LPF

```
%Suppose out target is to pass all frequencies below 1200 Hz
clc
clear
close all
f= input('Enter Target Frequency : '); % Target frequency 1200;
fs= input('Enter Sampling Frequency : '); % sampling frequency 8000;
n=input('Enter Order of the Filter : '); % order of the filter 50;
w=f/ (fs/2);    % Calculate cut off frequency between 0 to 1
b=fir1(n,w,'low'); % Zeros of the filter. b is window method that is real and has linear phase. The normalized gain of the filter at Wn is -6 dB.
% freqz(b,A,N,fs); a = gaing or no. of denominatior
freqz(b,1,128,fs); % Magnitude and Phase Plot of the filter
figure(2);
[h,w]=freqz(b,1,128,fs);
plot(w,abs(h)); % Normalized Magnitude Plot
grid
figure(3)
zplane(b,1);
```

HPF

```
%Now our target is to pass all frequencies above 1200 Hz
clc
clear
close all
f= input('Enter Target Frequency : '); % Target frequency 1200;
fs= input('Enter Sampling Frequency : '); % sampling frequency 8000;
n=input('Enter Order of the Filter : '); % order of the filter 50;
w=f/ (fs/2);  % Calculate cut off frequency between 0 to 1
b=fir1(n,w,'high'); % Zeros of the filter. b is window method that is real and has linear phase. The normalized gain of the filter at Wn is -6 dB.
freqz(b,1,128,fs);  % Magnitude and Phase Plot of the filter
figure(2)
[h,w]=freqz(b,1,128,fs);
plot(w,abs(h)); % Normalized Magnitude Plot
grid
figure(3)
zplane(b,1);
```

BPF

```
clc
clear
close all
```

```
% Target frequency between 1200 to 1800;
fs= input('Enter Sampling Frequency : '); % sampling frequency 8000;
n=input('Enter Order of the Filter : '); % order of the filter 50;
f = fs/2;
b=fir1(n,[1200/f 1800/f],'bandpass');
freqz(b,1,128,fs);
figure(2);
[h,w]=freqz(b,1,128,fs);
plot(w,abs(h)); % Normalized Magnitude Plot
grid;
figure(3);
zplane(b,1);
```

BSF

```
clc
clear
close all
% Target frequency between 1200 to 2800;
fs= input('Enter Sampling Frequency : '); % sampling frequency 8000;
n=input('Enter Order of the Filter : '); % order of the filter 40;
f = fs/2;
b=fir1(n,[1200/f 2800/f],'stop');
freqz(b,1,128,fs)
figure(2)
[h,w]=freqz(b,1,128,fs);
plot(w,abs(h)); % Normalized Magnitude Plot
grid
figure(3)
zplane(b,1);
```

NOTCH

```
clc
clear
close all
fs=8000;
n=40;
b=fir1(n,[1500/4000 1550/4000],'stop'); % frequncy difference between tf1 and tf2 are very narrow
freqz(b,1,128,fs)
figure(2)
[h,w]=freqz(b,1,128,fs);
plot(w,abs(h)); % Normalized Magnitude Plot
grid
figure(3)
zplane(b,1);
```

MULTI BAND

```
clc
clear
close all
n=50;
w=[0.2 0.4 0.6 .8]; % tf/(fs/2) = .2 then tf = .2/2*fs = 800
b=fir1(n,w);
freqz(b,1,128,8000);
figure(2);
[h,w]=freqz(b,1,128,8000);
plot(w,abs(h)); % Normalized Magnitude Plot
grid;
figure(3);
zplane(b,1);
```

