



INDIAN INSTITUTE OF INFORMATION TECHNOLOGY
NAGPUR

Control Systems

===== Project Report =====

**“Implementation of Control System
Experiments in MATLAB”**

=====

**-: Implemented By :-
Ambarish Prashant Chandurkar
(BT17ECE021)**

**-: Under the Guidance of :-
Dr. Chandrashekhar Sakode**

Table of Contents

Sr No.	Experiment Name	Date of Performance
1	Implementation of Time Response of a System in MATLAB	5 Apr 2019
2	Study of Root Locus Plot using MATLAB	5 Apr 2019
3	Plotting Bode Plots through MATLAB (Q 7, 8 of Assignment 2)	18 Apr 2019
4	Effect of variation of K_p , K_d and K_i of PID controller on system Parameters (Q nos. 1,2,3,4 of Assignment 2)	18 Apr 2019

Date : 5 Apr 2019

Lab 1 : Implementation of Time Response in MATLAB

Aim : To implement the time response of the given transfer function of the system in MATLAB

Apparatus : MATLAB Software.

MATLAB Code :

```
1 %% Experiment : Plot the time response of a system
2
3 clc;clearvars;close all;
4
5 %      2s + 1
6 % ----- => This is the first System
7 % s^2 + 2s + 5
8
9 num1 = [2 1]; %The numerator
10 den1 = [1 2 5]; %The denominator
11 sys1 = tf(num1,den1); %Create the system
12 figure(1);
13 step(num1,den1) %Plot the Step response.Save it,Print it and affix in the
14                %journal along with code
15
16
17 %      s
18 % ----- => This is the second System
19 % s + 3
20
21 num2 = [1 0]; %The numerator
22 den2 = [1 3]; %The denominator
23 sys2 = tf(num2,den2); %Create the system
24 figure(2);
25 step(num2,den2) %Plot the Step response.Save it,Print it and affix in the
26                %journal along with code
27
28 %      s + 1
29 % ----- => This is the third System, in pole zero form
30 % (s + 3)(s + 2)
31
32 G = zpk([-1],[-2 -3],1); %z = zeros, p = poles, k = gain
33 figure(3);
34 step(G) %Plot the Step response.Save it,Print it and affix in the
35         %journal along with code
36
```

```

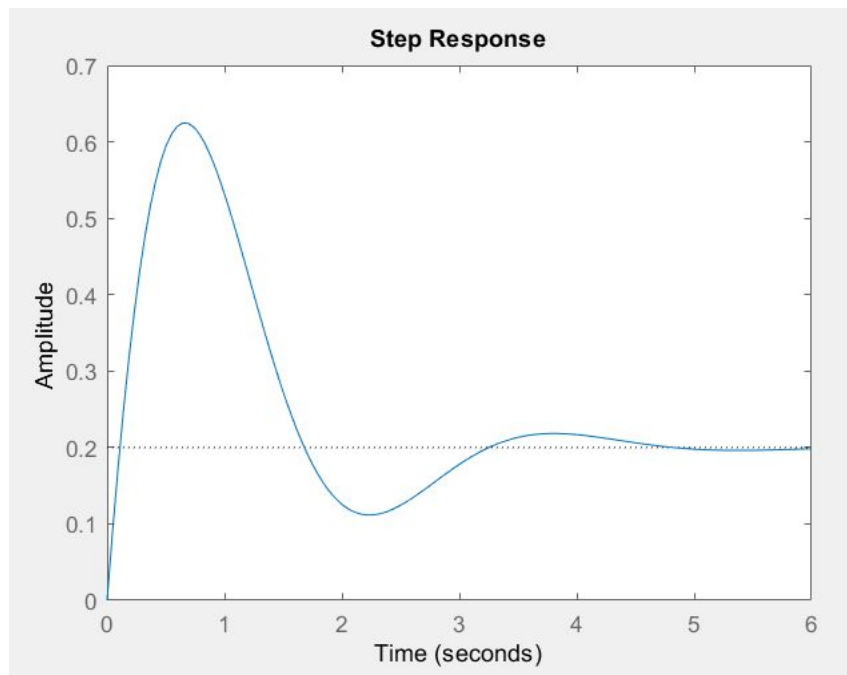
37 %      s + 1
38 % ----- => This is the third System, in coefficient form
39 % (s + 3)(s + 2)
40
41 num3 = [1 1]; %The numerator
42 den3 = [1 5 6]; %The denominator
43 sys3 = tf(num3,den3); %Create the system
44 figure(4);
45 step(num3,den3) %Plot the Step response.Save it,Print it and affix in the
46 %journal along with code
47
48
49 %      s + 1

50 % ----- => This is the fourth System, as a closed loop
51 % s^2 + 11 s + 13
52
53 %      |      s + 1      |
54 %R(s) ----->|-----> C(s)
55 %      |      | s^2 + 11 s + 13|      |
56 %      |      |                     |
57 %      |<-----|8|<-----|
58
59
60 num4 = [1 1]; %The numerator
61 den4 = [1 3 5]; %The denominator
62 Gofs = tf(num4,den4); %Create the Open loop Transfer func G(s)
63 Hofs = 8; % Create the Feedback Function H(s)
64 CLF1 = feedback(Gofs,Hofs); %This is the closed loop System
65 figure(5);
66 step(CLF1) %Plot the Step response.Save it,Print it and affix in the
67 %journal along with code
68
69
70 %      (s+1) (s+2) (s+3)
71 %----- => This is the sixth system
72 % (s^2 + 4.407s + 5.171) (s^2 + 3.593s + 5.995)
73
74 num5 = [1 1]; %The numerator
75 den5 = [1 3 5]; %The denominator
76 Gofs = tf(num5,den5); %Create the Open loop Transfer func G(s)
77 Hofs = zpk([-1],[-2 -3],1); %Create the feedback function
78 CLF2 = feedback(Gofs,Hofs); %This is the closed loop System
79 figure(6);
80 step(CLF2) %Plot the Step response.Save it,Print it and affix in the
81 %journal along with code
82

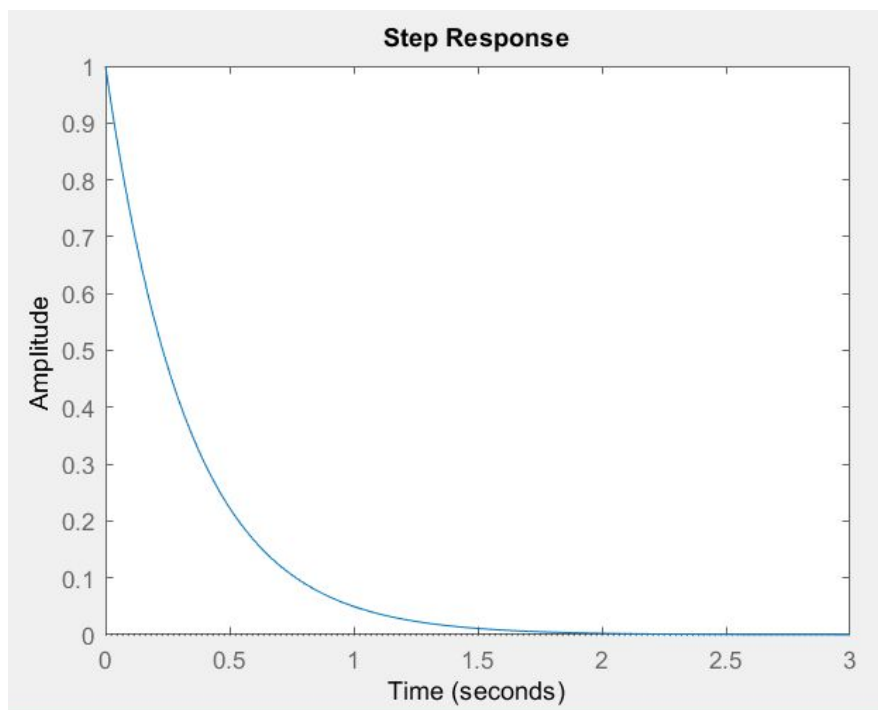
```

