Department of Electronic and Telecommunication Engineering

University of Moratuwa

EN2063—SIGNALS AND SYSTEMS

FILTER DESIGN REPORT
(FIR & IIR)

Name	Index Number
Manimohan T.	200377M

Due Date: - 19th January 2023,

Abstract

This is a detailed report discussing the complete procedure used for designing FIR (Bandpass) and IIR(Bandpass) digital filters for prescribed specifications and its implementation on MatLAB R2021a. This digital filter is designed for the prescribed specifications using the windowing method. The truncation of the impulse response is achieved using the Kaiser Window function. Here, first design an appropriate analog filter, and the required digital filter obtained by applying the bilinear transform to the transfer function of the analog filter. The report analyzes the magnitude and impulse responses of the filter to confirm its characteristics with initial parameter.

Table of Contents

List of Figures

List of Tables

1.Introduction

This paper outlines the step-by-step process for designing a band pass FIR digital filter to meet specified requirements using the windowing approach in conjunction with the Kaiser window and a band pass IIR digital filter to meet specified requirements using the bilinear transformation. MatLAB was used to implement this filter's software and evaluate it (Version R2021a).

To create FIR digital filters, a variety of techniques are used. In this report, the filter was defined and implemented using the Fourier series method using the closed form direct approach, and windowing was done using the Kaiser window. The filter was tuned to the required properties using the parameters of the Kaiser window.

Throughout the various design phases, the time domain and frequency domain representations of the filter are obtained to assess the filter's properties. The final outcomes are then examined and assessed to spot issues and make changes.

2.Results

The objective of this project is to provide experience in the design of FIR and IIR digital filters for prescribed specifications. For FIR filters, the windowing method (in conjunction with the Kaiser window) is used whereas, for IIR filters, the bilinear transformation method is used. The specifications of the digital filter are different from student to student and are derived using the index numbers of the students. Let us denote the index number as 200ABC·, where A, B and C are integers in the range 0 to 9, and · is a letter from the English alphabet.

My index number: -200377M

A = 3

B = 7

C = 7

Table 1: Filter specifications.

Parameter	Value	Value
Maximum passband ripple, $\widetilde{A}_p$	0.1 + (0.01 × 3) dB	0.13 dB
Minimum stopband attenuation, $\widetilde{A}_a$	50 + 7 dB	57 dB
Lower passband edge, Ω_{p1}	(7 × 100) + 400 rad/s	1100 rad/s
Upper passband edge, Ω_{p2}	(7 × 100) + 900 rad/s	1600 rad/s
Lower stopband edge, Ω_{s1}	(7 × 100) + 100 rad/s	800 rad/s
Upper stopband edge, Ω_{s2}	(7 × 100) + 1100 rad/s	1800 rad/s
Sampling frequency, Ω_{sm}	2((7 × 100) + 1500) rad/s	4400 rad/s

1) Using the windowing method in conjunction with the Kaiser window, design an FIR band pass digital filter that will satisfy the specifications given in Table 1.

a) Plot the impulse response.

