**EXPERIMENT 8**

By,

**Raja Aadhithan**

**19/1031**

Write a MATLAB Script to find the impulse response and step response of the designed FIR and IIR filters( in Lab6 and Lab7) .

%Write a MATLAB Script to design the FIR filter using Window Method. %1. Rectangular (rectwin) 2. Hamming 3. Hann 4. Kaiser

**Code:**

clc;

clear;

close all;

type = input('Enter the type of filter-1.Low Pass,2. High Pass,3.Band Pass,4. Band Stop: ');

win = input('Enter the type of window- 1.Rectangular, 2.Hamming, 3.Hann,4. Kaiser: ');

n = input('Enter the order of the filter: ');

if(type == 1)

n1 = input('Enter the cutoff frequency(in multiples of pi) : ');

if(win == 1)

fil = fir1(n,n1,'low',rectwin(n+1));

freqz(fil,1);

title('Low Pass Filter with Rectangular Window');

elseif(win == 2)

fil = fir1(n,n1,'low',hamming(n+1));

freqz(fil,1);

title('Low Pass Filter with Hamming Window');

elseif(win == 3)

fil = fir1(n,n1,'low',hann(n+1));

freqz(fil,1);

title('Low Pass Filter with Hann Window');

elseif(win == 4)

fil = fir1(n,n1,'low',kaiser(n+1));

freqz(fil,1);

title('Low Pass Filter with Kaiser Window');

end

elseif(type == 2)

n1 = input('Enter the cutoff frequency(in multiples of pi) : ');

if(mod(n,2))

n=n+1;

end

if(win == 1)

fil = fir1(n,n1,'high',rectwin(n+1));

freqz(fil,1);

title('High Pass Filter with Rectangular Window');

elseif(win == 2)

fil = fir1(n,n1,'high',hamming(n+1));

freqz(fil,1);

title('High Pass Filter with Hamming Window');

elseif(win == 3)

fil = fir1(n,n1,'high',hann(n+1));

freqz(fil,1);

title('High Pass Filter with Hann Window');

elseif(win == 4)

fil = fir1(n,n1,'high',kaiser(n+1));

freqz(fil,1);

title('High Pass Filter with Kaiser Window');

end

elseif(type == 3)

n1 = input('Enter the first cutoff frequency(in multiples of pi): ');

n2 = input('Enter the second cutoff frequency(in multiples of pi): ');

if(win == 1)

fil = fir1(n,[n1 n2],'bandpass',rectwin(n+1));

freqz(fil,1);

title('Band Pass Filter with Rectangular Window');

elseif(win == 2)

fil = fir1(n,[n1 n2],'bandpass',hamming(n+1));

freqz(fil,1);

title('Band Pass Filter with Hamming Window');

elseif(win == 3)

fil = fir1(n,[n1 n2],'bandpass',hann(n+1));

freqz(fil,1);

title('Band Pass Filter with Hann Window');

elseif(win == 4)

fil = fir1(n,[n1 n2],'bandpass',kaiser(n+1));

freqz(fil,1);

title('Band Pass Filter with Kaiser Window');

end

elseif(type == 4)

n1 = input('Enter the first cutoff frequency(in multiples of pi): ');

n2 = input('Enter the second cutoff frequency(in multiples of pi): ');

if(mod(n,2))

n=n+1;

end

if(win == 1)

fil = fir1(n,[n1 n2],'stop',rectwin(n+1));

freqz(fil,1);

title('Band Stop Filter with Rectangular Window');

elseif(win == 2)

fil = fir1(n,[n1 n2],'stop',hamming(n+1));

freqz(fil,1);

title('Band Stop Filter with Hamming Window');

elseif(win == 3)

fil = fir1(n,[n1 n2],'stop',hann(n+1));

freqz(fil,1);

title('Band Stop Filter with Hann Window');

elseif(win == 4)

fil = fir1(n,[n1 n2],'stop',kaiser(n+1));

freqz(fil,1);

title('Band Stop Filter with Kaiser Window');

end

end

figure(2);

impz(fil);

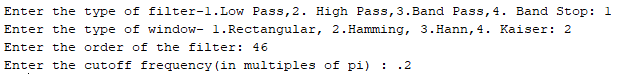
title('Impulse Response of the designed filter');

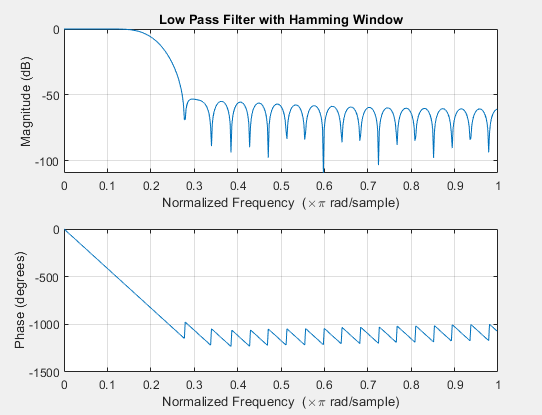
figure(3);

