Experiment No. 10

PROGRAM TO IMPLEMENT LOW PASS FIR FILTER

Aim: To implement low pass FIR filter for a given sequence.

Software Required:

Operating System	:	Windows Xp
Constructor	:	Simulator
Software	:	MAT Lab 7.5 & CCStudio3.1

MATLAB Program to Design Low Pass Filter Using the Kaiser Window Technique

```
clc;  clear all;  close all;
rp=input ('Enter the pass band ripple:');
rs=input ('Enter the stop band ripple:');
fp=input ('Enter the pass band frequency:');
fs=input ('Enter the stop band frequency:');
f=input ('Enter the sampling Frequency:');
beta=input('Enter the beta value:');
wp=2*fp/f;
ws=2*fs/f;
num=-20*log(sqrt(rp*rs))-13;
dem=14.6*(fs-fp)/f;
n=ceil(num/dem);
n1=n+1;
if (rem(n,2)~=0)
   n1=n;
   n=n-1;
end
y=kaiser(n1,beta);
   b=fir1(n,wp,'low',y);
 [h,om]=freqz(b,1,256);
subplot(2,1,1);
plot(om/pi,20*log10(abs(h)));
```

```
ylabel('Gain in dB--------->');
xlabel('Normalised Frequency-->');
 subplot(2,1,2);
plot(om/pi,angle(h));
ylabel('angle--------->');
xlabel('Normalised Frequency-->');
```

OUTPUT:

