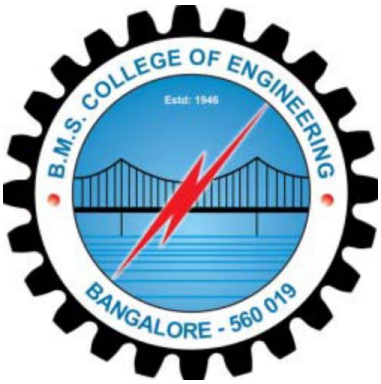
B.M.S. COLLEGE OF ENGINEERING

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum)

Bull Temple Road, Basavanagudi, Bangalore-560019



REPORT

ON

"DIGITAL SIGNAL PROCESSING LAB REPORT"

Submitted in partial fulfilment of the requirements for the partial completion of
DIGITAL SIGNAL PROCESSING
(16EC5DCDSP)

SUBMITTED BY:

AKSHAY S RAO **1BM17EC007**

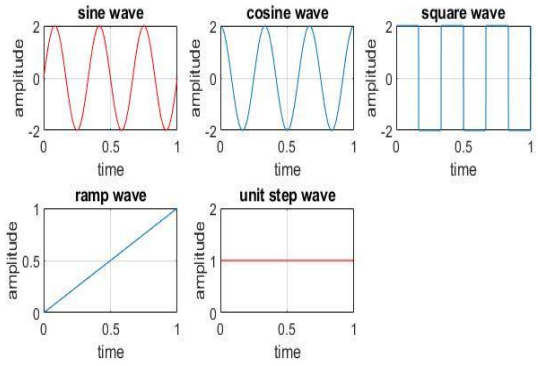
Course instructor
LALITHA S
Assistant Professor, Dept. of ECE

Aug-Dec 2019

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

Generation of Basic Continuous signals

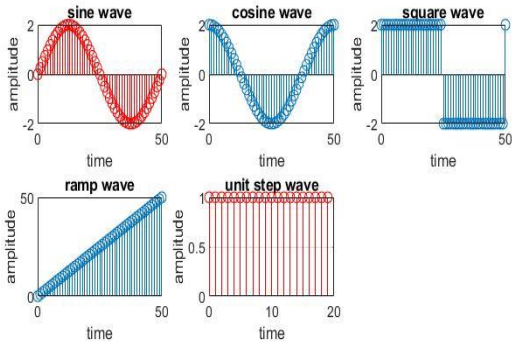
```
clc;
clear all;
close all;
t=0:.001:1;
f=input('Enter the value of frequency');
a=input('Enter the value of amplitude');
subplot(3,3,1);
y=a*sin(2*pi*f*t);
plot(t,y,'r');
xlabel('time');
ylabel('amplitude');
title('sine wave'); grid on;
subplot(3,3,2);
z=a*cos(2*pi*f*t);
plot(t,z);
xlabel('time');
ylabel('amplitude');
title('cosine wave'); grid on;
subplot(3,3,3);
s=a*square(2*pi*f*t);
plot(t,s);
xlabel('time'); ylabel('amplitude');
title('square wave'); grid on;
subplot(3,3,4);
plot(t,t);
xlabel('time'); ylabel('amplitude')
title('ramp wave'); grid on;
subplot(3,3,5);
A = ones(1,(length(t)));
plot(t,A,'r');
xlabel('time'); ylabel('amplitude');
title('unit step wave'); grid on;
```




```
Input:
Enter the value of frequency 3
Enter the value of amplitude 2
```

Generation of Basic discrete signals

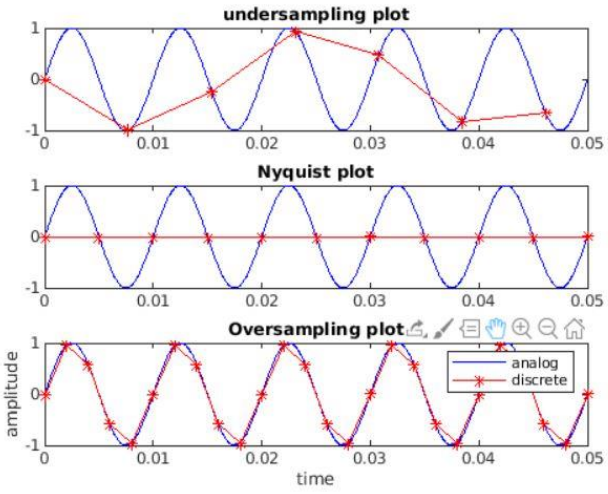
```
clc;
clear all;
close all;
n=0:1:50;
f=input('Enter the value of frequency');
a=input('Enter the value of amplitude');
N=input('Enter the length of unit step');
subplot(3,3,1);
y=a*sin(2*pi*f*n);
stem(n,y,'r');
xlabel('time');
ylabel('amplitude');
title('sine wave'); grid on;
subplot(3,3,2);
z=a*cos(2*pi*f*n);
stem(n,z);
xlabel('time');
ylabel('amplitude');
title('cosine wave');
grid on;
subplot(3,3,3);
s=a*square(2*pi*f*n);
stem(n,s);
xlabel('time');
ylabel('amplitude');
title('square wave'); grid on;
subplot(3,3,4);
stem(n,n);
xlabel('time');
ylabel('amplitude');
title('ramp wave'); grid on;
x=0:N-1; d=ones(1,N);
subplot(3,3,5);
stem(x,d,'r');
xlabel('time');
ylabel('amplitude');
title('unit step wave'); grid on;
Input:
Enter the value of frequency 1/50
Enter the value of amplitude 2
Enter the length of unit step 20
```



