Faculty of Engineering & Technology
Electrical & Computer Engineering Department

EE4302-Control Systems
Assignment

Prepared by:

Eman Asfour 1200206

Instructor: Dr. Jamal Siam

Section: 1

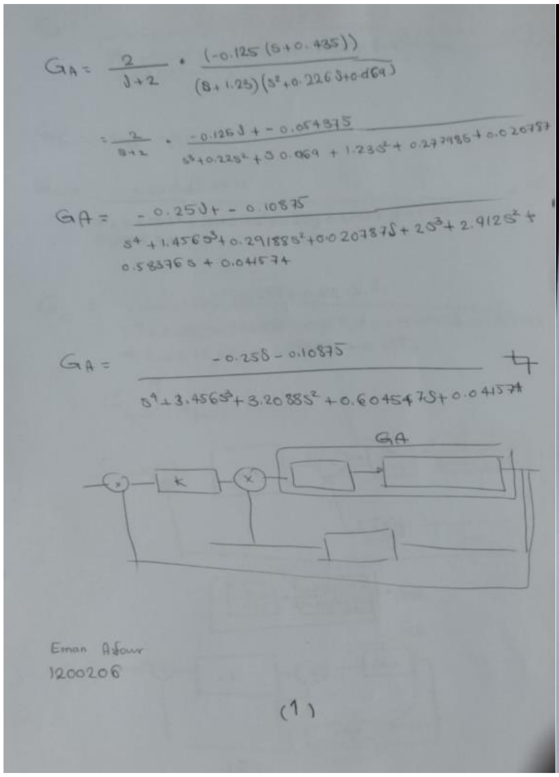
Date: 09/01/2024

Abstract

The primary objective of this assignment is to gain proficiency in utilizing MATLAB and understanding Simulink concepts, particularly focusing on system stability characterization, including stable oscillatory, stable non-oscillatory, unstable, and the limit of stability scenarios. The assignment involves learning how to determine the transfer function of a system and subsequently plot the root locus using specified parameter values. Through these tasks, students will enhance their practical knowledge of MATLAB and Simulink, gaining hands-on experience in analyzing and visualizing the stability of dynamic systems. The assignment not only emphasizes theoretical understanding but also provides a practical foundation in using MATLAB and Simulink for system analysis, offering a valuable skill set for engineering applications.

Table of Contents

1. Determine the transfer function of the system (Using Simulink and using math. Formulation)



$G_A = \frac{2}{s+2} \cdot \frac{(-0.125(s+0.435))}{(s+1.23)(s^2+0.226s+0.069)}$

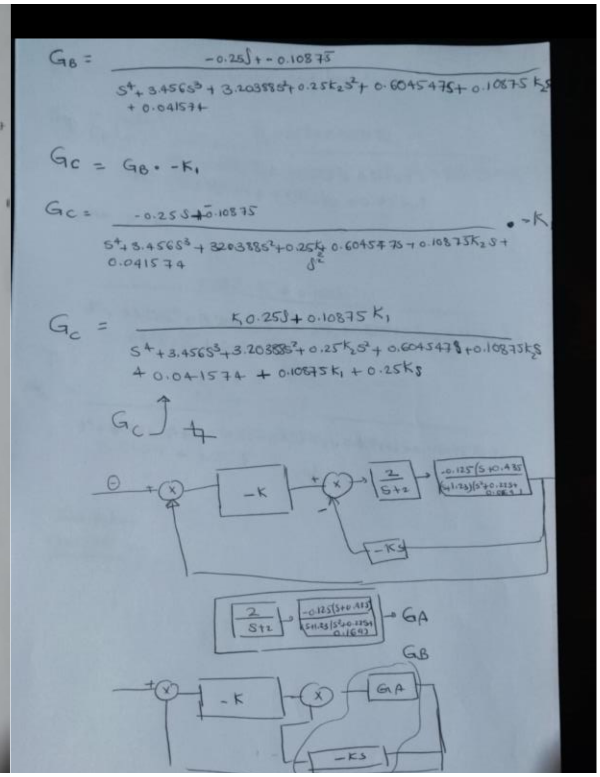
$= \frac{2}{s+2} \cdot \frac{-0.125s + -0.054375}{s^3+0.225s^2+s\,0.069+1.23s^2+0.27798s+0.02087}$

$G_A = \frac{-0.25s + -0.10875}{s^4+1.456s^3+0.29188s^2+0.020787s+2s^3+2.912s^2+0.58376s+0.041574}$

$G_A = \frac{-0.25s - 0.10875}{s^4+3.456s^3+3.208s^2+0.604575s+0.041574}$

Eman Afour
1200206

(1)

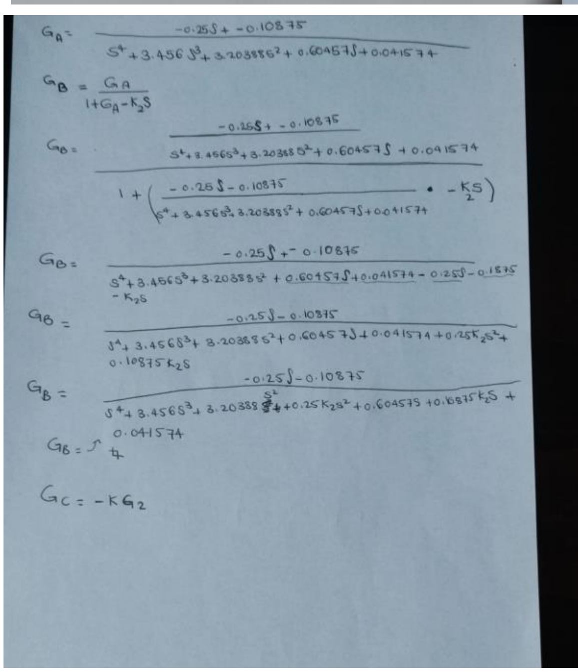


$G_B = \frac{-0.25s + -0.10875}{s^4+3.456s^3+3.20388s^2+0.25k_2s^2+0.604575s+0.10875k_2s+0.041574}$

$G_C = G_B \cdot -K_1$

$G_C = \frac{-0.25s + -0.10875}{s^4+3.456s^3+3.20388s^2+0.25k_2s^2+0.604575s+0.10875k_2s+0.041574} \cdot -K_1$

$G_c = \frac{K_1\,0.25s + 0.10875K_1}{s^4+3.456s^3+3.20388s^2+0.25k_2s^2+0.604575s+0.10875k_2s+0.041574+0.10875K_1+0.25K_1s}$