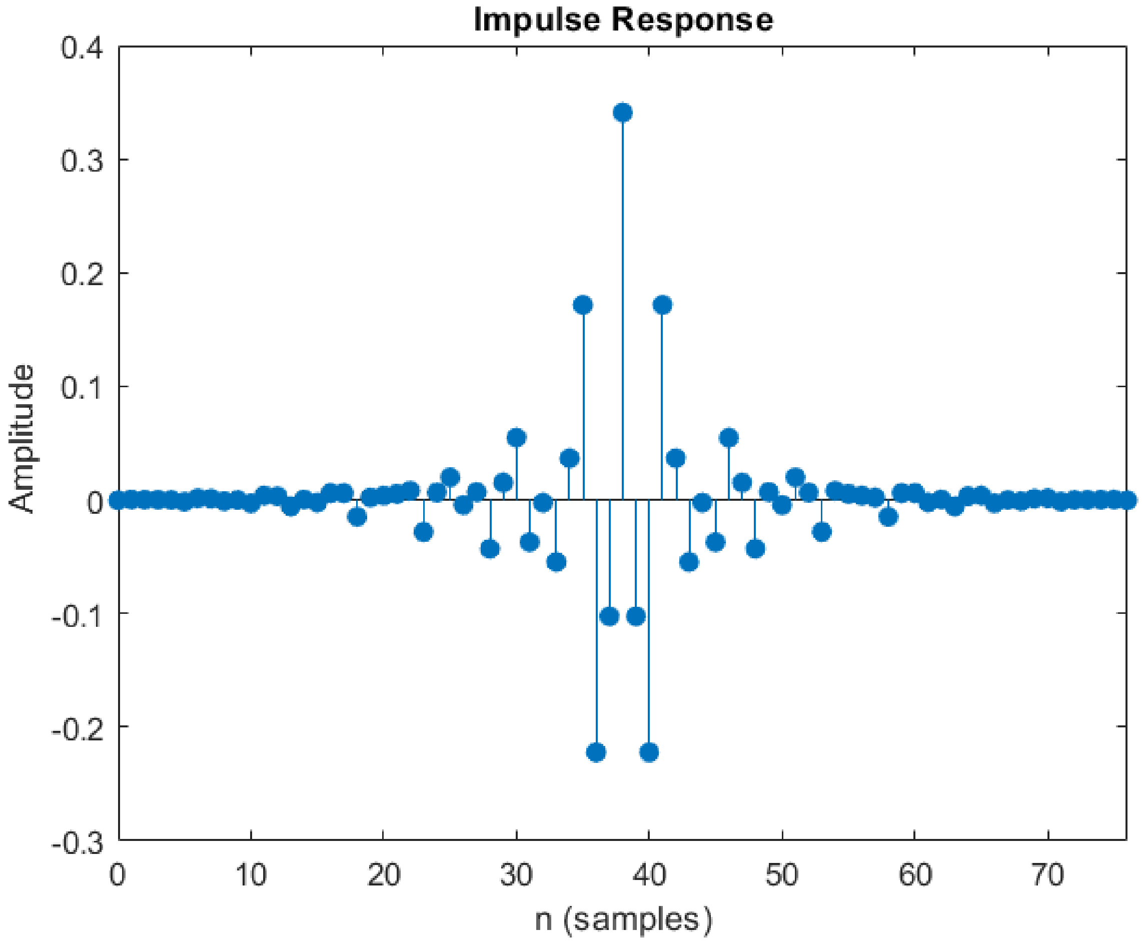

Figure 1 – Impulse Response of FIR Filter

b) Plot the magnitude response of the digital filter for $\pi \leq \omega < \pi$ rad/sample.

