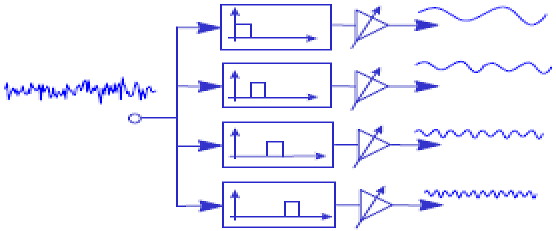
جامعة ديالى
كلية الهندسة

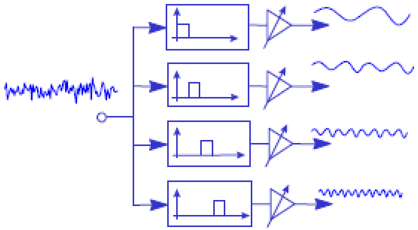
www.engineering.uodiyala.edu.iq

University Of Diyala – College Of Engineering

Digital Signal Processing



Course Instructor
Dr. Ali J. Abboud



Lecture No. 7: Digital Filters-FIR

Third Class

Department of Computers and Software Engineering

http://www.engineering.uodiyala.edu.iq/
https://www.facebook.com/Engineer.College1

Lecture Outline

- Functions of Digital Filters.
- Advantages and Disadvantages of Digital Filters.
- Different Methods for Describing Digital Filters.
- Classification of Digital Filters.
- Filter Structures.
- Types of Digital Filters.
- Finite Impluse Response (FIR) Filters Design.

Digital Filters

- A filter is essentially a system that selectively changes the wave shape of a signal in a desired manner.
- Objectives of filtering are to
  - Improve quality of a signal (remove noise)
  - To extract info from signals
  - To separate two or more signals previously combined
- A digital filter, is a mathematical algorithm implemented in hardware and/or software that operates on a digital i/p signal to produce a digital o/p signal for the purpose of achieving a filtering objectives.
- Compared to analog filters, digital filters are preferred in many applications like data compression, speech processing, image processing, biomedical signal processing, data transmission etc.

Functions of Digital Filters

The primary functions of filters are one of the followings:

(a) To confine a signal into a prescribed frequency band as in low-pass, high-pass, and band-pass filters.

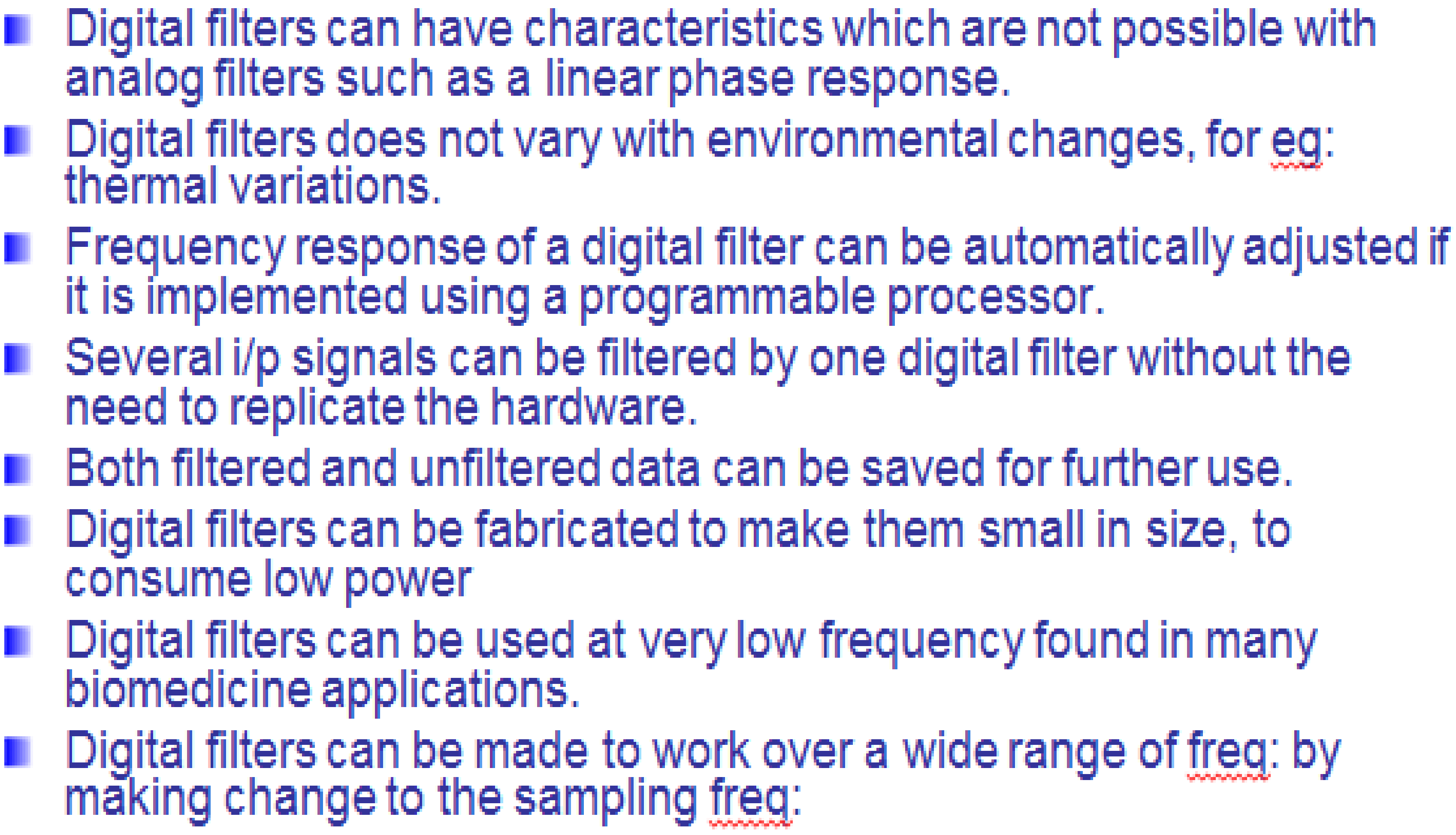
(b) **To decompose a signal into two or more sub-bands as in filter-banks, graphic equalizers, sub-band coders, frequency multiplexers.**

(c) To modify the frequency spectrum of a signal as in telephone channel equalization and audio graphic equalizers.

(d) To model the input-output relationship of a system such as telecommunication channels, human vocal tract, and music synthesizers.

(e) To enhance or restore original by removing noise or distortion from corrupted signal.

Advantages of Digital Filters



- Digital filters can have characteristics which are not possible with analog filters such as a linear phase response.
- Digital filters does not vary with environmental changes, for eg: thermal variations.
- Frequency response of a digital filter can be automatically adjusted if it is implemented using a programmable processor.
- Several i/p signals can be filtered by one digital filter without the need to replicate the hardware.
- Both filtered and unfiltered data can be saved for further use.
- Digital filters can be fabricated to make them small in size, to consume low power
- Digital filters can be used at very low frequency found in many biomedicine applications.
- Digital filters can be made to work over a wide range of freq: by making change to the sampling freq:

Disadvantages of Digital Filters

- Speed is limited in digital filters as compared to analog filter.
  - In real time situations, ADC and DAC processes introduces a speed constraint on digital filters.
  - The speed of operation of a digital filter also depends on the speed of digital processor.
- Finite Wordlength effect: Digital filters are subject to ADC noise due to quantization of continuous signal and round off noise incurred during computation which could lead to instability.
- Long design & development time for digital filters, especially hardware development, can be much longer than for analog filters.

Different Methods for Describing of Digital Filters

Filters can be described using the following time or frequency domain methods:

(a) *Time domain input-output relationship.*

difference equation is used to describe the output of a discrete-time filter in terms of a weighted combination of the input and previous output samples. For example a first-order filter may have the following difference equation

$$y(m)=a\,y(m-1)+x(m)$$

where $x(m)$ is the filter input, $y(m)$ is the filter output and a is the filter coefficient.

Different Methods for Describing of Digital Filters

(b) *Impulse Response.* A filter can be described in terms of its response to an impulse input. For example the response of the filter of Eq.   to a discrete-time impulse input at m=0 is

$$y(m) = a^m \qquad m=0,\ 1,\ 2,\ \ldots$$

$y(m) = a^m = 1, a, a^2, a^3,\ a^4, \ldots$ for m=0,1,2,3, 4 ... and it is assumed y(-1)=0.

Impulse response is useful because: (i) any signal can be viewed as the sum of a number of shifted and scaled impulses, hence the response a linear filter to a signal is the sum of the responses to all the impulses that constitute the signal, (ii) an impulse input contains all frequencies with equal energy, and hence it excites a filter at all frequencies and (iii) impulse response and frequency response are Fourier transform pairs.

Different Methods for Describing of Digital Filters

A useful method of gaining insight into the behavior of a filter is the pole-zero description of a filter. As described in Sec. X poles and zeros are the roots of the denominator and numerator of the transfer function respectively.

(d) *Frequency Response.* The frequency response of a filter describes how the filter alters the magnitude and phase of the input signal frequencies. The frequency response of a filter can be obtained by taking the Fourier transform of the impulse response of the filter, or by simple substitution of the frequency variable $e^{j\omega}$ for the z variable $z = e^{j\omega}$ in the z-transfer function as

$$H(z = e^{j\omega}) = \frac{Y(e^{j\omega})}{X(e^{j\omega})}$$

The frequency response of a filter is a complex variable and can be described in terms of the filter magnitude response and the phase response of the filter.

Classification of Digital Filters

Depending on the form of the filter equation and the structure of implementation, filters may be broadly classified into the following classes:

(a) Linear filters versus nonlinear filters.

(b) Time-invariant filters versus time-varying filters.

(c) Adaptive filters versus non-adaptive filters.

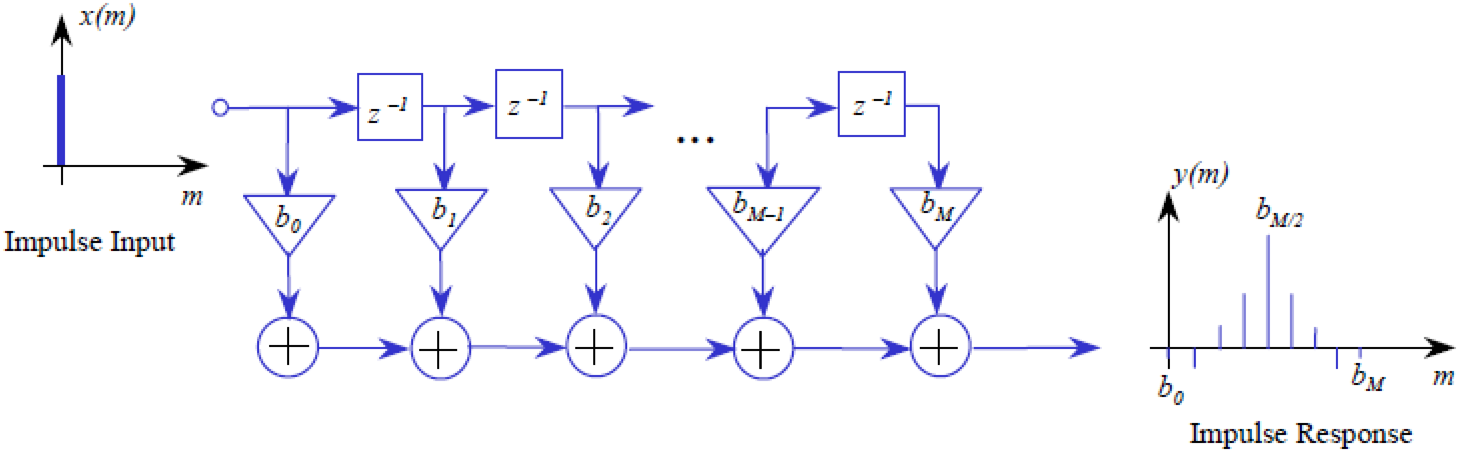
(d) Recursive versus non-recursive filters.

(e) Direct-form, cascade-form, parallel-form and lattice structures.

Filter Structures:
Direct, Cascade and Parallel Forms

We consider different structures for realization of a digital filter. These structures offer various trade-offs between complexity, cost of implementation, computational efficiency and stability.

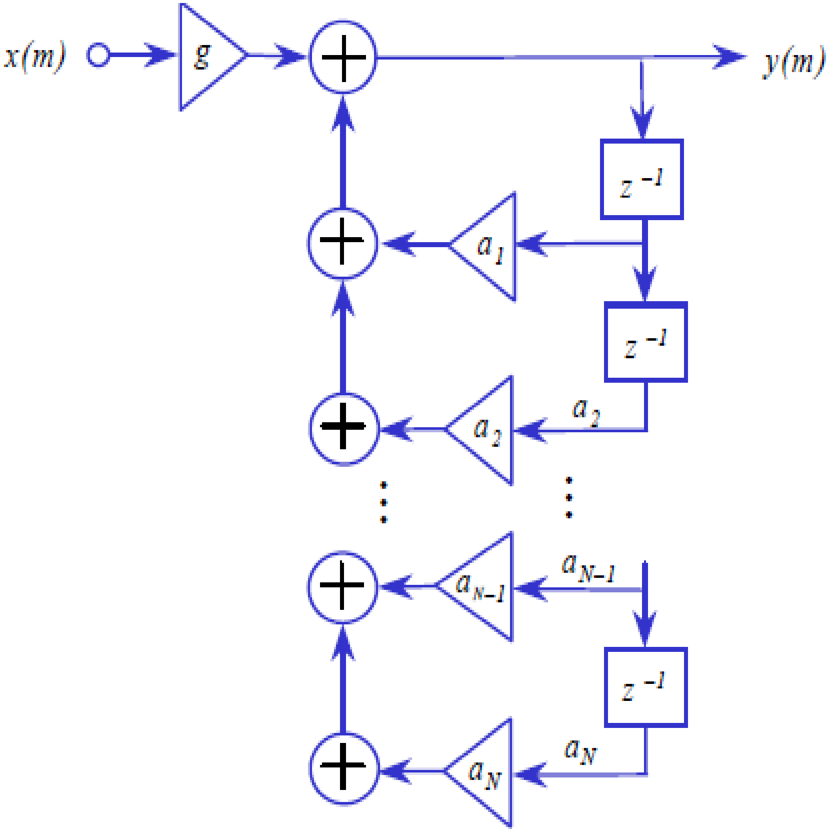
1. Direct Filter Structure



Direct-form Finite Impulse Response (FIR) filter.

Filter Structures: Direct, Cascade and Parallel Forms

1. Direct Filter Structure



Direct-form all-pole IIR filter.

Filter Structures:
Direct, Cascade and Parallel Forms

2. Cascade Filter Structure

The cascade implementation of a filter is obtained by expressing the filter transfer function $H(z)$ in a factorised form as

$$H(z)=\frac{\sum_{k=0}^{M} b_k z^{-k}}{1-\sum_{k=1}^{N} a_k z^{-k}}=G\frac{(1-z_1z^{-1})(1-z_1^*z^{-1})(1-z_2z^{-1})(1-z_2^*z^{-1})\cdots(1-z_{M/2}z^{-1})(1-z_{M/2}^*z^{-1})}{(1-p_1z^{-1})(1-p_1^*z^{-1})(1-p_2z^{-1})(1-p_2^*z^{-1})\cdots(1-p_{N/2}z^{-1})(1-p_{N/2}^*z^{-1})}$$

where G is the filter gain and the poles (p_ks) and zeros (z_ks) are either complex conjugate pairs or real-valued. The factorised terms in Equation can be grouped in terms of the complex conjugate pairs and expressed as cascades of second order terms as

Filter Structures:
Direct, Cascade and Parallel Forms

2. Cascade Filter Structure

$$H(z)=G\left(\frac{(1-z_1z^{-1})(1-z_1^*z^{-1})}{(1-p_1z^{-1})(1-p_1^*z^{-1})}\right)\times\left(\frac{(1-z_2z^{-1})(1-z_2^*z^{-1})}{(1-p_2z^{-1})(1-p_2^*z^{-1})}\right)\times\cdots\times\left(\frac{(1-z_{M/2}z^{-1})(1-z_{M/2}^*z^{-1})}{(1-p_{N/2}z^{-1})(1-p_{N/2}^*z^{-1})}\right)$$

Each bracketed term in Eq. is the z-transfer function of a second order IIR filter can be expressed in a compact notation as

$$H(z)=G\prod_{k=1}^{K}H_k(z)$$

Filter Structures:
Direct, Cascade and Parallel Forms

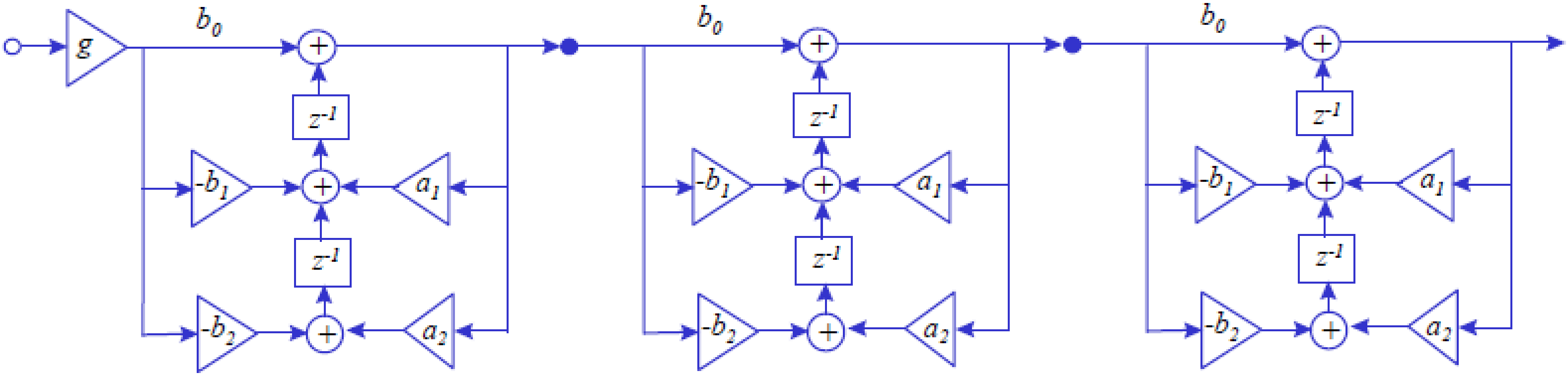
2. Cascade Filter Structure

For an IIR filter each second order cascade section has the form

$$H_k(z) = \frac{1 + b_k z^{-1} + b_k z^{-2}}{1 + a_{k1} z^{-1} + a_{k2} z^{-2}}$$

For an FIR filter, as shown in Fig. 4.8.b, each second order cascade section has the form

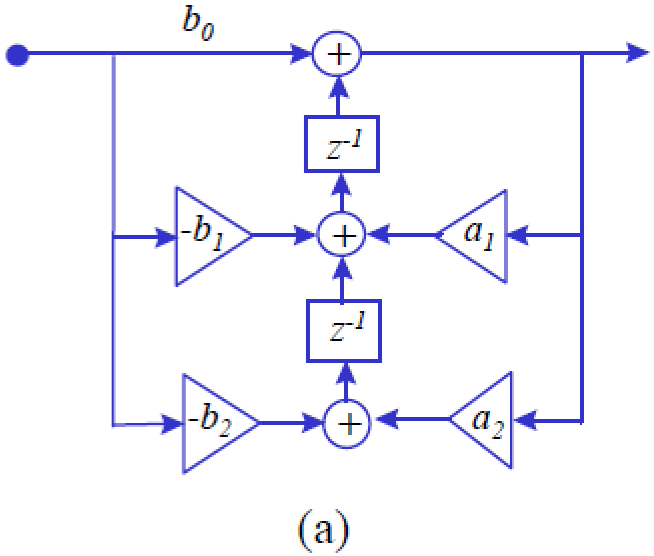
$$H_k(z) = 1 + b_k z^{-1} + b_k z^{-2}$$

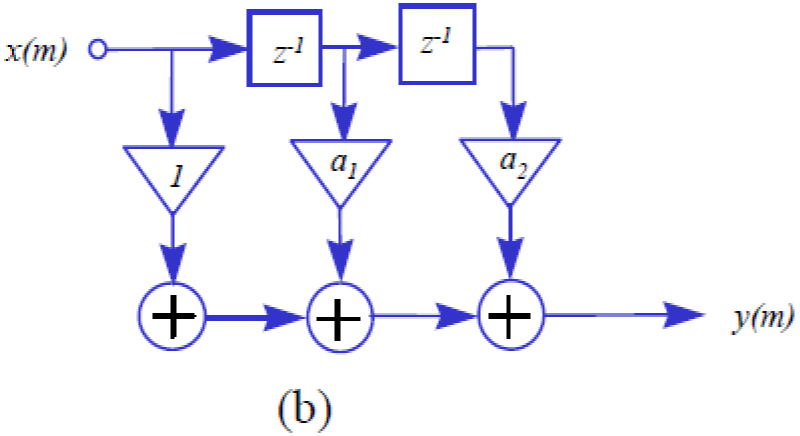


Realization of an IIR cascade structure from second order sections.

