Visvesvaraya National Institute of Technology (VNIT), Nagpur

Control Theory (ECL312 ECP312)

Lab Report

Submitted by :
Bipasha Parui (BT19ECE019)
Semester 4

Submitted to :
Dr.Punit K. Bhavsar and Dr.Deep Gupta
(Course Instructors)
Department of Electronics and Communication Engineering,
VNIT Nagpur

Contents

Experiment-1: ODE solver .

Code:

```
clc;
clear all;
F = 300; %force F
M = 10;  %mass M

f = 4;  %Viscous Frictnn Coeff
k = 20;  %Spring Constant

%Initial conditions : x(0) = 0, x'(0) = 0
x0 = [0, 0];

odefun = @(t, x) [x(2);F/M-f*x(2)/M-k*x(1)/M];
tp = linspace(0,50);

[t, x] = ode45(odefun, tspan, x0);

subplot(2,1,1)
plot(t, x(:,1))
xlabel('Time in seconds'); ylabel("Displacement x"); title('x')
grid on;
subplot(212)
plot(t, x(:,2))
xlabel('Time in seconds'); ylabel("Derivative of x"); title('dx/dt')
grid on
```

Code:

```
clc;
clear all;

pkg load control;

function myoutput = Order2_ODE(t,y)
  myoutput =[y(2);-2*y(2)-5];
end

 ##2nd order
#init = [3,4];
tp = [0 20];
[t,y] = ode45('Order2_ODE',tp,[3,4]);
figure(1)
plot(t,y(:,1),"")
title("y")
label("t")
ylabel("y")

figure(2)
plot(t,y(:,2),"?.")
title("dy/dt")
xlabel("t")
ylabel("dy/dt")
```

Graph: below

Theory: below

Results: We considered a mechanical system for ode solver defined by

$$F = M\frac{d^2x}{dt^2} + f\frac{dx}{dt} + Kx$$

and solved the equation using ODE45 solver.

Secondly we considered a defined 2nd order function

$$y''(t) + 2y(t)' + 5 = 0$$

and solved as shown in theory.

Conclusions: Thus we solve a 2nd order differential equation using an ODE45 solver by breaking it into furthur 1st order differential equations.

