DESIGNING OF DIGITAL FILTERS USING POLE ZERO PLACEMENT TECHNIQUES

Project Report submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology.

in

Electronics and Communication Engineering

under the Supervision of

Dr. (Prof.) Sunil Bhooshan

By

Priyanka Dhir (081017) and Shruti Gupta (081035)

to

Jaypee University of Information and Technology

Waknaghat, Solan – 173234, Himachal Pradesh

Certificate

This is to certify that project report entitled "Designing of digital filters using pole zero placement techniques", submitted by Priyanka Dhir and Shruti Gupta in partial fulfillment for the award of degree of Bachelor of Technology in Electronics and Communication Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other University or Institute for the award of this or any other degree or diploma.

Date: 17th May 2012

Supervisor's Name: Dr. (Prof.) Sunil Bhooshan

Designation: H.O.D. Electronics Deptt.

Acknowledgement

We thank our supervisor, Dr. (Prof.) Sunil Bhooshan, for his prompt and detailed advice, for his valuable guidance and his hard work, showing us what an academic professional ought to be. We thank his enthusiasm about our work and his exceptional patience. We also thank him for encouraging us to work hard. Without him, this project could not have been done.

We are also grateful to other members in our panel, Dr. T.S.Lamba, Mr. V.M. Shrivastav and Mr. Pardeep Garg. We appreciate their creative suggestions about our work and their valuable comments.

Date: 17th May 2012

Priyanka Dhir and Shruti Gupta

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS & ACRONYMS

DFT – Discrete Fourier Transform

IDFT – Inverse Discrete Fourier Transform

FFT – Fast Fourier Transform

IFFT – Inverse Fourier Transform

LPF – Low Pass Filter

$\sum$ Summation of

$\triangleq$ Congruent

LIST OF TABLES

ABSTRACT

This report provides the reader with a comprehensive study of a new technique to design Finite Impulse Response Filters using pole zero placements. In contrast to the traditional approach, in which the design is based on bilinear transform of an analog filter, the presented procedure allows for designing the filter directly in the digital domain using a dartboard approach. And hence, an intuitive technique known as pole-zero placements, is treated here in a quantitative way, whereby random placement of poles and zeros in different numbers has been used to study the effect on the response of the filter. The approaches used to achieve this project are literature review and computer software simulation using the software MATLAB. In the beginning, this report covers a brief introduction to the concept of digital and analog filters, a study of the required characteristics of realizable filters, existing methods of Finite Impulse Response Filter design namely windowing method, frequency sampling method and optimization techniques in detail along with their merits and demerits. And the second half of the report deals with the optimization of this new pole-zero placements technique that tries to overcome these shortcomings. In an attempt to deduce a general protocol for this technique, experimentation has been done with poles and zeros to better understand their effect on the transfer function associated with a first/ second order low pass filter and later shifted to higher orders. The study extensively covers work with poles and zeros - single, multiple and conjugate pairs to list out norms for optimizing transition band, tolerances and ripple. MATLAB has been used as the basic tool for the experimentation and various plots have been observed and included in the report as well for better understanding of the project.

Chapter 1: INTRODUCTION

1.1. Introduction

FIR filters are filters having a transfer function of a polynomial in z- and is an all-zero filter in the sense that the zeroes in the z-plane determine the frequency response magnitude characteristic. The z transform of an N-point FIR filter is given by

$$H(z) = \sum h(n)z^{-n}$$

FIR filters are particularly useful for applications where exact linear phase response is required.

The FIR filter is generally implemented in a non-recursive way which guarantees a stable filter.

FIR filter design essentially consists of two parts

(i) Approximation problem

(ii) Realization problem

The approximation stage takes the specification and gives a transfer function through four steps. They are as follows:

(i) A desired or ideal response is chosen, usually in the frequency domain.

(ii) An allowed class of filters is chosen (e.g. the length N for a FIR filters).

(iii) A measure of the quality of approximation is chosen.

(iv) A method or algorithm is selected to find the best filter transfer function.

The realization part deals with choosing the structure to implement the transfer function which may be in the form of circuit diagram or in the form of a program.

There are essentially three well-known methods for FIR filter design namely:

(1) The windowing method

(2) The frequency sampling technique

(3) Optimal filter design methods

1.1.1. The Windowing Method

The windowing method filter design process is based upon Fourier series. It is possible to represent a frequency function as a Fourier series, whose coefficients represent the coefficients of the filter.

To form a casual filter, the Fourier series is truncated and shifted. Truncating the Fourier series causes a phenomenon called the "Gibbs effect" - a spike occurs wherever there is a discontinuity in the desired magnitude of the filter. To counteract this, the filter coefficients are convolved in the frequency domain with the spectrum of a window function, thus smoothing the edge transitions at any discontinuity. This convolution in the frequency domain is equivalent to multiplying the filter coefficients with the window coefficients in the time domain.

The window design method starts with a very long series, in theory infinite, that is truncated to the desired length. Coefficients beyond the truncation are simply ignored. The window removes even more information. The equiripple method optimizes the series for a given number of coefficients.

The window method for digital filter design is fast, convenient, and robust, but generally suboptimal.

The window method consists of simply ``windowing" a theoretically ideal filter impulse response h(n) by some suitably chosen window function w(n), yielding

$$h_w(n)=w(n).h(n),\ n \in Z,$$

For example, the impulse response of the ideal low pass filter is the well known sinc function

$$H(n) = B.\text{sinc}(Bn) \overset{\Delta}{=} \frac{B\sin(\pi Bn)}{\pi Bn},\ n \in Z,$$

where $B=2f_c$ is the total normalized bandwidth of the low pass filter in Hz (counting both negative and positive frequencies), and f_c denotes the cut-off frequency in Hz. We cannot implement this filter in practice because it is noncausal and of infinite duration.

Since h(n) = sinc (BnT) decays away from time 0 as 1/n, we would expect to be able to truncate it to the interval [-N,N], for some sufficiently large N, and obtain a pretty good FIR filter which approximates the ideal filter. This would be an example of using the window method with the rectangular window. We saw that such a choice is optimal in the least-squares sense, but it designs relatively poor audio filters. Choosing other windows corresponds to tapering the ideal impulse response to zero instead of truncating it. Tapering better preserves the shape of the desired frequency response, as we will see. By choosing the window carefully, we can manage various trade-offs so as to maximize the filter-design quality in a given application.

Window functions are always time limited. This means there is always a finite integer Nw such that w(n)=0 for all |n| > Nw. The final windowed impulse response $h_w(n)=w(n).h(n)$, is thus always time-limited, as needed for practical implementation. The window method always designs a finite-impulse-response (FIR) digital filter (as opposed to an infinite-impulse-response (IIR) digital filter).

The major advantage of using window method is their relative simplicity as compared to other methods and ease of use. The fact that well defined equations are often available for calculating the window coefficients has made this method successful.

However, there are following **problems** in filter design using window method:
(i) This method is applicable only if $H_d(w)$ is absolutely integrable. When $H_d(w)$ is complicated or cannot easily be put into a closed form mathematical expression, evaluation of $h_d(n)$ becomes difficult.
(ii) The use of windows offers very little design flexibility e.g. in low pass filter design, the pass band edge frequency generally cannot be specified exactly since the window smears the discontinuity in frequency. Thus the ideal LPF with cut-off frequency f_c, is smeared by the window to give a frequency response with pass band response with pass band cutoff frequency f_l and stop band cut-off frequency f_h.

(iii) Window method is basically useful for design of prototype filters like low pass, high pass, band pass etc. This makes its use in speech and image processing applications very limited.

Gibbs phenomenon

In mathematics, the Gibbs phenomenon appears whenever the Fourier series – a series of continuous functions – is used to approximate a discontinuous continuously differentiable function. At the points of discontinuity, the partial Fourier series, rather than approximating precisely the discontinuous function shows ripples. These ripples typically increase in number and frequency as the approximation improves, decrease in energy / RMS amplitude, and do not die out but settle to a fixed height.
In digital signal processing, since the desired magnitude response of finite impulse response filters is usually a discontinuous function, the actual magnitude response of the filters produces ripples at the point of discontinuity, which increase in number and frequency as the length of the filter increases, decrease in RMS amplitude, but do not die out and settle to a fixed height of, as shown below, approximately 0.08949 normalized amplitude. These ripples are a manifestation of the Gibbs phenomenon.

Manifestation of the Gibbs phenomenon with finite impulse response filters

The desired magnitude response of a low pass filter is discontinuous, as an ideal magnitude response over the frequency spectrum should return a normalized amplitude of 1 (the original amplitude of the signal) up to the cutoff frequency and an amplitude of 0 afterwards. The following is the magnitude response of an ideal low pass filter.

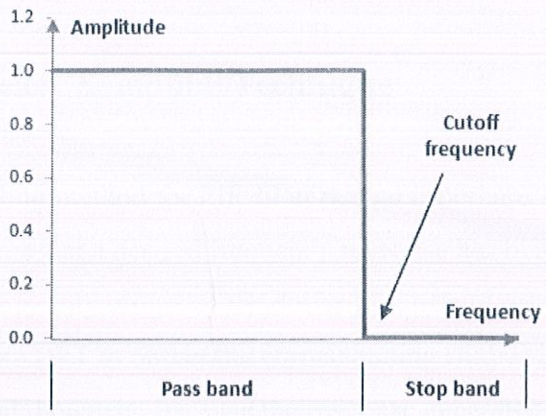


Fig. 1.1

The following picture shows the typical magnitude response of a digital finite impulse response low pass filter.

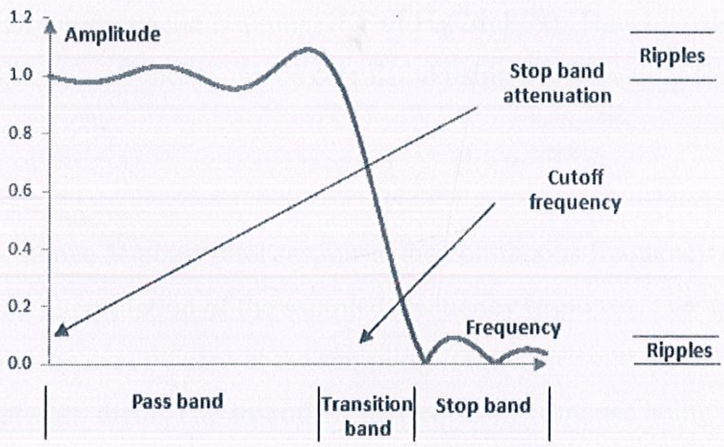


Fig. 1.2

The magnitude response of a typical finite impulse response low pass filter is in fact a Fourier series approximation of the desired magnitude response. Two derivations of the formulae for such low pass filters are shown in the topic Low pass filter. The ripples in the pass band and in the stop band in the actual magnitude response above are a manifestation of the Gibbs phenomenon.

1.1.2. The Frequency Sampling Technique

The frequency-sampling method for FIR filter design is perhaps the most direct technique imaginable when a desired frequency response has been specified. It consists simply of uniformly sampling the desired frequency response, and performing the inverse DFT to obtain the corresponding (finite) impulse response The results are not optimal, however, because the response generally deviates from what is desired between the samples.

In this method, the desired frequency response is provided as in the previous method. Now the given frequency response is sampled at a set of equally spaced frequencies to obtain N samples. Thus, sampling the continuous frequency response $H_d(w)$ at N points essentially gives us the N-point DFT of $H_d(2\pi nk/N)$. Thus by using the IDFT formula, the filter co-efficients can be calculated using the following formula

$$h(n) = \frac{1}{N} \sum_{n=0}^{N-1} H(k)e^{j(2\pi n/N)k}$$

Now using the above N-point filter response, the continuous frequency response is calculated as an interpolation of the sampled frequency response. The approximation error would then be exactly zero at the sampling frequencies and would be finite in frequencies between them. The smoother the frequency response being approximated, the smaller will be the error of interpolation between the sample points.
One way to reduce the error is to increase the number of frequency samples .The other way to improve the quality of approximation is to make a number of frequency samples specified as unconstrained variables. The values of these unconstrained variables are generally optimized by computer to minimize some simple function of the approximation error e.g. one might choose as unconstrained variables the frequency samples that lie in a transition band between two frequency bands in which

the frequency response is specified e.g. in the band between the pass band and the stop band of a low pass filter.

