

Department of Electronic and Telecommunication Engineering

University of Moratuwa

EN2063—SIGNALS AND SYSTEMS

Filter Design REPORT

|  |  |
| --- | --- |
| Name | Index Number |
| SAIRISAN.R | 200522V |

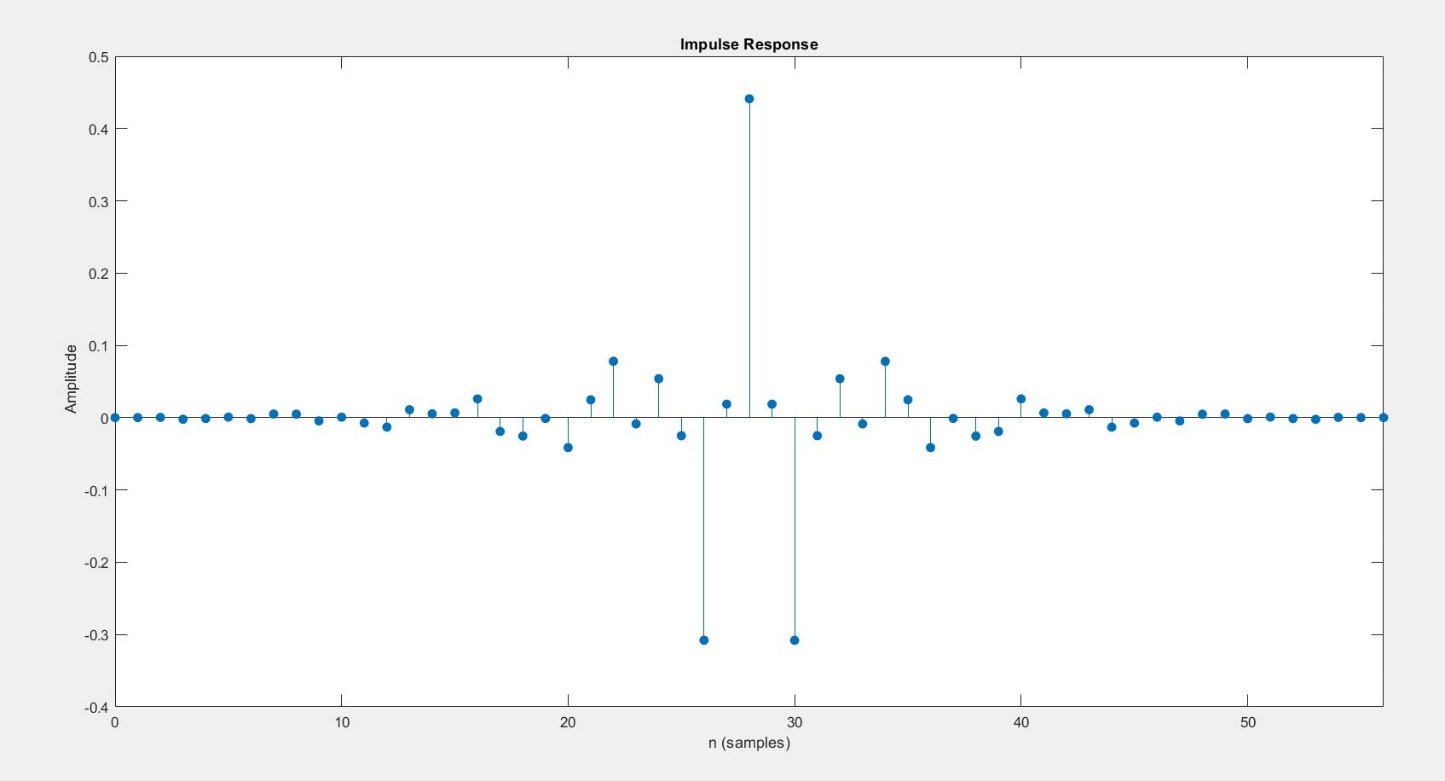
1.  a)

Fig 1.a – Impulse Response of FIR Filter

%DESIGN OF FIR FILTER...................................

%----------------------------------------------------------------------

%Index Number - 200552V

% A- First Digit of Index No.