$G_A = \frac{-0.25s + -0.10875}{s^4+3.456s^3+3.20388s^2+0.60457s+0.041574}$

$G_B = \frac{G_A}{1+G_A - K_2 s}$

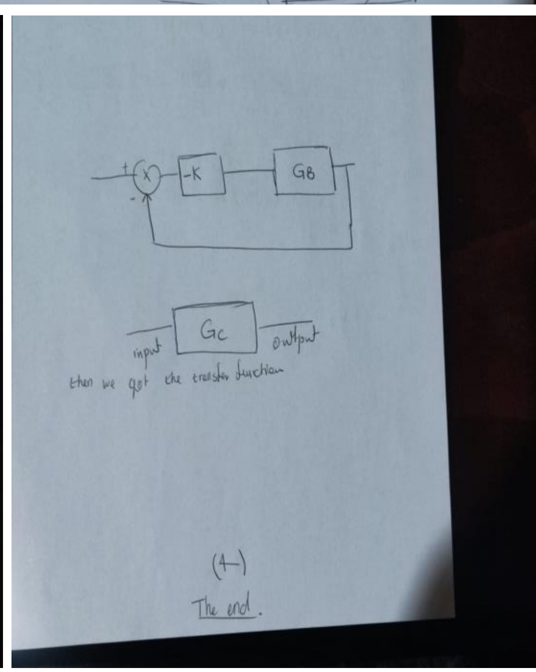
$G_B = \frac{\frac{-0.25s + -0.10875}{s^4+3.456s^3+3.20388s^2+0.60457s+0.041574}}{1+\left(\frac{-0.25s-0.10875}{s^4+3.456s^3+3.20388s^2+0.60457s+0.041574} \cdot -K_2 s\right)}$

$G_B = \frac{-0.25s + -0.10875}{s^4+3.456s^3+3.20388s^2+0.60457s+0.041574-0.25s-0.10875-K_2s}$

$G_B = \frac{-0.25s - 0.10875}{s^4+3.4568s^3+3.20388s^2+0.60457s+0.041574+0.25K_2s^2+0.10875K_2s}$

$G_B = \frac{-0.25s - 0.10875}{s^4+3.4568s^3+3.20388s^2+0.25K_2s^2+0.60457s+0.10875K_2s+0.041574}$

$G_C = -K G_2$



then we got the transfer function

(4)

The end.

2. Determine the transfer function using fixed $K_1 = 5$

$$G_c \left(\begin{array}{c}\text{the transfer}\\ \text{function}\end{array}\right) = \frac{K_1\, 0.25 s + 0.10875 K_1}{s^4 + 3.456 s^3 + 8.20388 s^2 + 0.25 K_2 s^2 + 0.60547 s + 0.10875 K_2 s + 0.041574 + 0.10875 K_1 + 0.25 K_1 s}$$

when $5 = K_1$

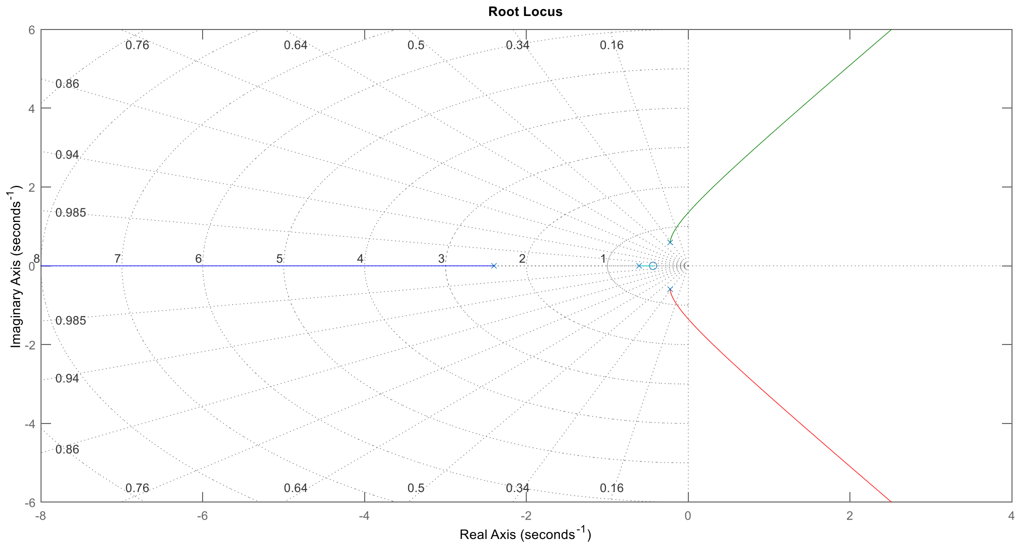
$$= \frac{0.25 s \cdot 5 + 0.10875 \cdot 5}{s^4 + 3.456 s^3 + 3.2038 s^2 + 0.25 s^2 K_2 + 0.60547 s + 0.10875 K_2 s + 0.10875 \cdot 5 + 0.25 s \cdot 5}$$

$$G_c = \frac{1.25 + 0.54375}{s^4 + 8.456 s^3 + 3.2038 s^2 + 0.25 s^2 K_2 + 0.60547 s + 0.10875 K_2 s + 0.54375 + 1.25 s}$$

Eman Asfour

1200206

3. Plot the root locus of the system using the assigned value of K1.



A =

$$\begin{matrix} -1.4560 & -0.2949 & -0.0208 & 2.0000 \\ 1.0000 & 0 & 0 & 0 \\ 0 & 1.0000 & 0 & 0 \\ 0 & -0.6250 & -0.2719 & -2.0000 \end{matrix}$$

B =

$$\begin{matrix} 0 \\ 0 \\ 0 \\ -5 \end{matrix}$$

C =

$$\begin{matrix} 0 & -0.1250 & -0.0544 & 0 \end{matrix}$$

D =

0

num =

$$\begin{matrix} 0 & 0 & 0 & 1.2500 & 0.5437 \end{matrix}$$

den =

1.0000   3.4560   3.2069   1.8605   0.5853

closed =

1.25 s + 0.5437

----

s^4 + 3.456 s^3 + 3.207 s^2 + 1.861 s + 0.5853

Continuous-time transfer function.