There are two different set of frequencies that can be used for taking the samples. One set of frequency samples are at $f_k = k/N$ where $k = 0,1,\ldots.N-1$. The other set of uniformly spaced frequency samples can be taken at $f_k = (k + ½)/N$ for $k = 0,1,\ldots.N-1$.The second set gives us the additional flexibility to specify the desired frequency response at a second possible set of frequencies. Thus a given band edge frequency may be closer to type-II frequency sampling point than to type-I in which case a type-II design would be used in optimization procedure.

The steps involved in this method are as follows:

(i) The desired magnitude response is provided along with the number of samples N. Given N, the designer determines how fine an interpolation will be used.

(ii) It was found by Rabiner that for designs they investigated, where N varied from 15 to 256, 16N samples of H(w) lead to reliable computations, so 16 to 1 interpolation was used.

(iii) Given N values of H(k) , the unit sample response of filter to be designed, h(n) is calculated using the inverse FFT algorithm.

(iv) In order to obtain values of the interpolated frequency response two procedures were suggested by Rabiner. They are

(a) h(n) is rotated by N/2 samples(N even) or (N-1)/2 samples for N odd to remove the sharp edges of impulse response, and then 15N zero-valued samples are symmetrically placed around the impulse response.

(b) h(n) is split around the N/2nd sample, and 15N zero-valued samples are placed between the two pieces of the impulse response.

(v) The zero augmented sequences are transformed using the FFT algorithm to give the interpolated frequency responses.

Merits of frequency sampling technique:

(i) Unlike the window method, this technique can be used for any given magnitude response.

(ii) This method is useful for the design of non-prototype filters where the desired magnitude response can take any irregular shape.

Demerits:

There are some disadvantages with this method i.e the frequency response obtained by interpolation is equal to the desired frequency response only at the sampled points. At the other points, there will be a finite error present.

1.1.3. Optimal Filter Design Methods

Many methods are present under this category. The basic idea in each method is to design the filter coefficients again and again until a particular error is minimized. The various methods are as follows:

(i) Least squared error frequency domain design

(ii) Weighted Chebyshev approximation

(iii) Nonlinear equation solution for maximal ripple FIR filters

(iv) Polynomial interpolation solution for maximal ripple FIR filters

As discussed, every method has its own advantages and disadvantages and is selected depending on the type of filter to be designed. The window method is basically used for the design of prototype filters like the Low-pass, high-pass, band-pass etc. They are not very suitable for designing of filters with any given frequency response. On the other hand, the frequency sampling technique is suitable for designing of filters with a given magnitude response. The ideal frequency response of the filter is approximated by placing appropriate frequency samples in the *z*- plane and then calculating the filter co-efficients using the IFFT algorithm. The disadvantage of the frequency sampling technique was that the frequency response gave errors at the points where it was not sampled. In order to reduce these errors the different optimization technique for FIR filter design were presented wherein the remaining frequency samples are chosen to satisfy an optimization criterion. But again, optimization methods are a tedious to achieve.

And hence, in an attempt to overcome some of these shortcomings, we are trying to optimize this new pole-zero placement technique.

Chapter 2: FILTERS – A REVIEW

2.1. What is a filter? What does a filter do?

A filter is an electrical network which alters the amplitude and/or phase characteristics of a signal with respect to frequency. Filters are often used in electronic systems to emphasize signals in certain frequency ranges and reject signals in other frequency ranges in order to eliminate the undesired signal. Therefore, any operation which can be used to reduce or remove noise is called a filter. Such a filter has a "gain" associated with it that is dependent on the signal frequency.

2.2. The Characteristics of a Filter

We have several traits to describe a filter:

1. **Transfer function (or network function)**:

   This is the ratio of the Laplace/ Z transforms of its output and input signals. If we delineate a filter with Z transform, then the transfer function H (z) can therefore be written as:

   $$H(z) = \frac{Y(z)}{X(z)}$$

   where X (z) and Y (z) are the Z transform of the input and output signal and z is the complex frequency variable.

2. **Amplitude response (filter gain)** :

   The transfer function magnitude versus frequency, i.e. the absolute value of

   $$H(z) = \frac{|Y(z)|}{|X(z)|}$$

   Knowing the transfer function magnitude (or gain) at each frequency allows us determine how well the filter can distinguish between signals at different frequencies.

3. **Phase response** :

The phase shift of the transfer functions versus frequency.

$$H(z) = \frac{\arg Y(z)}{\arg X(z)}$$

A change in the phase of a signal also represents a change in time.

4. **Filter order (or filter length)** :

It can be defined as the number of previous inputs (stored in the processor's memory) used to calculate the current output. In circuit theory, it means the total number of capacitors and inductors in the circuit. If we see the transfer function, then the order of the filter is the highest power of the variable z in its transfer function. Higher order filters will be more expensive than lower order filters since they use more components (capacitors and inductors) and definitely hard to be designed. However, high order filters are able to discriminate signals with different frequencies more effectively.

5. **The attenuation slope (or roll-off slope)** :

The rate of change of attenuation between the pass band and the stop band. It is usually expressed in dB/octave (an octave is a factor of 2 in frequency) or dB/decade (a decade is a factor of 10 in frequency).

6. **-3 dB frequencies (or cutoff frequencies)** :

The standard reference points for the roll-offs on each side of the pass band where the amplitude (gain) has decreased by 3 dB (to 22 or 0.707 of its maximum amplitude)

7. **Center frequency** :

The frequency corresponding to the peak value of the amplitude response of a filter.

For the band pass or band stop filter, it is equal to the geometric mean of the -3 dB frequencies:

$$f_c = \sqrt{f_h}\, f_l$$

where f_c is the center frequency, f_l is the lower -3 dB frequency, f_h is the higher -3 dB frequency.

8. **-3dB Bandwidth** :

It is a frequency band which is calculated by the higher -3 dB frequency (roll-off

point) minus the lower -3 dB frequency (roll-off point).

$$f_c = f_h - f_l$$

9. **Ripple** :

it is an amplitude (gain) variation in the pass band or stop band for a filter. For an ideal filter, it has absolutely constant gain within the pass band, zero gain in the stop band, and an abrupt boundary between the two. Unfortunately, this response characteristic is impossible

to implement in practice but it can be approximated to varying degrees of accuracy by real filter. Therefore, some ripples may occur.

10. **Quality factor (Q factor)** :

For a filter, this is a measure of the "sharpness" of the amplitude response. The Q of band pass or band stop filter is the ratio of the center frequency (f_c) to the -3 dB bandwidth ($f_h - f_l$):

$$Q = \frac{f_c}{f_h - f_l}$$

The higher the Q, the shaper the peak is. Fig. shows amplitude response curves for various Q factors.

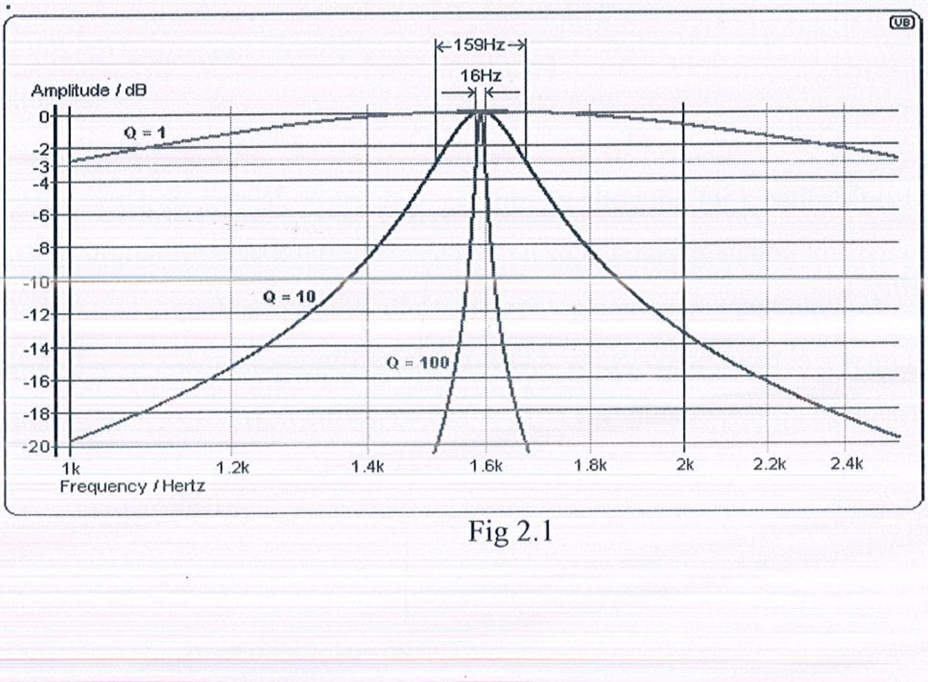


Fig 2.1

2.3. The Types of Filter

Analog and Digital filters

Digital filters are not subject to the component non-linearities that greatly complicate the design of analog filters. Analog filters consist of imperfect electronic components, whose values are specified to a limit tolerance (e.g. resistor values often have a tolerance of +/- 5%) and which may also change with temperature and drift with time. As the order of an analog filter increases, and thus its component count, the effect of variable component errors is greatly magnified. In digital filters, the coefficient values are stored in computer memory, making them far more stable and predictable.

Because the coefficients of digital filters are definite, they can be used to achieve much more complex and selective designs – specifically with digital filters, one can achieve a lower passband ripple, faster transition, and higher stopband attenuation than is practical with analog filters. Even if the design could be achieved using analog filters, the engineering cost of designing an equivalent digital filter would likely be much lower. Furthermore, one can readily modify the coefficients of a digital filter to make an adaptive filter or a user-controllable parametric filter. While these techniques are possible in an analog filter, they are again considerably more difficult.

Digital filters can be used in the design of finite impulse response filters. Analog filters do not have the same capability, because finite impulse response filters require delay elements.

Digital filters rely less on analog circuitry, potentially allowing for a better signal-to-noise ratio. A digital filter will introduce noise to a signal during analog low pass filtering, analog to digital conversion, digital to analog conversion and may introduce digital noise due to quantization. With analog filters, every component is a source of thermal noise (such as Johnson noise), so as the filter complexity grows, so does the noise.

igital filters do introduce a higher fundamental latency to the system. In an r, latency is often negligible; strictly speaking it is the time for an electrical opagate through the filter circuit. In digital filters, latency is a function of of delay elements in the system.

rs also tend to be more limited in bandwidth than analog filters. High digital filters require expensive ADC/DACs and fast computer hardware ing.

ple cases, it is more cost effective to use an analog filter. Introducing a requires considerable overhead circuitry, as previously discussed, wo low pass analog filters.

rization of digital filters

lter is characterized by its transfer function, or equivalently, its difference Mathematical analysis of the transfer function can describe how it will any input. As such, designing a filter consists of developing specifications to the problem (for example, a second-order low pass filter with a specific quency), and then producing a transfer function which meets the ons.

r function for a linear, time-invariant, digital filter can be expressed as a nction in the Z-domain; if it is causal, then it has the form:

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_N z^{-N}}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_M z^{-M}}$$

order of the filter is the greater of N or M. See Z-transform's LCCD or further discussion of this transfer function.

form for a recursive filter with both the inputs (Numerator) and outputs tor), which typically leads to an IIR infinite impulse response behaviour, enominator is made equal to unity i.e. no feedback, then this becomes an te impulse response filter.

