquad (9.8)$$

Higher order backward differences are found by applying the *first backward* difference repeatedly, as follows:

$$\nabla^{(k)}[y(n)] = \nabla^{(1)}[\nabla^{(k-1)}[y(n)]] \qquad (9.9)$$

Using the k^{th} order differences as approximations to the derivatives given in eq. (9.6), we have:

$$\sum_{k=0}^{N} c_k \nabla^{(k)}[y_a(nT)] = \sum_{k=0}^{M} d_k \nabla^{(k)}[x_a(nT)] \qquad (9.10)$$

The Z. Transform of the 1^{st} and k^{th} order differences are given below:

$$Z\{\nabla^{(1)}[y(n)]\} = Y(Z)\{1 - Z^{-1}\} / T \qquad (9.11)$$

$$Z\{\nabla^{(k)}[y(n)]\} = Y(Z)[\{1 - Z^{-1}\} / T]^k \qquad (9.12)$$

Letting $x(n) = x_a(nT)$, and $y(n) = y_a(nT)$. Taking the Z. Transform of eq. (9.10):

$$\sum_{k=0}^{N} c_k Y(Z)[\{1 - Z^{-1}\} / T]^k = \sum_{k=0}^{M} d_k X(Z)[\{1 - Z^{-1}\} / T]^k$$

$$H(Z) = \frac{Y(Z)}{X(Z)} = \frac{\sum_{k=0}^{M} d_k [\{1 - Z^{-1}\} / T]^k}{\sum_{k=0}^{N} c_k [\{1 - Z^{-1}\} / T]^k} \qquad (9.13)$$

Comparing eq. (9.7) and eq. (9.13), we find:

$$H(Z) = H_a(S) \Big|_{S \to \frac{(1-Z^{-1})}{T}} \qquad (9.14)$$

H.W: If $H(S) = \dfrac{1}{(S+1)(S+2)}$ use the numerical solutions of differential equations to obtain H(Z) for, a) T = 1 sec., and b) $f_s = 100$ Hz.

9.3.2 Bilinear transformation (BLT) Design method:

Figure (9.2) illustrates a flow chart of the BLT design used

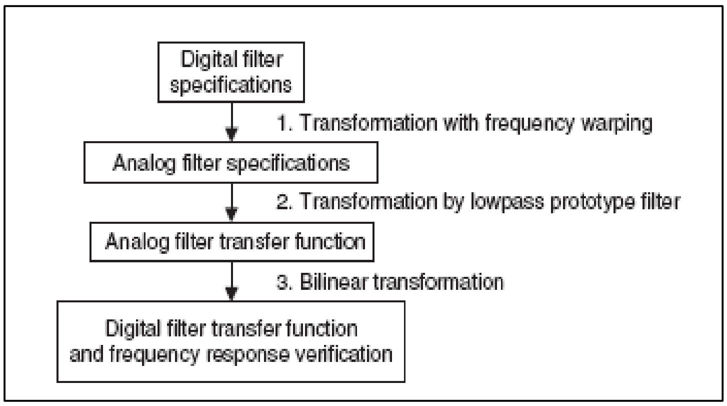


Fig. 9.2 General procedure for IIR filter design using bilinear transformation.

$$H(Z) = H_a(S) \Big|_{S \to \frac{2}{T}\frac{(1-Z^{-1})}{(1+Z^{-1})}} \qquad (9.15)$$

$$\because S = \frac{2}{T}\frac{(1-Z^{-1})}{(1+Z^{-1})}$$

$$j\Omega = \frac{2}{T}\frac{(1-e^{-jW})}{(1+e^{-jW})} \qquad (9.16)$$

$$j\Omega = \frac{2}{T}\frac{e^{-jW/2}}{e^{-jW/2}}\frac{e^{jW/2}-e^{-jW/2}}{e^{jW/2}+e^{-jW/2}}$$

$$\Omega = \frac{2}{T}\tan\left(\frac{W}{2}\right) \quad , rad/\sec \qquad (9.17)$$

$$W = 2\tan^{-1}\left(\frac{\Omega T}{2}\right) \quad , rad \qquad (9.18)$$

As (W/2) becomes smaller, we get more linear characteristics [(W/2) << 1]. If the bilinear transformation is applied to an $H_a(S)$ with critical frequency Ω_c, the digital filter will have critical frequency W_c.

$$W_c = 2\tan^{-1}\left(\frac{\Omega_c T}{2}\right) \qquad (9.19)$$

If the resulting H(Z) is used in an A/D-H(Z)-D/A structure, the equivalent critical frequency becomes:

$$W_c = \Omega_{ceq} T \qquad (9.20)$$

$$\Omega_{ceq} = \frac{2}{T}\tan^{-1}\left(\frac{\Omega_c T}{2}\right) \qquad (9.21)$$

Which will give Ω_c only if $\Omega_c T / 2$ is so small, that $\tan^{-1}(\Omega T / 2) \approx \Omega T / 2$.

In bilinear transformation, the design of digital filter does not depend on the sampling rate (T =1, ***prewarp case***). For a low-pass filter, with $S \to S / \Omega_c$, and applying eq. (9.17), then:

$$S = \frac{2(1-Z^{-1})}{T(1+Z^{-1})\,\Omega_c} = \frac{(1-Z^{-1})}{(1+Z^{-1})\tan\left(\frac{W_c}{2}\right)}$$

<u>Example (2):</u> Design and realize a digital low-pass filter using bilinear transformation method to satisfy the following c/cs:

1. − 3.01 dB cutoff frequency of 0.5 π rad
2. Magnitude down at least 15 dB at 0.75 rad.

