Digital Filtering

Jason R. Blough, Ph.D.
Mechanical Engineering – Engineering Mechanics Dept.
Michigan Technological University
Houghton, MI USA

Digital Filter Applications

- Sound quality editing for playback
- Sound quality equalization
- Durability testing – often used to condition shaker signals
  - Random and Sine testing
- Order tracking – Kalman, Vold-Kalman, …etc.
- Data acquisition systems – anti-alias filtering
- Decimation
- Zoom transform
- Interpolation – fixed and adaptive
- Active control – noise or vibration
- Communications – noise reduction
- Differentiation
- Hilbert Transform
- …many others…

Digital Filters

- An algorithm that transforms a digitized input data stream into an output data stream is a digital filter.

- Digital filters are used for many tasks:
  - Low pass filtering
  - High pass filtering
  - Band-pass filtering
  - Band-stop filtering
  - Data smoothing
  - Running averages

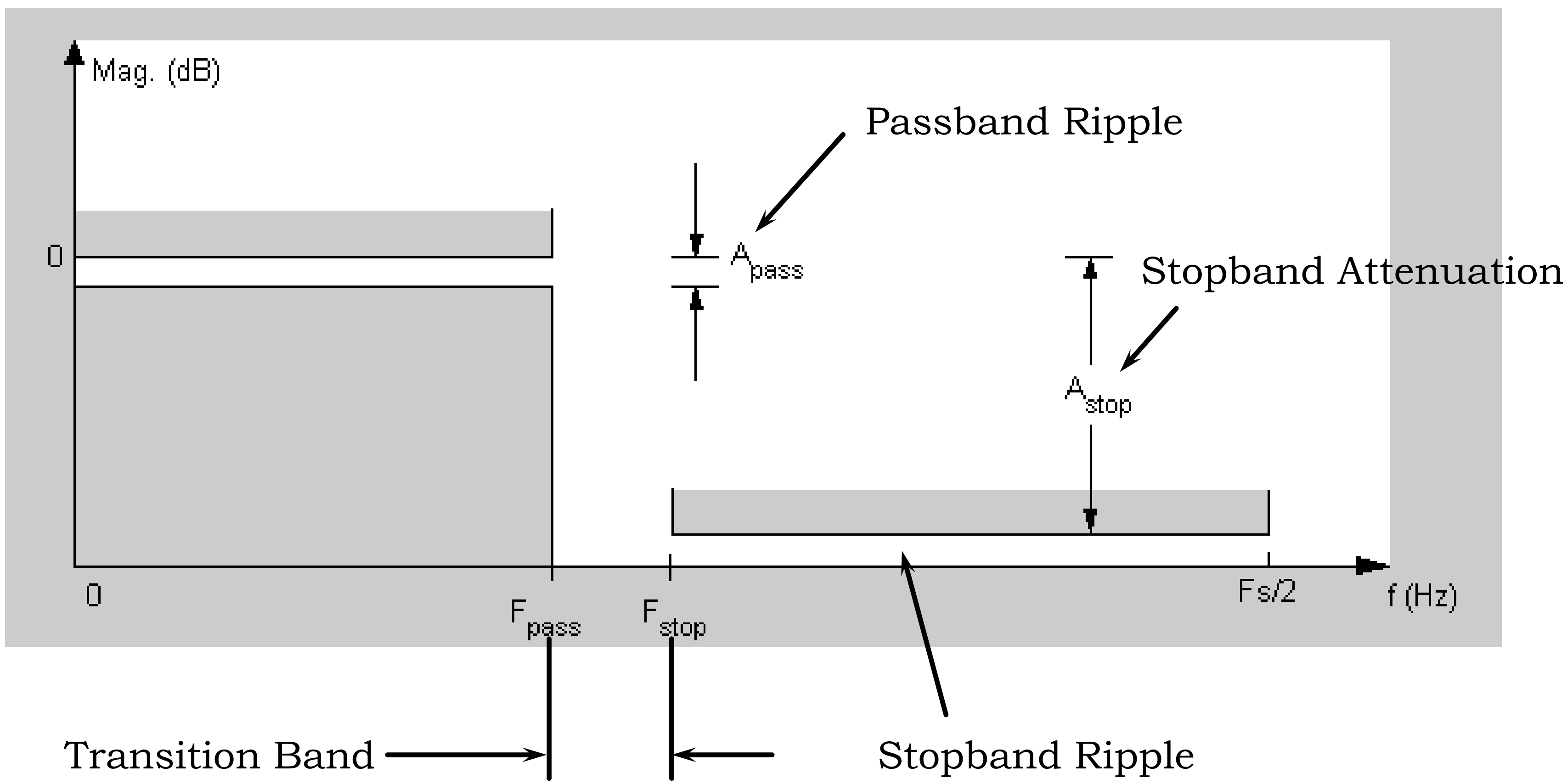
Advantages of Digital Filters

- Accuracy – characteristics are fully deterministic
- Repeatability – no time variance, drift, aging
- Implementation – implemented in hardware or software
- Compatibility with other signal processing applications
- Programmability – easy to program
- Flexibility – can design to shape signal in almost any way
- Cost effective – typically much cheaper than analog filters
- No channel to channel variation – use same algorithm on all channels so they are exactly amplitude/phase matched

Types of Digital Filters

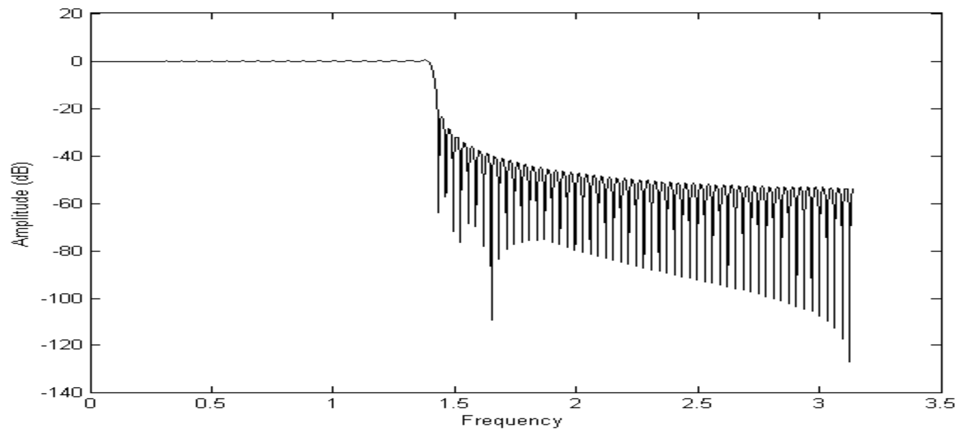
- FIR – Finite Impulse Response
- IIR – Infinite Impulse Response
- Adaptive IIR
- Adaptive FIR
- Kalman Filtering, Vold-Kalman Filtering
- Long FFT digital filtering

Filter Specifications

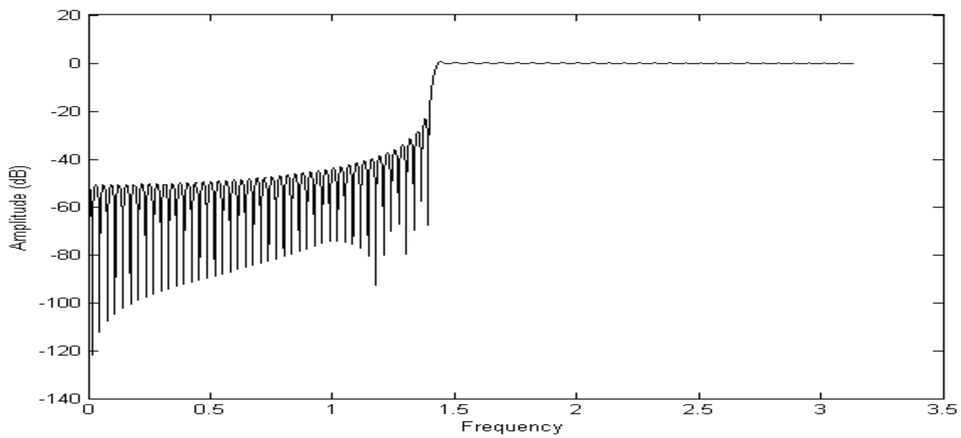


Digital Filter Shapes

- ➢ Low Pass Filter

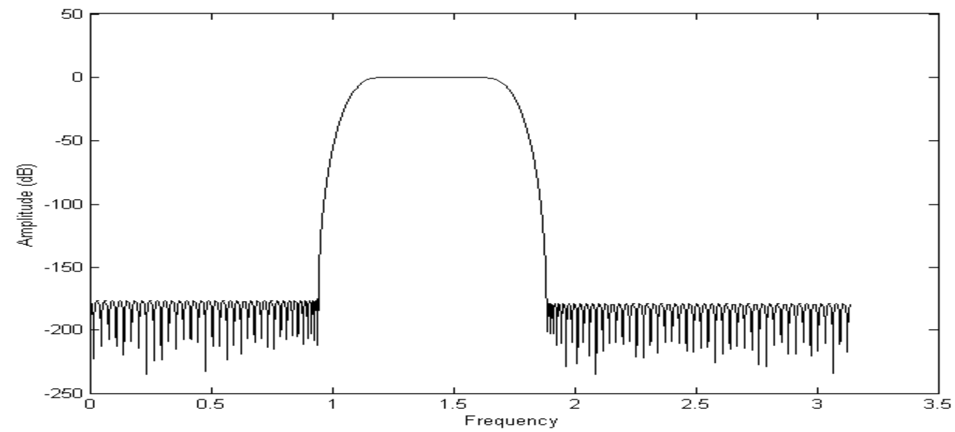


- ➢ High Pass Filter

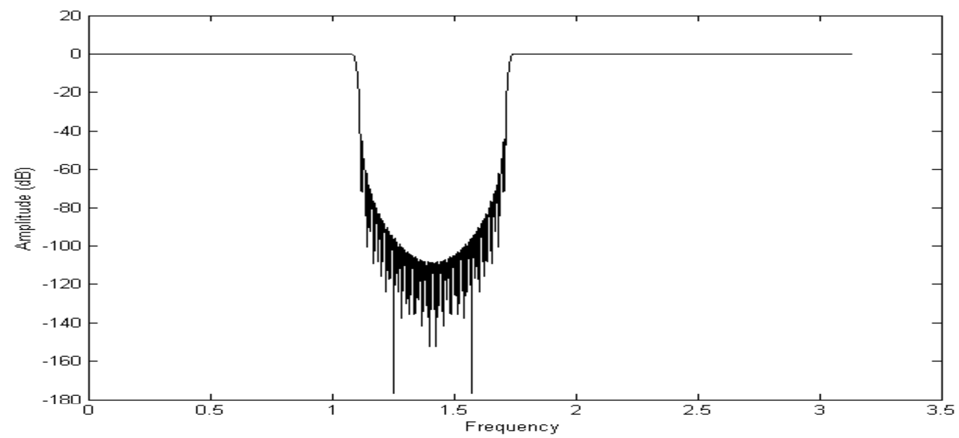


Digital Filter Shapes

- Band Pass Filter



- Band Stop Filter



FIR Filter

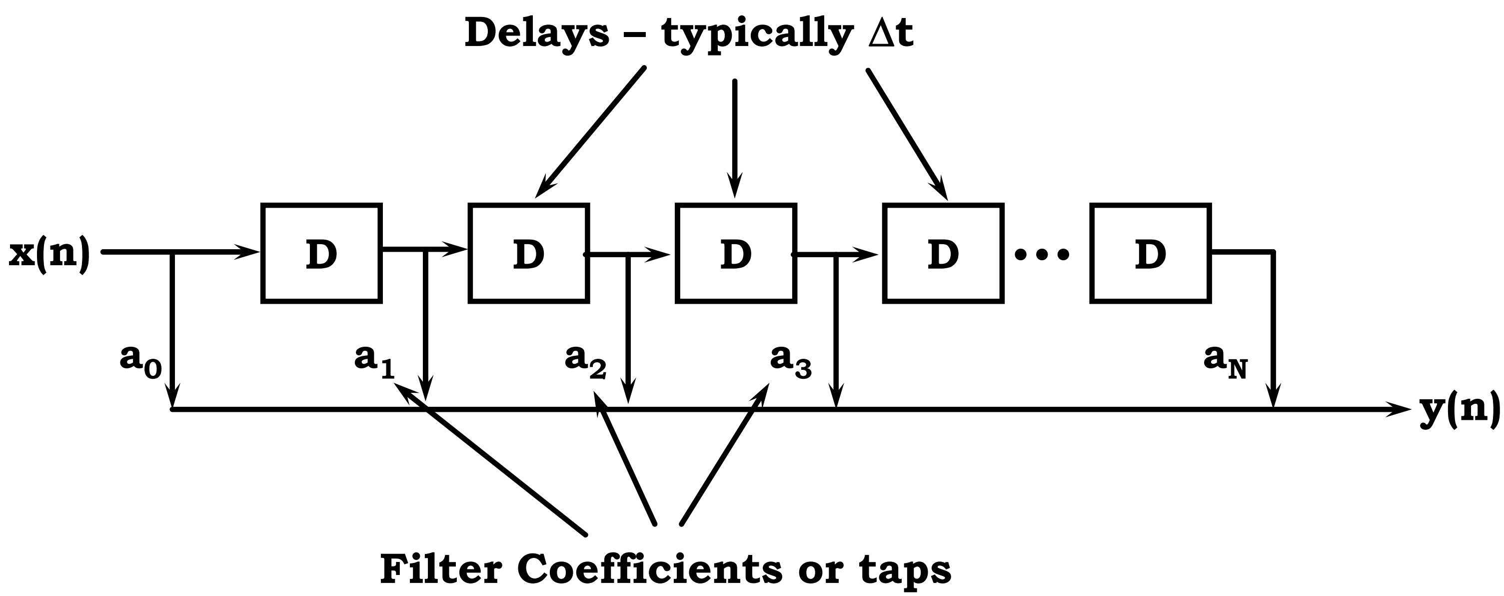
- FIR – Finite Impulse Response
- Time domain filter
- Only relies on the data to calculate a result
- Advantages:
  - Always stable
  - Linear phase
  - Easy to design
  - Easy to implement
  - Easy to implement in adaptive sense
  - Finite start-up transient
- Disadvantage
  - May be more computationally demanding as they are higher order than IIR for most applications.