Step Response Outputs :

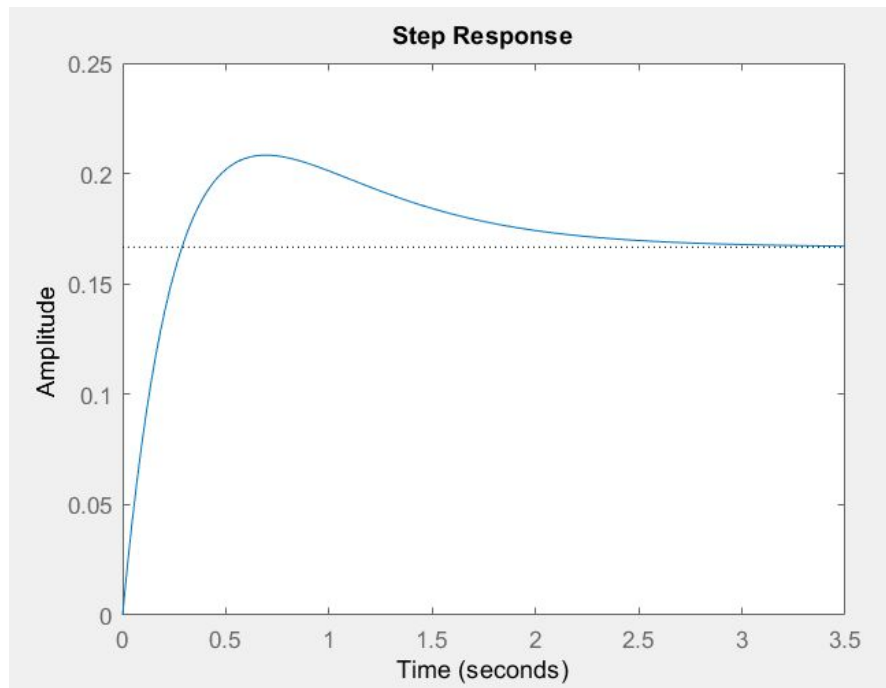
System 1



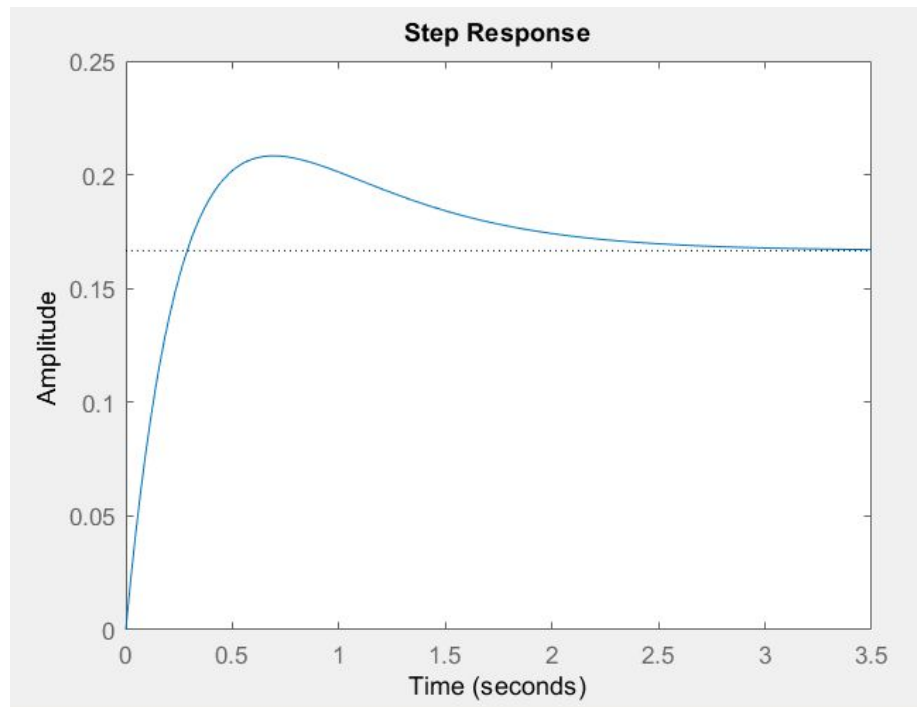
System 2



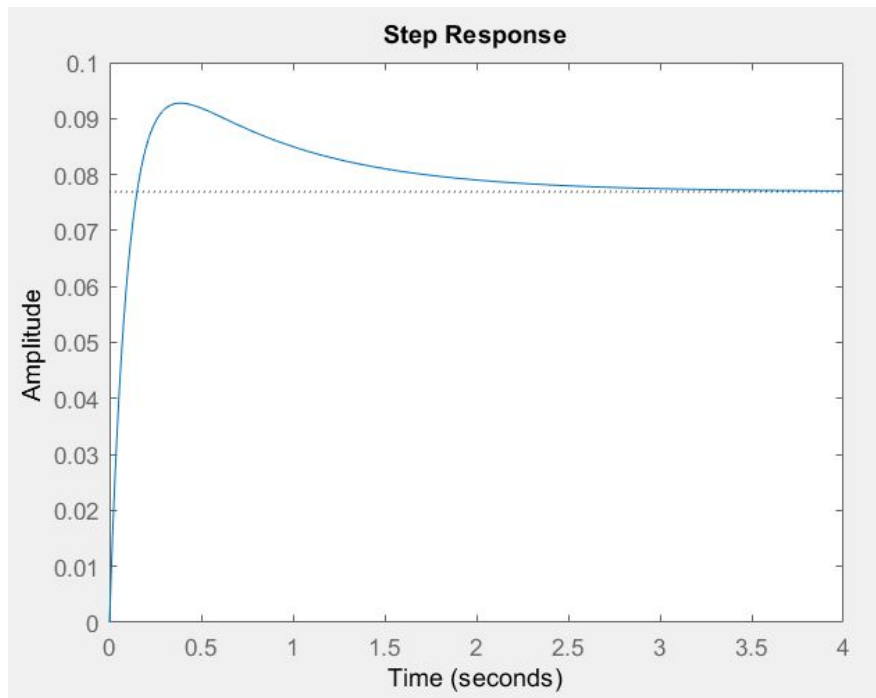
System 3



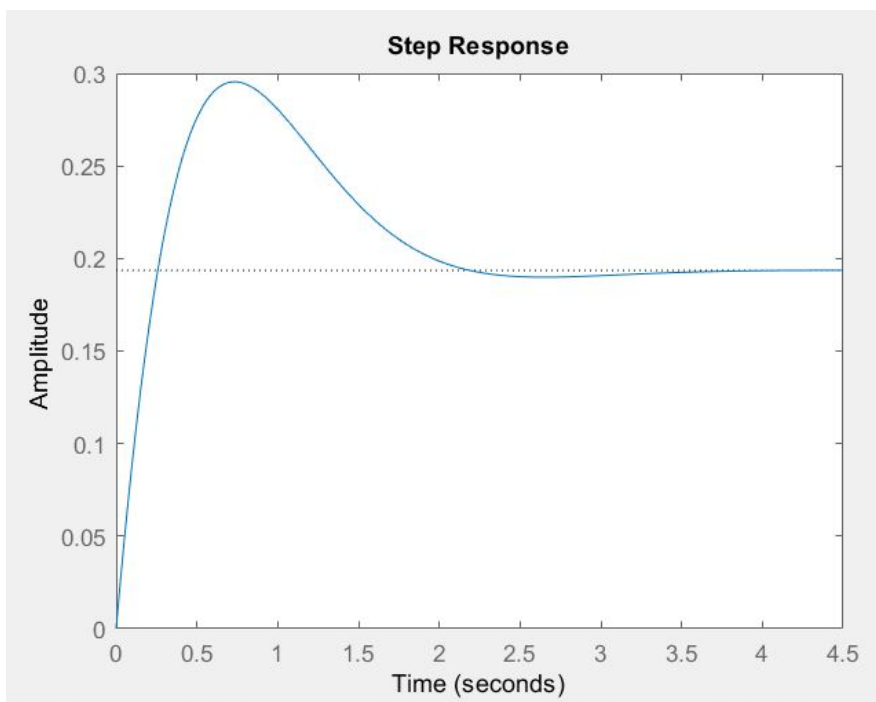
System 4



System 5



System 6



Conclusion :