Solution:

Step (1): applying eq. (9.17), where T=1 (***prewarp case***)

$$\Omega_c = 2\tan\left(\frac{W_1}{2}\right) = 2\tan(0.5\,\pi/2) = 2$$

$$\Omega_r' = 2\tan\left(\frac{W_2}{2}\right) = 2\tan(0.75\,\pi/2) = 4.8282$$

Step (2) : applying eq. (8.4) and (8.5a):

$$n = \left\lceil \frac{\log_{10}\{(10^{-0.1k_1}-1)/(10^{-0.1k_2}-1)\}}{2\log_{10}(1/\Omega_r)} \right\rceil$$

$$n = \left\lceil \frac{\log_{10}\{(10^{3.01/10}-1)/(10^{15/10}-1)\}}{2\log_{10}(2/4.8282)} \right\rceil = \lceil 1.9412 \rceil = 2$$

$$\Omega_c = 2/(10^{3.01/10}-1)^{1/4} = 2 \; rad$$

Referring to lecture 8, Table (1) to write the normalized Butterworth LPF equation, and then using LP → LP transformation:

$$H_a(S) = \frac{1}{S^2+\sqrt{2}\,S+1}\Big|_{S\to S/2} = \frac{4}{S^2+2\sqrt{2}\,S+4}$$

Step (3): Applying bilinear transformation, eq.(9.15), T = 1

$$H(Z) = \frac{4}{\left[\frac{2(1-Z^{-1})}{(1+Z^{-1})}\right]^2 + 2\sqrt{2}\left[\frac{2(1-Z^{-1})}{(1+Z^{-1})}\right] + 4} = \frac{1+2Z^{-1}+Z^{-2}}{3.4142135+0.5857865\,Z^{-2}}$$

y(n) = 0.2928932 { x(n) + 2 x(n−1) + x(n−2) } − 0.1715729 y(n−2)

3. <u>Digital-to digital transformation design method</u>

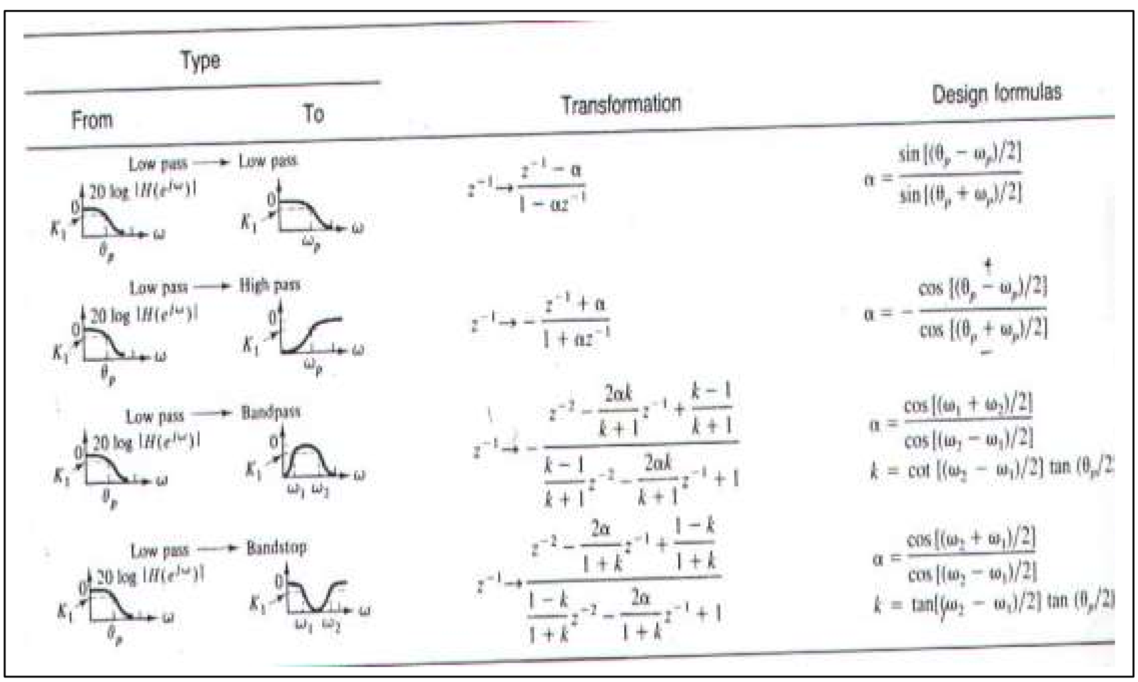
1. Use digital specifications to calculate the order of *digital unit bandwidth low-pass Butterworth prototype* and corresponding critical frequency Wp. The order of the digital filter can be obtained by using eq. (9.17) of the prewarped digital frequencies Ω_u, and Ω'_r in the standard formula for the analog Butterworth filter { eq. (8.4) }, as:

2. $$n = \left\lceil \frac{\log_{10}\{(10^{-0.1k_1}-1)/(10^{-0.1k_2}-1)\}}{2\log_{10}\{\tan(W_1/2)/\tan(W_2/2)\}} \right\rceil \qquad (9.22)$$

$$W_1 = 2\tan^{-1}\{\tan(W_p/2)\,(10^{-0.1k_1}-1)^{-1/2n}\} \qquad (9.23)$$

<u>**Note**</u>**:** Refer to lecture 8, Table (2) to substitute for Ω_r in eq. (9.22) in terms of eq.(9.17).

Table (1) Digital-to digital transformation



Type: From	Type: To	Transformation	Design formulas
Low pass	Low pass	$z^{-1} \to \frac{z^{-1}-\alpha}{1-\alpha z^{-1}}$	$\alpha = \frac{\sin[(\theta_p-\omega_p)/2]}{\sin[(\theta_p+\omega_p)/2]}$
Low pass	High pass	$z^{-1} \to -\frac{z^{-1}+\alpha}{1+\alpha z^{-1}}$	$\alpha = -\frac{\cos[(\theta_p-\omega_p)/2]}{\cos[(\theta_p+\omega_p)/2]}$
Low pass	Bandpass	$z^{-1} \to -\frac{z^{-2}-\frac{2\alpha k}{k+1}z^{-1}+\frac{k-1}{k+1}}{\frac{k-1}{k+1}z^{-2}-\frac{2\alpha k}{k+1}z^{-1}+1}$	$\alpha = \frac{\cos[(\omega_1+\omega_2)/2]}{\cos[(\omega_2-\omega_1)/2]}$, $k = \cot[(\omega_2-\omega_1)/2]\tan(\theta_p/2)$
Low pass	Bandstop	$z^{-1} \to \frac{z^{-2}-\frac{2\alpha}{1+k}z^{-1}+\frac{1-k}{1+k}}{\frac{1-k}{1+k}z^{-2}-\frac{2\alpha}{1+k}z^{-1}+1}$	$\alpha = \frac{\cos[(\omega_2+\omega_1)/2]}{\cos[(\omega_2-\omega_1)/2]}$, $k = \tan[(\omega_2-\omega_1)/2]\tan(\theta_p/2)$

(9.24) 3- From Table (1), calculate $\alpha = \dfrac{\sin\{(\theta_P - W_1)/2\}}{\sin\{(\theta_P + W_1)/2\}}$

4- Table (2) gives $H_{Bn}(Z)$ for *normalized low-pass Butterworth* digital filter. Calculate

$$H(Z) = H_{Bn}(Z) \Big|_{Z^{-1} \rightarrow \frac{(Z^{-1}-\alpha)}{(1+\alpha Z^{-1})}} \qquad (9.25)$$

Example (3): Use Digital-to digital transformation method. Find H(Z) for LP digital filter that satisfies the following requirements:

1- A − 3.0102 dB cutoff digital frequency of 0.5 π rad.