MichiganTech

FIR Filter

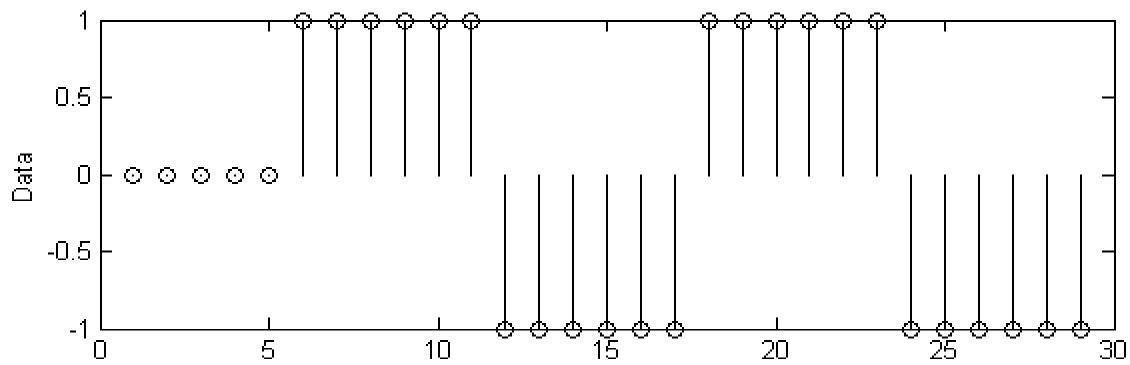
- Implementation:

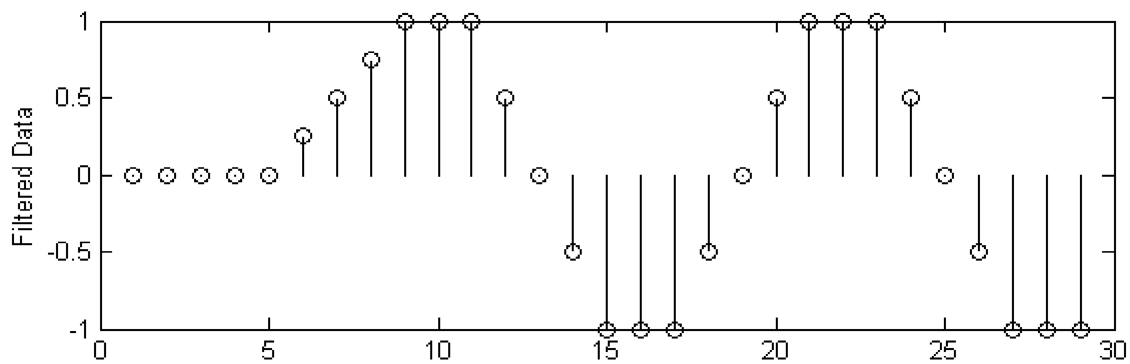
$$y(n) = \sum_{k=0}^{N} a_k \times x(n-k)$$



FIR Example – Data smoothing

$$y(n) = \frac{1}{4}\left[x(n) + x(n-1) + x(n-2) + x(n-3)\right] = \frac{1}{4}\sum_{k=0}^{3} x(n-k)$$

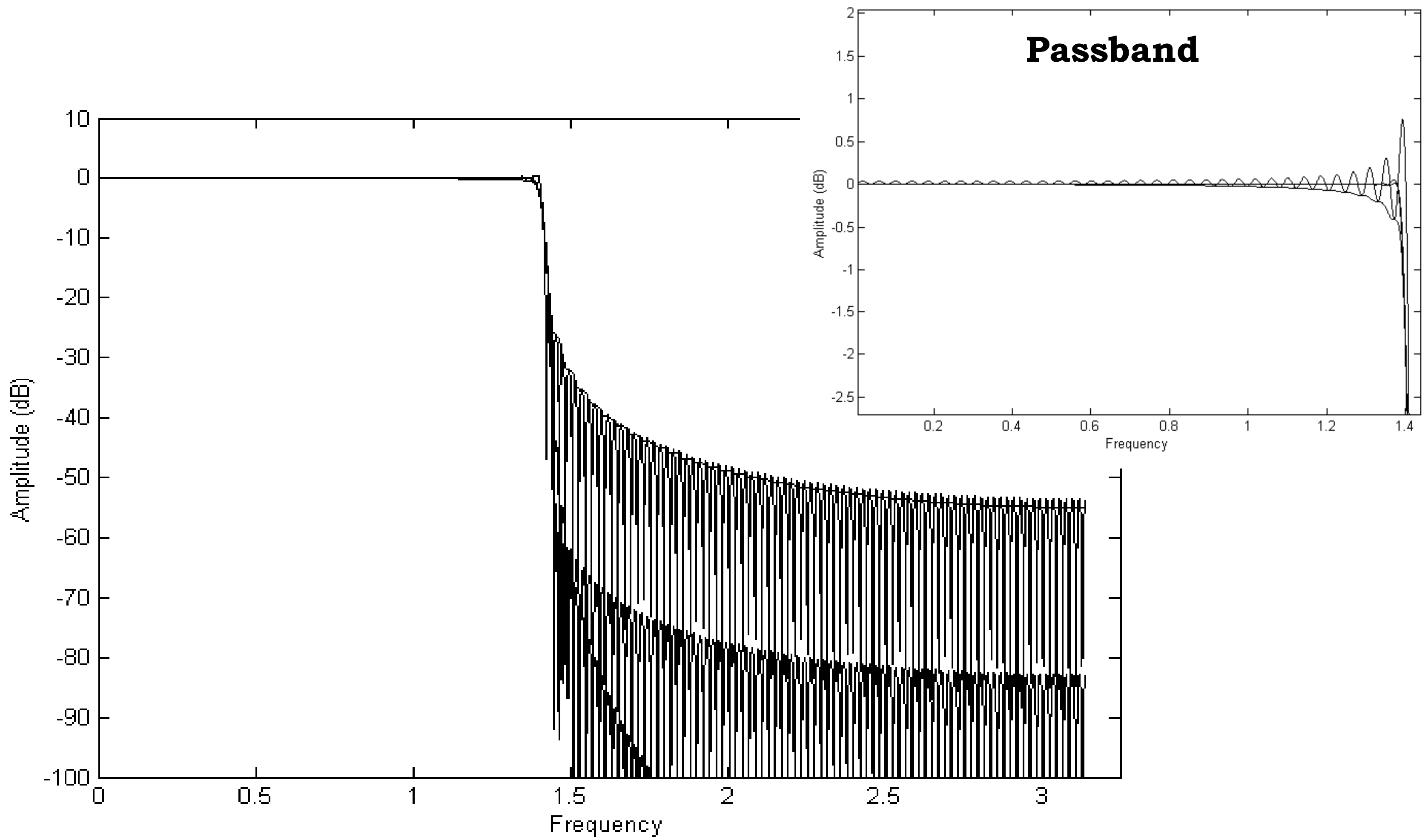




Filter Design Types - FIR

- FIR Filter Design Methods
  - Window – Pick a window and draw in the frequency domain.
    - Uses standard DSP windows which control part of the shape.
    - Pick the order yourself to get desired shape.
  - Least Squares – Algorithm that minimizes order based on shape criteria given.
    - All engineer provides is basic shape criteria.
  - Equiripple – Algorithm that designs filter to have equal ripple in stop-band

Different Filters - FIR



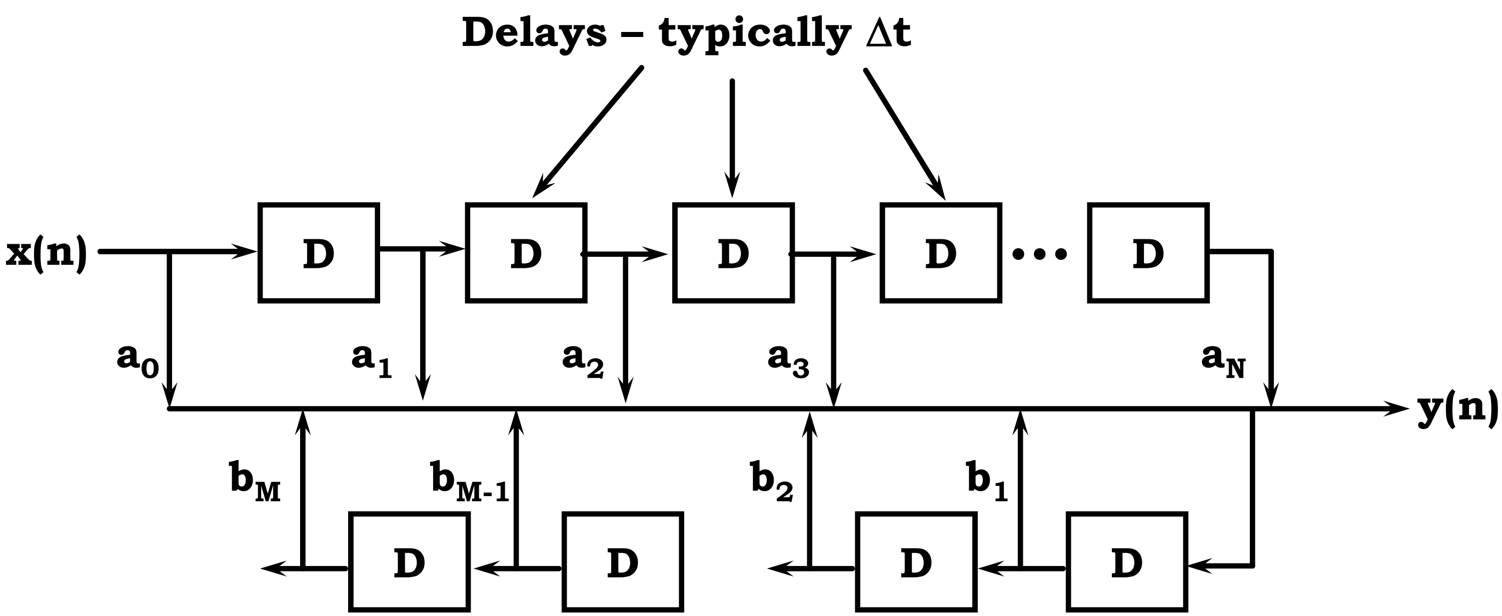
IIR Filter

- IIR – Infinite Impulse Response
- Time domain filter
- Relies on the data and past filter outputs to calculate a result
- Advantages:
  - Computationally efficient
- Disadvantage
  - Non-linear phase
  - Sensitive to round off error
  - More difficult to design some shapes
  - Infinite start-up transient

IIR Filter

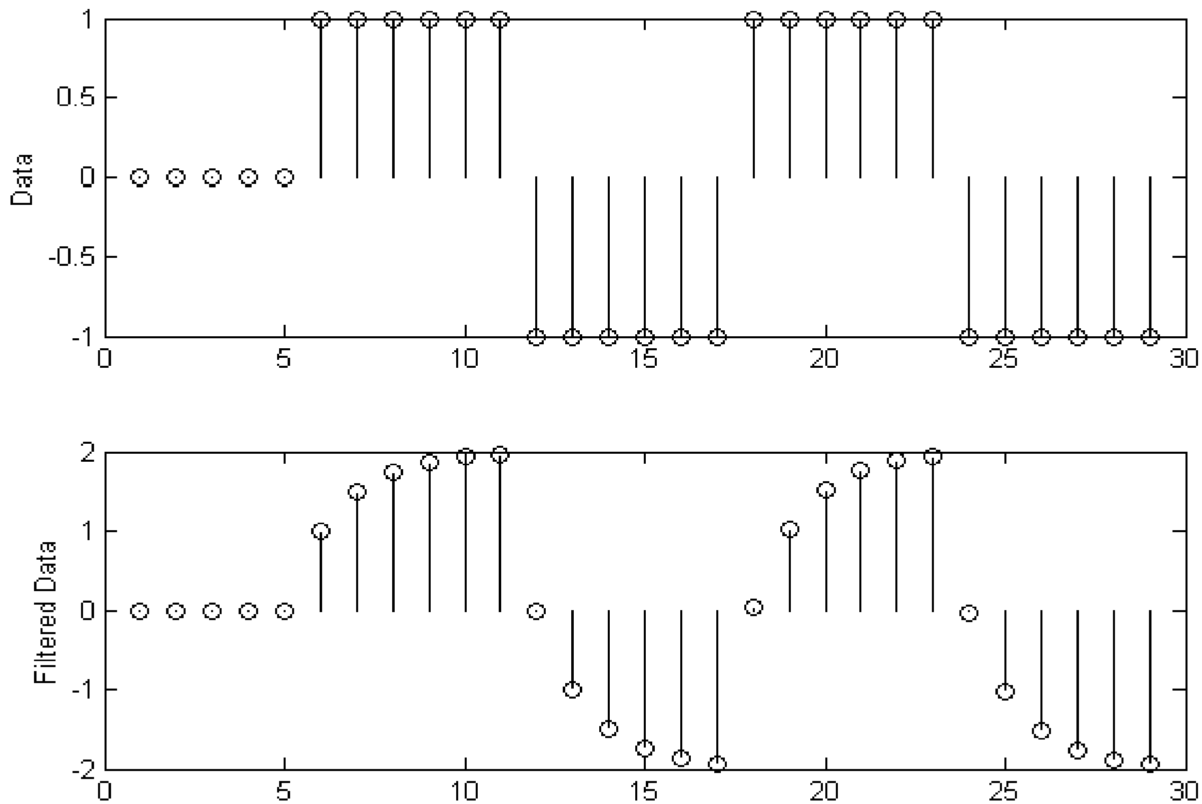
- Implementation:

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



IIR Filter – Exponential Averaging Filter

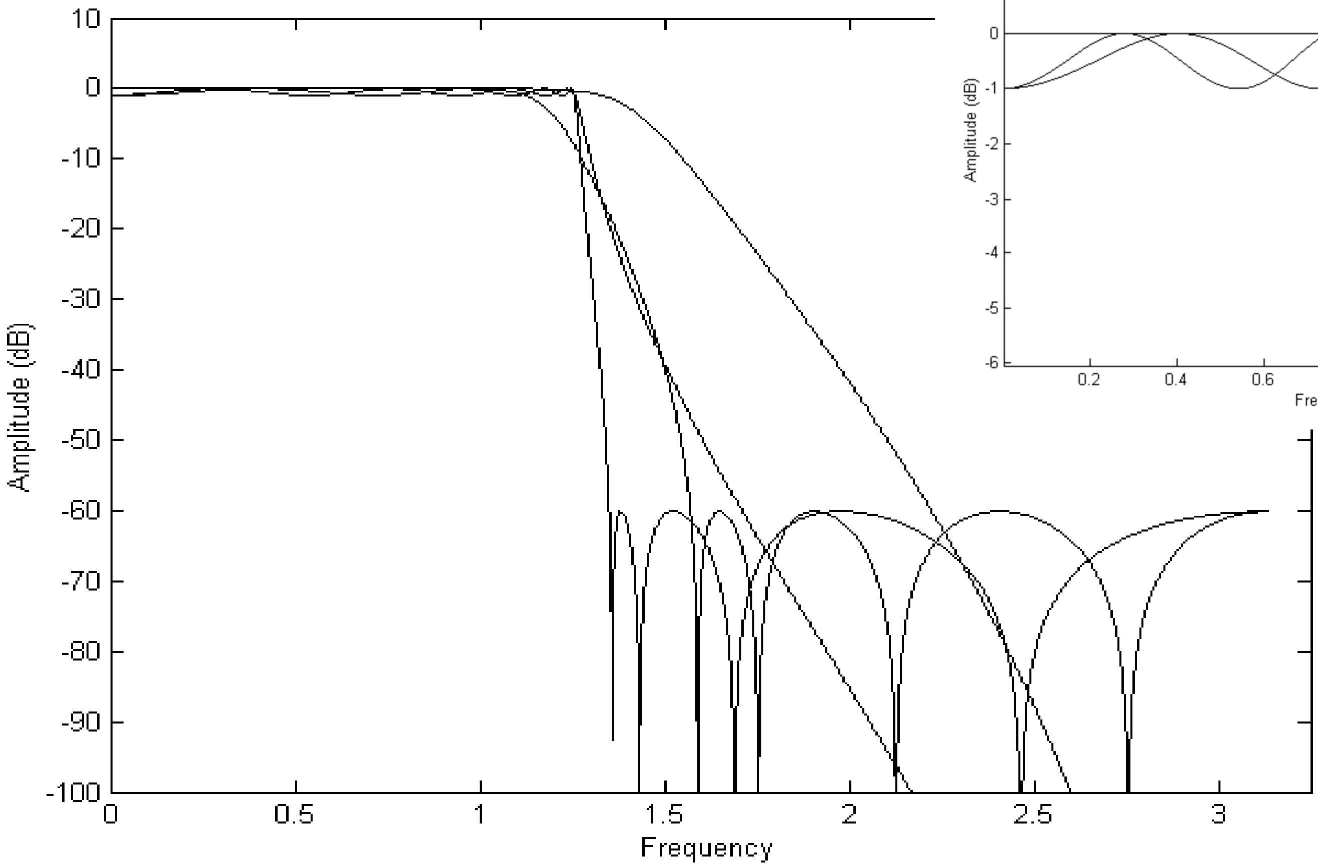
$$y(n) = x(n) + 0.5 \times y(n-1)$$

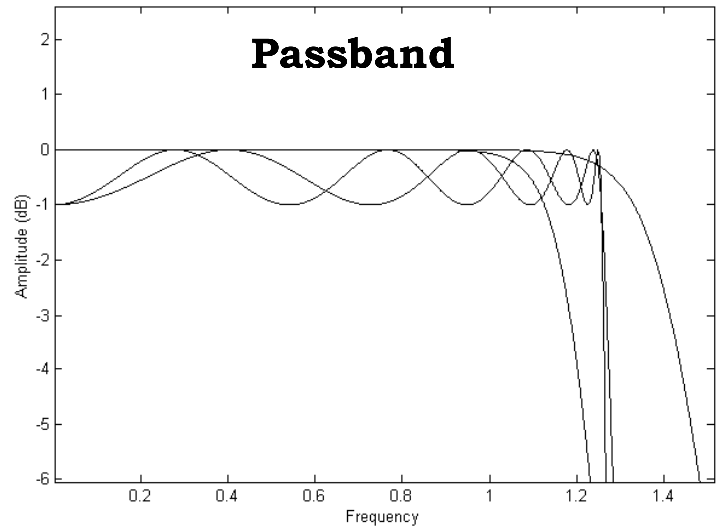


Filter Design Types - IIR

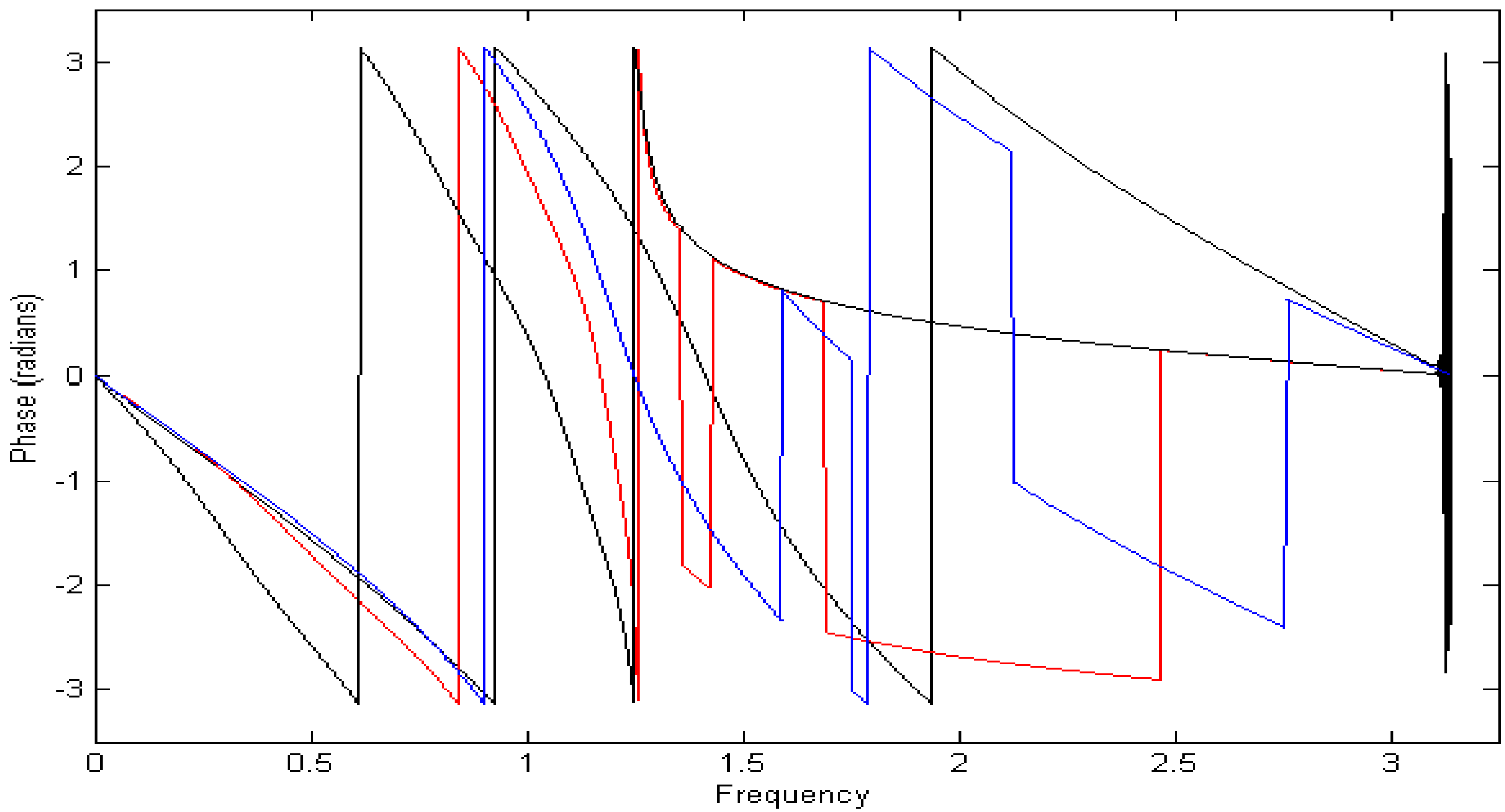
- IIR Filter Design Methods
  - Different designs based on analog filter designs.
    - Bilinear Transformation - Algorithms used to determine coefficients to approximate same shape as analog filters.
  - Butterworth – very flat passband, gradual rolloff
  - Chebyschev Type I – sharper rolloff, very smooth stopband, highly non-linear phase
  - Chebyschev Type II – sharper rolloff, equiripple in stopband, slightly non-linear phase
  - Elliptic – sharper rolloff, shortest transition width, equiripple in both stopband and passband, highly non-linear phase

Different Filters - IIR



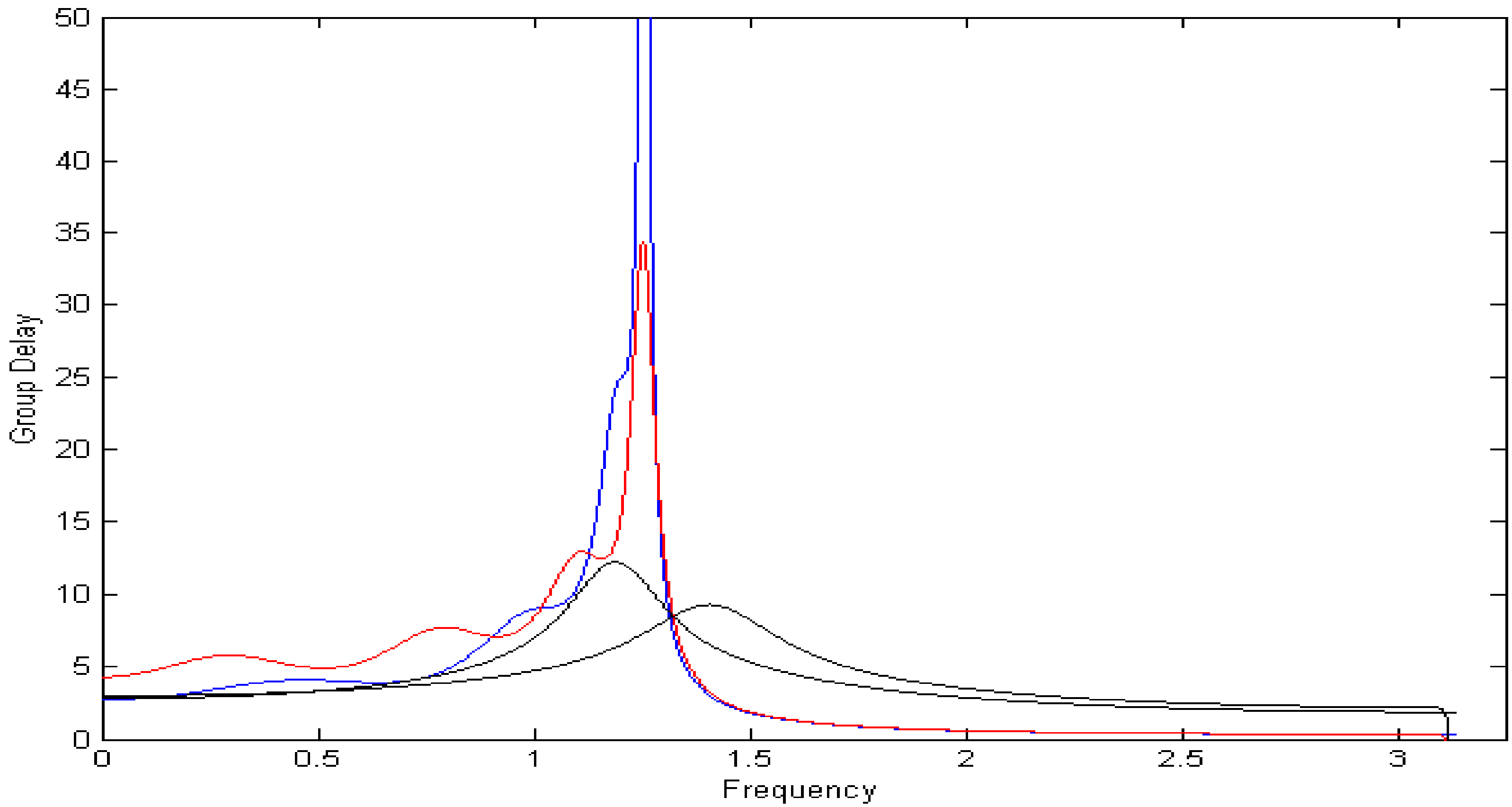


Different Filters – IIR Phase



Different Filters – IIR Group Delay

- Group Delay describes how well behaved the phase of a filter is.



Z-Transform

- Discrete-time counterpart of the Laplace transforms for continuous signals/systems
- Useful for solving constant coefficient difference equations, evaluation of linear system time-invariant system response, design of linear filters.

Recall the descrete-time Fourier transform:

$$X\left(e^{j\omega}\right) = \sum_{n=-\infty}^{\infty} x(n)e^{-jn\omega}$$

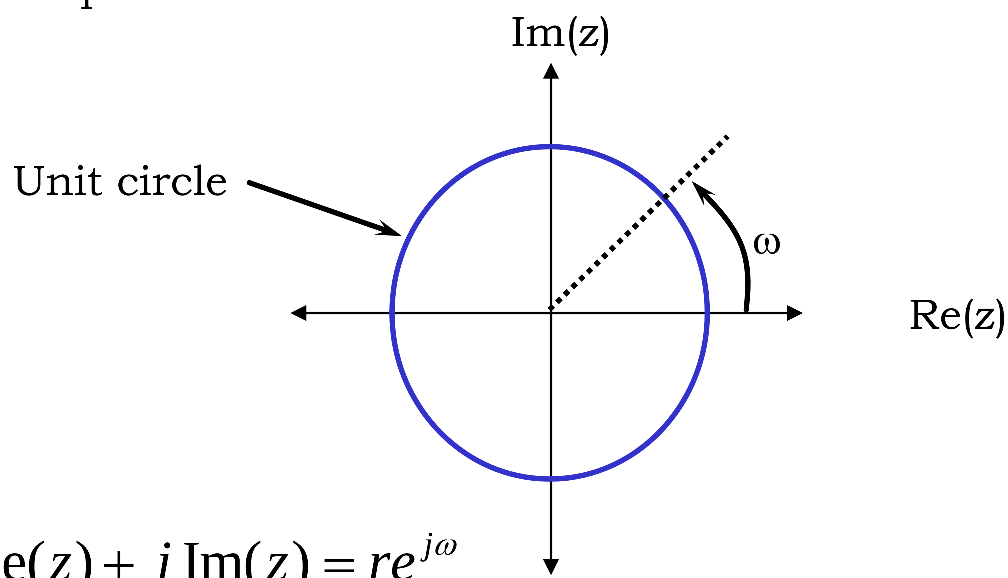
The **z-transform** is a generalization of the DTFT and is defined as:

$$X\left(z\right) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}$$

Where $z = re^{j\omega}$

Z-Transform

The z-transform is a complex valued function and is described using the complex plane.



$$z = \mathrm{Re}(z) + j\,\mathrm{Im}(z) = re^{j\omega}$$

The z-transform evaluated on the unit circle corresponds to the DFT

$$X\left(e^{j\omega}\right) = X\left(z\right)_{z=e^{j\omega}}$$

Z-Transform

Many of the signals of interest in DSP have *z*-transforms that are rational functions of *z*:

$$X(z) = \frac{B(z)}{A(z)} = \frac{\sum_{k=0}^{q} b(k) z^{-k}}{\sum_{k=0}^{p} a(k) z^{-k}}$$

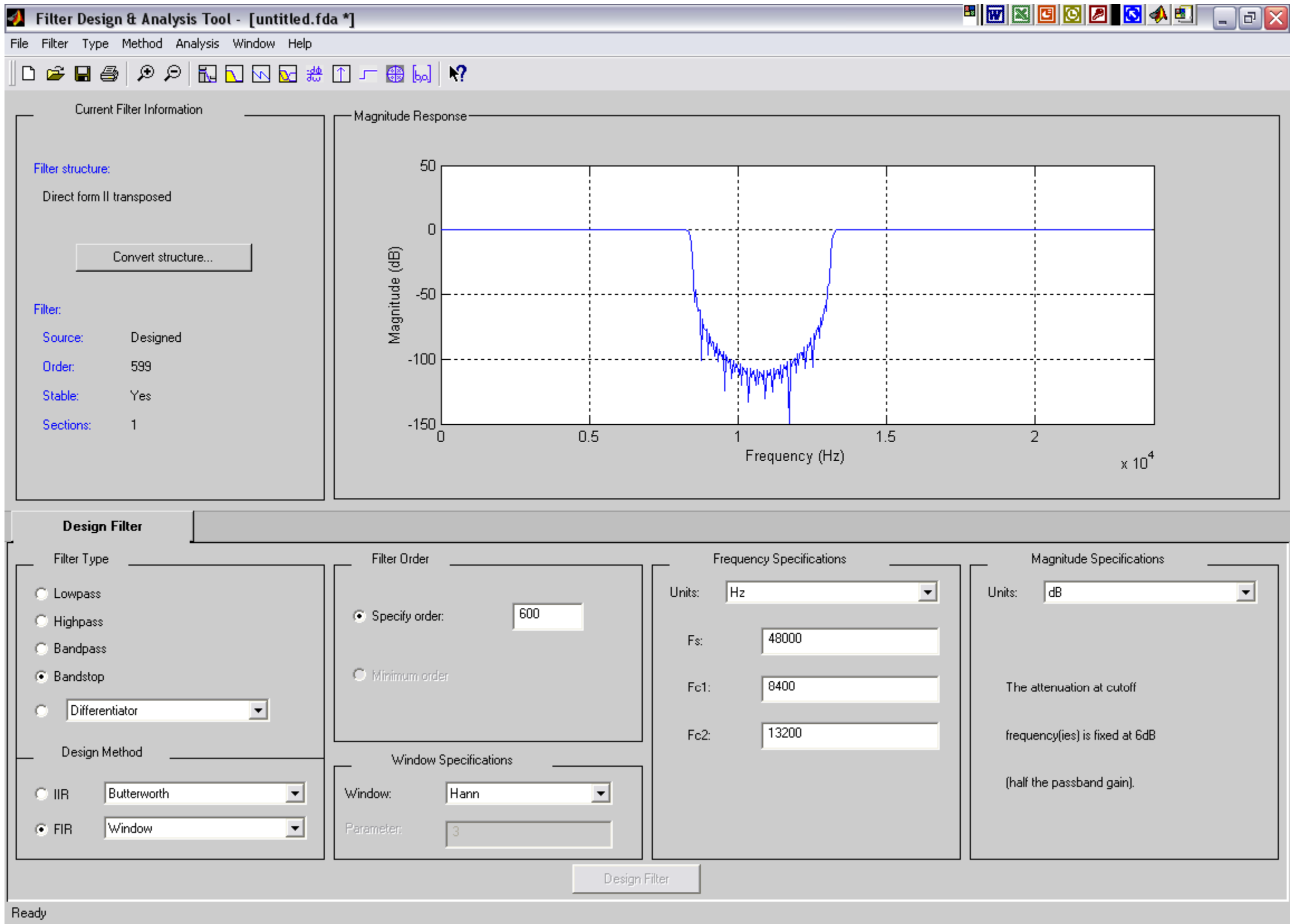
Factoring the numerator and denominator polynomials, a rational *z*-transform may be expressed as:

$$X(z) = C \frac{\prod_{k=1}^{q} (1 - \beta_k z^{-1})}{\prod_{k=1}^{p} (1 - \alpha_k z^{-1})}$$

The roots of the numerator polynomial are referred to as the ***zeros*** of *X(z),* and the roots of the denominator polynomial are refered to the ***poles***.