Filter Structures: Direct, Cascade and Parallel Forms

2. Cascade Filter Structure





Realization of a second order section of: (a) an IIR Filter, (b) an FIR filter.

Filter Structures: Direct, Cascade and Parallel Forms

3. Parallel Filter Structure

An alternative to the cascade implementation described in the previous section is to express the filter transfer function $H(z)$, using the partial fraction method, in a parallel form as parallel sum of a number of second order and first order terms as

$$H(z)=K+\sum_{k=1}^{N_1}\frac{e_{0k}+e_{1k}z^{-1}}{1-a_{1k}z^{-1}-a_{2k}z^{-2}}+\sum_{k=1}^{N_2}\frac{e_k}{1-a_kz^{-1}}$$

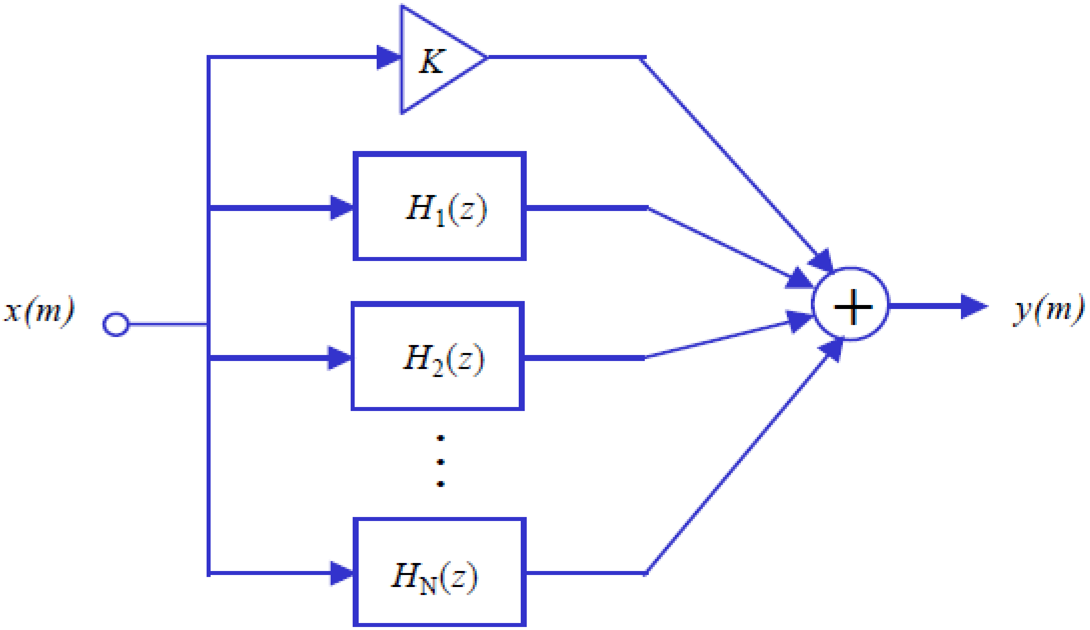
or

$$H(z)=K+\sum_{k=1}^{N_1+N_2}H_k(z)$$

where in general the filter is assumed to have N_1 complex conjugate poles, N_1 real zeros, and N_2 real poles and K is a constant. Figure shows a parallel filter structure.

Filter Structures: Direct, Cascade and Parallel Forms

3. Parallel Filter Structure

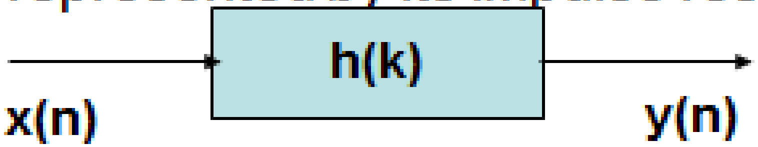


A parallel-form filter structure.

Types of Digital Filters

1. Finite impulse response.
2. Infinite impulse response.

- **FIR Digital Filter:**
  - FIR can be represented by its impulse response sequence h(k)



  - The i/p and o/p signals to the FIR filter are related by the convolution sum, given as

$$y(n) = \sum_{n=0}^{N-1} h(k)x(n-k) \qquad \rightarrow (1)$$

OR

$$y(n) = \sum_{k=0}^{N} b_k x(n-k) = b_0 x_n + b_1 x_{n-1} + \ldots b_N x_{n-N}$$

Types of Digital Filters

A non-recursive filter has no feedback and its input-output relation is given by

$$y(m)=\sum_{k=0}^{M} b_k x(m-k)$$

As shown in Fig   the output $y(m)$ of a non-recursive filter is a function only of the input signal $x(m)$. The response of such a filter to an impulse consists of a finite sequence of M+1 samples, where M is the filter order. Hence, the filter is known as a *Finite-Duration Impulse Response* (FIR) filter. Other names for a non-recursive filter include all-zero filter, feed-forward filter or moving average (MA) filter a term usually used in statistical signal processing literature.

Types of Digital Filter

- As we can see, the impulse response of FIR is of finite duration since FIR has only N values.
- Eq:1 is the difference eq: of FIR filter.
- In FIR filter the current o/p sample y(n) is a function only of past i/ps & present i/ps.
- Alternative representation of FIR filter is

$$H(z) = \sum_{k=0}^{N-1} h(k) z^{-k}$$

This is the transfer function of FIR and it's very useful in evaluating the freq: response.

Types of Digital Filters

IIR Digital Filter:

- These filters can be represented by

$$y(n)=\sum_{k=0}^{\infty} h(k)x(n-k)$$

$$y(n)=\sum_{k=0}^{N} b_k x(n-k)-\sum_{k=1}^{M} a_k y(n-k)$$

- a_k and b_k are coefficients of IIR filter.
- Choosing correct values of these coefficients
- In IIR filter eq:, o/p y(n) is a function of past o/ps, as well as present i/ps & past i/ps samples, that is the IIR is a feedback system of some sort.
- Alternative representation of the filters is

$$H(z)=\frac{\sum_{k=0}^{N} b_k z^{-k}}{1+\sum_{k=1}^{M} a_k z^{-k}}$$

Types of Digital Filters

A recursive filter has feedback from output to input, and in general its output is a function of the previous output samples and the present and past input samples as described by the following equation

$$y(m)=\sum_{k=1}^{N} a_k y(m-k)+\sum_{k=0}^{M} b_k x(m-k)$$

Fig shows a direct form implementation of Eq. . In theory, when a recursive filter is excited by an impulse, the output persists forever. Thus a recursive filter is also known as an *Infinite Duration Impulse Response* (IIR) filter. Other names for an IIR filter include feedback filters, pole-zero filters and

Types of Digital Filters

auto-regressive-moving-average (ARMA) filter a term usually used in statistical signal processing literature.

A discrete-time IIR filter has a z-domain transfer function that is the ratio of two z-transform polynomials as expressed in Eq. ; it has a number of poles corresponding to the roots of the denominator polynomial and it may also have a number of zeros corresponding to the roots of the numerator polynomial.

The main difference between IIR filters and FIR filters is that an IIR filter is more compact in that it can usually achieve a prescribed frequency response with a smaller number of coefficients than an FIR filter. A smaller number of filter coefficients imply less storage requirements and faster calculation and a higher throughput. Therefore, generally IIR filters are more efficient in memory and computational requirements than FIR filters. However, it must be noted that an FIR filter is always stable, whereas an IIR filter can become unstable (for example if the poles of the IIR filter are outside the unit circle) and care must be taken in design of IIR filters to ensure stability.

Choosing Between IIR and FIR Filters

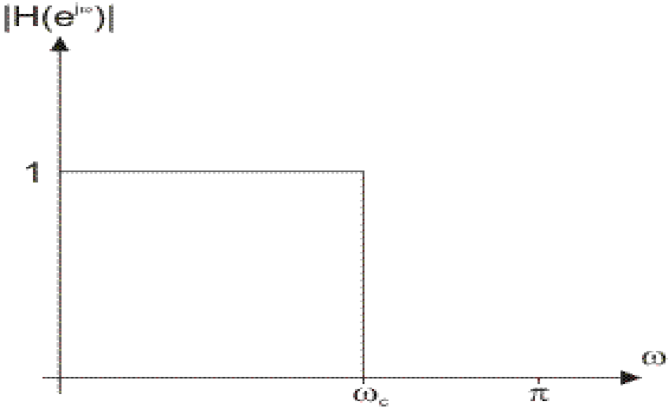
The choice b/w FIR & IIR filters depends largely on the relative adv: of the two filter types.

- FIR filters can have an exactly linear phase response i-e no phase distortion is introduced into the signal by the filter
- The phase responses of IIR filters are nonlinear, especially at the band edges.
- FIR filters are always stable. The stability of IIR filters cannot be always guaranteed.
- Effect of finite wordlength is less severe in FIR than in IIR.
- Analog filters can be readily transformed into IIR filter. This is not possible with FIR filters.

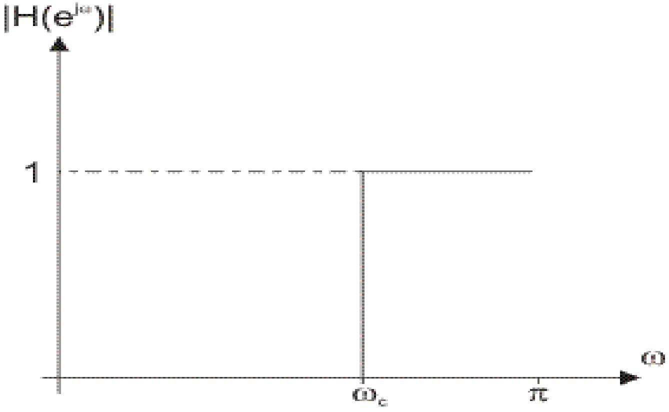
Filter Design Steps

1. **Specification of the filter requirements**
   1. Signal characteristics
   2. Data rates
   3. Desired amplitude and phase
   4. Manner of implementation etc
2. **Calculation of suitable filter coefficients**
   1. Impulse invariant (for IIR)
   2. Bilinear transformation (for IIR)
   3. Pole-zero placement (for IIR)
   4. Window (for FIR)
   5. Frequency sampling (for FIR)
   6. Optimal method (for FIR)
3. **Representation of the filter by a suitable structure**
   1. Transversal (for FIR)
   2. Linear phase (for FIR)
   3. Cascade (for IIR)
   4. Parallel (for IIR)
   5. Direct form (for IIR)
4. **Analysis of effects of finite word length on filter performance.**
5. **Implementation of filter in filter in software/hardware**

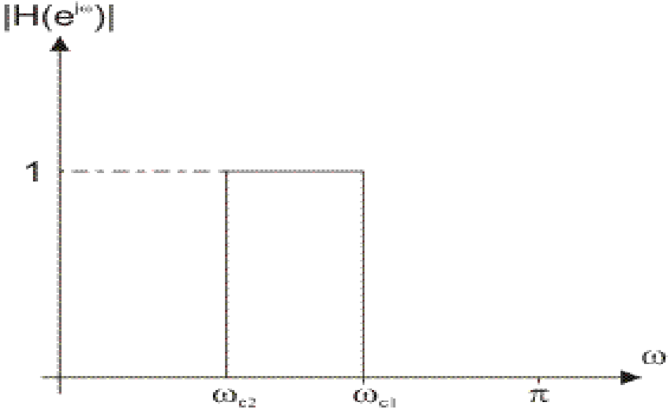
Kinds of Filters



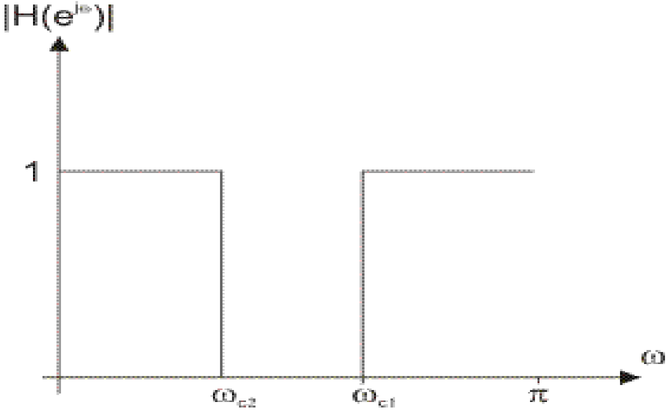
Low-pass Filter



High-pass Filter

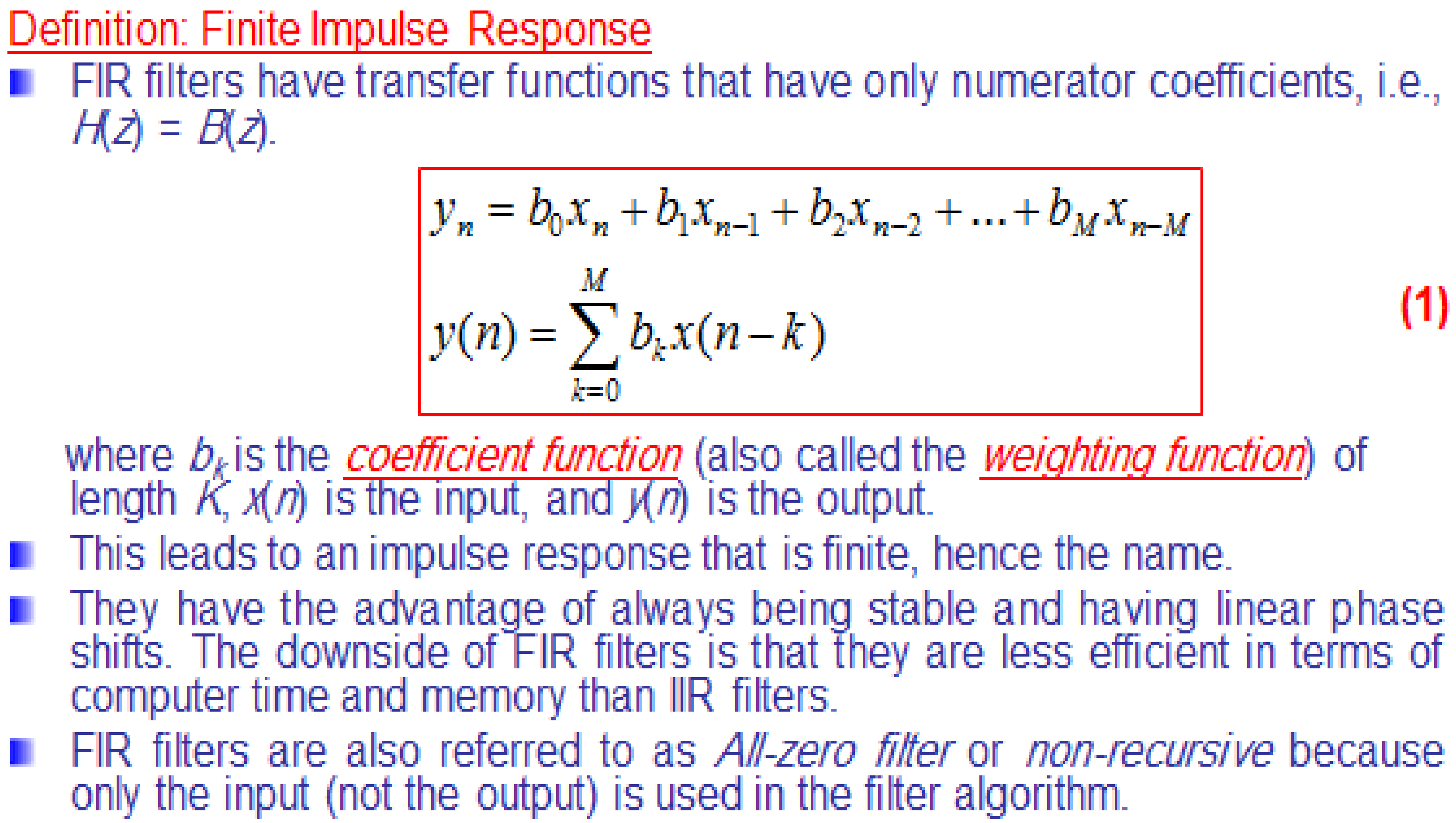


Band-pass Filter



Stop-pass Filter

Finite Impulse Response (FIR) Filters



Definition: Finite Impulse Response

- FIR filters have transfer functions that have only numerator coefficients, i.e., $H(z) = B(z)$.