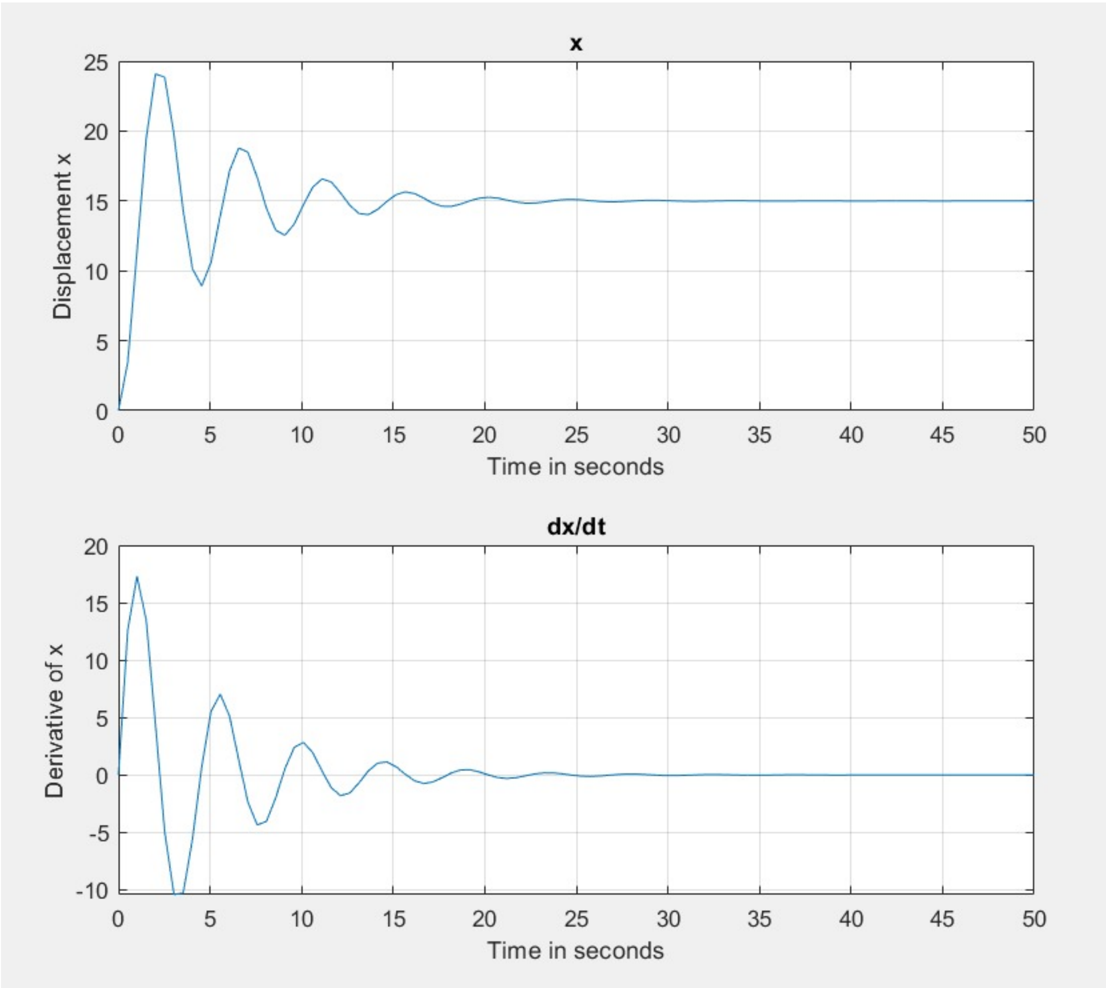


Figure 1: ODE solver for mechanical system

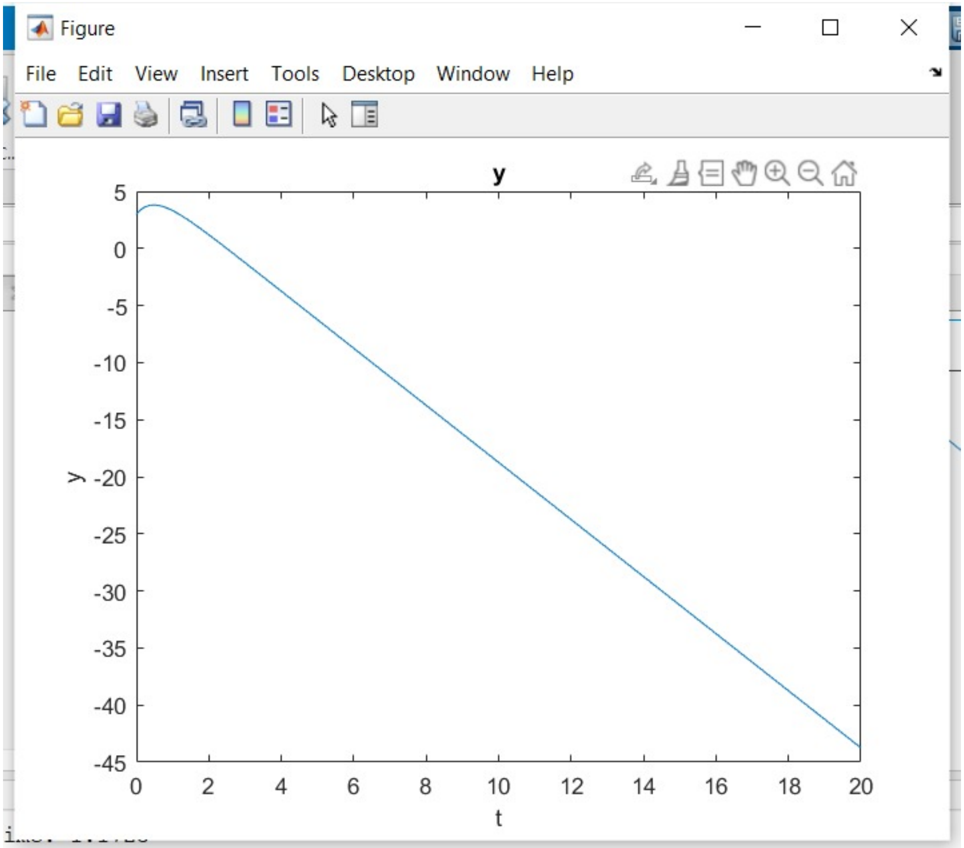


Figure 2: ODE solver for defined function

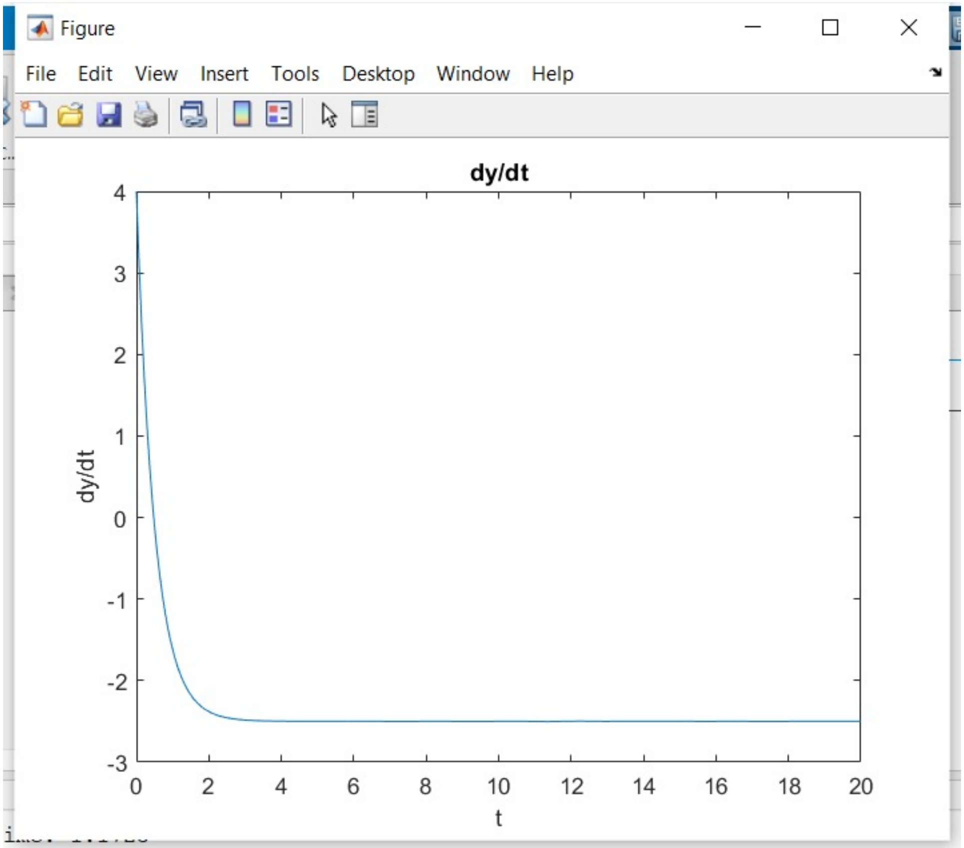
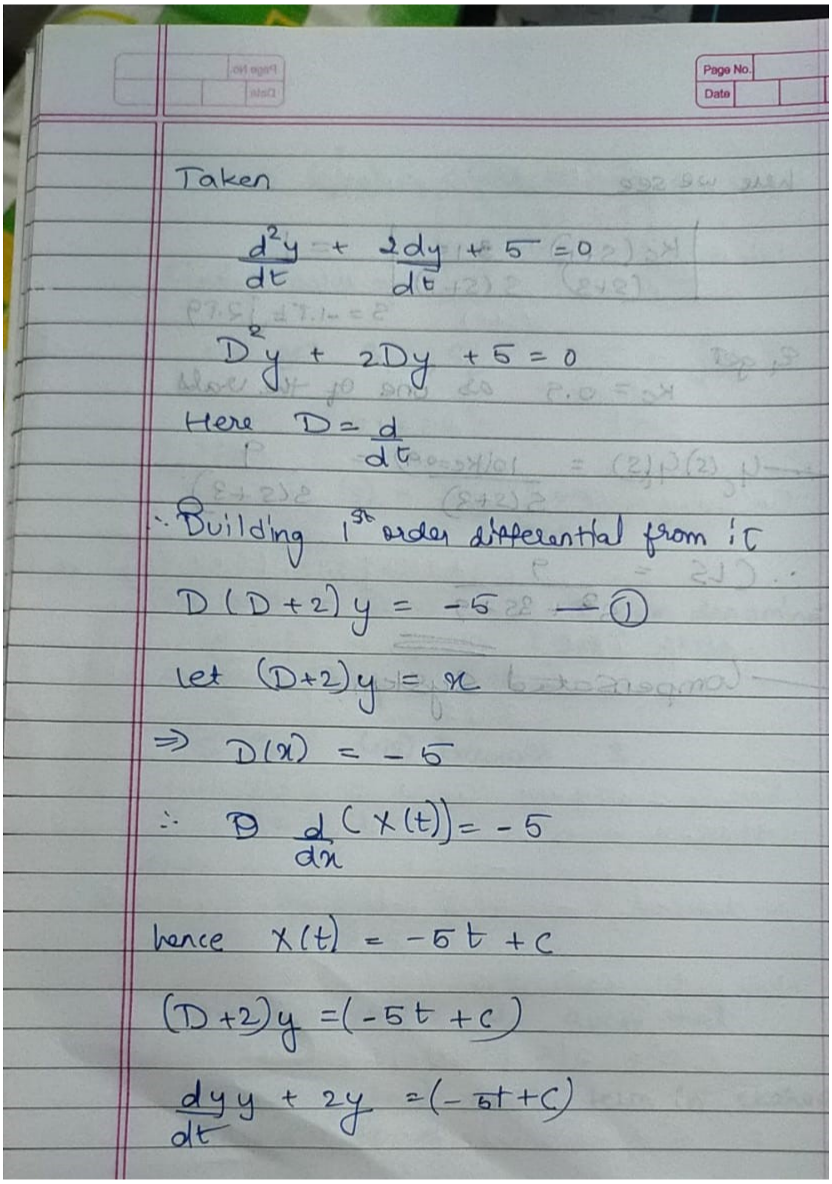


Figure 3: ODE solver for defined function



Taken

$$\frac{d^2y}{dt} + \frac{2dy}{dt} + 5 = 0$$

$$D^2y + 2Dy + 5 = 0$$

Here $D = \frac{d}{dt}$

∴ Building 1st order differential from i.c

$$D(D+2)y = -5 \quad (1)$$

let $(D+2)y = x$

$\Rightarrow D(x) = -5$

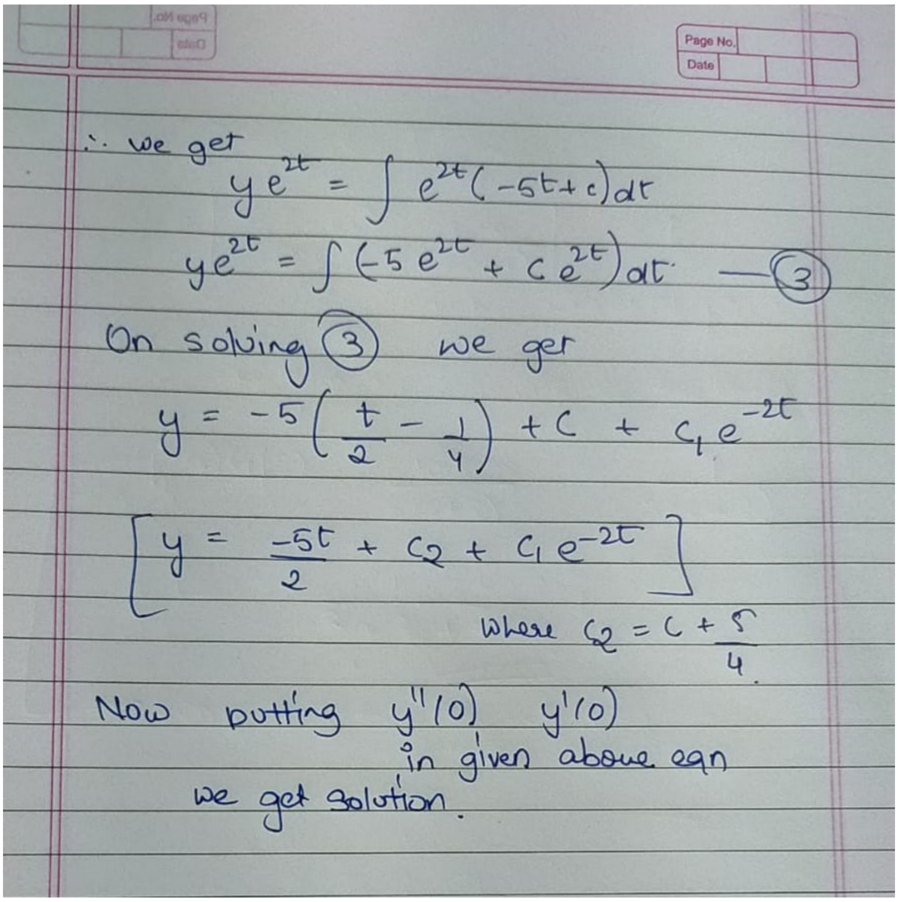
$\therefore \cancel{D}\ \frac{d}{dx}(x(t)) = -5$

hence $x(t) = -5t + c$

$$(D+2)y = (-5t + c)$$

$$\frac{dy}{dt}y + 2y = (-5t+c)$$

Figure 4: theory 1 ODE solver for defined function



Page No.
Date

∴ we get

$$y e^{2t} = \int e^{2t}(-5t + c)\,dt$$

$$y e^{2t} = \int \left(-5 e^{2t} + c e^{2t}\right) dt \quad \text{—} (3)$$

On solving (3) we get

$$y = -5\left(\frac{t}{2} - \frac{1}{4}\right) + C + C_1 e^{-2t}$$

$$\left[\, y = \frac{-5t}{2} + C_2 + C_1 e^{-2t} \,\right]$$

where $C_2 = C + \frac{5}{4}$.

Now putting $y''(0)$ $y'(0)$ in given above eqn we get solution.

Figure 5: theory2 ODE solver for defined function

Experiment-2: Time response of Zero order, 1st order and 2nd order system

Code:

```
clc;
clear all;
pkg load control;

%oth order

s = tf('s');
func = (8*s)/s;
step(func);
title("0th order step response");

%1st order
R1 = 5;
R2 = 10;
R3 = 15;
C = 50*1e-6;
func1 = tf(1, [R1*C,1]);
func2 = tf(1, [R2*C,1]);
func3 = tf(1, [R3*C,1]);
figure(1)
hold on
step(func1,'-b');
hold on
step(func2,'-r');
hold on
step(func3,'-m');
legend({'R = 5hms','R = 10ohms','R = 15ohms'});

figure(2)
step(func2,'-r');

%2nd order
R1 = 5;
R2 = 10;
R3 = 15;
C = 50*1e-6;
L = 10*1e-3;
func1 = tf(1/(L*C), [1,R1/L,1/(L*C)]);
func2 = tf(1/(L*C), [1,R2/L,1/(L*C)]);
func3 = tf(1/(L*C), [1,R3/L,1/(L*C)]);
figure(3)
step(func1,'-b');
hold on
```

```
step(func2,'-r');
hold on
step(func3,'-m');
legend({'R = 5hms','R = 10ohms','R = 15ohms'});
figure(4)
step(func2,'-r');
```

