# 7. FIR FILTER (LP/HP) USING KAISER WINDOW TECHNIQUE

**AIM:**

To design a FIR filter using Kaiser windowing technique and verify its frequency response.

**APPARATUS:**

PC with MATLAB

**THEORY:**

The window method for a causal linear-phase FIR filter is obtained by multipying an ideal filter that has an infinite-duration impulse response (IIR) by a finite-duration window function:

$$h[n] = h_d[n]w[n]$$

where h[n] is the practical FIR filter,$h_d[n]$ is the ideal IIR prototype filter, and w[n] is the finite-duration window function. An important consequence of this operation is that the DTFTs of $h_d[n]$ and w[n] undergo circular convolution in frequency

$$H(e^{j\omega}) = \frac{1}{2\pi}\oint_{2\pi} H_d(e^{j\theta})W(e^{j(\omega-\theta)})d\theta$$

**Kaiser Window:**

A very flexible family of window functions has been developed by Kaiser. These windows are nearly optimum in the sense of having the largest energy in the mainlobe for a given peak sidelobe level.

They are closely related to the prolate spheroidal wavefunctions which are the optimum time-limited, continuous-time functions in a similar sense. The Kaiser windows are of the form,

$$w_K(n) = \frac{I_0[\beta\sqrt{1-(1-2n/M)^2}]}{I_0[\beta]}, \qquad n = 0, 1, \ldots, M,$$

where $I_0$[ ] is the modified zeroth-order Bessel function of the first kind and ß is a shape parameter determining the tradeoff between the mainlobe width and the peak sidelobe level. Typical values for ß are in the range 4 <ß < 9. $I_0$[ ] is most easily computed from its power series expansion

$$I_0[x] = 1 + \sum_{m=1}^{\infty}\left[\frac{(x/2)^m}{m!}\right]^2,$$

**PROCEDURE:-**

- Open MATLAB
- Open new M-file
- Type the program
- Save in current directory
- Compile and Run the program
- For the output see command window\ Figure window

**PROGRAM:**

```
% To design of FIR filters using Kaiser window techniques.

clc;
```

```
clear all;
close all;

% Low Pass Filter
n=100;
fp=200;
fq=300;
fs=1000;
fn=2*fp/fs;
beta = input('Enter beta value ') % use 0.5 to 2.5
window=kaiser((n+1),beta);
b=fir1(n,fn,window);
[H W]=freqz(b,1,128);
subplot(2,1,1);
plot(W/pi,abs(H));
title('magnitude response of lpf');
ylabel('gain in db-------->');
xlabel('normalized frequency------>');
grid on
subplot(2,1,2);
plot(W/pi,angle(H));
title('phase response of lpf');
ylabel('angle-------->');
xlabel('normalized frequency------>');
grid on

% Highpass Filter
n=100;
fp=300;
fq=200;
fs=1000;
fn=2*fp/fs;
window=kaiser((n+1),beta);
b=fir1(n,fn,'high',window);
[H W]=freqz(b,1,128);
figure(2)
subplot(2,1,1);
plot(W/pi,abs(H));
title('mag res of hpf');
ylabel('gain in db-------->');
xlabel('normalized frequency------>');
grid on
subplot(2,1,2);
```

```
plot(W/pi,angle(H));
title('phase res of hpf');
ylabel('angle--------->');
xlabel('normalized frequency------>');
grid on
```