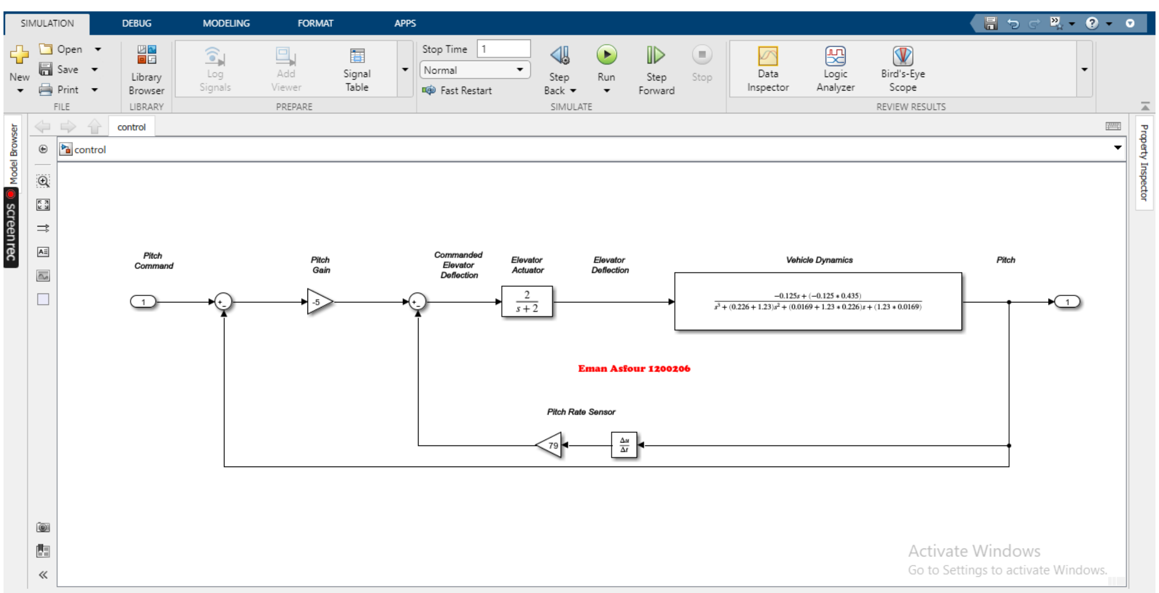
>> isstable (closed)

ans =

logical

1

```
%eman 1200206
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
closed=tf(num,den)
rlocus(closed)
```



4. Repeat the points 1 to 3 considering K2 fixed =8

1. Determine the transfer function using fixed $K_2 = 8$

$$G_c = \frac{K_1\,(0.25s + 0.10875)}{s^4 + 3.456s^3 + 3.20388s^2 + 0.25\underline{K_2}s^2 + 0.60547s + 0.10875\underline{K_2}s + 0.04157 + 0.10875K_1 + 0.25K_1s}$$

$$G_c = \frac{K_1\,(0.25s + 0.10875}{s^4 + 3.456s^3 + 3.20388s^2 + 0.25\cdot 8\cdot s^2 + 0.60547s + 0.10875\cdot 8s + 0.04157 + 0.10875K_1 + 0.25K_1s}$$

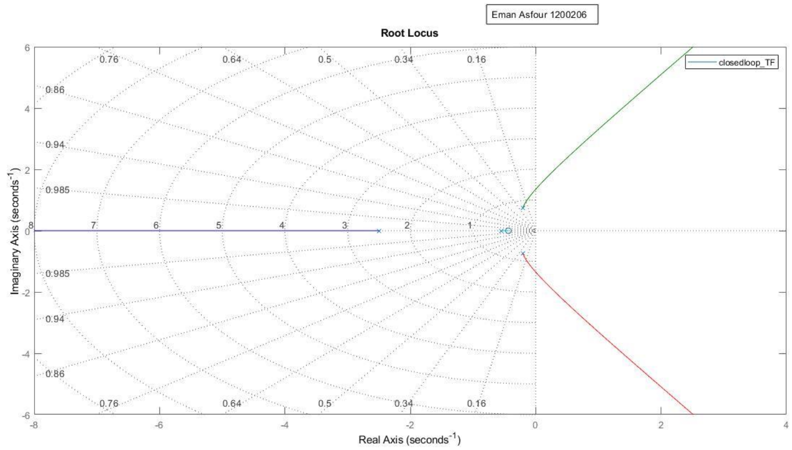
$0.25 \times 8 = 2$

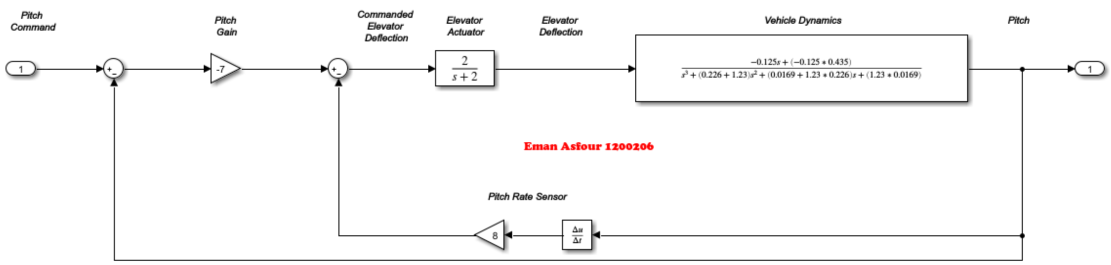
$0.10875 \times 8 = 0.87$

$$G_c = \frac{K_1\, 0.25s + 0.10875}{s^4 + 3.456s^3 + 3.20388s^2 + 2s^2 + \overbrace{(0.60547s + 0.87s)}^{1.475+7s} + 0.04457 + 0.10875K_1 + 0.25K_1s}$$

Eman Ashour

1200206





A =

-1.4560 -0.2949 -0.0208 2.0000

1.0000 0 0 0

0 1.0000 0 0

0 -0.8750 -0.3806 -2.0000

B =

0

0

0

-7

C =

0 -0.1250 -0.0544 0

D =

0

num =

0 0 0 1.7500 0.7612

den =

1.0000 3.4560 3.2069 2.3605 0.8028

closed =

1.75 s + 0.7612

----

s^4 + 3.456 s^3 + 3.207 s^2 + 2.361 s + 0.8028

Continuous-time transfer function.

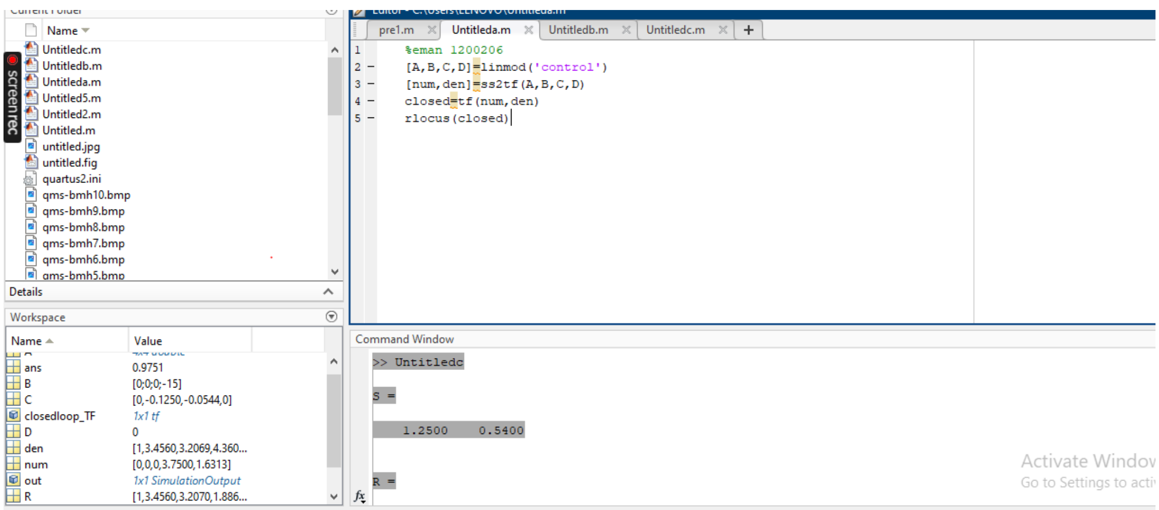
>> isstable (closed)