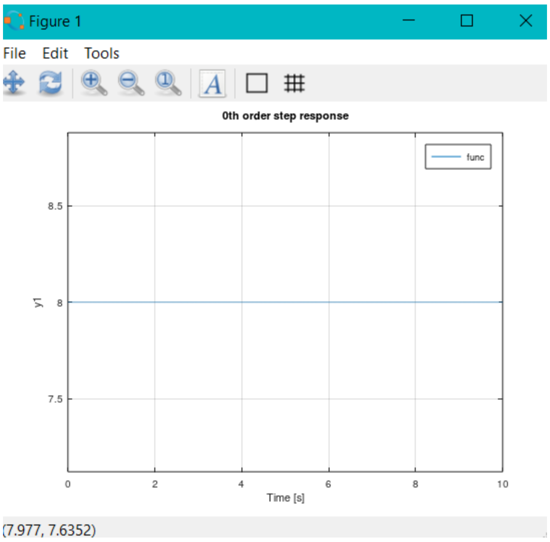


Figure 6: Step response of 0th order system

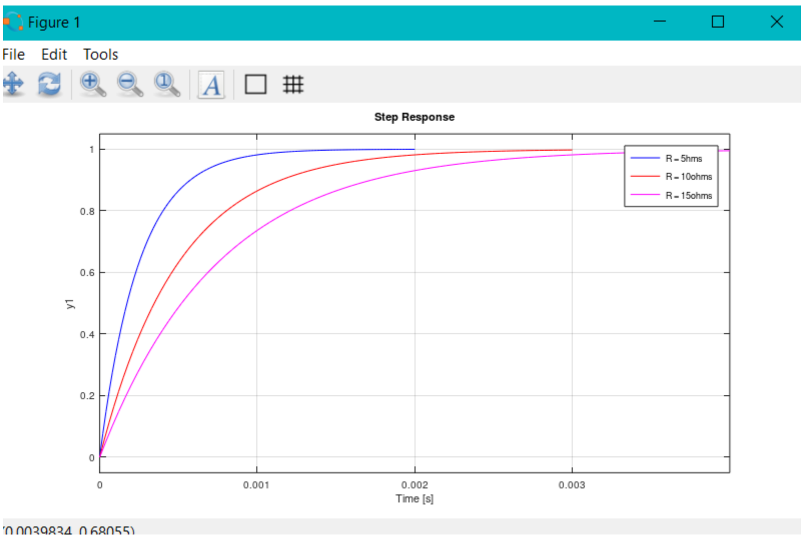


Figure 7: Step response of 1st order system

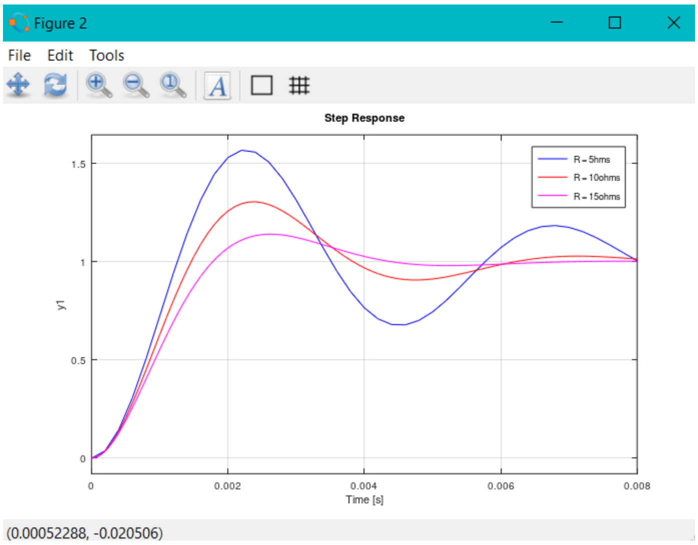


Figure 8: Step response of 2nd order system

Theory: below

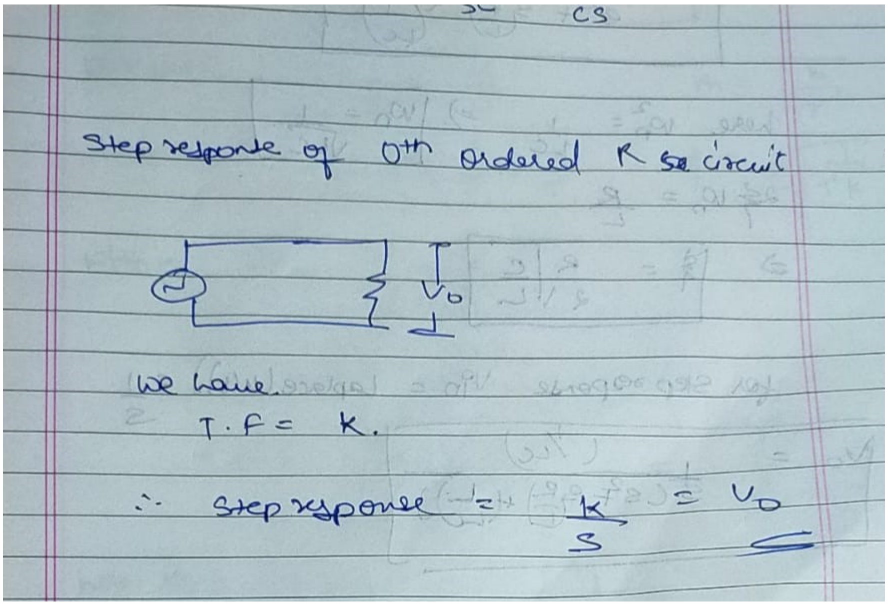


Figure 9: Step response of 0st order system

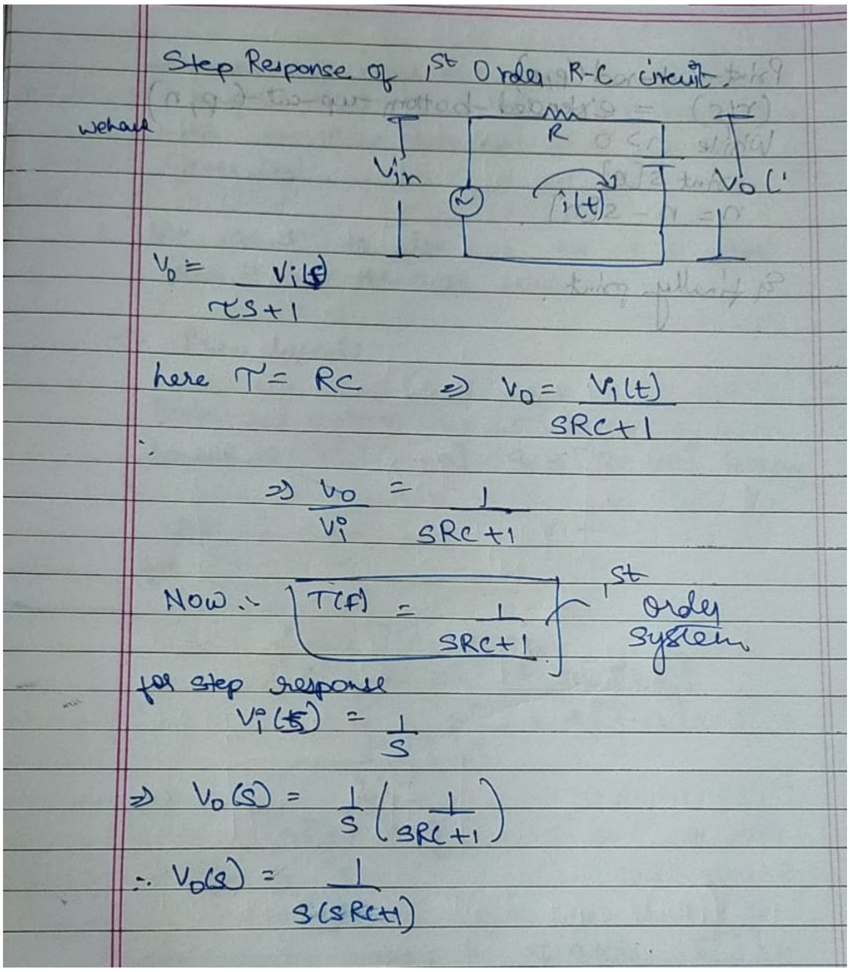


Figure 10: Step response of 1st order system

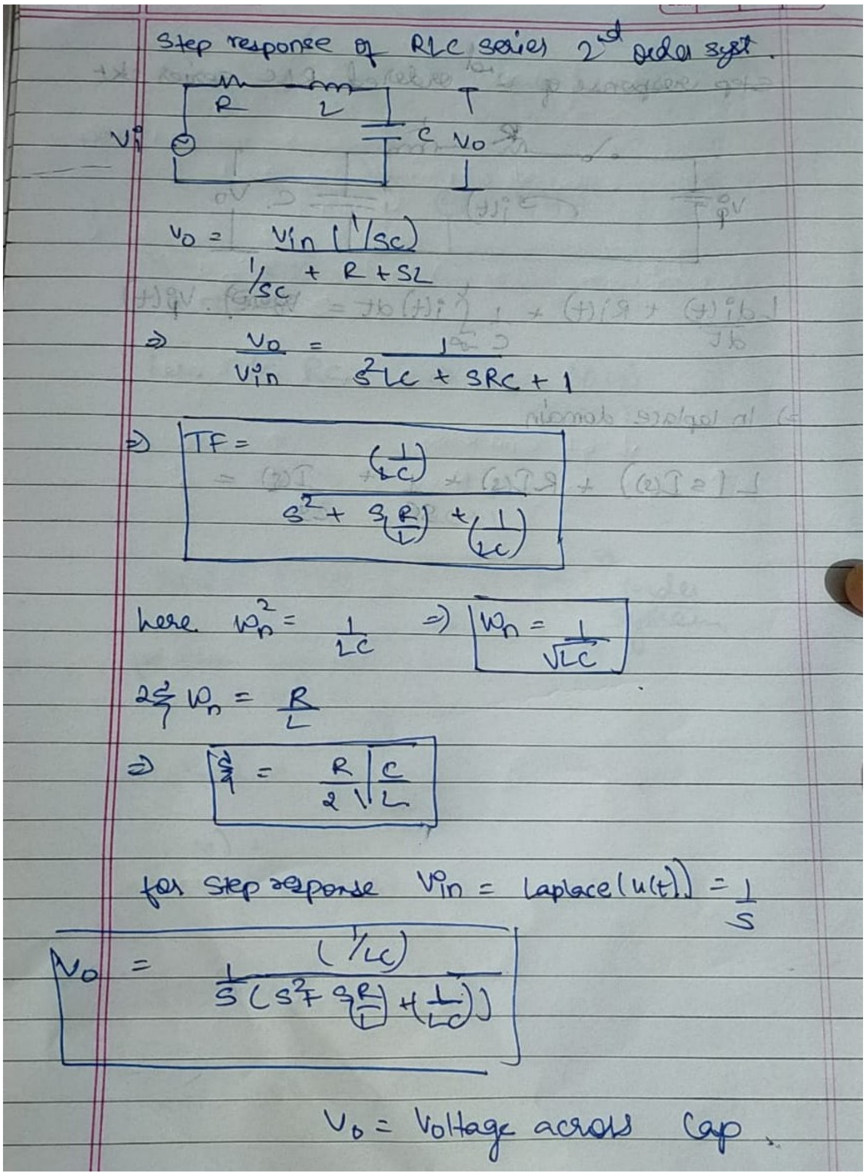


Figure 11: Step response of 2st order system

Results: a.
0. **0th order system** In Zero order system we have a constant as transfer function and hence output is linearly proportional to input i.e if there are two resistances R1 and R2 then considering o/p voltage at R2 we get transfer function

$$\frac{Vo}{Vin} = \frac{R2}{R1 + R2}$$

$$Vo = k * Vin$$

where K is our constant and hence we obtain a step function type output at y = k = 8 in our case
i. **1st order system** For the chosen system

$$G(s) = \frac{1}{sRC + 1}$$

with R = 10 ohms and C = 50micro F
On plotting stepinfo(func1) we get the performance parameters as follows

RiseTime: 0.0011 ; SettlingTime: 0.0020
SettlingMin: 0.9000 ; SettlingMax: 1.0000
Overshoot: 0 ; Undershoot: 0
Peak: 1.0000 ; PeakTime: 0.0053

ii. **2nd order system** For the chosen system

$$G(s) = \frac{\frac{1}{LC}}{s^2 + \frac{sR}{L} + \frac{1}{LC}}$$

with R = 10 ohms and C = 50micro F L = 10mH
On plotting stepinfo(func1) we get the performance parameters as follows

RiseTime: 9.8874e-04 ; SettlingTime: 0.0077
SettlingMin: 0.9071 ; SettlingMax: 1.3049
Overshoot: 30.4890 ; Undershoot: 0
Peak: 1.3049 ; PeakTime: 0.0024

b. From obtained tr, tp, Mp values we can calculate w and zeta valueand subsequently get back the R, L and C values from above relation in theory.