Verification of Sampling theorem

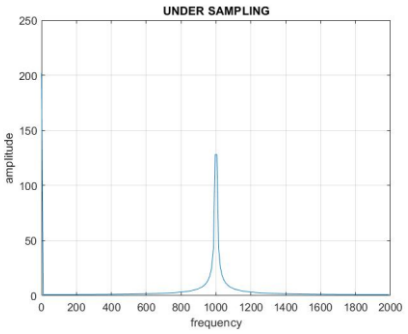
CODE:

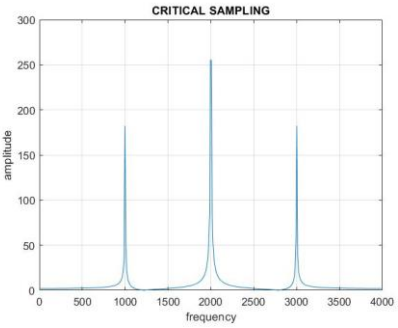
```
clc;
clear all;
tfinal = 0.05;
t  = 0:0.00005:tfinal;
fd = input('Enter analog freuency : ');
%define analog signal for comparison
xt = sin(2*pi*fd*t);

%simulate condition for undersampling i.e., fs1<2*fd
fs1 = 1.3*fd;

%define the time vector
n1 = 0:1/fs1:tfinal;

%Generate the undersampled signal
xn=sin(2*pi*n1*fd);

%plot the analog & sampled signals
subplot(3,1,1);
plot(t,xt,'b',n1,xn,'r*-');
title('undersampling plot');

%condition for Nyquist plot
fs2=2*fd;
n2=0:1/fs2:tfinal;
xn=sin(2*pi*fd*n2);
subplot(3,1,2);
plot(t,xt,'b',n2,xn,'r*-');
title('Nyquist plot');

%condition for oversampling
fs3=5*fd;
n3=0:1/fs3:tfinal;
xn=sin(2*pi*fd*n3);
subplot(3,1,3);
plot(t,xt,'b',n3,xn,'r*-');
title('Oversampling plot');
xlabel('time');
ylabel('amplitude');
legend('analog','discrete')
```

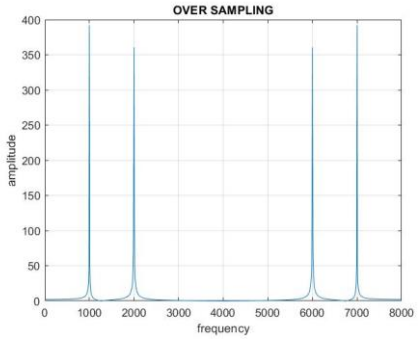


Verification of sampling theorem

```
f1 = input ('Enter the highest Frequency of the Signal F1 in Hz =
');
f2 = input ('Enter the highest Frequency of the Signal F2 in Hz =
');
freq = 2 * max(f1,f2);
% Under Sampling
fs = freq/2;
t = [0:1/fs:0.1];
x = cos(2*pi*f1*t) + cos(2*pi*f2*t);
XK = fft(x);
f = [0:length(XK)-1]*fs/length(XK);
figure(1);
plot(f,abs(XK));
xlabel('frequency'); ylabel('amplitude');
grid  on; title('UNDER SAMPLING');
%Critical Sampling
fs = freq;
t = [0:1/fs:0.1];
x = cos(2*pi*f1*t) + cos(2*pi*f2*t);
XK = fft(x);
f = [0:length(XK)-1]*fs/length(XK);
figure(2);
plot(f,abs(XK));
xlabel('frequency');
ylabel('amplitude');
grid  on;
title('CRITICAL SAMPLING');
%Over Sampling
fs = 2*freq;
t = [0:1/fs:0.1];
x = cos(2*pi*f1*t) + cos(2*pi*f2*t);
XK = fft(x);
f = [0:length(XK)-1]*fs/length(XK);
figure(3);
plot(f,abs(XK));
xlabel('frequency'); ylabel('amplitude');
grid  on; title('OVER SAMPLING');
```