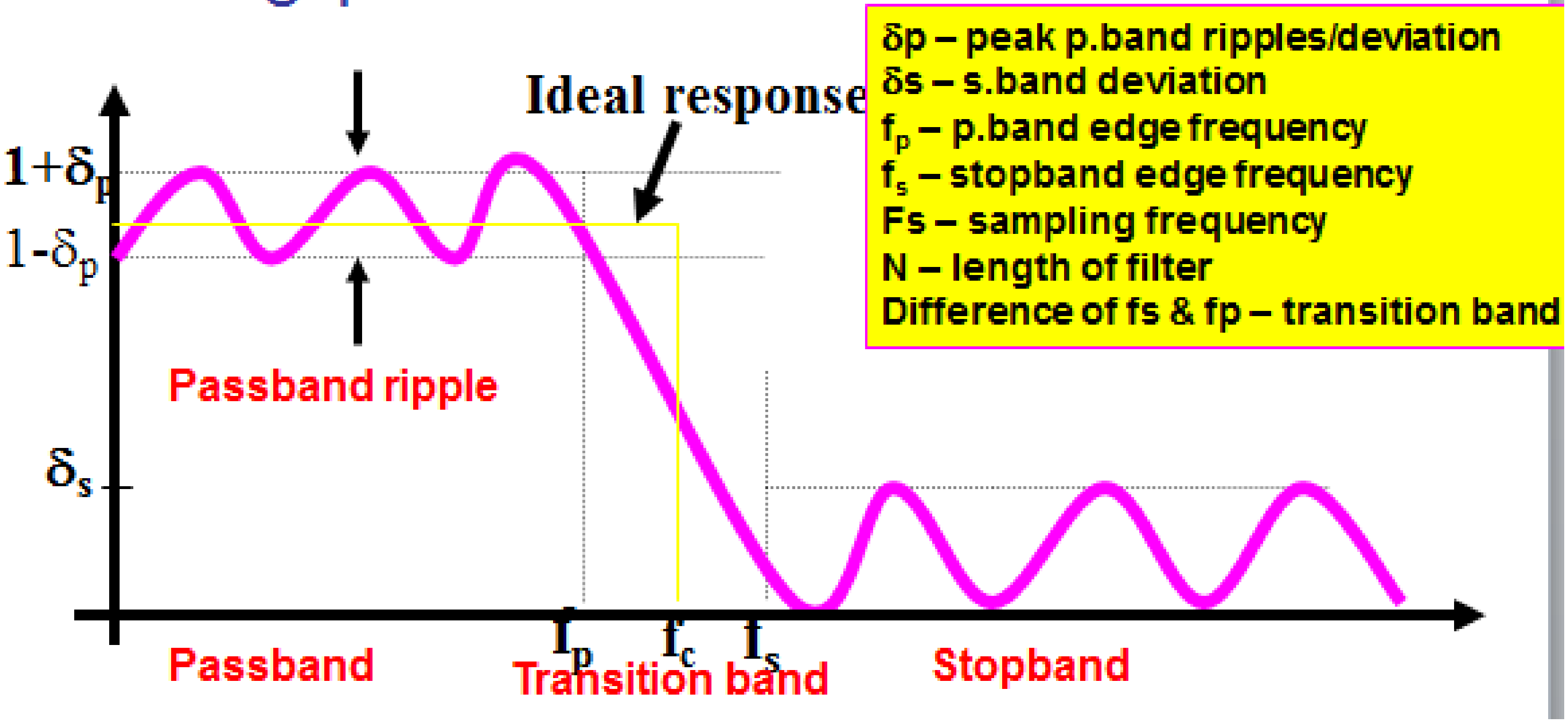
$$\boxed{\begin{aligned} y_n &= b_0 x_n + b_1 x_{n-1} + b_2 x_{n-2} + \ldots + b_M x_{n-M} \\ y(n) &= \sum_{k=0}^{M} b_k x(n-k) \end{aligned}} \qquad (1)$$

where b_k is the *coefficient function* (also called the *weighting function*) of length K, $x(n)$ is the input, and $y(n)$ is the output.

- This leads to an impulse response that is finite, hence the name.
- They have the advantage of always being stable and having linear phase shifts. The downside of FIR filters is that they are less efficient in terms of computer time and memory than IIR filters.
- FIR filters are also referred to as *All-zero filter* or *non-recursive* because only the input (not the output) is used in the filter algorithm.

STEP1:FIR Filter Specification

- Referring to this figure given below the following parameters are of interest.



δp – peak p.band ripples/deviation
δs – s.band deviation
f_p – p.band edge frequency
f_s – stopband edge frequency
Fs – sampling frequency
N – length of filter
Difference of fs & fp – transition band

Specification Example

- This example provides an idea of how digital filter specifications are defined.
  - A low pass (LP) digital filter is required for physiological noise reduction. The filter should meet the following specifications.
    - Passband edge frequency 10Hz
    - Stopband edge frequency <20Hz
    - Stopband attenuation >30dB
    - Passband ripple <0.026dB
    - Sampling frequency 256Hz

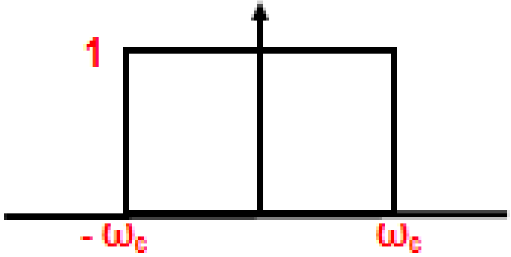
Step 2: Calculation of Suitable Filter Coefficients

- Ideally we wish to realize a “brick-wall” magnitude response and a linear phase response. In practice we can realize a reasonable approximation to this magnitude response, together with a linear phase response.
- The design problem may be stated, thus, given an ideal/desired frequency response $H(\omega)$,
  - decide on a filter length, and
  - decide on a filter coefficients which will give an actual response $H(\omega)$ approximating to $H_d(\omega)$ to within a given specification in magnitude and phase. (d subscript represent desired)

Design of FIR filter using Fourier Methods

- Consider the ideal low-pass zero-delay filter which is a rectangular window in the frequency domain with following specifications:

$$H_d(\omega)=\begin{cases}1, & -\omega_c<\omega<\omega_c\\ 0, & |\omega|>\omega_c\end{cases} \quad (2)$$



where ω_c is the cutoff frequency.

- The inverse Fourier transform of a rectangular window (square) function is calculated as

Design of FIR filter using Fourier Methods

Remind Fourier transform Pair!

$$H(\omega) = \sum_{n=-\infty}^{\infty} h[n]e^{-jwn}$$

$$h[n] = \frac{1}{2\pi}\int_{2\pi} H(\omega)e^{j\omega n}d\omega$$

On substituting (2) into this, we get

$$h_d[n] = \frac{1}{2\pi}\int_{-\omega_c}^{\omega_c} 1.e^{j\omega n}d\omega$$

which evaluates to

$$h_d[n] = \frac{1}{n\pi}\sin(n\omega_c) \qquad -\infty < n < \infty \qquad \textbf{(3)}$$

- A filter with the impulse response Eq. (3) would realize the ideal specification (2) exactly. However, it poses two problems
  - A- This impulse response is infinite in length i.e. the filter would require an infinite number of coefficients.
  - B- This impulse response is double-sided as shown in the fig 2 on the next slide. If we truncate the series symmetrically, it will have terms for negative n and the filter will be non-causal i.e. will require foreknowledge. Therefore, it will not be realizable in real world.

Design of FIR filter using Fourier Methods

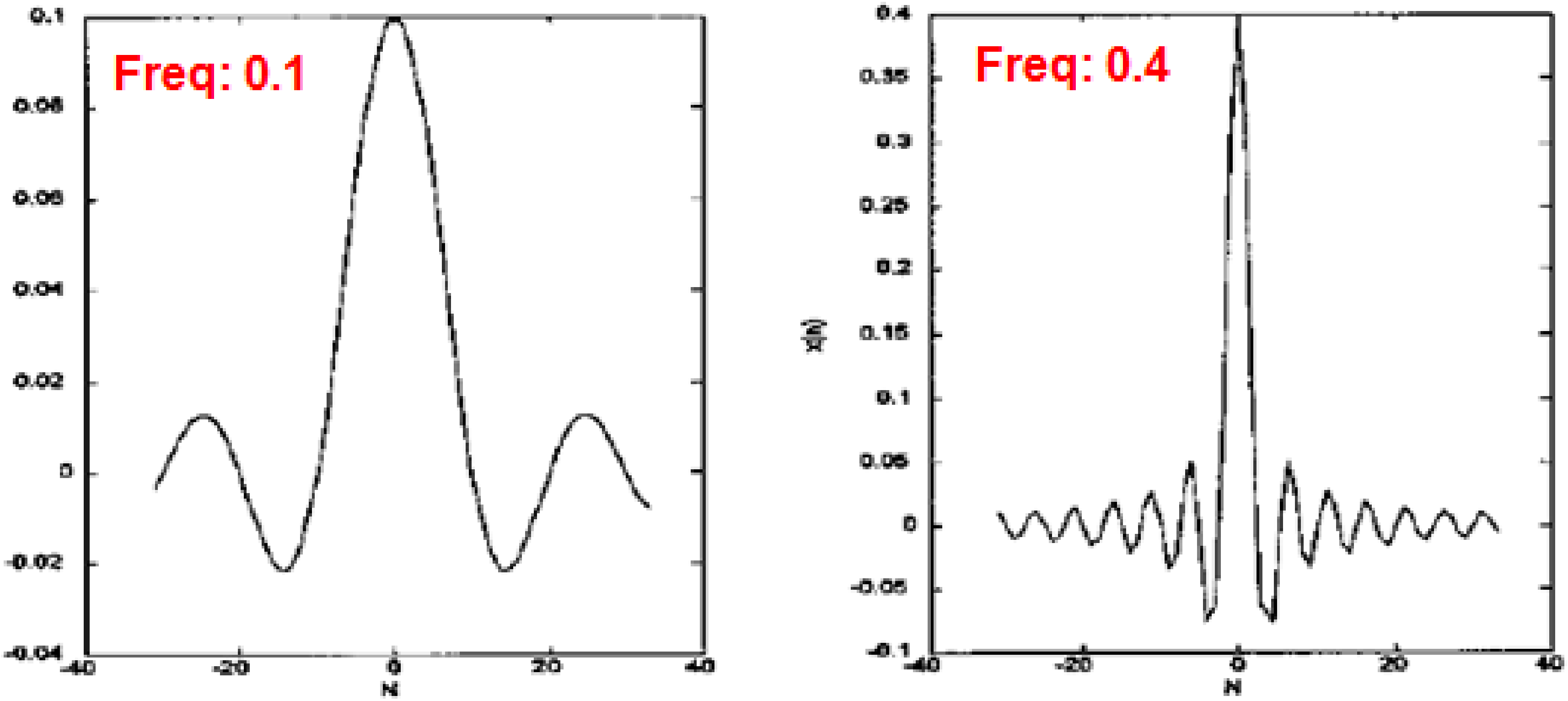


FIGURE 2 Symmetrical weighting function of a rectangular filter Eq. (8) truncated at 64 coefficients. Left: Lowpass filter with a cutoff frequency of 0.1*fs* /2 Hz. Right: Lowpass cutoff frequency of 0.4*fs* /2 Hz.

Design of FIR filter using Fourier Methods

- Solution – A: Truncate to a finite number of coefficients. The response will now be an approximation to the ideal.
  - Figure 3 shows the frequency response, obtained by taking the Fourier transform of the coefficients for two different lengths. This filter also shows a couple of artifacts associated with finite length:
    - An oscillation in the frequency curve which increases in frequency when the coefficient function is longer. This effect is called Gibbs Oscillation.
    - A peak in the pass-band which becomes narrower and higher when the coefficient function is lengthened.
- Solution – B: Shift the impulse response by N sampling intervals in time, where N = K/2. On denoting the shifted response by h[n], this gives

$$h[n] = h_d(n\text{-}N)$$

and now we have a standard finite length causal filter.

  - This will have the effect of delaying the response by N sampling intervals. Instead of the zero delay filter of (2), we will have a constant delay filter with delay time τ. This is equivalent to a linear phase filter, the phase being given by $\phi = 2\pi f\tau$ radians.
- On taking account of the truncation and the shift Eq. (3) becomes

$$h[n] = \frac{1}{(n-N)\pi}\sin((n-N)\omega_c) = \frac{\omega_c}{\pi}\frac{\sin((n-K/2)\omega_c)}{(n-K/2)\omega_c} = \frac{\omega_c}{\pi}\operatorname{sinc}((n-K/2)\omega_c) \qquad \text{(4)}$$

Artifacts of Truncation

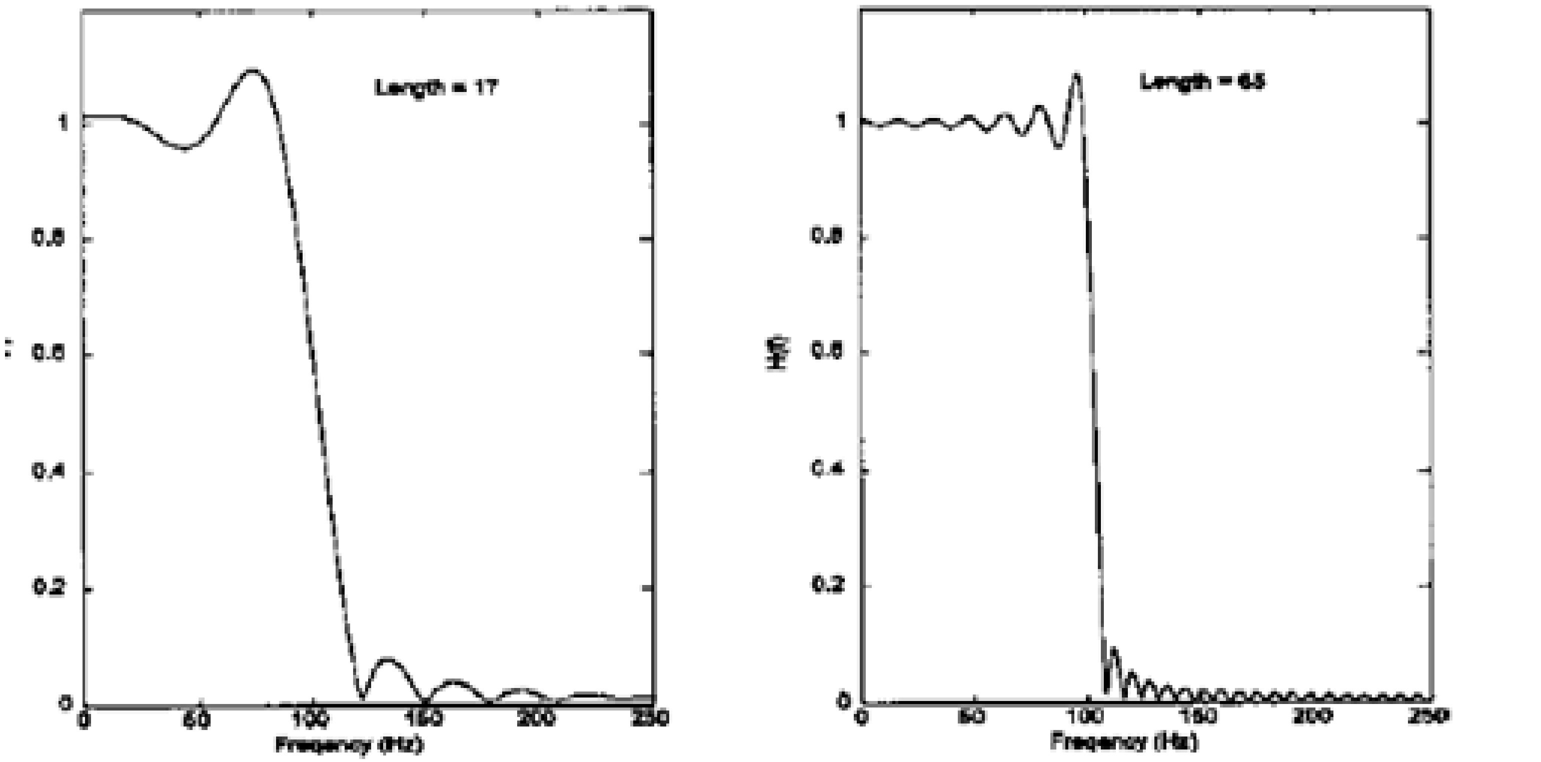


FIGURE 3 Frequency characteristics of an FIR filter based in a weighting function derived from Eq. (4). The weighting functions were abruptly truncated at 17 and 65 coefficients. The low-pass cutoff frequency is 100 Hz.

Reducing the Effect of Truncation

- Increasing the filter length has relatively little effect. A better method is to window the impulse response: each term h[n] is multiplied by the corresponding term w[n] in a window:

$$h_w[n] = h[n]w[n]$$

  Many windows have been proposed, which are described next.

- The purpose of a window function is to select a finite number of samples from the infinite no: of samples.