Conclusions: :
1. Zero order has a simple scaled step function utput as there is linear relation

between input u(t) and output

1. **First order system**
We have the time constant given by

$$\tau = RC$$

. Hence on increasing Resitance from 5 to 15 there is an increase in time constant causing the step response to reach steady state at higher time i.e. as R increases , time required to reach steady state also increases.

2. **Second order system**
Thus we have seen that by changing the resistance value in 1st order system we are effectively changing the zeta value while keeping w constant . As the resistance decrease, there is a decrease in zeta value given by

$$\frac{R}{2}\sqrt{\frac{C}{L}}$$

. As a result there is increase in oscillations leading to higher settling time i.e time to reach steady state.

Experiment-3: Study the effect of p controlled in process control simulator

Code:

```
clr;
clear all;
pkg load control;



p_contr1 = pid(2);
p_contr2 = pid(3);
p_contr3 = pid(8);
p_contr4 = pid(15);


%p control of 2nd order system
func1 = tf(100,[1,10,100])
fb_func1 = feedback(func1,[1]);
stepinfo(fb_func1);
p1 = feedback(func1*p_contr1,[1]);
p2 = feedback(func1*p_contr2,[1]);
stepinfo(p2);
p3 = feedback(func1*p_contr3,[1]);
p4 = feedback(func1*p_contr4,[1]);


figure(1)
step(fb_func1,'-m');
hold on
step(p1,'-r');
hold on
step(p2,'-b');
hold on
step(p3,'-g');
hold on
step(p4,'-c');
legend({'Closed loop (without p ...
    control)','kp=2','kp=3','kp=8','kp=15'})
```

Graph:

Theory:

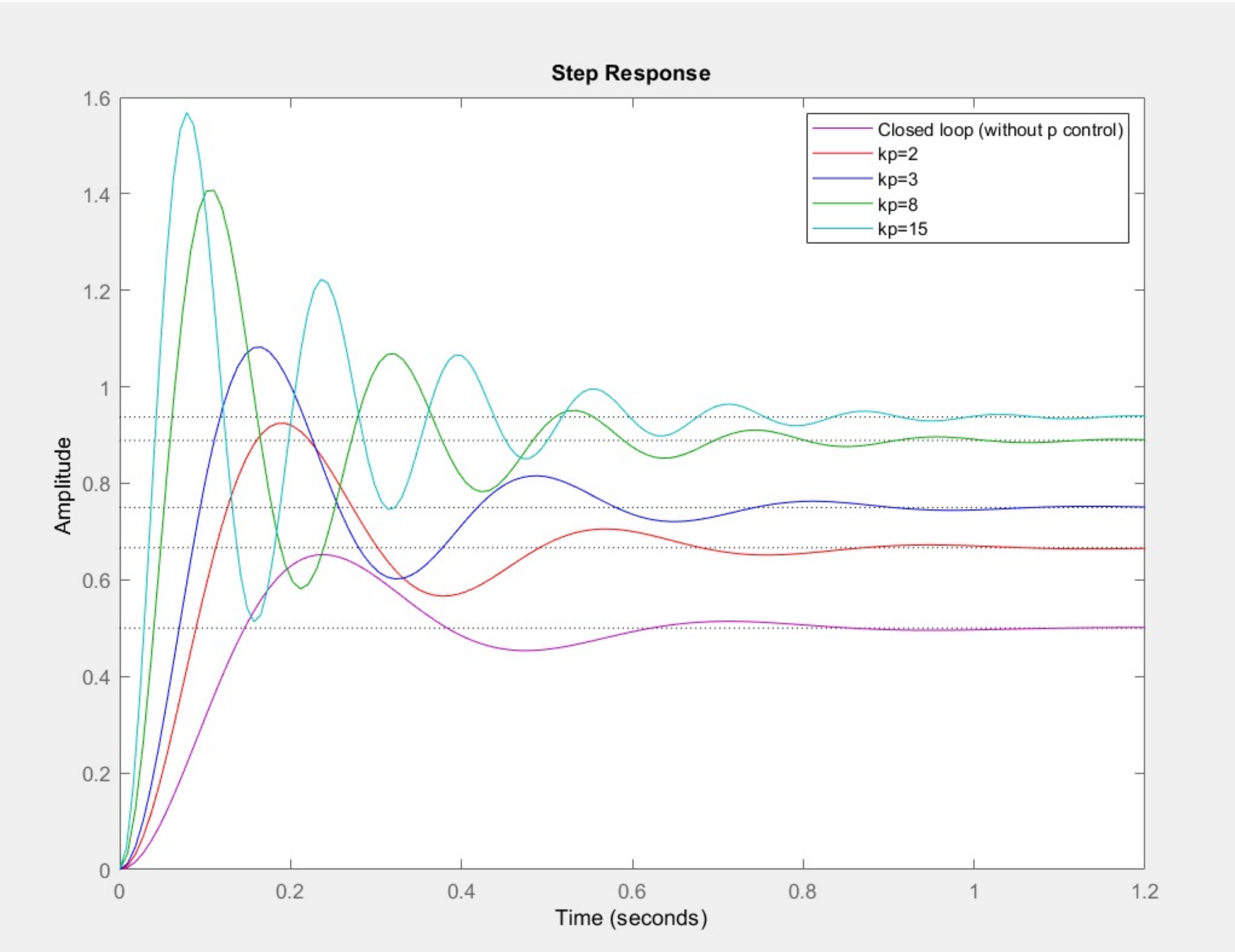


Figure 12: P control for kp¿0

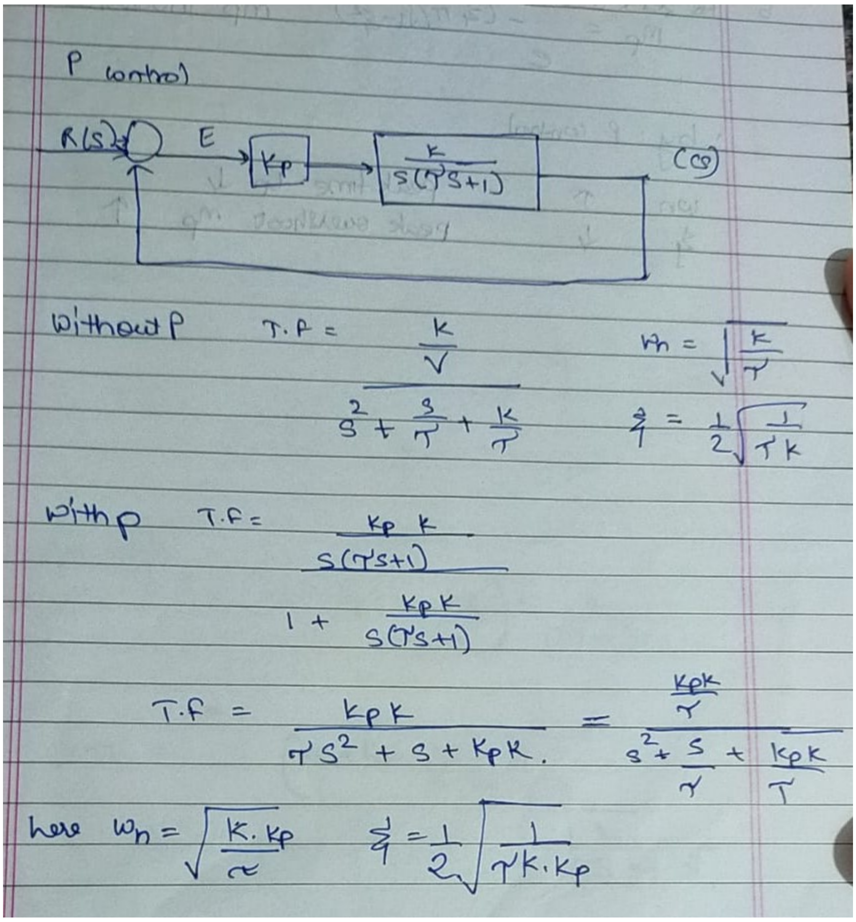
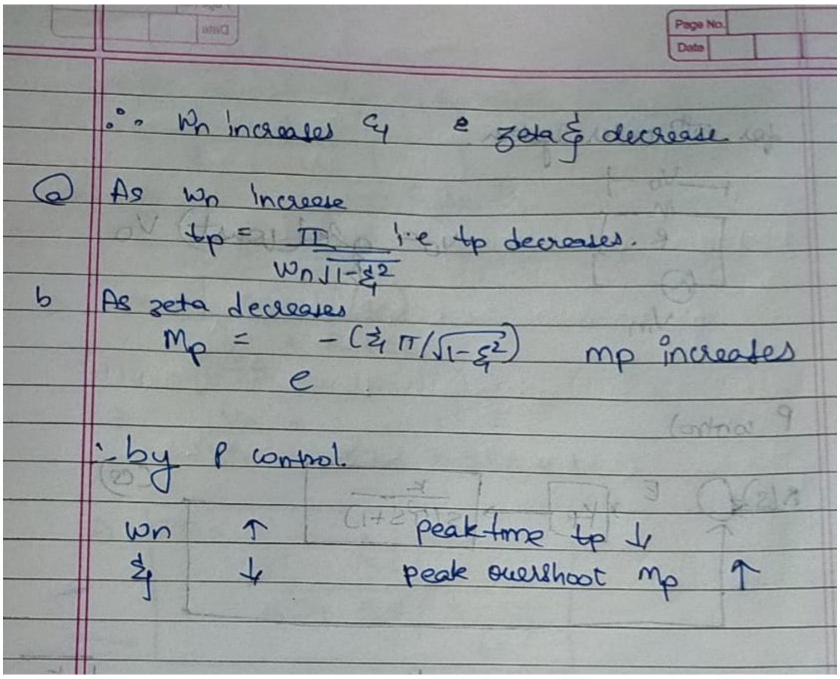


Figure 13: Step response of 2st order system



∴ ω_n increases ξ & zeta ξ decrease

a) As ω_n increase

$t_p = \frac{\pi}{\omega_n\sqrt{1-\xi^2}}$ i.e. t_p decreases.

b As zeta decreases

$M_p = e^{-(\xi\pi/\sqrt{1-\xi^2})}$ mp increases

∴ by P control.

ω_n	↑	peak time t_p	↓
ξ	↓	peak overshoot M_p	↑

Figure 14: Step response of 2st order system

Results: We considered open loop transfer function

$$G(s) = \frac{100}{s^2 + 10s + 100}$$

and plotted step response of closed loop system without p control n that for 4 different Kp values. on plotting stepinfo() for various responses we get the performance parameters as

i. Closed loop system

RiseTime: 0.1639 ; SettlingTime: 0.8076
SettlingMin: 0.9315 ; SettlingMax: 1.1629
Overshoot: 16.2929 ; Undershoot: 0
Peak: 1.1629 ; PeakTime: 0.3592

i. With p control
a. Kp= 2
RiseTime: 0.0756 ; SettlingTime: 0.7870
SettlingMin: 0.5664 ; SettlingMax: 0.9246
Overshoot: 38.6910 ; Undershoot: 0

Peak: 0.9246 ; PeakTime: 0.1934

b. Kp=15
RiseTime: 0.0288 ; SettlingTime: 0.7327
SettlingMin: 0.5132 ; SettlingMax: 1.5684
Overshoot: 67.2931 ; Undershoot: 0
Peak: 1.5684 ; PeakTime: 0.0785

Conclusions: We observe that by introducing p control i.e Kp in the transfer function, the peak time decreases which is what we desire,
but at the same tile the peak value and peak overshoot also increases
leading more and more oscillatory nature in the step response .
Hence we need another set of controls to prevent this increase in peak overshoot all the while reducing tp.