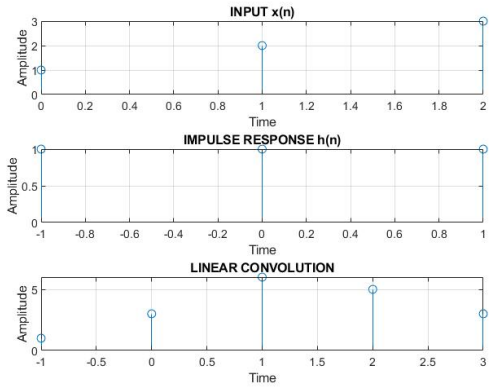





```
Input:
Enter the highest Frequency of the Signal F1 in Hz = 1000
Enter the highest Frequency of the Signal F2 in Hz = 2000
```

LINEAR CONVOLUTION USING MATLAB

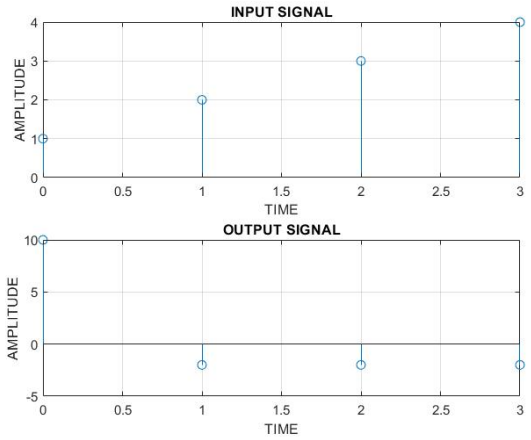
```
clc;
clear all;
close all;
a=input('Enter the starting point of x[n]=');
b=input('Enter the starting point of h[n]=');
x=input('Enter the co-efficients of x[n]=');
h=input('Enter the co-efficients of h[n]=');
y=conv(x,h);
subplot(3,1,1);
p=a:(a+length(x)-1);
stem(p,x); grid on;
xlabel('Time'); ylabel('Amplitude');
title('INPUT x(n)');
subplot(3,1,2);
q=b:(b+length(h)-1);
stem(q,h); grid on;
xlabel('Time');
ylabel('Amplitude');
title('IMPULSE RESPONSE h(n)');
subplot(3,1,3);
n=a+b:length(y)+a+b-1;
stem(n,y); grid on;
disp(y); xlabel('Time');
ylabel('Amplitude'); title('LINEAR CONVOLUTION');
```




```
Input:
Enter the starting point of x[n] = 0
Enter the starting point of h[n] = -1
Enter the co-efficients of x[n] = [1 2 3]
Enter the co-efficients of h[n] = [1 1 1]
     1     3     6     5     3
```

Discrete Fourier transform.

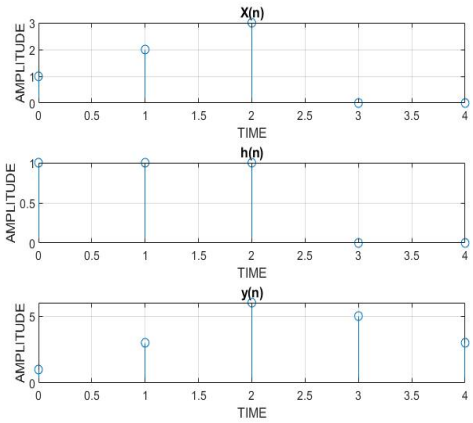
```
clc;
clear all;
close all;
N=input('Enter the value of N');
x=input('Enter the input sequence X(n):');
t=0:N-1;
subplot(2,1,1);
stem(t,x);
xlabel('TIME');
ylabel('AMPLITUDE');
title('INPUT SIGNAL'); grid on;
y=fft(x,N);
subplot(2,1,2);
stem(t,y);
xlabel('TIME');
ylabel('AMPLITUDE');
title('OUTPUT SIGNAL'); grid on;
```




```
Input:
Enter the value of N 4
Enter the input sequence X(n):[1 2 3 4]
```

Linear convolution using Fourier transform

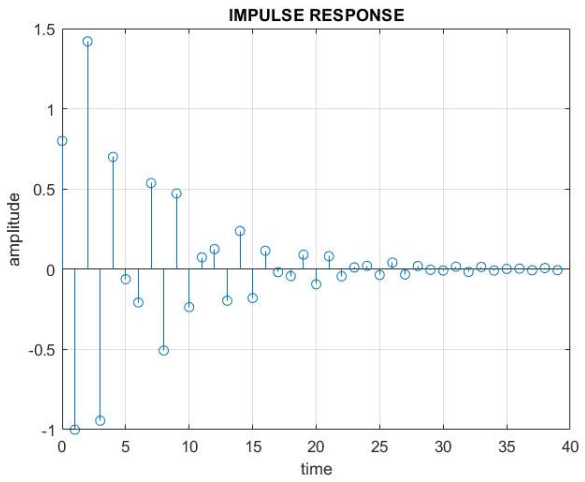
```
clc;
clear all; close all;
x=input('Enter the sequence x(n):');
h=input('Enter the sequence h(n):');
l1=length(x); l2=length(h);
x=[x,zeros(1,l2-1)];
h=[h,zeros(1,l1-1)];
l3=length(x);
c=0:l3-1;
subplot(3,1,1);
stem(c,x); grid on;
xlabel('TIME'); ylabel('AMPLITUDE');
title('X(n)');
subplot(3,1,2);
stem(c,h); xlabel('TIME');
ylabel('AMPLITUDE');
grid on;
title('h(n)');
q=fft(x,l3);r=fft(h,l3);
```



```
s=(r).*q;
y=ifft(s,l3);
subplot(3,1,3); stem(c,y);
xlabel('TIME'); ylabel('AMPLITUDE');
grid on; title('y(n)');

Input:
Enter the sequence x(n):[1 2 3]
Enter the sequence h(n):[1 1 1]
```