Filter Design Tools - MATLAB

- Signal Processing Toolbox – Filter Design and Analysis Tool



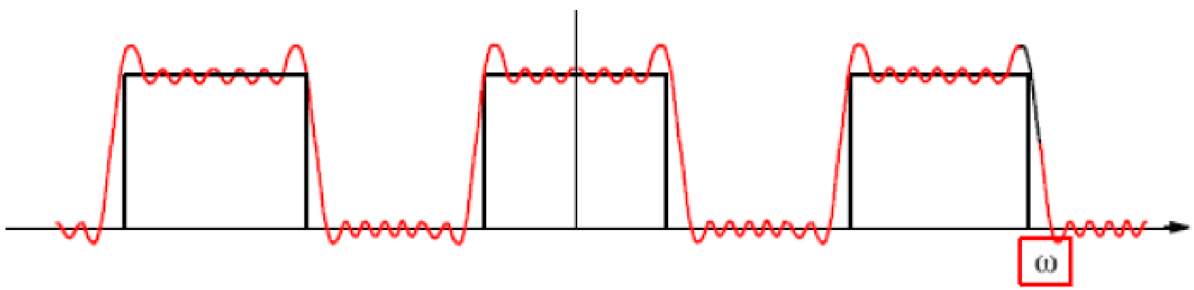
Design of FIR Filters

The frequency response of a filter can be expanded into the Fourier series.

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

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

The coefficients of the Fourier series are identical to the impulse response of the filter. Such a filter is not realizable however since it begins at -R and is infinitely long. It needs to be both truncated to make it finite and shifted to make it realizable. Direct truncation is possible but leads to the Gibbs phenomenon of overshoot and ripple illustrated below.



Design of FIR Filters

A solution to this is to truncate the Fourier series with a window function. This is a finite weighting sequence which will modify the Fourier coefficients to control the convergence of the series. Then

$$\hat{h}(n) = h(n)w(n)$$

where *w(n)* is the window function sequence and *hn* gives the required impulse response.

The desirable characteristics of a window function are

- a narrow main lobe containing as much energy as possible
- side lobes that decrease in energy rapidly as *w* tends to *p*.

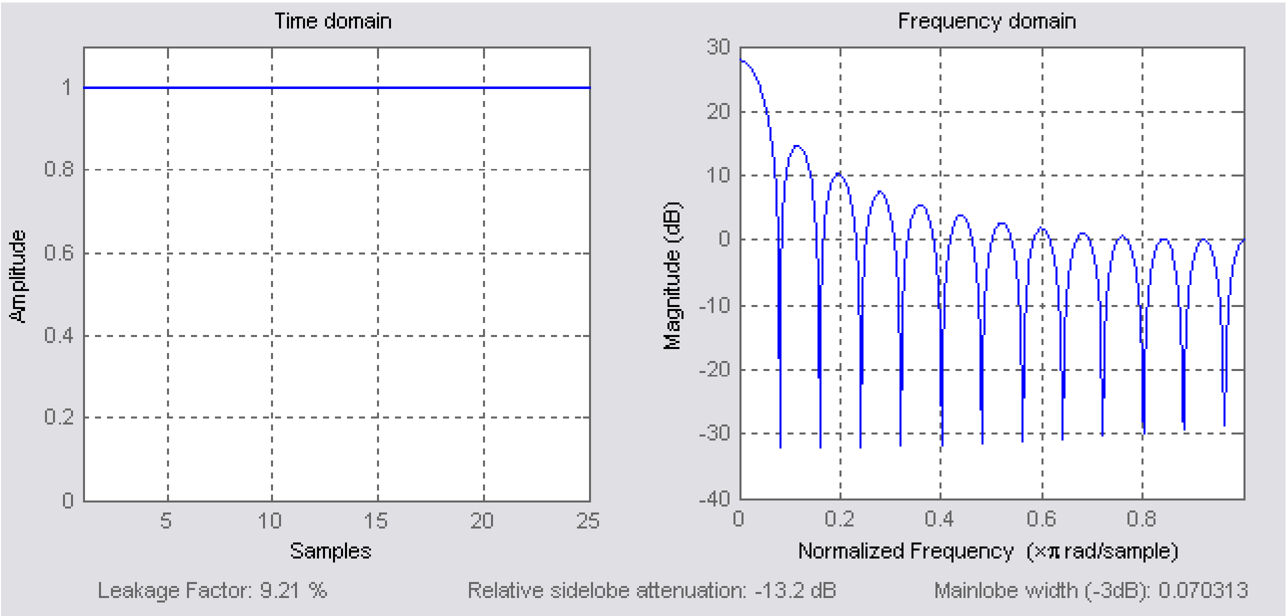
Design of FIR Filters

Rectangular

This is equivalent to direct truncation.

$$W(n) = 1 \; when \; \frac{-(N-1)}{2} < n < \frac{(N-1)}{2}$$

$$= 0 \text{ elsewhere}$$



Design of FIR Filters

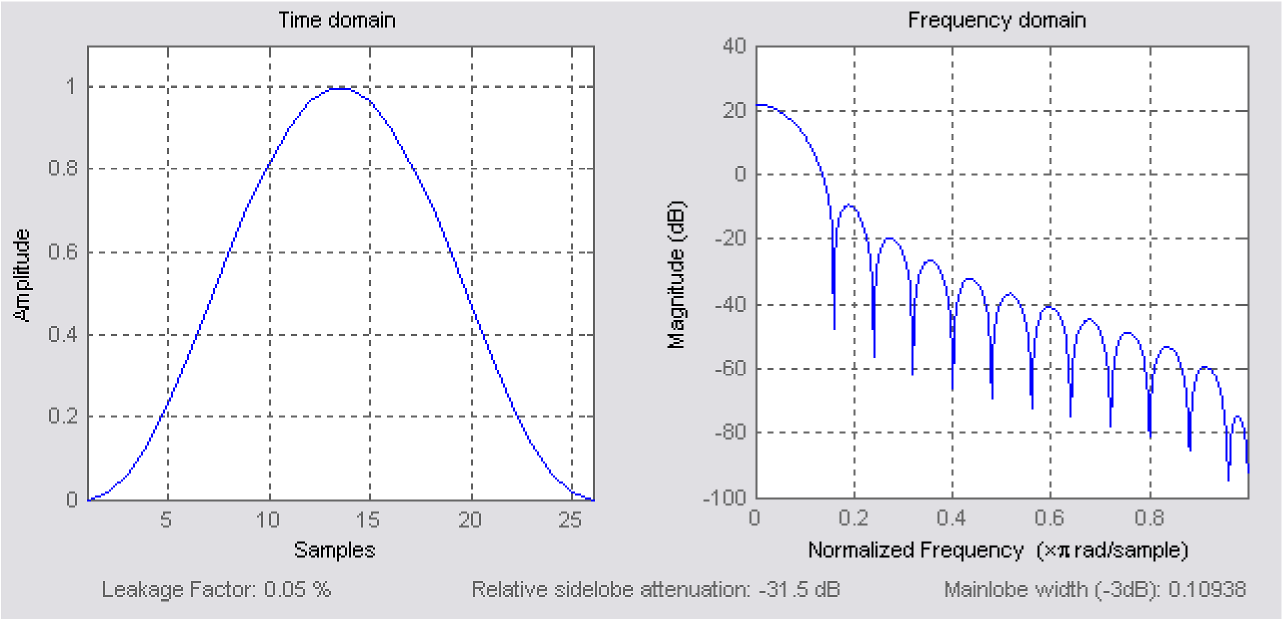
Hanning

This type of window trades off transition width for ripple cancellation. In this case

$$W(n) = \alpha + (1-\alpha)\cos\left(\frac{2\pi n}{N}\right) \quad when \quad \frac{-(N-1)}{2} < n < \frac{(N-1)}{2}$$

$$= 0 \text{ elsewhere}$$

$\alpha = 0.5$



MichiganTech

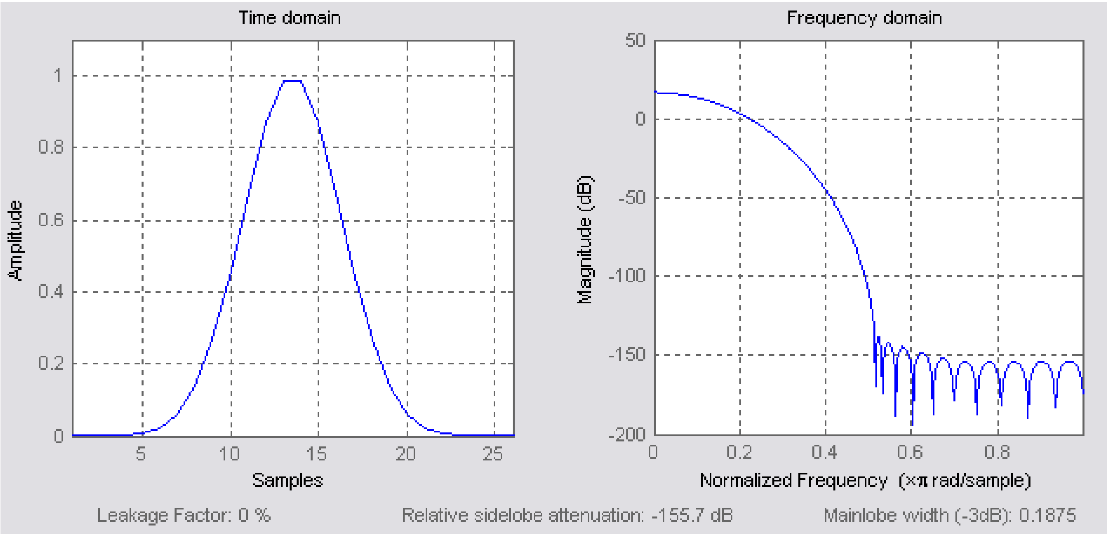
Design of FIR Filters

Kaiser

The Kaiser window function is a simplified approximation of a prolate spheroidal wave function which exhibits the desirable qualities of being a time-limited function whose Fourier transform approximates a band-limited function. It displays minimum energy outside a selected frequency band and is described by the following formula

$$W(n) = \frac{I_0\left(\beta\sqrt{1-[2n/(N-1)]^2}\right)}{I_0\beta} \quad when \; \frac{-(N-1)}{2} < n < \frac{(N-1)}{2}$$

Where I_0 is the zeroth order Bessel function and *b* is a constant representing a frequency trade-off between the height of the side lobe ripple and the width of the main lobe.

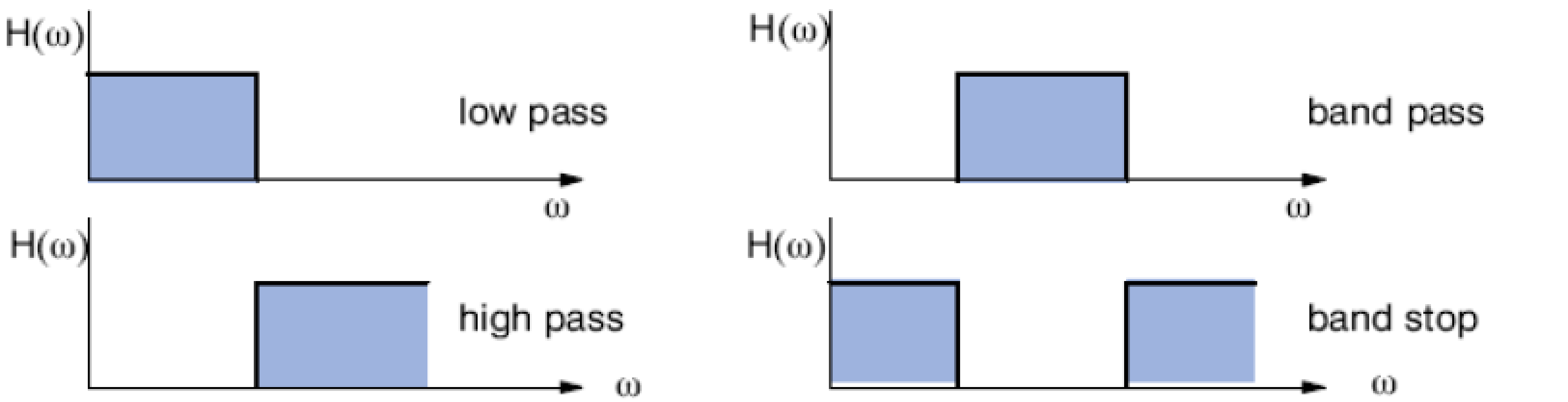


$$\beta = 20$$

Design of FIR Filters

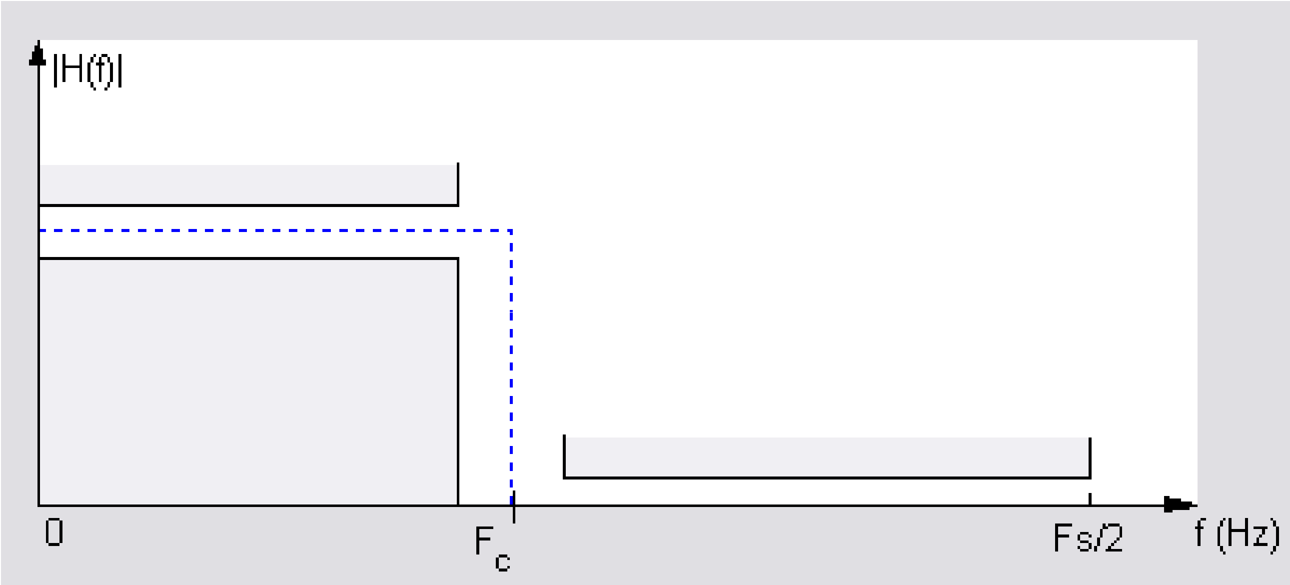
Steps in the Process

1. **Choose Filter Type (low pass, high pass, band pass, band stop)**
2. Specify cut-off frequency (ies) and sample rate
3. Form the Impulse Response Function
4. Specify order (truncation length)
5. Specify window (determines out of band properties)
6. Make it causal (determines linear phase shift)