Experiment-4: Study the effect of pid controlled in process control simulator

Code:

```
clear all;
pkg load control;


pid1 = pid(5,8,0.1);
pid2 = pid(5,10,0.06);
pid3 = pid(5,13,0.1);
pid4 = pid(5,10,0.3);


%p control of 2nd order system
func1 = tf(100,[1,10,100])
fb_func1 = feedback(func1,[1]);
stepinfo(fb_func1)
p1= feedback(func1*pid1,[1]);
stepinfo(p1)
p2= feedback(func1*pid2,[1]);
stepinfo(p2)
p3= feedback(func1*pid3,[1]);
stepinfo(p3)
p4= feedback(func1*pid4,[1]);
stepinfo(p4)
figure(1)
step(fb_func1)
figure(2)
step(p1);hold on, step(p2);holdon, step(p3);holdon, step(p4)
title('PID control for varying kd and ki at kp=5');
legend({'kd=8,ki=0.1','kd=10,ki=0.06','kd=13,ki=0.1','kd=10,ki=0.1'});

% PID controller Matlab defined values
[tf_pid,param_pid] = pidtune(func1,'PID');
pid_sysdef = feedback(tf_pid*func1,[1]);
stepinfo(pid_sysdef);

disp('PID controller details');
disp(tf_pid); %TF of PI controller
disp(param_pid);

figure(3)
step(fb_func1,[1],'-b');
hold on
step(feedback(tf_pid*func1,[1]),'-r');
title('System defined PID control');
```

```
legend({'Closed Loop Syste','PID Comtrol'})
```

Graph: plots

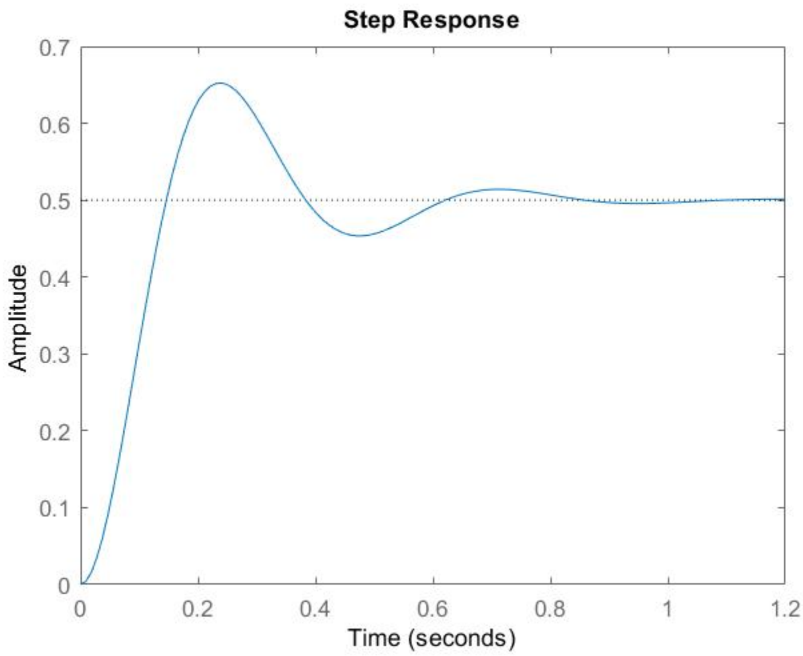


Figure 15: Closed loop system

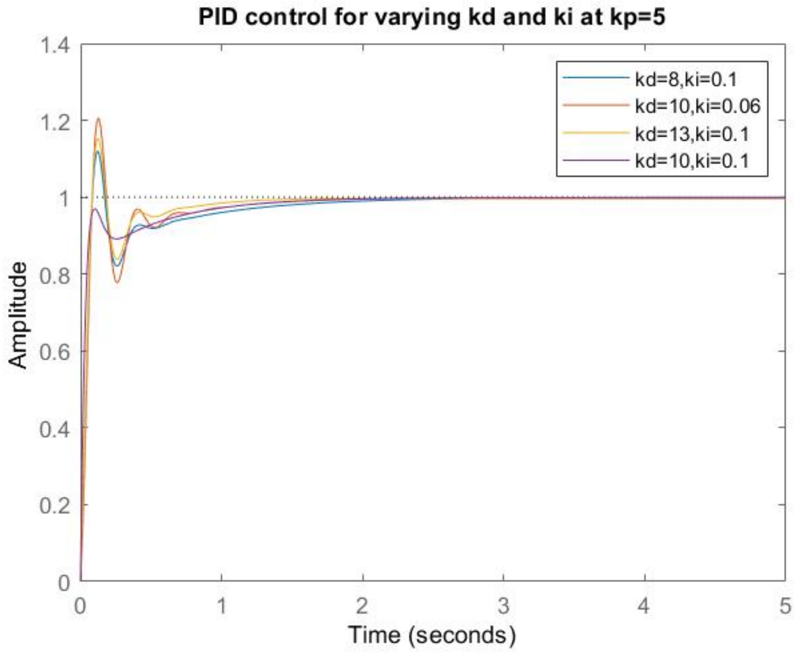


Figure 16: Pid control for different values

Theory: written

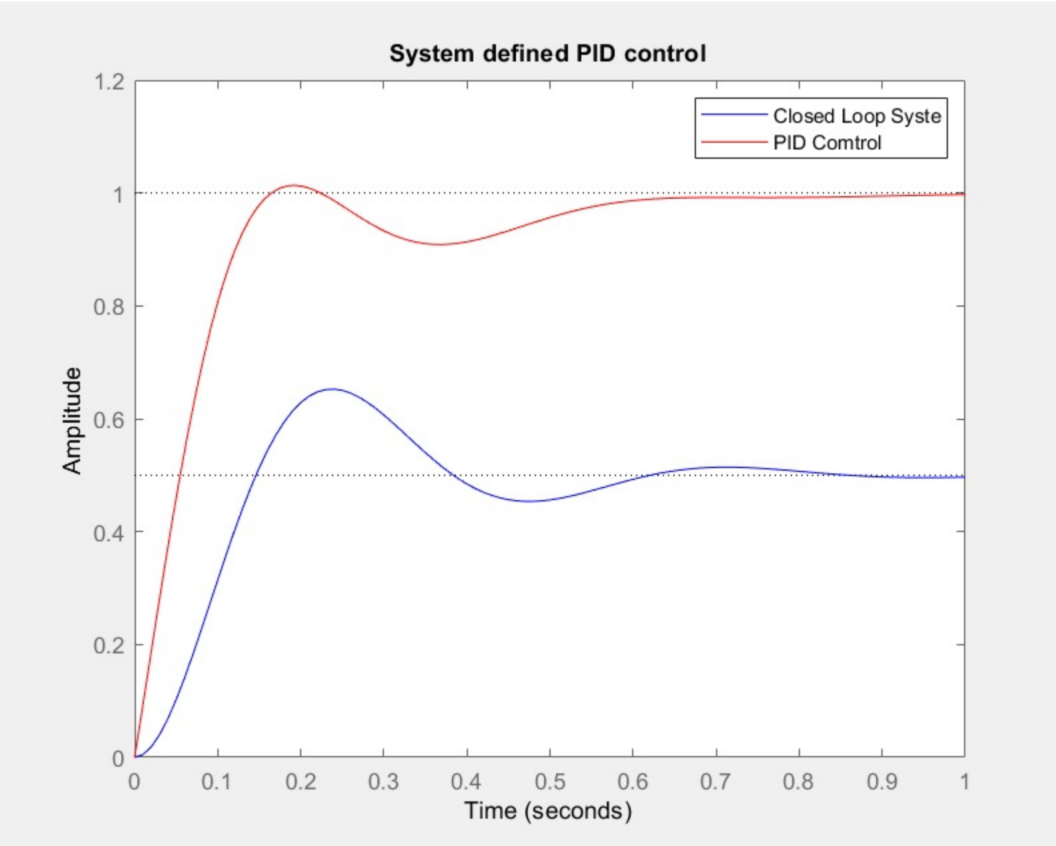
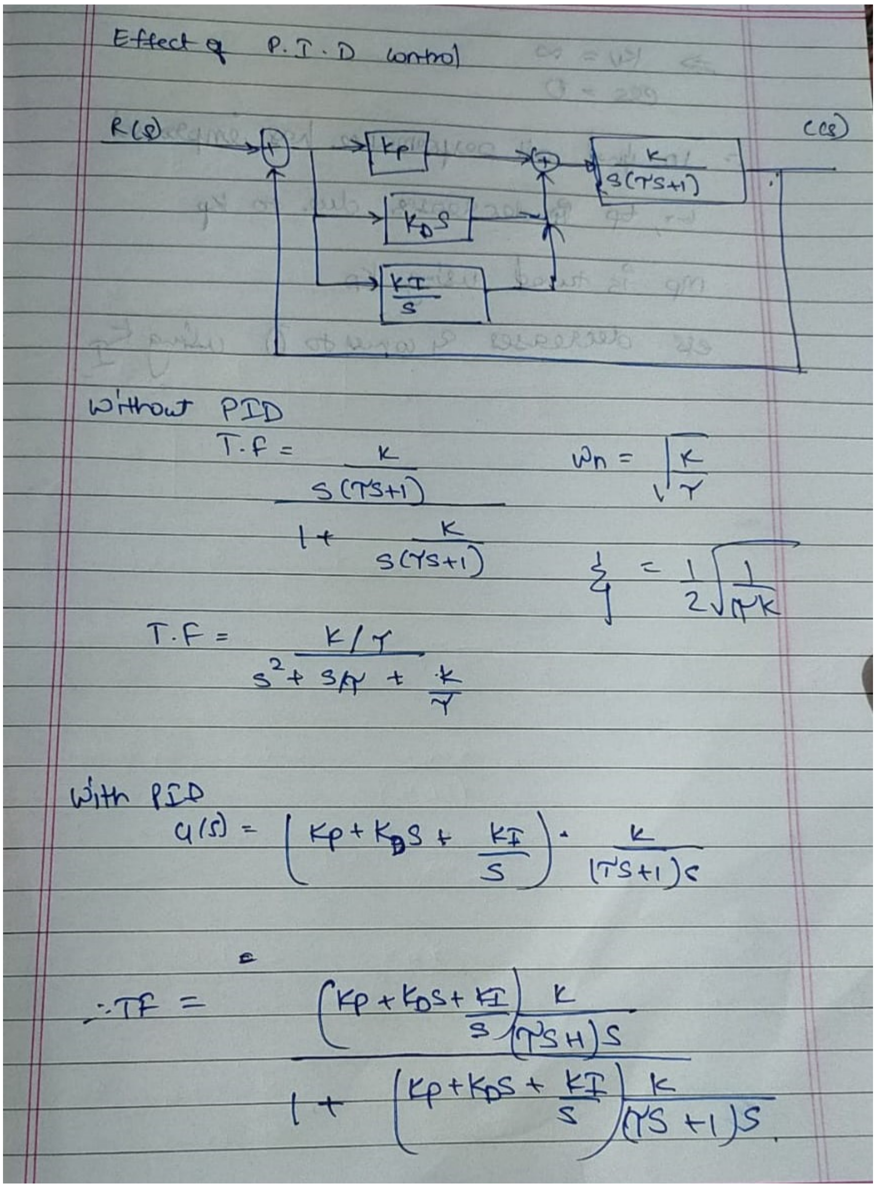