IIR LPF

```
%Suppose our target is to design a filter to pass all frequencies below 1200 Hz with pass band
%ripples = 1 dB and minimum stop band attenuation of 50 dB at 1500 Hz. The sampling frequency
%for the filter is 8000 Hz;
clc;
clear;
close all;
tf = 1200;
fs=8000;
[n,w]=buttord(tf/(fs/2),1500/(fs/2),1,50); % finding the order of the filter
[b,a]=butter(n,w); % finding zeros and poles for filter
figure(1);
freqz(b,a,512,fs);  % Magnitude and Phase Plot of the filter
figure(2)
[h,q] = freqz(b,a,512,fs);
plot(q,abs(h)); % Normalized Magnitude plot
grid;
figure(3);
f=1200:2:1500;
freqz(b,a,f,fs); % plotting the Transition band
figure(4);
zplane(b,a); % pole zero constellation diagram
```

IIR HPF

```
%ripples = 1 dB and minimum stop band attenuation of 50 dB at 1500 Hz. The sampling frequency
%for the filter is 8000 Hz;
%We will consider same filter but our target now is to pass all frequencies above 1200 Hz
clc;
```

```
clear;
close all;
tf = 1200;
fs=8000;

[n,w]=buttord(tf/(fs/2),1500/(fs/2),1,50); % finding the order of the filter
[b,a]=butter(n,w,'high');% finding zeros and poles for filter
figure(1);
freqz(b,a,512,fs);  % Magnitude and Phase Plot of the filter
figure(2);
[h,q] = freqz(b,a,512,fs);
plot(q,abs(h)); % Normalized Magnitude plot
grid;
figure(3);
f=1200:2:1500;
freqz(b,a,f,fs); % plotting the Transition band
figure(4);
zplane(b,a);% pole zero constellation diagram
```

IIR BPF

```
%with pass band ripples = 1 dB and minimum stop band attenuation of 50 dB. The
%sampling frequency for the filter is 8000 Hz;
clc
clear
close all
[n,w]=buttord([800/4000,2800/4000],[400/4000, 3200/4000],1,50); %Wp = [.2 .7], Ws = [.1 .8]
buttord(Wp,Ws,Rp,Rs
[b,a]=butter(n,w,'bandpass');
figure(1);
freqz(b,a,128,8000);
figure(2);
[h,w]=freqz(b,a,128,8000);
plot(w,abs(h));
grid;
figure(3);
f=600:2:1200;
freqz(b,a,f,8000); % Transition Band
figure(4);
f=2800:2:3200;
freqz(b,a,f,8000); % Transition Band
figure(5);
zplane(b,a);
```

IIR BSF

```
%with pass band ripples = 1 dB and minimum stop band attenuation of 50 dB. The
%sampling frequency for the filter is 8000 Hz;
clc
clear
close all
[n,w]=buttord([800/4000,2800/4000],[400/4000, 3200/4000],1,50); %Wp = [.1 .8], Ws = [.2 .7]
buttord(Ws,Wp,Rp,Rs)
[b,a]=butter(n,w,'stop');
figure(1);
freqz(b,a,128,8000)
[h,w]=freqz(b,a,128,8000);
figure(2)
plot(w,abs(h));
grid;
figure(3)
f=600:2:1200;
freqz(b,a,f,8000); % Transition Band
figure(4)
f=2800:2:3200;
freqz(b,a,f,8000); % Transition Band
figure(5)
zplane(b,a);
```

SAMPLING

```
% Define the parameters of the signal
clc;
clear;
close all;
f = input('Enter Frequency of the Sinusoid : ');          % Frequency of the sinusoid (in Hz): 10
fs = input('Enter Sampling Frequency : ');                % Sampling rate (in Hz): 200
t = 0:1/fs:1;                                   % Time vector
x = sin(2*pi*f*t);                              % Generate the sinusoidal signal
subplot(3,1,1);
plot(t,x);                                      % Plot the original signal
xlabel('Time (s)');
ylabel('Amplitude');
title('Original Signal');
% Sample the signal
Ts = 1/fs;         % Sampling interval (in seconds)
n = 0:Ts:1;        % Sampling instants
xn = sin(2*pi*f*n);  % Sampled signal
% Plot the sampled signal
subplot(3,1,2);
stem(n,xn);
xlabel('Time (s)');
ylabel('Amplitude');
```

```
title('Sampled Signal');

% Reconstruct the analog signal using ideal reconstruction
xr = zeros(size(t));  % Initialize the reconstructed signal as zero
for i = 1:length(n)
   xr = xr + xn(i)*sinc((t-(i-1)*Ts)/Ts);
end

% Plot the reconstructed signal
subplot(3,1,3);
plot(t,xr);
xlabel('Time (s)');
ylabel('Amplitude');
title('Reconstructed Signal');
```