2- Attenuation at and past 0.75 π rad is at least 15 dB

Solution:

$$n = \left\lceil \frac{\log_{10}\{(10^{0.30102}-1)/(10^{1.5}-1)\}}{2\log_{10}\{\tan(0.5\pi/2)/\tan(0.75\pi/2)\}} \right\rceil = \lceil 1.9412 \rceil = 2$$

$$W_P = 2\tan^{-1}\{(10^{0.30102}-1)^{-1/4}\tan(0.5\pi/2)\} = 0.5\pi, \quad \theta_P = 1 \text{ (normalized)}$$

$$\alpha = \frac{\sin\{(1-0.5\pi)/2\}}{\sin\{(1+0.5\pi)/2\}} = -0.293401993$$

Using Table (2) that gives $H_{Bn}(Z)$ for *normalized low-pass Butterworth* digital filter

$$H_{B2}(Z) = \frac{0.144106\,(1+Z^{-1})^2}{1-0.677496\,Z^{-1}+0.253921\,Z^{-2}}$$

Applying eq.(9.25) , then:

$$H(Z) = \frac{(1+Z^{-1})^2}{3.4142+0.5858\,Z^{-2}}$$

9.3.4 Impulse invariant design method

If $h_a(t)$ represents the response of an analog filter to a unit impulse δ(t), then the unit sample response of a discrete-time filter used in an A/D-H(Z)-D/A structure is selected to be the sampled version of h(n).

$$H(Z) = Z\{h(n)\} = Z\{h_a(t)\big|_{t=nT}\} \qquad (9.26)$$

If an analog filter with system function $H_a(S)$ is given, the corresponding impulse invariant design filter has

$$H(Z) = Z\{L^{-1}H_a(S)\big|_{t=nT}\} \qquad (9.27)$$

Example (4): Find H(Z) corresponding to the impulse invariant design using sampling rate of (1/T) samples / sec. for an analog filter $H_a(S)$ specified as: $H_a(S) = A / (S + \alpha)$

Solution:

$h_a(t) = L^{-1} H_a(S) = A\, e^{-\alpha t} u(t)$

$h(n) = h_a(t)_{\,t = nT} = A\, e^{-\alpha nT} u(nT)$

$$H(Z) = Z\{h(n)\} = \frac{AZ}{Z - e^{-\alpha T}}$$

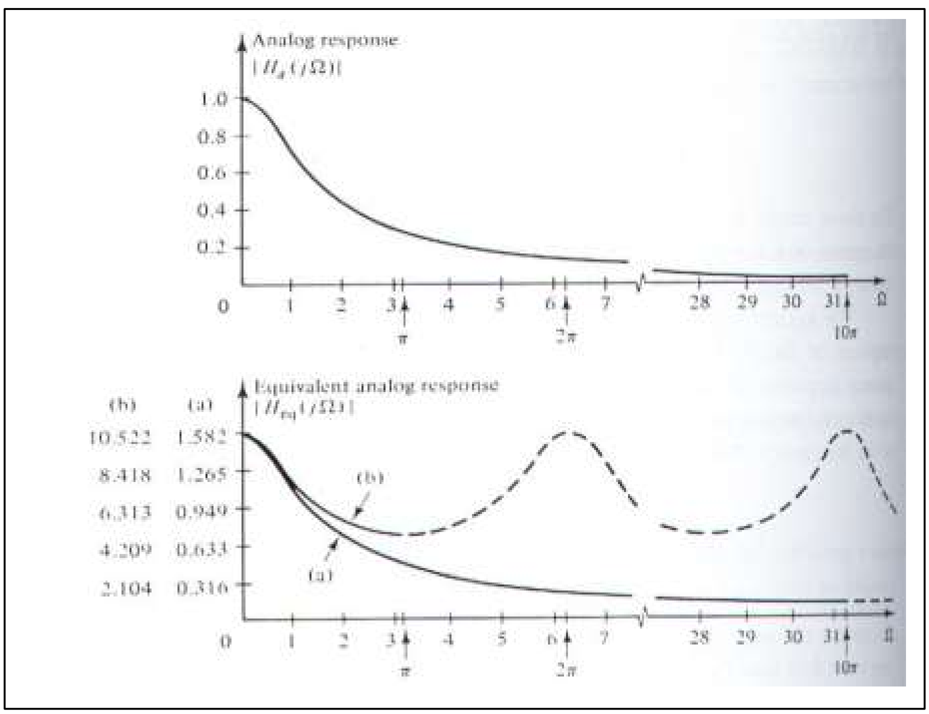
$$\left| H_a(j\Omega) \right| = \frac{A}{(\alpha^2 + \Omega^2)^{1/2}} \qquad , S = j\Omega$$

$$H(e^{jW}) = \frac{A e^{jW}}{1 - e^{-jW}} = \frac{A}{e^{-\alpha T} e^{jW} - e^{-\alpha T}} \qquad Z = e^{jW},$$

$$H_{eq}(j\Omega) = H(e^{jW}) \Big|_{W \to \Omega T} = \frac{A e^{j\Omega T}}{e^{j\Omega T} - e^{-\alpha T}} \qquad \Omega < \pi / T,$$

$$H_{eq}(j\Omega) = \frac{A}{\{1 - e^{-\alpha T}\cos(\Omega T)\} + j\, e^{-\alpha T}\sin(\Omega T)} \qquad \Omega < \pi / T,$$

$$\left| H_{eq}(j\Omega) \right| = \frac{A}{\sqrt{1 + e^{-2\alpha T} - 2 e^{-\alpha T}\cos(\Omega T)}} \qquad \Omega < \pi / T,$$



represents α = 1, T = 0.1 , $|H_a(j\Omega)|$ and $|H_{eq}(j\Omega)|$ are very close. (a)

represents α = 1, T = 1 , $|H_a(j\Omega)|$ and $|H_{eq}(j\Omega)|$ are different. (b)

Good results using impulse invariant design are obtained when the time between samples is selected small.

