PID controller

Lab: 10

Fall 2022

CSE-3L Control Systems

Submitted by: **Amir Suliman**

Registration No: **19PWCSE1805**

Class Section: **B**

"On my honor, as a student of the University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Student Signature: ______________

Submitted to:

Dr: Muniba Ashfaq

January 11, 2023

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Objectives:

- To know about PID controller.

Steady Error:

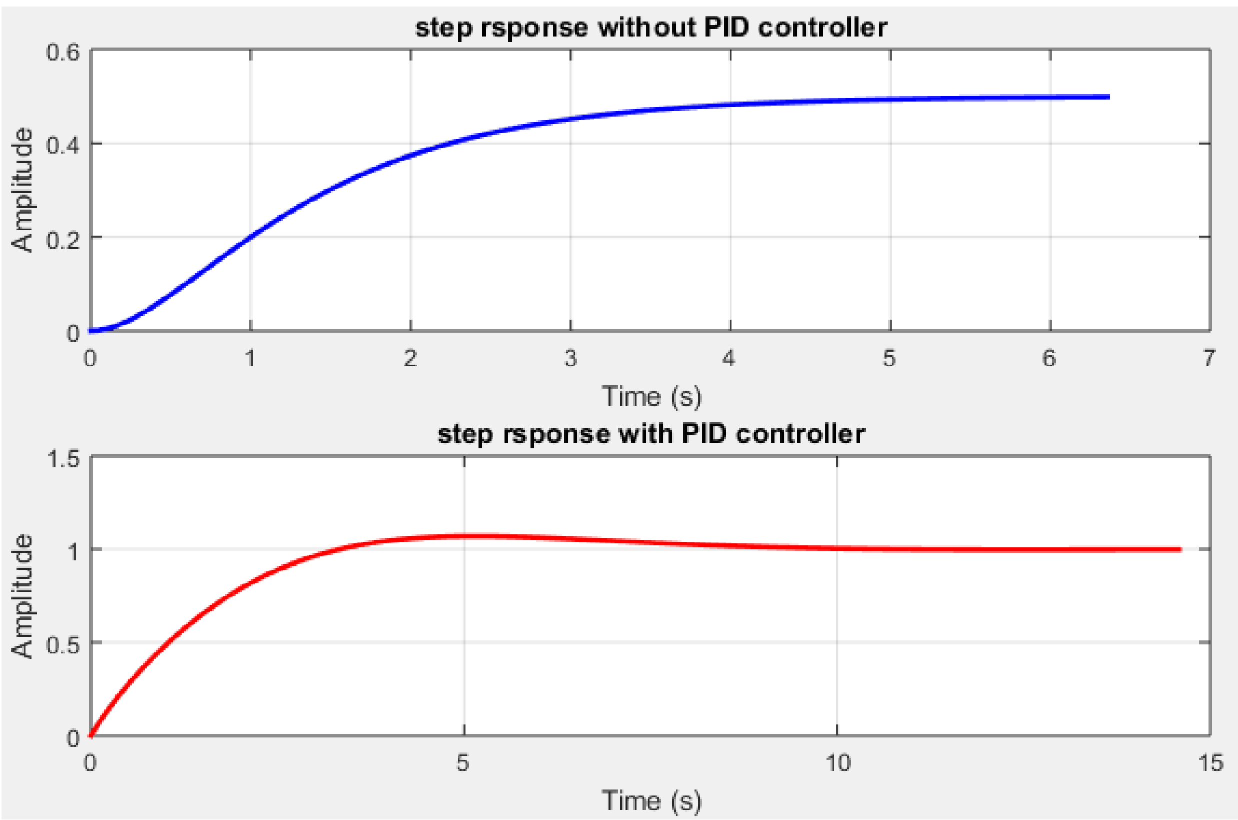
In control systems, the steady-state error measures how closely a system's output follows the input after any transients have died out. It is defined as the difference between the desired output and the actual output of a system when the input is constant. There are three types of steady-state error: steady-state error for a step input, steady-state error for a ramp input, and steady-state error for a parabolic input. In general, the steady-state error decreases as the system becomes more accurate and the control system becomes more precise.