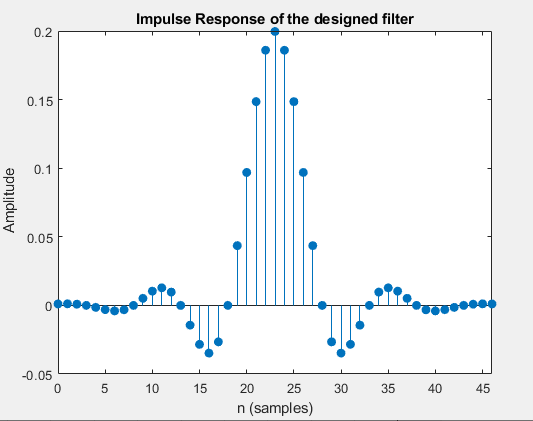
stepz(fil);

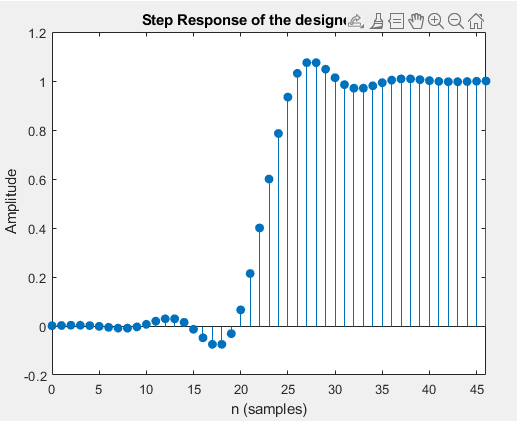
title('Step Response of the designed filter');

**Output:**

****

****

****

****

**Write a MATLAB Script to design the Butterworth, Chebyshev and Elliptic filters based on 1. Bilinear Transformation 2. Impulse Invariant Transformation (Lab 7)**

**Code:**

%Write a MATLAB Script to design the Butterworth, Chebyshev and Elliptic filters based on

% Bilinear Transformation and Impulse Invariant Transformation

clc;

clear;

close all;

type = input('Enter the type of filter- 1.Low Pass, 2.High Pass, 3.Band Pass, 4.Band Stop: ');

g = input('Enter the design of filter- 1.Butterworth, 2.Chebyshev type1, 3.Chebyshev type2, 4.Elliptic: ');

k = input('Enter the type of Filter discretization functions1.Bilinear, 2.Impulse Invariant: ');

rp = input('Enter the pass band ripple: ');

rs = input('Enter the stop band attenuation: ');

wp = input('Enter the pass band frequency(Hz): ');

ws = input('Enter the stop band frequency(Hz): ');

fs = input('Enter the sampling frequency(Hz): ');

wp = wp/(fs/2);

ws = ws/(fs/2);

if(type == 1)

if(g == 1)

[n,wn] = buttord(wp,ws,rp,rs);

[b,a] = butter(n,wn,'low');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Low Pass Butterworth Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Low Pass Butterworth Filter',n));

elseif(g == 2)

[n,wp] = cheb1ord(wp,ws,rp,rs);

[b,a] = cheby1(n,rp,wp,'low');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Low Pass Chebyshev Type 1 Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Low Pass Chebyshev Type 1 Filter',n));

elseif(g == 3)

[n,ws] = cheb2ord(wp,ws,rp,rs);

[b,a] = cheby2(n,rs,ws,'low');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Low Pass Chebyshev Type 2 Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Low Pass Chebyshev Type 2 Filter',n));

elseif(g == 4)

[n,wp] = ellipord(wp,ws,rp,rs);

[b,a] = ellip(n,rp,rs,wp,'low');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Low Pass Elliptic Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Low Pass Elliptic Filter',n));

end

elseif(type == 2)

if(g == 1)

[n,wn] = buttord(wp,ws,rp,rs);

[b,a] = butter(n,wn,'high');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital High Pass Butterworth Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog High Pass Butterworth Filter',n));

elseif(g == 2)

[n,wp] = cheb1ord(wp,ws,rp,rs);

[b,a] = cheby1(n,rp,wp,'high');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital High Pass Chebyshev Type 1 Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog High Pass Chebyshev Type 1 Filter',n));

elseif(g == 3)

[n,ws] = cheb2ord(wp,ws,rp,rs);

[b,a] = cheby2(n,rs,ws,'high');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital High Pass Chebyshev Type 2 Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog High Pass Chebyshev Type 2 Filter',n));

elseif(g == 4)

[n,wp] = ellipord(wp,ws,rp,rs);