On the basis of the length of impulse response

- Infinite Impulse Response (IIR) Filters : **Infinite impulse response (IIR)** is a property of signal processing systems. Systems with this property are known as *IIR systems* or, when dealing with filter systems, as *IIR filters*. IIR systems have an impulse response function that is non-zero over an infinite length of time. This is in contrast to finite impulse response (FIR) filters, which have fixed-duration impulse responses. The simplest analog IIR filter is an RC filter made up of a single resistor (R) feeding into a node shared with a single capacitor (C). This filter has an exponential impulse response characterized by an RC time constant. Because the exponential function is asymptotic to a limit, and thus never settles at a fixed value, the response is considered infinite.

- Finite Impulse Response (FIR) Filters : In signal processing, a **finite impulse response (FIR)** filter is a filter whose impulse response (or response to any finite length input) is of *finite* duration, because it settles to zero in finite time. This is in contrast to infinite impulse response (IIR) filters, which have internal feedback and may continue to respond indefinitely (usually decaying). The impulse response of an Nth-order discrete-time FIR filter (i.e., Kronecker delta impulse input) lasts for N + 1 samples, and then settles to zero. FIR filters can be discrete-time or continuous-time, and digital or analog.

Comparison of FIR and IIR Filters:

FIR FILTERS	IIR FILTERS
Linear Phase response	Non-linear phase response
Better delay characteristics	Difficult to implement-delay & distort adjustments
Require more memory	Require less memory
Non- recursive.Hence,only zeros are involved.	Recursive- consists of poles & zeros both.
Stable	Unstable & difficult to implement

Table 2.1.

<u>On the basis of frequency selection</u>

- **Low pass filters** :

A **low-pass filter** is an electronic filter that passes low-frequency signals but attenuates (reduces the amplitude of) signals with frequencies higher than the cutoff frequency. The actual amount of attenuation for each frequency varies from filter to filter. It is sometimes called a **high-cut filter**, or **treble cut filter** when used in audio applications. Low-pass filters exist in many different forms, including electronic circuits (such as a *hiss filter* used in audio), anti-aliasing filters for conditioning signals prior to analog-to-digital conversion, digital filters for smoothing sets of data, acoustic barriers, blurring of images, and so on. The moving average operation used in fields such as finance is a particular kind of low-pass filter, and can be analyzed with the same signal processing techniques as are used for other low-pass filters. Low-pass filters provide a smoother form of a signal, removing the short-term fluctuations, and leaving the longer-term trend.

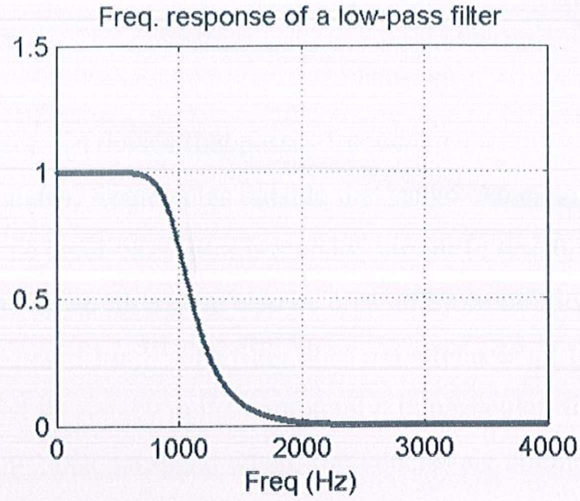


Fig.2.2.

- **High pass filters** :

A **high-pass filter** (HPF) is an electronic filter that passes high-frequency signals but attenuates (reduces the amplitude of) signals with frequencies lower than the cutoff frequency. The actual amount of attenuation for each frequency varies from filter to filter. A high-pass filter is usually modeled as a linear time-invariant system. It is sometimes called a **low-cut filter** or **bass-cut filter**.[1] High-pass filters have many uses, such as blocking DC from circuitry sensitive to non-zero average voltages or RF devices. They can also be used in conjunction with a low-pass filter to make a bandpass filter.

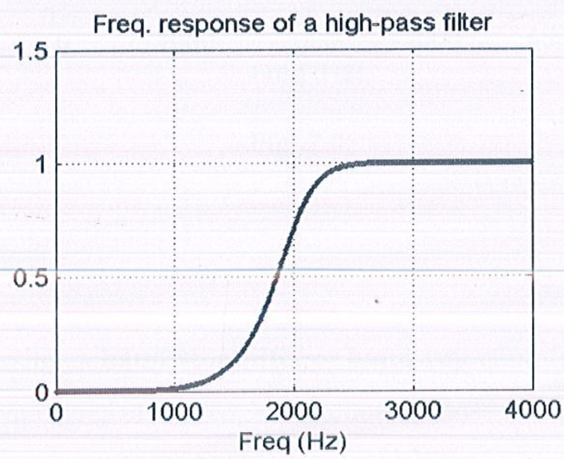


Fig. 2.3.

- **Band pass filters** :

A **band-pass filter** is a device that passes frequencies within a certain range and rejects (attenuates) frequencies outside that range. An example of an analogue electronic band-pass filter is an RLC circuit (a resistor–inductor–capacitor circuit). These filters can also be created by combining a low-pass filter with a high-pass filter.[1] The filter does not attenuate all frequencies outside the desired frequency range completely; in particular, there is a region just outside the intended passband where frequencies are attenuated, but not rejected. This is known as the filter roll-off, and it is usually expressed in dB of attenuation per octave or decade of frequency. Generally, the design of a filter seeks to make the roll-off as narrow as possible, thus allowing the filter to perform as close as possible to its intended design. Often, this is achieved at the expense of pass-band or stop-band *ripple*.

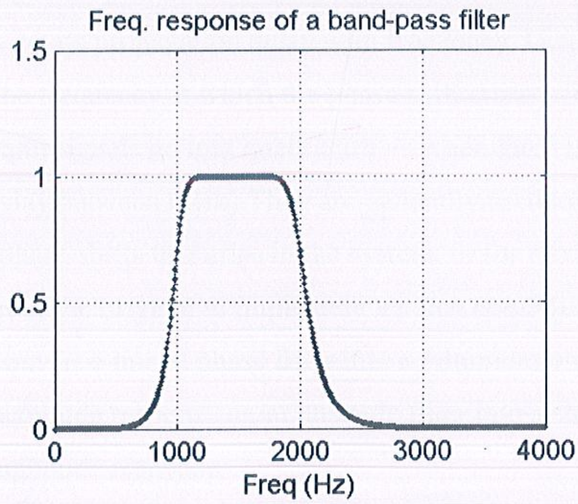


Fig. 2.4.

- **Band reject filters** :

In signal processing, a **band-stop filter** or **band-rejection filter** is a filter that passes most frequencies unaltered, but attenuates those in a specific range to very low levels. It is the opposite of a band-pass filter. A **notch filter** is a band-stop filter with a narrow stopband (high Q factor).

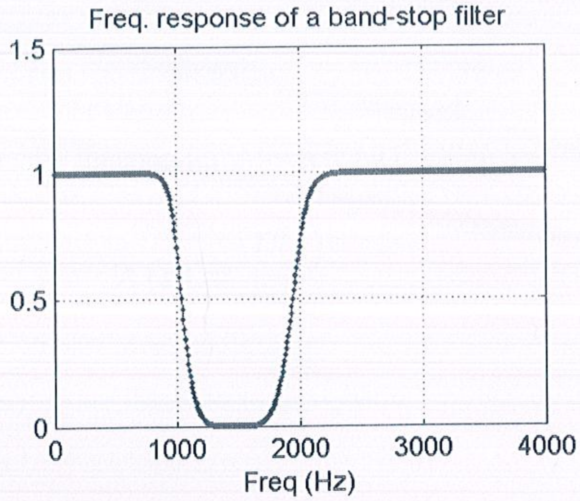


Fig. 2.5.

- **All pass filters** :

An **all-pass filter** is a signal processing filter that passes all frequencies equally, but changes the phase relationship between various frequencies. It does this by varying its propagation delay with frequency. Generally, the filter is described by the frequency at which the phase shift crosses 90° (i.e., when the input and output signals go into quadrature — when there is a quarter wavelength of delay between them).They are generally used to compensate for other undesired phase shifts that arise in the system, or for mixing with an unshifted version of the original to implement a notch comb filter.They may also be used to convert a mixed phase filter into a minimum phase filter with an equivalent magnitude response or an unstable filter into a stable filter with an equivalent magnitude response.

Pole–zero plot

In mathematics, signal processing and control theory, a pole–zero plot is a graphical representation of a rational transfer function in the complex plane which helps to convey certain properties of the system such as:

- Stability
- Causal system / anticausal system

- Region of convergence (ROC)
- Minimum phase / non minimum phase

In general, a rational transfer function for a discrete LTI system has the form:

$$X(z) = \frac{P(z)}{Q(z)}$$

- z_i such that $P(z_i) = 0$ are the zeros of the system
- z_j such that $Q(z_j) = 0$ are the poles of the system

2.4. Software used: MATLAB

MATLAB (matrix laboratory) is a numerical computing environment and fourth-generation programming language. Developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, Java, and Fortran.
In 2004, MATLAB had around one million users across industry and academia. MATLAB users come from various backgrounds of engineering, science, and economics. MATLAB is widely used in academic and research institutions as well as industrial enterprises.
The MATLAB application is built around the MATLAB language, and most use of MATLAB involves typing MATLAB code into the Command Window (as an interactive mathematical shell), or executing text files containing MATLAB code and functions.

SPTool Overview: An Interactive Signal Processing Environment

Syntax: sptool

SPTool is an interactive GUI for digital signal processing used to

- Analyze signals

- Design filters
- Analyze (view) filters
- Filter signals
- Analyze signal spectra

We can accomplish these tasks using four GUIs that you access from within SPTool: The Signal Browser is for analyzing signals. We can also play portions of signals using your computer's audio hardware.

The FDATool: A Filter Design and Analysis GUI is available for designing or editing FIR and IIR digital filters. Most Signal Processing Toolbox filter design methods available at the command line are also available in FDATool. Additionally, We can use FDATool to design a filter by using the Pole/Zero Editor to graphically place poles and zeros on the z-plane.

The Filter Visualization Tool (FVTool) is for analyzing filter characteristics. The Spectrum Viewer is for spectral analysis. We can use Signal Processing Toolbox spectral estimation methods to estimate the power spectral density of a signal.

Pole/Zero Editor We can use the Pole/Zero Editor to design arbitrary FIR and IIR filters by placing and moving poles and zeros on the complex z-plane.

Positioning Poles and Zeros

We can use mouse to move poles and zeros around the pole/zero plot and modify your filter design.

Icon	Description
	Enable moving poles or zeros by dragging on the plot
	Add pole
	Add zero
	Erase poles or zeros

We can move both members of a conjugate pair simultaneously by manipulating just one of the poles or zeros.

To ungroup conjugates, select the desired pair and clear Conjugate pair in the Specifications region on the Filter Designer.

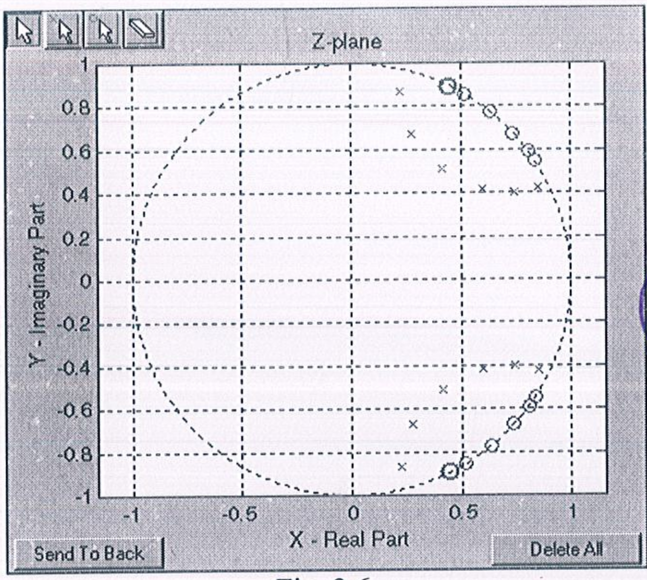


Fig. 2.6.