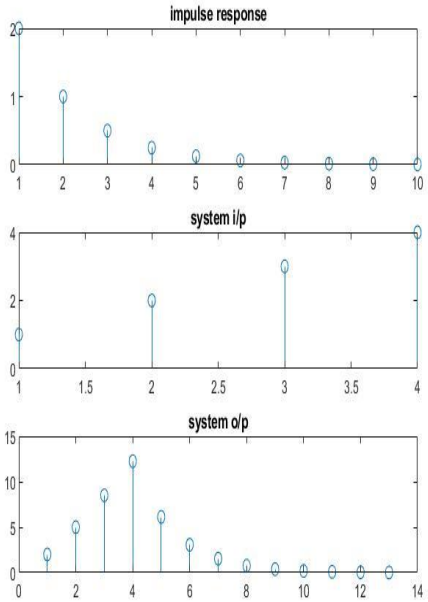
Impulse response of an LTI system

```
clc; clear all; close all;
N=input('Enter the required length of impulse response N=');
n=0:N-1;
b=input('Enter the co-efficients of x(n),b=');
a=input('Enter the co=efficients of y(n),a=');
x=[1,zeros(1,N-1)];
y=filter(b,a,x);
stem(n,y);
xlabel('time'); ylabel('amplitude');
title('IMPULSE RESPONSE');
grid on;
Input:
Enter the required length of impulse response N=40
Enter the co-efficients of x(n),b= [0.8 -0.44 0.36 0.02]
Enter the co-efficients of y(n),a=  [1 0.7 -0.45 -0.6]
```



LINEAR CONSTANT COEFFICIENT DIFFERENCE EQUATION

```
clc;
clear all;close all;
a=input('enter the co-efficient of x');
b=input('enter the co-efficient of y');
% to find h(z)
[r,p,k]=residuez(a,b);
disp('residues are=');disp(r);
disp('poles of h(z) are=');disp(p);
disp('constant of h(z) are=');disp(k);
rr=roots(a);

% to find the impulse response
n=input('enter the length of the response');
x=[1,zeros(1,n-1)];
h=filter(a,b,x);
disp('impulse response is');
subplot(3,1,1); grid on;
stem(h);
title('impulse response');
% to find system response
x=input('enter the signal');
y=conv(x,h);
disp('output of the system is' );
disp(y);
subplot(3,1,2);
grid on;
stem(x);
title('system i/p');
subplot(3,1,3); grid on;
stem(y); title('system o/p');
```



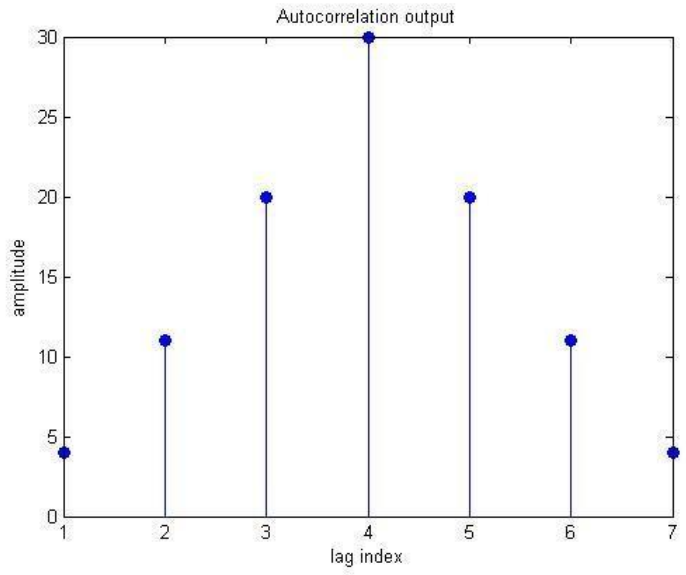

```
Input:
Enter the co-efficient of x[2]
Enter the co-efficient of y[1 -0.5]
Residues are = 2
poles of h(z) are= 0.5000
enter the length of the response 10
impulse response is
enter the signal[1 2 3 4]
output of the system is
```

```
Columns 1 through 9

    2.0000    5.0000    8.5000   12.2500    6.1250    3.0625    1.5313
0.7656    0.3828

  Columns 10 through 13

    0.1914    0.0938    0.0430    0.0156
```