[b,a] = ellip(n,rp,rs,wp,'high');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital High Pass Elliptic Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog High Pass Elliptic Filter',n));

end

elseif(type == 3)

if(g == 1)

[n,wn] = buttord(wp,ws,rp,rs);

[b,a] = butter(n,wn,'bandpass');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Bandpass Butterworth Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Bandpass Butterworth Filter',n));

elseif(g == 2)

[n,wp] = cheb1ord(wp,ws,rp,rs);

[b,a] = cheby1(n,rp,wp,'bandpass');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Bandpass Chebyshev Type 1 Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Bandpass Chebyshev Type 1 Filter',n));

elseif(g == 3)

[n,ws] = cheb2ord(wp,ws,rp,rs);

[b,a] = cheby2(n,rs,ws,'bandpass');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Bandpass Chebyshev Type 2 Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Bandpass Chebyshev Type 2 Filter',n));

elseif(g == 4)

[n,wp] = ellipord(wp,ws,rp,rs);

[b,a] = ellip(n,rp,rs,wp,'bandpass');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Bandpass Elliptic Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Bandpass Elliptic Filter',n));

end

elseif(type == 4)

if(g == 1)

[n,wn] = buttord(wp,ws,rp,rs);

[b,a] = butter(n,wn,'stop');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Bandstop Butterworth Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Bandstop Butterworth Filter',n));

elseif(g == 2)

[n,wp] = cheb1ord(wp,ws,rp,rs);

[b,a] = cheby1(n,rp,wp,'stop');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Bandstop Chebyshev Type 1 Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Bandstop Chebyshev Type 1 Filter',n));

elseif(g == 3)

[n,ws] = cheb2ord(wp,ws,rp,rs);

[b,a] = cheby2(n,rs,ws,'stop');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Bandstop Chebyshev Type 2 Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Bandstop Chebyshev Type 2 Filter',n));

elseif(g == 4)

[n,wp] = ellipord(wp,ws,rp,rs);

[b,a] = ellip(n,rp,rs,wp,'stop');

if(k ==1)

[bz,az] = bilinear(b,a,fs);

elseif(k==2)

[bz,az] = impinvar(b,a,fs);

end

freqz(bz,az,1024,fs);

title(sprintf('n = %d Digital Bandstop Elliptic Filter',n));

figure(2);

freqz(b,a,1024,fs);

title(sprintf('n = %d Analog Bandstop Elliptic Filter',n));

end

end

figure(3);

impz(bz,az,25);

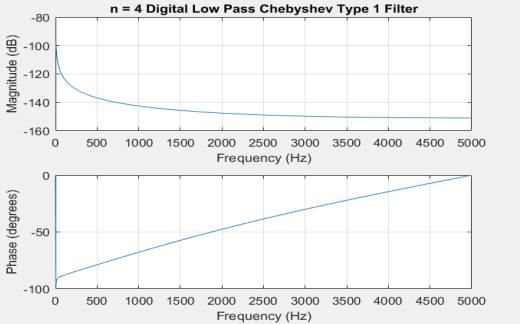
title('Impulse Response of the designed filter');

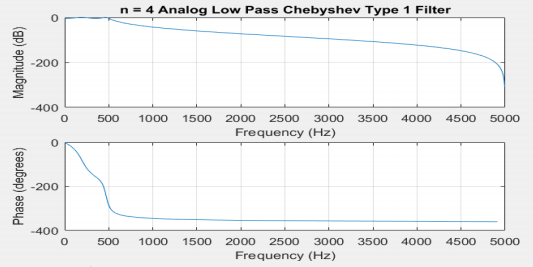
figure(4);

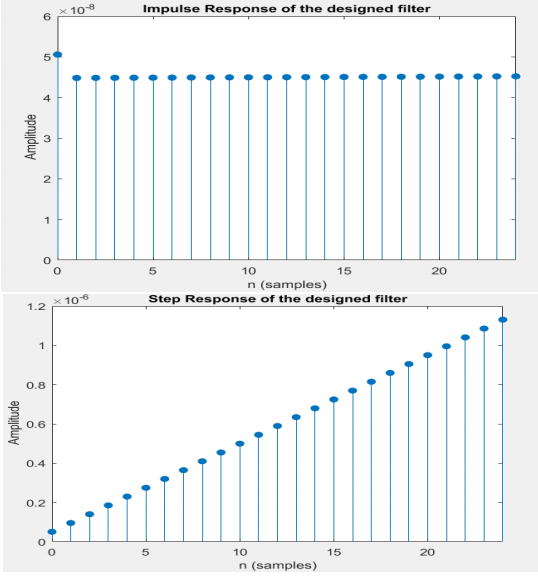
stepz(bz,az,25);

title('Step Response of the designed filter');

**OUTPUT:**

****

****

****