ans =

logical

1

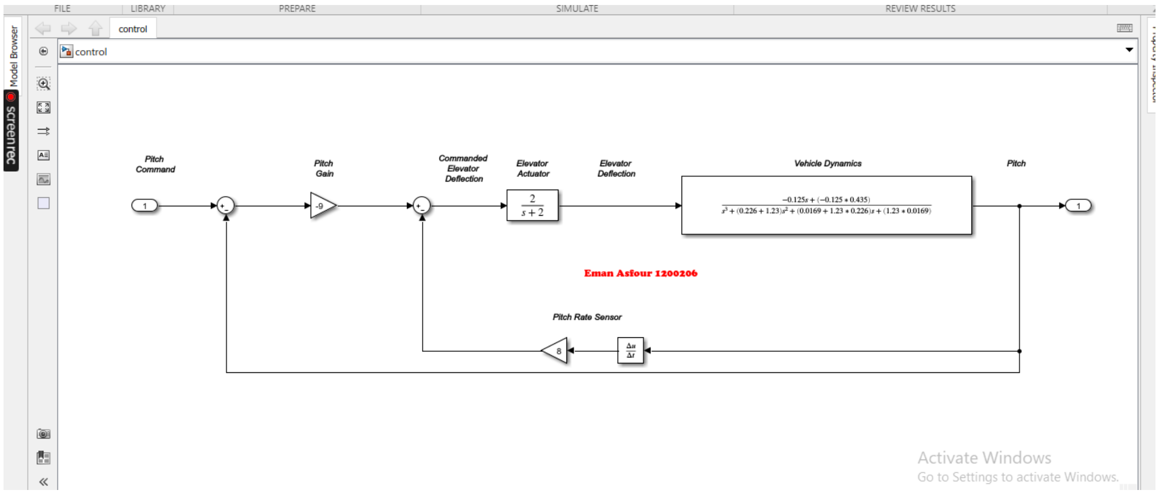
```
%eman 1200206
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
closed=tf(num,den)
rlocus(closed)
```

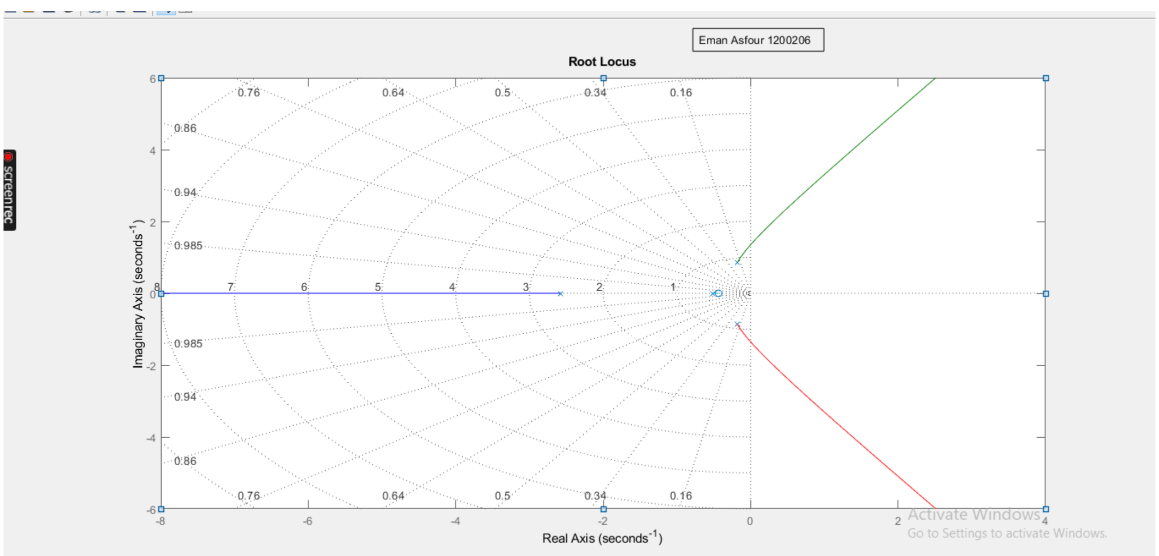


5. Determine values of K1 (with at least three different fixed values of K2) for which the system is stable.

- *Case 1*

K2=-8
K1=-9





A =

$$\begin{matrix} -1.4560 & -0.2949 & -0.0208 & 2.0000 \\ 1.0000 & 0 & 0 & 0 \\ 0 & 1.0000 & 0 & 0 \\ 0 & -1.1250 & -0.4894 & -2.0000 \end{matrix}$$

B =

0

0

0

-9

C =

0   -0.1250   -0.0544   0

D =

0

num =

0   0   0   2.2500   0.9788

den =

1.0000   3.4560   3.2069   2.8605   1.0203

closed =

```
             2.25 s + 0.9788
  -------------------------------------------
  s^4 + 3.456 s^3 + 3.207 s^2 + 2.861 s + 1.02
```

Continuous-time transfer function.

>> isstable (closed)