Auto co-relation

```
x=input ('Enter sequence x(n)=');
rxx= conv(x,fliplr(x));
disp('rxx=');
disp(rxx);
figure(1);
stem(rxx,'filled');
title('Autocorrelation output');
xlabel('lag index');
Input:
Enter sequence x(n)=[1 2 3 4]
```

Rxx = 4 11 20 30 20 11 4

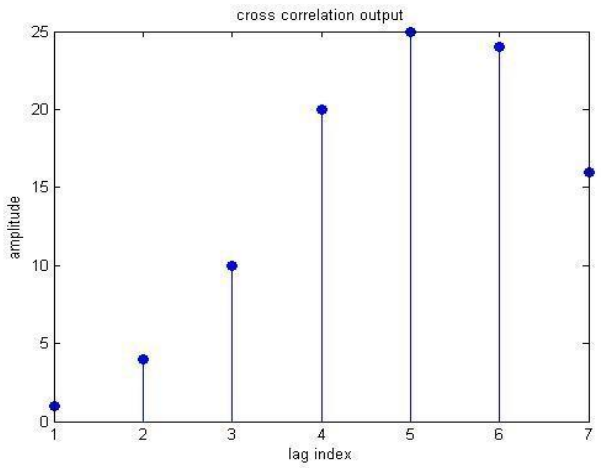


Cross correlation

```
x=input ('Enter sequence x(n)=');
y=input ('Enter sequence y(n)=');
rxy= conv(x,fliplr(y)); disp('rxy=');
disp(rxy);
figure(1);
stem(rxy,'filled');
title('cross correlation output');
xlabel('lag index'); ylabel('amplitude');
```

Input:

```
Enter sequence x(n) = [1 2 3 4]
Enter sequence y(n) = [4 3 2 1]
Rxy = 1 4   10  20  25  24  16
```

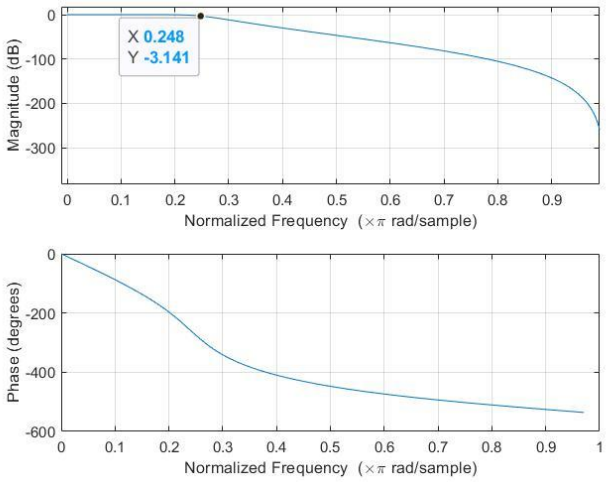


CODE FOR BUTTERWORTH LOW PASS:

```
clc;
close all;
clear all;
passband_attenuation = input ('enter the passband attenuation:');
stopband_attenuation = input('enter the stop band attenuation:');
passband_frequency   = input('enter the pass band frequency:');
stopband_frequency   =  input('enter the stop band frequency:');

[N,Wn] = buttord(passband_frequency, stopband_frequency...
           ,passband_attenuation, stopband_attenuation );
[b,a] = butter(N,Wn);
freqz(b,a);
```

Input:

```
enter the passband attenuation:0.4
enter the stop band attenuation:30
enter the pass band frequency:0.2
enter the stop band frequency:0.4

>> [N,Wn]

ans =

    6.0000    0.2469
```

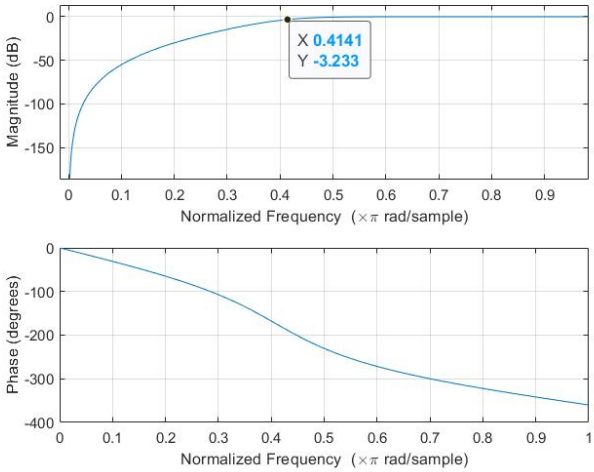


CODE FOR BUTTERWORTH HIGH PASS:

```
clc;
close all;
clear all;
passband_attenuation = input ('Enter the pass band
attenuation:');
stopband_attenuation = input ('Enter the pass band
attenuation:');
passband_frequency   = input ('Enter the pass band frequency:');
stopband_frequency   = input ('Enter the stop band frequency:');
[N,wn] = buttord(passband_frequency, stopband_frequency,....
         passband_attenuation, stopband_attenuation);
[b,a] = butter(N, wn, 'high');
freqz(b,a);
```