9.4 Pole-Zero Placement Method for Simple Infinite Impulse Response Filters Design

This section introduces a pole-zero placement method for a simple IIR filter design. Let us first examine effects of the pole-zero placement on the magnitude response in the z-plane shown in Fig .(9.3).

In the z-plane, when we place a pair of complex conjugate zeros at a given point on the unit circle with an angle θ, we will have a numerator factor of $(z - e^{j\theta})(z - e^{-j\theta})$ in the transfer function. Its magnitude contribution to the frequency response at $z = e^{jW}$ is $(e^{jW} - e^{j\theta})(e^{jW} - e^{-j\theta})$. When $W = \theta$, the magnitude will reach zero.

When a pair of complex conjugate poles are placed at a given point within the unit circle, we have a denominator factor of $(z - r\,e^{j\theta})(z - r\,e^{-j\theta})$, where r is the radius chosen to be less than and close to 1 to place the poles inside the unit circle. The magnitude contribution to the frequency response at $W = \theta$ will rise to a large magnitude, since the first factor $(e^{j\theta} - r\,e^{j\theta}) = (1 - r)\,e^{-j\theta}$ gives a small magnitude of $1 - r$, which is the length between the pole and the unit circle at the angle $W = \theta$. Note that the magnitude of $e^{-j\theta}$ is 1.

Therefore, we can reduce the magnitude response using zero placement, while we increase the magnitude response using pole placement. Placing a combination of poles and zeros will result in different frequency responses. such as lowpass, highpass, bandpass, and bandstop. It is easy to compute filter coefficients for simple IIR filters. Practically, the pole-zero placement method has good performance when the bandpass and bandstop filters have very narrow bandwidth requirements and the lowpass and highpass filters have either very low cutoff frequencies close to the DC or very high cutoff frequencies close to the folding frequency (the Nyquist limit).

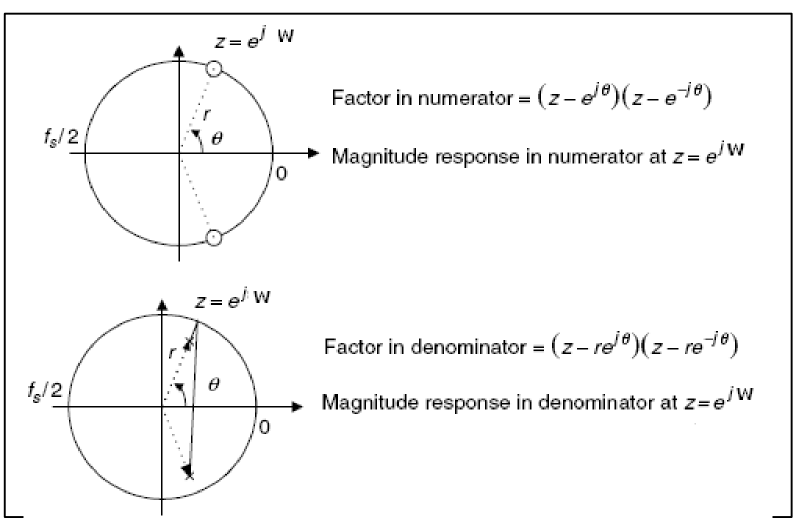
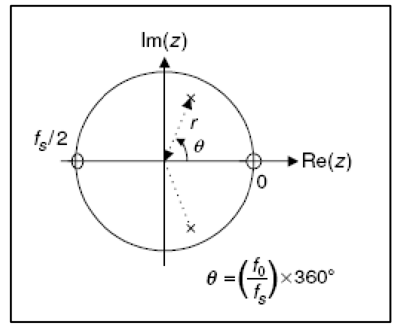


Fig. (9.3) Effects of pole-zero placement on the magnitude response.

9.4.1 Second-Order Bandpass Filter Design

Poles in a band-pass filter are complex conjugate, with the magnitude r controlling the bandwidth and the angle θ controlling the center frequency. The zeros are placed at z = 1, corresponding to DC, and z = -1, corresponding to the folding frequency. The poles will raise the magnitude response at the center frequency while the zeros will cause zero gains at DC (zero frequency) and at the folding frequency. The following equations give the band-pass filter design formulas using pole-zero placement:



$$r \approx 1 - (BW_{3dB}/f_s) \times \pi, \text{ good for } 0.9 \le r < 1$$

$$\theta = \left(\frac{f_0}{f_s}\right) \times 360^0$$

$$H(z) = \frac{K(z-1)(z+1)}{(z - re^{j\theta})(z - re^{-j\theta})} = \frac{K(z^2 - 1)}{(z^2 - 2rz\cos\theta + r^2)},$$

$$K = \frac{(1-r)\sqrt{1 - 2r\cos 2\theta + r^2}}{2|\sin\theta|}. \quad (9.28)$$

Where, K is a scale factor to adjust the band-pass filter to have a unit pass-band gain

Example (5): A second-order bandpass filter is required to satisfy the following specifications:

Sampling rate = 8,000 Hz .1

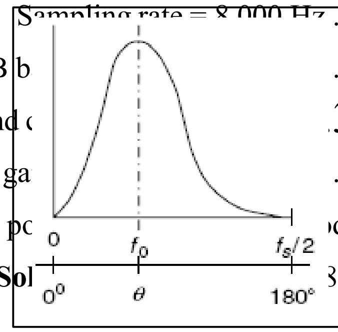
A 3 dB b .2

Narrow passband c .3

Zero ga .4

Find the transfer function using the pc d.