When we place two or more poles (or two or more zeros) directly on top of each other, a number is displayed next to the symbols (on the left for poles, and on the right for zeros) indicating the number of poles or zeros at that location (e.g., for three zeros). This number makes it easy to keep track of all the poles and zeros in the plot area, even when several are superimposed on each other and are not visually differentiable. Note, however, that this number does not indicate the multiplicity of the poles or zeros to which it is attached.

To detect whether or not a set of poles or zeros are truly multiples, use the zoom tools to magnify the region around the poles or zeros in question. Because numerical limitations usually prevent any set of poles or zeros from sharing exactly the same value, at a high enough zoom level even truly multiple poles or zeros appear distinct from each other.

A common way to assess whether a particular group of poles or zeros contains multiples is by comparing the mutual proximity of the group members against a selected threshold value. As an example, the residuez function defines a pole or zero as being a multiple of another pole or zero if the absolute distance separating them is less than 0.1% of the larger pole or zero's magnitude.

Chapter 3: OBSERVATIONS

As we already know, the pole angle dictates the filter frequency and the pole radius dictates the Q-factor. So, while working with various pole-zero plots we could observe that changing the pole angle and the radius affects the frequency response in different ways.

Observation 1:

Starting from the most basic pattern that can be observed, placing one pole near the unit circle, we observed a narrow peak. And then placing more poles on approximately the same radial distance we observed the widening of the peak due to the summation of the various narrow peaks.

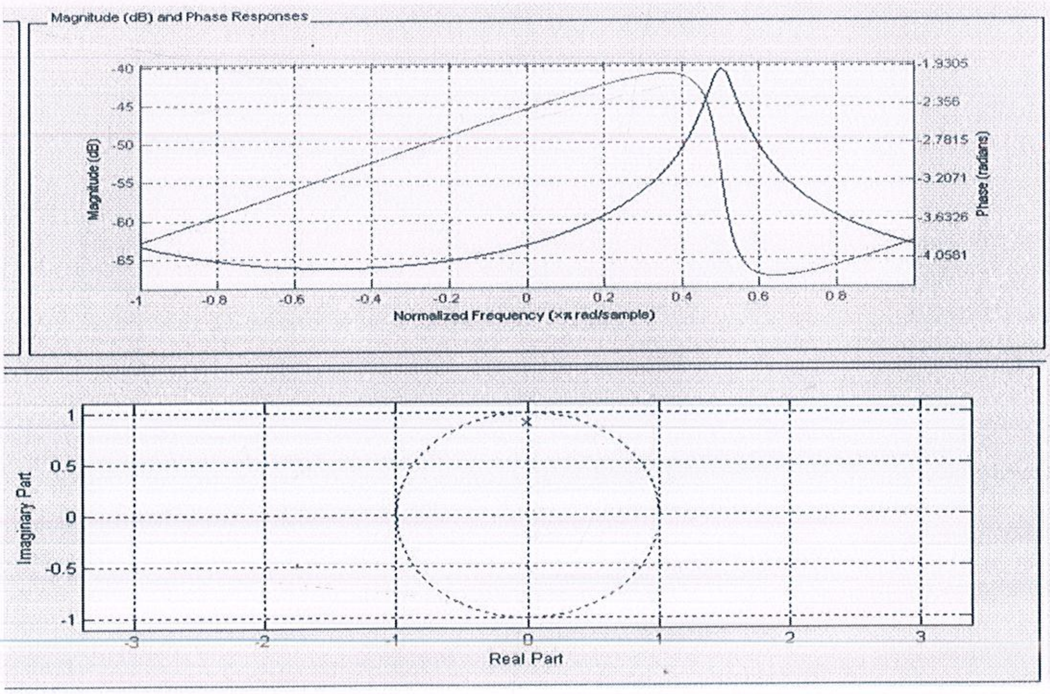


Fig. 3.1

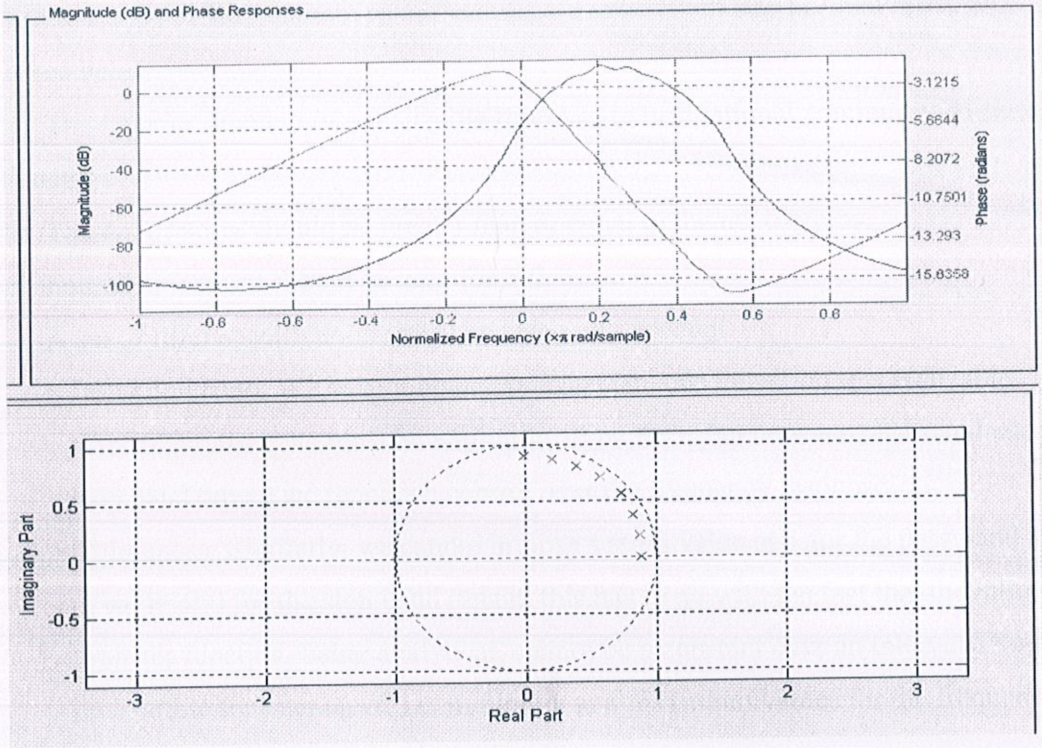

Magnitude (dB) and Phase Responses
Magnitude (dB)
Phase (radians)
Normalized Frequency (×π rad/sample)
Imaginary Part
Real Part

Fig. 3.2

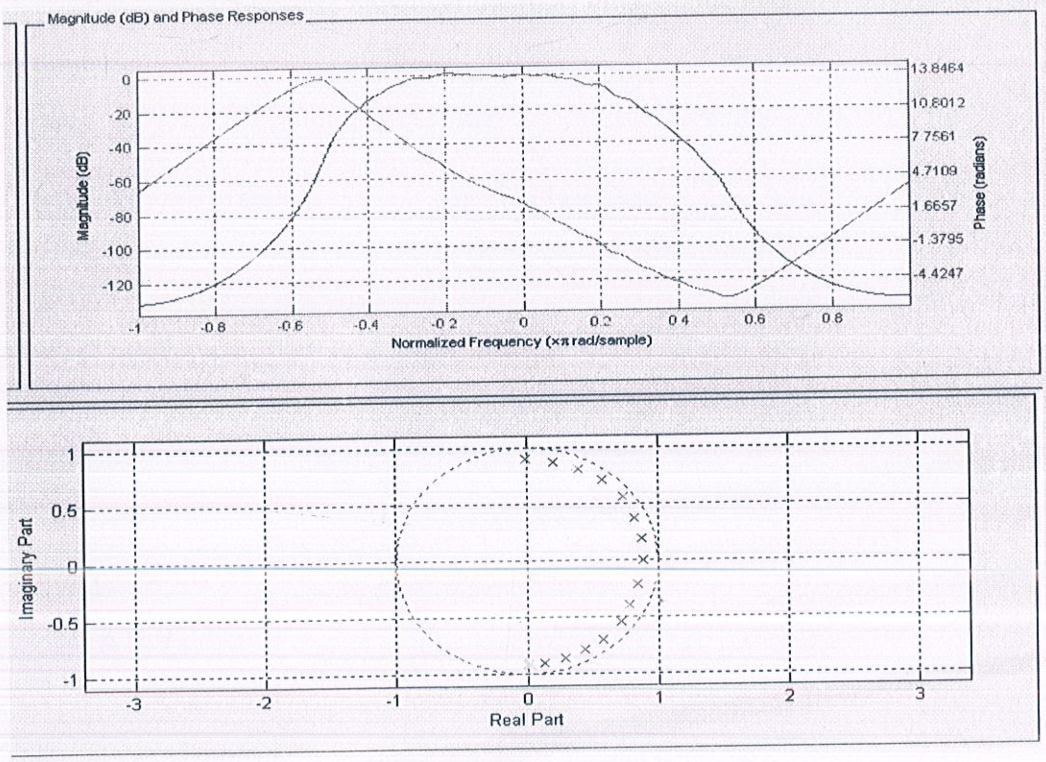

Magnitude (dB) and Phase Responses
Magnitude (dB)
Phase (radians)
Normalized Frequency (×π rad/sample)
Imaginary Part
Real Part

Fig. 3.3

But to realize a filter, certain specifications are to be taken into account. They are listed below:

1. The transfer function of a realizable filter has to be a rational continuous-in-time function.
2. The no. of poles should be greater than or equal to the no. of zeros.
3. Fourier transform should be zero for $t<0$.
4. A set of filter specifications must be taken into account:
   - Transition band: We cannot obtain an instantaneous transition in a realizable filter. Therefore, we must be willing to allow for a gap between pass band and stop band where we renounce control over the frequency response.
   - Tolerances: Similarly, we cannot impose a strict value of 1 for the pass band and a value of 0 for the stop band (again, this has to do with the fact that the rational transfer function, being analytical, cannot be a constant over an interval). So we must allow for tolerances, i.e. minimum and maximum values for the frequency response over pass band and stop band.
5. If we are dealing with real-valued filter coefficients, it is sufficient to specify the desired frequency response only over the $[0,\pi]$ interval, the magnitude response being symmetric.

Observation 2:

So, keeping these specifications in mind, we worked with realizable filters by changing the radial distance for conjugate pairs. First we placed a conjugate pair closer to the perimeter of the unit circle and observed a sharp peak but as we brought the pair closer to the centre the peak widened.