Input:

```
Enter the pass band attenuation:0.4
Enter the pass band attenuation:30
Enter the pass band frequency:0.6
Enter the stop band frequency:0.2
>> [N,wn]

ans =

    4.0000    0.4179
```

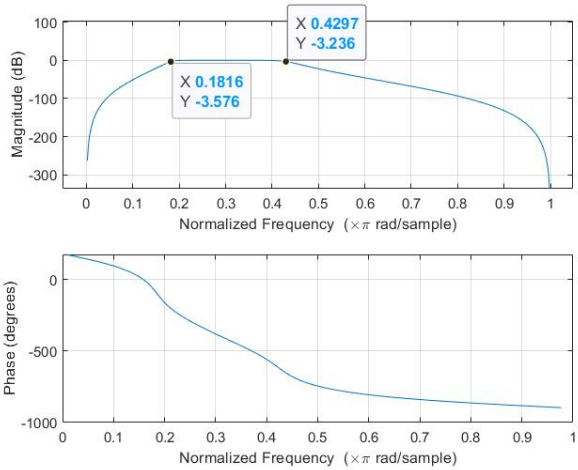


CODE FOR BUTTERWORTH BANDPASS:

```
clc;
close all;
clear all;
passband_attenuation=input('enter the passband attenuation:');
stopband_attenuation=input('enter the stop band attenuation:');
passband_frequency=input('enter the pass band frequency:');
stopband_frequency=input('enter the stop band frequency:');
[N,wn]=buttord(passband_frequency, stopband_frequency,...
        passband_attenuation, stopband_attenuation);
[b,a]=butter(N,wn);
freqz(b,a)
```

Input:

```
enter the passband attenuation:0.3
enter the stop band attenuation:22
enter the pass band frequency:[0.2,
0.4]
enter the stop band frequency:[0.1,
0.5]
>> [N,wn]

ans =

    6.0000    0.1833    0.4285
```

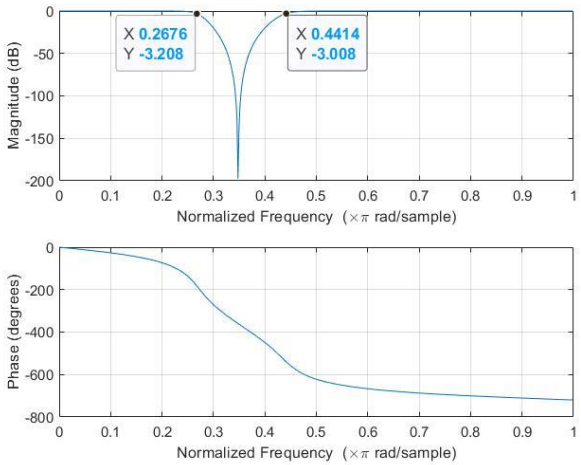


CODE FOR BUTTERWORTH BAND STOP:

```
clc;
close all;
clear all;
passband_attenuation = input('enter the passband attenuation:');
stopband_attenuation = input('enter the stop band attenuation:');
passband_frequency   = input('enter the pass band frequency:');
stopband_frequency   = input('enter the stop band frequency:');
[N,wn] = buttord(passband_frequency, stopband_frequency,...
            passband_attenuation, stopband_attenuation);
[b,a]  = butter(N,wn,'stop');
freqz(b,a);
```

Input:

```
enter the passband attenuation:0.5
enter the stop band attenuation:20
enter the pass band frequency:[0.2,
0.5]
enter the stop band frequency:[0.3,
0.4]
>> [N,wn]

ans =

    4.0000    0.2668    0.4414
```

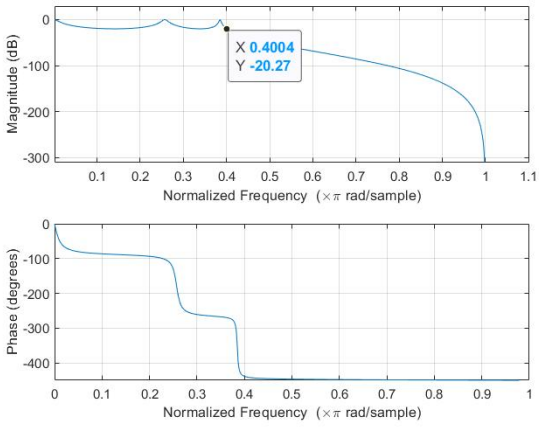


CODE FOR CHEBYSHEV LOW PASS:

```
clc;
close all;
clear all;
passband_attenuation = input('enter the passband attenuation:');
stopband_attenuation = input('enter the stop band attenuation:');
passband_frequency   = input('enter the pass band frequency:');
stopband_frequency   = input('enter the stop band frequency:');
[N,wn] = cheb1ord(passband_frequency, stopband_frequency,...
            passband_attenuation, stopband_attenuation);
[b,a]=cheby1(N,passband_attenuation,wn);
freqz(b,a);
```