Designing FIR Filters: Windows

Rectangular Window:

- The rectangular window is defined as:

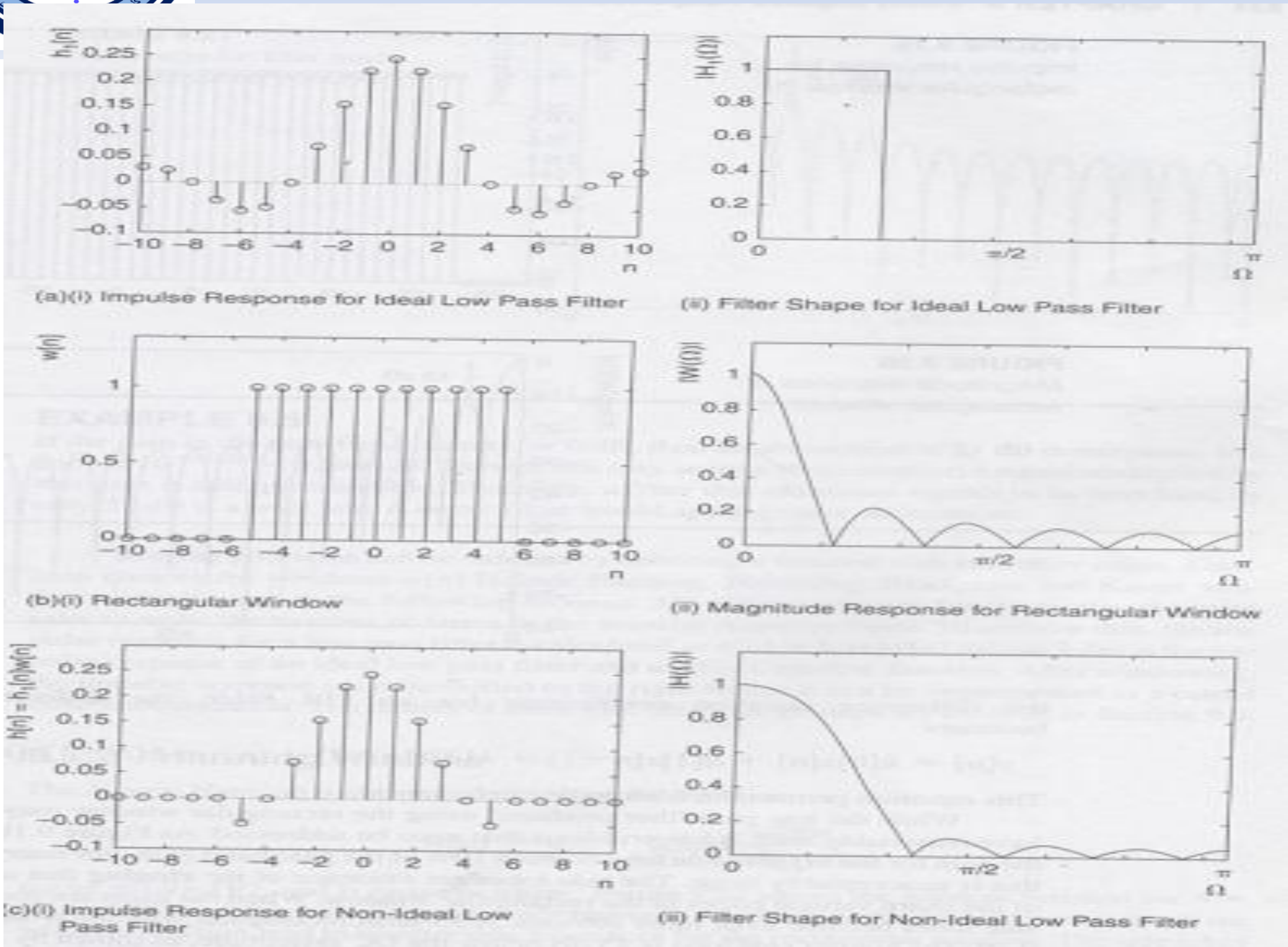
$$\boxed{w[n]=\begin{cases}1 & 0\le n\le K\\ 0 & elsewhere\end{cases}} \qquad (5)$$

- The filter coefficients b[n] = h[n] can now be computed as:

$$b[n]=h_d[n-N]w[n]=\frac{\omega_c}{\pi}\frac{\sin((n-N)\omega_c)}{(n-N)\omega_c} \qquad (6)$$

- The side lobes in the mag: resp: of rectangular window are evidence of **ringing** that occurs because of the sharp vertical edges of the rectangular window.
- Ringing problem can be reduced by choosing a window with smoother edges, such windows are hamming, hanning, balckman and kaiser described next.

Windows Method



(a)(i) Impulse Response for Ideal Low Pass Filter

(ii) Filter Shape for Ideal Low Pass Filter

(b)(i) Rectangular Window

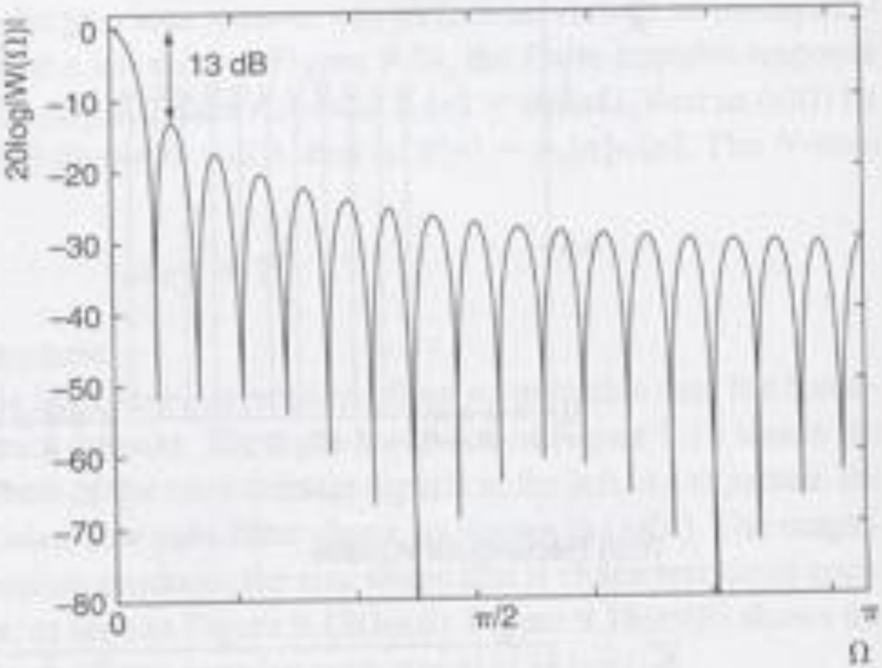
(ii) Magnitude Response for Rectangular Window

(c)(i) Impulse Response for Non-Ideal Low Pass Filter

(ii) Filter Shape for Non-Ideal Low Pass Filter

Windows Method

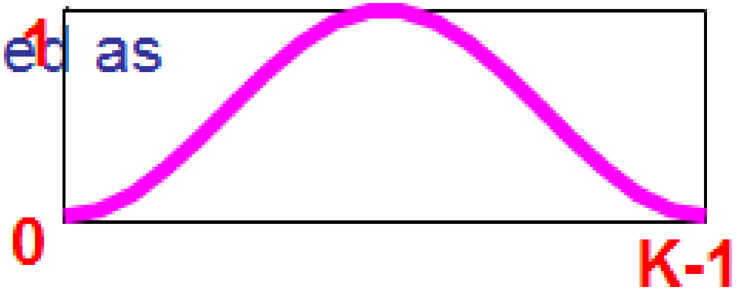
FIGURE 9.20
Magnitude response of rectangular window.



Hamming Window

Hamming Window:

- The Hamming window is defined as



$$w[n] = 0.54 + 0.46\cos\left(\frac{2\pi n}{N-1}\right) \qquad (7)$$

- The filter coefficients can be found as follows: h[n] = hd[n – N]×w[n]

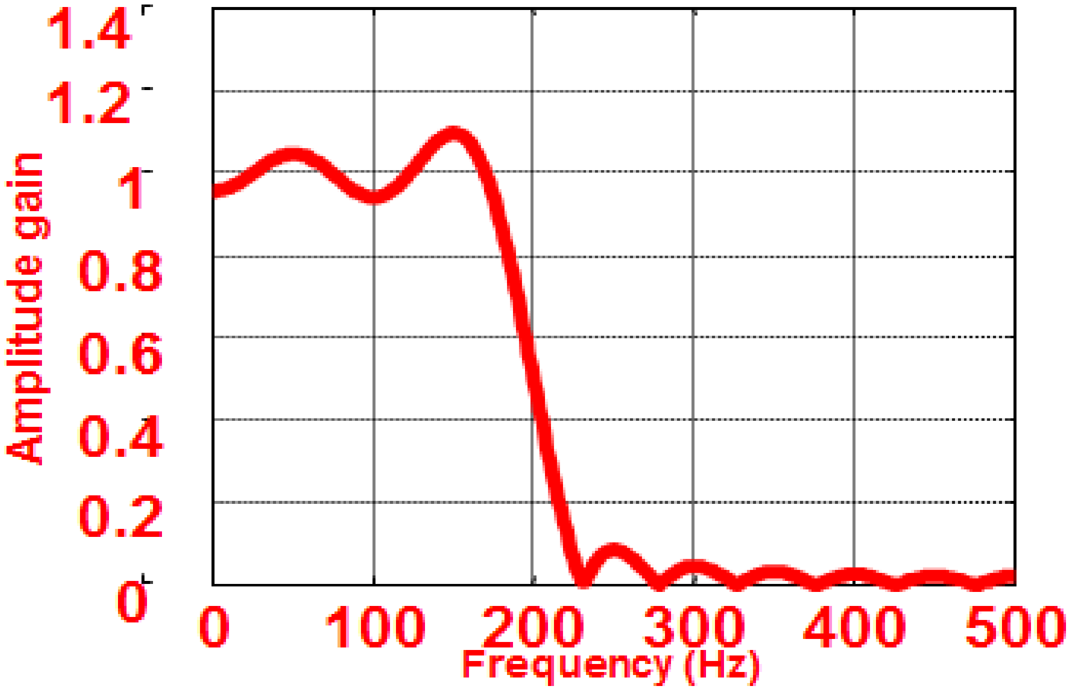
$$b[n] = \left[\frac{\omega_c}{\pi}\,\frac{\sin(n-K/2)\omega_c}{(n-K)\omega_c}\right]\left[0.54 + 0.46\cos\left(\frac{2\pi n}{N-1}\right)\right] \qquad (8)$$

Example1

Example 1: Design a low-pass FIR filter with cut-off frequency at 200 Hz relative to a Nyquist frequency of 1KHz. Use the rectangular window to truncate the impulse response. The length of the filter is 21.

Solution 1: The normalized cutoff frequency is $\omega_c=2\pi f_c/f_s=2\pi(200/(1000)= 0.4\pi$ where N = 20

b[0] = 0.0000 = b[20]
b[1] = -0.03364 = b[19]
b[2] = -0.02339 = b[18]
b[3] = 0.02673 = b[17]
b[4] = 0.05046 = b[16]
b[5] = 0.00000 = b[15]
b[6] =-0.07568 = b[14]
b[7] =-0.06237 = b[13]
b[8] =0.09355 = b[12]
b[9] = 0.30273 = b[11]
b[10]= 0.40000



Example2

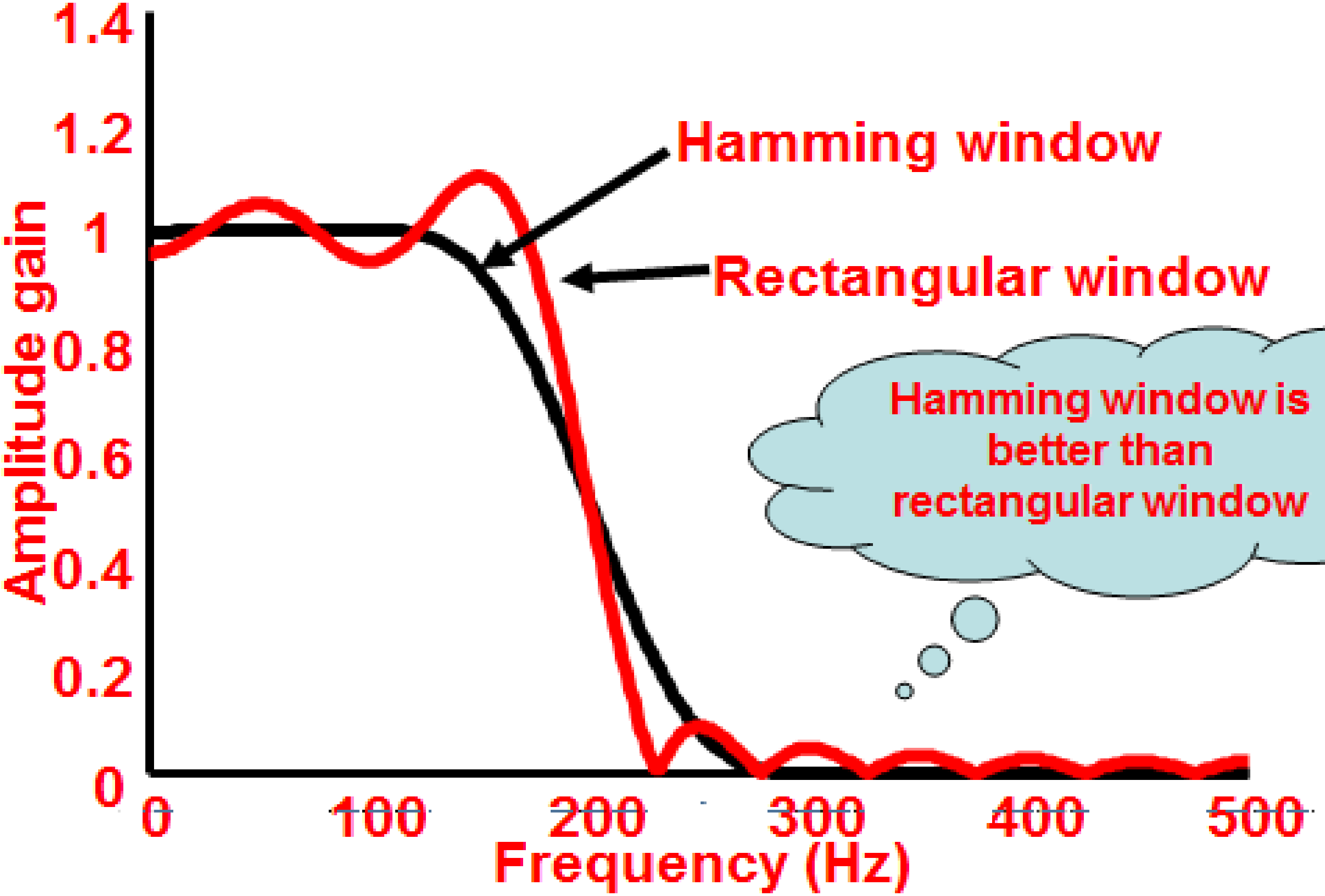
Example 2: Repeat example 1 for a Hamming window.

Solution 2: The filter coefficients can be computed by using equation (16) as:

b[0] = 0.00000 = b[20]   b[1] = -0.00345 = b[19]
b[2] = -0.00393 = b[18]   b[3] = 0.00721 = b[17]
b[4] = 0.02007 = b[16]   b[5] = 0.00000 = b[15]
b[6] = -0.05163 = b[14]   b[7] = -0.05054 = b[13]
b[8] = 0.08533 = b[12]   b[9] = 0.29592 = b[11]
b[10]= 0.40000

The magnitude response of the filter and its comparison with the magnitude response of the filter with rectangular window is given in the figure on the next slide.

Hamming Window



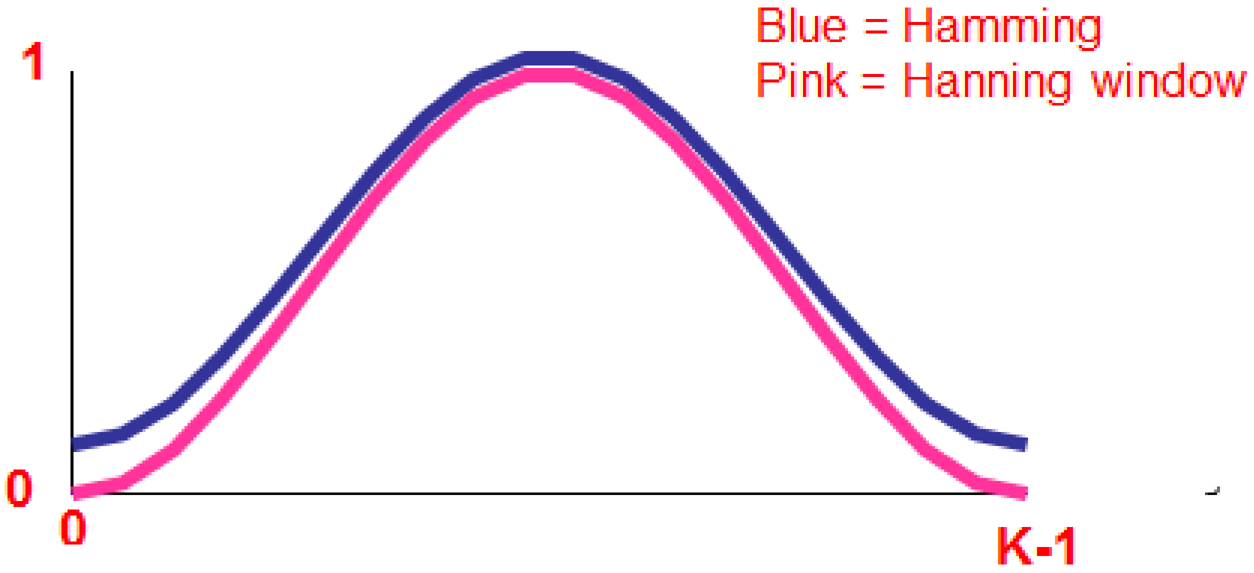
Hamming window is better than rectangular window

Hanning Window

Hanning Window:

- The Hanning Window is defined as

$$w[n] = 0.5\left[1+\cos\left(\frac{2\pi n}{N-1}\right)\right] \qquad (9)$$



Hanning Window: Example3

Example 3: Repeat Example 1 with the Hann Window

Solution 3: The coefficients of the filter in this case can be computed as follows:

$$b[n] = \left[\frac{\omega_c}{\pi}\cdot\frac{\sin\left(n-K/2\right)\omega_c}{\left(n-K\right)\omega_c}\right]\left[0.5\left\{1-\cos\left(\frac{2\pi n}{N-1}\right)\right\}\right]$$

The coefficients are:

b[0] = 0.00000 = b[20]
b[1] = -0.00082 = b[19]
b[2] = -0.00223 = b[18]
b[3] = 0.00551 = b[17]
b[4] = 0.01743 = b[16]
b[5] = 0.0000 = b[15]
b[6] = -0.04953 = b[14]
b[7] = -0.04951 = b[13]
b[8] = 0.08462 = b[12]
b[9] = 0.29532 = b[11]
b[10] = 0.40000

Blackman Window

Blackman Window:

- The blackman window is defined by the eq:

$$\boxed{w[n] = 0.42 + 0.5\cos\frac{2\pi n}{N-1} + 0.08\cos\frac{4\pi n}{N-1}}$$

- A LP filter designed with a Blackman window gives side lobes that are 75dB below the pass band as seen in figure here.