% B- Second Digit of Index No.

% C- Third Digit of Index No.

% Ap- Maximum Passband ripple in DB

% Aa- Minimum Stopband attenuation in DB

% digital\_Wp1- Lowe Passband Edge in hertz

% digital\_Wp2- Upper Passband Edge in hertz

% digital\_Ws1- Lower Stopband Edge in hertz

% digital\_Ws2- Upper Stopband edge in hertz

% digital\_Wsm -Sampling Frequency in hertz

clear all,

A=5;

B=5;

C=2;

Ap= 0.1+(0.01\*A); Rp = Ap;

Aa= 50+B; Rs = Aa;

Wsm= 2\*(((C\*100)+1500));

Analog\_Wp1= ((C\*100)+400);

Analog\_Wp2= ((C\*100)+900);

Analog\_Ws1= ((C\*100)+100);

Analog\_Ws2= ((C\*100)+1100);

fcuts = [Analog\_Ws1 Analog\_Wp1 Analog\_Wp2 Analog\_Ws2]; %setting bandfrequencies (stop and passbands)

mags = [0 1 0];

devs = [10^(-Rs/20) 10^(-Rp/20) 10^(-Rs/20)]; % converstion of decible attenuation.

[n,Wn,beta,ftype] = kaiserord(fcuts,mags,devs,Wsm); %to computes the order of an FIR filter and the independent parameter to a Kaiser window

n = n + rem(n,2);

hh = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale'); %to calculates the coefficients of FIR filters designed using the windowing method

[h,f]=freqz(hh,1,1024,Wsm); % computes the frequency, magnitude, and phase response of a digital filter

%Code change only here....................

% plot(f,db(h))

% ylim([-75 10])

% % xlim([0 1])

% xlabel("Normalized Frequency ( × \pi rad/sample)");

% ylabel('Magnitute (DB)')

% title('56th order FIR Bandpass Filter')

impz(hh); %impulse response of the filter..

b)

