Roll: 1715006 ; Name : Adowan Shahriar ; Lab : 1

Objectives :

1. To find magnitude and angle of complex number.
2. To be able to express any factored form expression to polynomial form.

→ To find roots of any function.

→ To find transfer function of LTI system.

→ To convert LTI models into polynomial and factored form.

→ To plot any function.

Introduction :

In engineering, a tranfer function of an electronic or control system component is a mathematical function which theoretically models the device's output for each possible input.

A system represented by a differential equation is difficult to model as a block diagram. With Laplace transform this can represent the input, output and system as ~~sp~~ separate entities. Let ;

us first define the Laplace transform ;

$$\mathcal{L}[f(t)] = F(s) = \int_0^\infty f(t) e^{-st} dt$$

where $s = \sigma + j\omega$, a complex variable.

The transfer function of a control system is defined as the ratio of the Laplace transform of the output variable to Laplace transform of the input variable assuming all initial conditions to be zero.

Let us begin by writing a general n^{th}-order, linear time invariant differential equation,

$$a_n \frac{d^n c(t)}{dt^n} + a_{n-1} \frac{d^{n-1} c(t)}{dt^{n-1}} + \dots a_0 c(t)$$
$$= b_m \frac{d^m r(t)}{dt^m} + b_{m-1} \frac{d^{m-1} r(t)}{dt^{m-1}} + \dots b_0 r(t) \quad \text{--- (i)}$$

where, $c(t)$ is the output, $r(t)$ is the input, and a_i's, b_i's and the form of the differential equation represent the system. Taking the Laplace transfer of both sides,

$$a_n s^n C(s) + a_{n-1} s^{n-1} C(s) + \dots + a_0 C(s)$$
$$= b_m s^m R(s) + b_{m-1} s^{m-1} R(s) + \dots + b_0 R(s) \quad \text{--- (ii)}$$

Equation (ii) is completely algebric if we assume initial conditions are 0.

Now forming the ratio of the ~~outform~~ output transform, C(S), divided by the input transform, R(S):

$$\frac{C(S)}{R(S)} = G(S) = \frac{(b_m s^m + b_{m-1} s^{m-1} + \dots + b_0)}{(a_n s^n + a_{n-1} s^{n-1} + \dots + a_0)} \quad \text{--- (iii)}$$

The transfer function can be represent as block-diagram as shown in figure 1. below:

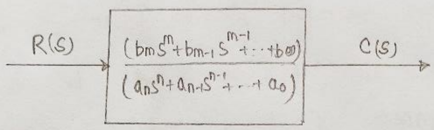


Figure 1: Block of diagram of a transfer function

And a linear system possesses two properties: superposition and homogenity. The property of superposition means that the output response of a system to the sum of inputs is the sum of the responses to the individual inputs. The property of Homogenity describes the response of the system to a multiplication of the input by a scalar.

Superposition: $r_1(t) + r_2(t) \longrightarrow c_1(t) + c_2(t)$

Homogenity: $A\, r_1(t) \longrightarrow A\, c_1(t)$

Task:1

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
M=5 % Assign 5 to M and display.
N=6 % Assign 6 to N and display.
P=M+N % Assign M+N to P and display.
Q=3+4j % Define complex number, Q.
MagQ=abs(Q) % Find magnitude of Q.
ThetaQ=(180/pi)*angle(Q)
```

Output:

```
P =

    11


Q =

   3.0000 + 4.0000i


MagQ =

     5


ThetaQ =

   53.1301
```

Task 2:

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
 % Display label.
P1= [1 7 -3 2 3]

```

Output:

```
P1 =

     1     7    -3     2     3
```

Task 3:

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
P2 = [3 5 7 8]; % Assign 3s^3 + 5s^2 +7s + 8 to P2
% without displaying.
x=3*5 % Evaluate 3*5 and display result.

```

Output:

Command Window

```
x =

    15
```

Task 4:

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
P3=poly([-2 -5 -6]) % Store polynomial
% (s+2) (s+5)(s+6) as P3 and
% display the coefficients.
```

Output:

Command Window

```
P3 =

     1    13    52    60
```

Task 5:

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
P4=[5 7 9 -3 2] % Form 5s^4+7s^3+9s^2-3s+2 and
% display.
rootsP4=roots(P4) % Find roots of 5s^4+7s^3+9s^2
%-3s+2,
% assign to rootsP4, and display.
```

Output:

Command Window

```
P4 =

     5     7     9    -3     2


rootsP4 =

  -0.8951 + 1.2351i
  -0.8951 - 1.2351i
   0.1951 + 0.3659i
   0.1951 - 0.3659i
```

Task 7:

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
P5=conv([1 7 1 0 9],[1 -3 6 2 1]) % Form (s^3+7s^2+10s+9)(s^4-
% 3s^3+6s^2+2s+1), assign to P5,
% and display.

