**DSP lab Assignment 4**

1. Verification of time invariant system

2. Verification of causality and non-causality

3. Verification of Linearity and non-Linearity

4. Verification of Stability and unstability

5. program for Discrete Time Fourier Transform

6. Program for the design of butterworth high pass analog filter

7. Program for the design of Butterworth band pass analog filter

8. program for the design of Butterworth band pass analog filter

9. program for the design of chebyshev low pass filter

10. program for the design of chebyshev high pass filter

11. program for the design of chebyshev band pass filter

12. program for the design of chebyshev band stop filter

Solution 1: **Verification of time invariant system**

**code:**

x1=input('enter input sequence x1:');

a=input('enter a scaling a:');

n0=input('enter shift n0:');

x2=[zeros(1,n0),x1];

y1=a.\*x1;

y2=a.\*x2;

y3=[zeros(1,n0),x1];

if(y2==y3)

display(' the system is time variant');

else

display(' the system is time variant');

end;

subplot(2,2,1);

stem(x1);title('input signal');

subplot(2,2,2);

stem(x2);title(' signal after shifting');

subplot(2,2,3);

stem(y2);title(' L.H.S');

subplot(2,2,4);

stem(y1);title(' R.H.S');

Solution 2: **Verification of causality and non-causality**

**code:**

x1=input('enter input sequence x1:');

n1=input('enter lower limit n1:');

n2=input('enter higher limit n2:');

Flag=0;

for n=n1:n2

arg=n;

if arg>=0;

Flag=1;

end;

end;

y1=x1(n)+x1(n+1);

if(flag==1)

display(' the system is causal');

else

display(' the system is non\_causal');

end;

subplot(2,2,1);

stem(x1);title(' Input signal');

xlabel('Time');ylabel('Amplitude');

subplot(2,2,2);

stem(y1);title('Output signal');

xlabel('Time');ylabel('Amplitude');

Solution 3: **Verification of Linearity and non-Linearity**

**code:**

x1=input('enter first input signal x1:');

x2=input('enter second input signal x2:');

a=input('enter first scaling constant a:');

b=input('enter second scaling constant:');

subplot(2,2,1);

stem(x1);title('first input signal');

xlabel('Time');ylabel('Amplitude');

subplot(2,2,2);

stem(x2);title('Second input signal ');

xlabel('Time');ylabel('Amplitude');

y1=x1;

y2=x2;

RHS=a\*y1+a\*y2;

x3=a\*(x1+x2);

LHS=x3;

if(RHS==LHS)

display('The system is Linear.');

else

display('The system non\_Linear.');

end;

subplot(2,2,3);

stem(LHS);title(' L.H.S');

xlabel('Time');ylabel('Amplitude');

subplot(2,2,4);

stem(RHS);title(' R.H.S');

xlabel('Time');ylabel('Amplitude');

Solution 4: **Verification of Stability and unstability**

**code:**

disp('Stability')

nr=input ('enter the numerator co\_efficient:');

dr=input ('enter the denomintor co\_efficient:');

z=tf(nr,dr,1);

[r,p,k]=residuez(nr,dr);

figure

zplane(nr,dr);

if abs(p)<1

disp('system is stable');

else

disp('system is unstable');

end;

Solution 5: **program for Discrete Time Fourier Transform**

**Matlab code:**

clc;

clear all;

close all;

a=input('Enter the sequence :');

N=length(a);

disp('The length of the sequence is:');

for k=1:N

y(k)=0;

for i=1:N

y(k)=y(k)+a(i)\*exp((-2\*pi\*j/N)\*((i-1)\*(k-1)));

end;

end;

k=1:N

disp('The result is:');y

figure(1);

subplot(211);

stem(k,abs(y(k)));

grid;

xlabel('sample values n-->');

ylabel('Amplitudes-->');

title('Magnitude response of the DFT of given sequence');

subplot(212);

stem(angle(y(k))\*180/pi);

grid;

xlabel('sample values n-->');

ylabel('phase-->');

title('Phase response of the DFT of given sequence');

Solution 6: **Program for the design of Butterworth low pass analog filter.**

**Matlab Code:**

clc;

close all;

clear all;

format long

rs=input('enter the stop band ripple...');

rp=input('enter the pass band ripple...');

ws=input('enter the stop band freq...');

wp=input('enter the passband freq...');

fs=input('enter the sampling freq...');

w1=2\*wp/fs;

w2=2\*ws/fs;

[n,wn]=buttord(w1,w2,rp,rs);

[z,p,k]=butter(n,wn);

[b,a]=zp2tf(z,p,k);

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

w=0:0.01:pi;

[h,om]=freqs(b,a,w);

m=20\*log10(abs(h));

an=angle(h);

subplot(2,1,1);

plot(om/pi,m);

ylabel('GainindB-->');

xlabel('(a)Normalised frequency-->');

subplot(2,1,2);

plot(om/pi,an)

xlabel('(b)Normalised frequency-->');

ylabel('Phase in radians-->');

Solution 7: **Program for the design of butterworth high pass analog filter**

**Matlab Code:**

clc;

close all;

clear all;

format long

rp=input('enter the passband ripple...');

rs=input('enter the stopband ripple...');

wp=input('enter the passband freq.....');

ws=input('enter the stopband freq.....');

fs=input('enter the sampling freq.....');

w1=2\*wp/fs;

w2=2\*ws/fs;

[n,wn]=buttord(w1,w2,rp,rs,'s');

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

w=0:0.01:pi;