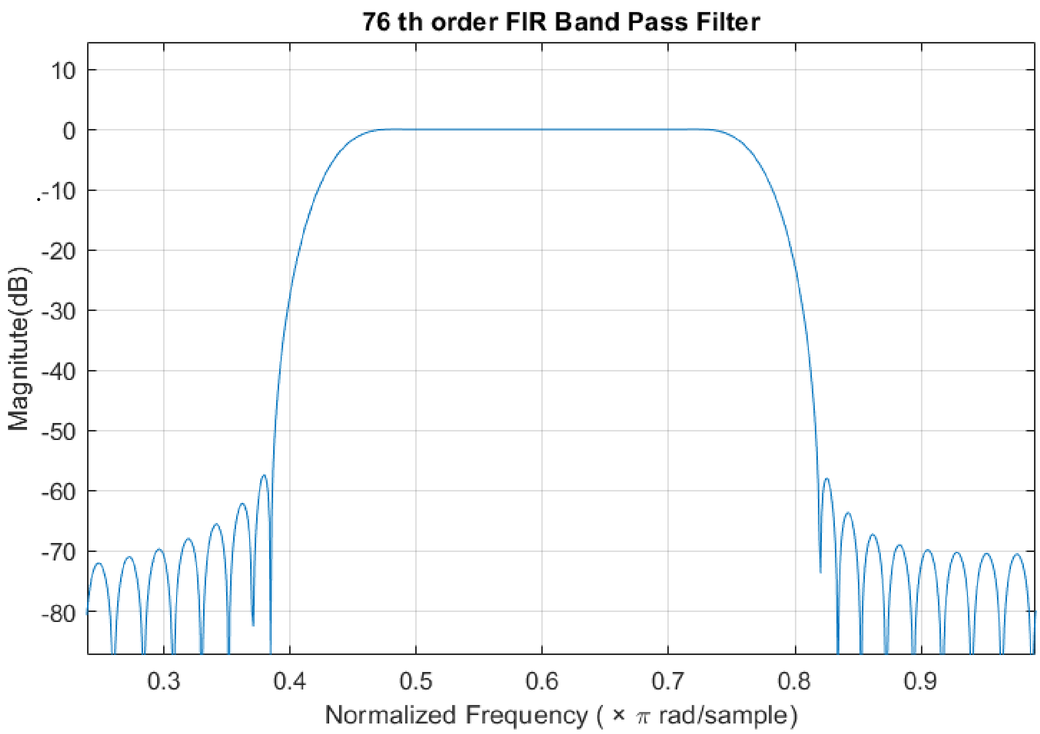

Figure 2 – Frequency response of the FIR filter in Normalized Frequency (π rad/sample)

c) Plot the magnitude response for $\omega_{p1} \leq \omega \leq \omega_{p2}$ (in the passband), where ω_{p1} and ω_{p2} are the passband edges in the discrete-time angular frequency domain.

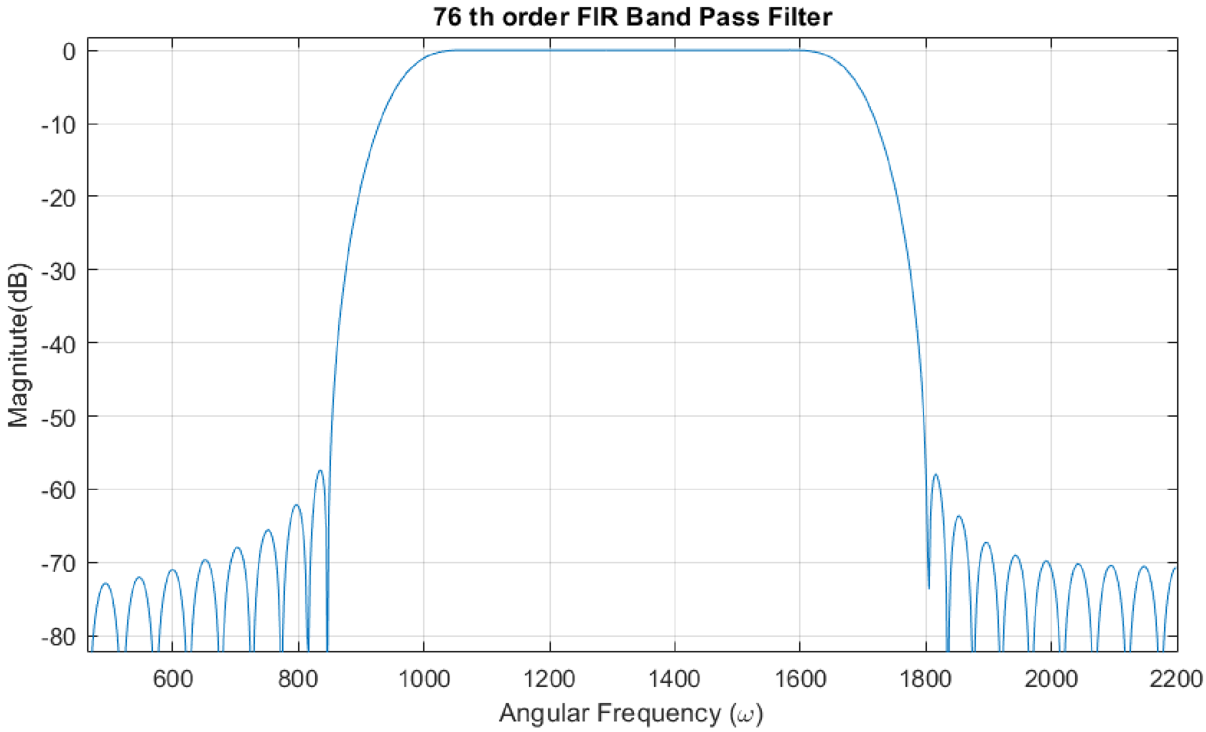

Figure 3 – Frequency response of the FIR filter in Passband Angular Frequency ($\omega_{p1} < \omega < \omega_{p2}$)

2) Using the bilinear transformation method, design an IIR bandpass digital filter that will satisfy the specifications given in Table 1. Here, you need to first design an appropriate analog filter, and the required digital filter should be obtained by applying the bilinear transform to the transfer function of the analog filter. Note that, prewarping of frequencies is essential in order to obtain the required digital filter. The approximation method (or the type) of the IIR filter is determined according to your index number as follows. Let D be the remainder after dividing the digit C of your index number by 4. Then the approximation method should be selected as presented in Table 2.