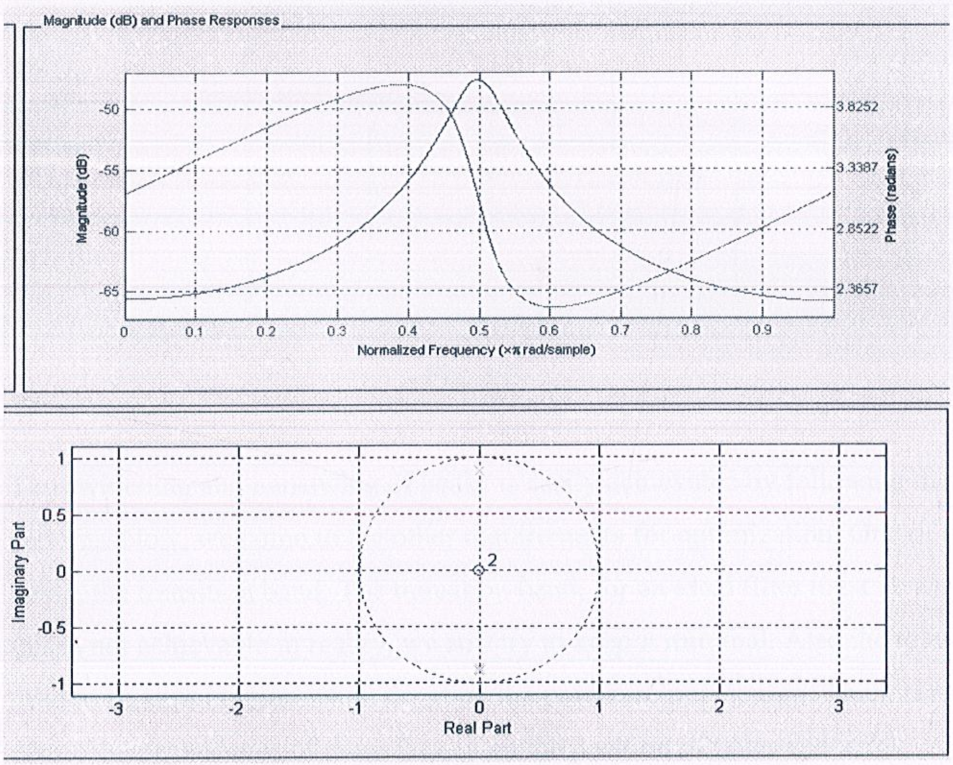


Fig. 3.4

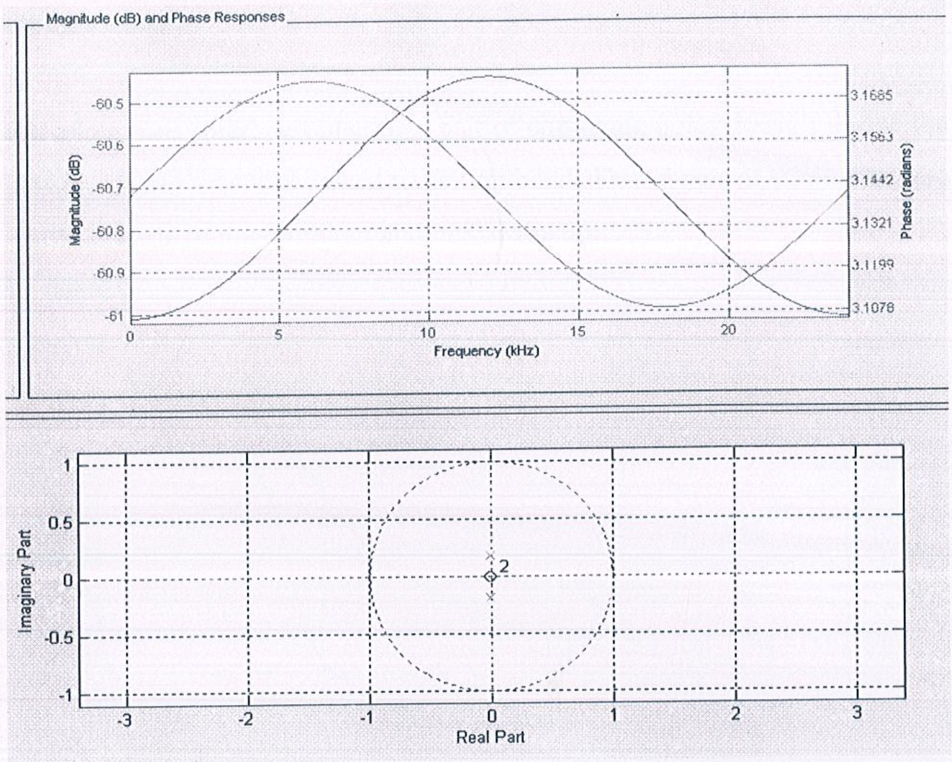


Fig. 3.5

Thus widening and narrowing of peaks is easily achievable by following these patterns. Now, we come to the other requirements for optimization. One of them being, the transition band. The transition band, for an ideal filter must be zero. Since, this is not achievable in reality, we still try to keep it minimal. Also the tolerances need to be kept in mind while deciding the pole-zero positioning. The following plots depict the dependence of these factors on the position of poles and zeros.

Observation 3:

For a low pass filter, just placing a pair of conjugate poles doesn't define the drop clearly, giving way to a spread transition band. Thus a pair of conjugate zeros work just fine to solve this problem and define the dip.

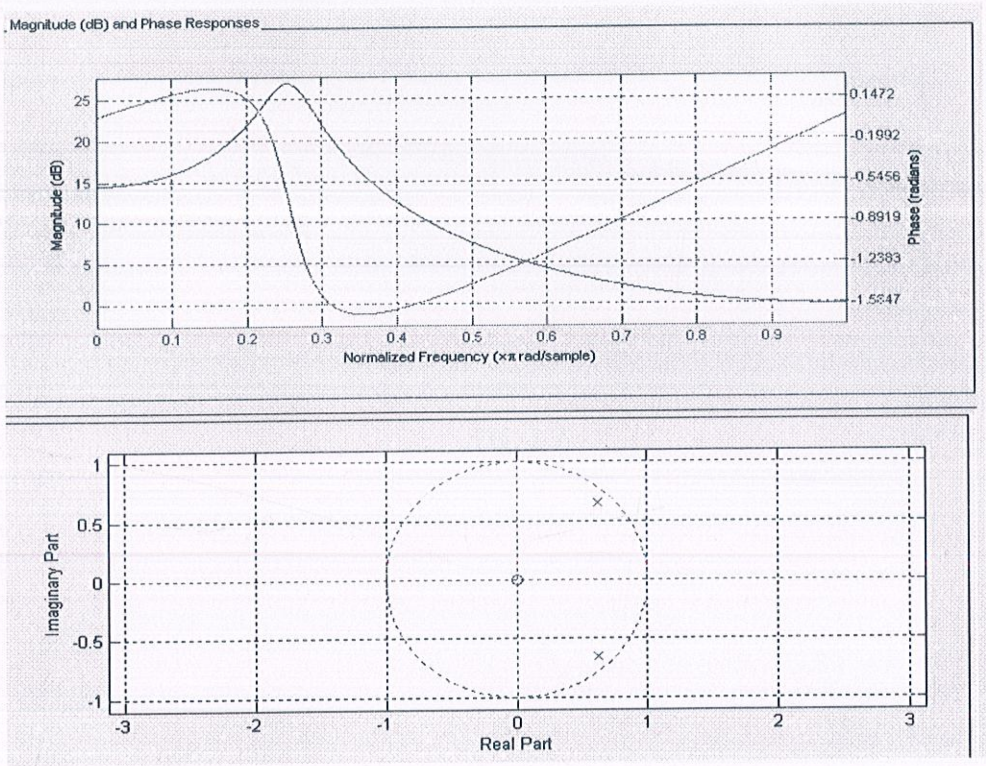
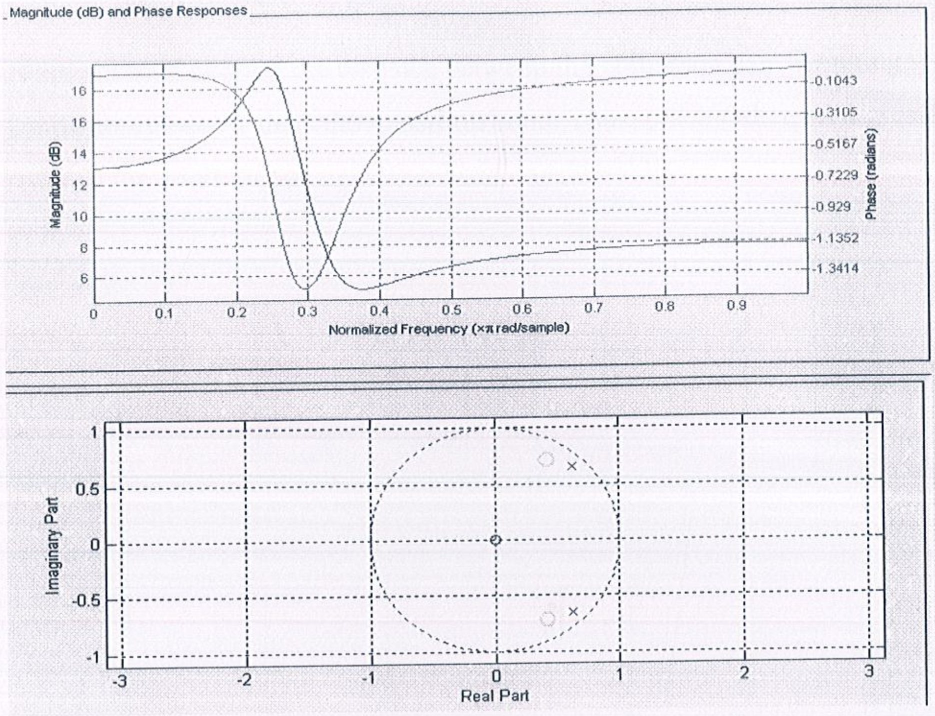


Fig. 3.6



Magnitude (dB) and Phase Responses
Magnitude (dB)
Phase (radians)
Normalized Frequency (×π rad/sample)
-0.1043
-0.3105
-0.5167
-0.7229
-0.929
-1.1352
-1.3414
Imaginary Part
Real Part

Fig. 3.7

Observation 4:

Next, we tried altering the distance between the conjugate pair. Bringing the two conjugates closer widens the transition band. Thus, placing them at a greater distance reduced the width of our transition band considerably.

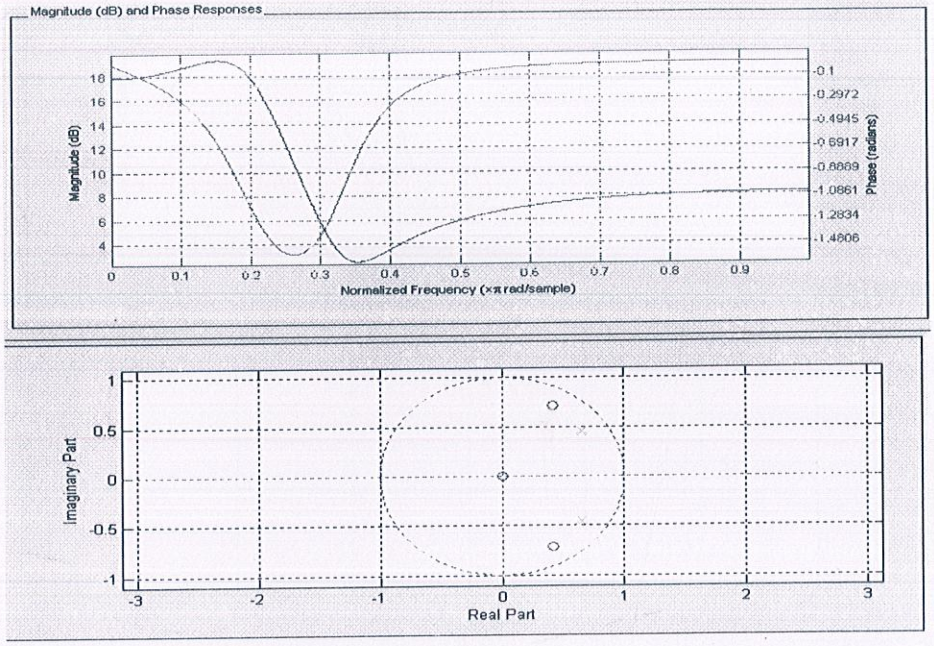


Fig. 3.8

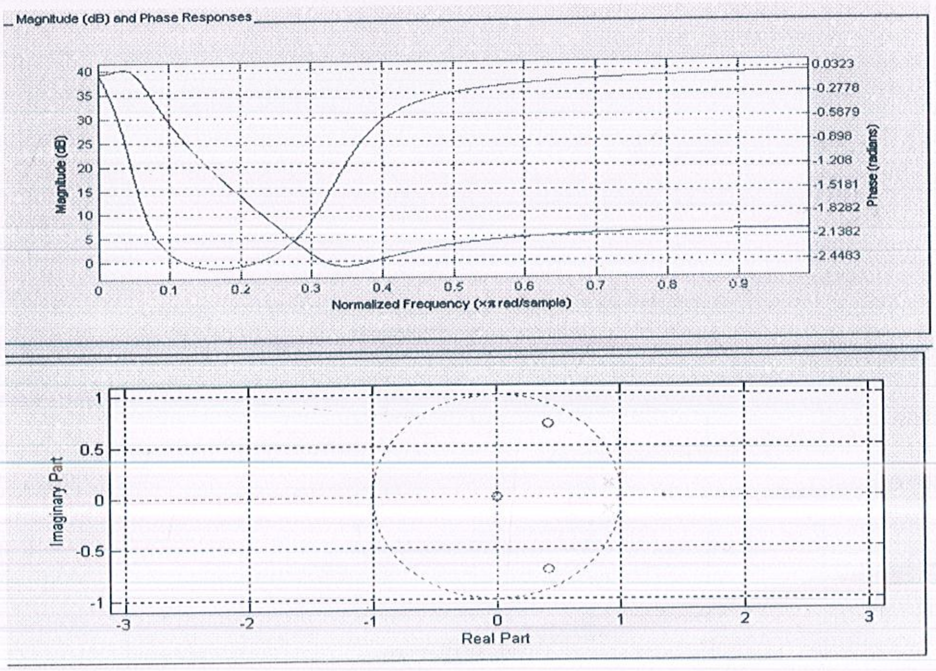


Fig. 3.9

Observation 5:

Coming to higher order filters, the proximity of conjugate pairs of poles to each other widens the response in the passband and stopband making a flat response achievable.