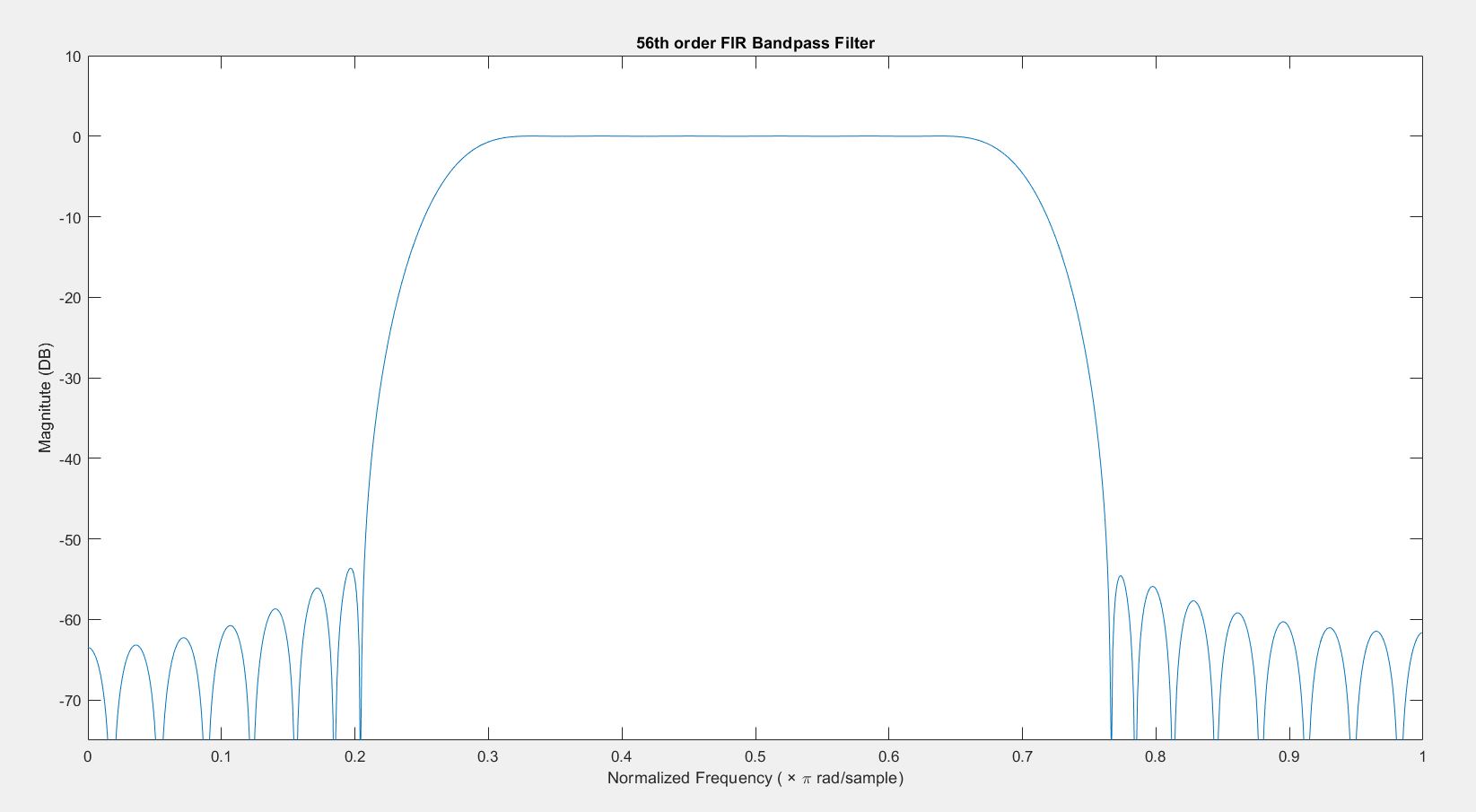


Fig 1.b –Frequency response of the FIR filter in Normalized Frequency (π rad/sample.)

%DESIGN OF FIR FILTER...................................

%----------------------------------------------------------------------

%Index Number - 200552V

% A- First Digit of Index No.

% B- Second Digit of Index No.

% C- Third Digit of Index No.

% Ap- Maximum Passband ripple in DB

% Aa- Minimum Stopband attenuation in DB

% digital\_Wp1- Lowe Passband Edge in hertz

% digital\_Wp2- Upper Passband Edge in hertz

% digital\_Ws1- Lower Stopband Edge in hertz

% digital\_Ws2- Upper Stopband edge in hertz

% digital\_Wsm -Sampling Frequency in hertz

clear all,

A=5;

B=5;

C=2;

Ap= 0.1+(0.01\*A); Rp = Ap;

Aa= 50+B; Rs = Aa;

Wsm= 2\*(((C\*100)+1500));

Analog\_Wp1= ((C\*100)+400);

Analog\_Wp2= ((C\*100)+900);

Analog\_Ws1= ((C\*100)+100);

Analog\_Ws2= ((C\*100)+1100);

fcuts = [Analog\_Ws1 Analog\_Wp1 Analog\_Wp2 Analog\_Ws2]; %setting bandfrequencies (stop and passbands)

mags = [0 1 0];

devs = [10^(-Rs/20) 10^(-Rp/20) 10^(-Rs/20)]; % converstion of decible attenuation.

[n,Wn,beta,ftype] = kaiserord(fcuts,mags,devs,Wsm); %to computes the order of an FIR filter and the independent parameter to a Kaiser window

n = n + rem(n,2);

hh = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale'); %to calculates the coefficients of FIR filters designed using the windowing method

[h,f]=freqz(hh,1,1024,Wsm); % computes the frequency, magnitude, and phase response of a digital filter

%Code change only here....................

plot((2\*f)/(Wsm),db(h))

ylim([-75 10])

xlim([0 1])

xlabel("Normalized Frequency ( × \pi rad/sample)");

ylabel('Magnitute (DB)')

title('56th order FIR Bandpass Filter')

% impz(hh); %impulse response of the filter..