In this experiment, Control System Design and Simulation was thoroughly understood , through the aid of MATLAB.

(GO TO NEXT PAGE)

Date : 5 Apr 2019

Lab 2 : Study of Root Locus Plot using MATLAB

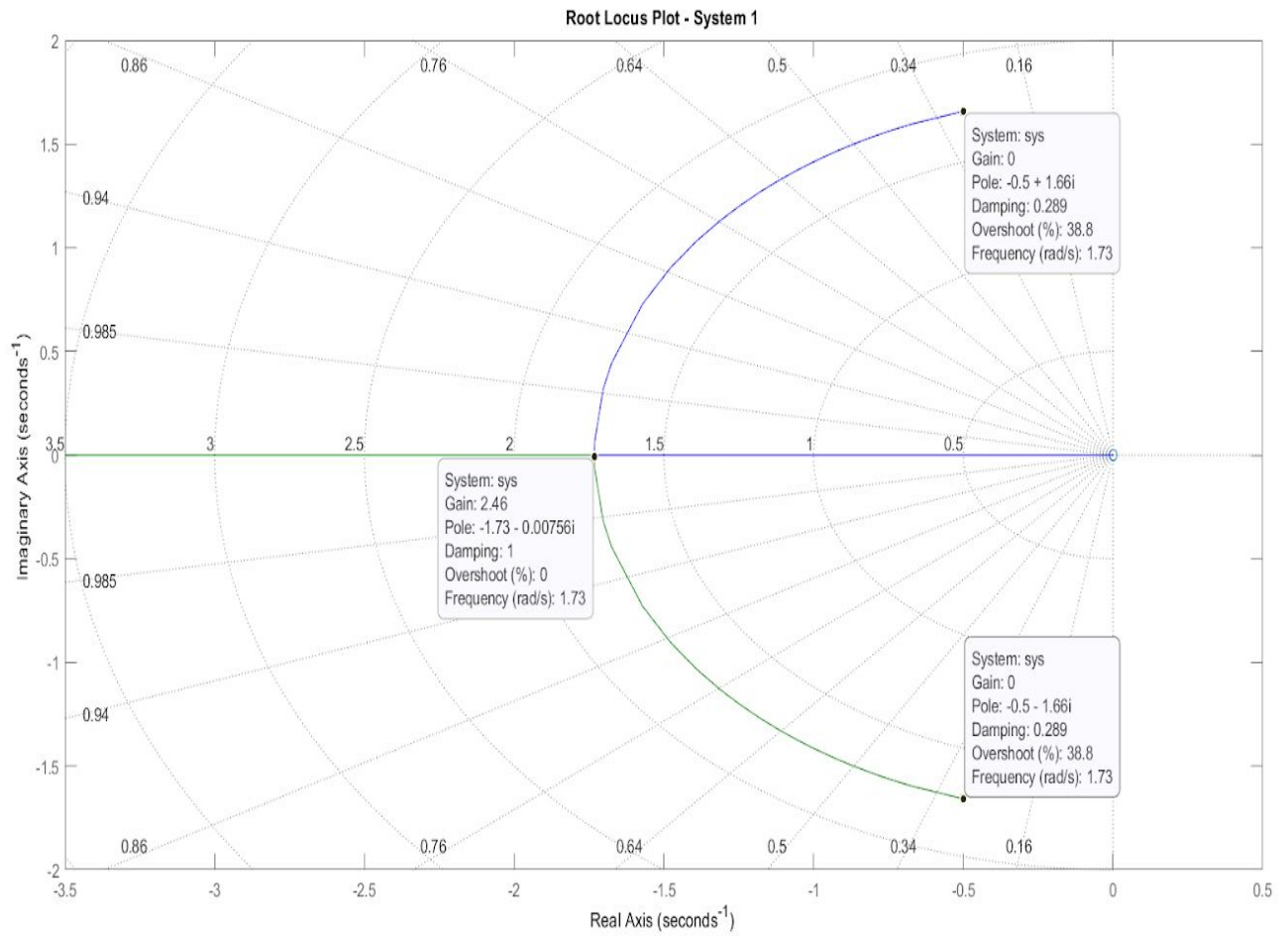
Aim : To plot the Root Locus of the given system function; using MATLAB

Apparatus : MATLAB Software.