Table 2: IIR filter approximation method.

D	Approximation method
0	Butterworth
1	Chebyshev
2	Inverse-Chebyshev
3	Elliptic

According to my index number is 200377M.

Then C = 7 then D = C%4 = 7%4 = 3

So, I will use the Elliptic Approximation method.

a) Tabulate the coefficients of the transfer function of the IIR filter.

Table 3: Filter Coefficients of transfer function of IIR filter (b_1, a_1)

b_1	0.0076	0.0123	0.0045	0.0047	0.0127	-0.0000	-0.0127	-0.0047	-0.0045	-0.0123	-0.0076
a_1	1.0000	3.0918	7.1890	11.0474	14.1389	13.7132	11.2721	6.9995	3.6165	1.2163	0.3135

b) Plot the magnitude response of the digital filter for $\pi \leq \omega < \pi$ rad/sample.

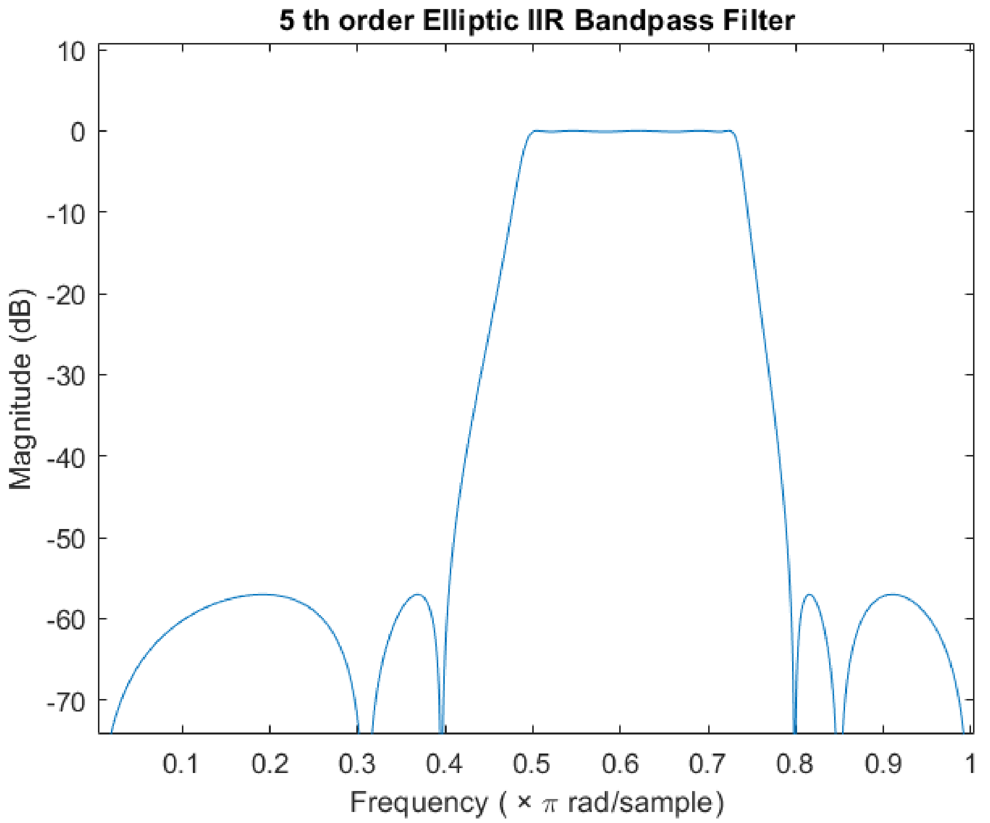

Figure 4 – Frequency response of the IIR Elliptic filter in Normalized Frequency (π rad/sample)

c) Plot the magnitude response for $\omega_{p1} \leq \omega \leq \omega_{p2}$ (in the passband), where ω_{p1} and ω_{p2} are the passband edges in the discrete-time angular frequency domain.