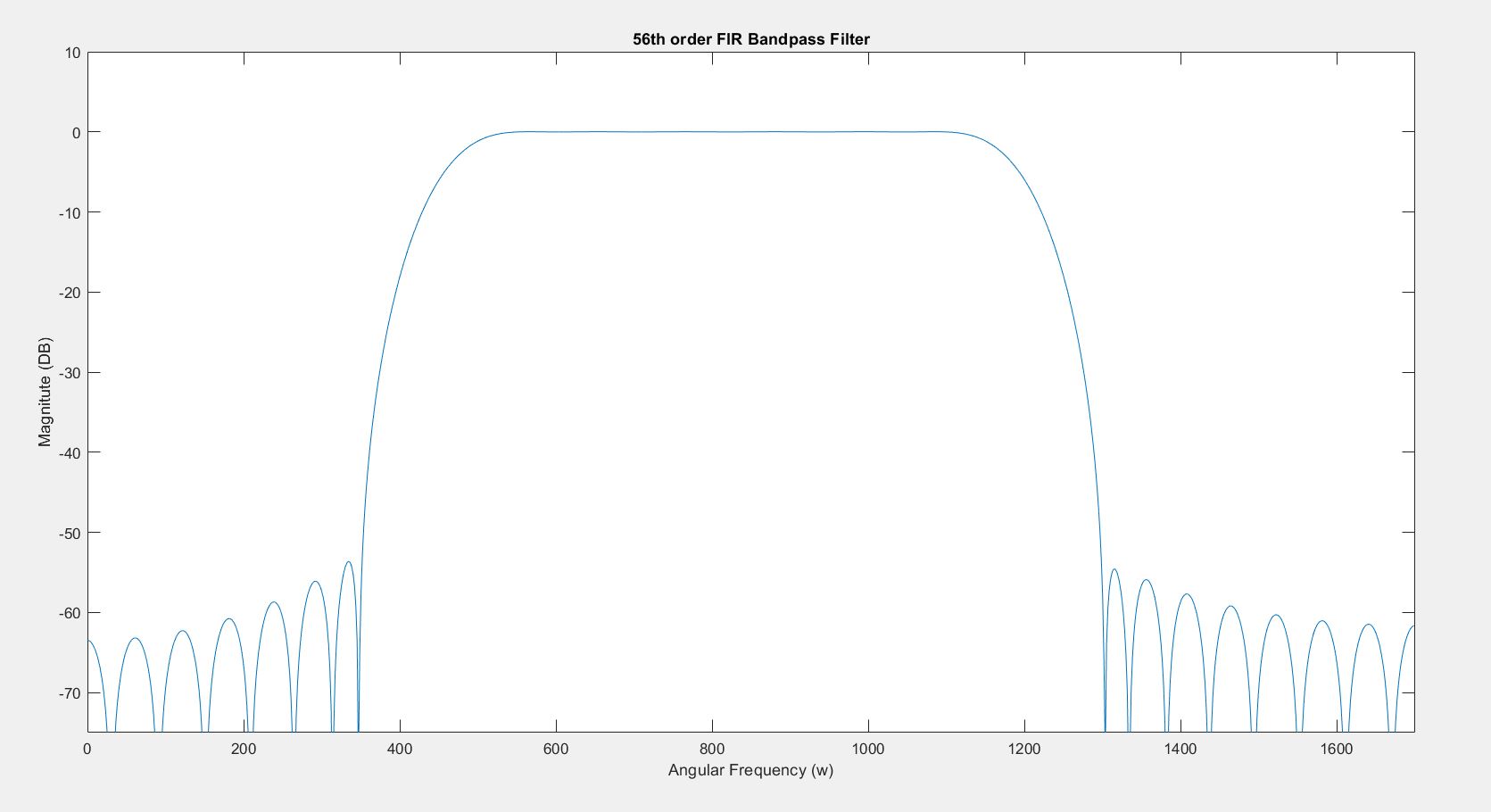
c)

Fig 1.c –Frequency response of the FIR filter in Passband Angular Frequency (Wp1<W<Wp2)

%DESIGN OF FIR FILTER...................................

%----------------------------------------------------------------------

%Index Number - 200552V

% A- First Digit of Index No.

% B- Second Digit of Index No.