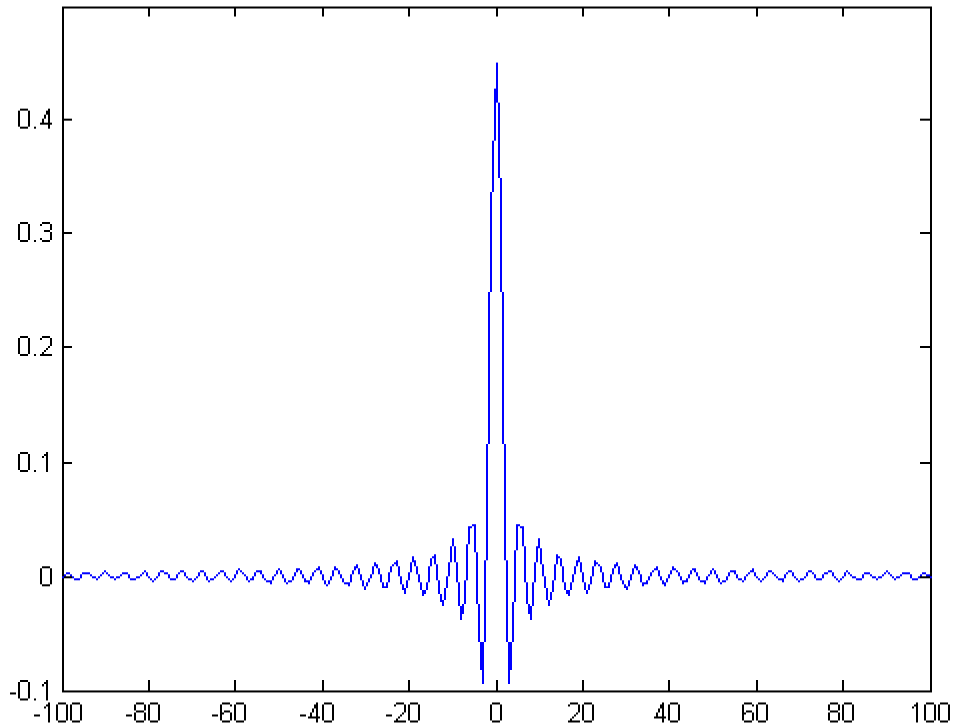
Design of FIR Filters

2. **Specify cut-off frequency (ies) and sample rate**



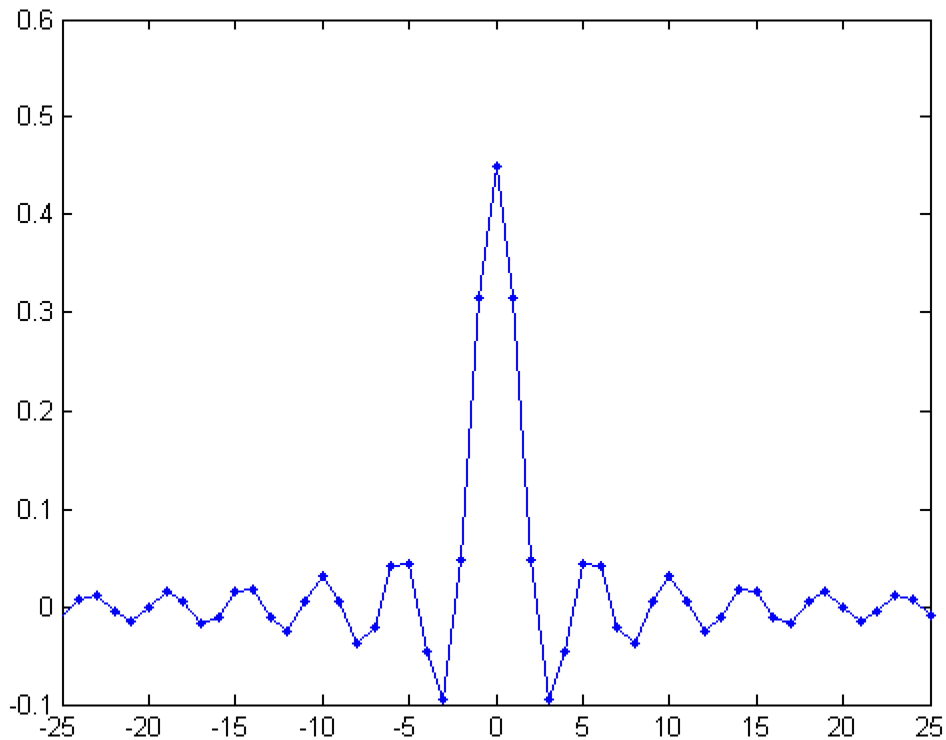
Design of FIR Filters

3. Form the Impulse Response Function



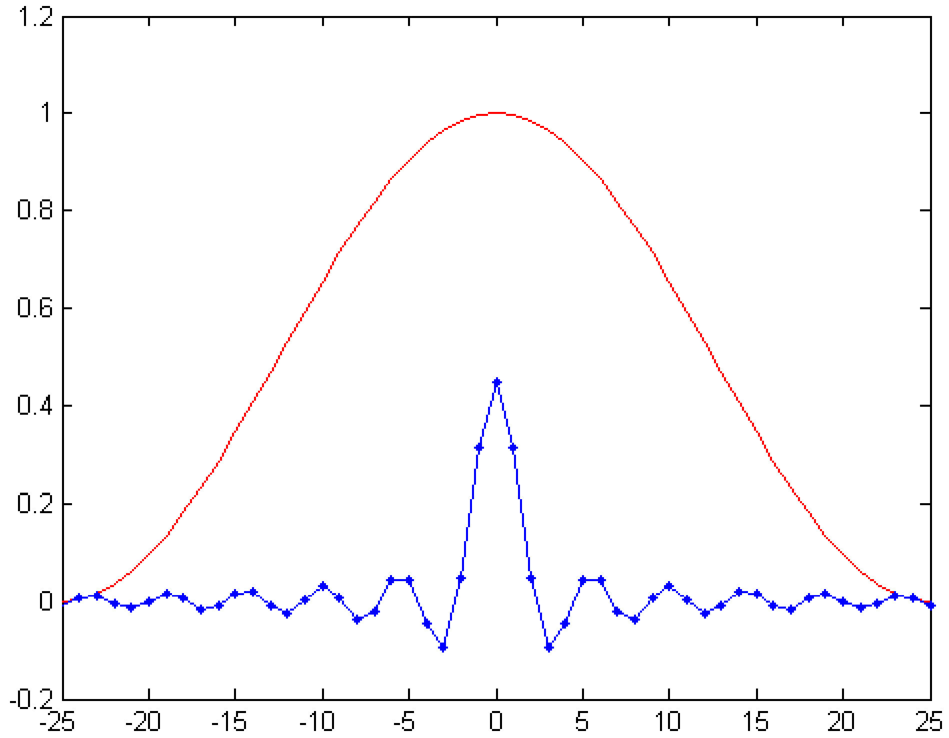
Design of FIR Filters

4. Specify order (truncation length)



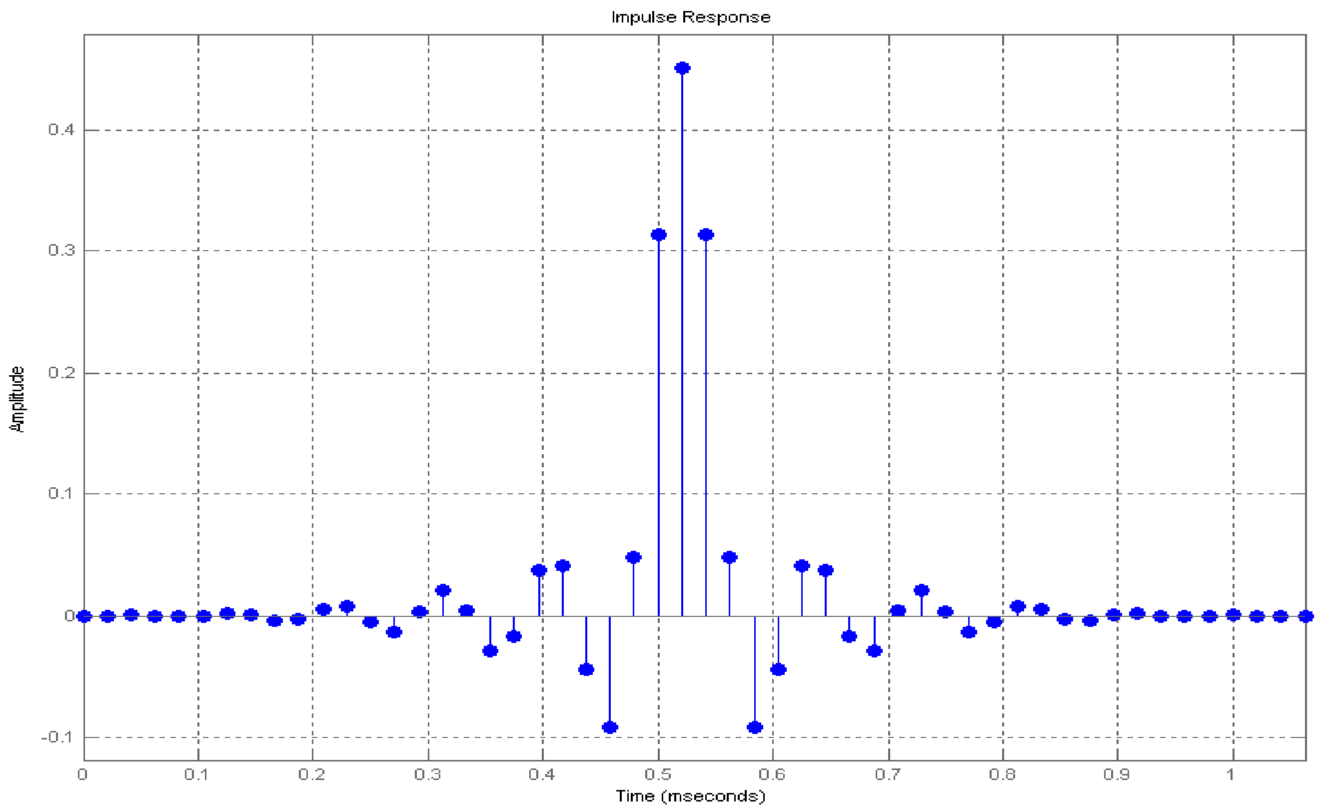
Design of FIR Filters

5. **Specify window (determines out of band properties)**



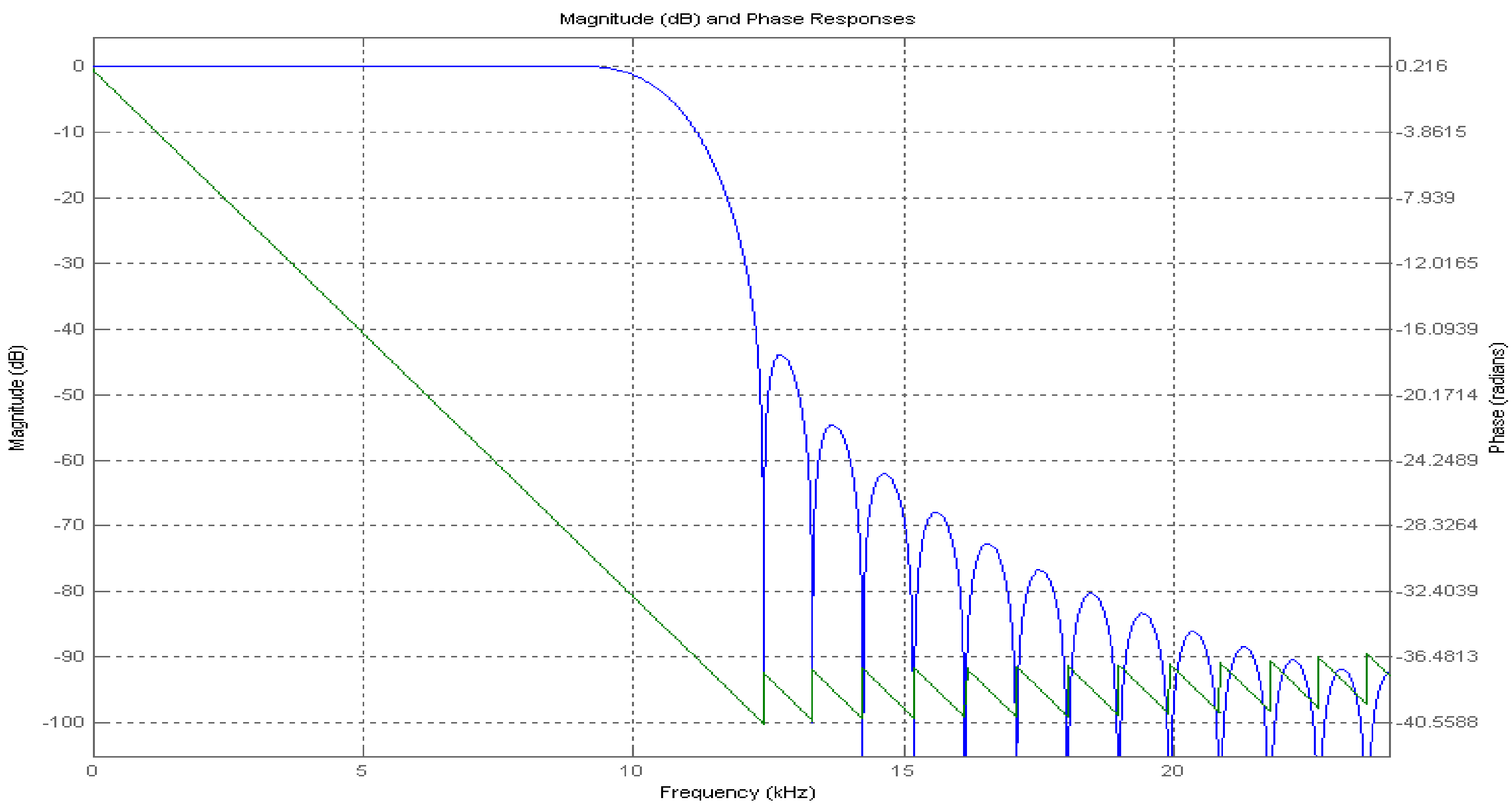
Design of FIR Filters

6. Make it causal (determines linear phase shift)



Design of FIR Filters

➢ **Frequency Response Characteristics**



Sinc Filters

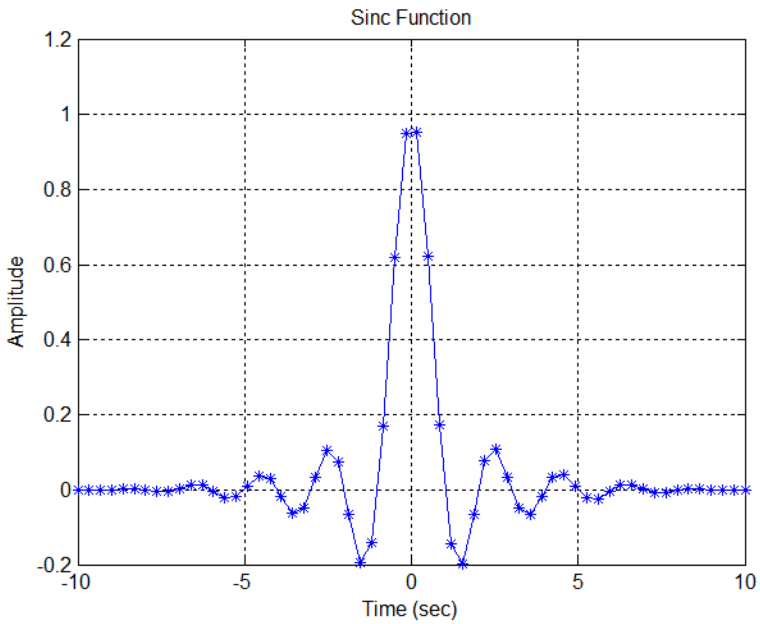
- Very effective design process for lowpass FIR filters is based off of the sinc function