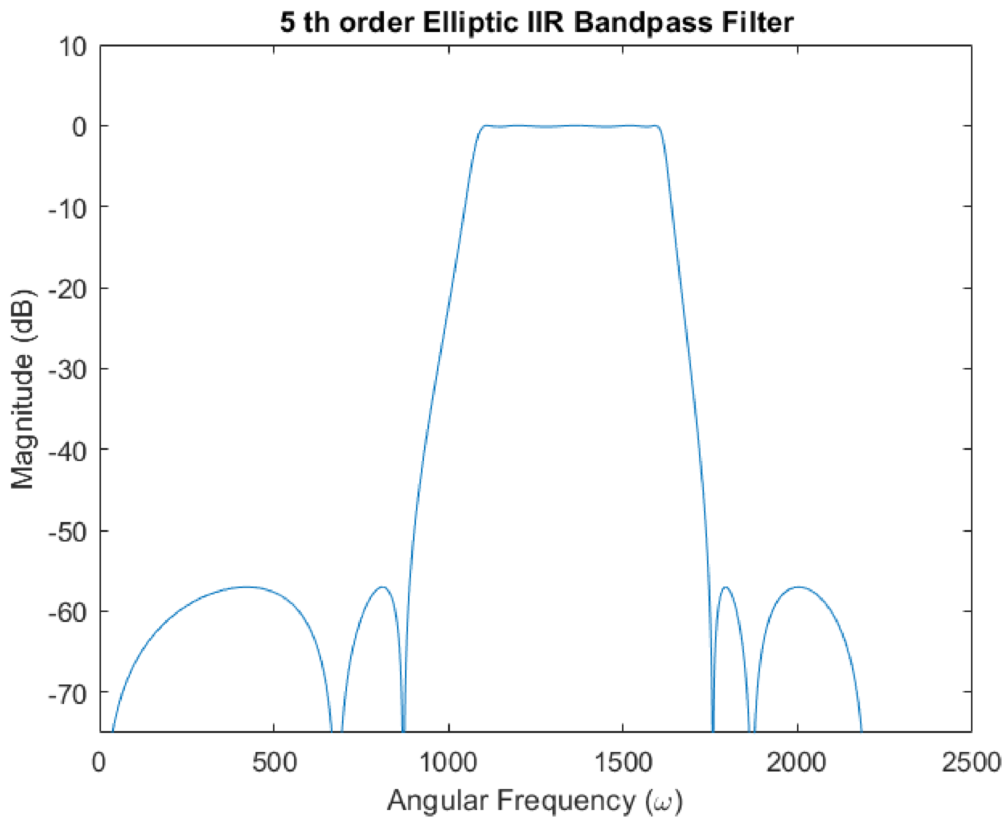

Figure 5 – Frequency response of the IIR Elliptic filter in Passband Angular Frequency ($\omega_{p1} < \omega < \omega_{p2}$)

3) Compare the order and the number of multiplications and additions required to process a sample by the two designed filters. Assume that the two filters are implemented using the difference equations, and the symmetry of coefficients can be exploited to reduce the number of multiplications.

Table 4: Filter Coefficients and Number of multiplies comparisons.

Filter	FIR Filter	IIR Filter
Filter Coefficients	n+1 = 76+1 = 77	2n+1= 2×5+1=11
The number of multiplies = number of coefficients × number of samples per second (Ω_{sm}).	77 × 4400 samples/s = 338800/s	11 × 4400 samples/s = 48400/s

The FIR band pass filter's (in direct method) order (n = 76) is higher than the IIR band pass filter's order (n = 5), which is based on an analog Elliptic bandpass filter and was constructed using an indirect method. In other words, a FIR bandpass filter with the same specifications needs a higher order to perform as well as a low order IIR bandpass filter. Comparing the FIR bandpass filter to the IIR bandpass filter, this is a disadvantage. To design bandpass IIR filters, just a few filter coefficients are required.