**INPUT:**:

enter the passband ripple:  0.02

enter the stopband ripple:  0.01

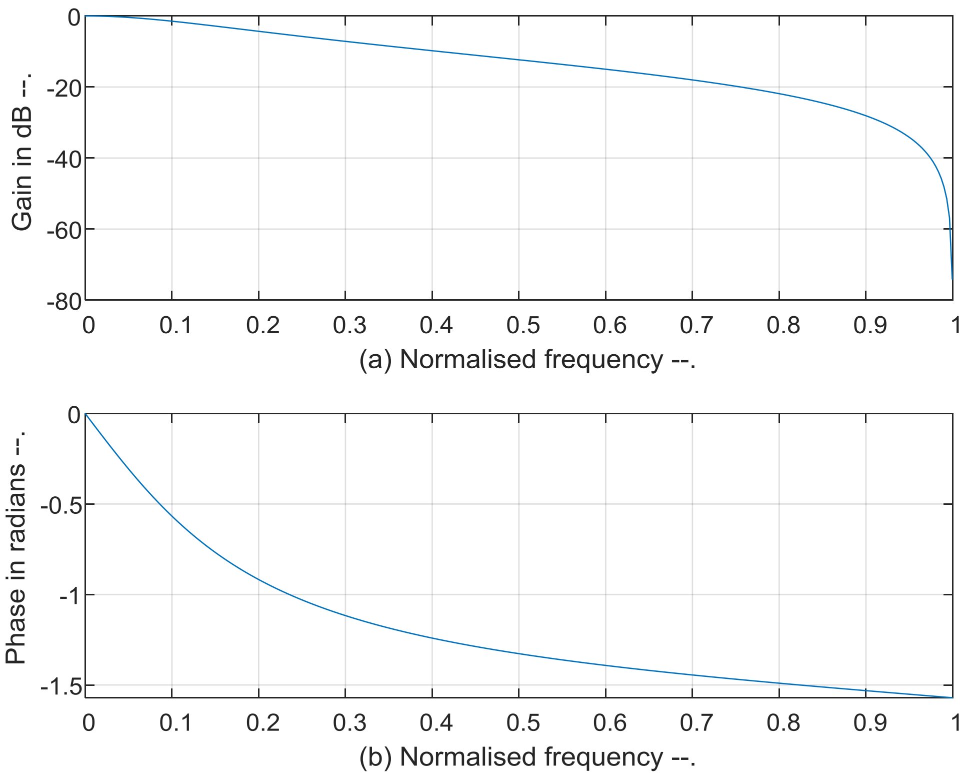
enter the passband frequency:  1000

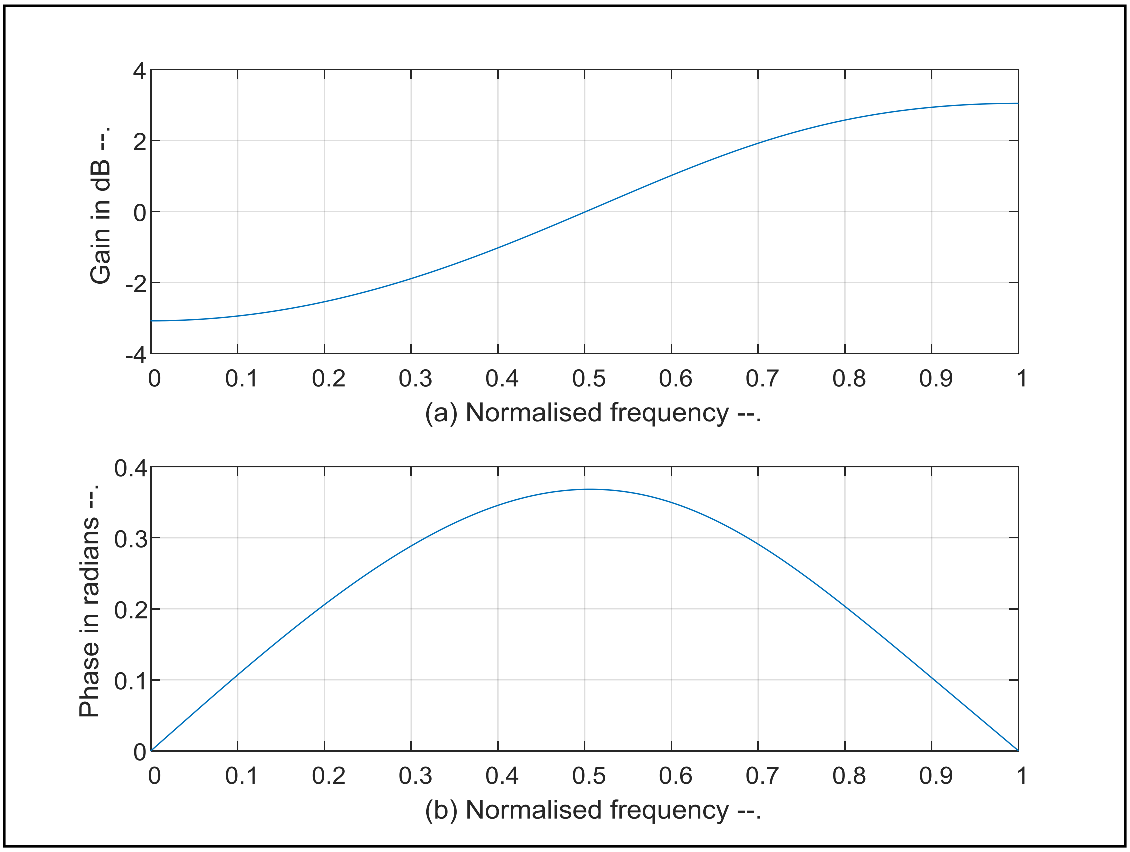
enter the stopband frequency:  1500

enter the sampling frequency:  10000

enter the beta value:  1.2

**OUTPUT:**






**RESULT:**

Thus the MATLAB program for FIR LP\HP using Triangular window Techniques was executed and its frequency response is also verified.