Figure 17: Pid control



Without PID

$$T.F = \frac{\frac{K}{S(\tau S+1)}}{1+\frac{K}{S(\tau S+1)}}$$

$$\omega_n = \sqrt{\frac{K}{\tau}}$$

$$\xi = \frac{1}{2}\sqrt{\frac{1}{\tau K}}$$

$$T.F = \frac{K/\tau}{S^2 + S/\tau + \frac{K}{\tau}}$$

With PID

$$G(S) = \left(K_P + K_D S + \frac{K_I}{S}\right) \cdot \frac{K}{(\tau S+1)S}$$

$$\therefore TF = \frac{\left(K_P + K_D S + \frac{K_I}{S}\right)\frac{K}{(\tau S+1)S}}{1 + \left(K_P + K_D S + \frac{K_I}{S}\right)\frac{K}{(\tau S+1)S}}$$

Figure 18: Step response of 2st order system

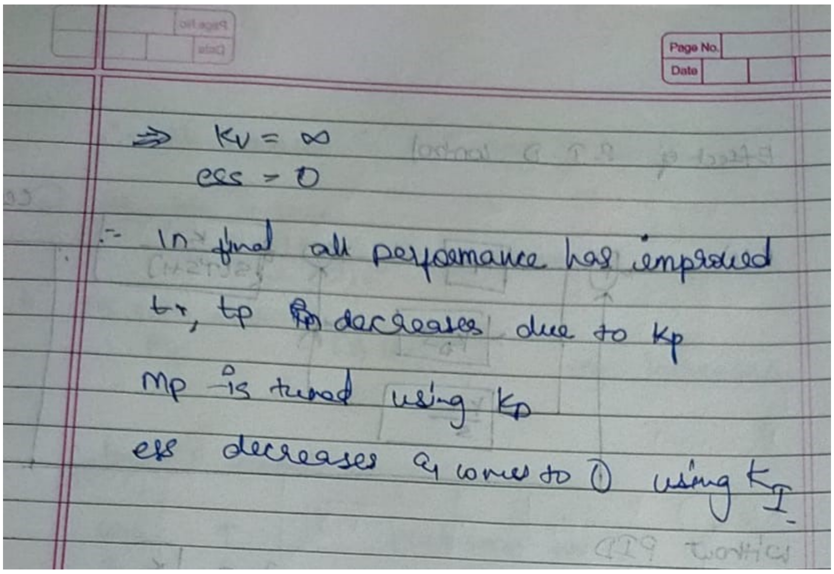


Figure 19: Step response of 2st order system

Results: We considered open loop transfer function

$$G(s) = \frac{100}{s^2 + 10s + 100}$$

and plotted step response of closed loop system without pid control n that for pid control on plotting stepinfo() for various responses we get the performance parameters as

i. Closed loop system

RiseTime: 0.1639 ; SettlingTime: 0.8076
SettlingMin: 0.9315 ; SettlingMax: 1.1629
Overshoot: 16.2929 ; Undershoot: 0
Peak: 1.1629 ; PeakTime: 0.3592

i. With pid control
Kp: 2.0951 , Ki: 11.9203 , Kd: 0.0921

RiseTime: 0.1104 ; SettlingTime: 0.5671
SettlingMin: 0.9087 ; SettlingMax: 1.0135
Overshoot: 1.3478 ; Undershoot: 0
Peak: 1.0135 ; PeakTime: 0.1935

Conclusions: We observe that by introducing pid control we are able to tune all the performance parameters, by tuning Kp we have tuned the rise time and peak time of the response, by tuning the Kd we have prevented mp i.e peak overshoot fro increasing very much and by tuning pi we have made sure that ess i.e steady state error becomes 0.

Experiment-5a: Study the effect Lead Compensator with uncompensated function

Code: Lead Compensator

```
clc;
clear all;
pkg load control;

G_s=tf(10,[1,1,0]);
figure(1)
rlocus(G_s);

G_cc = 0.9*tf([1,1],[1,3]);
%Gc=0.9*(s+1)/(s+3);
figure(2)
rlocus(G_s*G_cc);5

func_G = feedback(G_s,[1]);
func_Gcomp = feedback(G_s*G_cc,[1]);
figure(3)
step(func_G,'-r');
hold on
step(func_Gcomp,'-b');
title("Step response of compensated and uncompendsated system");
legend({'Uncompensated system','Compensated System'});
```

Graph:

Theory: below

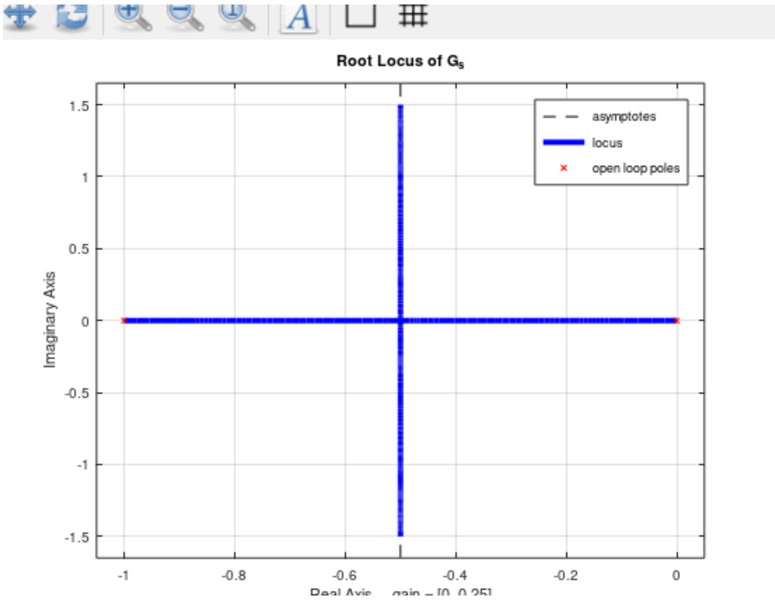


Figure 20: root locus of uncompensated$_{o}penloop$

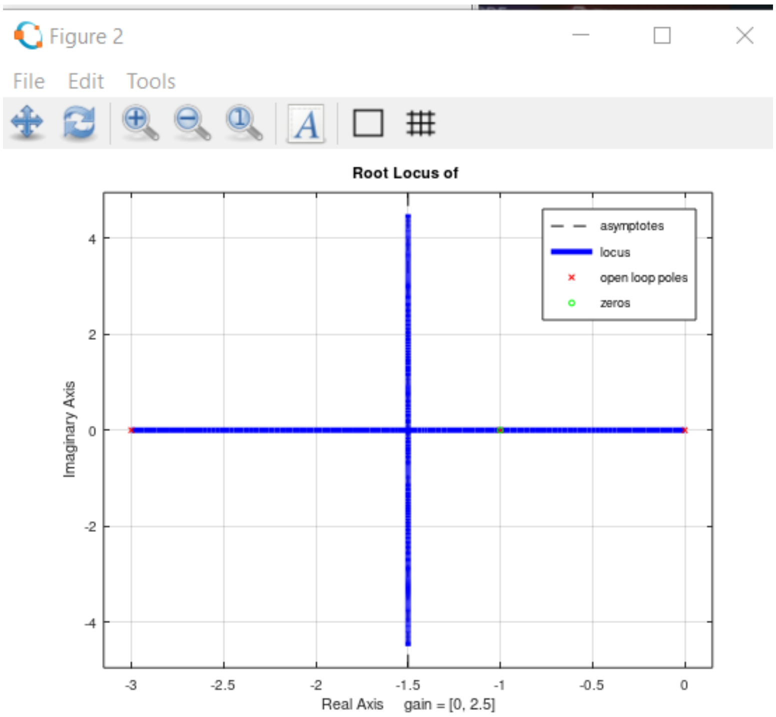


Figure 21: root locus of Lead compensator Gc(s)