```

Output:

Command Window

```
P5 =

     1     4   -14    41    30   -18    55    18     9
```

Task:8

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
numf=[7 9 12] % Define numerator of F(s).
denf=conv(poly([0-7]),[1 10 100]); % Define denominator of F(s).
[K, p, k]=residue (numf ,denf) % Find residues and assign to K;
% find roots of denominator and
% assign to p; find
% constant and assign to k.
```

Output:

```
Command Window

numf =

     7     9    12


K =

   1.6519 + 3.9033i
   1.6519 - 3.9033i
   3.6962 + 0.0000i


p =

  -5.0000 + 8.6603i
  -5.0000 - 8.6603i
  -7.0000 + 0.0000i


k =

     []
```

Task 9:

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```matlab
clc
clear all
close all
numy=32; % Define numerator.
deny=poly([0 -4 -8]); % Define denominator.
[r, p, k]=residue (numy, deny) % Calculate residues, poles, and
% direct quotient.

```

Output:

```
r =

     1
    -2
     1


p =

    -8
    -4
     0


k =

     []
```

Task 10:

```
%Vector Method, Polynomial Form % Display label.
numf=150*[1 2 7] % Store 150 (s^2+2s+7) in numf and
% display.
denf=[1 5 4 0] % Store s (s+1) (s+4) in denf and
% display.
%?F(s)? % Display label.
F=tf (numf, denf) % Form F(s) and display.
clear % Clear previous variables from
% workspace.
%?Vector Method, Factored Form? % Display label.
numg=[-2 -4] % Store (s+2)(s+4) in numg and
% display.
deng=[-7 -8 -9] % Store (s+7) (s+8) (s+9) in deng
% and display.
K=6 % Define K.
%?G(s)? % Display label.
G=zpk(numg,deng,K) % Form G(s) and display.
clear % Clear previous variables from
% workspace.
%?Rational Expression Method, Polynomial Form?
% Display label.
s=tf('s') % Define ?s? as an LTI object in
% polynomial form.
F=150*(s^2+2*s+7)/[s*(s^2+5*s+4)] % Form F (s) as an LTI transfer
% function in polynomial form.
G=20*(s+2)*(s+4)/[(s+7)*(s+8)*(s+9)] % Form G(s) as an LTI transfer
 % function in polynomial form.
clear % Clear previous variables from
```

```
% workspace.
%Rational Expression Method, Factored Form?
% Display label.
s=zpk('s') % Define ?s? as an LTI object in
% factored form.
F=150*(s^2+2*s+7)/[s*(s^2+5*s+4)]
% Form F (s) as an LTI transfer
% function in factored form.
G=20*(s+2)*(s+4)/[(s+7)*(s+8)*(s+9)]
% Form G(s) as an LTI transfer
% function in factored form.
```

Output:

```
numf =

       150        300       1050


denf =

    1     5     4     0


F =

  150 s^2 + 300 s + 1050
  ----------------------
    s^3 + 5 s^2 + 4 s
```

Continuous-time transfer function.

```
numg =

    -2    -4


deng =

    -7    -8    -9


K =

     6


G =

    6 (s+2) (s+4)
  -----------------
  (s+7) (s+8) (s+9)
```

Continuous-time zero/pole/gain model.

```
s =
```

```
  s

Continuous-time transfer function.


F =

  150 s^2 + 300 s + 1050
  ----------------------
    s^3 + 5 s^2 + 4 s

Continuous-time transfer function.


G =

     20 s^2 + 120 s + 160
  --------------------------
  s^3 + 24 s^2 + 191 s + 504

Continuous-time transfer function.


s =

  s

Continuous-time zero/pole/gain model.
```

```
F =

  150 (s^2 + 2s + 7)
  ------------------
    s (s+4) (s+1)

Continuous-time zero/pole/gain model.

G =

   20 (s+2) (s+4)
  -----------------
  (s+7) (s+8) (s+9)

Continuous-time zero/pole/gain model.

>>
```

Task 11:

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
%?Coefficients for F(s)? % Display label.
numftf=[10 4 0 60] % Form numerator of F (s)=
% (10s^2+40s+60)/(s^3+4s^2+5s
% +7).
denftf=[1 4 5 7] % Form denominator of F (s)=
% (10s^2+40s+60)/(s^3+4s^2+5s
% +7).
%?Roots for F(s)? % Display label.
[numfzp,denfzp]=tf2zp (numftf,denftf)
% Convert F(s) to factored form.
%?Roots for G(s)? % Display label.
numgzp=[-2 -4] % Form numerator of
K=6 % G(s)=10(s+2)(s+4)=[s(s + 3)
% (s+5)].
dengzp=[0 -3 -5] % Form denominator of
% G(s)=10(s+2)(s+4)/[s(s+3)(s+5)].
%?Coefficients for G(s)? % Display label.
[numgtf,dengtf]=zp2tf(numgzp',dengzp',K)
% Convert G(s) to polynomial form.
```

Output:

numftf =

10 4 0 60