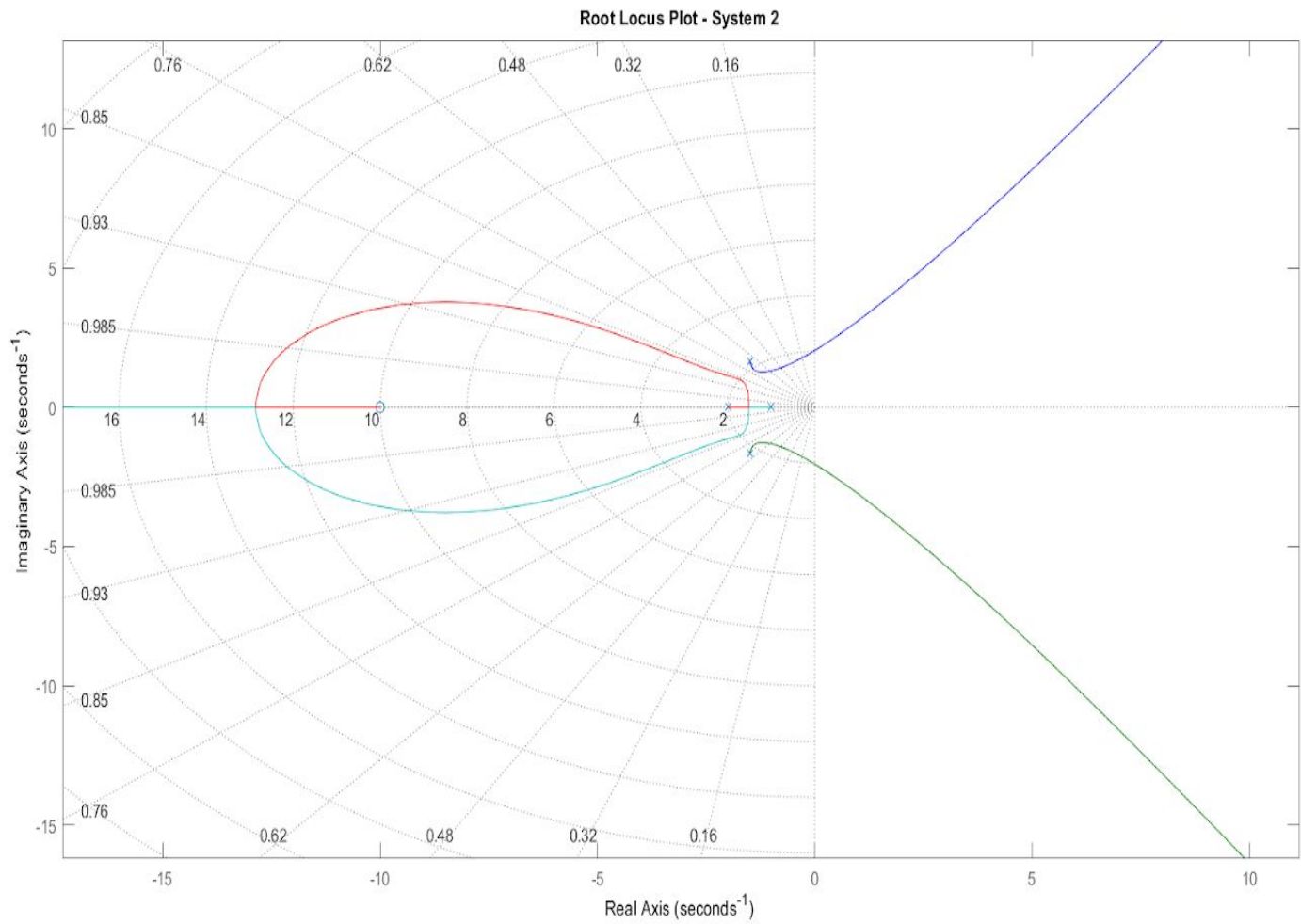
MATLAB Code :

```
1 %% Experiment : Plot the root locus
2 %Solve theoretically and then cross check with plot
3 clc;close all;clearvars;
4 %% Example 1:
5 %      s
6 %  -----  <== System 1
7 %  s^2 + s + 3
8
9 num1 = [0 1 0]; %Numerator Coefficients
10 den1 = [1 1 3]; %Denominator Coefficients
11 sys1 = tf(num1,den1) %Creation of System
12 figure(1);
13 rlocus(num1,den1); %Plotting of Root Locus Plot
14 title('Root Locus Plot - System 1');
15 %% Example 2:
16 %      s + 10
17 %  -----  <== System 2
18 %  s^4 + 6 s^3 + 16 s^2 + 21 s + 10
19
20 num2 = [0 1 10]; %Numerator Coefficients
21 p1 = [0 1 1]; %Denominator Factors p1 p2 p3
22 p2 = [0 1 2];
23 p3 = [1 3 5];
24 p4 = conv(p1,p2);
25 p5 = conv(p4,p3); %Denominator Coefficients
26 sys2 = tf(num2,p5) %Creation of System
27 figure(2)
28 rlocus(num2,p5); %Plotting of Root Locus Plot
29 title('Root Locus Plot - System 2');
```

Root Locus Output Plots :



(GO TO NEXT PAGE)



Conclusion :

The root locus plot of the given systems have been plotted theoretically and verified against the MATLAB Outputs