***

3.Appendix

Question 01)a)

```
%DESIGN OF FIR FILTER............
%Manimohan T.
%200377M
%Last Updated : 2023 Jan 17
clear all;
close all ;

% Get A,B,C
indexNo = str2double(inputdlg('Enter Index Number (only first six digits)', 'Index
Number',1)); %Index Number
A = mod(floor(indexNo/100),10); %A- First Digit of Index No.
B = mod(floor(indexNo/10),10); % B- Second Digit of Index No.
C = mod(indexNo,10); % C- Third Digit of Index No.
%A = 3; B = 7; C = 7;

%Required Filter Specifications
Ap= 0.1+(0.01*A); Rp = Ap; % Ap- Maximum Passband ripple in dB
Aa= 50+B; Rs = Aa;    % Aa- Minimum Stopband attenuation in dB
Wsm= 2*(((C*100)+1500));  %Sampling Frequency in Hz
%Analog filter properties
Wp1= ((C*100)+400); %Lower Passband Edge in Hz
Wp2= ((C*100)+900); %Upper Passband Edge in Hz
Ws1= ((C*100)+100);  %Lower Stopband Edge in Hz
Ws2= ((C*100)+1100);  %Upper Stopband edge in Hz

Cutoffs = [Ws1 Wp1 Wp2 Ws2]; %setting bandfrequencies (stop and passbands)
magnitude = [0 1 0];
deviation = [10^(-Rs/20) 10^(-Rp/20) 10^(-Rs/20)]; % conversation of decible
attenuation.
[n,Wn,beta,ftype] = kaiserord(Cutoffs,magnitude,deviation,Wsm); %to computes the
order of an FIR filter and the independent parameter to a Kaiser window
n = n + rem(n,2);
Coefficient = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale'); %to calculates the
coefficients of FIR filters designed using the windowing method
[h,f]=freqz(Coefficient,1,1024,Wsm); % computes the frequency, magnitude, and phase
response of a digital filter
```

Question 01)b)

```
%DESIGN OF FIR FILTER...........
%Manimohan T.
%200377M
%Last Updated : 2023 Jan 17
clear all;
close all ;

% Get A,B,C
indexNo = str2double(inputdlg('Enter Index Number(only first six digits)', 'Index
Number',1)); %Index Number
A = mod(floor(indexNo/100),10); %A- First Digit of Index No.
B = mod(floor(indexNo/10),10); % B- Second Digit of Index No.
C = mod(indexNo,10); % C- Third Digit of Index No.
%A = 3; B = 7; C = 7;

%Required Filter Specifications
Ap= 0.1+(0.01*A); Rp = Ap; % Ap- Maximum Passband ripple in dB
Aa= 50+B; Rs = Aa;   % Aa- Minimum Stopband attenuation in dB
Wsm= 2*(((C*100)+1500));  %Sampling Frequency in Hz
%Analog filter properties
Wp1= ((C*100)+400); %Lower Passband Edge in Hz
Wp2= ((C*100)+900); %Upper Passband Edge in Hz
Ws1= ((C*100)+100);  %Lower Stopband Edge in Hz
Ws2= ((C*100)+1100);  %Upper Stopband edge in Hz

Cutoffs = [Ws1 Wp1 Wp2 Ws2]; %setting bandfrequencies (stop and passbands)
magnitude = [0 1 0];
deviation = [10^(-Rs/20) 10^(-Rp/20) 10^(-Rs/20)]; % conversation of attenuation in
dB.
[n,Wn,beta,ftype] = kaiserord(Cutoffs,magnitude,deviation,Wsm); %to evaluate the
order of an FIR filter and independent parameter to  Kaiser window
n = n + rem(n,2);
Coefficient = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale'); %to calculates the
coefficients of FIR filter (windowing method)
[h,f]=freqz(Coefficient,1,1024,Wsm); % compute the frequency, magnitude, and phase
response of digital filter

%Code change only here....................
disp(n)
plot((2*f)/(Wsm),db(h))
grid
ylim([-75 10])
xlim([0 1])
xlabel("Normalized Frequency ( × \pi rad/sample)");
ylabel("Magnitute(dB)")
title(sprintf('%d th order FIR Band Pass Filter',n))
```