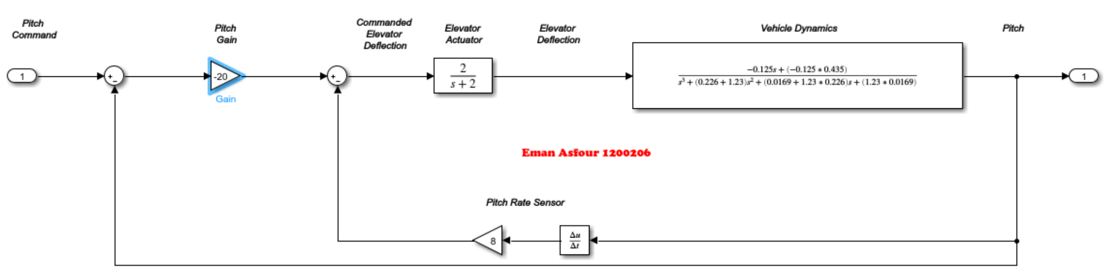
ans =

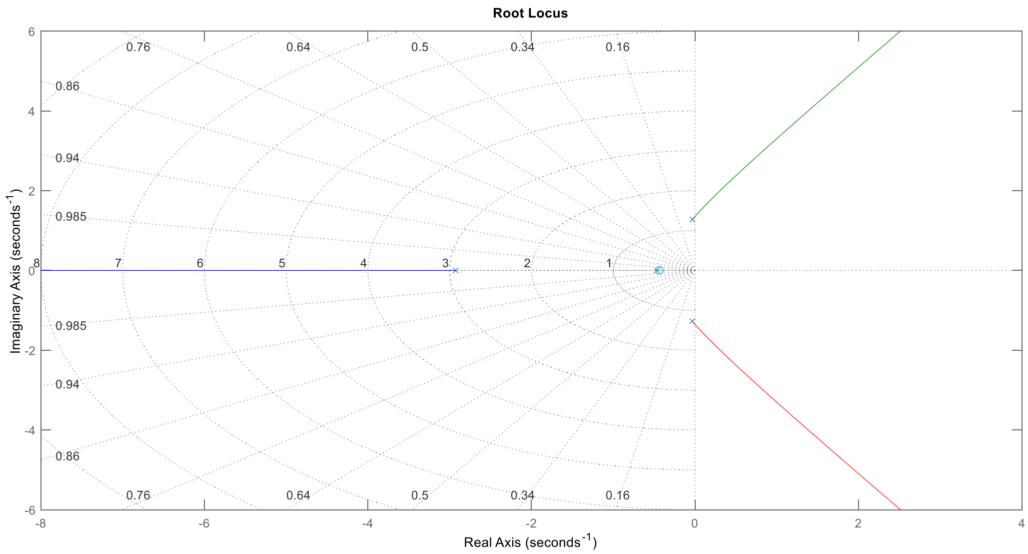
logical

1

- *Case 2*

K2=-8
K1=-20






```
>> Untitleda

A =

  -1.4560   -0.2949   -0.0208    2.0000
   1.0000         0         0         0
        0    1.0000         0         0
        0   -2.5000   -1.0875   -2.0000

B =

     0
     0
     0
   -20

C =

        0   -0.1250   -0.0544         0

D =

     0

num =

        0         0         0    5.0000    2.1750

den =

   1.0000    3.4560    3.2069    5.6105    2.2166

closed =
```

5 s + 2.175

----

s^4 + 3.456 s^3 + 3.207 s^2 + 5.611 s + 2.217

Continuous-time transfer function.

>> isstable (closed)

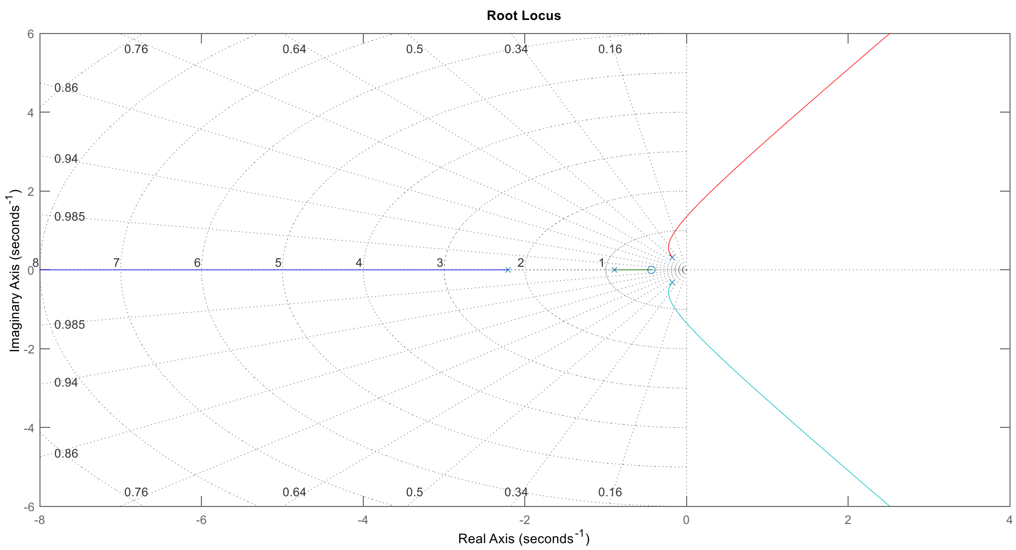
ans =

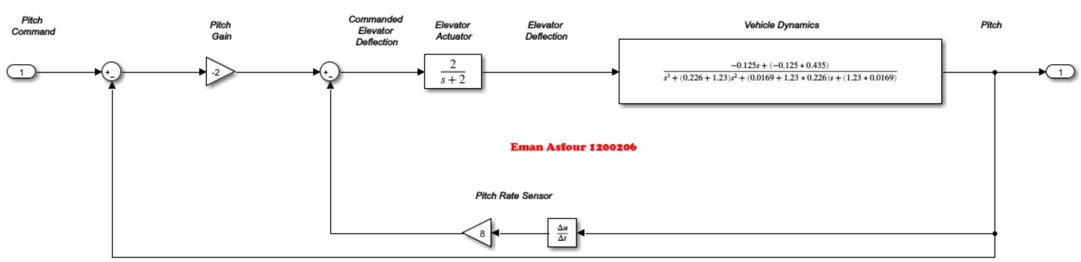
logical

1

- *Case 3*

K2=-8
K1=-2





>> Untitleda

A =

-1.4560 -0.2949 -0.0208 2.0000

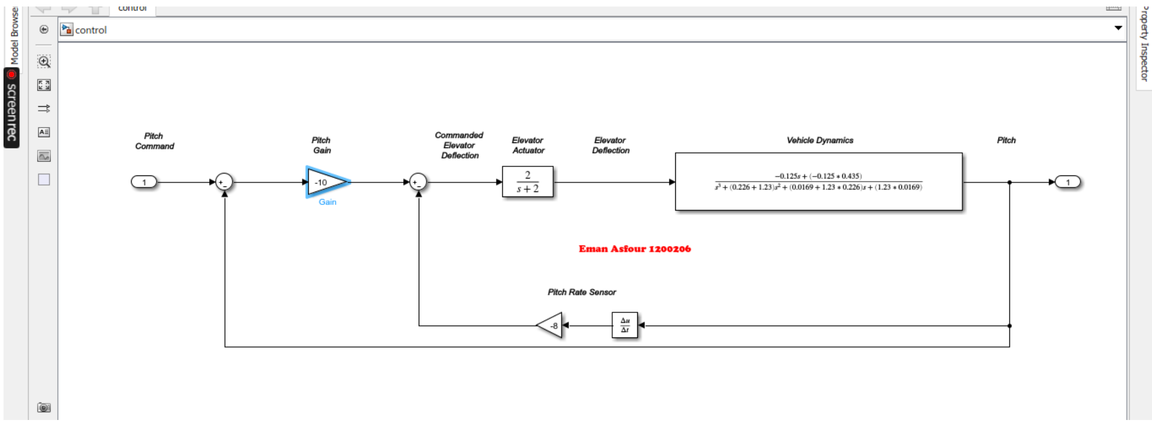
```
    1.0000         0         0         0
         0    1.0000         0         0
         0   -0.2500   -0.1087   -2.0000

B =

     0
     0
     0
    -2

C =

         0   -0.1250   -0.0544         0

D =

     0

num =

         0         0         0    0.5000    0.2175

den =

    1.0000    3.4560    3.2069    1.1105    0.2591

closed =

                 0.5 s + 0.2175
  ---------------------------------------------
  s^4 + 3.456 s^3 + 3.207 s^2 + 1.111 s + 0.2591

Continuous-time transfer function.

>> isstable (closed)

ans =

  logical

   1
```

