Faculty of Engineering & Technology Electrical & Computer Engineering Department

CONTROL SYSTEMS ENEE3302

Project

Prepared by:

Name: Saif Battah. Number: 1170986

Instructor: Dr. Hakam Shehadeh

Section: 1.

Date: 5/9/2022.

Project solution:

At the beginning we use to define the given parameters and constants to the matlab project system as follows:

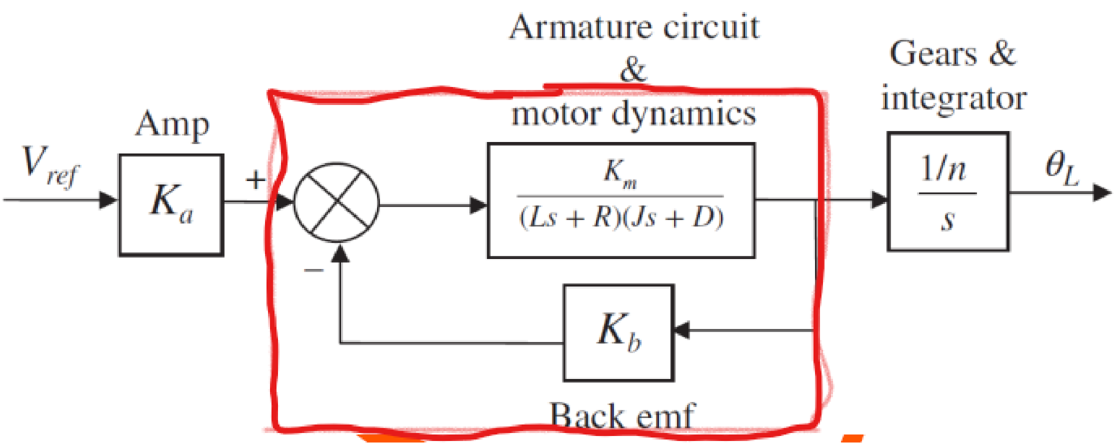
```
Ka = 12;
L = 0.006;
R = 1.4;
Kb = 0.00867;
n = 200;
Km = 4.375;
JL = 1;
DL = 0.5;
Jm = 0.00844;
Dm = 0.00013;
J = Jm + JL/n^2;
D = Dm + DL/n^2;
```

Then we define the Laplace variable as follows:

```
s = tf('s');
```

---

a) We define the armature circuit & motor dynamics T.F as follows:



Code:

```
Garm_motor = Km/((L*s+R)*(J*s+D));
```

After that we define the whole system open-loop T.F as follows:

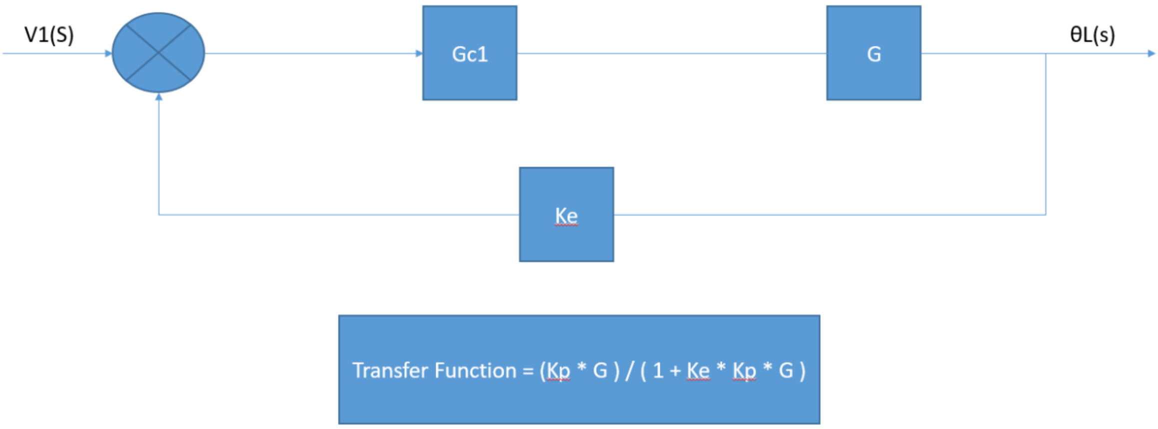
```
G = Ka*feedback(Garm_motor,Kb)*1/(n*s)
```

And the result of the code is:

```
G =

                  52.5
  --------------------------------
  0.01016 s^3 + 2.37 s^2 + 7.626 s

Continuous-time transfer function.
```