Date : 18 Apr 2019

Lab 3 : Plotting Bode Plots through MATLAB

Aim : To plot the Bode Plots of the given System transfer functions using MATLAB

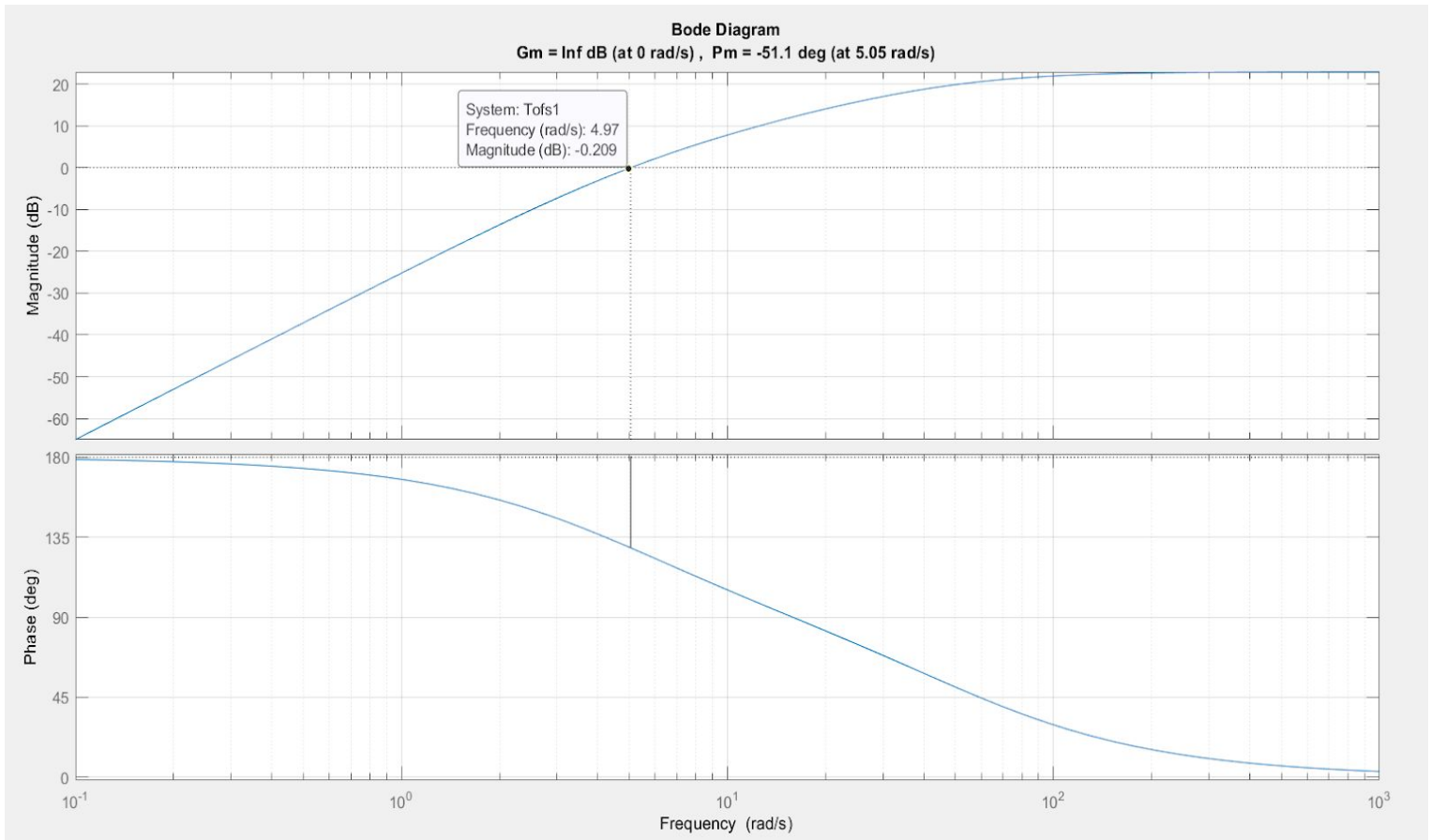
Apparatus : MATLAB Software.

MATLAB Code :

```
1 %% Experiment : To plot the Bode plots of given Systems
2 clc;close all;clearvars;
3 %% Q7 : Find the Gain 'K' for which Gain Cross over Frequency is '5 rad/sec'
4 % Ans : Gain K = 0.056, as seen from Bode plot (Solution in Assignment)
5 %      0.056 s^2
6 % ----- <== System 1
7 % 0.004 s^2 + 0.22 s + 1
8
9 Num1 = [0.056 0 0]; %Numerator Coefficients
10 Den1 = [0.2 1]; %Denominator factor 1 Coefficients
11 Den2 = [0.02 1]; %Denominator factor 2 Coefficients
12 Tofs1 = tf(Num1,conv(Den1,Den2)); %Creation of System
13 figure(1)
14 margin(Tofs1) %Plots the Bode Plot and also gives the "Gain Crossover Frequency"
15
16 %% Q8 : Find the Phase Crossover Frequency
17 % Ans = 9.976 (Theoretically), Verified from bode plot
18 %      1000
19 % ----- <== System 2
20 % s^3 + 24.95 s^2 + 99.53 s + 999
21
22 Num2 = 1000; %Numerator Coefficients
23 Den3 = [1 22.5]; %Denominator factor 1 Coefficients
24 Den4 = [1 2.45 44.4]; %Denominator factor 2 Coefficients
25 Tofs2 = tf(Num2,conv(Den3,Den4)); %Creation of System
26 figure(2)
27 margin(Tofs2) %Plots the Bode Plot and also gives the "Phase Crossover Frequency"
```

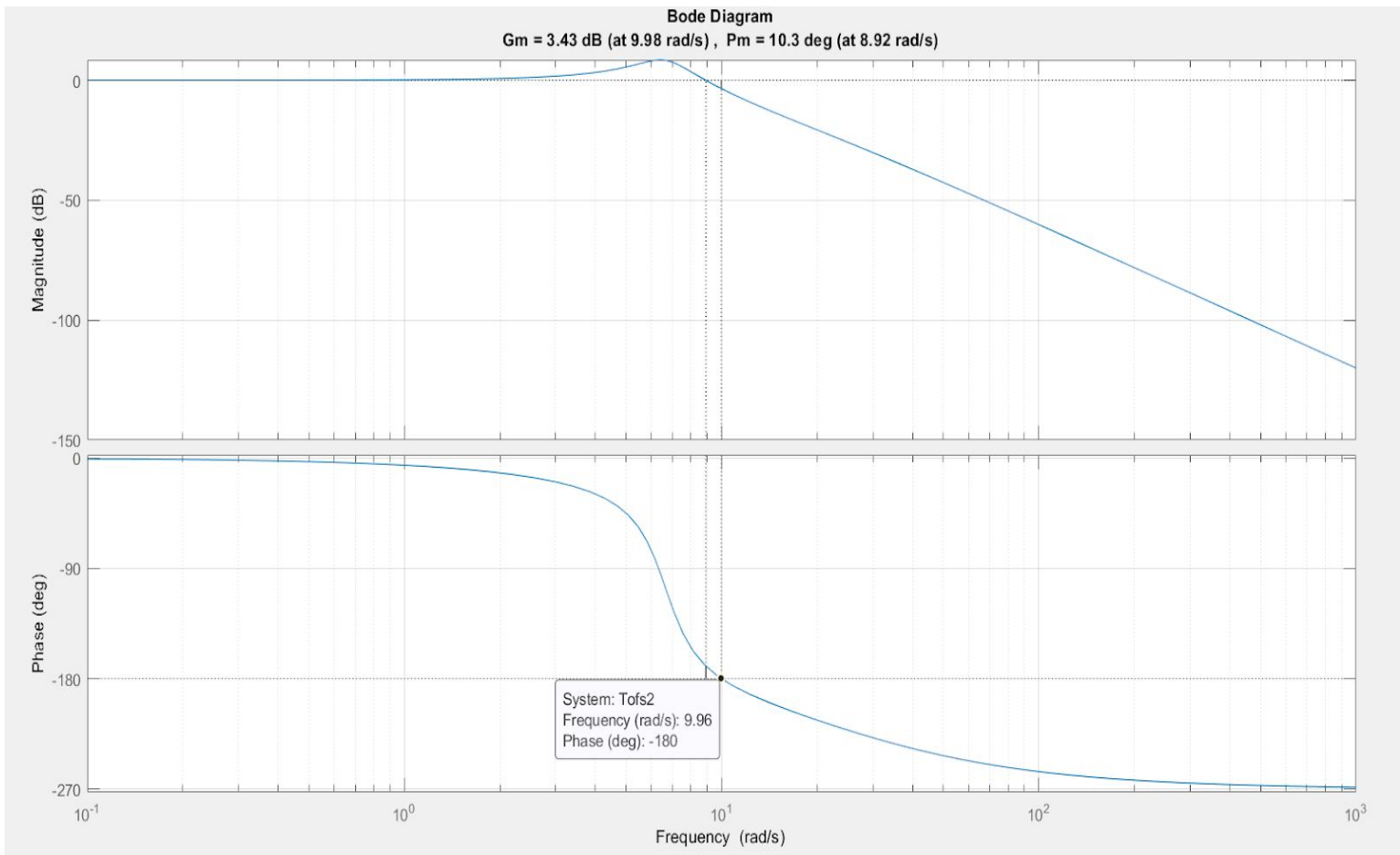
Bode Plots :

System Q.7



From the above Bode plot, we observe that for $K = 0.056$, Gain Crossover Frequency is $\omega_p = 5$ rad/sec, as required.