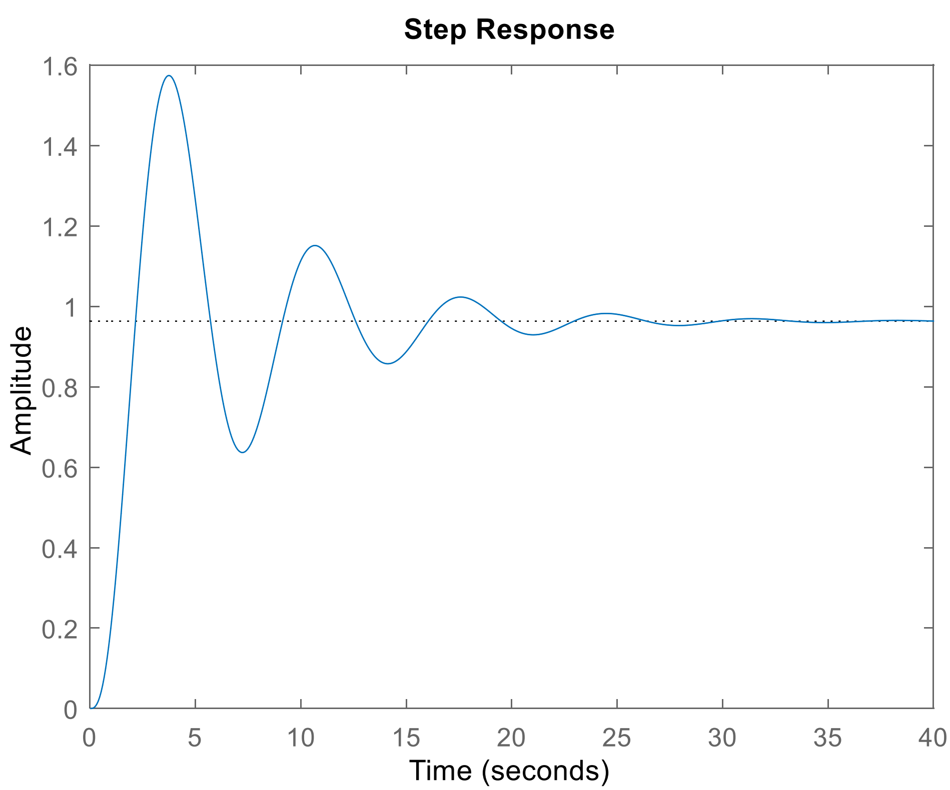
6. Determine the step response for four different values of K1 to include different stability conditions (if possible: stable oscillatory, stable non-oscillatory, unstable, and at the limit of stability) Note: you can change K2 value to achieve the requirements if necessary).

- ***Stable Oscillatory***

K1=-10

K2=-8





```
>> Untitledb
A =
   -1.4560   -0.2949   -0.0208    2.0000
    1.0000         0         0         0
         0    1.0000         0         0
         0   -0.8750   -0.3806   -2.0000
B =
     0
     0
     0
    -7
C =
         0   -0.1250   -0.0544         0
D =
     0


num =
         0         0         0    1.7500    0.7612
den =
    1.0000    3.4560    3.2069    2.3605    0.8028
closed =
                 1.75 s + 0.7612
  ---------------------------------------------
  s^4 + 3.456 s^3 + 3.207 s^2 + 2.361 s + 0.8028
Continuous-time transfer function.
s =

  struct with fields:

        RiseTime: 1.2906
    SettlingTime: 22.1380
     SettlingMin: 0.6365
```

SettlingMax: 1.5740

Overshoot: 63.4163

Undershoot: 0

Peak: 1.5740

PeakTime: 3.7573

ans =

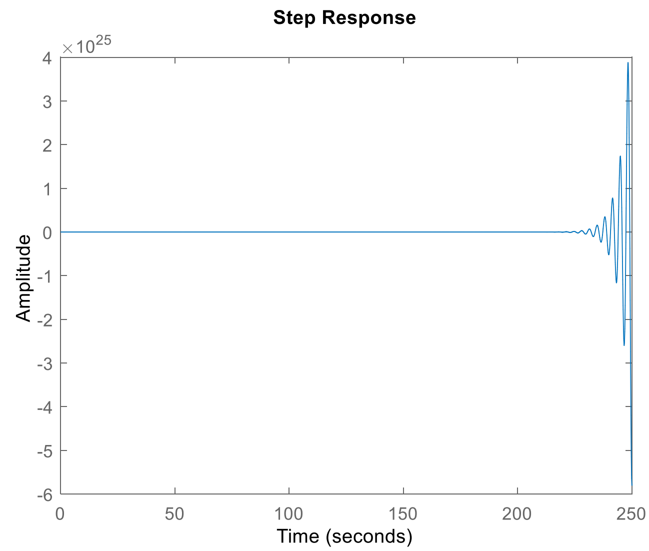
0.9632

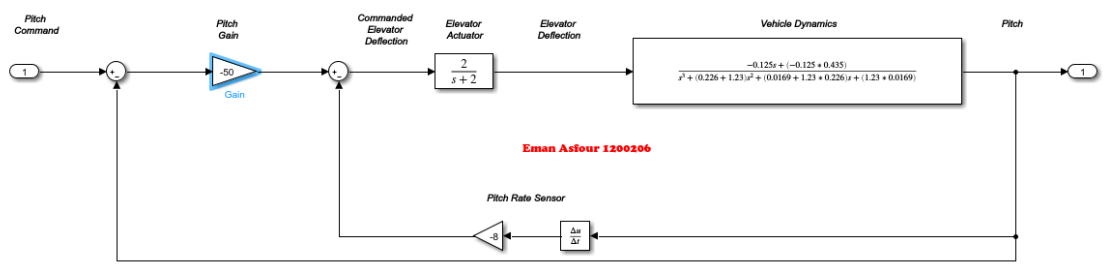
```
%eman 1200206
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
closed=tf(num,den)
step(closed)
s=stepinfo(closed)
dcgain(closed)
disp(step_info);
```