Input:

```
enter the passband attenuation:30
enter the stop band attenuation:50
enter the pass band frequency:0.4
enter the stop band frequency:0.3
>> [N,wn]

ans =

    4.0000    0.4000
```

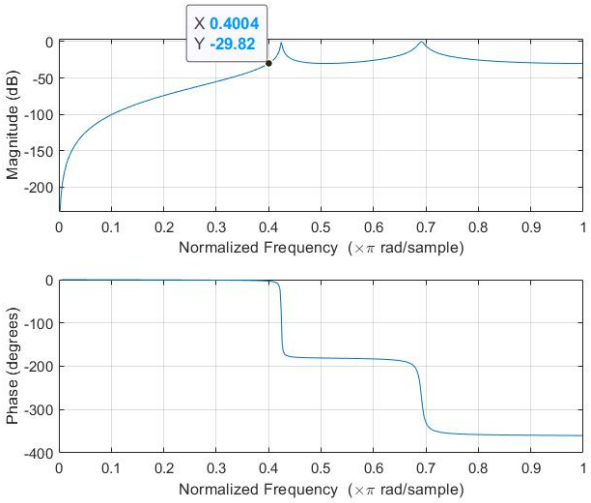


CODE FOR CHEBYSHEV HIGH PASS:

```
clc;
close all;
clear all;
passband_attenuation = input('enter the passband attenuation:');
stopband_attenuation = input('enter the stop band attenuation:');
passband_frequency   = input('enter the pass band frequency:');
stopband_frequency   = input('enter the stop band frequency:');
[N,wn] = cheb1ord(passband_frequency, stopband_frequency,...
            passband_attenuation, stopband_attenuation);
[b,a]=cheby1(N, passband_attenuation, wn, 'high');
freqz(b,a);
```

Input:

```
enter the passband attenuation:30
enter the stop band attenuation:50
enter the pass band frequency:0.4
enter the stop band frequency:0.3
>> [N,wn]

ans =

    4.0000    0.4000
```

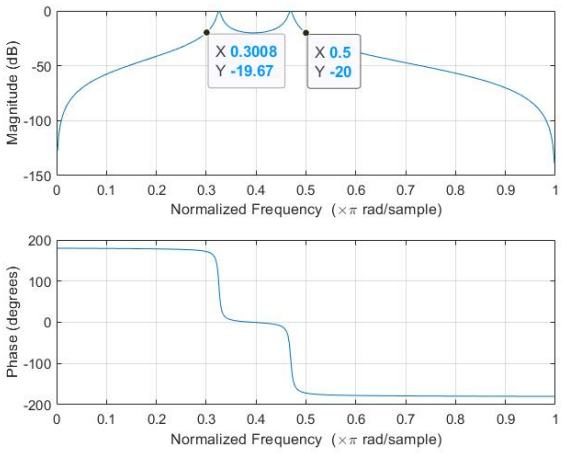


CODE FOR CHEBYSHEV BAND PASS:

```
clc;
close all;
clear all;
passband_attenuation = input('enter the passband attenuation:');
stopband_attenuation = input('enter the stop band attenuation:');
passband_frequency   = input('enter the pass band frequency:');
stopband_frequency   = input('enter the stop band frequency:');
[N,wn] = cheb1ord(passband_frequency, stopband_frequency,...
            passband_attenuation, stopband_attenuation);
[b,a]=cheby1(N, passband_attenuation, wn);
freqz(b,a);
```

Input:

```
enter the passband attenuation:20
enter the stop band attenuation:50
enter the pass band frequency:[0.3,
0.5]
enter the stop band frequency:[0.1,
0.8]
>> [N,wn]

ans =

    2.0000    0.3000    0.5000
```

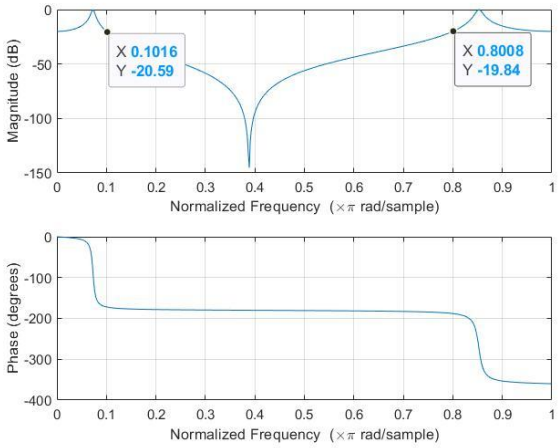


CODE FOR CHEBYSHEV BAND STOP:

```
clc;
close all;
clear all;
passband_attenuation = input('enter the passband attenuation:');
stopband_attenuation = input('enter the stop band attenuation:');
passband_frequency   = input('enter the pass band frequency:');
stopband_frequency   = input('enter the stop band frequency:');
[N,wn] = cheb1ord(passband_frequency, stopband_frequency,...
            passband_attenuation, stopband_attenuation);
[b,a]=cheby1(N, passband_attenuation, wn, 'stop');
freqz(b,a);
```