System Q.8



From the above Bode Plot , it is clear that the Phase Crossover Frequency is 9.96 rad/sec.

Conclusion :

The Q7 and Q8 of Assignment 2 have been solved theoretically, and verified against the output of MATLAB

Date : 18 Apr 2019

Lab 4 : Effect of variation of Kp , Kd and Ki of PID controller on system Parameters

Aim : To obtain ζ , ω_n , t_{ss} and M_p of the system when the parameters Kp , Kd, Ki are varied for a PID Controller.

Apparatus : MATLAB Software.

MATLAB Code :

```
1 %% Q1 : Given parameters are Kp = Kd = Ki = 0
2 % T(s) = 0 ; System Function
3
4 clc;close all;clearvars;
5 Kp = 0; %Given parameters
6 Kd = 0;
7 Ki = 0;
8 PIDController = pid(Kp,Kd,Ki);
9 PIDController.u = 'E'; %Input of PID Block
10 PIDController.y = 'PIDOut'; %Output of PID Block
11 GofS = zpk([], [0,-10],8); %The Open loop Transfer function
12 GofS.u = 'PIDOut'; %Input of Gofs
13 GofS.y = 'Cofs'; %Output of Gofs
14 Sum = sumblk('E = Rofs - Cofs'); %Calculation of Error
15 Tofs = connect(GofS,PIDController,Sum,'Rofs','Cofs'); %Creation of Closed Loop
system
16
17 step(Tofs);
18 SysCharac = stepinfo(Tofs);

20 %% Q2 Case I: Given parameters are Kp = 5 ; Kd = Ki = 0
21 % 40
22 % ----- = T(s) ; System Function
23 % (s^2 + 10s + 40)
24
25 clc;close all;clearvars;
26 Kp = 5; %Given parameters
27 Kd = 0;
28 Ki = 0;
29 PIDController = pid(Kp,Kd,Ki);
30 PIDController.u = 'E'; %Input of PID Block
31 PIDController.y = 'PIDOut'; %Output of PID Block
32 GofS = zpk([], [0,-10],8); %The Open loop Transfer function
33 GofS.u = 'PIDOut'; %Input of Gofs
34 GofS.y = 'Cofs'; %Output of Gofs
35 Sum = sumblk('E = Rofs - Cofs'); %Calculation of Error
36 Tofs = connect(GofS,PIDController,Sum,'Rofs','Cofs'); %Creation of Closed Loop
system
37
38 step(Tofs);
39 SysCharac = stepinfo(Tofs);
```

```

41 %% Q2 Case II : Given parameters are Kp = 10 ; Kd = Ki = 0
42 %           80
43 % ----- = T(s) ; System Function
44 % (s^2 + 10s + 80)
45
46 clc;close all;clearvars;
47 Kp = 10; %Given parameters

48 Kd = 0;
49 Ki = 0;
50 PIDController = pid(Kp,Kd,Ki);
51 PIDController.u = 'E'; %Input of PID Block
52 PIDController.y = 'PIDOut'; %Output of PID Block
53 GofS = zpk([], [0,-10],8); %The Open loop Transfer function
54 GofS.u = 'PIDOut'; %Input of Gofs
55 GofS.y = 'Cofs'; %Output of Gofs
56 Sum = sumblk('E = Rofs - Cofs'); %Calculation of Error
57 Tofs = connect(GofS,PIDController,Sum, 'Rofs', 'Cofs'); %Creation of Closed Loop
system
58
59 step(Tofs);
60 SysCharac = stepinfo(Tofs);

62 %% Q3 Case I : Given parameters are Kd = 5; Kp = Ki = 0
63 %           40
64 % ----- = T(s) ; System Function
65 % (s+10.37) (s^2 - 0.3718s + 3.857)
66
67 clc;close all;clearvars;
68 Kp = 0; %Given parameters
69 Kd = 5;
70 Ki = 0;
71 PIDController = pid(Kp,Kd,Ki);
72 PIDController.u = 'E'; %Input of PID Block
73 PIDController.y = 'PIDOut'; %Output of PID Block
74 GofS = zpk([], [0,-10],8); %The Open loop Transfer function
75 GofS.u = 'PIDOut'; %Input of Gofs
76 GofS.y = 'Cofs'; %Output of Gofs
77 Sum = sumblk('E = Rofs - Cofs'); %Calculation of Error
78 Tofs = connect(GofS,PIDController,Sum, 'Rofs', 'Cofs'); %Creation of Closed Loop
system
79
80 step(Tofs);
81 SysCharac = stepinfo(Tofs);

```



```

83 %% Q3 Case II : Given parameters are Kd = 10; Kp = Ki = 0
84 %           80
85 % ----- = T(s) ; System Function
86 % (s+10.7) (s^2 - 0.6989s + 7.477)
87 clc;close all;clearvars;
88 Kp = 0; %Given parameters
89 Kd = 10;
90 Ki = 0;
91 PIDController = pid(Kp,Kd,Ki);
92 PIDController.u = 'E'; %Input of PID Block
93 PIDController.y = 'PIDOut'; %Output of PID Block
94 GofS = zpk([], [0,-10],8); %The Open loop Transfer function

95 GofS.u = 'PIDOut'; %Input of Gofs
96 GofS.y = 'Cofs'; %Output of Gofs
97 Sum = sumblk('E = Rofs - Cofs'); %Calculation of Error
98 Tofs = connect(GofS,PIDController,Sum,'Rofs','Cofs'); %Creation of Closed Loop
system
99
100 step(Tofs);
101 SysCharac = stepinfo(Tofs);

103 %% Q4 Case I : Given parameters are Ki = 5; Kp = Kd = 0
104 %           40 s
105 % ----- = T(s) ; System Function
106 % s (s+50)
107
108 clc;close all;clearvars;
109 Kp = 0; %Given parameters
110 Kd = 0;
111 Ki = 5;
112 PIDController = pid(Kp,Kd,Ki);
113 PIDController.u = 'E'; %Input of PID Block
114 PIDController.y = 'PIDOut'; %Output of PID Block
115 GofS = zpk([], [0,-10],8); %The Open loop Transfer function
116 GofS.u = 'PIDOut'; %Input of Gofs
117 GofS.y = 'Cofs'; %Output of Gofs
118 Sum = sumblk('E = Rofs - Cofs'); %Calculation of Error
119 Tofs = connect(GofS,PIDController,Sum,'Rofs','Cofs'); %Creation of Closed Loop
system
120
121 step(Tofs);
122 SysCharac = stepinfo(Tofs);