[h,om]=freqs(b,a,w);

m=20\*log10(abs(h));

an=angle(h);

subplot(2,1,1);

plot(om/pi,m);

ylabel('GainindB-->');

xlabel('(a)Normalised frequency-->');

subplot(2,1,2);

plot(om/pi,an)

xlabel('(b)Normalised frequency-->');

ylabel('Phase in radians-->');

solution 8: **Program for the design of Butterworth band pass analog filter**

**Matlab Code:**

clc;

close all;

clear all;

format long

rp=input('enter the passband ripple...');

rs=input('enter the stopband ripple...');

wp=input('enter the passband freq...');

ws=input('enter the stopband freq...');

fs=input('enter the sampling freq...');

w1=2\*wp/fs;

w2=2\*ws/fs;

[n]=buttord(w1,w2,rp,rs,'s');

wn=[w1 w2];

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

w=0:0.01:pi;

[h,om]=freqs(b,a,w);

m=20\*log10(abs(h));

an=angle(h);

subplot(2,1,1);

plot(om/pi,m);

ylabel('GainindB-->');

xlabel('(a)Normalised frequency-->');

subplot(2,1,2);

plot(om/pi,an)

xlabel('(b)Normalised frequency-->');

ylabel('Phase in radians-->')

Solution 9:

**Program for the design of Butterworth band pass analog filter**

Matlab Code:

clc;

close all;

clear all;

format long

rp=input('enter the passband ripple...');

rs=input('enter the stopband ripple...');

wp=input('enter the passband freq...');

ws=input('enter the stopband freq...');

fs=input('enter the sampling freq...');

w1=2\*wp/fs;

w2=2\*ws/fs;

[n]=buttord(w1,w2,rp,rs,'s');

wn=[w1 w2];

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

w=0:0.01:pi;

[h,om]=freqz(b,a,w);

m=20\*log10(abs(h));

an=angle(h);

subplot(2,1,1);

plot(om/pi,m);

ylabel('Gain in dB-->');

xlabel('(a)Normalised frequency-->');

subplot(2,1,2);

plot(om/pi,an)

xlabel('(b)Normalised frequency-->');

ylabel('Phase in radians-->');

solution 10:

**program for the design of chebyshev low pass filter**

**Matlab Code:**

clc;

close all;

clear all;

format long

rs=input('enter the stopband ripple...');

rp=input('enter the passband ripple...');

ws=input('enter the stopband freq...');

wp=input('enter the passband freq...');

fs=input('enter the sampling freq...');

w1=2\*wp/fs;

w2=2\*ws/fs;

[n,wn]=cheb1ord(w1,w2,rp,rs);

[b,a]=cheby1(n,rp,wn);

w=0:0.01:pi;

[h,om]=freqz(b,a,w);

m=20\*log10(abs(h));

an=angle(h);

subplot(2,1,1);

plot(om/pi,m);

ylabel('GainindB-->');

xlabel('(a)Normalised frequency-->');

subplot(2,1,2);

plot(om/pi,an)

xlabel('(b)Normalised frequency-->');

ylabel('Phase in radians-->');

Solution 11 :

**program for the design of chebyshev high pass filter**

**Matlab Code:**

clc;

close all;

clear all;

format long

rp=input('enter the pass band ripple...');

rs=input('enter the stop band ripple...');

wp=input('enter the pass band freq.....');

ws=input('enter the stop band freq.....');

fs=input('ente the sampling freq......');

w1=2\*wp/fs;

w2=2\*ws/fs;

[n,wn]=cheb1ord(w1,w2,rp,rs,'s');

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

w=0:0.01:pi;

[h,om]=freqs(b,a,w);

m=20\*log10(abs(h));

an=angle(h);

subplot(2,1,1);

plot(om/pi,m);

ylabel('GainindB-->');

xlabel('(a) Normalized frequency-->');

subplot(2,1,2);

plot(om/pi,an);

xlabel('(b) Normalized frequency')

ylabel('Phase in radians')

Solution 12:

**program for the design of chebyshev band pass filter**

**Matlab Code:**

clc;

close all;

clear all;

format long

rp=input('enter the pass band ripple...');

rs=input('enter the stop band ripple...');

wp=input('enter the pass band freq...');

ws=input('enter the stop band freq...');

fs=input('enter the sampling freq...');

w1=2\*wp/fs;

w2=2\*ws/fs;

[n]=cheb1ord(w1,w2,rp,rs,'s');

wn=[w1 w2];

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

w=0:0.01:pi;

[h,om]=freqs(b,a,w);

m=20\*log10(abs(h));

an=angle(h);

subplot(2,1,1);

plot(om/pi,m);

ylabel('GainindB-->');

xlabel('(a)Normalized frequency-->');

subplot(2,1,2);

plot(om/pi,an)

xlabel('(b)Normalised frequency-->');

ylabel('Phase in radians-->');

Solution 13:

**program for the design of chebyshev band stop filter.**

**Matlab Code:**

clc;

close all;

clear all;

format long

rs=input('enter the stop band ripple...');

rp=input('enter the pass band ripple...');

ws=input('enter the stop band freq...');

wp=input('enter the pass band freq...');

fs=input('enter the sampling freq...');

w1=2\*wp/fs;

w2=2\*ws/fs;

[n]=cheb1ord(w1,w2,rp,rs);

wn=[w1 w2];

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

w=0:0.1/pi:pi;

[h,om]=freqz(b,a,w);

m=20\*log10(abs(h));

an=angle(h);

subplot(2,1,1);

plot(om/pi,m);

ylabel('GainindB-->');

xlabel('(a)Normalized frequency-->');

subplot(2,1,2);

plot(om/pi,an)

xlabel('(b)Normalized frequency-->');

ylabel('Phase in radians-->');