Control and Instrumentation Laboratory (EC692) Experiment No.: 1

Dhiman Sarkar
Roll: 19101105086
Dept. of Electronics and Comm. Engineering
Jalpaiguri Government Engineering College
Email: ds2286@ece.jgec.ac.in

Alok Barman
Roll: 19101105087
Dept. of Electronics and Comm. Engineering
Jalpaiguri Government Engineering College
Email: ab2287@ece.jgec.ac.in

Alka Tigga
Roll: 19101105088
Dept. of Electronics and Comm. Engineering
Jalpaiguri Government Engineering College
Email: at2288@ece.jgec.ac.in

Azizul Mallick
Roll: 19101105089
Dept. of Electronics and Comm. Engineering
Jalpaiguri Government Engineering College
Email: am2289@ece.jgec.ac.in

***Abstract*—Familiarization with MATLAB control system toolbox and representation of pole zero and transfer function of control system.**

I. Brief Description of The Experiment

a) Defining a Polynomial to be used with Control-System-Toolbox.
b) Finding the roots of a polynomial.
c) Finding the Product of two polynomial.
d) Defining a Transfer Function.
e) Finding the Pole and Zero of a Transfer Function.
f) Generation Test Signals - Unit Step, Unit Ramp, Unit Impulse.

II. MATLAB Script

```
%% Defining a Polynomial to be used with Control−System−Toolbox
p1 = [1 2 6 7] % Polynomial Coefficient Matrix of  F(s) = s^3 + 2s^2 + 6s + 7

%% Finding the roots of a polynomial
Roots = roots(p1)

%% Product of two polynomial
p2 = [1 0 7] %Polynomial Coefficient Matrix of  F(s) = s^2 + 7
p3 = conv(p1,p2)

%% Defining a Transfer Function
tFcn = tf(p2,p1) % H(s) = (s^2+7)/(s^3+2s^2+6s^2+7)

%% Finding the Poles and Zeros of a Transfer Function
Poles = pole(tFcn)
Zeros = zero(tFcn)
pzmap(tFcn); %Figure.(2)

%% Generation Test Signals
t = −1:0.001:1;
unitStep = []; %Since Rise Time, Δt = 0.01, Transisiton Frequency = 1/0.001 = 1kHz
unitRamp = [];
unitImpulse = t==0;
for i = 1:length(t)
    if t(i)>=0
        unitStep = [unitStep, 1];
        unitRamp = [unitRamp, t(i)];
    else
        unitStep = [unitStep,0];
        unitRamp = [unitRamp, 0];
    end
end
figure;
subplot(3,1,1);
plot(t,unitStep,'LineWidth',2); %Figure.(1)
ylim([0,1.1]);grid on;box on;
title('Unit Step Signal');

subplot(3,1,2);
plot(t,unitRamp,'LineWidth',2); %Figure.(1)
ylim([0,1.1]);grid on;box on;
title('Unit Ramp Signal');

subplot(3,1,3);
plot(t,unitImpulse,'LineWidth',.5); %Figure.(1)
ylim([0,1.1]);grid on;box on;
title('Unit Impulse Signal');
```

III. Output Results and Plots

```
p1 =

     1 2 6 7


Roots =

   −0.3181 + 2.2430i
   −0.3181 − 2.2430i
   −1.3639 + 0.0000i


p2 =

     1 0 7


p3 =

     1 2 13 21 42 49


tFcn =

         s^2 + 7
  ---------------------
  s^3 + 2 s^2 + 6 s + 7

Continuous−time transfer function.
```

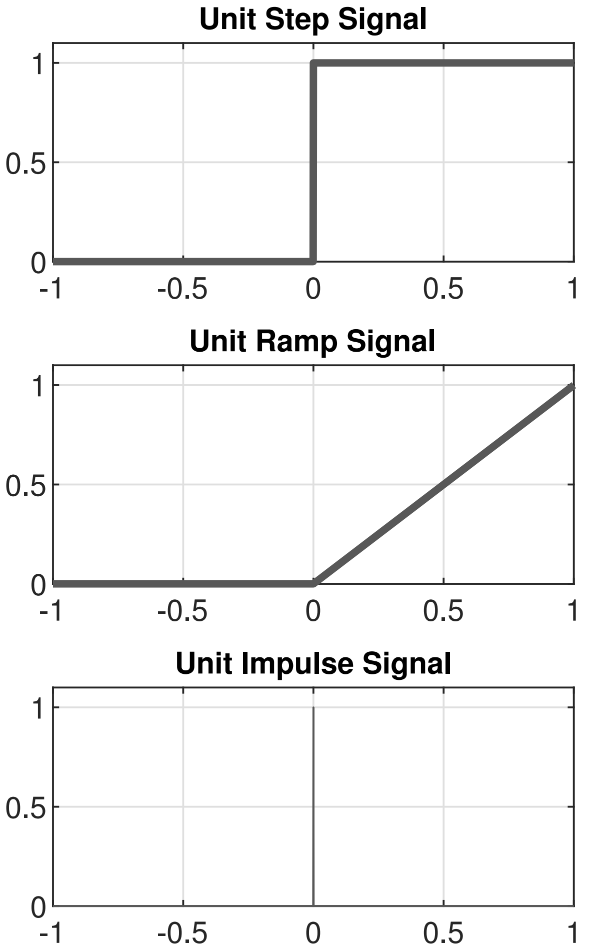

Figure 1. Generated Test Signals Output [see line number - 35,40,45

```
Poles =

  -0.3181 + 2.2430i
  -0.3181 - 2.2430i
  -1.3639 + 0.0000i


Zeros =

   0.0000 + 2.6458i
   0.0000 - 2.6458i
```

IV. Observations & Conclusion

We have already studied frequency domain analysis in *Signals & Systems* course. And now after being familiarized with MATLAB and MATLAB Control System Toolbox, we've learned to compute calculations in s-domain for more complex systems.

V. Acknowledgment

We would like to express our sincere gratitude to our course instructor Mr. Mirwaiz Rahaman and Mrs. Purba Basu for providing their invaluable guidance, comments and suggestions.

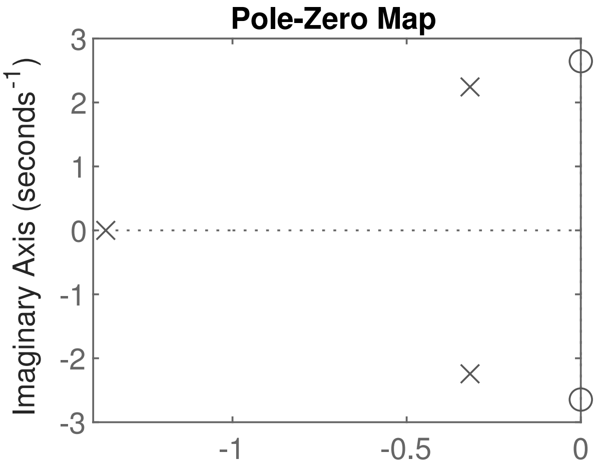

Figure 2. Pole-Zero Map of the transfer function *tFcn* defined in line number.(12) of the MATLAB script.