Sol 3),



$$r = 1 - (200/8000)\pi = 0.9215$$

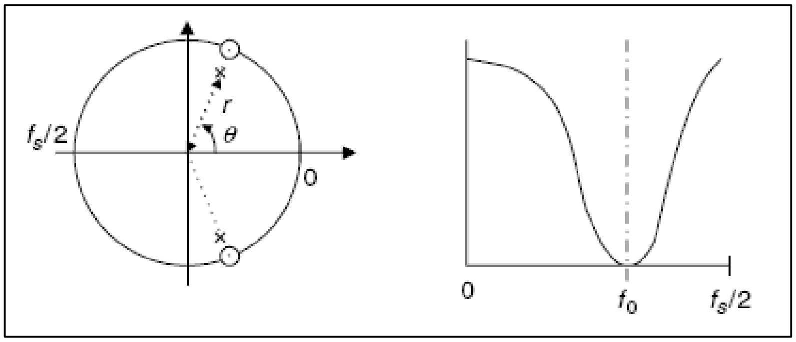
$$\theta = \left(\frac{1000}{8000}\right) \times 360 = 45^0.$$

$$K = \frac{(1 - 0.9215)\sqrt{1 - 2 \times 0.9215 \times \cos 2 \times 45^0 + 0.9215^2}}{2|\sin 45^0|} = 0.0755.$$

$$H(z) = \frac{0.0755(z^2 - 1)}{(z^2 - 2 \times 0.9215z\cos 45^0 + 0.9215^2)} = \frac{0.0755 - 0.0755z^{-2}}{1 - 1.3031z^{-1} + 0.8491z^{-2}}.$$

9.4.2 Second-Order Bandstop (Notch) Filter Design

For this type of filter, the pole placement is the same as the bandpass filter. The zeros are placed on the unit circle with the same angles with respect to the poles. This will improve passband performance. The magnitude and the angle of the complex conjugate poles determine the 3 dB bandwidth and the center frequency, respectively.



Design formulas for band-stop filters are given in the following equations:

$$r \approx 1 - (BW_{3dB}/f_s) \times \pi, \text{ good for } 0.9 \le r < 1$$

$$\theta = \left(\frac{f_0}{f_s}\right) \times 360^0$$

$$H(z) = \frac{K(z - e^{j\theta})(z + e^{-j\theta})}{(z - re^{j\theta})(z - re^{-j\theta})} = \frac{K(z^2 - 2z\cos\theta + 1)}{(z^2 - 2rz\cos\theta + r^2)}.$$

$$K = \frac{(1 - 2r\cos\theta + r^2)}{(2 - 2\cos\theta)}. \quad (9.29)$$

Example (6): A second-order notch filter is required to satisfy the following specifications:

Sampling rate = 8,000 Hz .1

A 3 dB bandwidth: BW = 100 Hz .2

Narrow pass-band centered at f_0 = 1,500 Hz: .3

Find the transfer function using the pole-zero placement approach.

Solution:

$$r \approx 1 - (100/8000) \times \pi = 0.9607,$$

$$\theta = \left(\frac{1500}{8000}\right) \times 360^0 = 67.5^0.$$

$$K = \frac{(1 - 2 \times 0.9607\cos 67.5^0 + 0.9607^2)}{(2 - 2\cos 67.5^0)} = 0.9620.$$

$$H(z) = \frac{0.9620(z^2 - 2z\cos 67.5^0 + 1)}{(z^2 - 2 \times 0.9607z\cos 67.5^0 + 0.9607^2)}$$

$$= \frac{0.9620 - 0.7363z^{-1} + 0.9620z^{-2}}{1 - 0.7353z^{-1} + 0.9229}.$$

9.4.3 First-Order Low-pass Filter Design

The first-order pole-zero placement can be operated in two cases. The first situation is when the cutoff frequency is less than $f_s/4$. Then the pole-zero placement is shown in Fig. (9.4a).

As shown in Fig.(9.4a), the pole $z = \alpha$ is placed in the real axis. The zero is placed at $z = -1$ to ensure zero gain at the folding frequency (Nyquist limit). When the cutoff frequency is above $f_s / 4$, the pole-zero placement is adopted as shown in Fig.(9.4b).

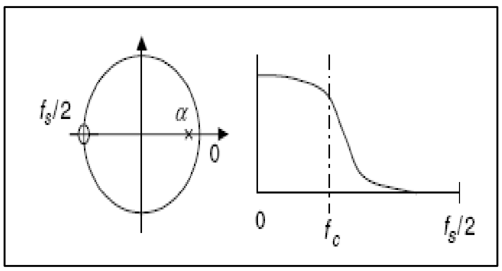


Fig. (9.4a) Pole-zero placement for the first-order lowpass filter with $f_c < f_s/4$.

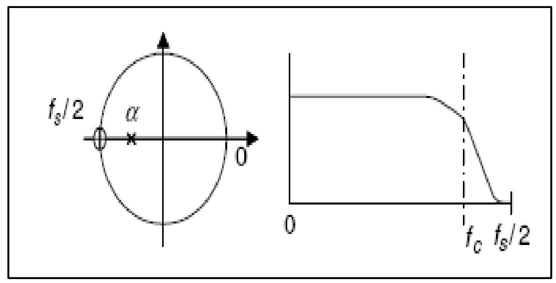


Fig.(9.4b) Pole-zero placement for the first-order lowpass filter with $f_c > f_s/4$.

Design formulas for lowpass filters using the pole-zero placement are given in the following equations:

$$H(z) = \frac{K(z+1)}{(z-\alpha)},$$

$$K = \frac{(1-\alpha)}{2}. \tag{9.30}$$

Example (7): A first-order lowpass filter is required to satisfy the following specifications:

1. Sampling rate = 8,000 Hz
2 A 3 dB cutoff frequency: $f_c = 100$ Hz
3. Zero gain at 4,000 Hz.

Find the transfer function using the pole-zero placement method.

Solution: Since the cutoff frequency of 100 Hz is much less than $f_s / 4 = 2{,}000$ Hz, we determine the pole as:

$$\alpha \approx 1 - 2 \times (100/8000) \times \pi = 0.9215,$$

Which is above 0.9. Hence, we have a good approximation. The unit-gain scale factor is calculated by:

$$K = \frac{(1 - 0.9215)}{2} = 0.03925.$$

$$H(z) = \frac{0.03925(z + 1)}{(z - 0.9215)} = \frac{0.03925 + 0.03925z^{-1}}{1 - 0.9215z^{-1}}.$$

Note that we can also determine the unit-gain factor K by substituting Z = $e^{j0} = 1$ to the transfer function H(Z) = (Z + 1) / (Z - α), then find a DC gain. Set the scale factor to be a reciprocal of the DC gain. This can be easily done, that is,

$$DC\ gain = \left.\frac{z + 1}{z - 0.9215}\right|_{z=1} = \frac{1 + 1}{1 - 0.9215} = 25.4777.$$

Hence, $K = 1/25.4777 = 0.03925$.

9.4.4 First-Order High-pass Filter Design

Similar to the low-pass filter design, the pole-zero placements for first-order high-pass filters in two cases are shown in Figures (9.5a) and (9.5b).