- ***Unstable***

K1=-50

K2=-8





>> Untitledb

A =

-1.4560 -0.2949 -0.0208 2.0000

1.0000 0 0 0

0 1.0000 0 0

0 -6.2500 -2.7188 -2.0000

B =

0

0

0

-50

C =

0 -0.1250 -0.0544 0

D =

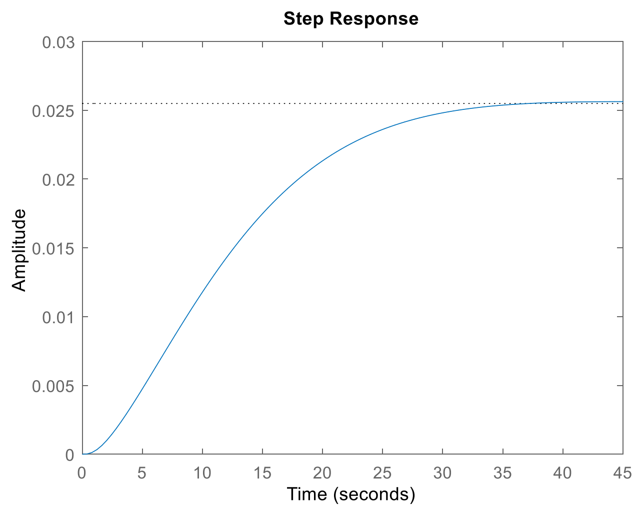
```
num =

         0         0         0   12.5000    5.4375

den =

    1.0000    3.4560    3.2069   13.1105    5.4791

closed =

                 12.5 s + 5.438
  ---------------------------------------------
  s^4 + 3.456 s^3 + 3.207 s^2 + 13.11 s + 5.479

Continuous-time transfer function.

s =

  struct with fields:

        RiseTime: NaN
    SettlingTime: NaN
     SettlingMin: NaN
     SettlingMax: NaN
       Overshoot: NaN
      Undershoot: NaN
            Peak: Inf
        PeakTime: Inf

ans =

    0.9924
```

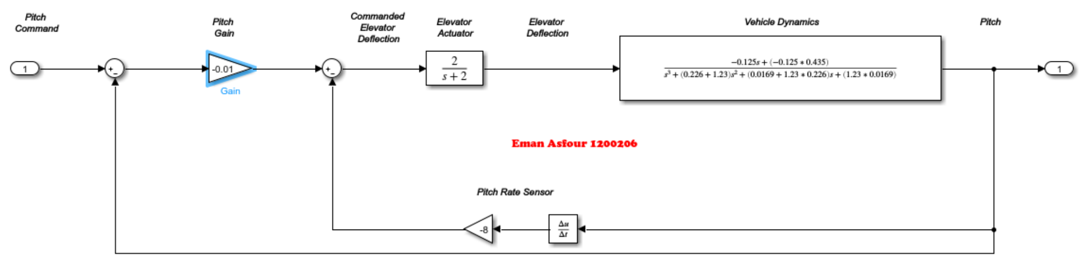
```
%eman 1200206
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
closed=tf(num,den)
step(closed)
s=stepinfo(closed)
dcgain(closed)
disp(step_info);
```

➢ *Stable Non-Oscillatory*

K1=-0.01

K2=-8





s =

struct with fields:

RiseTime: 19.8389

SettlingTime: 31.1158

SettlingMin: 0.0231

SettlingMax: 0.0256

Overshoot: 0.5308

Undershoot: 0

Peak: 0.0256

PeakTime: 45.1095

ans =

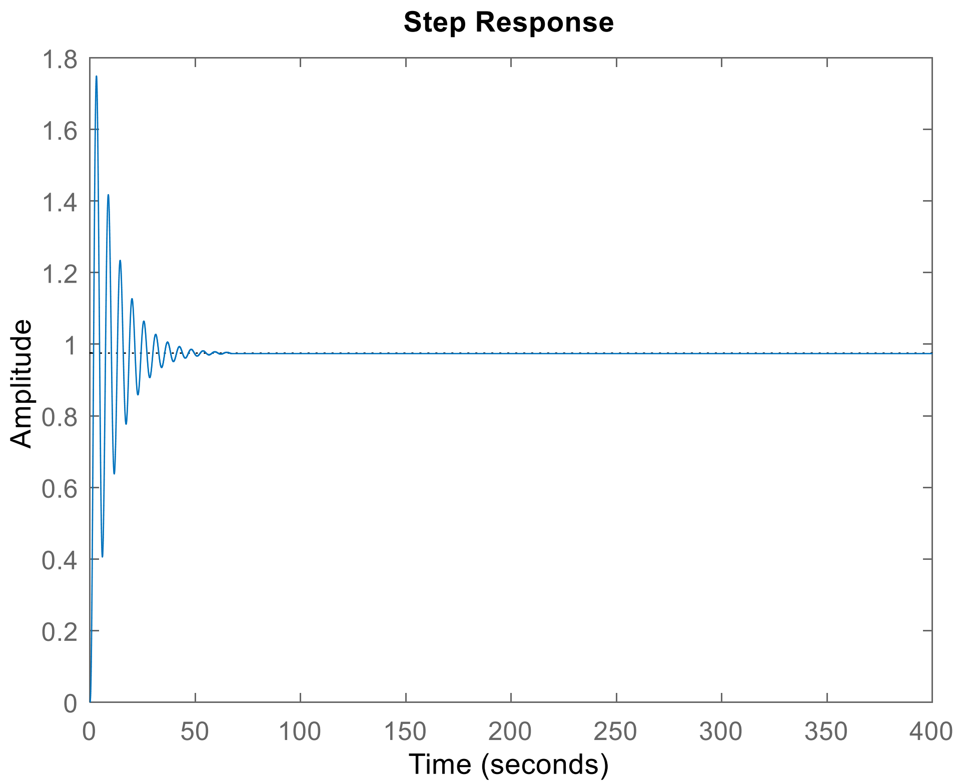
0.0255

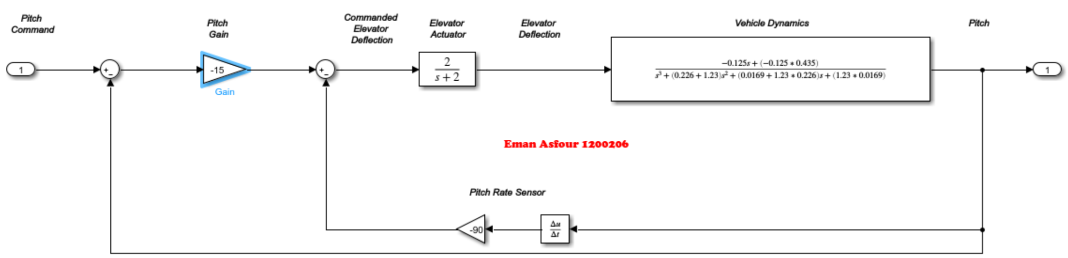
```
%eman 1200206
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
closed=tf(num,den)
step(closed)
s=stepinfo(closed)
dcgain(closed)
disp(step_info);
```

- ***Limit of Stability***

K1=-15

K2=-90





s =

struct with fields:

RiseTime: 1.0300

SettlingTime: 40.2179

SettlingMin: 0.4056

SettlingMax: 1.7491

Overshoot: 79.3692

Undershoot: 0

Peak: 1.7491

PeakTime: 3.1365

ans =

0.9751

```
%eman 1200206
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
closed=tf(num,den)
step(closed)
s=stepinfo(closed)
dcgain(closed)
disp(step_info);
```

7. Determine the transient parameters (rising time, settling time, overshoot (if exists) and compare and discuss the different responses

➢ ***Stable Oscillatory***

```
s =
  struct with fields:
        RiseTime: 1.2906
    SettlingTime: 22.1380
     SettlingMin: 0.6365
     SettlingMax: 1.5740
       Overshoot: 63.4163
      Undershoot: 0
            Peak: 1.5740
        PeakTime: 3.7573
ans =
    0.9632
```

The system, represented in state-space form with matrices A, B, C, and D, exhibits stability according to its transfer function. However, the notable 63.41% overshoot in the step response raises concerns about potential oscillation. Despite a relatively fast rise time, the extended settling time may be problematic depending on application requirements. The gains K1 = -10 and K2 = -8 play specific roles, though their exact functions need clarification.