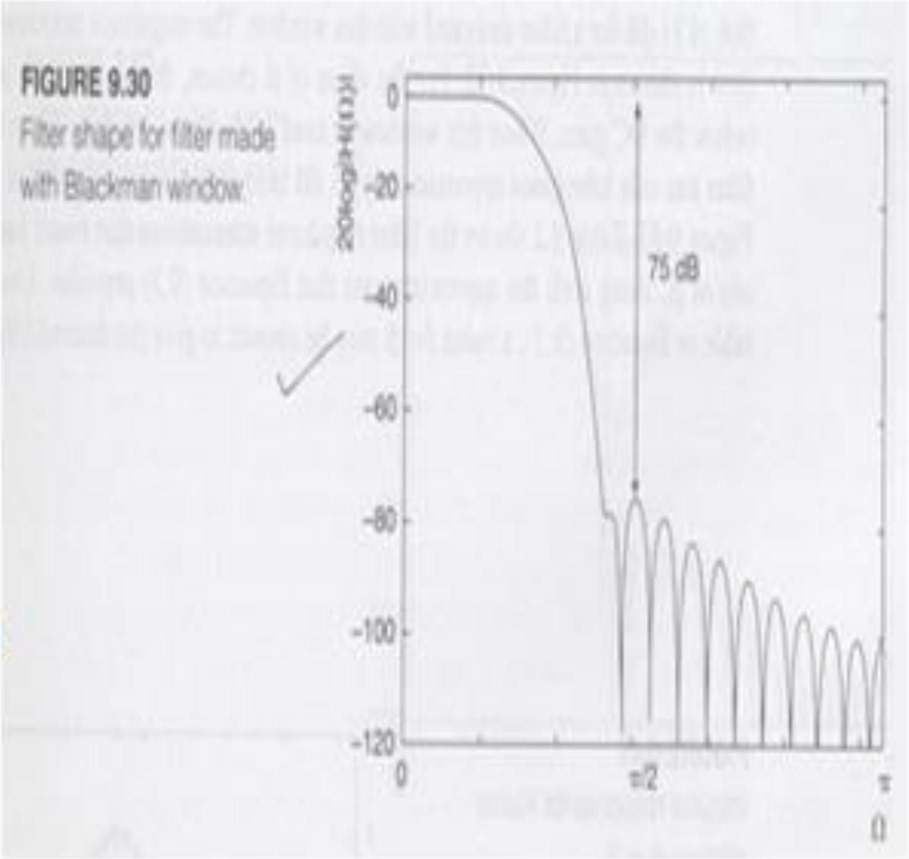
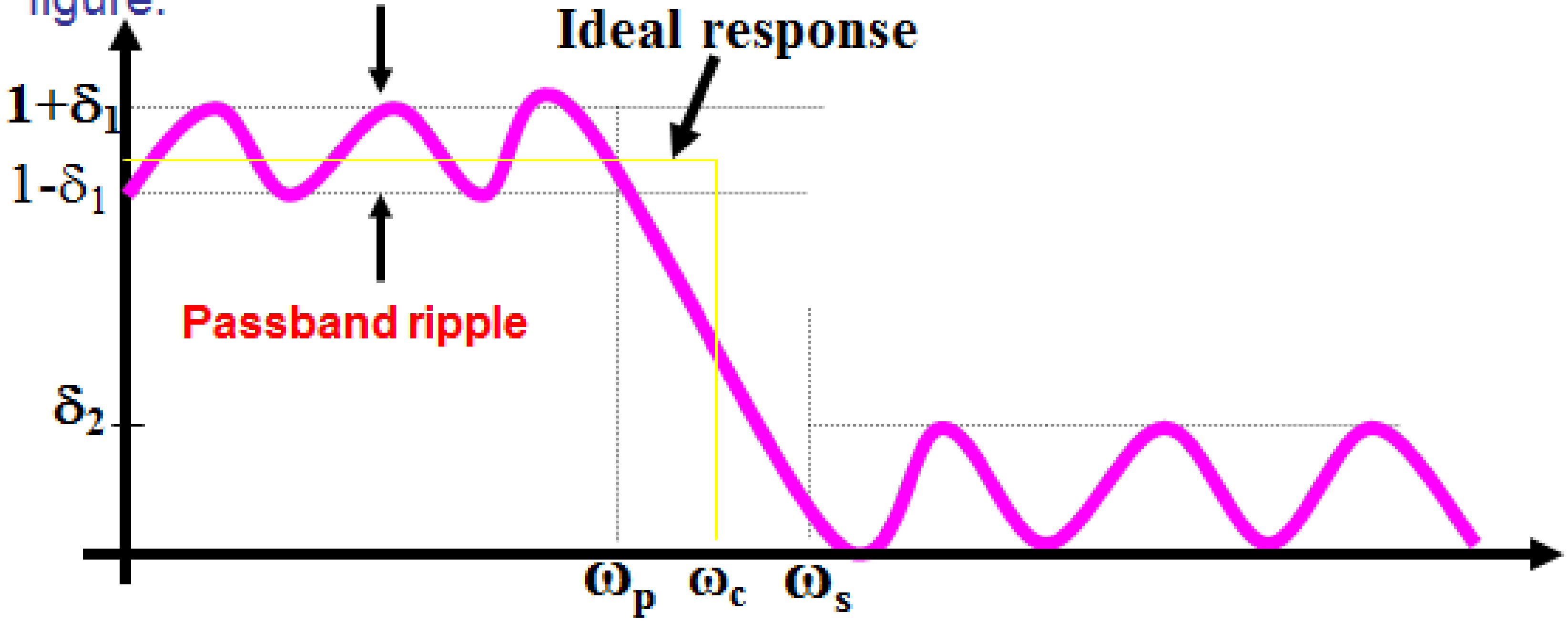


FIGURE 9.30
Filter shape for filter made with Blackman window

Kaiser Window

Kaiser Window:

- The windows studied so far are very simple, but do not provide good control over the filter design specifications. Kaiser window is a highly suitable for achieving the required pass-band and/or stop-band ripple, along with the cut-off requirements. Consider the following figure.



Kaiser Window

The passband/stopband frequencies $\{\omega_p, \omega_s\}$ are related to the ideal cutoff frequency ω_c and transition width $\Delta\omega$ by

$$\omega_c = ½\,(\omega_p + \omega_s), \qquad \Delta\omega = \omega_s - \omega_p \qquad (10)$$

The pass-band and stop-band overshoots $\{A_p, A_s\}$ are expressed in dB as

$$A_p = 20\log_{10}\frac{1+\delta_1}{1-\delta_1}, \qquad A_s = -20\log_{10}\delta_2 \qquad \textbf{(11)}$$

Equation (20) can be inverted to give:

$$\delta_1 = \frac{10^{A_p/20} - 1}{10^{Ap/20} + 1}, \qquad \delta_2 = 10^{-A_s/20} \qquad \textbf{(12)}$$

Although δ_1 and δ_2 can be specified independently of each other, it is a property of all windows designs that the final designed filter will have equal pass-band and stop-band ripples.

Kaiser Window

- Therefore, we must design the filter on the basis of the smaller of the two ripples, that is,

$$\delta=\min(\delta_1, \delta_2) \qquad (13)$$

- The Kaiser window is mathematically represented as:

$$\boxed{w[n] = I_0\left[\alpha\sqrt{\left(\tfrac{K-1}{2}\right)^2 - \left(n - \tfrac{K-1}{2}\right)^2}\,\right] \Big/ I_0\left[\alpha\left(\tfrac{K-1}{2}\right)\right]} \qquad \mathbf{(14)}$$

where $I_0(x)$ is the *modified Bessel function of the first kind and 0th order.*
The parameter α can be computed as follows:

$$\alpha = \begin{cases} 0.1102(A-8.7) & A \geq 50 \\ 0.5842(A-21) & 21 < A < 50 \\ 0 & A \leq 21 \end{cases} \qquad \mathbf{(15)}$$

- The filter length can be obtained as

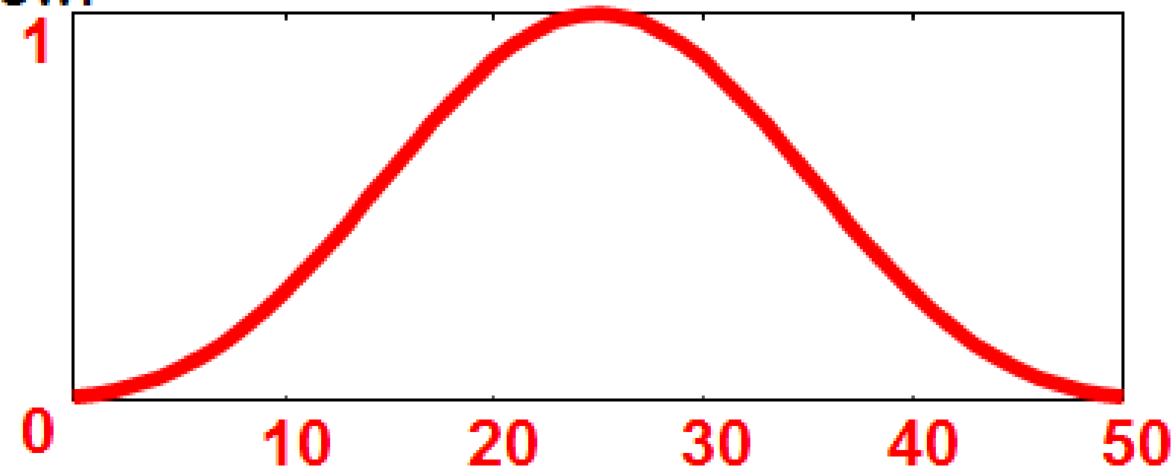
$$K-1 = \frac{Df_s}{\Delta f} \qquad \mathbf{(16)}$$

Kaiser Window

- where f_s is the sampling frequency in Hz, Δf is the transition Width in Hz and

$$D = \begin{cases} \frac{A-7.95}{14.36} & A > 21 \\ 0.922 & A \le 21 \end{cases} \qquad (17)$$

- The shape of the Kaiser window for $K = 51$ & $\alpha = 7$ is given below:



Kaiser Window: Example

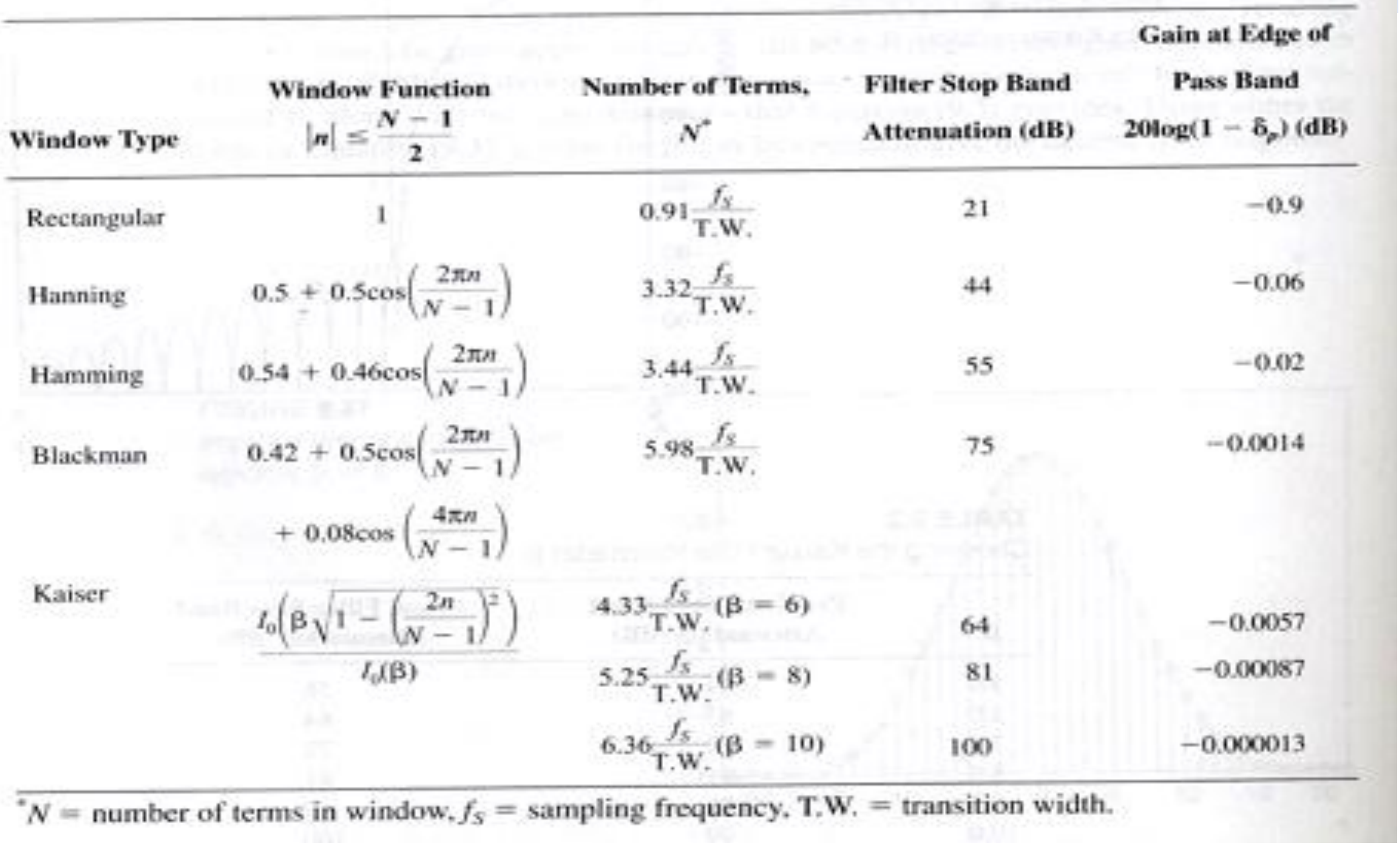
Example 4: Using Kaiser Window, design a low-pass digital filter with the following specifications:
Sampling frequency = 20 KHz, pass-band frequency = 4 KHz
Stop-band frequency = 5 KHz, A_p = 0.1 dB, A_s = 80 dB

Solution 4: $\delta_p = \dfrac{10^{A_p/20} - 1}{10^{A_p/20} + 1} = 0.0058,$ $\qquad \delta_s = 10^{-80/20} = 0.0001$

Therefore, $\delta = \min(\delta 1, \delta 2)$, which in db is $A = -20\log_{10} \delta s = 80$.
$\alpha = 0.1102(A - 8.7) = 7.857$, $D = (A - 7.95)/14.36 = 5.017$.
The transition width = $\Delta f = fs - fp = 5 - 4 = 1$KHz
fc = ½ (fs + fp) = 4.5 KHz, wc = $(2\pi fc/fs) = 0.45\pi$.
$K - 1 = Dfs/\Delta f \Rightarrow K = 103 \Rightarrow (K-1)/2 = 102/2 = 51$
The windowed impulse response will be

$$h[n] = \frac{\sin(0.45\pi(n-51))}{\pi(n-51)} \cdot \frac{I_0\left[7.857\sqrt{\left[\left(\frac{K-1}{2}\right)^2 - \left(n - \frac{K-1}{2}\right)^2\right]}\right]}{I_0\left[\alpha\left(\frac{K-1}{2}\right)\right]}$$

Summary of Windows



Window Type	Window Function $\lvert n\rvert \le \frac{N-1}{2}$	Number of Terms, N^*	Filter Stop Band Attenuation (dB)	Gain at Edge of Pass Band $20\log(1-\delta_p)$ (dB)
Rectangular	1	$0.91\frac{f_S}{\text{T.W.}}$	21	-0.9
Hanning	$0.5 + 0.5\cos\left(\frac{2\pi n}{N-1}\right)$	$3.32\frac{f_S}{\text{T.W.}}$	44	-0.06
Hamming	$0.54 + 0.46\cos\left(\frac{2\pi n}{N-1}\right)$	$3.44\frac{f_S}{\text{T.W.}}$	55	-0.02
Blackman	$0.42 + 0.5\cos\left(\frac{2\pi n}{N-1}\right)$ $+ 0.08\cos\left(\frac{4\pi n}{N-1}\right)$	$5.98\frac{f_S}{\text{T.W.}}$	75	-0.0014
Kaiser	$\frac{I_0\left(\beta\sqrt{1-\left(\frac{2n}{N-1}\right)^2}\right)}{I_0(\beta)}$	$4.33\frac{f_S}{\text{T.W.}}(\beta = 6)$	64	-0.0057
		$5.25\frac{f_S}{\text{T.W.}}(\beta = 8)$	81	-0.00087
		$6.36\frac{f_S}{\text{T.W.}}(\beta = 10)$	100	-0.000013

$^*N$ = number of terms in window, f_S = sampling frequency, T.W. = transition width.

Example

- Example: A filter must have a stop band attenuation of 75dB and a transition width of 1kHz for 16kHz sampling. Which window should be selected, and how many terms should be used?
- Solution: The two windows in table1 that best satisfy the stop band requirement are the Blackman and the beta =8 kaiser window.

The blackman window requires 5.98(16000/1000)=95.68 = 95terms. The beta=8 kaiser window requires 5.25(16000/1000)=84=85terms. In general, the number of terms are rounded off to the nearest odd integer.

Choosing Window Type

- The best choice is the window that meets the stop band requirements as closely as possible.
- Choosing an odd number of terms will gurantee a symmetrical impulse response, which will eliminate phase distortion.
- If the filter length (N), transition width, desired window etc are not provided in the specifications or problem statement then they also must be calculated by using the following steps

Steps for LP FIR filter design

1. Choose a pass band edge frequency in Hz for the filter by the formula

   f_1= desired pass band edge freq: + transition width/2

2. Calculate w = 2pif_1/f_s and and substitute this value in the impulse response of ideal LP filter eq: h[n] =sin(nw)/npi
3. Choose a window from table1: that will satisfy the stop band attenuation and other filter requirements.
4. Calculate the Finite impulse response h[n] for the filter from h[n]=h_1[n]w[n]
5. Shift the impulse response values to the right (N-1)/2 steps to ensure value occurs at n=0, thereby making the low pass filter causal.