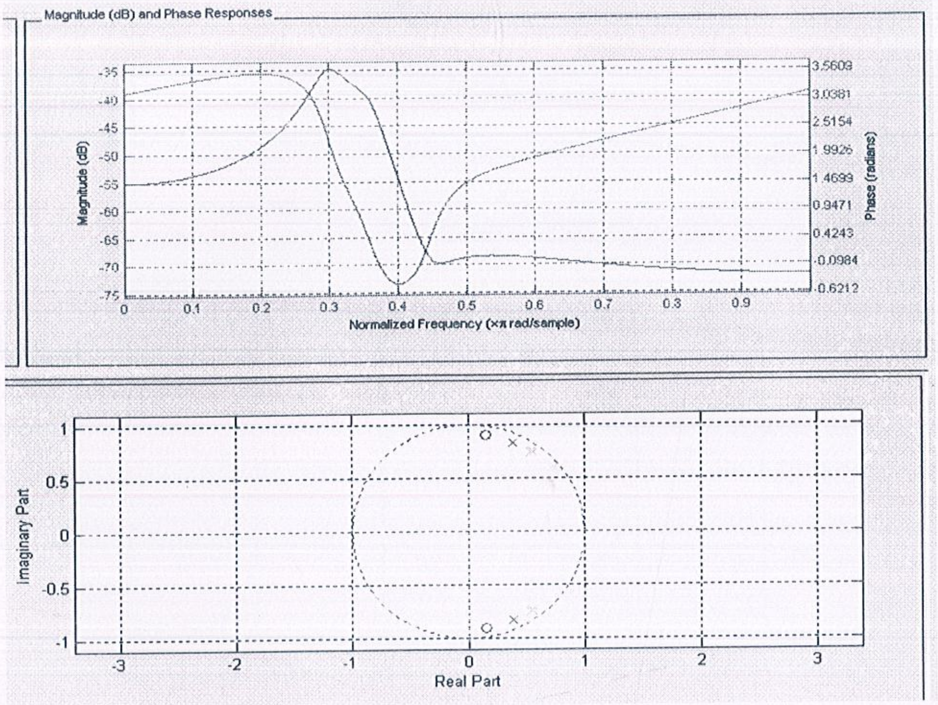


Fig. 3.10

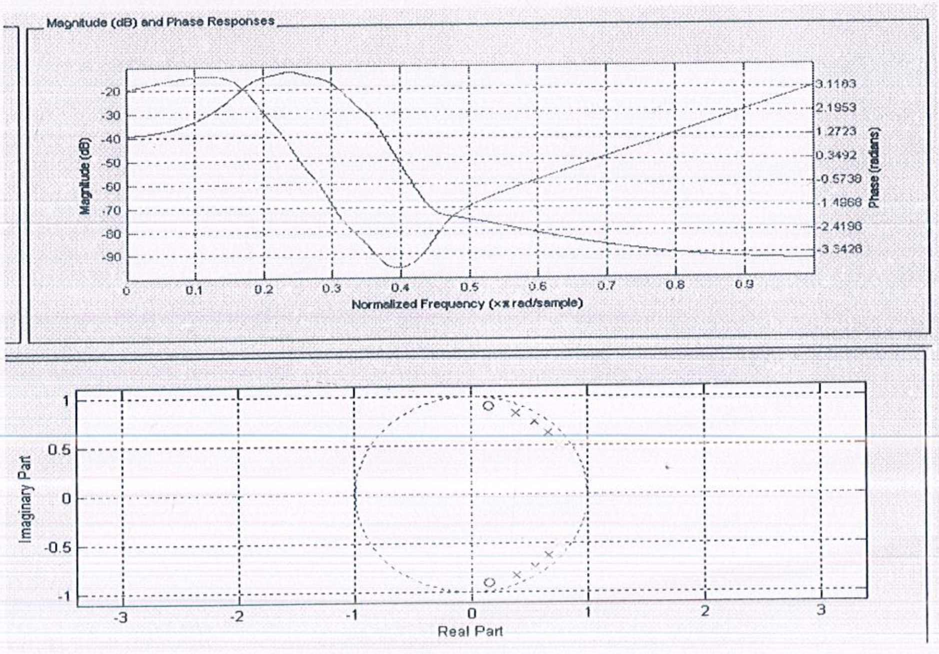


Fig. 3.11

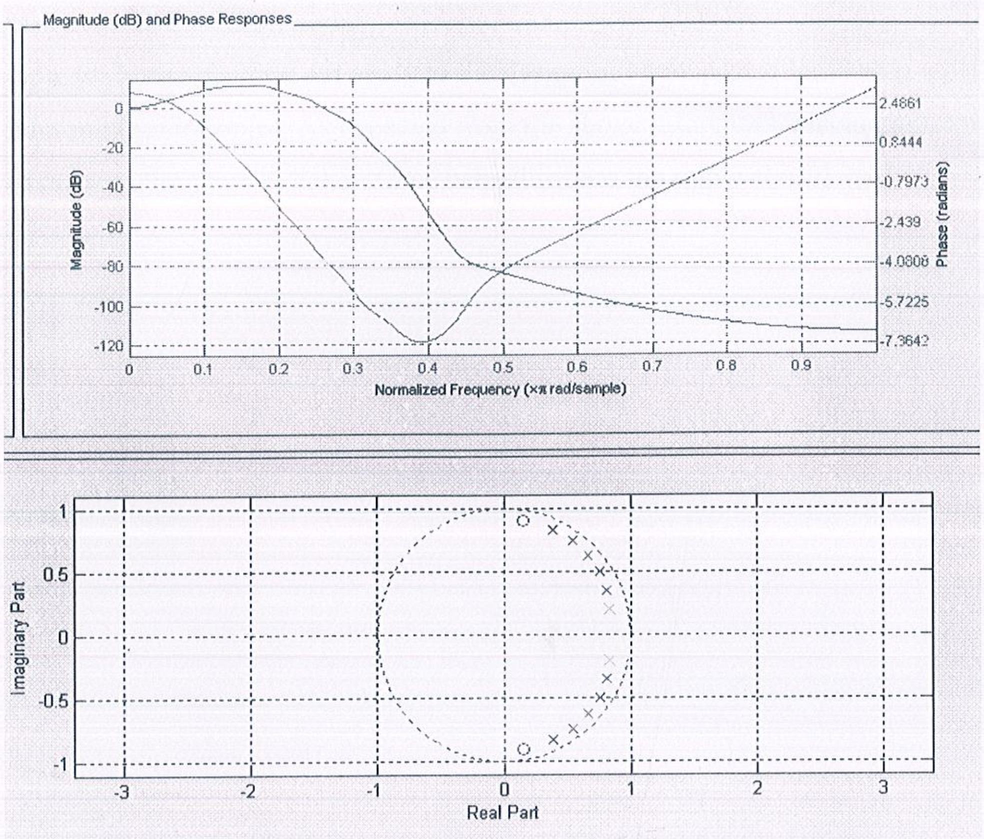

Magnitude (dB) and Phase Responses
Magnitude (dB)
Phase (radians)
0
-20
-40
-60
-80
-100
-120
2.4861
0.8444
-0.7973
-2.439
-4.0808
-5.7225
-7.3642
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Normalized Frequency (×π rad/sample)
Imaginary Part
1
0.5
0
-0.5
-1
-3
-2
-1
0
1
2
3
Real Part

Fig. 3.12

Observation 6:

Now we have specified our study to the design of a low pass filter having a passband with minimum ripples. As expected from the above inferences, placing more number of poles at the same radial distance helped reduce the ripple considerably.