Question 01)c)

```
%DESIGN OF FIR FILTER............
%Manimohan T.
%200377M
%Last Updated : 2023 Jan 17
clear all;
close all ;

% Get A,B,C
indexNo = str2double(inputdlg('Enter Index Number (only first six digits)', 'Index
Number',1)); %Index Number
A = mod(floor(indexNo/100),10); %A- First Digit of Index No.
B = mod(floor(indexNo/10),10); % B- Second Digit of Index No.
C = mod(indexNo,10); % C- Third Digit of Index No.
%A = 3; B = 7; C = 7;

%Required Filter Specifications
Ap= 0.1+(0.01*A); Rp = Ap; % Ap- Maximum Passband ripple in dB
Aa= 50+B; Rs = Aa;   % Aa- Minimum Stopband attenuation in dB
Wsm= 2*(((C*100)+1500));  %Sampling Frequency in Hz
%Analog filter properties
Wp1= ((C*100)+400); %Lower Passband Edge in Hz
Wp2= ((C*100)+900); %Upper Passband Edge in Hz
Ws1= ((C*100)+100);  %Lower Stopband Edge in Hz
Ws2= ((C*100)+1100);  %Upper Stopband edge in Hz

Cutoffs = [Ws1 Wp1 Wp2 Ws2]; %setting bandfrequencies (stop and passbands)
magnitude = [0 1 0];
deviation = [(10^(-Rs/20)) (10^(-Rp/20)) (10^(-Rs/20))]; % conversation of attenuation
in dB.
[n,Wn,beta,ftype] = kaiserord(Cutoffs,magnitude,deviation,Wsm); %to evaluate the order
of an FIR filter and independent parameter to  Kaiser window
n = n + rem(n,2);
Coefficient = fir1(n,Wn,ftype,kaiser(n+1,beta),'noscale'); %to calculates the
coefficients of FIR filter (windowing method)
[H,f]=freqz(Coefficient,1,1024,Wsm); % compute the frequency, magnitude, and phase
response of digital filter

disp(n)
%Code change only here.....................
plot(f,db(H))
grid
ylim([-75 10])
xlim([0 2800])
xlabel(" Angular Frequency (\omega)");
ylabel("Magnitute(dB)")
title(sprintf('%d th order FIR Band Pass Filter',n))
```

Question 02)a)

```
%DESIGN OF IIR FILTER...........
%Manimohan T.
%200377M
%Last Updated : 2023 Jan 17
clear all;
close all ;

% Get A,B,C
indexNo = str2double(inputdlg('Enter Index Number (only first six digits)', 'Index
Number',1)); %Index Number
A = mod(floor(indexNo/100),10); %A- First Digit of Index No.
B = mod(floor(indexNo/10),10); % B- Second Digit of Index No.
C = mod(indexNo,10); % C- Third Digit of Index No.
%A = 3; B = 7; C = 7;

%Required Filter Specifications
Ap= 0.1+(0.01*A); Rp = Ap;
Aa= 50+B; Rs = Aa;
Wsm= 2*(((C*100)+1500));

%Required Digital Angluar Frequencies.............
wp1= ((C*100)+400)/(2*pi);
wp2= ((C*100)+900)/(2*pi);
ws1= ((C*100)+100)/(2*pi);
ws2= ((C*100)+1100)/(2*pi);

%Prewarping Analog Frequency for Respective Frequencies.
Wp1 = 2*Wsm*tan(wp1*(2*pi/(Wsm/(2*pi)))/2);
Wp2 = 2*Wsm*tan(wp2*(2*pi/(Wsm/(2*pi)))/2);
Ws1 = 2*Wsm*tan(ws1*(2*pi/(Wsm/(2*pi)))/2);
Ws2 = 2*Wsm*tan(ws2*(2*pi/(Wsm/(2*pi)))/2);

%Elliptic Approximation
[n,Wn] = ellipord([Wp1 Wp2],[Ws1 Ws2],Rp,Rs,'s'); % return minimum order and array
of stopband angular freq
[b,a] = ellip(n,Rp,Rs,Wn,'bandpass','s'); % for calculating analog filter
cofficients

%disp(n); % order of filter

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

disp(b1); %coefficients of the transfer function of the IIR filter
disp(a1); %coefficients of the transfer function of the IIR filter
```