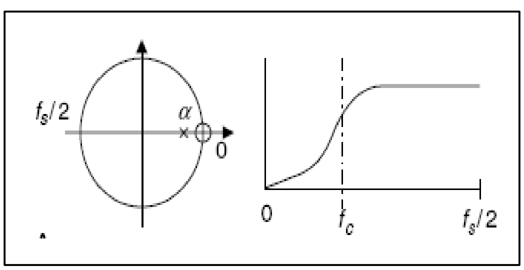


Fig.(9.5a) Pole-zero placement for the first-order highpass filter with $f_c < f_s/4$.

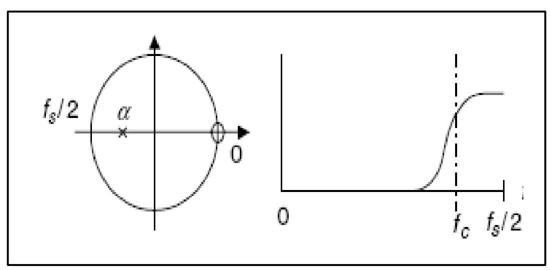


Fig.(9.5b) Pole-zero placement for the first-order highpass filter with $f_c > f_s/4$.

Formulas for designing highpass filters using the pole-zero placement are listed in the following equations:

When $f_c < f_s/4$, $\alpha \approx 1 - 2 \times (f_c/f_s) \times \pi$, good for $0.9 \le r < 1$.

When $f_c > f_s/4$, $\alpha \approx -(1 - \pi + 2 \times (f_c/f_s) \times \pi)$, good for $-1 < r \le -0.9$

$$H(z) = \frac{K(z-1)}{(z-\alpha)}$$

$$K = \frac{(1+\alpha)}{2}. \tag{9.31}$$

Example (8): A first-order highpass filter is required to satisfy the following specifications:

1. Sampling rate = 8,000 Hz
2. A 3 dB cutoff frequency: fc = 3800 Hz
3. Zero gain at 0 Hz.

Find the transfer function using the pole-zero placement method.

Solution:

Since the cutoff frequency of 3,800 Hz is much larger than $f_s / 4$ = 2,000 Hz, we determine the pole as:

$$\alpha \approx -(1 - \pi + 2 \times (3800/8000) \times \pi) = -0.8429,$$

$$K = \frac{(1-0.8429)}{2} = 0.07854$$

$$H(z) = \frac{0.07854(z-1)}{(z+0.8429)} = \frac{0.07854 - 0.07854z^{-1}}{1+0.8429z^{-1}}.$$

Note that we can also determine the unit-gain scale factor K by substituting $Z = e^{j180} = -1$ into the transfer function $H(Z) = (Z-1) / (Z-\alpha)$, finding a passband gain at the Nyquist limit $f_s/2$ = 4,000 Hz. We then set the scale factor to be a reciprocal of the passband gain. That is,

$$passband\ gain = \left.\frac{z-1}{z+0.8429}\right|_{z=1} = \frac{-1-1}{-1+0.8429} = 12.7307.$$

Hence, $K = 1/12.7307 = 0.07854$.

9. 5 Application: Digital Audio Equalizer

For an audio application such as the CD player, the digital audio equalizer is used to make the sound as one desires by changing filter gains for different audio frequency bands. Other applications include adjusting the sound source to take room acoustics into account, removing undesired noise, and boosting the desired signal in the specified pass-band. The simulation is based on the consumer digital audio processor—such as a CD player—handling the 16-bit digital samples with a sampling rate of 44.1 kHz and an audio signal bandwidth at 22.05 kHz. A block diagram of the digital audio equalizer is depicted in Fig (9.6).

A seven-band audio equalizer is adopted for discussion. The center frequencies are listed in Table (2). The 3 dB bandwidth for each band-pass filter is chosen to be 50% of the center frequency. As shown in Fig (9.6), g_0 through g_6 are the digital gains for each band-pass filter output and can be adjusted to make sound effects, while $y_0(n)$ through $y_6(n)$ are the digital amplified bandpass filter outputs. Finally, the equalized signal is the sum of the amplified bandpass filter outputs and itself. By changing the digital gains of the equalizer, many sound effects can be produced. A IIR bandpass Butterworth filters are chosen for the audio equalizer. The coefficients are achieved using the BLT method.

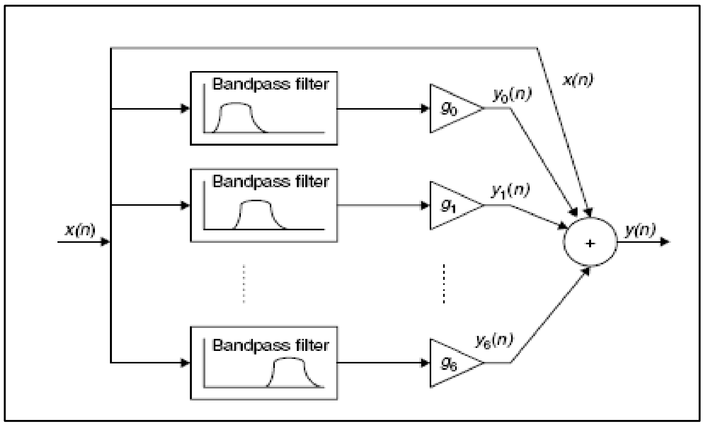


Fig. (9.6) Simplified block diagram of the audio equalizer.

Table (2) Specifications for an audio equalizer to be designed.

Center frequency (Hz)	100	200	400	1000	2500	6000	15000
Bandwidth (Hz)	50	100	200	500	1250	3000	7500

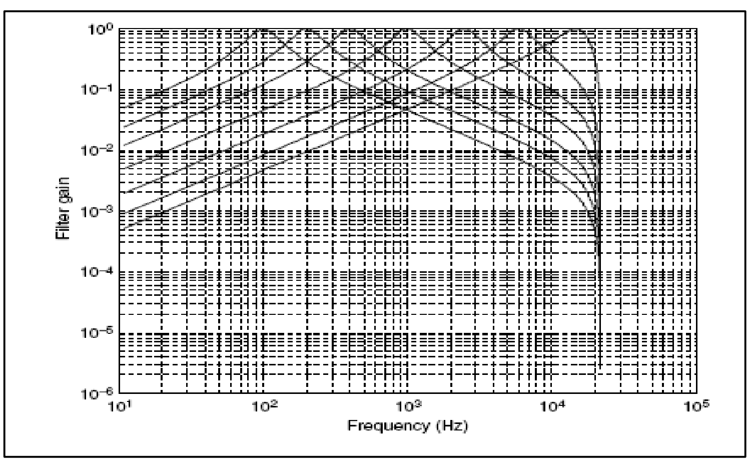


Fig. (9.7) Magnitude frequency responses for the audio equalizer.