Which is the Open-Loop Transfer Function

b)



c)

Code

```
>> syms s Kp
Gs=52.5/(0.01016*s^3+2.37*s^2+7.626*s);
Gopenloop=Kp*Gs;
Gclosedloop=simplify(Gopenloop/(1+Gopenloop))
```

Closed-Loop Transfer Function

```
Gclosedloop =

(656250*Kp)/(127*s^3 + 29625*s^2 + 95325*s + 656250*Kp)
```

d)

Code

```
for i=1:0.1:100
sys=feedback(G*i,1);
st=isstable(sys);
if st == 0
Kp=i;
disp(['The overall system is stable for the controller gain Kp < ',num2str(Kp)])
break;
end
end
```

Result

```
The overall system is stable for the controller gain Kp < 33.9
```

So, System is stable if and only if Kp is less than 33.9

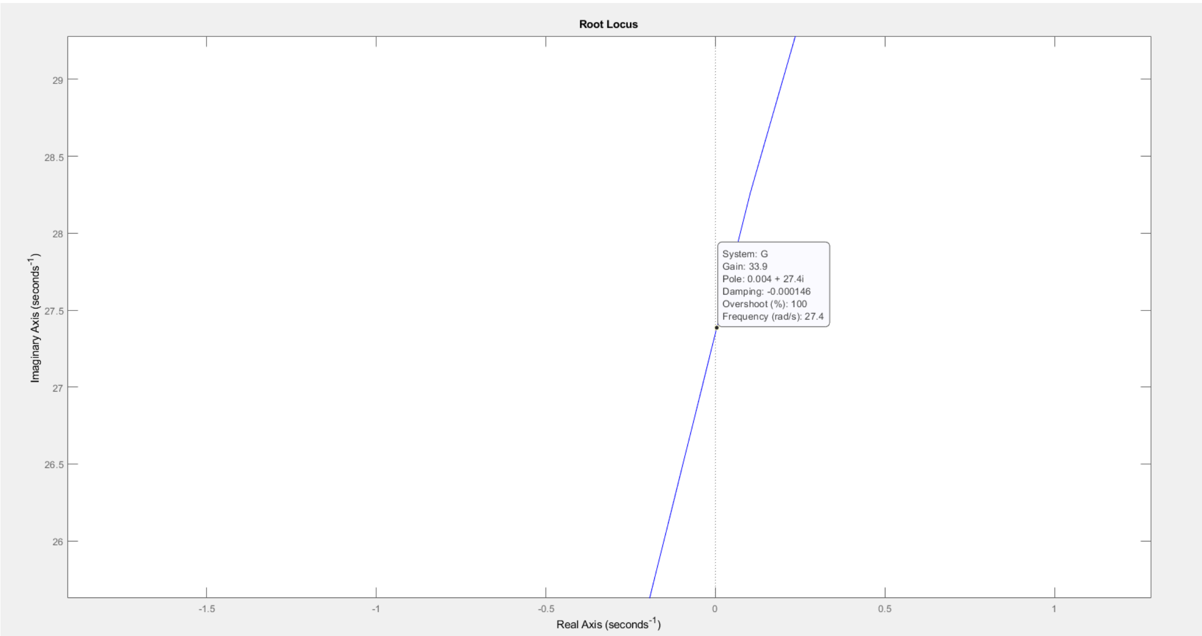
OR

By Using root locus technique

Code

```
figure;
rlocus(G);
```

Result



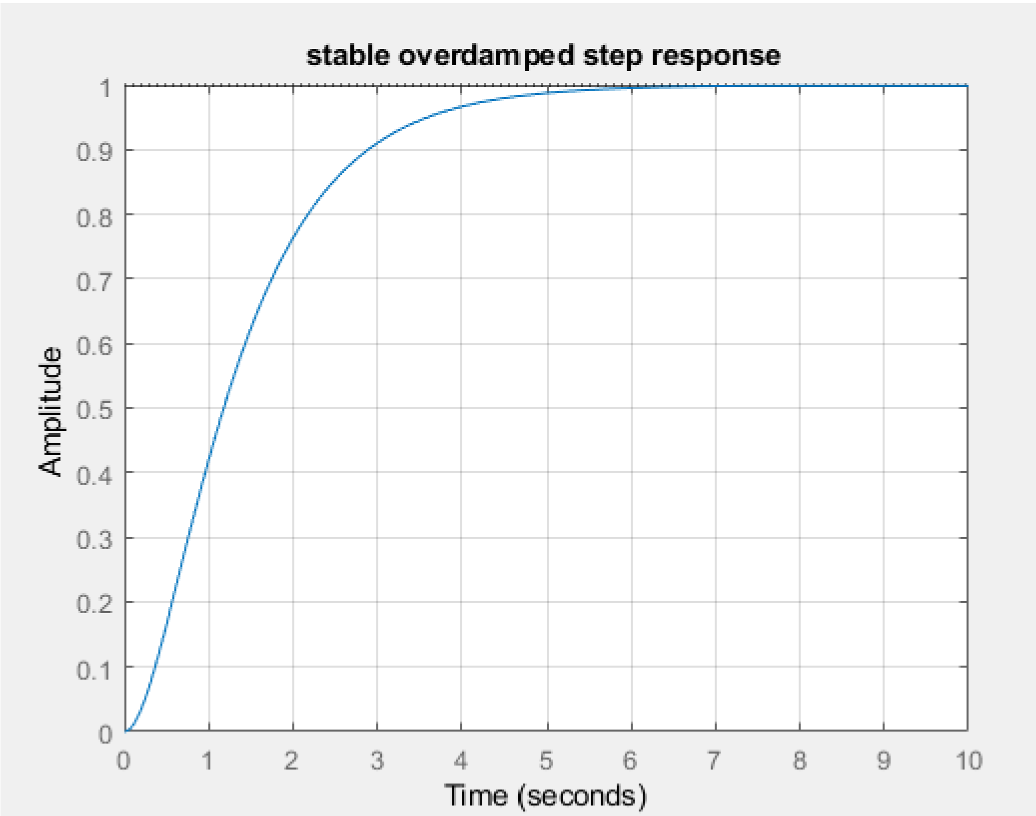
e)

Code

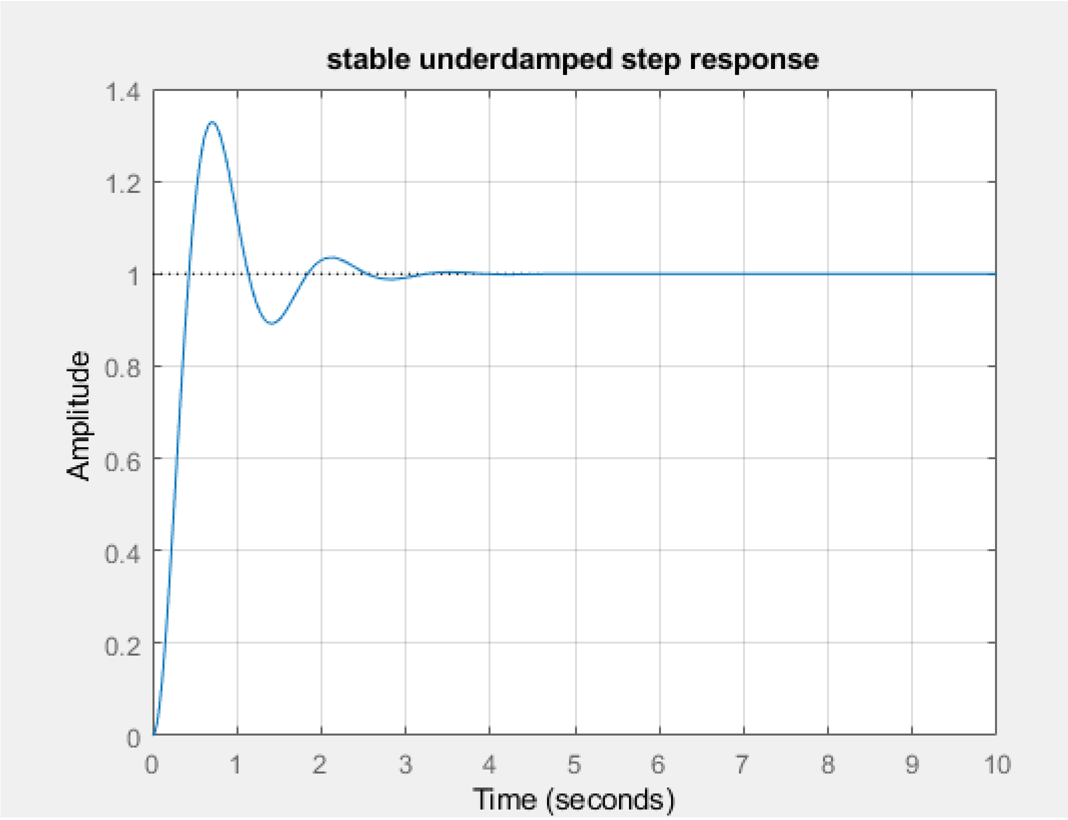
```
Kp = [0.1 1 33.78 45];
figure;
step(feedback(G*Kp(1),1));grid on;
title('stable overdamped step response');xlim([0 10]);

figure;
step(feedback(G*Kp(2),1));grid on;
title('stable underdamped step response');xlim([0 10]);

figure;
step(feedback(G*Kp(3),1));grid on;
title('marginally stable undamped step response');xlim([0 10]);

figure;
step(feedback(G*Kp(4),1));grid on;
title('unstable step response');xlim([0 10]);
```