Example

- Example: A LP filter must be designed according to the following specifications:
  - Pass band edge 2kHz
  - Stop band edge 3kHz
  - Stop band attenuation 40dB
  - Sampling frequency 10kHz

Solution:

transition width = stop band edge freq: - pass band edge freq:

= 3 – 2 = 1kHz

Example

Step1 of the design method then yields

f_1 = 2000 + 1000 /2 = 2500Hz

For step 2 , w = 2 pi f1/fs = 0.5pi

Step 3 demands a choise of window. For a stop band attenuation of 40dB, Table1 suggests the Hanning window, with

N = 3.32 f_S/Transition width = 33.2

Choosing N = 33, the window function becomes

$$w[n] = 0.5\left[1 + \cos\left(\frac{2\pi n}{32}\right)\right]$$

Calculating the h[n] will give the following values on next slide

Example

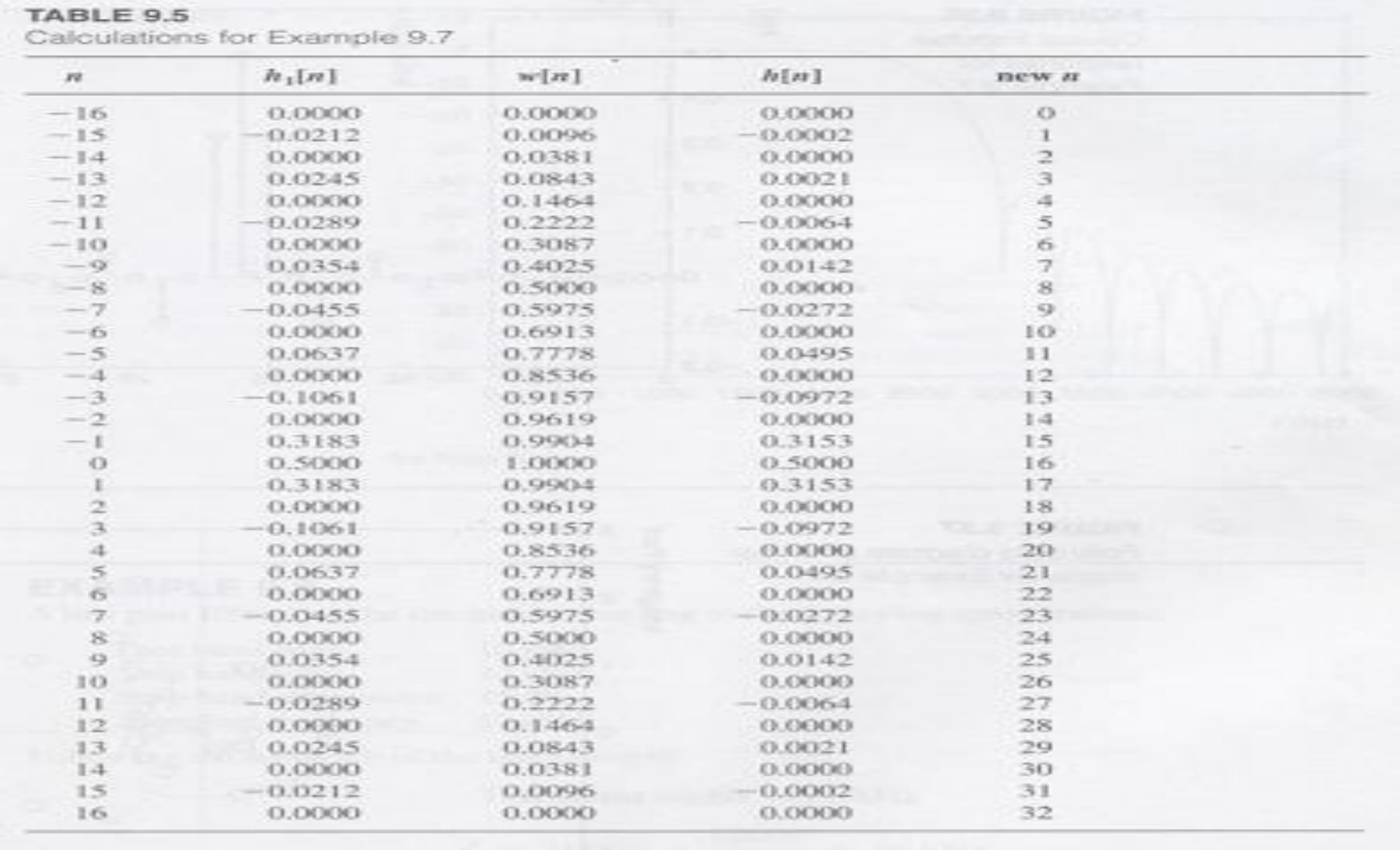


TABLE 9.5
Calculations for Example 9.7

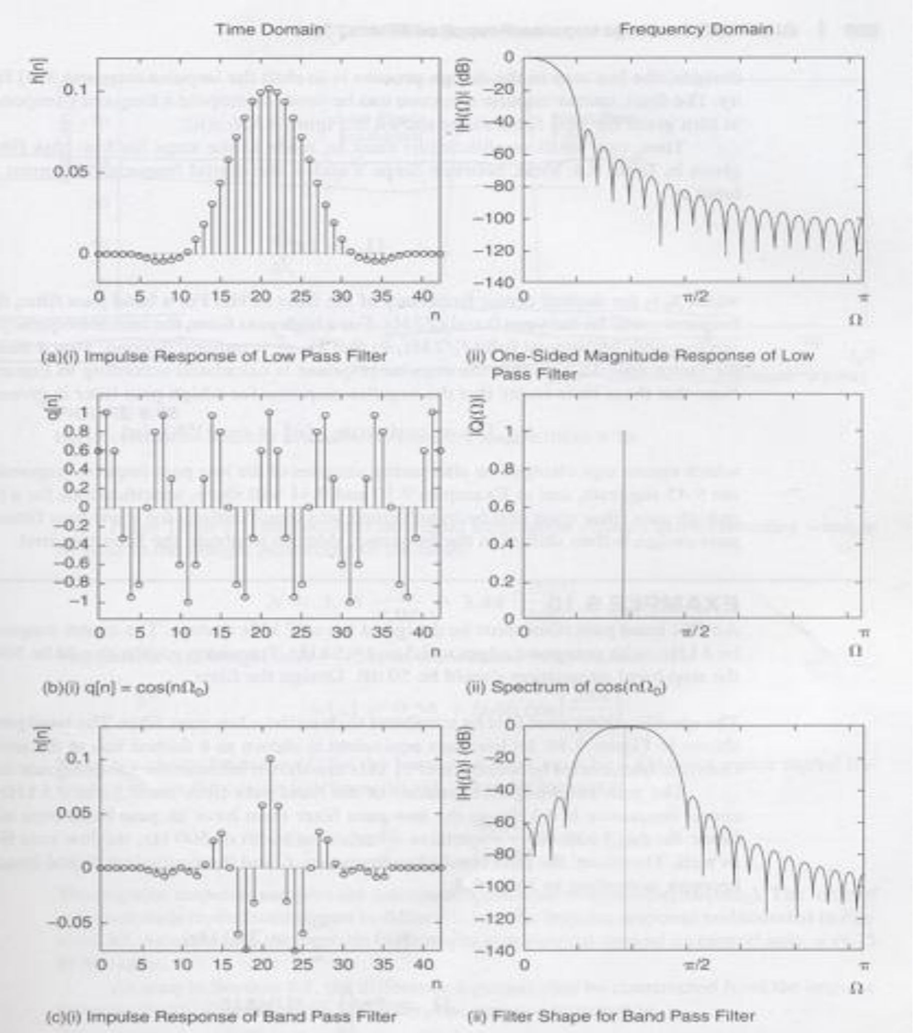
n	$h_1[n]$	$w[n]$	$h[n]$	new n
−16	0.0000	0.0000	0.0000	0
−15	−0.0212	0.0096	−0.0002	1
−14	0.0000	0.0381	0.0000	2
−13	0.0245	0.0843	0.0021	3
−12	0.0000	0.1464	0.0000	4
−11	−0.0289	0.2222	−0.0064	5
−10	0.0000	0.3087	0.0000	6
−9	0.0354	0.4025	0.0142	7
−8	0.0000	0.5000	0.0000	8
−7	−0.0455	0.5975	−0.0272	9
−6	0.0000	0.6913	0.0000	10
−5	0.0637	0.7778	0.0495	11
−4	0.0000	0.8536	0.0000	12
−3	−0.1061	0.9157	−0.0972	13
−2	0.0000	0.9619	0.0000	14
−1	0.3183	0.9904	0.3153	15
0	0.5000	1.0000	0.5000	16
1	0.3183	0.9904	0.3153	17
2	0.0000	0.9619	0.0000	18
3	−0.1061	0.9157	−0.0972	19
4	0.0000	0.8536	0.0000	20
5	0.0637	0.7778	0.0495	21
6	0.0000	0.6913	0.0000	22
7	−0.0455	0.5975	−0.0272	23
8	0.0000	0.5000	0.0000	24
9	0.0354	0.4025	0.0142	25
10	0.0000	0.3087	0.0000	26
11	−0.0289	0.2222	−0.0064	27
12	0.0000	0.1464	0.0000	28
13	0.0245	0.0843	0.0021	29
14	0.0000	0.0381	0.0000	30
15	−0.0212	0.0096	−0.0002	31
16	0.0000	0.0000	0.0000	32

BP and HP FIR Filters

- BP & HP filters can be designed starting with a low pass filter and shifting in freq: to produce the desired filter.
- The trick is to convolve the impulse with the LP mag: response to convert it into HP & BP.
- A single impulse function in the freq: domain must be located at the desired filter center freq:
- Then by convolving a LP filter shape with this impulse function has the effect of shifting a copy of the two sided LP filter shape to a new location and result is a BP or HP FIR filter.
- The figure on next slide illustrates the whole concept.

BP and HP FIR Filters

- The necessary impulse in this fig: 2b can be conveniently provided by a cosine function since its spectrum has a single spike.
- When convolution occurs in the freq: domain, multiplication must occur in the time domain.
- To convert noncausal LP filter $(h_1[w]w[n])$ into BP or HP filter, then the impulse response of LP filter must be multiplied by the cosine function given as $q[n]=cos(nw_0)$
- The multiplication is illustrated in the left column of this figure.
- The last step in the design process is to shift the impulse response h[n] for causality.
- Hence the figure 2c shows the new filter shape.



(a)(i) Impulse Response of Low Pass Filter

(ii) One-Sided Magnitude Response of Low Pass Filter

(b)(i) $q[n] = \cos(n\Omega_0)$

(ii) Spectrum of $\cos(n\Omega_0)$

(c)(i) Impulse Response of Band Pass Filter

(ii) Filter Shape for Band Pass Filter

BP and HP FIR Filters

- Two small modifications are made in the steps of LP FIR filter design.

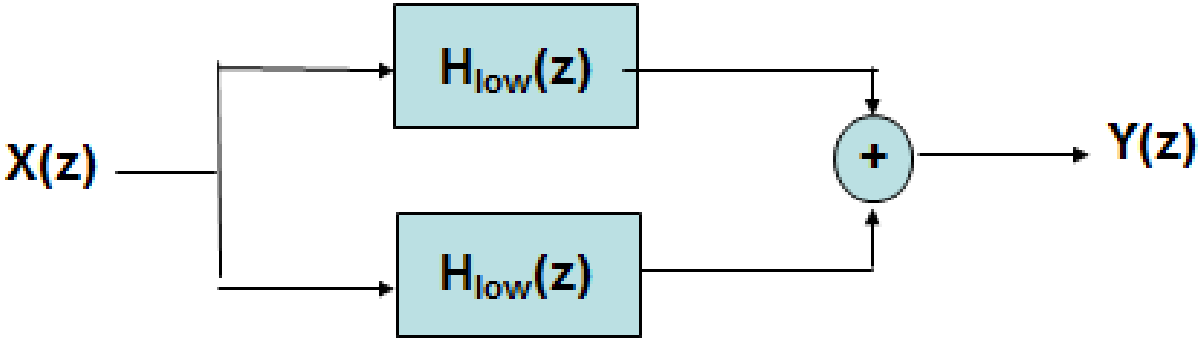
1. Between step 3 and 4, the digital freq: $w_0 = 2pif_0/f_s$
   - For BP, f_0 lie between 0 to $f_s/2$ Hz
   - For HP, $f_0 = f_s/2$ Hz
2. Step 4 must include the factor $\cos(nw_0)$

Band stop FIR Filters

- BS filter suppress one range of frequencies, the stop band, while pass all others.
- A simple method is available, provided the pass band edge frequencies are chosen correctly.
- A band stop filter can be constructed by combining a LP filter with a HP filter.
- The LP filter sets the pass band edge frequency for the lower end of the stop band and the HP filter sets the pass band edge freq: for the higher end.
- The impulse response for the two filters $h_{low}[n]$ & $h_{high}[n]$ can be found using previous method.

Band stop FIR Filters

- Both must be summed in the z domain to produce the impulse response for the band stop filter



This figure shows the addition of LP & HP resp: In z-domain

$$Y(z)=H_{low}(z)x(z) + H_{high}(z)x(z)$$

Overall transfer function is

$$H_{band\ stop}(z)=H_{low}(z)+H_{high}(z)$$

- The impulse response of the three filters are related in same way

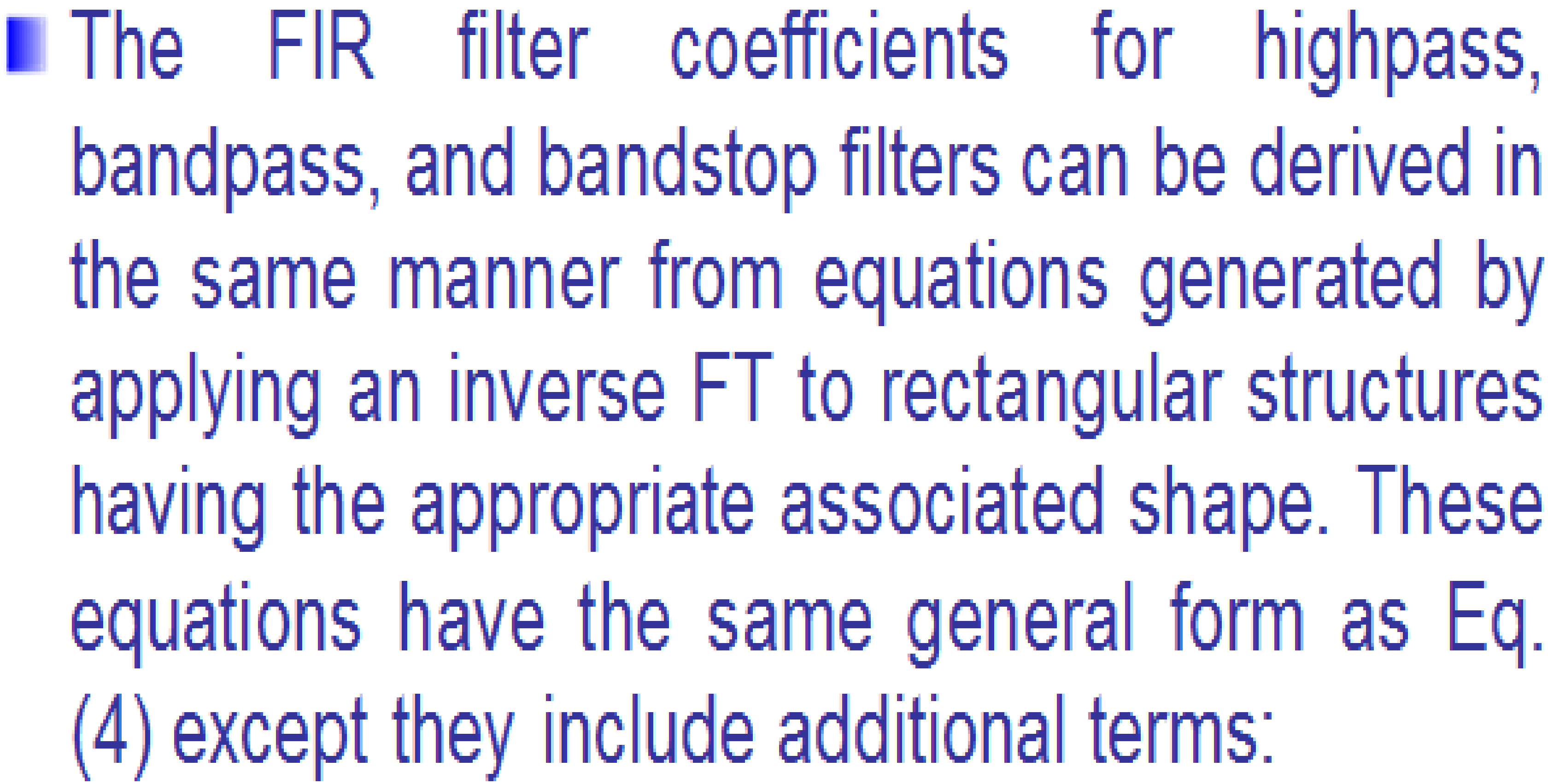
$$h_{bandstop}[n]=h_{low}[n]+h_{high}[n]$$

- Thus, the impulse response for a band stop filter is simply the sum of the impulse responses for HP & LP filters.

Band stop FIR Filters

- BP filters may designed in a similar fashion.
- By multiplying the impulse response of LP filter and HP filter we can get the band pass filter. $H_{bandpass}[n]=h_{low}[n]xh_{high}[n]$

Another method for designing HP,BP & BS FIR filters



- The FIR filter coefficients for highpass, bandpass, and bandstop filters can be derived in the same manner from equations generated by applying an inverse FT to rectangular structures having the appropriate associated shape. These equations have the same general form as Eq. (4) except they include additional terms:

Generalized FIR Filter design

High Pass Filter:

- The magnitude response of a normalized ideal high-pass filter is given by

$$H_d(\omega) = \begin{cases} e^{-j\omega K/2} & \omega_c \le |\omega| \le \pi \\ 0 & 0 \le |\omega| < \omega_c \end{cases} \qquad (18)$$

- By evaluating the inverse Fourier Transform, we obtain the ideal impulse response sequence

$$\boxed{b_d[n] = \frac{\sin\left[\pi(n-K/2)\right]}{\pi(n-K/2)} - \frac{\sin\left[\omega_c(n-K/2)\right]}{\pi(n-K/2)}} \qquad (19)$$

Band-Pass Filter:

- The magnitude of an ideal constant delay band-pass filter is given by

$$H_d(\omega) = \begin{cases} e^{-j\omega K/2} & \omega_L \le |\omega| \le \omega_H \\ 0 & elsewhere \end{cases} \qquad (20)$$

- where ω_L and ω_H are the low and high edges of the pass-band and K/2 is the delay. When IFT of (20) is performed, we find the Band-pass impulse response is given by

$$\boxed{b_d[n] = \frac{\sin[\omega_H(n-K/2)]}{\pi(n-K/2)} - \frac{\sin[\omega_L(n-K/2)]}{\pi(n-K/2)}} \qquad (21)$$

Example

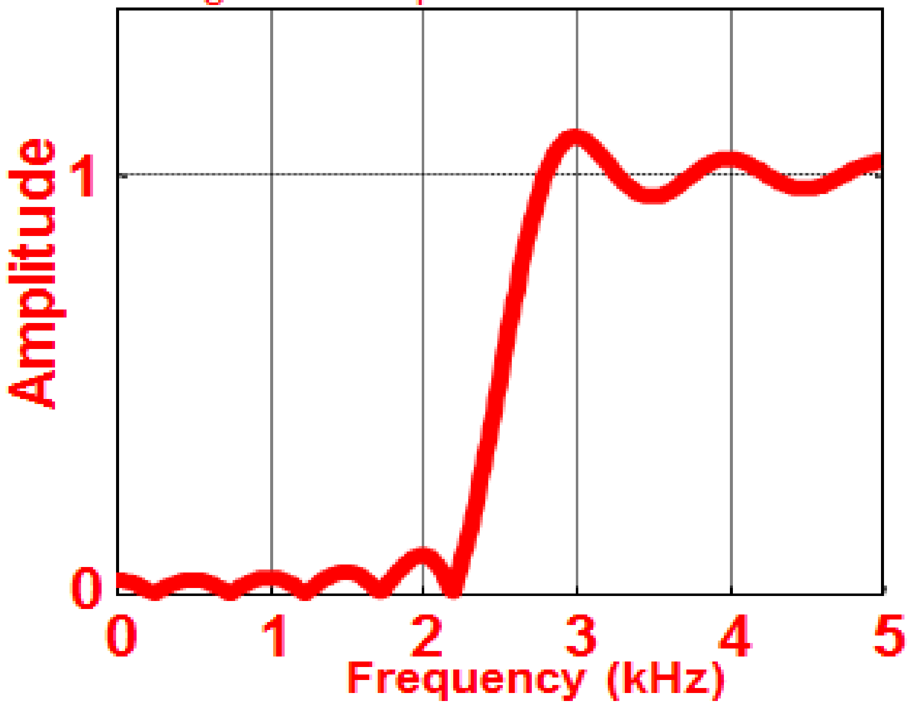
Example 5: Design a high-pass filter with a cutoff frequency of 2.5 KHz and a sample interval of 0.0001 s; The length of the filter is 21 and the rectangular window is used to truncate the ideal impulse response function.