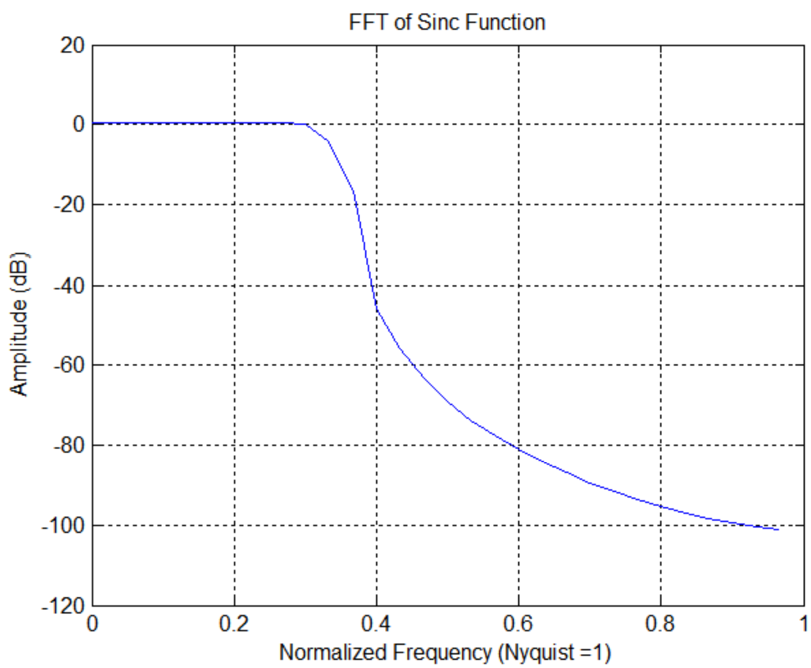
$$h(t) = \frac{\sin(\omega t)}{\omega t}$$

- Filter cutoff is determined by number of filter points between zero crossings of sinc function
  - 4 points between zeros defines filter cutoff of ¼ Nyquist
  - 2 points between zeros defines filter cutoff of ½ Nyquist

- Use different windows to truncate sinc function and control passband ripple and stopband attenuation
- Number of zero crossings determines stopband attenuation

Sinc Filter



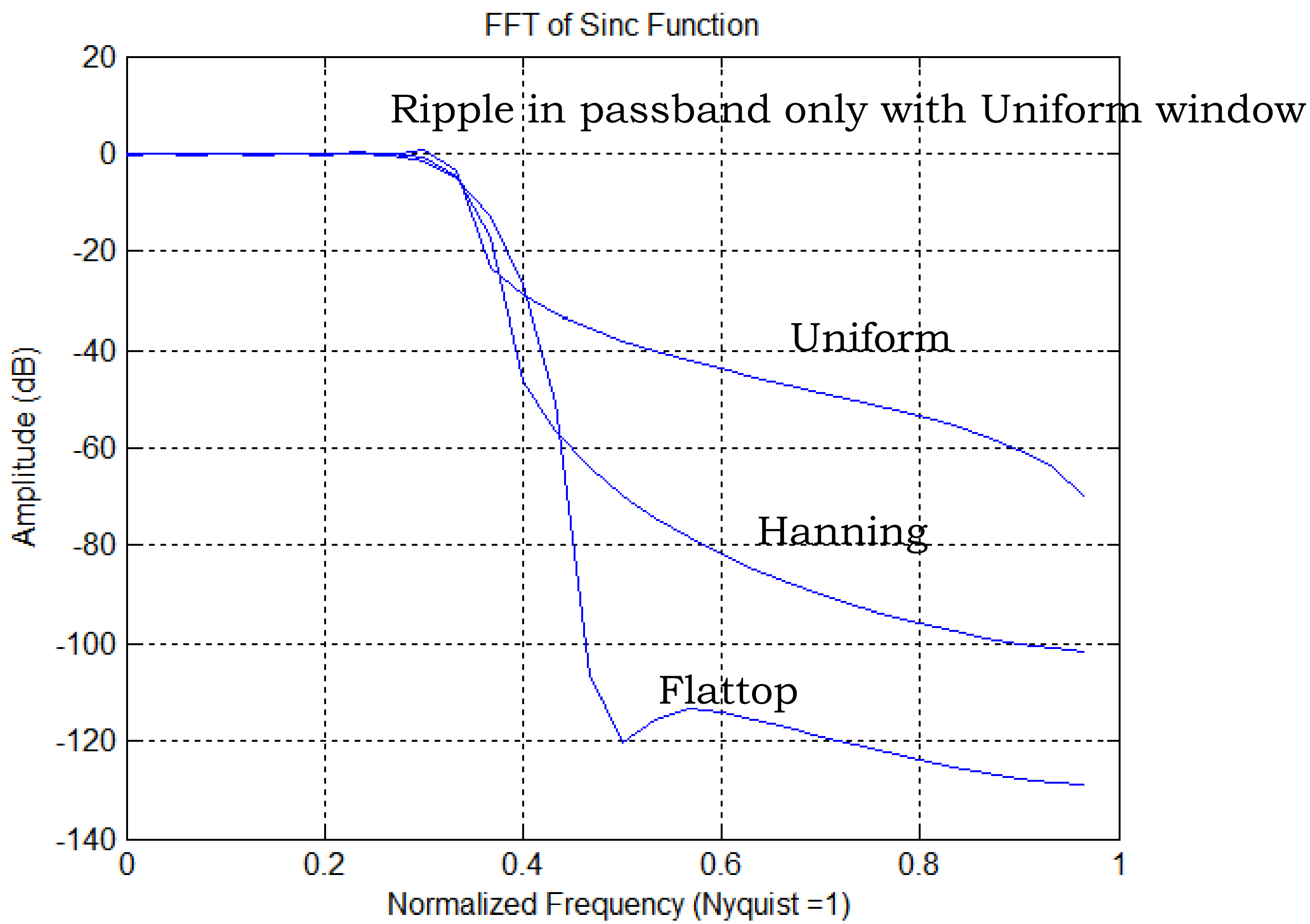


Hanning Window

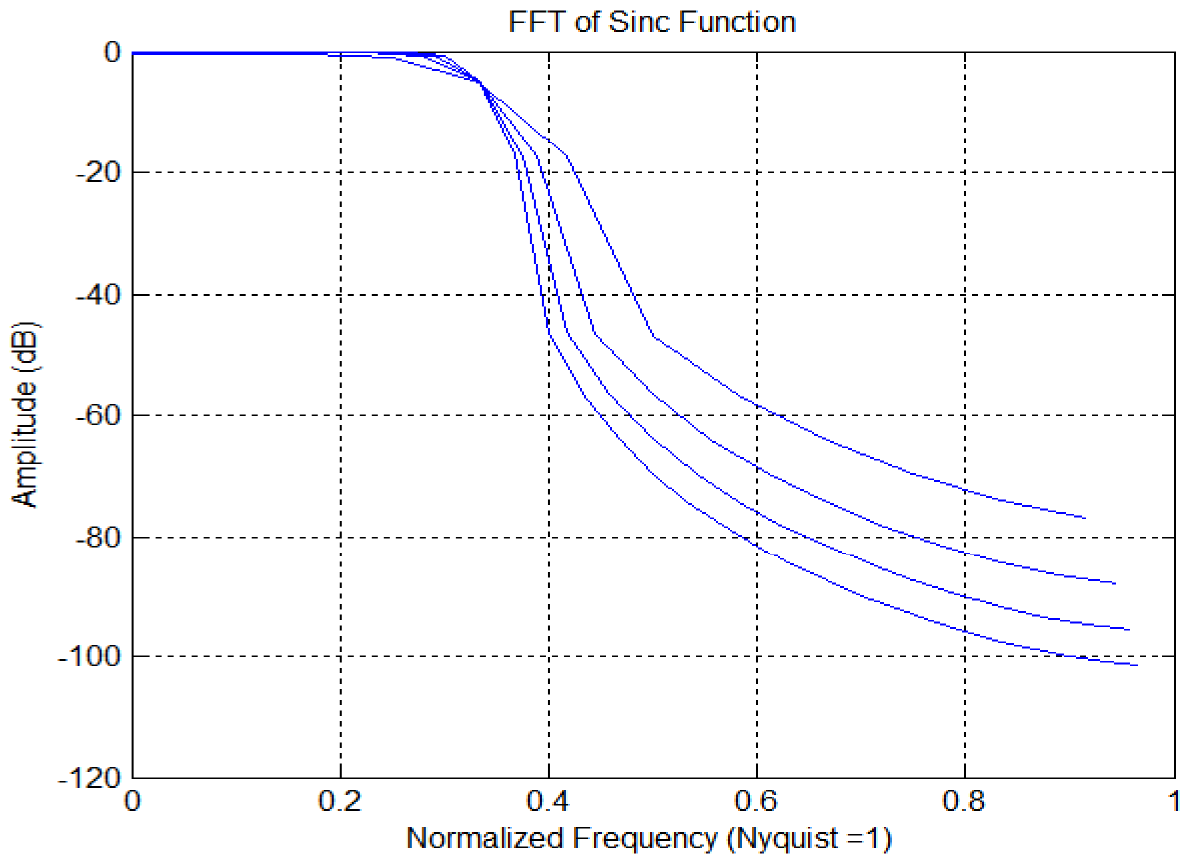
3 points/zero crossing

10 zero crossings each side

Sinc Filter – Window Effects



Sinc Filter – Effect of Number of Zero Crossings

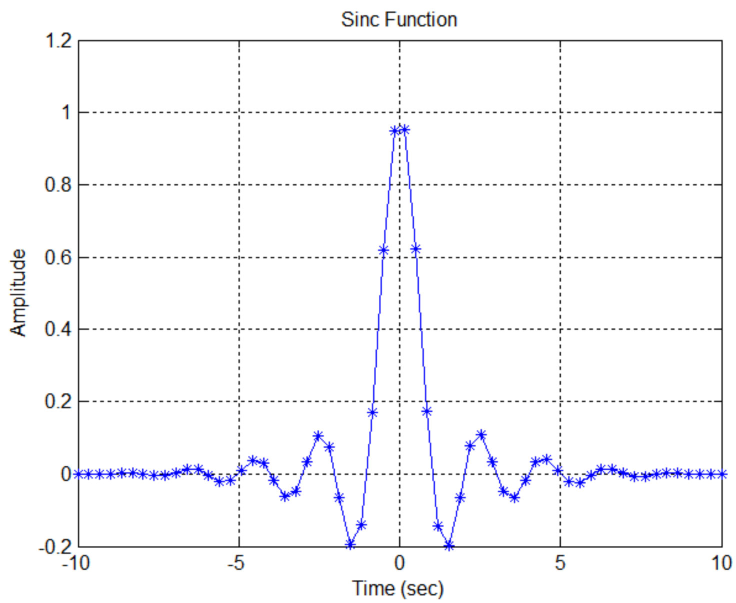


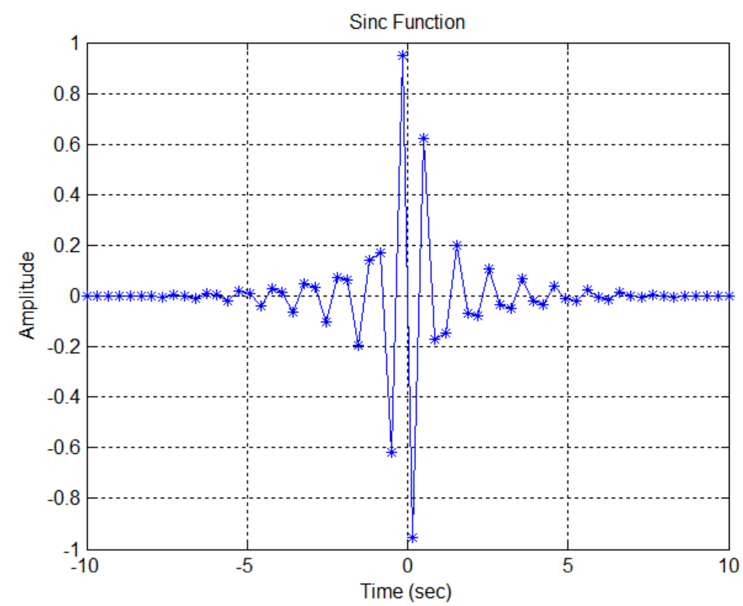
Filter has 10, 8, 6, and 4 zero crossings on each side …filters organized in same way from left to right in transition band

Spectral Reversal

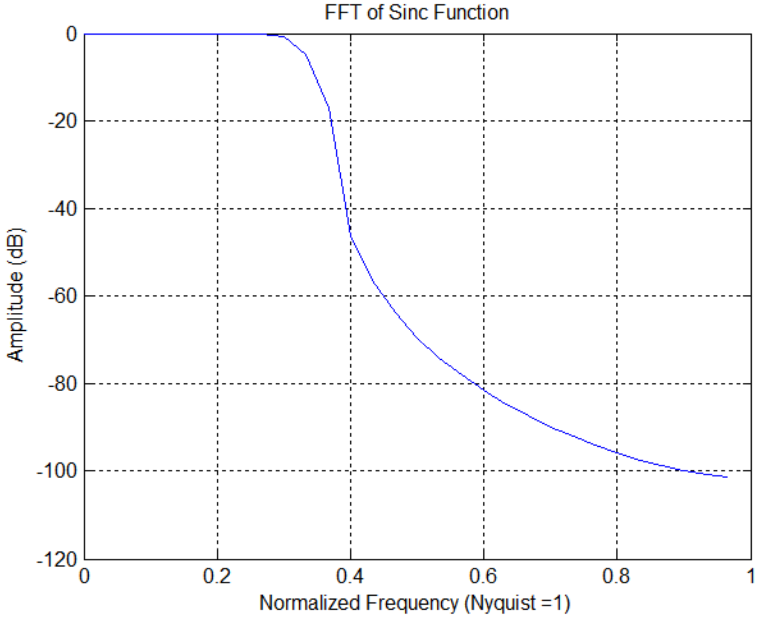
- Low Pass FIR filter can be transformed to High Pass FIR filter through spectral reversal
  - Mirrors frequency response of filter about ½ Nyquist