% C- Third Digit of Index No.

% Ap- Maximum Passband ripple in DB

% Aa- Minimum Stopband attenuation in DB

% digital\_Wp1- Lowe Passband Edge in hertz

% digital\_Wp2- Upper Passband Edge in hertz

% digital\_Ws1- Lower Stopband Edge in hertz

% digital\_Ws2- Upper Stopband edge in hertz

% digital\_Wsm -Sampling Frequency in hertz

clear all,

A=5;

B=5;

C=2;

Ap= 0.1+(0.01\*A); Rp = Ap;

Aa= 50+B; Rs = Aa;

Wsm= 2\*(((C\*100)+1500));

Analog\_Wp1= ((C\*100)+400);

Analog\_Wp2= ((C\*100)+900);

Analog\_Ws1= ((C\*100)+100);

Analog\_Ws2= ((C\*100)+1100);

fcuts = [Analog\_Ws1 Analog\_Wp1 Analog\_Wp2 Analog\_Ws2]; %setting bandfrequencies (stop and passbands)

mags = [0 1 0];

devs = [10^(-Rs/20) 10^(-Rp/20) 10^(-Rs/20)]; % converstion of decible attenuation.

[n,Wn,beta,ftype] = kaiserord(fcuts,mags,devs,Wsm); %to computes the order of an FIR filter and the independent parameter to a Kaiser window

n = n + rem(n,2);

hh = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale'); %to calculates the coefficients of FIR filters designed using the windowing method

[h,f]=freqz(hh,1,1024,Wsm); % computes the frequency, magnitude, and phase response of a digital filter

%Code change only here....................

plot(f,db(h))

ylim([-75 10])

xlim([0 1700])

xlabel("Normalized Frequency ( × \pi rad/sample)");

ylabel('Magnitute (DB)')

title('56th order FIR Bandpass Filter')

% impz(hh); %impulse response of the filter..

1. a)

Fig 2.a –Filter Coefficients of IIR filter b1, a1

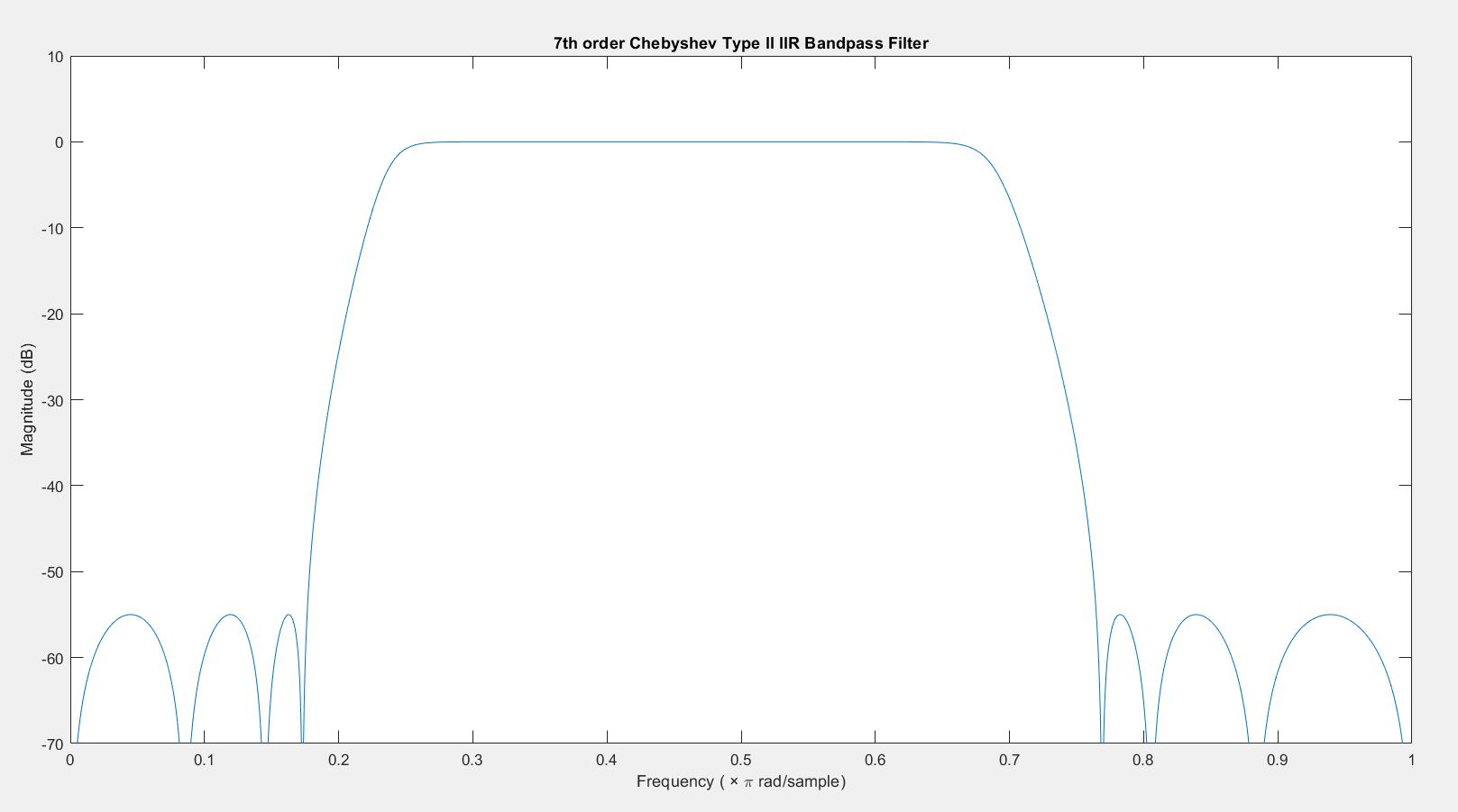
b)