Code:

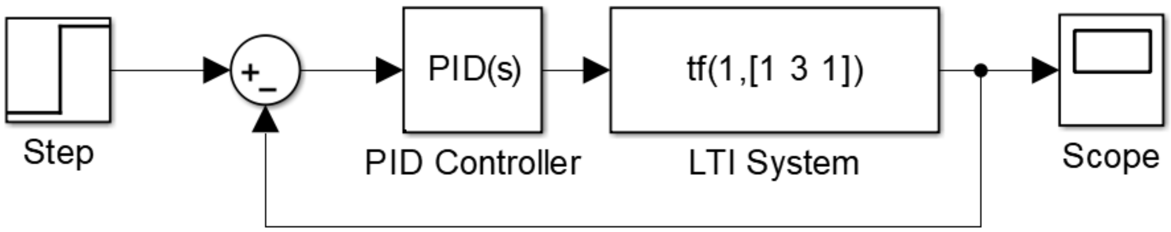
```
clc
clear
close all;

% Make first system
num = [1];
denum = [1 3 1];
G = tf(num, denum);

% First make negative feedback sytem
Feedback_system = feedback(G,1);

% This is the step rsponse without PID controller
[Feedback_response,t] = step(Feedback_system);


% these are pid coefficients
% changing these values will change the step response
kp = 12;
ki = 999999999999;
kd = -99999999999;

% pid_system = pid(kp, ki, kd);
% now use pidtune
pid_system = pidtune(G,'pid');

% Here G is in series with PID in the forward path
controlled_feedback_system = feedback(G*pid_system,1);

% This is the plot with pid
[Controlled_Feedback_response,t1] = step(controlled_feedback_system);

% Plot the input signal and the system's response
figure
subplot(2,1,1);
```

```
plot(t,Feedback_response,'b','LineWidth',2);
xlabel('Time (s)');
ylabel('Amplitude');
title('step rsponse without PID controller');
grid on

subplot(2,1,2);
plot(t1,Controlled_Feedback_response,'r','LineWidth',2);
xlabel('Time (s)');
ylabel('Amplitude');
title('step rsponse with PID controller');
grid on
```

Output:



Simulink Design:



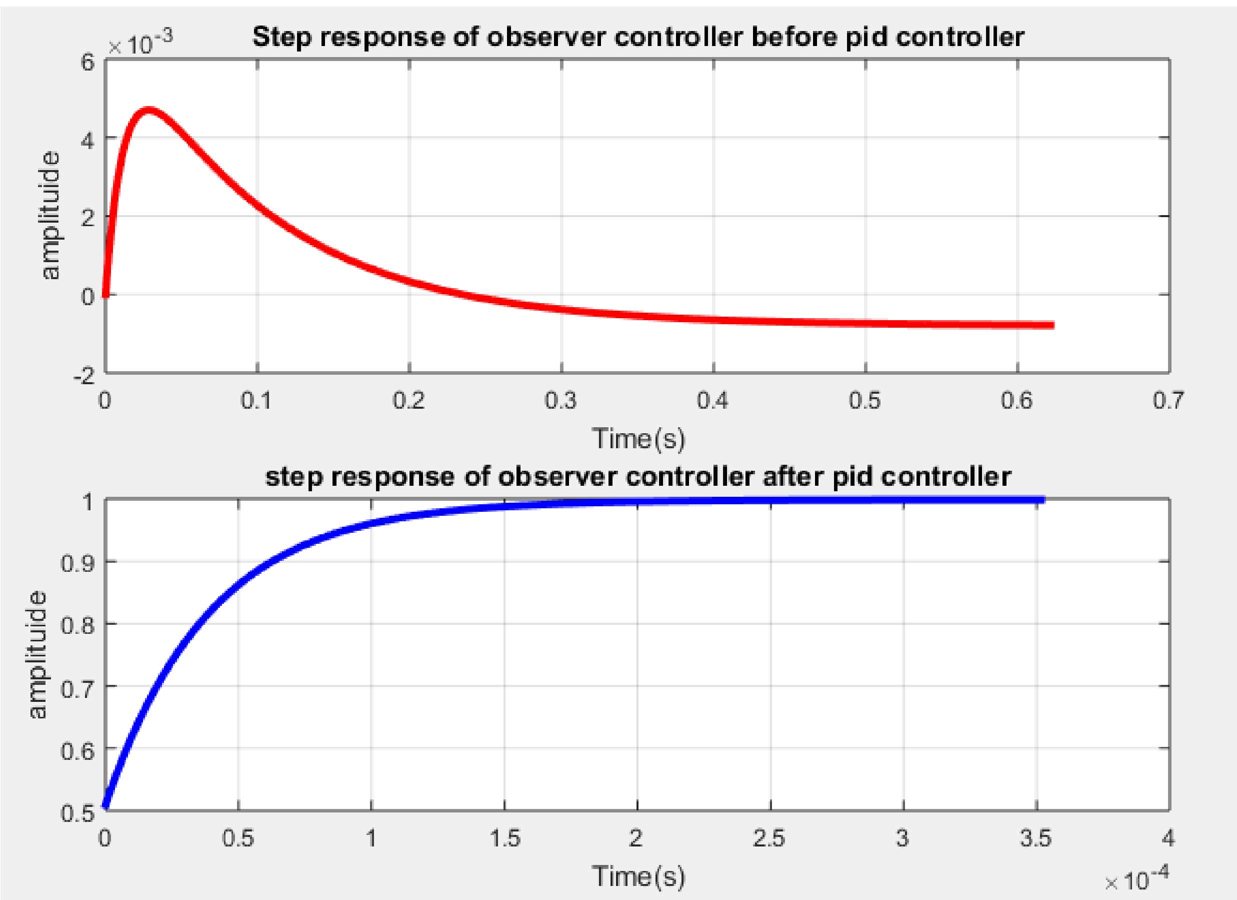
Code for steady state error using step response:

```
clc;
clear;
close all;

% declaring matrices
A = [1 2; 3 4];
B = [0.4; 0.3];
C = [1 0.4];
D = [0];

% =======================PID Controller====================
% Design observer
L = [-62.0588; 392.6471];
% make a system from num and denum of the observer system
[num,dum] = ss2tf(A-L*C, B, C, 0);
G = tf(num, dum)

% First make negative feedback sytem
Feedback_system = feedback(G,1);

% these are pid coefficients
% changing these values will change the step response
kp = 99999;
ki = 1;
kd = 2;
pid_system = pid(kp, ki, kd);

%pid_system = pidtune(G,'pid');
% Here G is in series with PID in the forward path
controlled_feedback_system = feedback(G*pid_system,1);

% ================ step response of observer state feedback controller
===============
% This is the step rsponse without PID controller
figure
subplot(2,1,1)
[y,t] = step(G);
plot(t,y,'r','Linewidth',3)
title('Step response of observer controller before pid controller')
ylabel('amplituide')
```

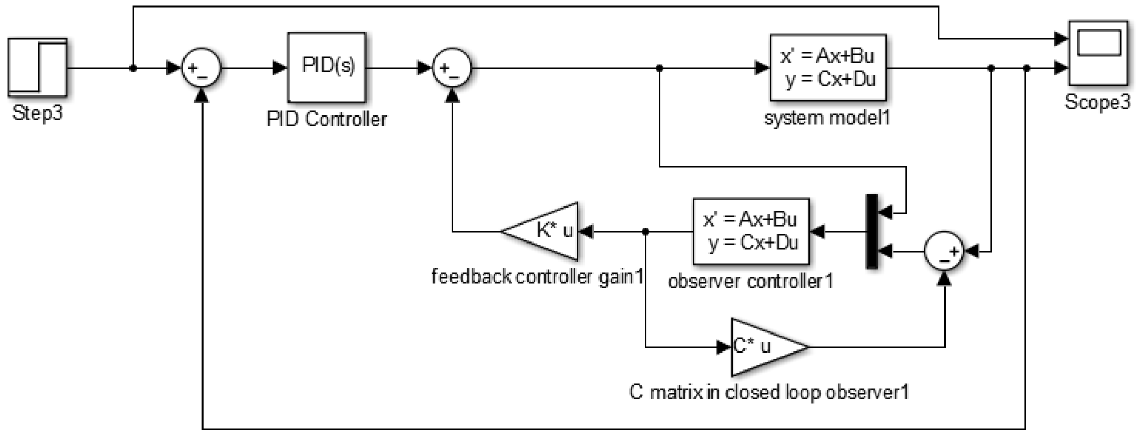
```
xlabel('Time(s)')
grid on
% This is the step rsponse with PID controller
subplot(2,1,2)
[x,t1] = step(controlled_feedback_system);
plot(t1,x,'b','Linewidth',3)
grid on
title ('step response of observer controller after pid controller')
ylabel('amplituide')
xlabel('Time(s)')
```