Results

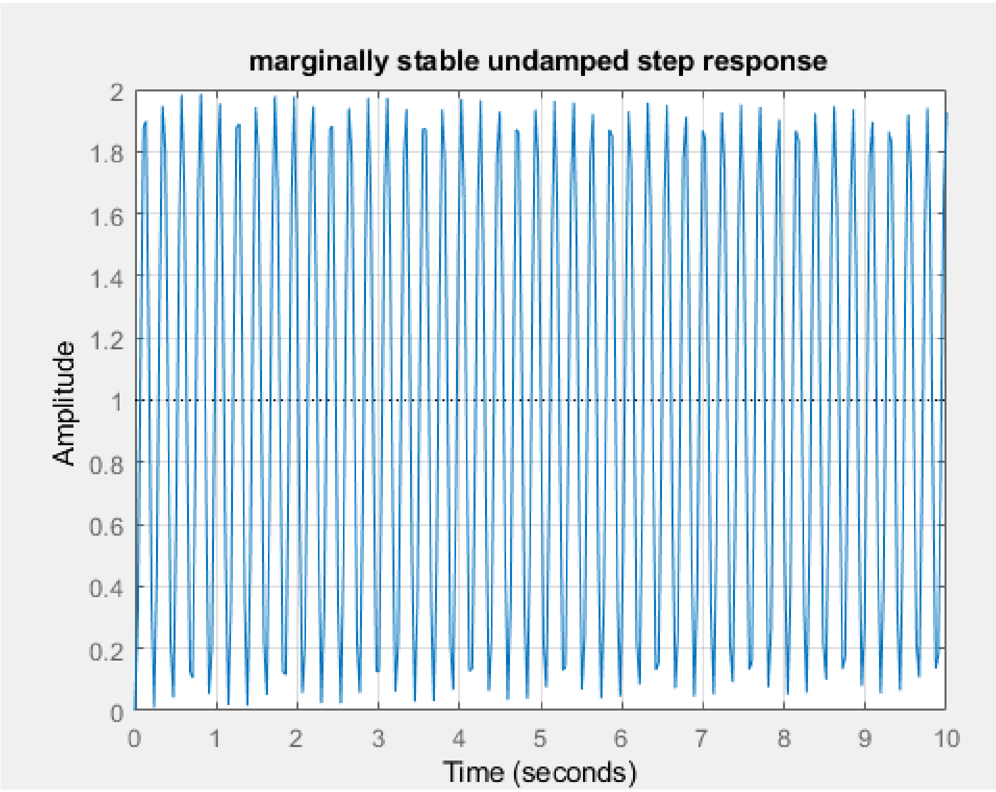
Stable Overdamped step response



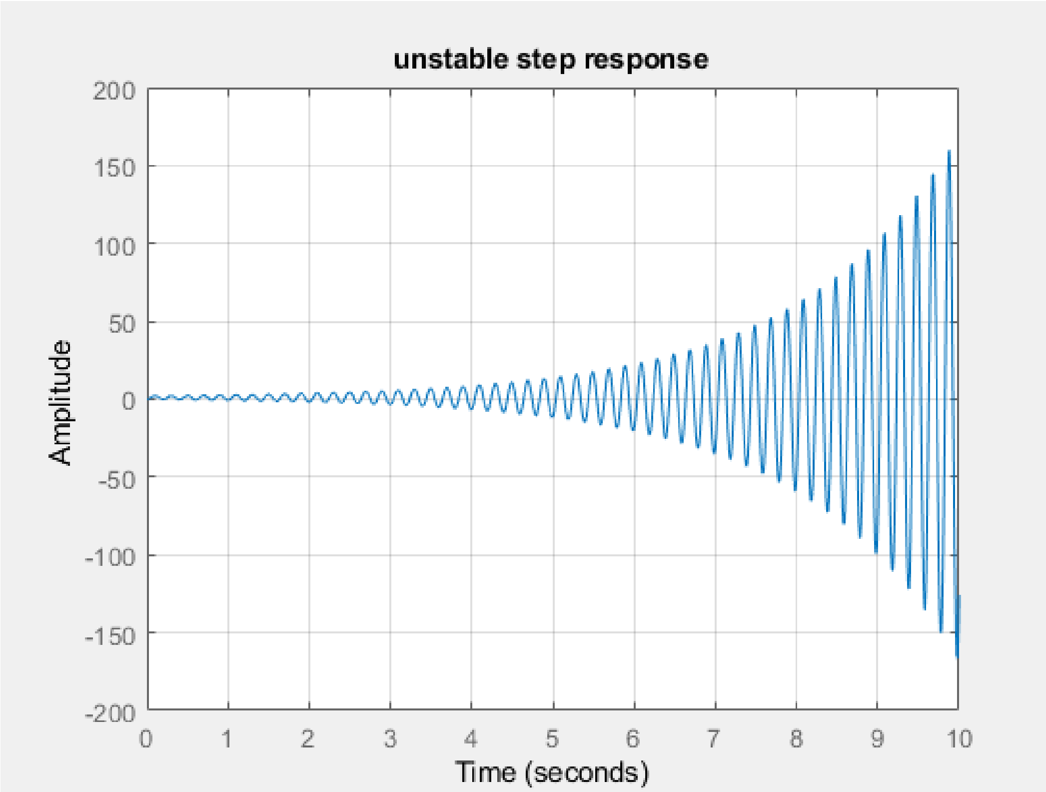
Stable Underdamped step response



Marginally Stable Undamped step response



Unstable step response



---

f)

Code

```
% Kp is Given a stable value, Kp = 15
sys=feedback(G*15,1);
stepinfo(sys)
```

Result

```
ans =

  struct with fields:

        RiseTime: 0.0596
    SettlingTime: 4.3168
     SettlingMin: 0.2707
     SettlingMax: 1.8513
       Overshoot: 85.1310
      Undershoot: 0
            Peak: 1.8513
        PeakTime: 0.1717

>>
```

g)

Zero, Look at the stable step response figure!

MATLAB CODE

```
clc;
close all;
clear all;

% defining the parameters and constants

Ka=12;
L=0.006;
R=1.4;
Kb=0.00867;
n=200;
Km=4.375;
JL=1;
DL=0.5;
Jm=0.00844;
Dm=0.00013;
J=Jm+JL/n^2;
D=Dm+DL/n^2;


% define the laplce variable s
s = tf('s');

% define the armature and motor dynamics transfer function
Garm_motor = Km/((L*s+R)*(J*s+D));

%define the Gears & integrator function
Ggears_integrator = 1/(n*s);

% open-loop transfer function
G = Ka*feedback(Garm_motor,Kb)*Ggears_integrator

% closed-loop transfer function
T = feedback(G,1)

%===========================================================================
% determine the value of Kp for which the system is stable
for i=1:0.1:100
%Calculate the closed loop transfer function for various Kp = i
sys=feedback(G*i,1);
is_stable=isstable(sys); % checks the system is stable or not
if is_stable == 0 % if the system is not stable
Kp=i;
disp(['The overall system is stable for the controller gain Kp <
',num2str(Kp)])
```

```matlab
break;
end
end
% OR
% Using root locus technique, on the open loop system
figure;
rlocus(G);

% The step response for four different values of Kp

% stable overdamped
Kp = [0.1 1 33.78 45];
figure;
step(feedback(G*Kp(1),1));grid on;
title('stable overdamped step response');xlim([0 10]);

% stable underdamped
figure;
step(feedback(G*Kp(2),1));grid on;
title('stable underdamped step response');xlim([0 10]);

% marginally stable
figure;
step(feedback(G*Kp(3),1));grid on;
title('marginally stable undamped step response');xlim([0 10]);

% Unstable
figure;
step(feedback(G*Kp(4),1));grid on;
title('unstable step response');xlim([0 10]);

%==========================================================================

% Kp is Given a stable value, Kp = 15
sys=feedback(G*15,1);
stepinfo(sys)
```