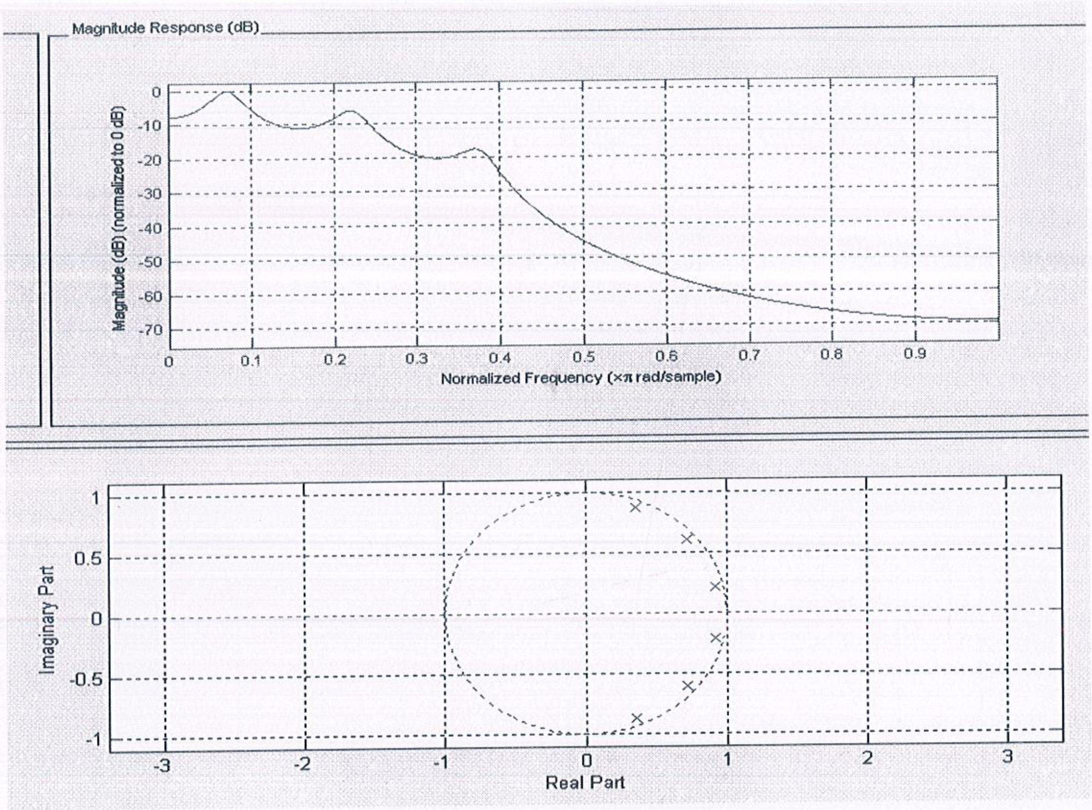


Fig. 3.13

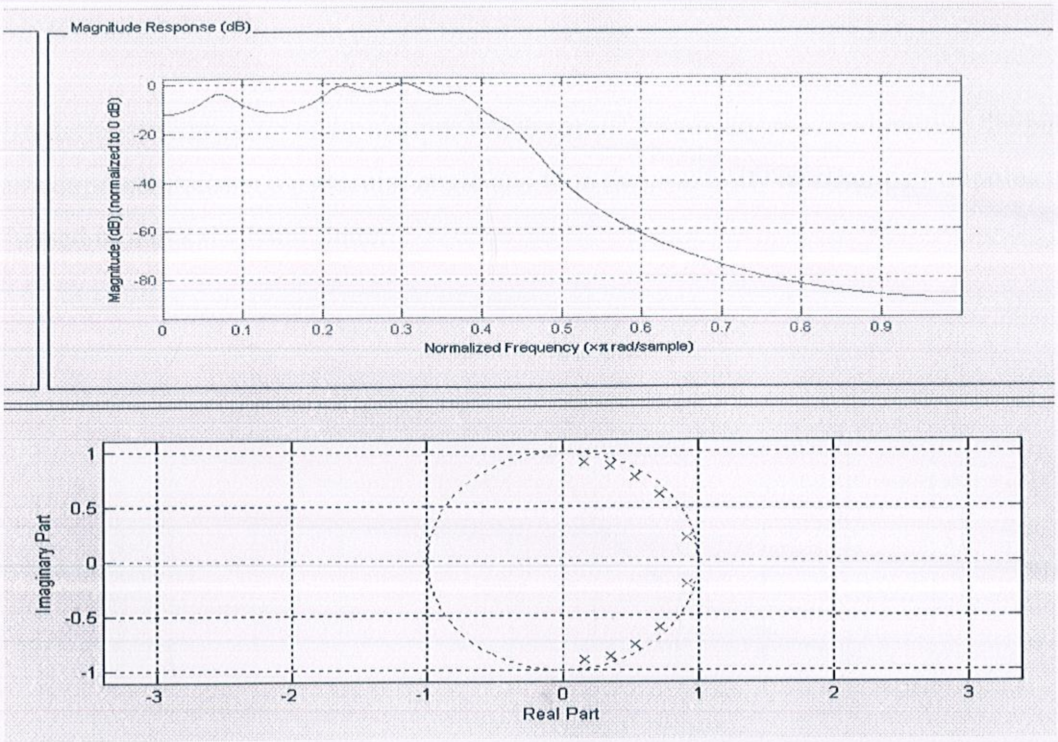


Fig. 3.14

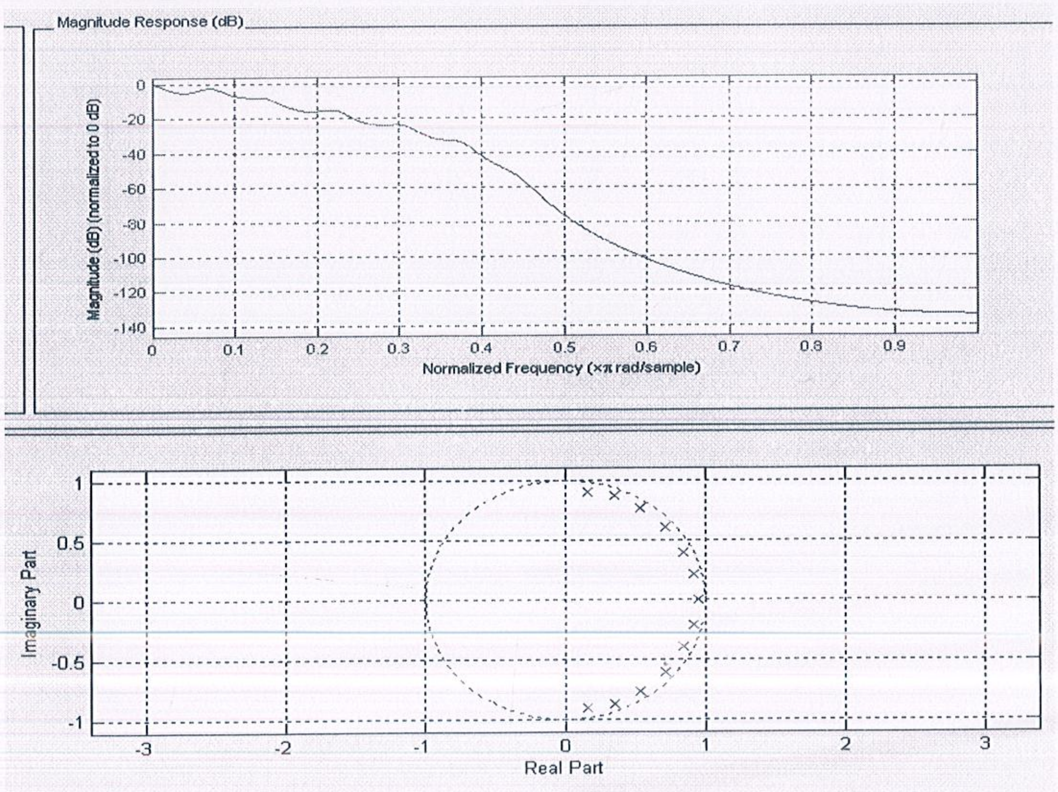


Fig. 3.15

Observation 7:

To define a sudden dip and a sharper lower cut-off frequency, we tried putting the pair of conjugate poles at a particular angle and then the proximity of other pair of poles helped to make the band flatter.

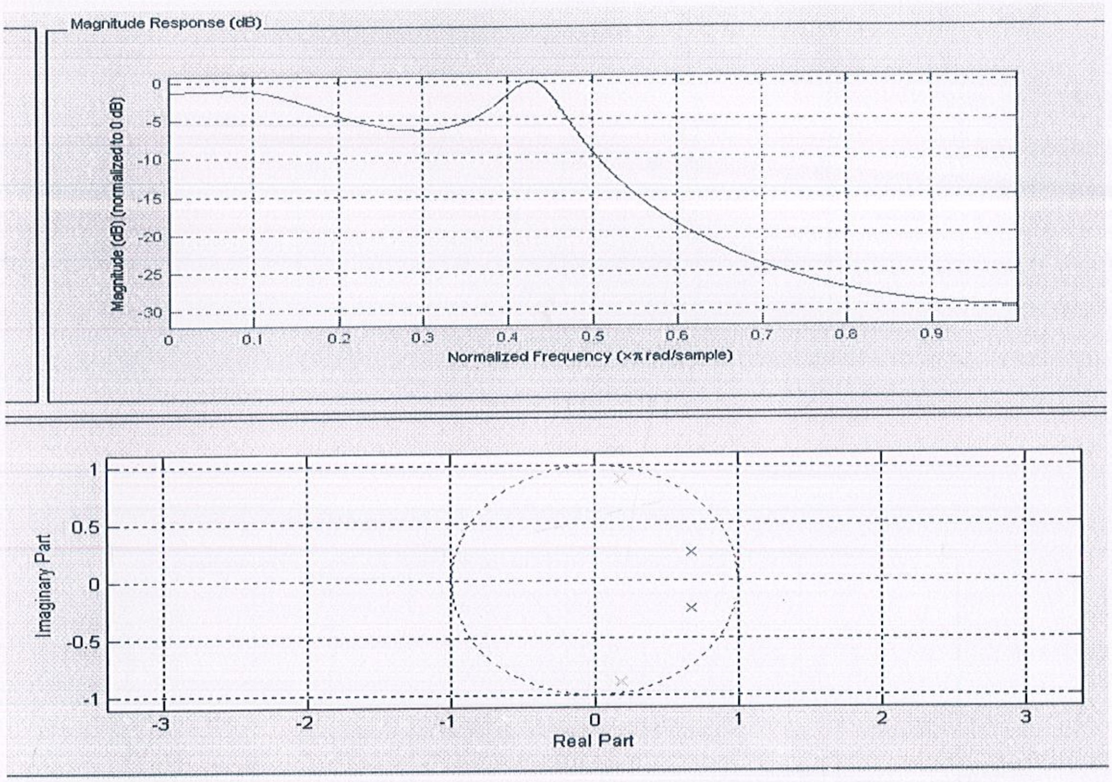


Fig. 3.16

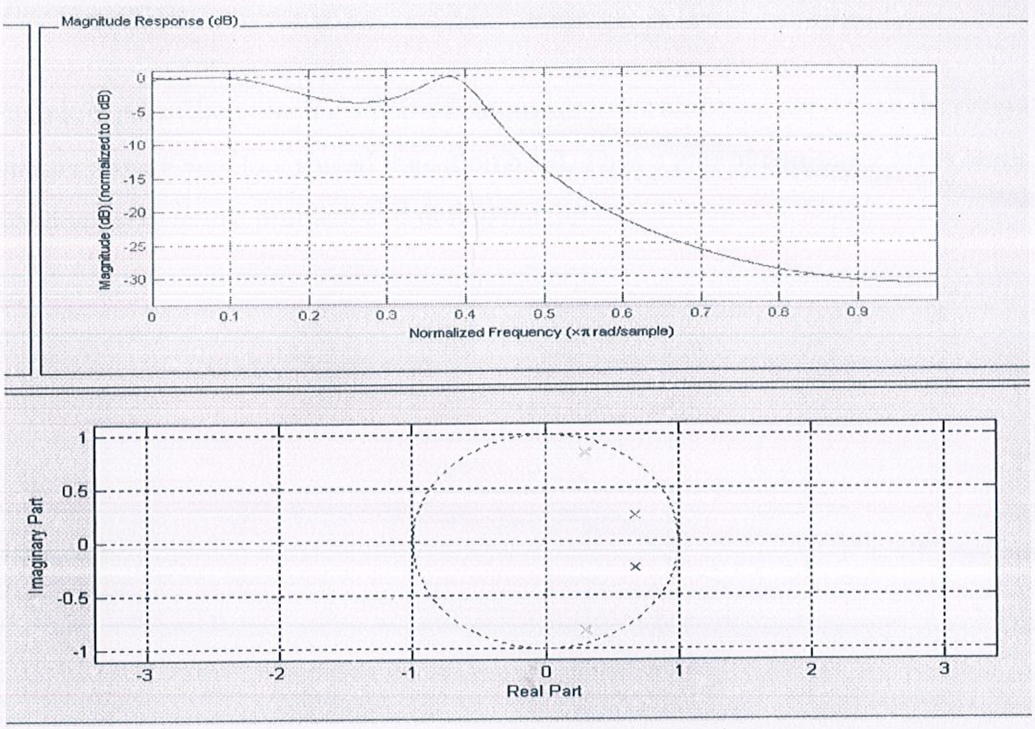


Fig. 3.17

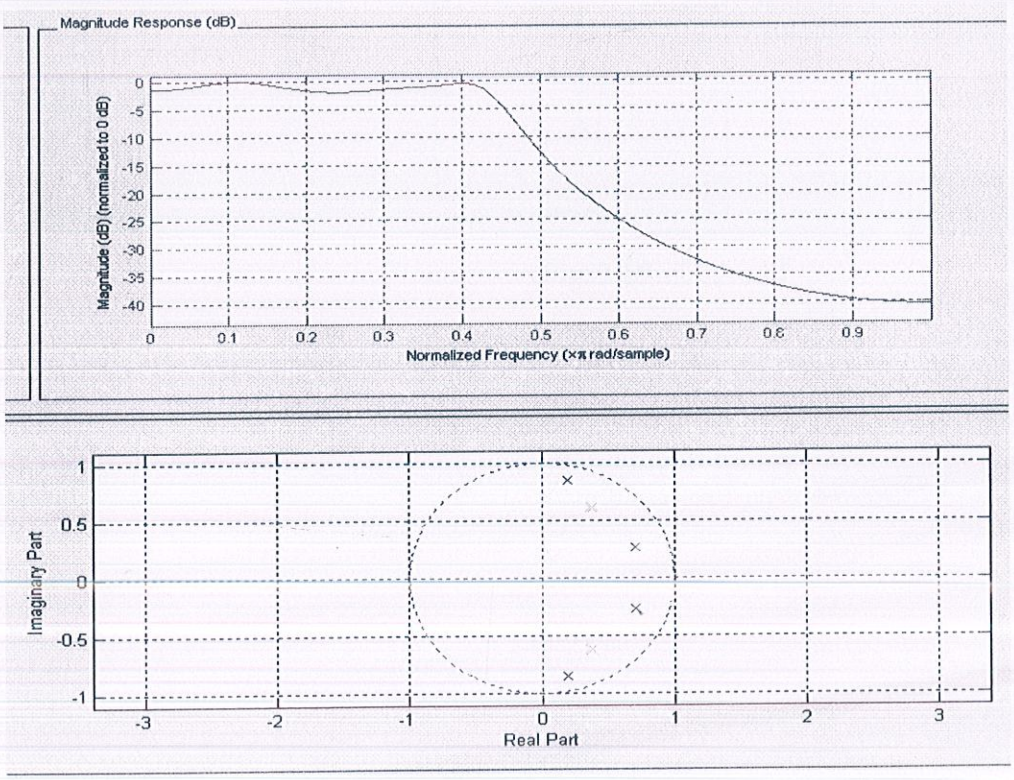


Fig. 3.18

Observation 8:

For a low pass filter, placing a pair of conjugate poles alone cannot define the drop sharply, giving way to a spread transition band. Thus a pair of conjugate zeros work just fine to solve this problem and define the dip.