Output:

PID controller:



Simulink Design:

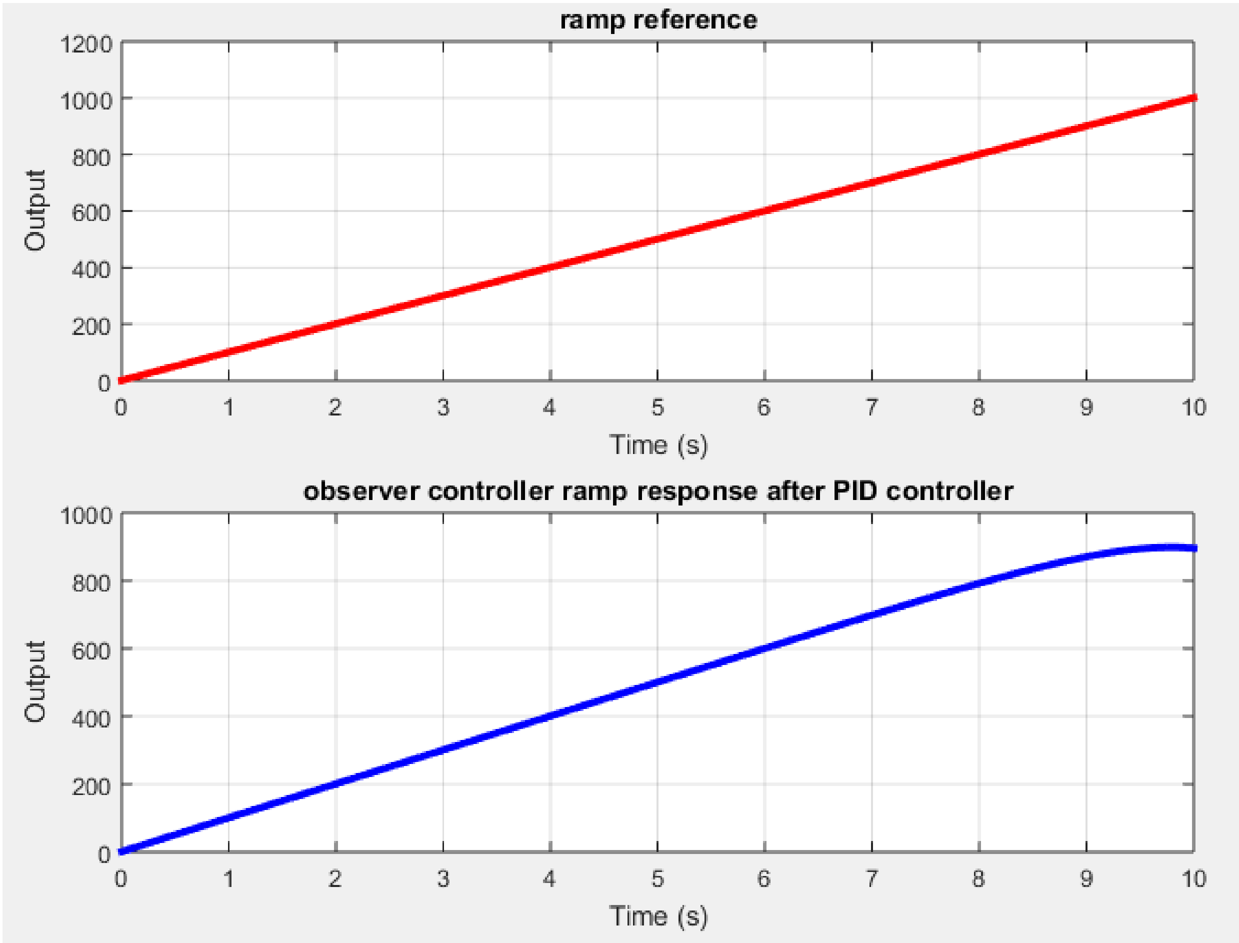


Code for steady state error using ramp signal:

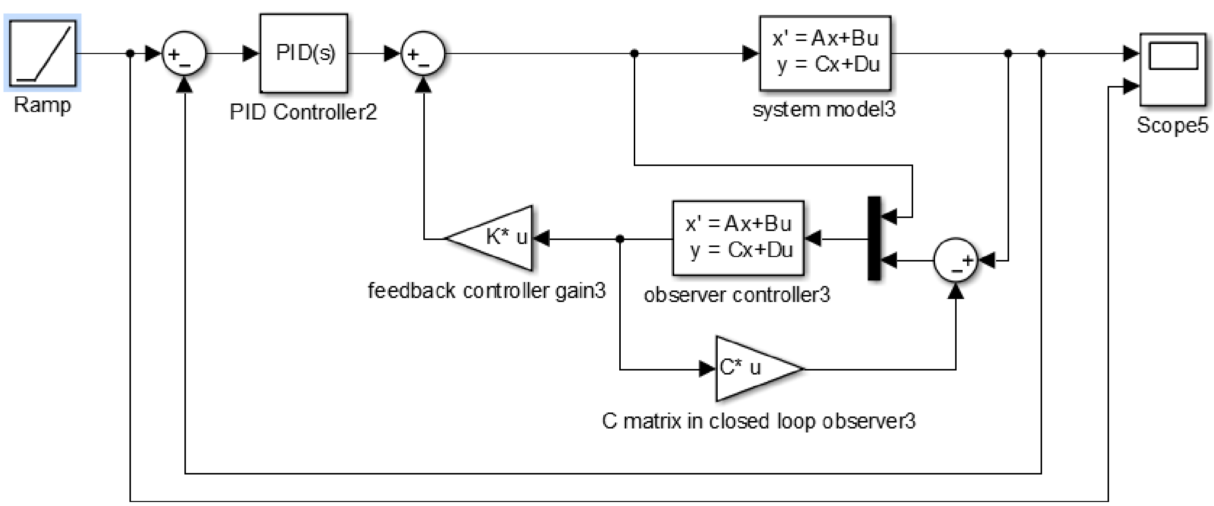
```
clc;
clear;
close all;

% declaring matrices
A = [1 2; 3 4];
B = [0.4; 0.3];
C = [1 0.4];
D = [0];

% =======================PID Controller====================
% Design observer
L = [-62.0588; 392.6471];
% make a system from num and denum of the observer system
[num,dum] = ss2tf(A-L*C, B, C, 0);
G = tf(num, dum)

% First make negative feedback sytem
Feedback_system = feedback(G,1);

% these are pid coefficients
% changing these values will change the step response
kp = 60993;
ki = -121906;
kd = 200000000;
pid_system = pid(kp, ki, kd);

%pid_system = pidtune(G,'pid');
% Here G is in series with PID in the forward path
controlled_feedback_system = feedback(G*pid_system,1);

% ================observer state feedback controller ramp
response===============
% Define the ramp signal
t = (1:1001);
ramp_signal = t;
```

```
t1 = (0:0.01:10);
% Simulate the observer state feedback controller response to the ramp
reference signal
[ramp_y, t1] = lsim(controlled_feedback_system, ramp_signal, t1);

% Plot the ramp reference signal
figure
subplot(2,1,1)
plot(t1, ramp_signal, 'r','Linewidth',3);
title('ramp reference')
xlabel('Time (s)');
ylabel('Output');
grid on

% Plot the observer controller system response
[ramp_y, t1] = lsim(controlled_feedback_system, ramp_signal, t1);
subplot(2,1,2)
plot(t1, ramp_y, 'b','Linewidth',3);
title('observer controller ramp response after PID controller')
xlabel('Time (s)');
ylabel('Output');
grid on
```