The audio test signal having frequency components of 100 Hz, 200 Hz, 400 Hz, 1,000 Hz, 2,500 Hz, 6,000 Hz, and 15,000 Hz.

The gains set for the filter banks are: $g_o = 10$; $g_1 = 10$; $g_2 = 0$; $g_3 = 0$; $g_4 = 0$; $g_5 = 10$; $g_6 = 10$. The frequency components at 100 Hz, 200 Hz, 6,000 Hz, and 15,000 Hz will be boosted by $20 \log_{10} 10 = 20$ dB. The top plot in Fig. (9.8), shows the spectrum for the audio test signal, while the bottom plot depicts the spectrum for the equalized audio test signal. Before audio digital equalization, the spectral peaks at all bands are at the same level; after audio digital equalization, the frequency components at bank 0, bank 1, bank 5, and bank 6 are amplified. The operation of the digital equalizer boosts the low frequency components and the high frequency components.

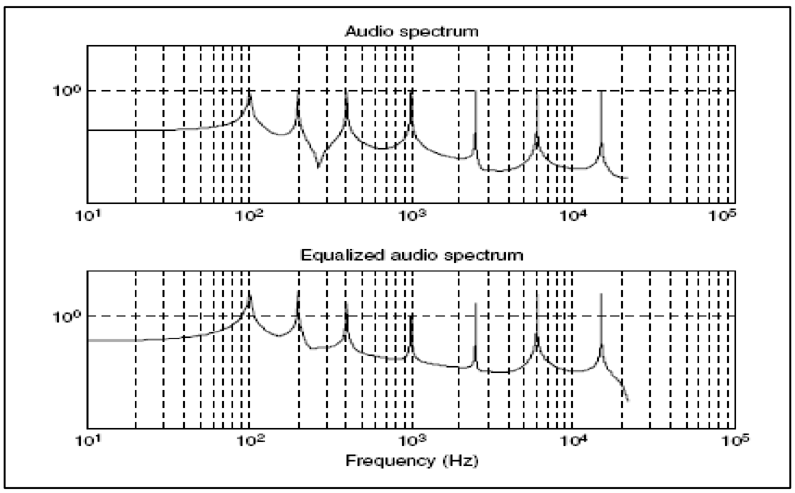


Fig. (9.8) Audio spectrum and equalized audio spectrum.

1. Rectangular:

$$w_R(n)=\begin{cases}1 & 0\le n\le N-1\\ 0 & otherwise\end{cases} \qquad (9.45)$$

2. Bartlett:

$$w_B(n)=\begin{cases}\dfrac{2n}{(N-1)} & 0\le n\le (N-1)/2\\ 2-\dfrac{2n}{(N-1)} & (N-1)/2\le n\le (N-1)\\ 0 & elsewhere\end{cases} \qquad (9.46)$$

3. Hanning:

$$w_{Han}(n)=\begin{cases}0.5\left[1-\cos\left(\dfrac{2\pi n}{(N-1)}\right)\right], & 0\le n\le N-1\\ 0 & elsewhere\end{cases} \qquad (9.47)$$

4. Hamming:

$$w_{Ham}(n)=\begin{cases}0.54-0.46\cos\left(\dfrac{2\pi n}{(N-1)}\right), & 0\le n\le N-1\\ 0 & elsewhere\end{cases} \qquad (9.48)$$

5. Blackman:

$$w_{Bl}(n)=\begin{cases}0.42-0.5\cos\left(\dfrac{2\pi n}{(N-1)}\right)+0.08\cos\left(\dfrac{4\pi n}{(N-1)}\right), & 0\le n\le N-1\\ 0 & elsewhere\end{cases} \qquad (9.49)$$

An ideal LP filter with linear phase of slope $-\alpha$ and cutoff w_c can be characterized in frequency domain by:

$$H_d(e^{jW})=\begin{cases}e^{-jW\alpha} & |W|<W_c\\ 0 & W_c<|W|<\pi\end{cases} \qquad (9.50)$$

Using inverse F.T (eq. (4.11), PP. 28):

$$h_d(n)=\frac{\sin[w_c(n-\alpha)]}{\pi(n-\alpha)} \qquad (9.51)$$

For a causal FIR filter, and using :

$$h(n)=h_d(n)\,.\,w(n) \qquad (9.52)$$

Substituting eq.(9.51) into eq.(9.52), yield:

$$h(n) = \frac{\sin[\, w_c(n-\alpha)\,]}{\pi(n-\alpha)} \cdot w(n) \qquad (9.53)$$

For h(n) to be a linear phase filter, α = (N-1) / 2.

Table (3) shows $h_d(n)$ for LPF, HPF, BPF, and BSF:

Table (3) $h_d(n)$ and $h_d(\alpha)$ for LPF, HPF, BPF, and BS

Filter Type	**$h_d(n)$**	**$h_d(\alpha)$**
LPF	$h_d(n) = \frac{\sin[\, w_c(n-\alpha)\,]}{\pi\,(n-\alpha)}$	$h_d(\alpha) = w_c / \pi$
HPF	$h_d(n) = -\frac{\sin[\, w_c(n-\alpha)\,]}{\pi\,(n-\alpha)}$	$h_d(\alpha) = 1 - (w_c / \pi)$
BPF	$N_1 = \frac{2\pi k}{w_l - w_1}$, $N_2 = \frac{2\pi k}{w_2 - w_u}$ N = max (N_1 , N_2) $h_d(n) = \frac{\sin\{w_u(n-\alpha)\} - \sin\{w_l(n-\alpha)\}}{\pi(n-\alpha)}$	$h_d(\alpha) = (w_u - w_l) / \pi$
BSF	$N_1 = \frac{2\pi k}{w_1 - w_l}$, $N_2 = \frac{2\pi k}{w_u - w_2}$ N = max (N_1 , N_2) $h_d(n) = \frac{\sin\{w_l(n-\alpha)\} - \sin\{w_u(n-\alpha)\}}{\pi(n-\alpha)}$	$h_d(\alpha) = (\pi - w_u - w_l) / \pi$

In general, for all the above filters with N odd:

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

$$H(e^{jW}) = e^{-jW(\frac{N-1}{2})}\left\{h\left(\frac{N-1}{2}\right) + \sum_{n=0}^{(N-3)/2} 2\,h(n)\cos\left[\,W\left(n-\frac{N-1}{2}\right)\right]\right\}$$

$\Phi(W) = -W\alpha$, with $\alpha = (N-1)/2$

Notes:

- The stop-band gain for the LPF designed is relatively insensitive to the size of the window and the selection of w_c depending mainly on the type of window.
- The transition width of the designed LPF is approximately equal to the main lobe of the window used. See Table (4)

Table (4) Design table for FIR LPF

Window	**Transition Width (w_t)**	**Minimum stop-band attenuation**
Rectangular	$4\pi / N$	− 21 dB
Bartlett	$8\pi / N$	− 25 dB
Hanning	$8\pi / N$	− 44 dB
Hamming	$8\pi / N$	− 53 dB
Blackman	$12\pi / N$	− 74 dB

<u>Design procedure for an FIR filter</u> .8

Requirements: k_1, w_1, k_2, and w_2 represents the cutoff and stop-band requirements for digital filters.

From Table (4), select the window type such that the stop-band gain exceeds k_2 .1

Selects the number of points in the window, 2

$$w_t = w_2 - w_1 \geq k\,(2\pi / N),$$

N is preferred odd $\quad N \geq k\,(2\pi) / (w_2 - w_1),$

Select α and w_c, where: .3

$$w_c = w_1, \text{ and } \alpha = (N - 1)/2$$

Find h(n) from eq. (9.52) using the specified window type and Table (3) . .4

Use eq. (9.42) or eq.(9.43) to plot the frequency response $H(e^{jW})$, and check to see if the given specifications are satisfied. .5

If the attenuation requirement at w_1 is not satisfied, increase w_c and return to step 4, and 5 . .6

If the frequency response requirements are satisfied, check to see if a further reduction of N might be possible. If a further reduction in N is not possible, then h(n) found is the desired design, otherwise, reduce N and return to step 3. .7

If the filter is to be used in A/D- H(Z) – D/A structure, the equivalent analog specifications must be converted to digital specifications. For analog critical frequencies, Ω_i, the corresponding digital specifications using a sampling rate of 1 / T samples /sec. ; .8

$$w_i = \Omega_i T$$

<u>Example (9):</u> Design a LP digital filter to be used in A/D- H(Z) – D/A structure that will have a − 3 dB cutoff of 30 π rad / sec. and an attenuation of 50 dB at 45 π rad/sec. *The filter is required to have linear phase*. The system will use a sampling rate of 100 samples/sec.

Solution:

$w_c = w_1 = \Omega_u T = 30\,\pi\,(1/100) = 0.3\,\pi$ rad

$w_2 = w_r = \Omega_r T = 45\,\pi\,(1/100) = 0.45\,\pi$ rad

Hamming window is chosen. .1

From step (2): .2

$(8\,\pi / N) = k\,(2\,\pi / N)$, Then $k = 4$

$N \geq 4\,(2\,\pi) / (0.45 - 0.3)\,\pi = 53.3 = 55$

$w_c = w_u = 0.3\,\pi$ rad , and $\alpha = (N - 1)/2 = 27$3.

4. Using eq. (9.48) for w_{Ham} and the value of $h_d(n)$ from Table (3) to find h(n):

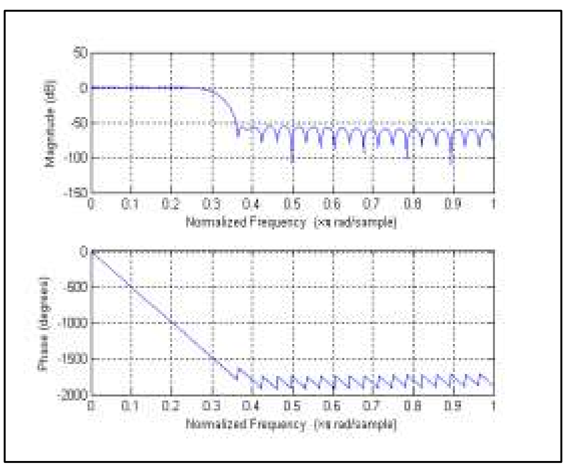
$$h(n) = \frac{\sin[\,0.3\pi\,(n-27)\,]}{\pi\,(n-27)}\,.\{0.54 - 0.46\cos(2\pi\,n/54)\}\,,\ 0 \leq n \leq 54$$

$$H(e^{jW}) = e^{-jW(27)}\,\{h(27) + \sum_{n=0}^{26} 2\,h(n)\,\{\cos[\,W(n-27)\,]\}$$

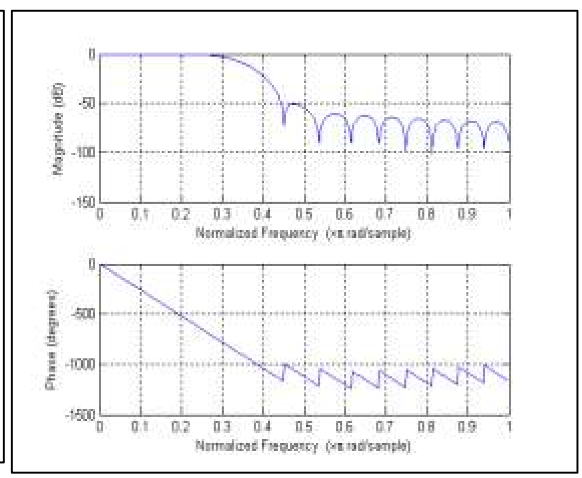
From the results obtained from MATLAB program, the attenuation is seen to be too much at $w_c = w_1$. The design is improved by making $w_c = 0.33$ rad / sec, then $N = 29$, $\alpha = 14$ and

$$h(n) = \frac{\sin[\,0.33\,\pi\,(n-14)\,]}{\pi\,(n-14)}\,.\{0.54 - 0.46\cos(2\pi\,n/28)\}\,,\ 0 \leq n \leq 28$$

$$H(e^{jW}) = e^{-jW(14)}\,\{h(14) + \sum_{n=0}^{13} 2\,h(n)\,\{\cos[\,W(n-14)\,]\}$$



N= 55, $w_c = w_u = 0.3\,\pi$ rad



N= 29, $w_c = w_u = 0.33\,\pi$ rad