**1.stability**

clear;

clc;

mode(0);

s=%s;

H=s^4+2\*s^3+3\*s^2+4\*s+5;

//H=s^5+7\*s^4+6\*s^3+42\*s^2+8\*s+56;

disp(H,'The given characteristics equation 1-G(s)H(s)=');

c=coeff(H);

len=length(c);

r=routh\_t(H);

disp(r,"Rouths table=");

x=0;

for i=1:len

if(r(i,1)<0)

x=x+1;

end

end

if(x>=1)

printf("From Rouths table, it is clear that the system is unstable.")

else

printf("From Rouths table, it is clear that the system is stable.")

end

**2.unit impulse signal**

clear all;

close ;

t2 = -5:5;

x2 =[ zeros (1 , N ) ,ones (1 , 1), zeros(1,N )];

subplot (2 ,4 ,2) ;

plot2d3 ( t2 , x2 )

xlabel ("time");

ylabel ("aplitude");

title ("unit signal");

**Unit step signal**

clear all;

close ;

t2 = -5:5;

x2 =[ zeros (1 , N ) ,ones (1 , N +1) ];

subplot (2 ,4 ,2) ;

plot2d3 ( t2 , x2 )

xlabel ("time");

ylabel ("aplitude");

title ("unit signal");

**Unit ramp signal**

*// Clear all variables and close all figures*

clear;

close;

*// Set limit*

N = 5;

t1 = -5:5;

*// Define the ramp signal*

x4 = [zeros(1, N), 0:N];

*// Plot the ramp signal*

subplot(2, 4, 4);

plot2d3(t1, x4);

xlabel("time");

ylabel("Amplitude");

title("Ramp signal");

**Unit random signal**

clear;

close;

*// Define time range*

t1 = -5:5;

*// Generate random signal*

x6 = grand(1, length(t1), "nor", 0, 1); *// Normal distribution, mean 0, std deviation 1*

*// Plot the random signal*

subplot(2, 4, 6);

plot2d(t1, x6);

xlabel("time");

ylabel("Amplitude");

title("Random signal");

**Unit exponential signal**

*// Clear all variables and close all figures*

clear;

close;

*// Set limit*

N = 5;

t1 = 0:1:20;

*// Define the exponential response signal (for positive values)*

x3 = [zeros(1, N), exp(t1(N+1:$))];

*// Plot the exponential response signal*

subplot(2, 4, 3);

plot2d3(t1, x3);

xlabel("time");

ylabel("Amplitude");

title("Exponential response signal");

**Unti sinusoidal signal**

*// Clear all variables and close all figures*

clear;

close;

*// Define time range*

t1 = -5:0.1:5; *// Finer interval for smooth sinusoidal plot*

*// Define the sinusoidal signal*

frequency = 1; *// Frequency of the sinusoid*

amplitude = 1; *// Amplitude of the sinusoid*

x5 = amplitude \* sin(2 \* %pi \* frequency \* t1);

*// Plot the sinusoidal signal*

subplot(2, 4, 5);

plot2d(t1, x5);

xlabel("time");

ylabel("Amplitude");

title("Sinusoidal signal");

**3. DIF fft to dit dff**

clear;

clc ;

close ;

x = [1,-1,-1,-1,1,1,1,-1];

X = fft (x , -1);

disp(X,'X(z) = ');

**4.** **Analog Filter design Using Transformation method**

**i) Validate Transformation with Standard Sampling Period T=1.Convert Analog transfer function into Digital using Bilinear Transformation of H(s)=(s^2+4.525)/(s^2+0.692\*s+0.504) using sci lab**

clear;

clc ;

close ;

s=%s;

z=%z;

HS=(s^2+4.525)/(s^2+0.692\*s+0.504);

T=1;

HZ=horner(HS,(2/T)\*(z-1)/(z+1));

disp(HZ,'H(z) =');

(ii)

clear;

clc ;

close ;

s=%s;

T=0.2;

HS=10/(s^2+7\*s+10);

elts=pfss(HS);

disp(elts,'Factorized HS = ');

p1=-5; p2=-2;

z=%z;

HZ=T\*((-3.33/(1-%e^(p1\*T)\*z^(-1)))+(3.33/(1%e^(p2\*T)\*z^(-1)))) ;

disp(HZ,'HZ = ');