Solution 5: The normalized cutoff frequency is $\omega_c = 2500 \times 0.0001 = 0.25$

The filter coefficients can be found from equation (19) with M = 20. These coefficients are given below:

b[0] =	-0.00000 =	b[20]
b[1] =	-0.03537 =	b[19]
b[2] =	-0.00000 =	b[18]
b[3] =	0.04547 =	b[17]
b[4] =	-0.00000 =	b[16]
b[5] =	-0.06366 =	b[15]
b[6] =	-0.00000 =	b[14]
b[7] =	0.10610 =	b[13]
b[8] =	-0.00000 =	b[12]
b[9] =	-0.31831 =	b[11]
b[10]=	0.500000	

The magnitude response is as shown below:



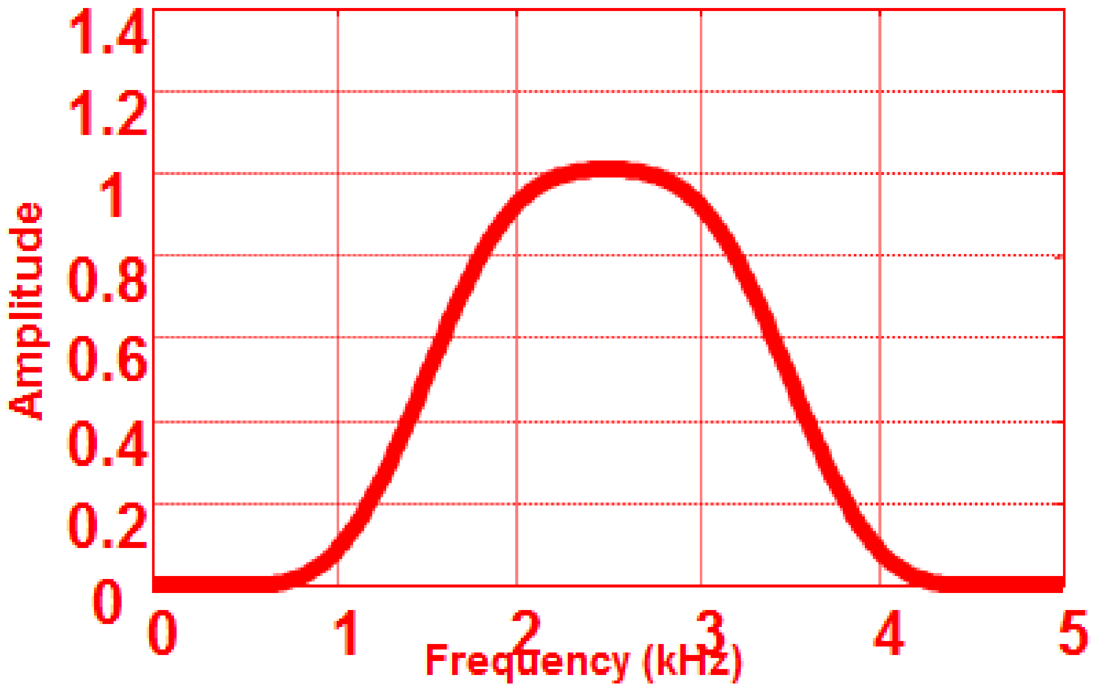
Example

Example 6: Design a Band-Pass Digital FIR filter of length 21. The band edges are specified to be at 1.5 and 3.5 kHz and the sampling interval is 0.0001s. Use the Hamming window.

Solution 6: The normalized cutoff frequencies are .15 and 0.35. The filter coefficients are obtained by multiplying (21) and (7) and are given below:

b[0] =	0.00000	= b[20]
b[1] =	-0.00000	= b[19]
b[2] =	-0.01270	= b[18]
b[3] =	0.00000	= b[17]
b[4] =	0.02481	= b[16]
b[5] =	0.00000	= b[15]
b[6] =	0.06381	= b[14]
b[7] =	-0.00000	= b[13]
b[8] =	-0.27614	= b[12]
b[9] =	0.000000	= b[11]
b[10] =	0.40000	

The magnitude response of the Designed filter



Generalized FIR Filter design

Band-Stop Filter:

- The ideal frequency response of a Band-Stop filter is given by

$$H_d(\omega)=\begin{cases} e^{-j\omega K/2} & |\omega| \le \omega_{L,} |\omega| \ge \omega_H \\ 0 & elsewhere \end{cases} \qquad (22)$$

- where w_L and w_H now specify the low and high edges of the stop-band. Taking the inverse Fourier transform of (22) yields

$$\boxed{b_d[n]=\frac{\sin[\omega_L(n-K/2)]}{\pi(n-K/2)}+\frac{\sin[\pi(n-K/2)]}{\pi(n-K/2)}-\frac{\sin[\omega_H(n-K/2)]}{\pi(n-K/2)}} \qquad (23)$$

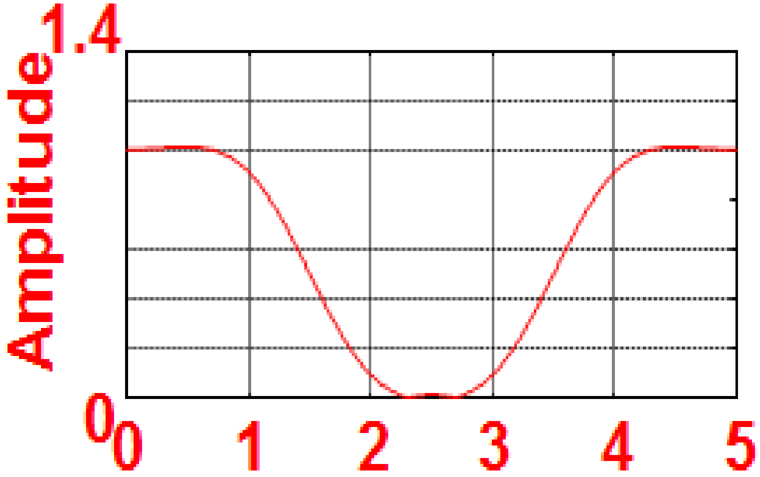
Example

Example 7: Design a band-stop filter with edges frequencies at 1.5 and 3.5 kHz relative to a sample interval of 0.0001. Use Hann window. Length of the filter is 21.

Solution 7: The normalized frequencies are 0.15 and .35. The filter coefficients can be obtained by multiplying (23) and (9) and are given by:

0	0.00000	7	0.00000	14	-0.06123
1	0.00000	8	0.27382	15	0.00000
2	0.00723	9	0.00000	16	-0.02155
3	0.00000	10	0.60000	17	0.00000
4	-0.02155	11	0.00000	18	0.00723
5	0.00000	12	0.27382	19	0.00000
6	-0.06123	13	0.00000	20	0.00000

The magnitude response is as under:



Optimal Method

- This method of calculating FIR filter coefficients is very powerful, very flexible, very easy to apply.
- Mostly used in many FIR applications.

Basic Concept:

- In window method the peak ripple of filters designed occurs near the band edges and decreases away from the band edges (figa)
- If the ripples were distributed more evenly (equiripple behavior) over the pass band & stop band (figb), a better approximation of the desired freq: response can be achieved.
- The optimal method is based on the concept of equiripple pass band & stop band.

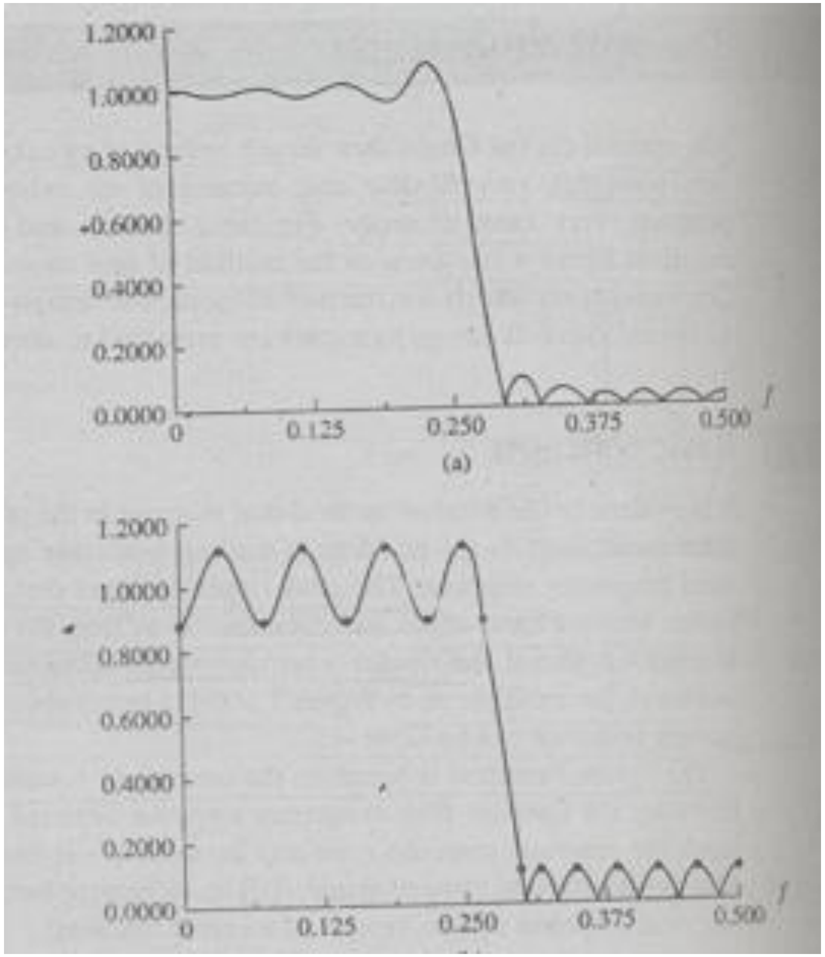


Figure: a , b

Optimal Method

- The difference b/w the ideal filter $H_d(w)$ & the practical response $H(w)$ can be viewed as an error function

$$E(w)=W(w)\ [H_D(w) - H(w)]$$

Where W(w) = weighting function of pass band & stop band

- In Optimal method, the object is to determine the filter coefficients, such that the value of the max: weighted error |E(w)|, is minimized in pass band & stop band.
- By minimizing the max: error |E(w)| the resulting filter response will have equiripple pass band & stop band.

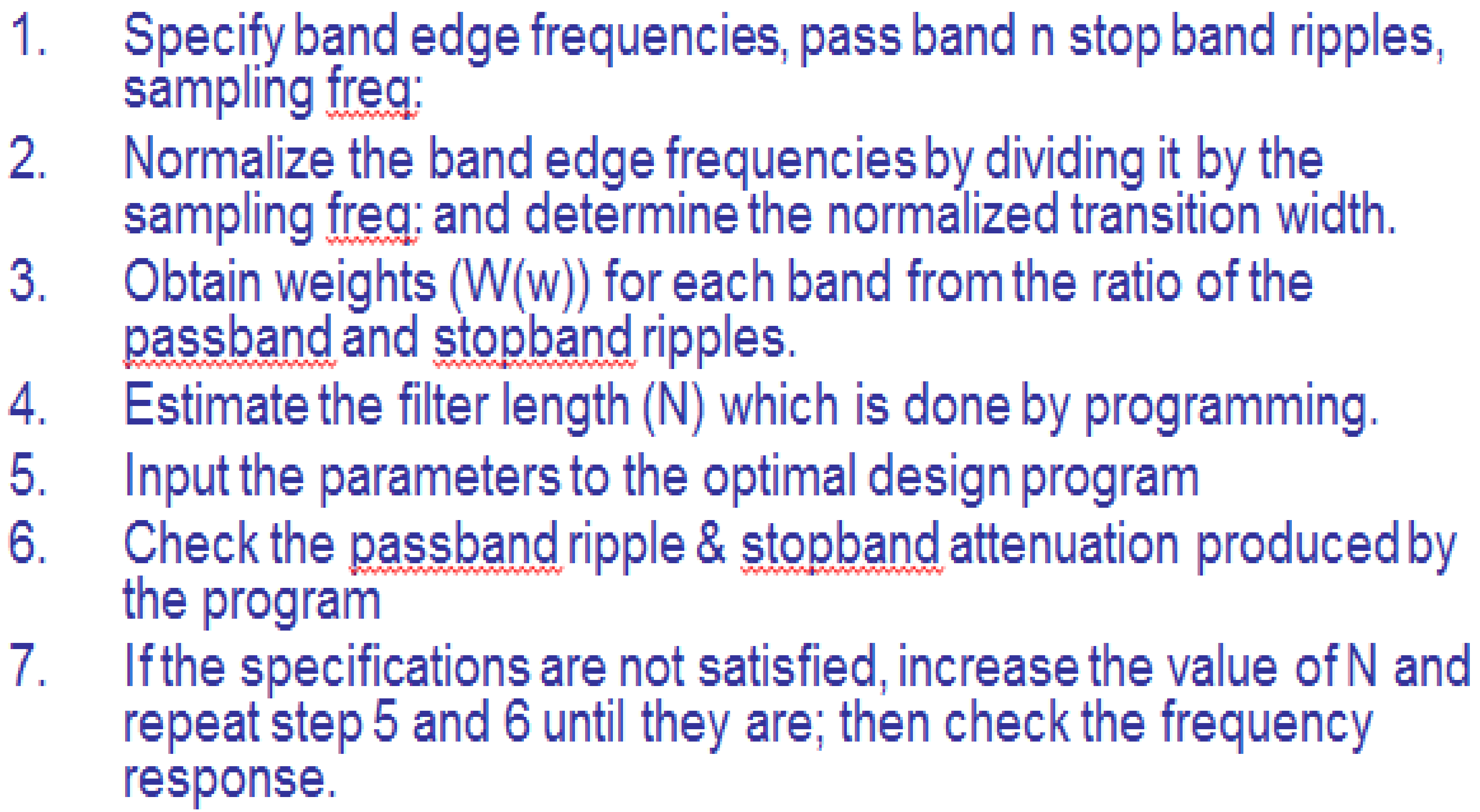
Optimal Method

- The minima and maxima points the extremal frequencies.
- Theses extremal frequencies are not known. For a given set of filter specifications, the locations of the extremal frequencies must be found.
- A powerful technique known as Remez exchange algorithm is used to find the extremal frequencies.
- The key steps are:
  1. Use Remez exchange algorithm to find optimum set of the extremal frequencies.
  2. Determine the frequency response using extremal frequencies.
  3. Obtain impulse response coefficients.

Optimal Method

- The most important is the first step where an iterative process is used to determine the extremal frequencies.
- A FORTRAN program implements this process. An equivalent C program is also available & widely used.
- Certain parameters are inputted in that computer program which calculates the extremal frequencies.
- The parameters required to use the optimal program are
    - N – number of filter coefficients i-e filter length
    - Jtype – Type of filter (Jtype = 1 = multiple pass band, stop band, LP, HP, BP, BS, Jtype = 2 = differentiate, Jtype = 3 = Hilbert transformers)
    - W(w) – weighting of pass band & stop band
    - Ngrid – number of frequeny points.
    - Edge – the band edge frequencies.
  - These symbols are user defined & not fixed.
- In practice, the (number of filter coefficients) N is unknown and its value may be estimated using certain relationships.
- The computer programs are available for computing the values of N so manually solving it is useless and time wasting.

Steps of procedure for calculating coefficients by Optimal method.



1. Specify band edge frequencies, pass band n stop band ripples, sampling freq:
2. Normalize the band edge frequencies by dividing it by the sampling freq: and determine the normalized transition width.
3. Obtain weights (W(w)) for each band from the ratio of the passband and stopband ripples.
4. Estimate the filter length (N) which is done by programming.
5. Input the parameters to the optimal design program
6. Check the passband ripple & stopband attenuation produced by the program
7. If the specifications are not satisfied, increase the value of N and repeat step 5 and 6 until they are; then check the frequency response.

Example

- **Example:** **The linear phase BP filter is required to meet the following specifications:**

  passband 900 – 1100Hz
  passband ripple <0.87dB
  stopband attenuation >30dB
  sampling frequency 15kHz
  transition frequency 450Hz

Use the optimal method to obtain suitable coefficients. Plot the filter spectrum

Solution: the filter has three bands: 0 to 450Hz, 900 to 1100Hz and 1550 to 7500Hz. First we will normalize them :

$$450 \to 450/15000 = 0.03$$

$$900 \to 900/15000 = 0.06$$

$$1100 \to 1100/15000 = 0.0733$$

$$1550 \to 1550/15000 = 0.1033$$

$$7500 \to 7500/15000 = 0.5$$

Example

- Next step is to find to choose weights for the bands. For that the deviations are converted into ordinary units using the following formulas:

$$0.87dB = 20\log(1+\delta_p) \Rightarrow \delta_p = anti\log(0.87/20)-1$$

$$\delta_p = 0.10535$$

$$30dB = -20\log(\delta_s) \Rightarrow \delta_s = 0.031623$$

- The ratio of δ_p & δ_s is 3.33 or 10/3

  $\delta_p / \delta_s = 10/3$

  The weighting is applied in opposite sense, i-e 10= stopband weight and 3 = passband weight.