```
denftf =

     1     4     5     7


numfzp =

  -1.9607 + 0.0000i
   0.7804 + 1.5656i
   0.7804 - 1.5656i


denfzp =

  -3.1163 + 0.0000i
  -0.4418 + 1.4321i
  -0.4418 - 1.4321i


numgzp =

    -2    -4


K =

     6
```

```
dengzp =

     0    -3    -5


numgtf =

     0     6    36    48


dengtf =

     1     8    15     0

>>
```

Task 12:

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
%?Fzpk1(s)? % Display label.
Fzpk1=zpk([-2-4],[0-3-5],10) % Form Fzpk1 (s)=
% 10(s+2)(s+4)/[s(s+3)(s+5)].
%?Ftf1? % Display label.
Ftf1=tf (Fzpk1) % Convert Fzpk1 (s) to
% coefficients form.
%?Ftf2? % Display label.
Ftf2=tf([10 40 60],[1 4 5 7]) % Form Ftf2(s)=
% (10s^2+40s+60)/(s^3+4s^2+5s
% +7).
%?Fzpk2? % Display label.
Fzpk2=zpk(Ftf2) % Convert Ftf2 (s) to
% factored form.
```

Output:

```
Fzpk1 =

 10 (s+6)
 --------
  (s+8)

Continuous-time zero/pole/gain model.



Ftf1 =
```

```
  10 s + 60
  ---------
    s + 8

Continuous-time transfer function.


Ftf2 =


   10 s^2 + 40 s + 60
  ---------------------
  s^3 + 4 s^2 + 5 s + 7

Continuous-time transfer function.


Fzpk2 =


         10 (s^2 + 4s + 6)
  ---------------------------------
  (s+3.116) (s^2 + 0.8837s + 2.246)

Continuous-time zero/pole/gain model.

>>
```

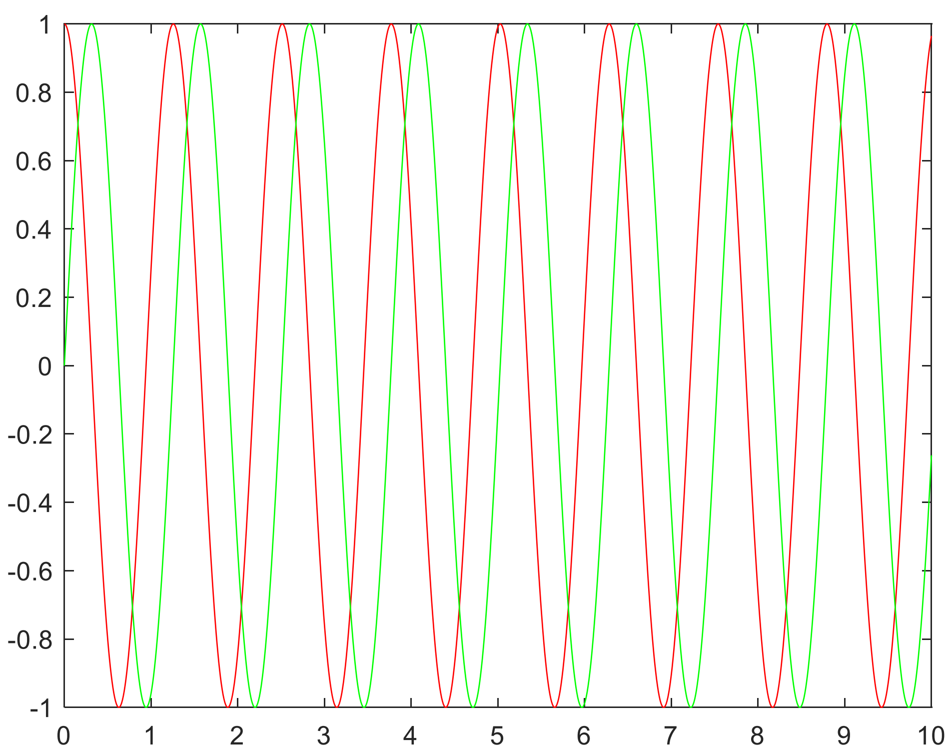
Task 13:

Editor - C:\Users\DELL\Untitled.m

Untitled.m

```
clc
clear all
close all
%?(ch2p12)? % Display label.
t=0:0.01:10; % Specify time range and increment.
f1=cos(5*t); % Specify f1 to be cos(5t).
f2=sin(5*t); % Specify f2 to be sin (5t).
plot(t,f1,'r',t,f2,'g') % Plot f1 in red and f2 in green.
```

Output:



Discussion:

This was our first lab performing regarding the covid situation and we're too excited to participate. We've revised a lot of terms of the MATLAB functions. We've learned many of new functions also. We can find the magnitude and angle of a complex number using 'abs()' & 'angle ( )' respectively.

Any factored form function can be expressed as polynomial by 'poly ( )', we can find roots of polynomials by 'roots()' command. Polynomials can be multiplied together using 'conv (a, b)' and partial fraction can be found using [var1, var2, var3] = residue(b,a) rational expression by var1 = tf('var1') or zpK('var1') and last but not the least plotting by 'plot()' function.

Conclusion:

We can conclude learning of many new functions, terms, procedures, formula and calculating we learned how to find magnitude and angle on matlab function. We did converted factorial form expression to polynomial form, find roots, transfer function of LTI system and all these by using MATLAB and it's in-built functions