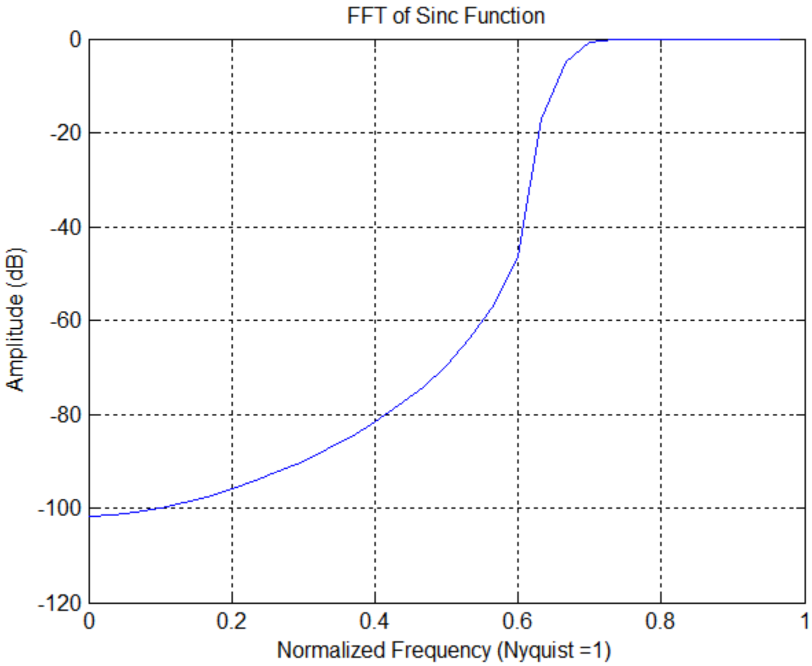
- Multiple every other point in filter by -1





Spectral Reversal

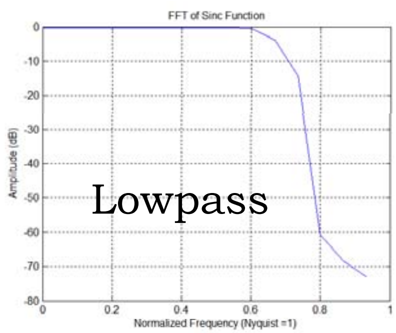




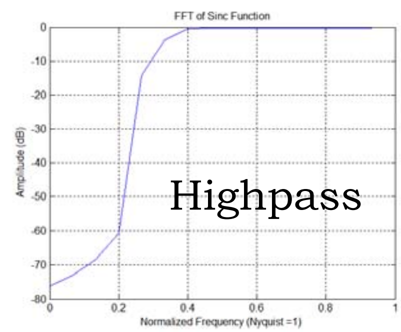
Bandpass Filters

$$X(n) \longrightarrow \boxed{h_1(n) \otimes h_2(n)} \longrightarrow Y(n)$$

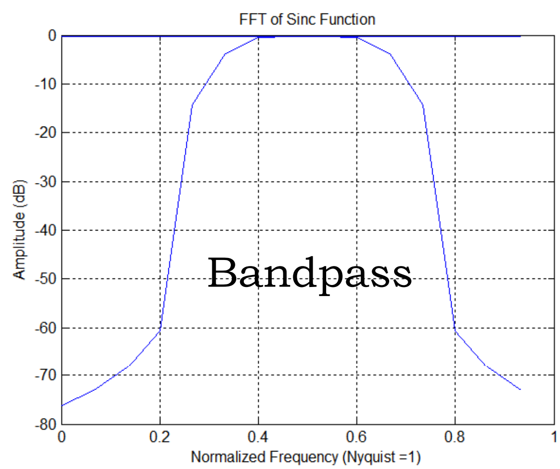
X(n) →



→

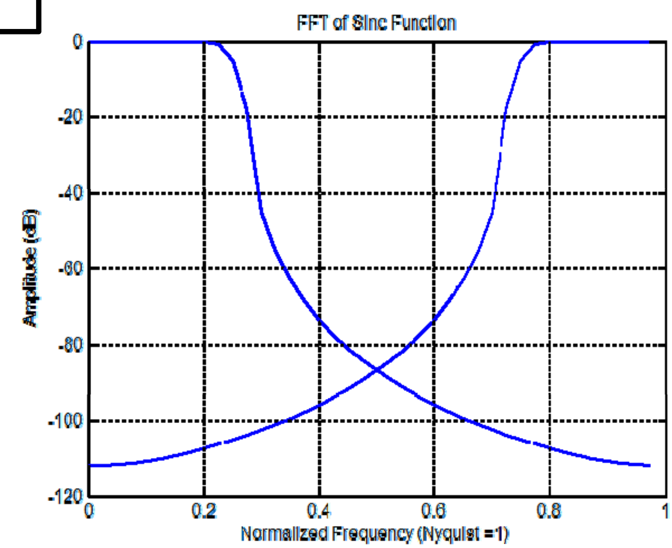


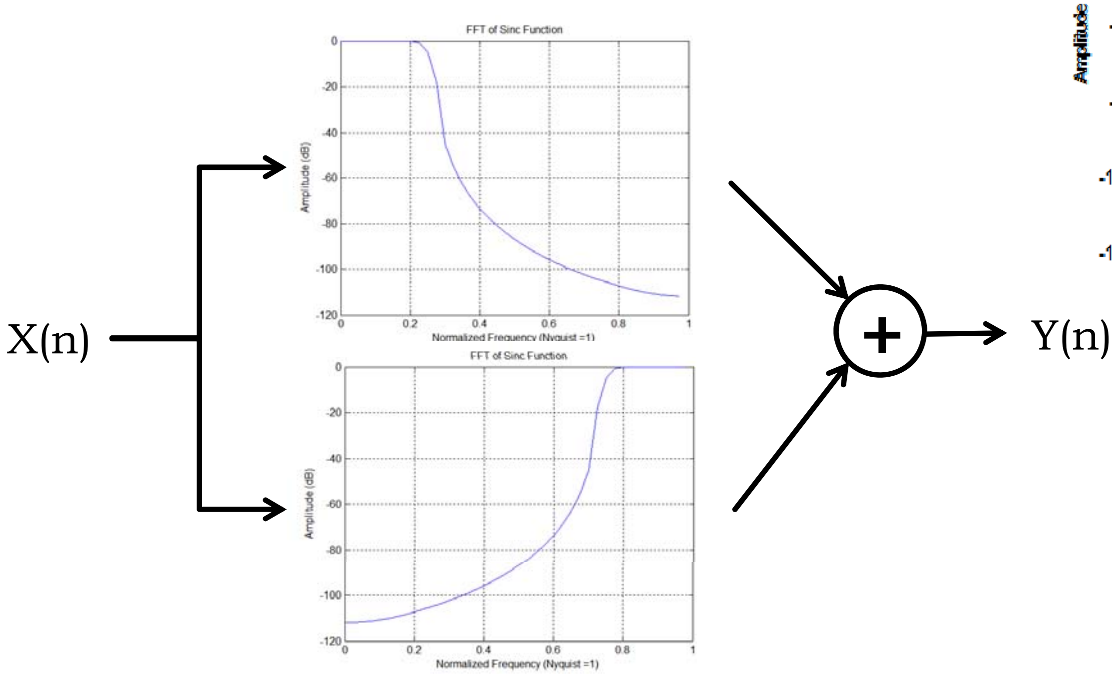
→ Y(n)



Bandstop Filters

$$X(n) \longrightarrow \boxed{h_1(n)+h_2(n)} \longrightarrow Y(n)$$



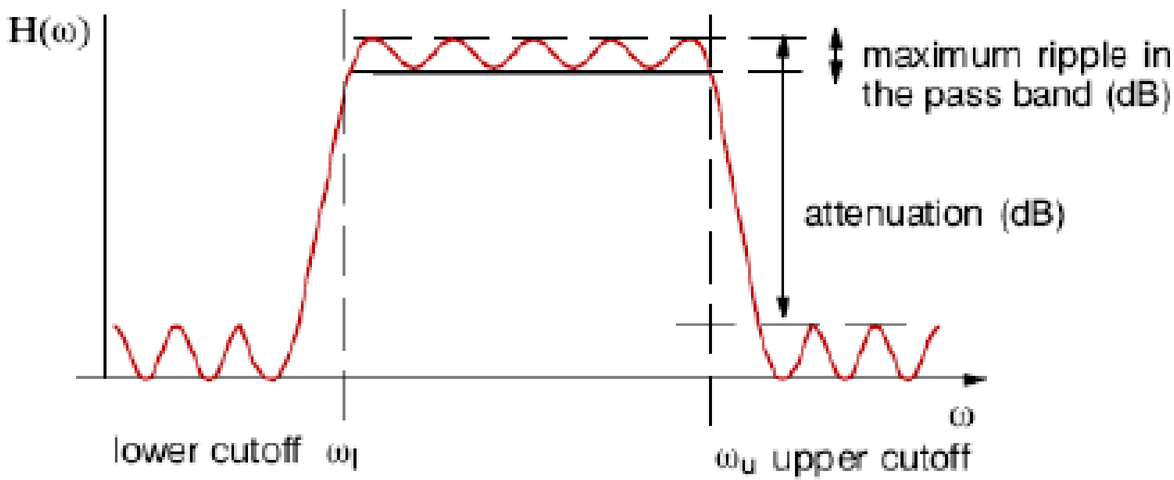


Design of IIR Filters

Design of IIR filters using analog prototypes

The steps involved in this design process are described in the following subsections.

Step 1) Specify the filter characteristics



Design of IIR Filters

Step 2) Compute the analog frequencies

A prototype low pass filter will be designed based on the required digital cutoff frequency w_c. First however the digital frequency w_d must be converted to an analog one w_a. This is achieved through a bilinear transformation from the digital (z) plane to the analog *(s)* plane where *s* and *z* are related by

$$s = \frac{2}{T}\left(\frac{1 - z^{-1}}{1 + z^{-1}}\right)$$

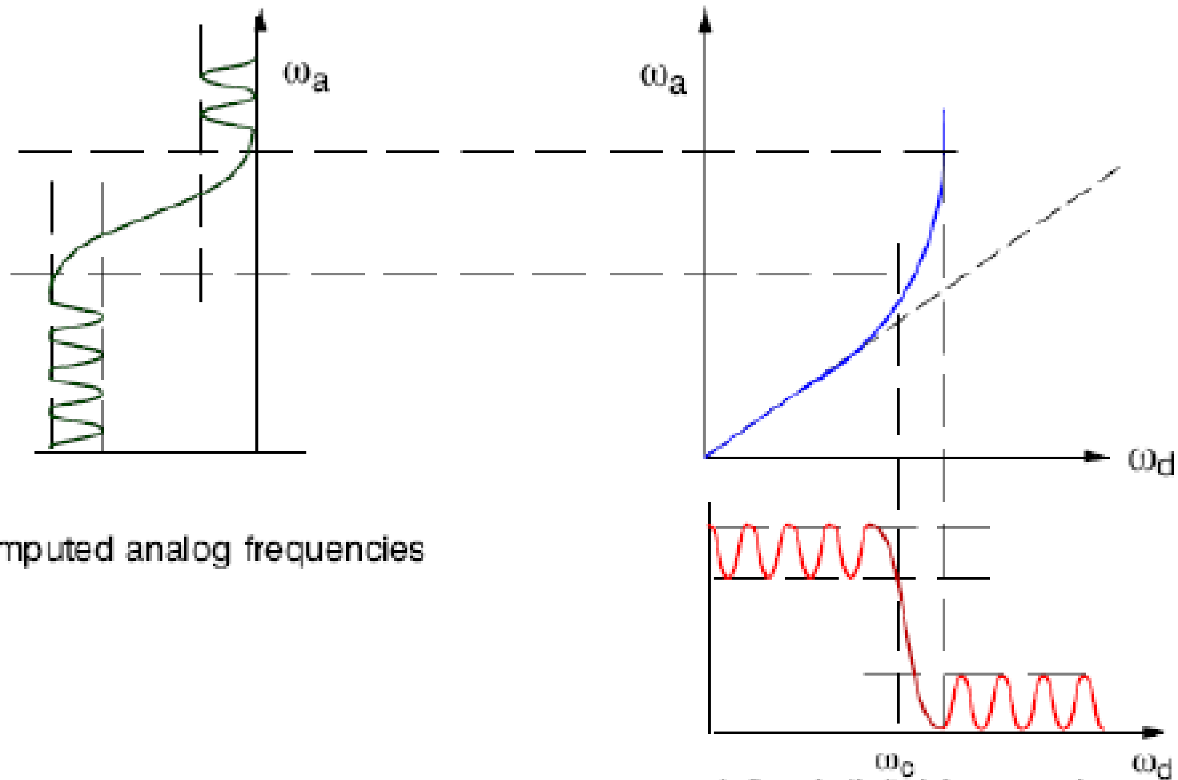
When $z = e^{jwT}$ (the unit circle) and $s = jw_a$

$$s = \frac{2}{T}\left(\frac{1 - e^{-j\omega T}}{1 + e^{-j\omega T}}\right) = \frac{2}{T} j \tan(\omega_d T/2)$$

$$\omega_a = \frac{2}{T}\tan(\omega_d T/2)$$

Design of IIR Filters

The analog *w* axis is mapped onto one revolution the of the unit circle, but in a non-linear fashion. It is necessary to compensate for this nonlinearity (warping) as shown below



computed analog frequencies

defined digital frequencies

Design of IIR Filters

Step 3) Select the suitable analog filter

- Bessel
- Butterworth
- Chebyshev type I
- Inverse Chebyshev type II
- Cauer (elliptical)

Design of IIR Filters

Bessel filters

The goal of the Bessel approximation for filter design is to obtain a flat delay characteristic in the passband. The delay characteristics of the Bessel approximation are far superior to those of the Butterworth and the Chebyshev approximations, however, the flat delay is achieved at the expense of the stopband attenuation which is even lower than that for the Butterworth. The poor stopband characteristics of the Bessel approximation make it impractical for most filtering applications !

Bessel filters have sloping pass and stop bands and a wide transition width resulting in a cutoff frequency that is not well defined.