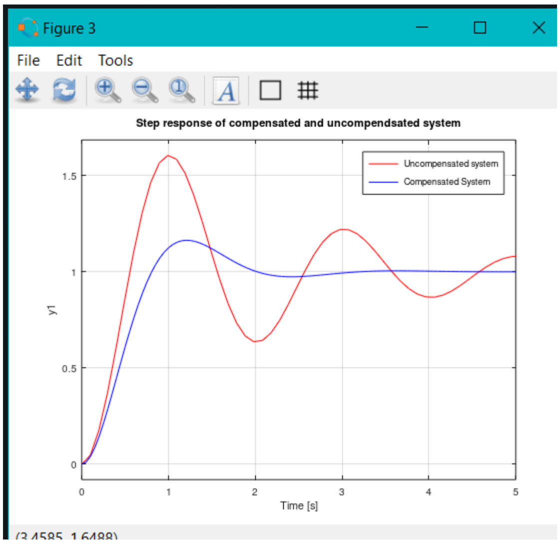
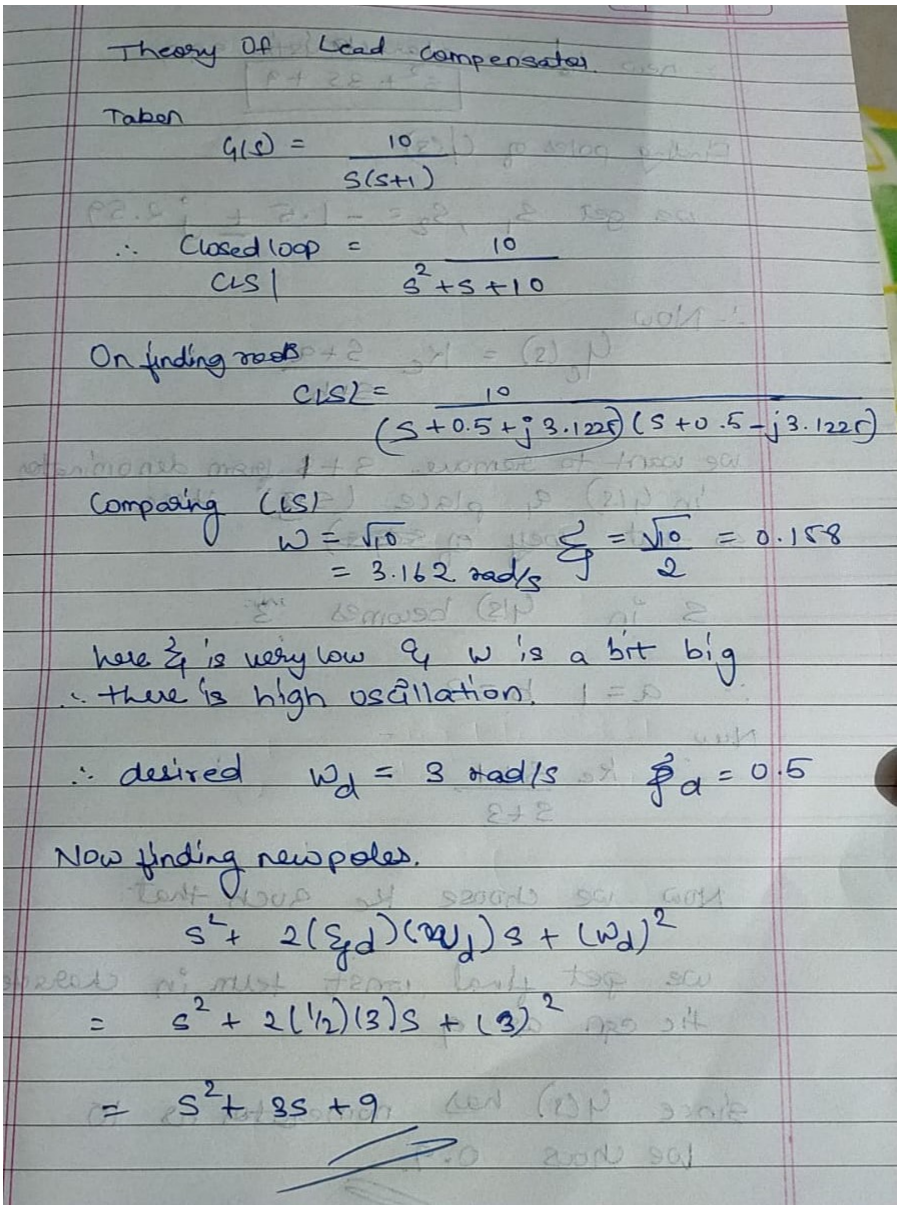


Figure 22: Step response

<u>Results</u>: 1. For lead compensator we saw that the root locus changes. There is evident decrease in oscillations and the compensated system reaches steady state at a shorter time as compared to uncompensated one.



Theory of Lead compensator.

Taken

$$G(s) = \frac{10}{s(s+1)}$$

$\therefore$ Closed loop CLS $= \dfrac{10}{s^2+s+10}$

On finding roots

$$CLS = \frac{10}{(s+0.5+j3.1225)(s+0.5-j3.1225)}$$

Comparing CLS

$\omega = \sqrt{10} = 3.162$ rad/s $\xi = \dfrac{\sqrt{10}}{2} = 0.158$

here ξ is very low & ω is a bit big
$\therefore$ there is high oscillation.

$\therefore$ desired $\omega_d = 3$ rad/s $\xi_d = 0.5$

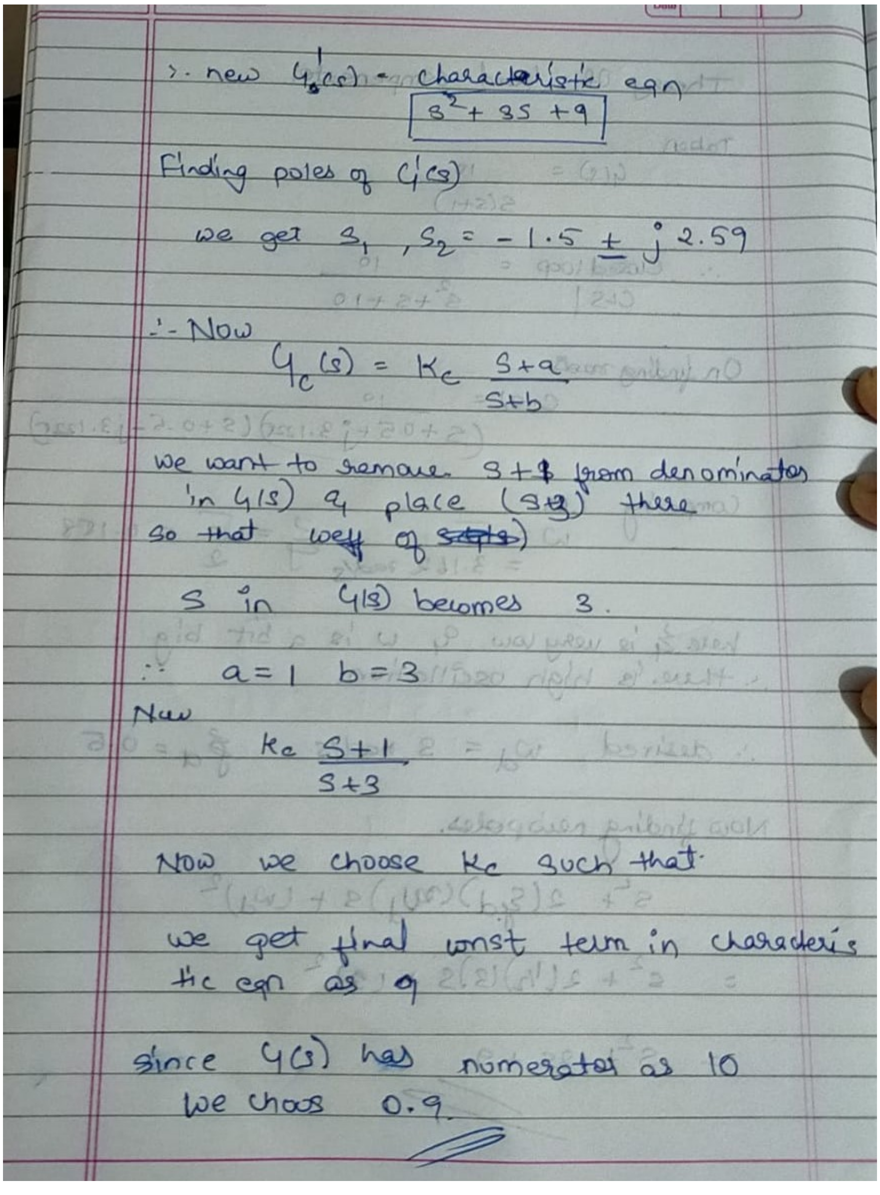
Now finding new poles.

$$s^2 + 2(\xi_d)(\omega_d)s + (\omega_d)^2$$

$$= s^2 + 2(1/2)(3)s + (3)^2$$

$$= s^2 + 3s + 9$$

Figure 23: Lead compensator expl.1



∴ new $G_c'(s)$ = characteristic eqn

$$\boxed{s^2 + 3s + 9}$$

Finding poles of $G'(s)$

we get $s_1, s_2 = -1.5 \pm j2.59$

∴ Now

$$G_c(s) = K_c \frac{s+a}{s+b}$$

we want to remove $s+1$ from denominator in $G(s)$ & place $(s+3)$ there so that ~~that~~ coeff of ~~s~~)

s in $G(s)$ becomes 3.

∴ $a = 1 \quad b = 3$

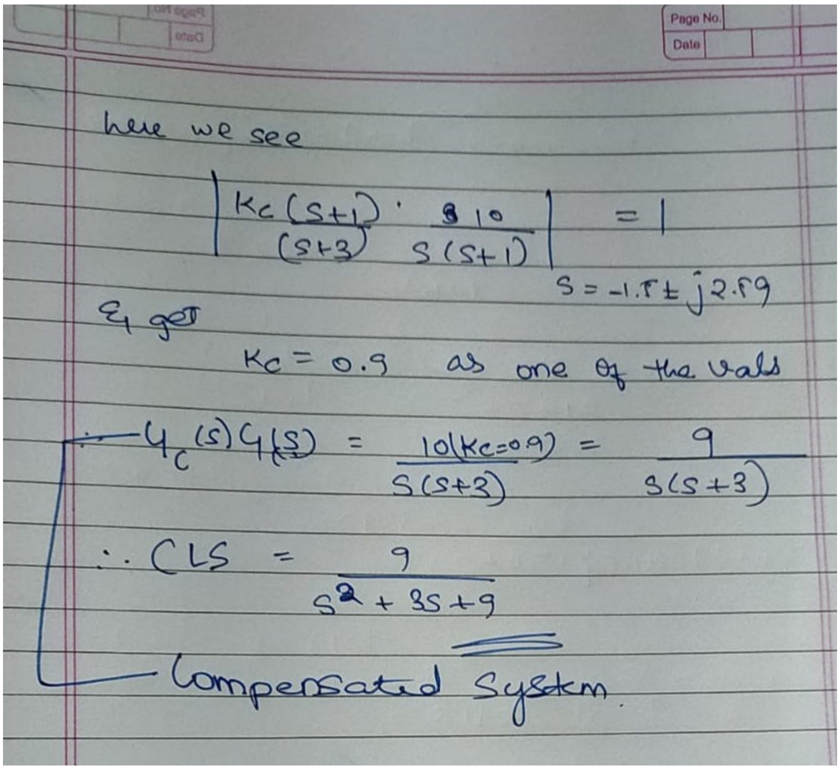
Now

$$K_c \frac{s+1}{s+3}$$

Now we choose K_c such that we get final const term in characteristic eqn as 9

since $G(s)$ has numerator as 10 we choos 0.9.

Figure 24: Lead compensator expl.2



here we see

$$\left|\frac{K_c(s+1)}{(s+3)} \cdot \frac{10}{s(s+1)}\right| = 1$$

$s = -1.5 \pm j2.59$

we get

$K_c = 0.9$ as one of the vals

$$\therefore G_c(s)G(s) = \frac{10(K_c=0.9)}{s(s+3)} = \frac{9}{s(s+3)}$$

$$\therefore CLS = \frac{9}{s^2+3s+9}$$

Compensated System.