Z-TRANSFORM

```
clc
clear
close all
syms z n
a=1/16^n;                    %x(n) =  [1/16^n]u(n)
ZTrans=ztrans(a);      %Z transform
disp('Z-transform : ');
disp(ZTrans);
InvrZ=iztrans(ZTrans);      %InverseZtransform
disp('Incerse Z-transform : ');
disp(InvrZ);


B=[0 1 1];
A=[1 -2 3];
pl = roots(A); % To display pole value
disp('Pole :');
disp(pl)
zr = roots(B); % To display zero value
disp('Zeros :');
disp(zr)
figure(1);
zplane(B,A); % Compute and display pole-zero diagram

syms z n
I=iztrans(3*z/(z+1));
disp(I);

clc;clear;
```

```
z=roots([1,0,2]);
p=roots([1,2,-1,1]);

clc;clear;
z=roots([1,0,0,1]);
p=roots([1,0,2,0,1]);

clc;clear;
ZZ=roots([4,8,10]);
PP=roots([2,8,18,20]);

% using pzmap method calculate poles and zeros
clc;clear;
num=[1,0,0,1];
den=[1,0,2,0,1];
systf=tf(num,den);
figure(2);
pzmap(systf);
```

DFT AND FFT

```
% x(n) = {1,2,3,4,5} stating point 2
clc
clear
close all
n=-1:3; % stating with -1 and end with 3 i.e (-1, 0, 1, 2, 3)
x=1:5; % sample 1 to 5
k=0:500; %N = 500 i.e 0 to 500 = 501
w=(pi/500)*k;
X=x*(exp(-1j*pi/500))*(n'*k);
magX=abs(X);
angX=angle(X);
realX=real(X);
imagX=imag(X);

subplot(2,2,1);
plot(k/500,magX);
grid;
xlabel('Frequency in pi units ');
title('Magnitude part');
```

```
subplot(2,2,2);
plot(k/500,angX/pi);
grid;
xlabel('Frequency in pi units ');
title('Angle part');


subplot(2,2,3);
plot(k/500,realX);
grid;
xlabel('Frequency in pi units ');
title('Real part');


subplot(2,2,4);
plot(k/500,imagX);
grid;
xlabel('Frequency in pi units ');
title('Imaginary part');
```

FFT

```
%%%%%%            FFT PART            %%%%%%%%%%%%%%
% Find The spectrum of the following signal
% f=0.25+2*sin(2*pi*5*k)+1*sin(2*pi*12.5*k)+1.5*sin(2*pi*20*k)+0.5*sin(2*pi*35*k);
clc
clear
close all
N=256; % Number of Samples
T=1/128; %sampling frequency 128Hz
k=0:N-1;
time=k*T;
f=0.25+2*sin(2*pi*5*k*T)+1*sin(2*pi*12.5*k*T)+1.5*sin(2*pi*20*k*T)+0.5*sin(2*pi*35*k*T);
subplot(2,1,1);
plot(time,f);
title('Signal sampled at 128Hz');
grid on;
F=fft(f);
magF=abs([F(1)/N,F(2:N/2)/(N/2)]);
hertz=k(1:N/2)*(1/(N*T));

subplot(2,1,2);
stem(hertz,magF);
title('Frequency Components');
grid on;
```