Output:



Simulink Design:

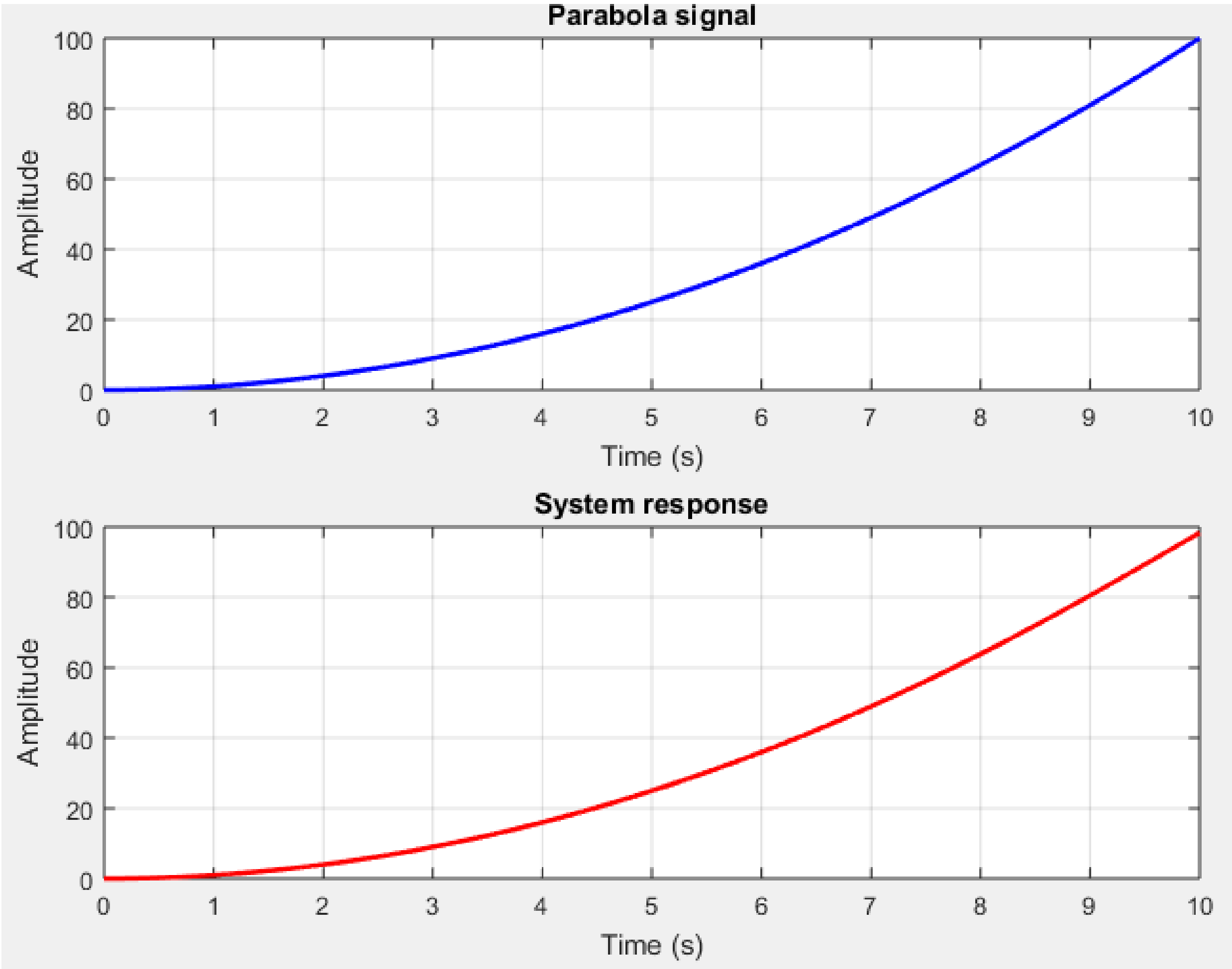


Code for steady state error using parabola response:

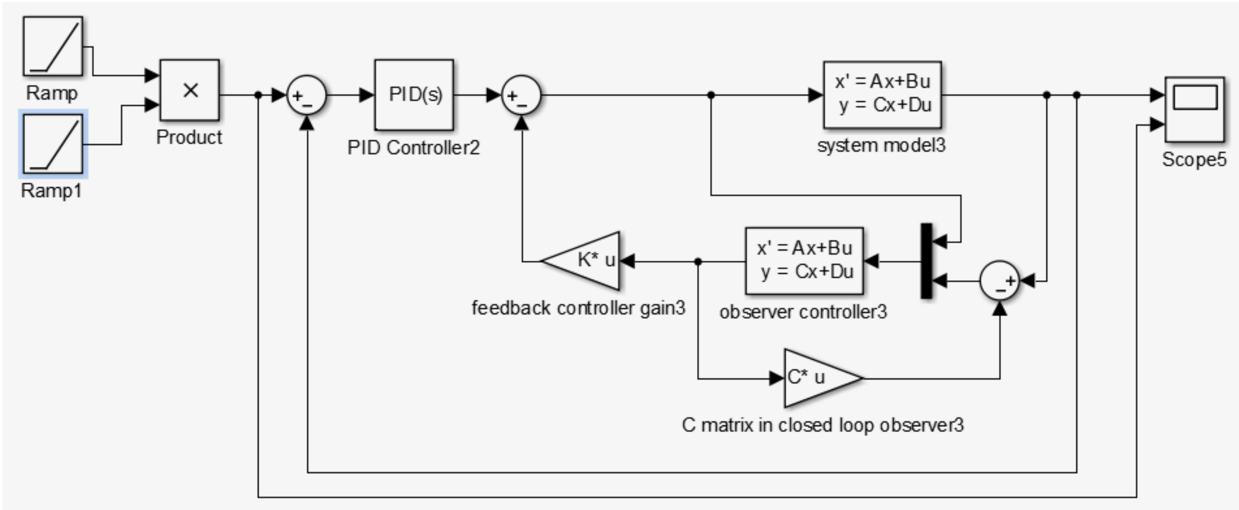
```
clc;
clear;
close all;

% declaring matrices
A = [1 2; 3 4];
B = [0.4; 0.3];
C = [1 0.4];
D = [0];



% =======================PID Controller====================
% Design observer
L = [-62.0588; 392.6471];
% make a system from num and denum of the observer system
[num,dum] = ss2tf(A-L*C, B, C, 0);
G = tf(num, dum)

% First make negative feedback sytem
Feedback_system = feedback(G,1);

% these are pid coefficients
% changing these values will change the step response
kp = 60993;
ki = -121906;
kd = 200000000;
pid_system = pid(kp, ki, kd);

%pid_system = pidtune(G,'pid');
% Here G is in series with PID in the forward path
controlled_feedback_system = feedback(G*pid_system,1);

% ================ Parabolic response of observer state feedback controller
===============
t = 0:0.01:10;
parabola_signal = t.^2;
[Y_parabola,T_parabola] =
lsim(controlled_feedback_system,parabola_signal,t);

% Plot the input signal and the system's response
figure
subplot(2,1,1);
plot(t,parabola_signal,'b','LineWidth',2);
xlabel('Time (s)');
ylabel('Amplitude');
title('Parabola signal');
grid on

subplot(2,1,2);
plot(T_parabola,Y_parabola,'r','LineWidth',2);
xlabel('Time (s)');
ylabel('Amplitude');
```

```
title('System response');
grid on
```

Output:



Simulink Design:



Ramp
Ramp1
×
Product
PID(s)
PID Controller2
x' = Ax+Bu
y = Cx+Du
system model3
Scope5
K* u
feedback controller gain3
x' = Ax+Bu
y = Cx+Du
observer controller3
C* u
C matrix in closed loop observer3