Fig 2.a1 – IIR Inverse Chebyshev IIR Digital Filter Frequency response

Fig 2.b – Frequency response of the IIR Inverse Chebyshev filter in Normalized Frequency (π rad/sample.)

%DESIGN OF IIR FILTER...................................

%----------------------------------------------------------------------

%Index Number - 200552V

% A- First Digit of Index No

% B- Second Digit of Index No

% C- Third Digit of Index No

% Ap- Maximum Passband ripple in DB

% Aa- Minimum Stopband attenuation in DB

% digital\_Wp1- Lowe Passband Edge in hertz

% digital\_Wp2- Upper Passband Edge in hertz

% digital\_Ws1- Lower Stopband Edge in hertz

% digital\_Ws2- Upper Stopband edge in hertz

% digital\_Wsm -Sampling Frequency in hertz

clear all,

A=5;

B=5;

C=2;

Ap= 0.1+(0.01\*A); Rp = Ap;

Aa= 50+B; Rs = Aa;

Wsm= 2\*(((C\*100)+1500));

%Needed Digital Angluar Frequencies............................................

digital\_Wp1= ((C\*100)+400)/(2\*pi);

digital\_Wp2= ((C\*100)+900)/(2\*pi);

digital\_Ws1= ((C\*100)+100)/(2\*pi);

digital\_Ws2= ((C\*100)+1100)/(2\*pi);

%Prewarping Analog Frequency for Respective Frequencies.

analog\_Wp1 = 2\*Wsm\*tan(digital\_Wp1\*(2\*pi/(Wsm/(2\*pi)))/2);

analog\_Wp2 = 2\*Wsm\*tan(digital\_Wp2\*(2\*pi/(Wsm/(2\*pi)))/2);

analog\_Ws1 = 2\*Wsm\*tan(digital\_Ws1\*(2\*pi/(Wsm/(2\*pi)))/2);

analog\_Ws2 = 2\*Wsm\*tan(digital\_Ws2\*(2\*pi/(Wsm/(2\*pi)))/2);

[n,Ws] = cheb2ord([analog\_Wp1 analog\_Wp2],[analog\_Ws1 analog\_Ws2],Rp,Rs,'s'); % return minimum order and array of stopband angular freq

[b,a] = cheby2(n,Rs,[analog\_Ws1 analog\_Ws2],'bandpass','s'); % for calculating analog filter cofficients

% disp(n);

[b1,a1] = bilinear(b,a,Wsm); % for the for converting analog filter cofficients to digital IIR.