**20.Analog Butterworth Filter**

//First Order Butterworth Low Pass Filter

clear;

clc;

close;

s = poly(0,'s');

Omegac = 0.2\*%pi;

H = Omegac/(s+Omegac);

T =1;//Sampling period T = 1 Second

z = poly(0,'z');

Hz = horner(H,(2/T)\*((z-1)/(z+1)))

HW =frmag(Hz(2),Hz(3),512);

W = 0:%pi/511:%pi;

plot(W/%pi,HW)

a=gca();

a.thickness = 3;

a.foreground = 1;

a.font\_style = 9;

xgrid(1)

xtitle('Magnitude Response of Single pole LPF Filter Cutoff frequency = 0.2\*pi','Digital Frequency--

>','Magnitude');

Disp(“Hz”,Hz);

//High Pass Filter Using Digital Filter Transformation

clear;

clc;

close;

s = poly(0,'s');

Omegac = 0.2\*%pi;

H = Omegac/(s+Omegac);

T =1;//Sampling period T = 1 Second

z = poly(0,'z');

Hz\_LPF = horner(H,(2/T)\*((z-1)/(z+1)));

alpha = -(cos((Omegac+Omegac)/2))/(cos((Omegac-Omegac)/2));

HZ\_HPF=horner(Hz\_LPF,-(z+alpha)/(1+alpha\*z))

HW =frmag(HZ\_HPF(2),HZ\_HPF(3),512);

W = 0:%pi/511:%pi;

plot(W/%pi,HW)

a=gca();

a.thickness = 3;

a.foreground = 1;

a.font\_style = 9;

xgrid(1)

xtitle('Magnitude Response of Single pole HPF Filter Cutoff frequency = 0.2\*pi','Digital Frequency--

&gt;','Magnitude');

disp(“HZ\_HPF”,HZ\_HPF);

clear;

clc;

close;

omegaP = 0.2\*%pi;

omegaL = (2/5)\*%pi;

omegaU = (3/5)\*%pi;

z=poly(0,'z');

H\_LPF = (0.245)\*(1+(z^-1))/(1-0.509\*(z^-1))

alpha = (cos((omegaU+omegaL)/2)/cos((omegaU-omegaL)/2));

k = (cos((omegaU - omegaL)/2)/sin((omegaU - omegaL)/2))\*tan(omegaP/2);

NUM =-((z^2)-((2\*alpha\*k/(k+1))\*z)+((k-1)/(k+1)));

DEN = (1-((2\*alpha\*k/(k+1))\*z)+(((k-1)/(k+1))\*(z^2)));

HZ\_BPF=horner(H\_LPF,NUM/DEN)

disp(HZ\_BPF,'Digital BPF IIR Filter H(Z)= ')

HW =frmag(HZ\_BPF(2),HZ\_BPF(3),512);

W = 0:%pi/511:%pi;

plot(W/%pi,HW)

a=gca();

a.thickness = 3;

a.foreground = 1;

a.font\_style = 9;

xgrid(1)

xtitle('Magnitude Response of BPF Filter', 'Digital Frequency--->','Magnitude');

Disp(“HZ\_BPF”,HZ\_BPF);

4.

clear;

clc;

close;

omegaP = 0.2\*%pi;

omegaL = (2/5)\*%pi;

omegaU = (3/5)\*%pi;

z=poly(0,'z');

H\_LPF = (0.245)\*(1+(z^-1))/(1-0.509\*(z^-1))

alpha = (cos((omegaU+omegaL)/2)/cos((omegaU-omegaL)/2));

k = tan((omegaU - omegaL)/2)\*tan(omegaP/2);

NUM =((z^2)-((2\*alpha/(1+k))\*z)+((1-k)/(1+k)));

DEN = (1-((2\*alpha/(1+k))\*z)+(((1-k)/(1+k))\*(z^2)));

HZ\_BSF=horner(H\_LPF,NUM/DEN)

HW =frmag(HZ\_BSF(2),HZ\_BSF(3),512);

W = 0:%pi/511:%pi;

plot(W/%pi,HW)

a=gca();

a.thickness = 3;

a.foreground = 1;

a.font\_style = 9;

xgrid(1)

xtitle('Magnitude Response of BSF Filter','Digital Frequency--->','Magnitude');

Disp(“HZ\_BSF”,HZ\_BSF);