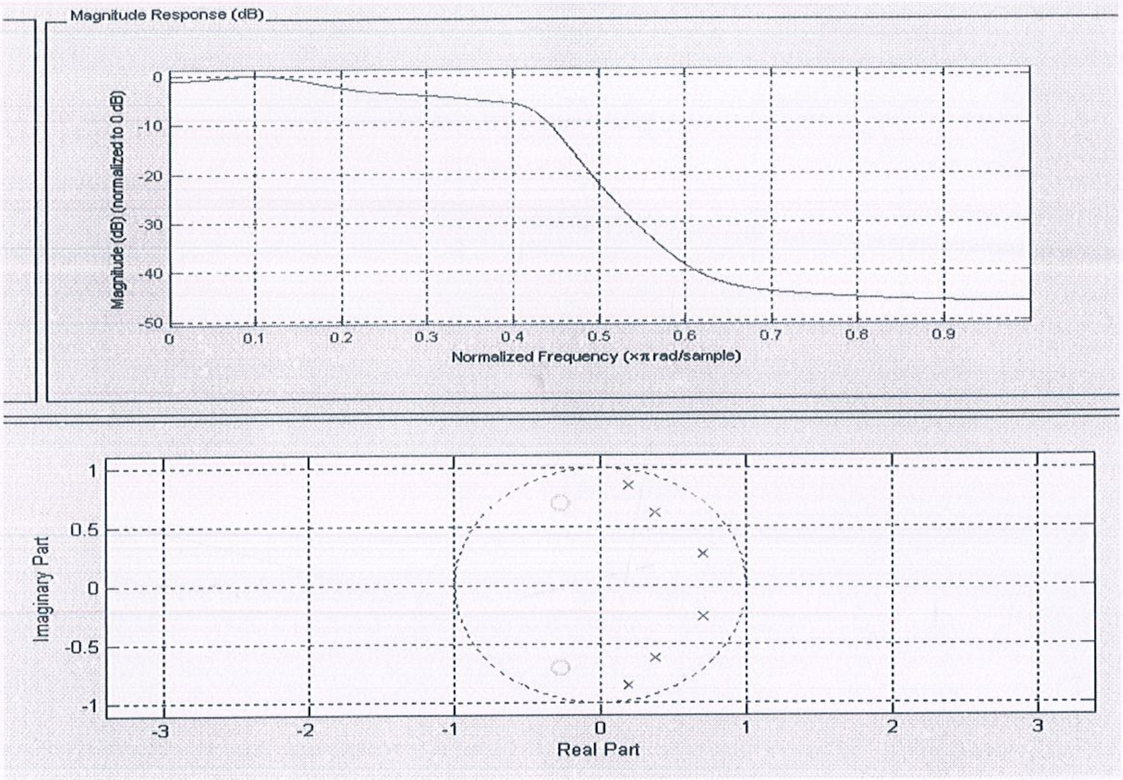


Fig. 3.19

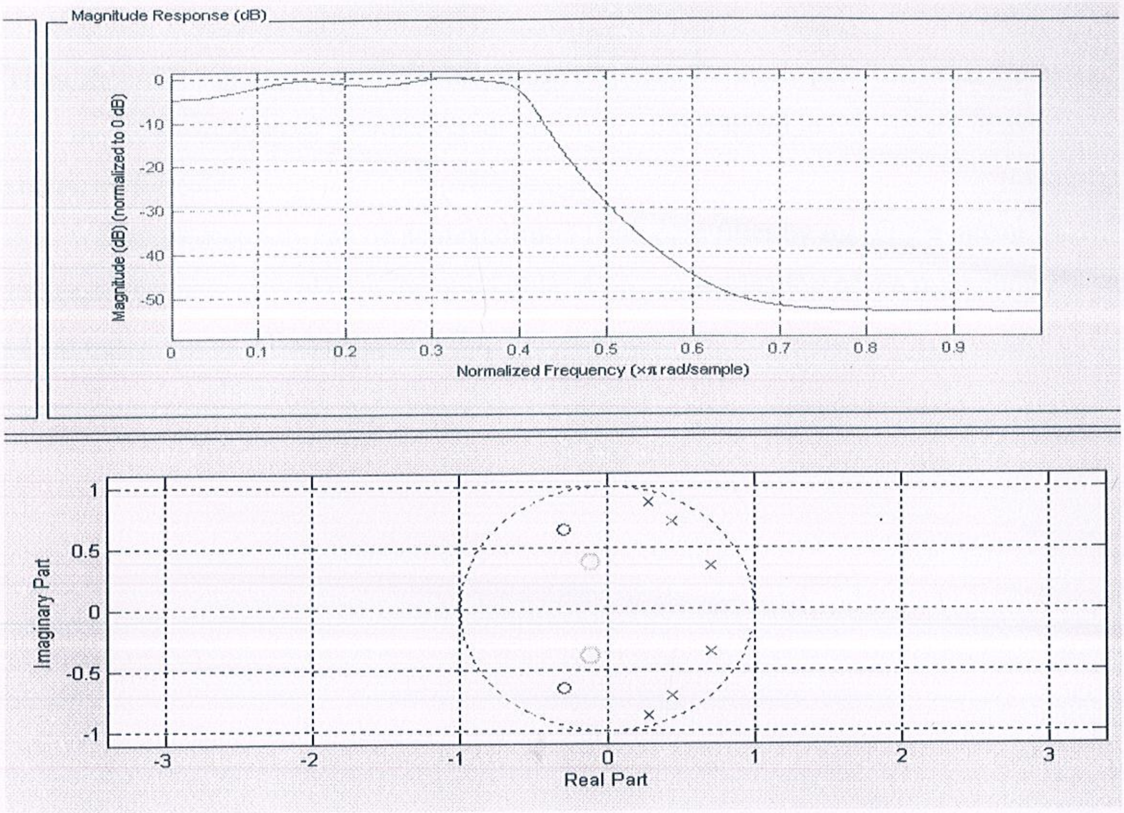


Fig. 3.20

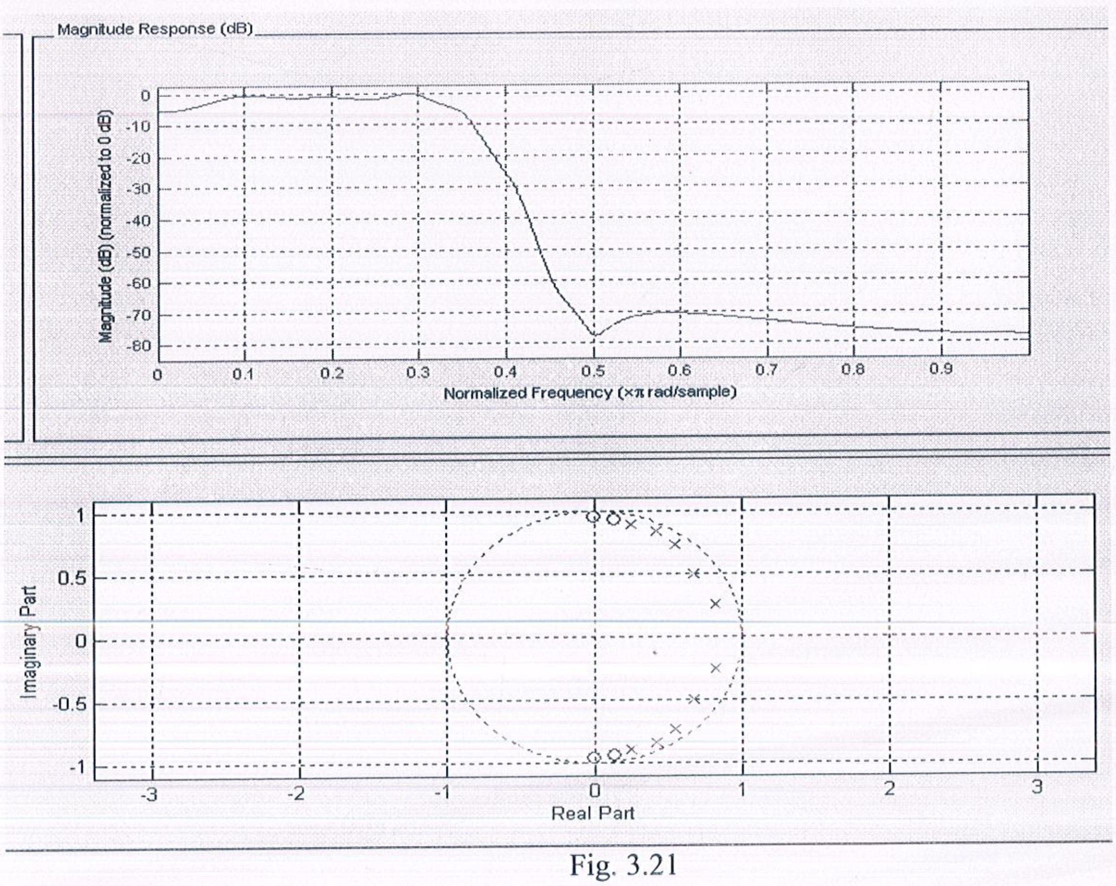


Fig. 3.21

Thus, the most superiorly possible magnitude response for a low pass filter that we have been able to achieve ,coalescing all the observations listed in this report, is as shown on fig 3.21.

Similar accuracy can be achieved for a High pass filter by forming a mirror image of this pole zero plot on the unit circle. Also, a combination of the right choice of the two gives us a Bandpass or a Bandreject filters.

CONCLUSIONS

- For any digital filter,

  A narrow peak can be obtained by –

  - A single pole placed in the unit circle
  - Placing the pair of conjugate poles near the circumference of the unit circle

  The peak can be widened by –

  - Placing multiple poles at the same radial distance
  - Placing the pair of conjugate poles closer to the centre of the unit circle

- For low pass filters,

  To achieve a minimal transition band-

  - Placement of a pair of conjugate zeros is necessary near the stop band frequency.
  - The pair of conjugate poles must be placed as far as possible. Their proximity widens the transition band.

  To achieve optimum tolerances in pass band and stop band

  - Multiple pairs of conjugate poles placed at the same radial distance flatten the pass band and stop band response, optimizing the tolerances.

REFERENCES

- J.G. Proakis and D.G. Manolakis, *Digital Signal Processing-Principles, Algorithms and Applications* New Delhi: Prentice-Hall, 2000

- http://www.mathworks.com/access/helpdesk/help/toolbox/signal/signal.shtml Signal Processing Toolbox, The Mathworks,Inc., accessed October 25,2002

- Arojit Roychowdhury and Prof P.C. Pandey, *FIR Filter Design Techniques* , IIT Bombay, submitted November2002