% disp(b1);

% disp(b1);

%Code change only here....................

[H,f]=freqz(b1,a1,2048,Wsm); % computes the frequency, magnitude, and phase response of a digital filter

plot(2\*f/Wsm,db(H)); % Normalized the frequency respect to nyquist sampling rate.

ylim([-70 10])

xlabel("Frequency ( × \pi rad/sample)");

ylabel("Magnitude (dB)");

title('7th order Chebyshev Type II IIR Bandpass Filter')

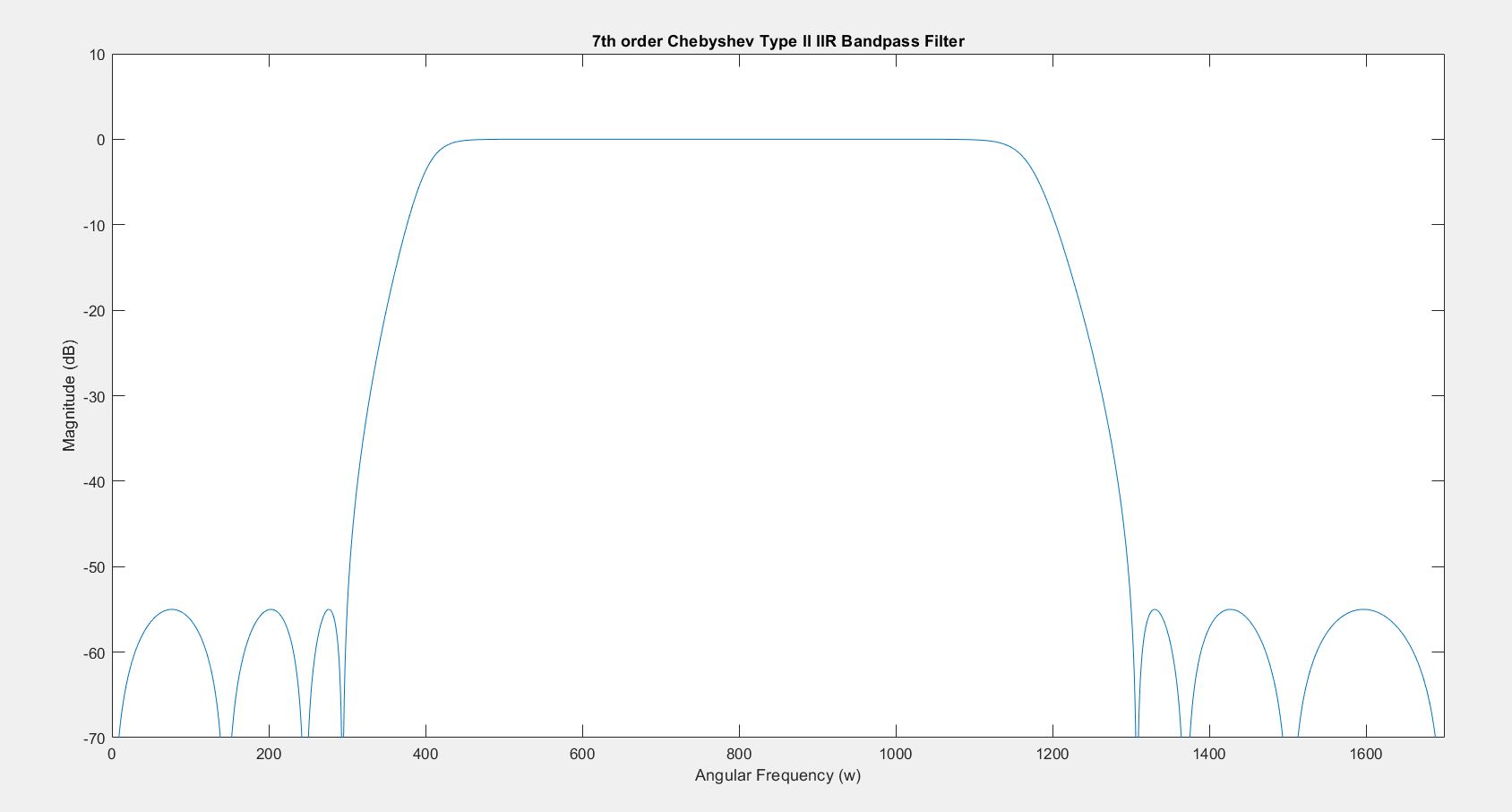
c)