Enter the pass band ripple:0.04

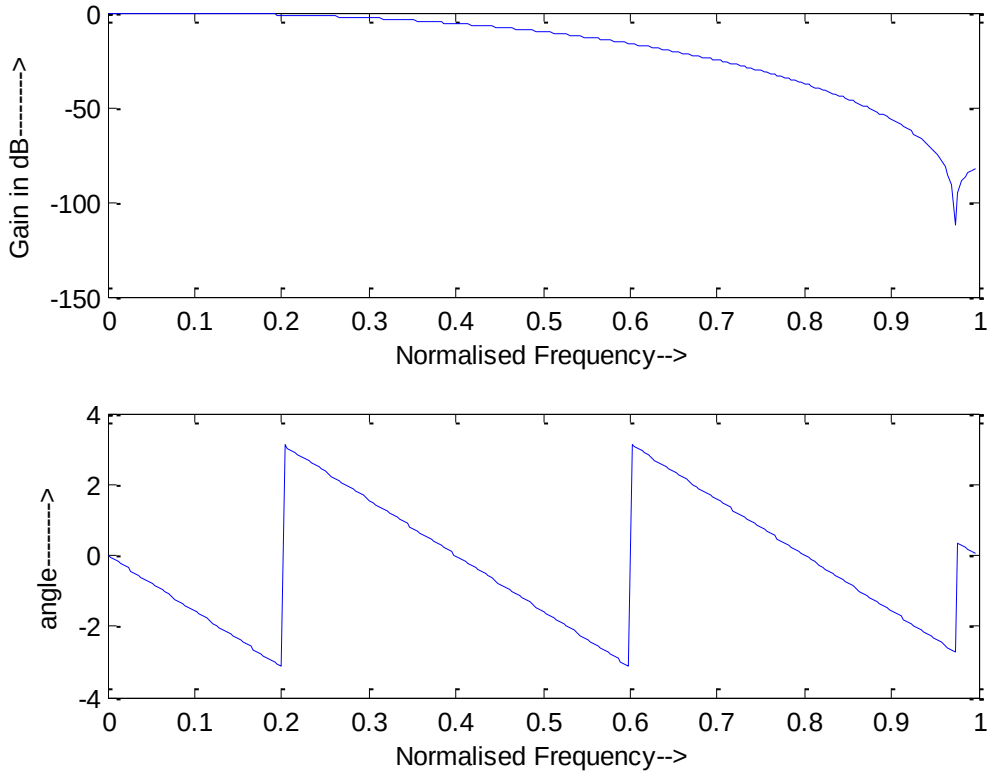
Enter the stop band ripple:0.06

Enter the pass band frequency:200

Enter the stop band frequency:500

Enter the sampling Frequency:1000

Enter the beta value:9



%FIR FILTER DESIGN WINDOW TECHNIQUES

```
clc;    clear all;    close all;
rp=input('enter passband ripple');
rs=input('enter the stopband ripple');
fp=input('enter passband freq');
fs=input('enter stopband freq');
f=input('enter sampling freq ');
wp=2*fp/f;
ws=2*fs/f;
num=-20*log10(sqrt(rp*rs))-13;
dem=14.6*(fs-fp)/f;
n=ceil(num/dem);
n1=n+1;
if(rem(n,2)~=0)
n1=n;
n=n-1;
end
c=input('enter your choice of window function 1. rectangular 2. triangular 3.kaiser: \n ');
if(c==1)
y=rectwin(n1);
disp('Rectangular window filter response');
end
if (c==2)
y=triang(n1);
disp('Triangular window filter response');
end
if(c==3)
y=kaiser(n1);
disp('kaiser window filter response');
end
```

%LPF

```
b=fir1(n,wp,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,1);plot(o/pi,m);
title('LPF');
ylabel('Gain in dB-->');       xlabel('(a) Normalized frequency-->');
%HPF
b=fir1(n,wp,'high',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,2);plot(o/pi,m);
title('HPF');
ylabel('Gain in dB-->');       xlabel('(b) Normalized frequency-->');
%BPF
wn=[wp ws];
b=fir1(n,wn,y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,3);plot(o/pi,m);
title('BPF');
ylabel('Gain in dB-->');       xlabel('(c) Normalized frequency-->');
%BSF
b=fir1(n,wn,'stop',y);
[h,o]=freqz(b,1,256);
m=20*log10(abs(h));
subplot(2,2,4);plot(o/pi,m);
title('BSF');
ylabel('Gain in dB-->');       xlabel('(d) Normalized frequency-->');
```

OUTPUT:

enter passband ripple0.04

enter the stopband ripple0.06

enter passband freq1000

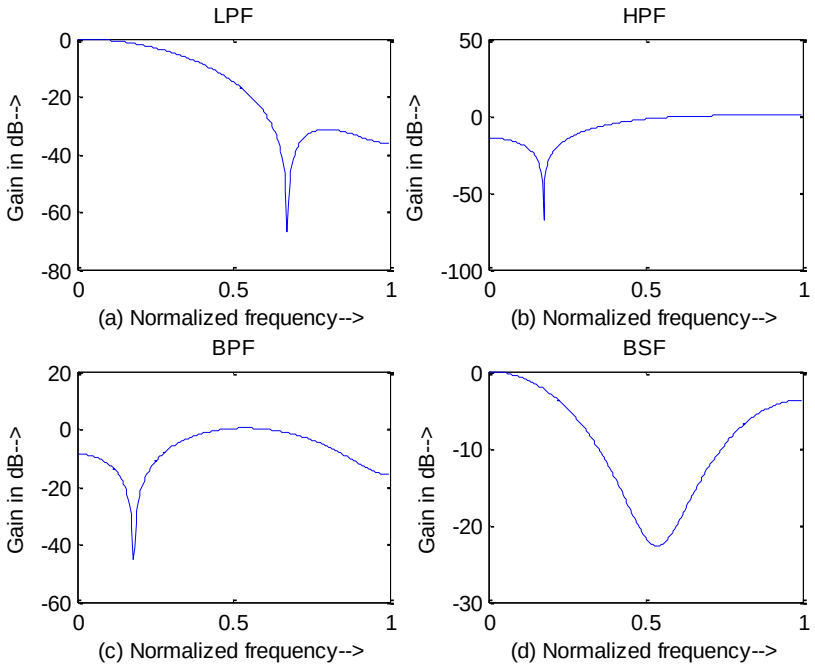
enter stopband freq2000

enter sampling freq 5000

enter your choice of window function 1. rectangular 2. triangular 3.kaiser:

1

Rectangular window filter response



enter passband ripple0.04

enter the stopband ripple0.06

enter passband freq1000

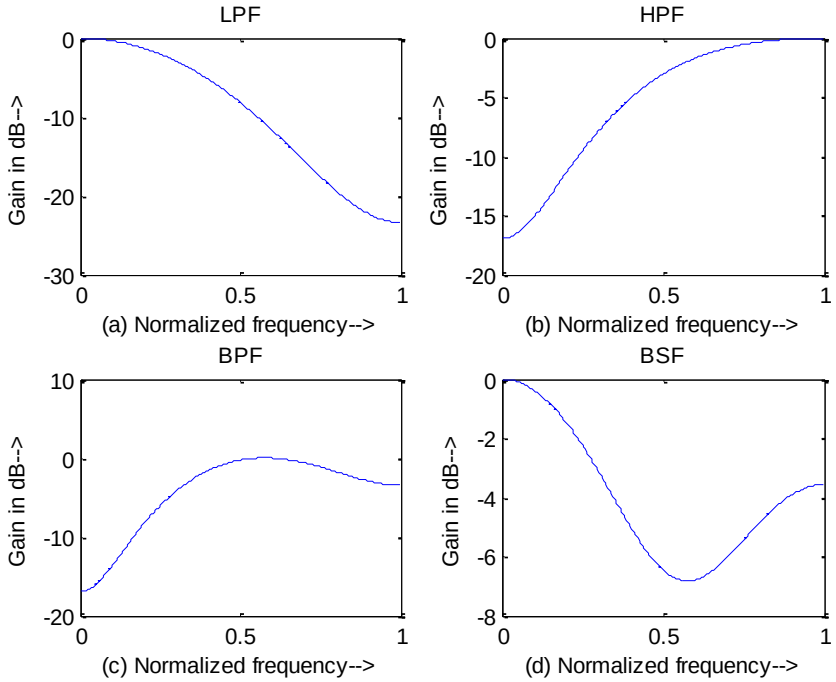
enter stopband freq2000

enter sampling freq 5000

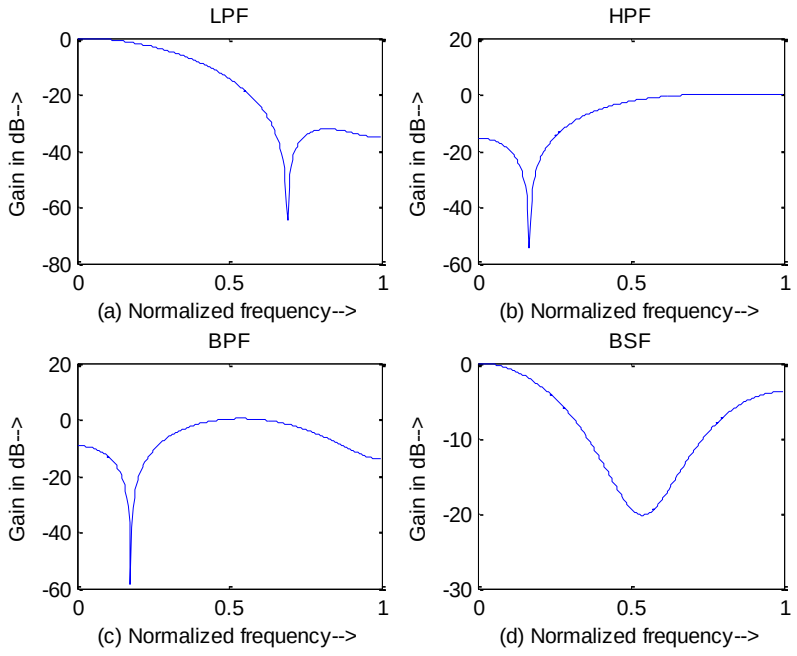
enter your choice of window function 1. rectangular 2. triangular 3.kaiser:

2

Triangular window filter response



enter passband ripple0.04
enter the stopband ripple0.06
enter passband freq1000
enter stopband freq2000
enter sampling freq 5000
enter your choice of window function 1. rectangular 2. triangular 3.kaiser:
3
kaiser window filter response



RESULT: The frequency responses of Low pass and High pass FIR filter using Rectangular, Triangular & Kaiser windows are plotted.