```

```

124 %% Q4 Case II : Given parameters are Ki = 10; Kp = Kd = 0
125 %      80 s
126 % ----- = T(s) ; System Function
127 % s (s+90)
128
129 clc;close all;clearvars;
130 Kp = 0; %Given parameters
131 Kd = 0;
132 Ki = 10;
133 PIDController = pid(Kp,Kd,Ki);
134 PIDController.u = 'E'; %Input of PID Block
135 PIDController.y = 'PIDOut'; %Output of PID Block
136 GofS = zpk([], [0,-10], 8); %The Open loop Transfer function
137 GofS.u = 'PIDOut'; %Input of Gofs
138 GofS.y = 'Cofs'; %Output of Gofs
139 Sum = sumblk('E = Rofs - Cofs'); %Calculation of Error
140 Tofs = connect(GofS,PIDController,Sum, 'Rofs', 'Cofs'); %Creation of Closed Loop
system
142 step(Tofs);
143 SysCharac = stepinfo(Tofs);

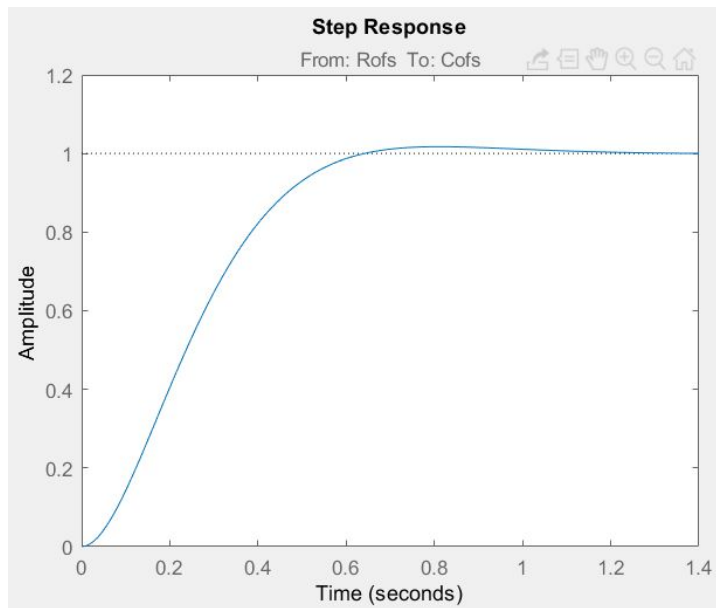
```

Step Response and System Parameters :

System Q.1



System Q.2 - Case I



Command Window

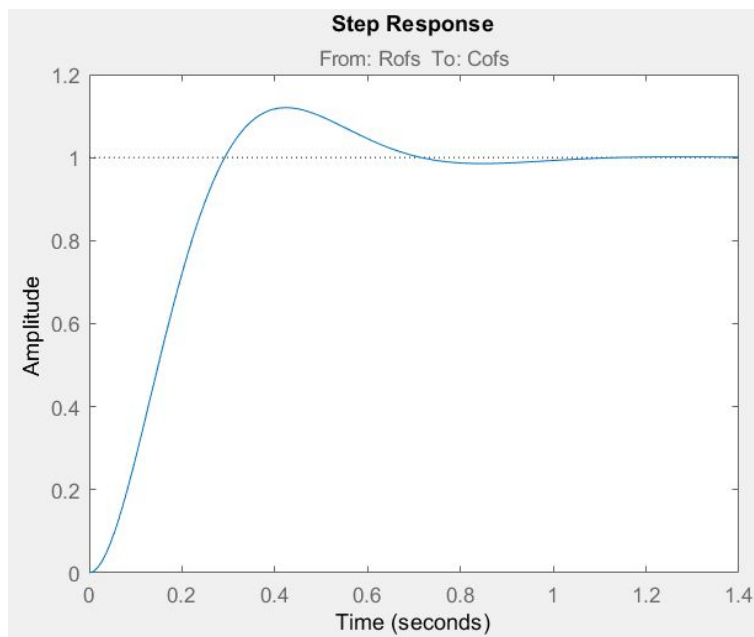
```
>> SysCharac
```

```
SysCharac =
```

```
struct with fields:
```

```
RiseTime: 0.3847  
SettlingTime: 0.5824  
SettlingMin: 0.9040  
SettlingMax: 1.0173  
Overshoot: 1.7322  
Undershoot: 0  
Peak: 1.0173  
PeakTime: 0.8105
```

System Q.2 - Case II



Command Window

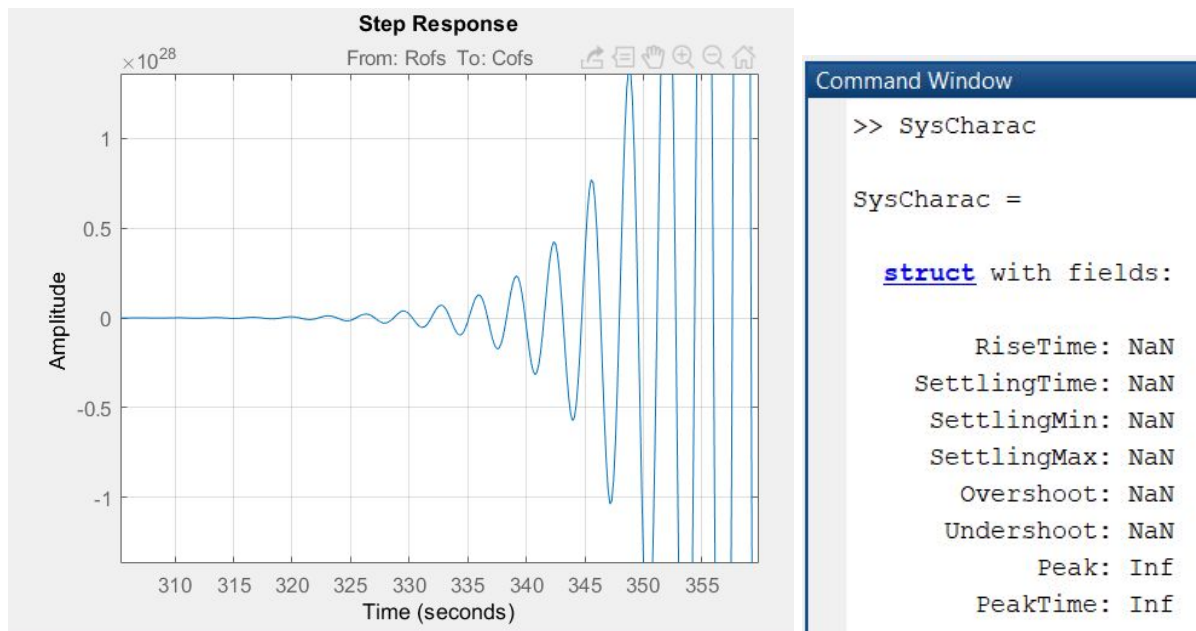
```
>> SysCharac
```

```
SysCharac =
```