Fig 2.c –Frequency response of the IIR filter in Passband Angular Frequency (Wp1<W<Wp2)

%DESIGN OF IIR FILTER...................................

%----------------------------------------------------------------------

%Index Number - 200552V

% A- First Digit of Index No

% B- Second Digit of Index No

% C- Third Digit of Index No

% Ap- Maximum Passband ripple in DB

% Aa- Minimum Stopband attenuation in DB

% digital\_Wp1- Lowe Passband Edge in hertz

% digital\_Wp2- Upper Passband Edge in hertz

% digital\_Ws1- Lower Stopband Edge in hertz

% digital\_Ws2- Upper Stopband edge in hertz

% digital\_Wsm -Sampling Frequency in hertz

clear all,

A=5;

B=5;

C=2;

Ap= 0.1+(0.01\*A); Rp = Ap;

Aa= 50+B; Rs = Aa;

Wsm= 2\*(((C\*100)+1500));

%Needed Digital Angluar Frequencies............................................

digital\_Wp1= ((C\*100)+400)/(2\*pi);

digital\_Wp2= ((C\*100)+900)/(2\*pi);

digital\_Ws1= ((C\*100)+100)/(2\*pi);

digital\_Ws2= ((C\*100)+1100)/(2\*pi);

%Prewarping Analog Frequency for Respective Frequencies.

analog\_Wp1 = 2\*Wsm\*tan(digital\_Wp1\*(2\*pi/(Wsm/(2\*pi)))/2);

analog\_Wp2 = 2\*Wsm\*tan(digital\_Wp2\*(2\*pi/(Wsm/(2\*pi)))/2);

analog\_Ws1 = 2\*Wsm\*tan(digital\_Ws1\*(2\*pi/(Wsm/(2\*pi)))/2);

analog\_Ws2 = 2\*Wsm\*tan(digital\_Ws2\*(2\*pi/(Wsm/(2\*pi)))/2);

[n,Ws] = cheb2ord([analog\_Wp1 analog\_Wp2],[analog\_Ws1 analog\_Ws2],Rp,Rs,'s'); % return minimum order and array of stopband angular freq

[b,a] = cheby2(n,Rs,[analog\_Ws1 analog\_Ws2],'bandpass','s'); % for calculating analog filter cofficients

% disp(n);

[b1,a1] = bilinear(b,a,Wsm); % for the for converting analog filter cofficients to digital IIR.

% disp(b1);

% disp(b1);

%Code change only here....................

[H,f]=freqz(b1,a1,2048,Wsm); % computes the frequency, magnitude, and phase response of a digital filter

plot(f,db(H)); % Normalized the frequency respect to nyquist sampling rate.

ylim([-70 10])

xlim([0 1700])

xlabel("Frequency ( × \pi rad/sample)");

ylabel("Magnitude (dB)");

title('7th order Chebyshev Type II IIR Bandpass Filter')



The order of the FIR filter (=56) which designed in direct method is higher than the Order of the IIR filter (=7) which designed in indirect method based on analog inverse Chebyshev filter. In other words, for a same specification FIR filter needs higher order to give performance like low order IIR filter. This is a drawback of FIR filter compare to IIR Filter. Few filter coefficients are enough to design IIR filters.

**FIR Filter**

Filter Coefficients =n+1=57

The number of multiplies = number of coefficients number of samples per second.

The number of multiplies = 57 3400samples/s = 193800/s

**IIR Filter**

Filter Coefficients =2n+1=15

The number of multiplies = number of coefficients number of samples per second.

The number of multiplies = 15 3400samples/s = 51000/s