Input:

```
enter the passband attenuation:20
enter the stop band attenuation:50
enter the pass band frequency:[0.1 0.8]
enter the stop band frequency:[0.3,
0.5]
>> [N,wn]

ans =

    2.0000    0.1000    0.8000
```

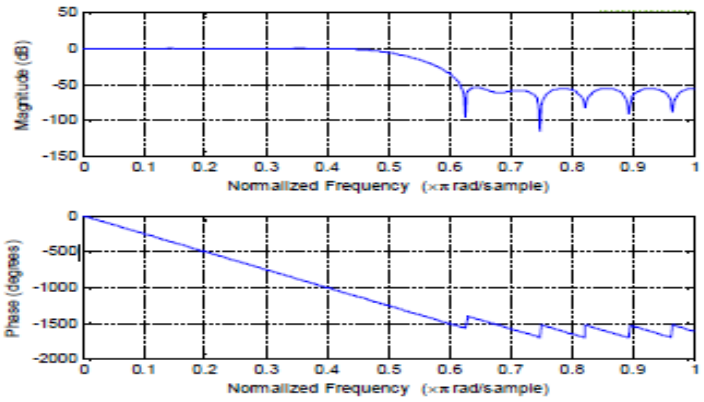


CODE FOR HAMMING LOW PASS:

```
clc;
clear all;
close all;
N  = input ('Enter the order of the filter: ');
cutoff_frequency = input ('Enter the cutoff frequency: ');
H  = fir1(N, cutoff_frequency, hamming(N+1));
freqz(H);
```

```
Enter the order of the filter: 28
Enter the cutoff frequency: 0.5
```

FIGURE:-

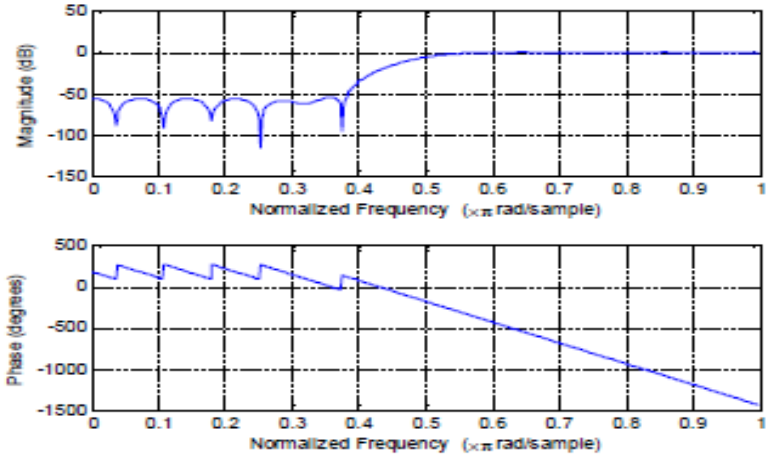


CODE FOR HAMMING HIGH PASS

```
clc;
clear all;
close all;
N  = input ('Enter the order of the filter: ');
cutoff_frequency = input ('Enter the cutoff frequency: ');
H  = fir1(N, cutoff_frequency, 'high', hamming(N+1));
freqz(H);
```

```
Enter the order of the filter: 28
Enter the cutoff frequency: 0.5
```

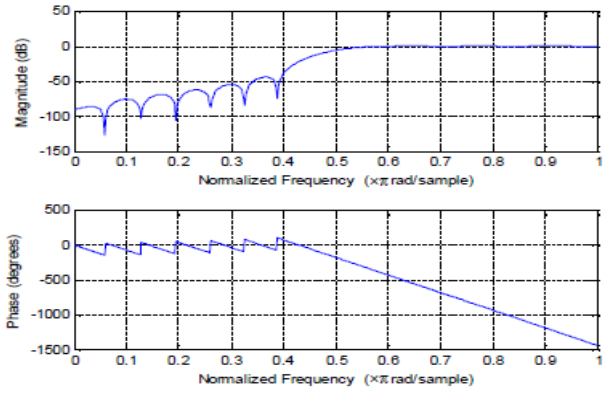
FIGURE:-



CODE FOR HANNING HIGH PASS

```
clc;
clear all;
close all;
N  = input ('Enter the order of the filter: ');
cutoff_frequency = input ('Enter the cutoff frequency: ');
H  = fir1(N, cutoff_frequency, 'high', hanning(N+1));
freqz(H);
```

```
Enter the order of the filter: 28
Enter the cutoff frequency: 0.5
```



CODE FOR HANNING LOW PASS

```
clc;
clear all;
close all;
N  = input ('Enter the order of the filter: ');
cutoff_frequency = input ('Enter the cutoff frequency: ');
H  = fir1(N, cutoff_frequency, hanning(N+1));
freqz(H);

Enter the order of the filter: 10
Enter the cutoff frequency: 0.6
```