The transfer function is given by

$$H(s) = \frac{d_0}{B_n(s)}$$

where $B_n(s)$ is the *nth* order Bessel polynomial

$$B_n(s) = (2n - 1)B_{n-1}(s) + s^2 B_{n-2}(s)$$

and d_0 is a normalizing constant.

$$d_0 = \frac{(2n)!}{2^n n!}$$

Design of IIR Filters

Butterworth filters

These are characterized by the response being maximally flat in the pass band and monotonic in the pass band and stop band. Maximally flat means as many derivatives as possible are zero at the origin. The squared magnitude response of a Butterworth filter is

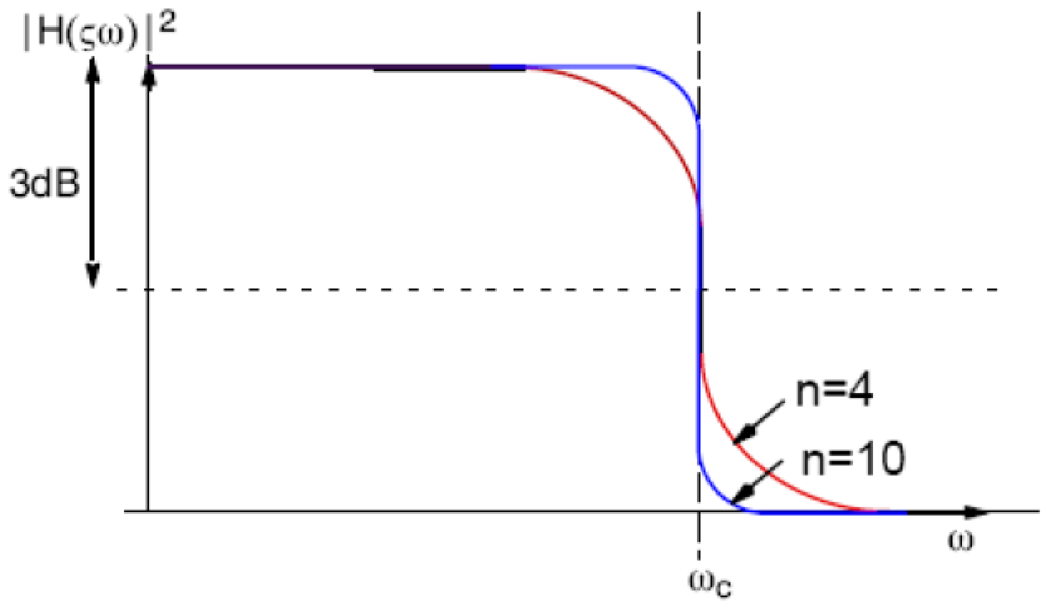
$$|H(s)|^2 = \frac{1}{1 + (\frac{s}{s_c})^{2n}}$$

where n is the order of the filter. The transfer function of this filter can be determined by evaluating equation

$$|H(j\omega)|^2 = H(s)H(-s) = \frac{1}{1 + (-\frac{s^2}{j\omega_c^2})^n}$$

Design of IIR Filters

Butterworth filters are all-pole filters i.e. the zeros of *H(s)* are all at $s=\infty$.
They have magnitude (1/?2) when $\omega/\omega_c = 1$ i.e. the magnitude response is down 3dB at the cutoff frequency.



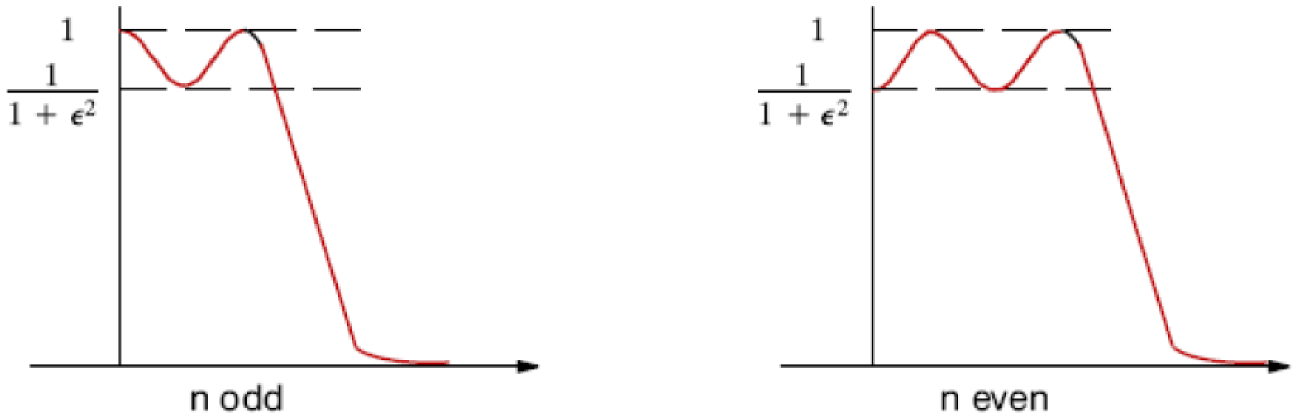
Design of IIR Filters

<u>Chebyshev (type I) filters</u>

These are all pole filters that have equi-ripple pass bands and monotone stop bands. The formula is

$$|H(\omega)|^2 = \frac{1}{1 + \epsilon^2 C_n^2(\omega)}$$

where $C_n(w)$ are the Chebyshev polynomials and e is the parameter related to the ripple in the pass band as shown below for *n* odd and even.



For the same loss requirements, the Chebyshev approximation usually requires a lower order than the Butterworth approximation, but at the expense of an equi-ripple passband. Therefore, the transition width of a Chebyshev filter is narrower than for a Butterworth filter of the same order.

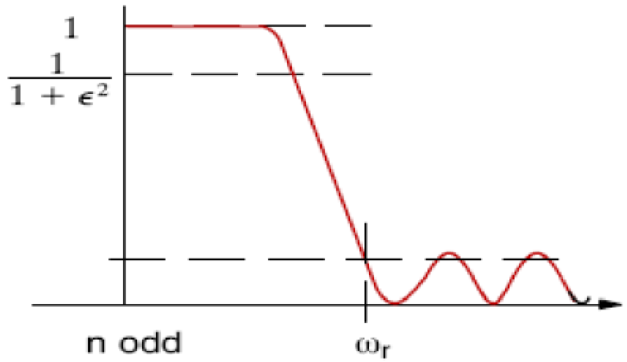
Design of IIR Filters

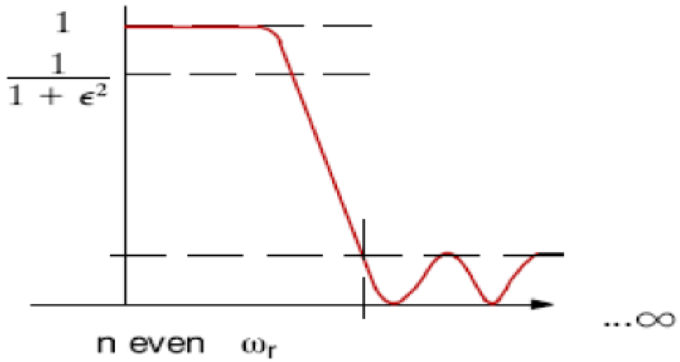
Inverse Chebyshev (type II) filters

These contain poles and zeros and have equi-ripple stop bands with maximally flat pass bands. In this case

$$|H(\omega)|^2 = \frac{1}{1 + \epsilon^2\left[\frac{C_n(\omega_r)}{C_n(\omega_r/\omega)}\right]^2} \qquad Eqn\ 5\text{–}32$$

where $C_n(w)$ are the Chebyshev polynomials, e is the pass band ripple parameter and w_r is the lowest frequency where the stop band loss attains a specified value. These parameters are illustrated below for *n* odd and even.



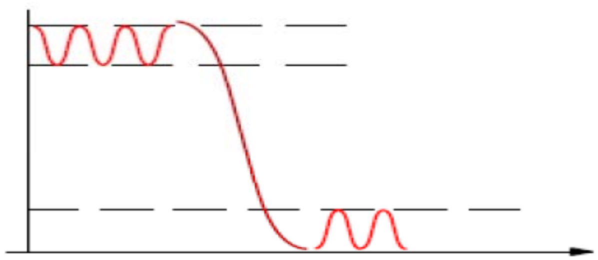


For the same loss requirements, the Inverse Chebyshev approximation usually requires a lower order than the Butterworth approximation, but at the expense of an equi-ripple stopband.

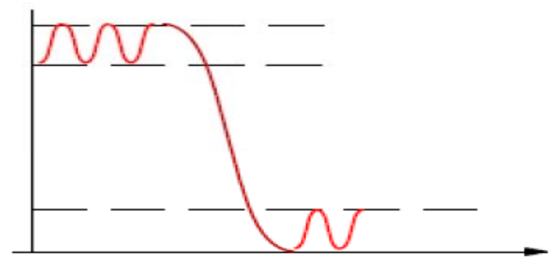
Design of IIR Filters

Cauer (elliptical) filter

These filters are optimum in the sense that for a given filter order and ripple specifications, they achieve the fastest transition between the pass and the stop band (i.e. the narrowest transition band). They have equi-ripple stop bands and pass bands.



n odd



n even

The transfer function is given by

$$|H(\omega)|^2 = \frac{1}{1 + \epsilon^2 R_n^2(\omega L)}$$

where $R_n(wL)$ is called a Chebyshev rational function and L is a parameter describing the ripple properties of $R_n(wL)$. The determination of $R_n(wL)$ involves the use of the Jacobi elliptic function. e is a parameter related to the passband ripple.

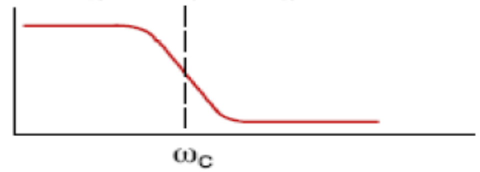
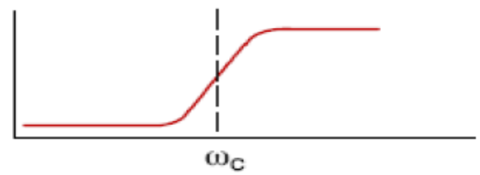
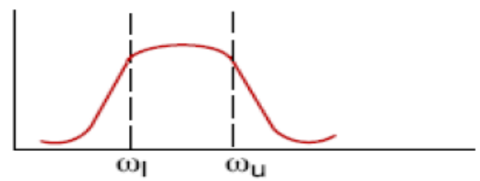
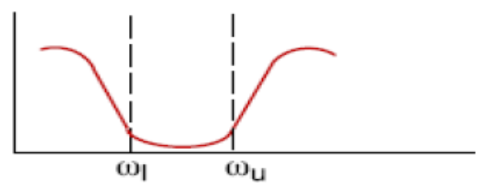
This group of filters is characterized by the property that the group delay is maximally flat at the origin of the *s* plane. However this characteristic is not normally preserved by the bilinear transformation and it has poor stop band characteristics.

For a given requirement, this approximation will in general require a lower order than the Butterworth or the Chebyshev ones. The Cauer approximation will thus lead to the least costly filter realization, but at the expense of the worst delay characteristics.

Design of IIR Filters

Step 4) Transform the prototype low pass filter

At this point we have selected a suitable low pass filter prototype with a normalized cutoff frequency $w_c = 1$. The next stage is to transform this low pass filter into the type of analog filter required with the desired cutoff frequencies. To achieve this the following transformations are applied.

Transform	Replace s by	Frequency response
Low pass to low pass	$s \rightarrow \frac{s}{\omega_c}$	
Low pass to high pass	$s \rightarrow \frac{\omega_c}{s}$	
Low pass to band pass	$s \rightarrow \frac{s^2 + \omega_u \omega_l}{s(\omega_u - \omega_l)}$	
Low pass to band stop	$s \rightarrow \frac{s(\omega_u - \omega_l)}{s^2 + \omega_u \omega_l}$	

MichiganTech

Design of IIR Filters

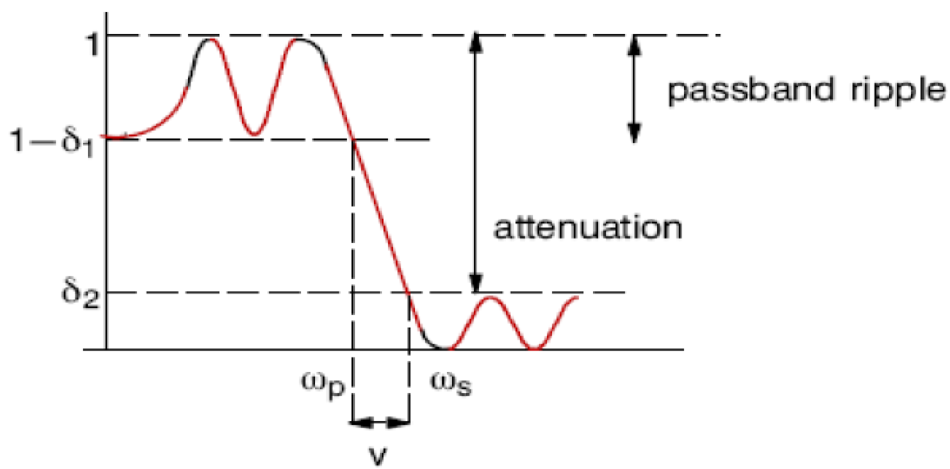
Step 5) Apply a bilinear transformation

The final stage in this design process is to apply a bilinear transformation to map the *(s)* plane to the *(z)* plane to obtain the desired digital filter.

$$H(z) = H(s)\Big|_{s=\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)}$$

The final result is a set of filter coefficients *a* and *b*, stored in vectors of length *n+1*,where *n* is the order of the filter. A facility, described below, enables you to determine the optimum order of a filter required for a particular design.

<u>Determining the filter order</u>



Specifications required to determine filter order

Design of IIR Filters

Ripple passband	This determines the ripple parameter d_1. It is expressed in dB
Attenuation	When this is defined, the ripple parameter d_2 is determined. It is expressed in dB.
Lower frequency Upper frequency	These are the two edge frequencies w_p (end of the pass band) and w_s (start of the stop band) of a low pass or high pass filter. Band pass and band stop filters will require a second pair of frequencies to be defined.
Sampling frequency	This is the sampling frequency at which the filter must operate.

The filter can be any one of the types mentioned above and the prototype can be either a Butterworth, Chebyshev type I or type II or a Cauer filter. This process does not apply to the Bessel filter because of the particular condition pertaining to these filters in that the filter order affects the cutoff frequency. The minimum filter order required is determined from a set of functions described below.

One function relates the pass band and stop band ripple specifications to a filter design parameter *h* where

$$\eta = 2\frac{\sqrt{\delta_1\delta_2}}{(1-\delta_1)\sqrt{(1-\delta_1^2)-\delta_2^2}}$$

Design of IIR Filters

Another parameter relates the pass band cut off frequency w_p, the transition width v and the low pass filter transition ratio k where

$$k = \frac{\omega_p}{\omega_s} = \frac{\tan \omega_p/2}{\tan \omega_s/2}$$

analog digital

A final function relates the filter order n, the low pass filter transition ratio k and the filter design parameter h. This relationship depends on the type of prototype analog filter.

$$n = \frac{\eta}{k}$$ Butterworth

$$n = \frac{\cosh^{-1}(1/\eta)}{\ln\left(\frac{1-\sqrt{1-k^2}}{k}\right)}$$ Chebyshev

$$n = \frac{K(k)K\sqrt{(1-\eta^2)}}{K(\eta)K(1-k^2)}$$ Elliptic

where $K()$ is the complete elliptical integral of the first kind.

Zero Phase Filtering

- Standard filtering
  - Convolve the impulse response of the filter with the data.
  - All filters have a phase delay – filtered data will have a phase delay!

- Zero phase filtering
  - Convolve the impulse response of the filter with the data.
  - Convolve the filtered data in reverse with the filter.

  - Results in zero phase delay between original data and filtered data
  - Twice the attenuation in the stop band!
  - Twice the computation time – applying the filter twice!

Long FFT Filtering

- Filtering is time domain convolution.
  - Equivalent procedure is multiplication in the frequency domain.

- Procedure:
  - FFT entire time history as one block.
  - Create desired filter shape in the frequency domain.
  - Multiply the FFT'd data and filter spectral line by spectral line in the frequency domain.
  - IFFT the resultant spectrum to get filtered time history.

- Advantage: Arbitrary filter shape possible.