Figure 25: Lead compensator expl.3

Experiment-5b: Study the effect Lag Compensator with uncompensated function

Code: Lag Compensator

```
clc;
clear all;
pkg load control;


G_s=tf(1.06,[1,3,2,0]);
figure(1)
func_G = feedback(G_s,[1]);
rlocus(func_G);

G_cc = tf([1,0.03],[1,0.003]);


func_Gcomp = feedback(G_s*G_cc,[1]);
figure(2)
rlocus(G_cc*G_s);
figure(3)
step(func_G ,'-r');
hold on
step(func_Gcomp,'-b');
title("Step response of lag compensated and uncompendsated system");
legend({'Uncompensated system','Compensated System'});
```

Graph:

Theory: below

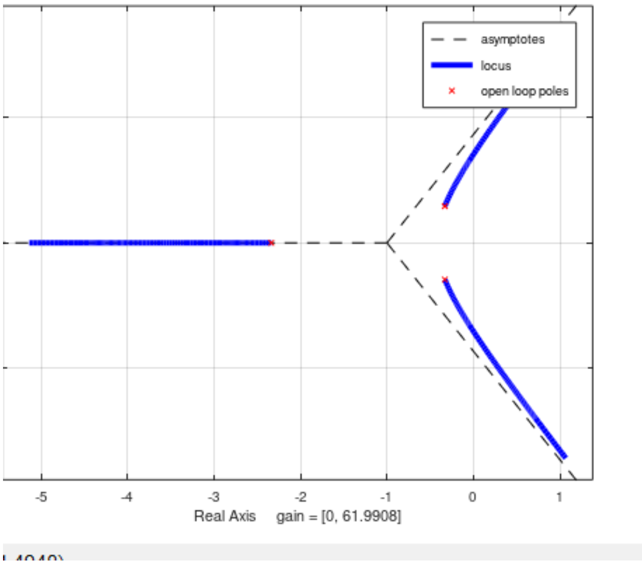


Figure 26: root locus of uncompensated closed loop

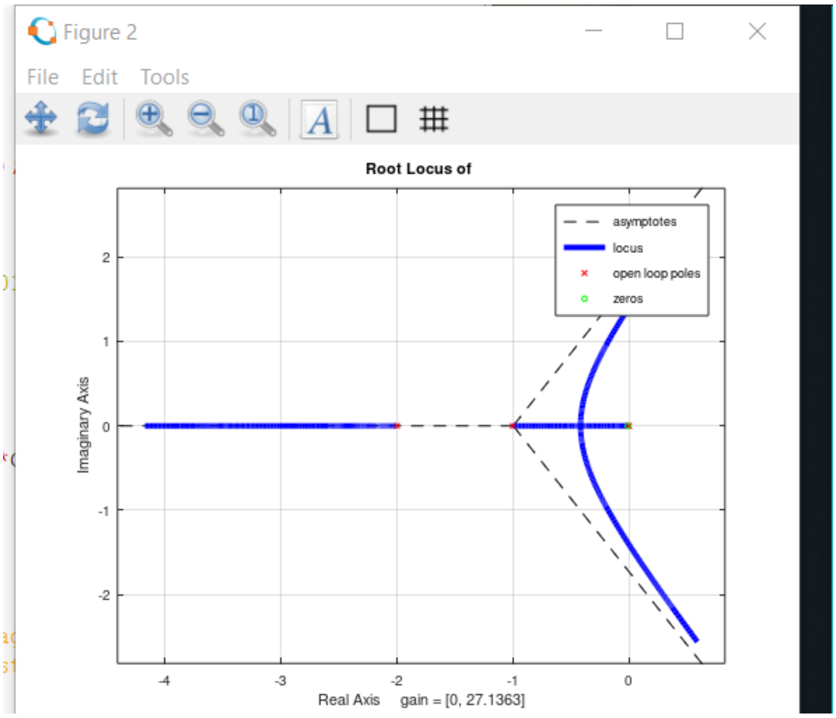


Figure 27: root locus of Lag compensator Gc(s)

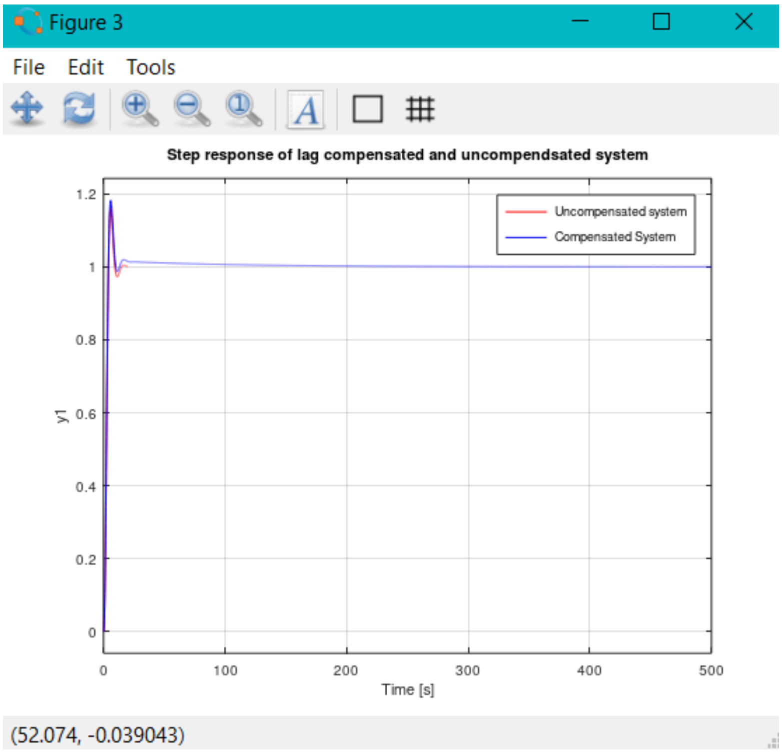
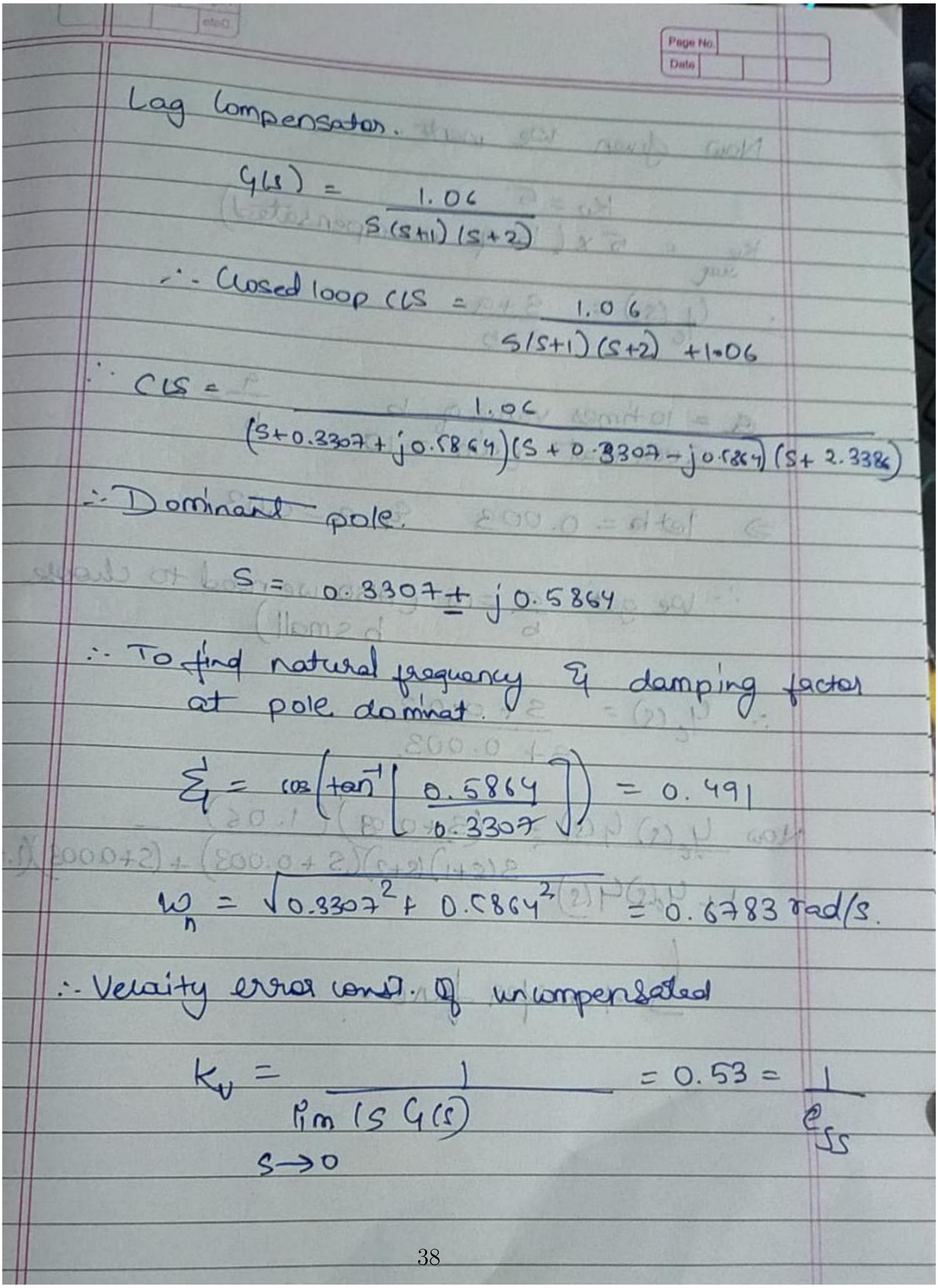


Figure 28: Step response

Results: 2. For lag compensator we see that the root locus does not overlap.Moreover we see that the zeros of the compensated system comes very near the origin but at the left side of jw axis as desired.
And we are able to get the desired velocity value.

Conclusion: Hence we were able to perform lead and lag compensation.



Lag Compensator.

$$G(s) = \frac{1.06}{s(s+1)(s+2)}$$

$\therefore$ Closed loop $C/S = \frac{1.06}{s(s+1)(s+2) + 1.06}$

$\therefore$ $C/S = \frac{1.06}{(s+0.3307+j0.5864)(s+0.3307-j0.5864)(s+2.3386)}$

$\therefore$ Dominant pole.

$$s = 0.3307 \pm j0.5864$$

$\therefore$ To find natural frequency & damping factor at pole dominat.

$$\xi = \cos\left(\tan^{-1}\left[\frac{0.5864}{0.3307}\right]\right) = 0.491$$

$$\omega_n = \sqrt{0.3307^2 + 0.5864^2} = 0.6783 \text{ rad/s.}$$

$\therefore$ Velocity error const. of uncompensated

$$K_v = \frac{1}{\lim_{s \to 0} (s\,G(s))} = 0.53 = \frac{1}{e_{ss}}$$

Figure 29: Lag compensator expl.1

Now given we want.

$$K_v = 5$$

$$K_{v_{req}} = 5 \times (K_v \text{ of uncompensated})$$

$$\therefore \; G_c(s) = \frac{s+a}{s+b}$$

a = 10 times value of b such that gain = 10 $\quad \frac{a}{b} = 10$

$\Rightarrow$ let b = 0.003 & a = 0.03

$\therefore$ We get $\frac{a}{b} = 10$ (Also we need to choose b small)

$$\therefore \; G_c(s) = \frac{s+0.03}{s+0.003}$$

Now $$\frac{G_c(s)G(s)}{1+G_c(s)G(s)} = \frac{(s+0.03)(1.06)}{s(s+1)(s+2)(s+0.003)+(s+0.003)(1.06)}$$

↳ transfer func.

Figure 30: Lag compensator expl.2