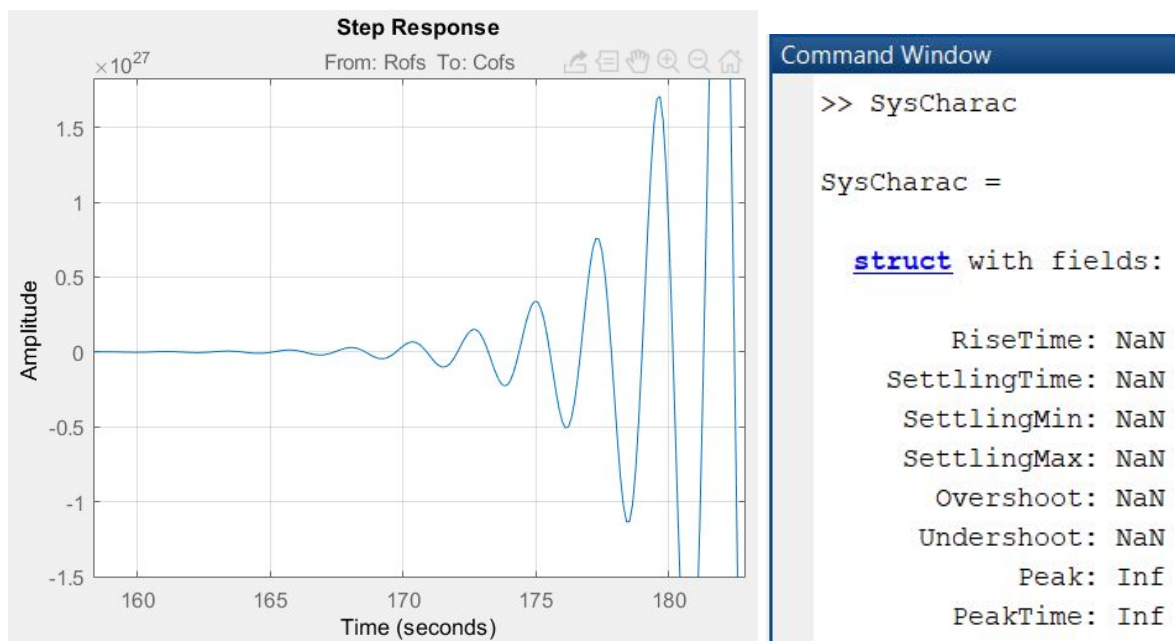
```
struct with fields:
```

```
RiseTime: 0.1968  
SettlingTime: 0.6544  
SettlingMin: 0.9174  
SettlingMax: 1.1203  
Overshoot: 12.0265  
Undershoot: 0  
Peak: 1.1203  
PeakTime: 0.4237
```

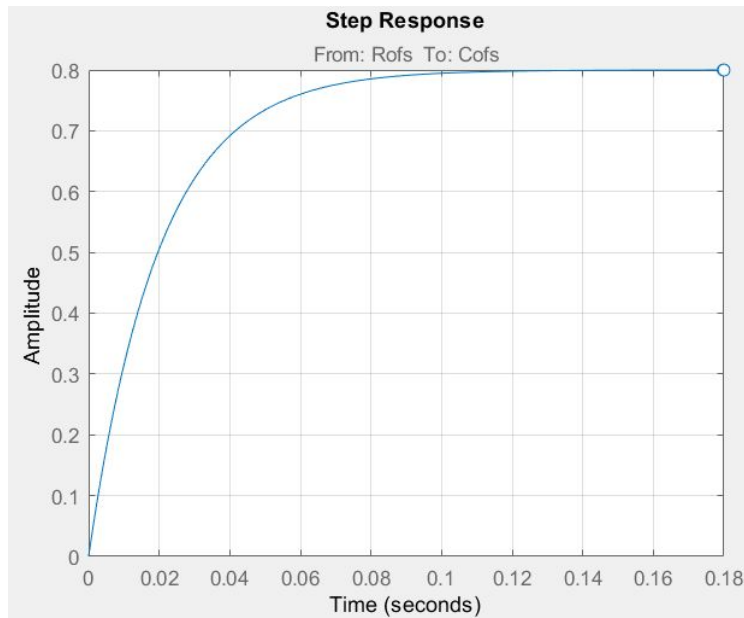
System Q.3 - Case I



System Q.3 - Case II



System Q.4 - Case I



Command Window

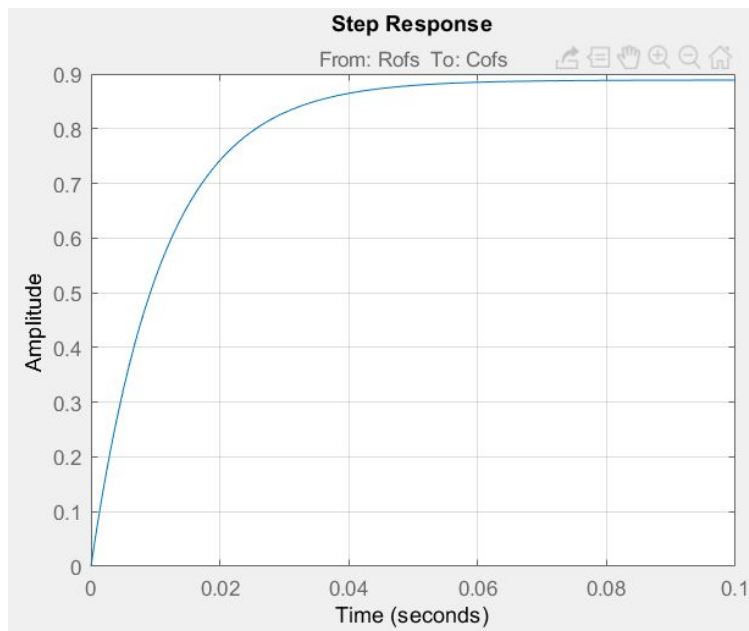
```
>> SysCharac
```

```
SysCharac =
```

```
struct with fields:
```

```
RiseTime: 0.0439  
SettlingTime: 0.0782  
SettlingMin: 0.7200  
SettlingMax: 0.8000  
Overshoot: 0  
Undershoot: 0  
Peak: 0.8000  
PeakTime: 0.6162
```

System Q.4 - Case II



Command Window

```
>> SysCharac
```

```
SysCharac =
```

```
struct with fields:
```

```
RiseTime: 0.0244  
SettlingTime: 0.0435  
SettlingMin: 0.8000  
SettlingMax: 0.8889  
Overshoot: 0  
Undershoot: 0  
Peak: 0.8889  
PeakTime: 0.3423
```

Conclusion :

From this experiment, it is now well understood the effect of variation of the parameters of the PID Controller on the transient response of the system.