- Using the program the value of N is calculated and taken as N = 41.
- The input parameters are applied to the optimal program and the coefficients are calculated which are given on the next slide along with the frequency spectrum of BP filter.

Example

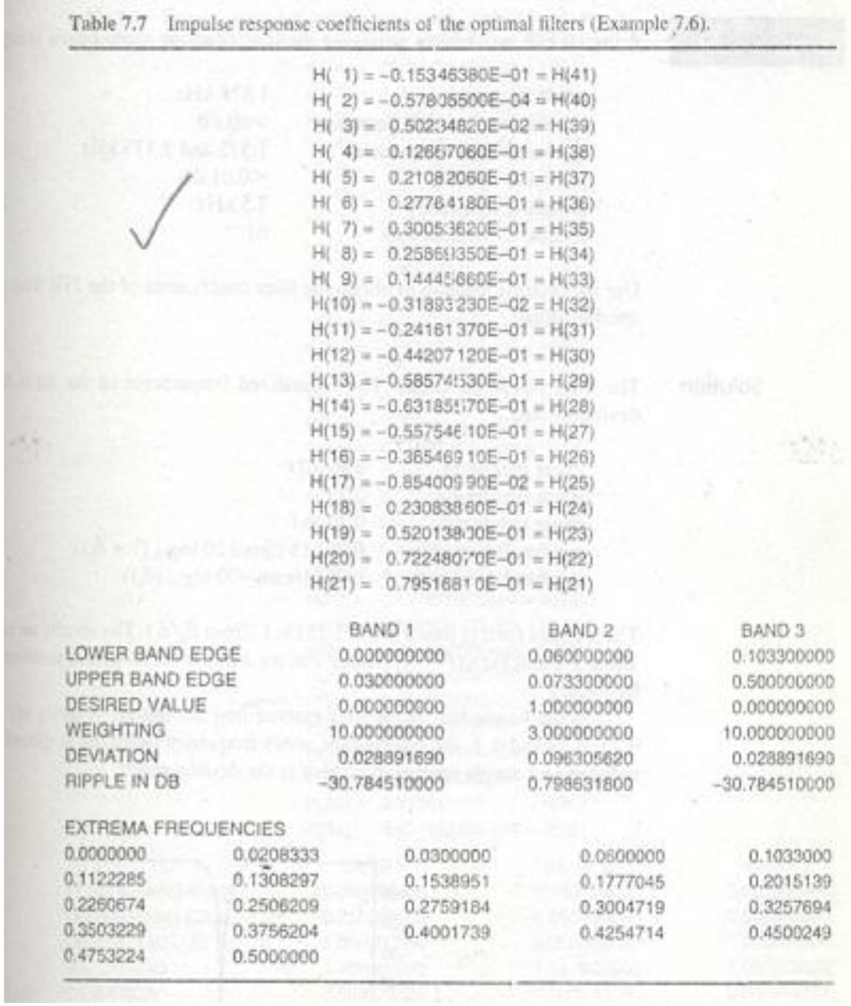


Table 7.7 Impulse response coefficients of the optimal filters (Example 7.6).

```
                    H( 1) = -0.15346380E-01 = H(41)
                    H( 2) = -0.57805500E-04 = H(40)
                    H( 3) =  0.50234820E-02 = H(39)
                    H( 4) =  0.12667060E-01 = H(38)
                    H( 5) =  0.21082060E-01 = H(37)
                    H( 6) =  0.27784180E-01 = H(36)
                    H( 7) =  0.30053620E-01 = H(35)
                    H( 8) =  0.25869350E-01 = H(34)
                    H( 9) =  0.14445660E-01 = H(33)
                    H(10) = -0.31893230E-02 = H(32)
                    H(11) = -0.24161370E-01 = H(31)
                    H(12) = -0.44207120E-01 = H(30)
                    H(13) = -0.58574530E-01 = H(29)
                    H(14) = -0.63185570E-01 = H(28)
                    H(15) = -0.55754610E-01 = H(27)
                    H(16) = -0.36546910E-01 = H(26)
                    H(17) = -0.85400990E-02 = H(25)
                    H(18) =  0.23083860E-01 = H(24)
                    H(19) =  0.52013800E-01 = H(23)
                    H(20) =  0.72248070E-01 = H(22)
                    H(21) =  0.79516810E-01 = H(21)

                        BAND 1          BAND 2          BAND 3
LOWER BAND EDGE      0.000000000     0.060000000     0.103300000
UPPER BAND EDGE      0.030000000     0.073300000     0.500000000
DESIRED VALUE        0.000000000     1.000000000     0.000000000
WEIGHTING           10.000000000     3.000000000    10.000000000
DEVIATION            0.028891690     0.096305620     0.028891690
RIPPLE IN DB       -30.784510000     0.798631800   -30.784510000

EXTREMA FREQUENCIES
0.0000000      0.0208333      0.0300000      0.0600000      0.1033000
0.1122285      0.1308297      0.1538951      0.1777045      0.2015139
0.2260674      0.2506209      0.2759184      0.3004719      0.3257694
0.3503229      0.3756204      0.4001739      0.4254714      0.4500249
0.4753224      0.5000000
```

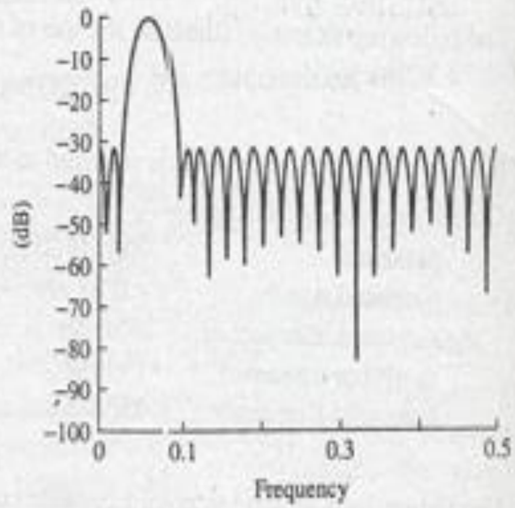


Figure 7.13 Frequency response of filter (normalized frequency scale).

Frequency sampling method

- Another method for calculating FIR coefficients is frequency sampling method.
- This method can design standard frequency selective filter (LP, HP, BP, BS) as well as arbitrary frequency responses.
- In this method, samples are used to design the FIR filters.
- The big attraction in this method is that recursive as well as nonrecursive FIR filters can be designed.

STEP 3: Realization Structures for FIR Filters

- Realization structures are essentially block diagram representation of different theoretically equivalent ways the transfer function can be arranged. $y(n)=\sum_{m=0}^{N-1} h(m)x(n-m)$
- They consists of multipliers, adders & delay elements.

1. **Transversal (tapped delay) structure**
   - The i/p & o/p of the filter for this structure are related simply by

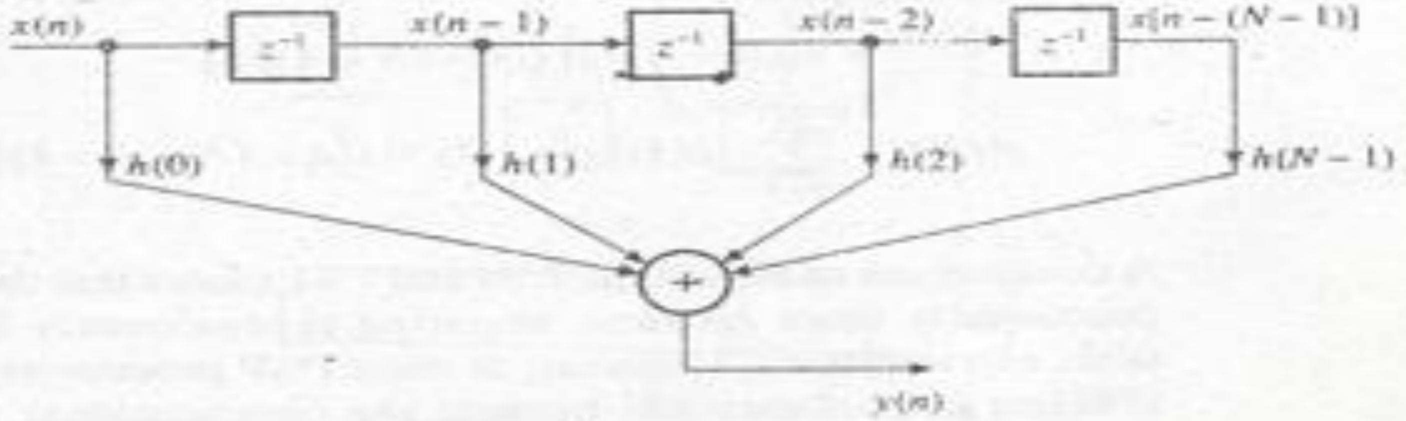
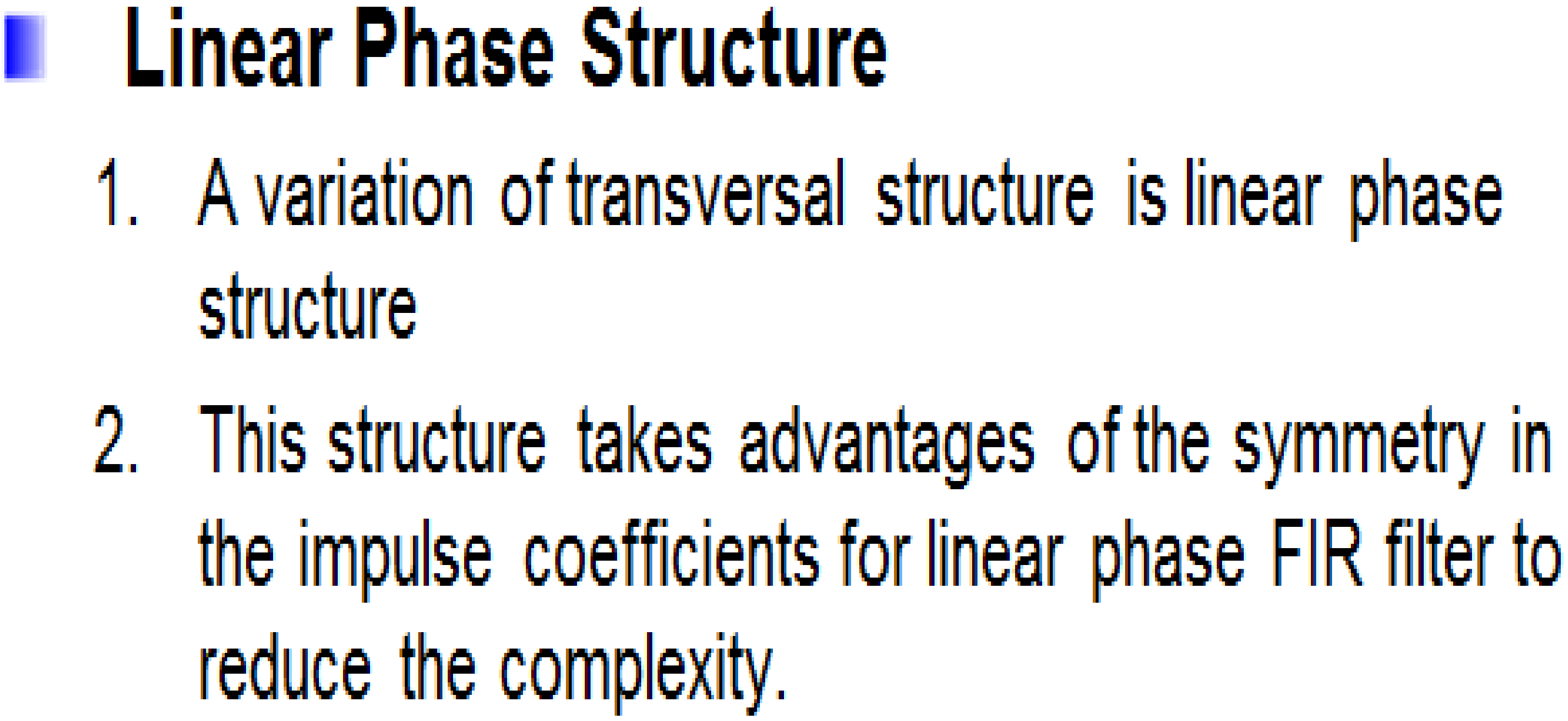


Figure 7.28 Transversal filter structure

Linear Phase Structure

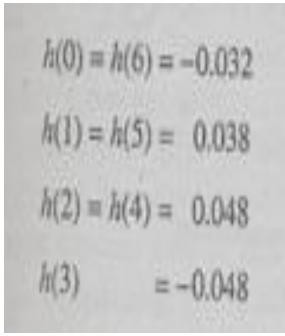


Linear Phase Structure

1. A variation of transversal structure is linear phase structure
2. This structure takes advantages of the symmetry in the impulse coefficients for linear phase FIR filter to reduce the complexity.

Linear Phase Structure: Example

- Example: A linear Phase FIR filter has seven coeff: which are listed below. Draw the realization diagrams for the filter using (a)transversal (b) linear phase structure.

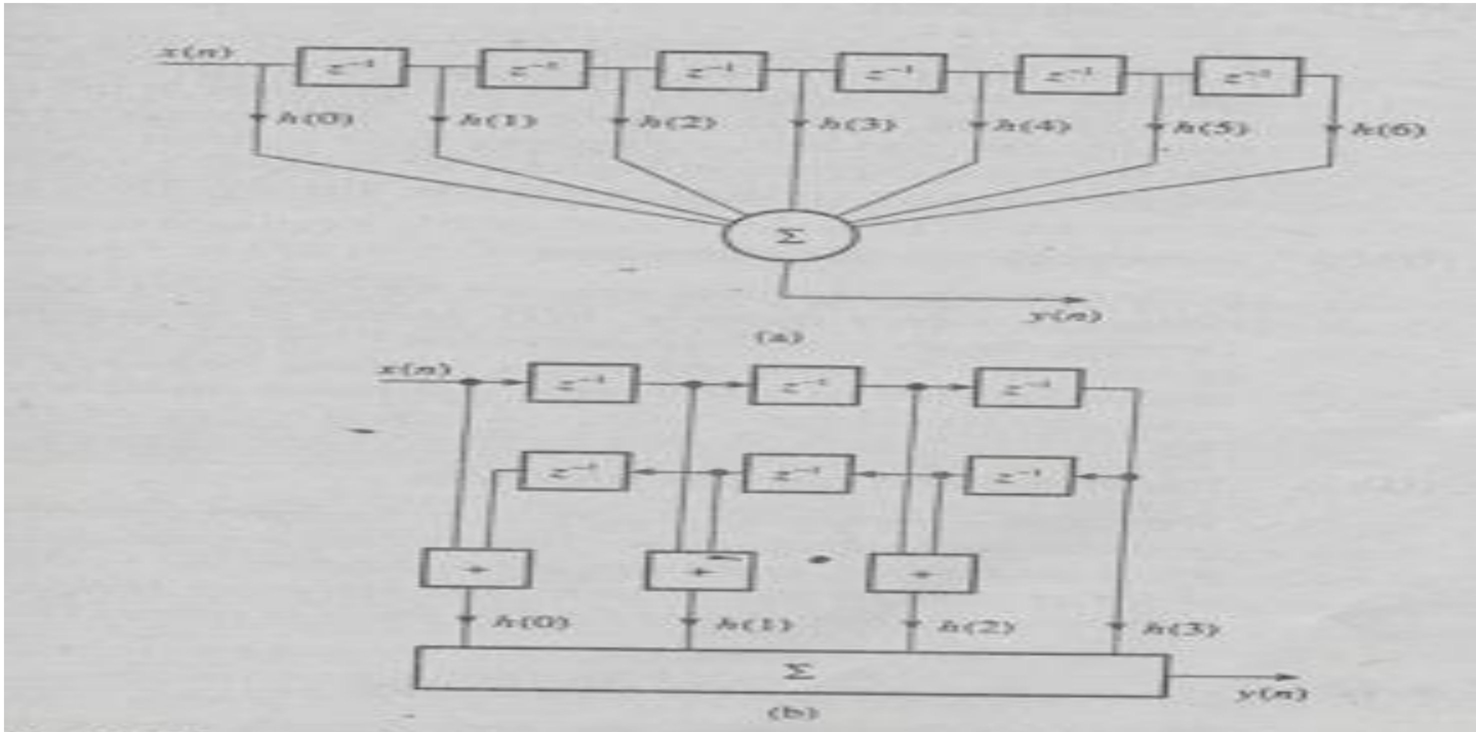


$h(0) = h(6) = -0.032$

$h(1) = h(5) = 0.038$

$h(2) = h(4) = 0.048$

$h(3) = -0.048$



29 (a) Transversal and (b) linear phase structure for Example 7.17.

STEP 4: Finite Wordlength effects in FIR filters

- In practice, FIR digital filters are often implemented using DSP processors. In this case, number of bits used to represent the i/p data to the filter coefficients and in performing arithmetic operations must be small for efficiency and to limit the cost of digital filter.
- Problem caused by using finite number of bits are referred to a finite wordlength effects.
- There are 4 ways in which finite wordlength effects the performance of FIR filters.

1. ADC noise – ADC quantization noise.
   - Solution: the effect can be reduced by using additional bits.
2. Coefficient quantization errors – these result from representing filter coefficients with a limited number of bits.
   - Solution: The effect can be reduced by using enough bits to represent the filter coefficients.
3. Arithmetic flow: this occurs when partial sums or filter o/p exceeds the permissible wordlength of the system.
   - Solution: scale the filter coefficients by dividing each coefficients by a factor such that the filter such that the filter o/p sample never exceed the permissible wordlength.

STEP 5: FIR Implementation Techniques

- To implement a filter, we need the following basic components:
  1. Memory (RAM) to store the present and past input samples, x(n) & x(n-k)
  2. Memory (RAM or ROM) for storing the filter coefficients, the h(k)
  3. A multiplier (software or hardware)
  4. Adders or ALU
- These components together with means of controlling them constitute the digital filter.
- Filter implementation is divided into 2 parts: software & hardware. These days most devices used in filtering are programmable.
- In real time operation the hardware implementation is the best option. Hardware implementation have three approaches: standard microprocessor (such as Motorola 68000), DSP processor (such as Texas instruments TMS320),building block and algorithmic specific.
- In building block approach, dedicated pieces of hardware are used.
- DSP processors have architectures and instruction sets optimized for FIR filtering operation. They are more flexible then algorithmic specific processors but they are slower.

End of Chapter