Question 02)b)

```
%DESIGN OF IIR FILTER...........
%Manimohan T.
%200377M
%Last Updated : 2023 Jan 17
clear all;
close all ;

% Get A,B,C
indexNo = str2double(inputdlg('Enter Index Number (only first six digits)', 'Index
Number',1)); %Index Number
A = mod(floor(indexNo/100),10); %A- First Digit of Index No.
B = mod(floor(indexNo/10),10); % B- Second Digit of Index No.
C = mod(indexNo,10); % C- Third Digit of Index No.
%A = 3; B = 7; C = 7;

%Required Filter Specifications
Ap= 0.1+(0.01*A); Rp = Ap;
Aa= 50+B; Rs = Aa;
Wsm= 2*(((C*100)+1500));

%Required Digital Angluar Frequencies.............
wp1= ((C*100)+400)/(2*pi);
wp2= ((C*100)+900)/(2*pi);
ws1= ((C*100)+100)/(2*pi);
ws2= ((C*100)+1100)/(2*pi);

%Prewarping Analog Frequency for Respective Frequencies.
Wp1 = 2*Wsm*tan(wp1*(2*pi/(Wsm/(2*pi)))/2);
Wp2 = 2*Wsm*tan(wp2*(2*pi/(Wsm/(2*pi)))/2);
Ws1 = 2*Wsm*tan(ws1*(2*pi/(Wsm/(2*pi)))/2);
Ws2 = 2*Wsm*tan(ws2*(2*pi/(Wsm/(2*pi)))/2);

%Elliptic Approximation
[n,Wn] = ellipord([Wp1 Wp2],[Ws1 Ws2],Rp,Rs,'s'); % return minimum order and array
of stopband angular freq
[b,a] = ellip(n,Rp,Rs,Wn,'bandpass','s'); % for calculating analog filter
cofficients

%disp(n); % order of filter

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

%Code change only here....................
[H,f]=freqz(b1,a1,2048,Wsm); % to evaluate  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([-75 10])
xlabel("Frequency ( × \pi rad/sample)");
ylabel("Magnitude (dB)");
title(sprintf('%d th order  Elliptic IIR Bandpass Filter',n))
```

Question 02)c)

```
%DESIGN OF IIR FILTER...........
%Manimohan T.
%200377M
%Last Updated : 2023 Jan 17
clear all;
close all ;

% Get A,B,C
indexNo = str2double(inputdlg('Enter Index Number (only first six digits)', 'Index
Number',1)); %Index Number
A = mod(floor(indexNo/100),10); %A- First Digit of Index No.
B = mod(floor(indexNo/10),10); % B- Second Digit of Index No.
C = mod(indexNo,10); % C- Third Digit of Index No.
%A = 3; B = 7; C = 7;

%Required Filter Specifications
Ap= 0.1+(0.01*A); Rp = Ap;
Aa= 50+B; Rs = Aa;
Wsm= 2*(((C*100)+1500));

%Required Digital Angluar Frequencies............
wp1= ((C*100)+400)/(2*pi);
wp2= ((C*100)+900)/(2*pi);
ws1= ((C*100)+100)/(2*pi);
ws2= ((C*100)+1100)/(2*pi);

%Prewarping Analog Frequency for Respective Frequencies.
Wp1 = 2*Wsm*tan(wp1*(2*pi/(Wsm/(2*pi)))/2);
Wp2 = 2*Wsm*tan(wp2*(2*pi/(Wsm/(2*pi)))/2);
Ws1 = 2*Wsm*tan(ws1*(2*pi/(Wsm/(2*pi)))/2);
Ws2 = 2*Wsm*tan(ws2*(2*pi/(Wsm/(2*pi)))/2);

%Elliptic Approximation
[n,Wn] = ellipord([Wp1 Wp2],[Ws1 Ws2],Rp,Rs,'s'); % return minimum order and array
of stopband angular freq
[b,a] = ellip(n,Rp,Rs,Wn,'bandpass','s'); % for calculating analog filter
cofficients

%disp(n); % order of filter

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

%Code change only here...........
[H,f]=freqz(b1,a1,2048,Wsm); % to evaluate  the frequency, magnitude, and phase
response of a digital filter
plot(f,db(H)); % Normalized the frequency respect to nyquist sampling rate.
ylim([-75 10])
xlim([0 2500])
xlabel("Angular Frequency (\omega)");
ylabel("Magnitude (dB)");
title(sprintf('%d th order  Elliptic IIR Bandpass Filter',n))
```

***