➢ ***Unstable***

```
s =
  struct with fields:
        RiseTime: NaN
    SettlingTime: NaN
     SettlingMin: NaN
     SettlingMax: NaN
       Overshoot: NaN
      Undershoot: NaN
            Peak: Inf
        PeakTime: Inf
ans =
    0.9924
```

The system exhibits clear signs of instability, as indicated by NaN and Inf values in the step response characteristics and the lack of convergence to a steady state. The chosen gains, K1 = -50 and K2 = -8, are likely responsible for this instability. The step response image visually confirms the persistent oscillations and increasing amplitude over time. To address the issue, a reassessment of controller gains, exploration of alternative control techniques, and a thorough analysis of system dynamics through pole-zero analysis are recommended

- ***Stable Non-Oscillatory***

```
s =

  struct with fields:

        RiseTime: 19.8389

    SettlingTime: 31.1158

     SettlingMin: 0.0231

     SettlingMax: 0.0256

       Overshoot: 0.5308

      Undershoot: 0

            Peak: 0.0256

        PeakTime: 45.1095

ans =

    0.0255
```

The system is confirmed to be stable, as evidenced by finite rise time, settling time, and the absence of NaN or Inf values in the step response characteristics. However, its performance metrics reveal a relatively slow response with a significant 53.08% overshoot that dampens out smoothly, reaching a final value of 0.0255 at a steady state. Visual analysis of the step response graph aligns with the numerical data, showing a smooth rise and a single overshoot peak that settles without additional oscillations. While stability is evident, the slower response and overshoot suggest potential room for performance improvement, emphasizing the need to consider adjustments to controller gains or explore alternative control techniques. The image further confirms the stable and non-oscillatory behavior, highlighting the importance of a thorough understanding of the application context and goals for a comprehensive evaluation and identification of suitable enhancements.

➢ *Limit of Stability*

```
s =

  struct with fields:

    RiseTime: 1.0300

    SettlingTime: 40.2179

     SettlingMin: 0.4056

     SettlingMax: 1.7491

       Overshoot: 79.3692

      Undershoot: 0

            Peak: 1.7491

        PeakTime: 3.1365

ans =

    0.9751
```

The system, while technically stable, exhibits characteristics that suggest it is in a precarious near-instability state. The fast rise time of 1.03 seconds indicates a prompt response, but the prolonged settling time of 40.22 seconds, coupled with an excessive 79.37% overshoot, raises concerns about potential oscillations. The gain values, K1 = -15 and K2 = -90, have seemingly pushed the system close to the edge of instability. Although the final value stabilizes at 0.9751, the step response graph likely depicts a rapid rise with significant overshoot, followed by prolonged oscillations that eventually dampen out.

8. Determine the steady-state error for the selected values of K1 and K2

It is done in the last part

9. Determine the canonical controller state-space representation of the feedback system

```
%Eman 1200206
S=[1.25 0.54]
R=[1 3.456 3.207 1.8861 0.584]
[ A D B C ]-tfss(S, R);
s1=transopse(A)
s1=transopse(D)
s1=transopse(B)
%9. Determine the canonical controller state-space
representation of the feedback system
```

>> Untitledc

S =

1.2500 0.5400

R =

1.0000 3.4560 3.2070 1.8861 0.5840

10. Study the effect of using K2=0 (eliminating the rate feedback controller) on the system stability, transient performance, and steady-state error under stability conditions (for at least three cases, if exists).

From part 6

Stable Oscillatory

K1 = *-10*

Unstable

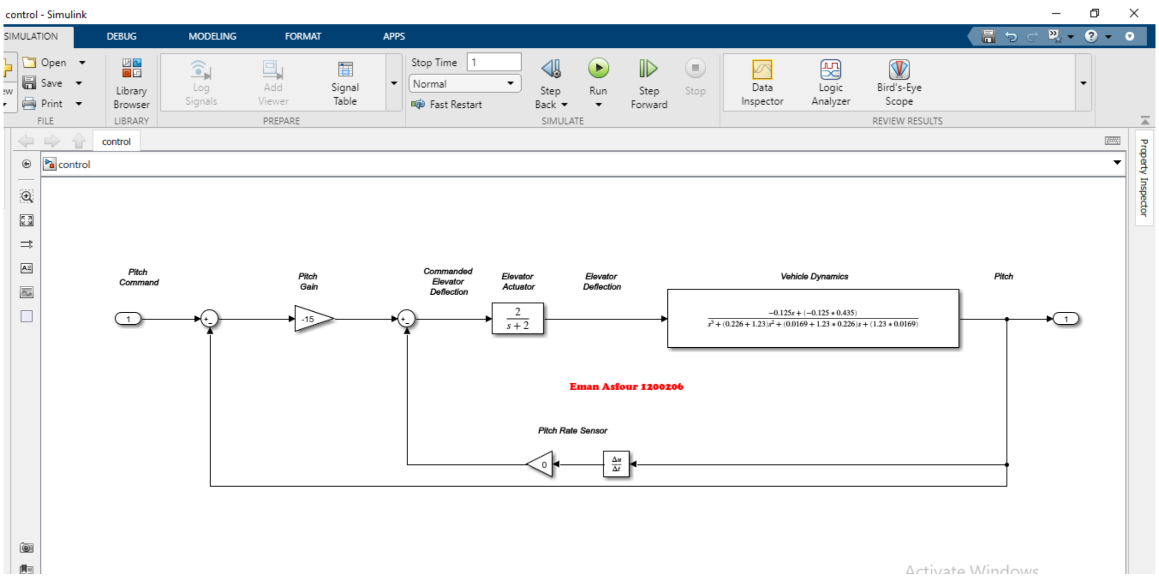
K= -50

Stable non-oscillatory

K1 = -0.01

Limit of stability

K= -15



s =

struct with fields:

RiseTime: 1.0300

SettlingTime: 40.2179

SettlingMin: 0.4056

SettlingMax: 1.7491

Overshoot: 79.3692

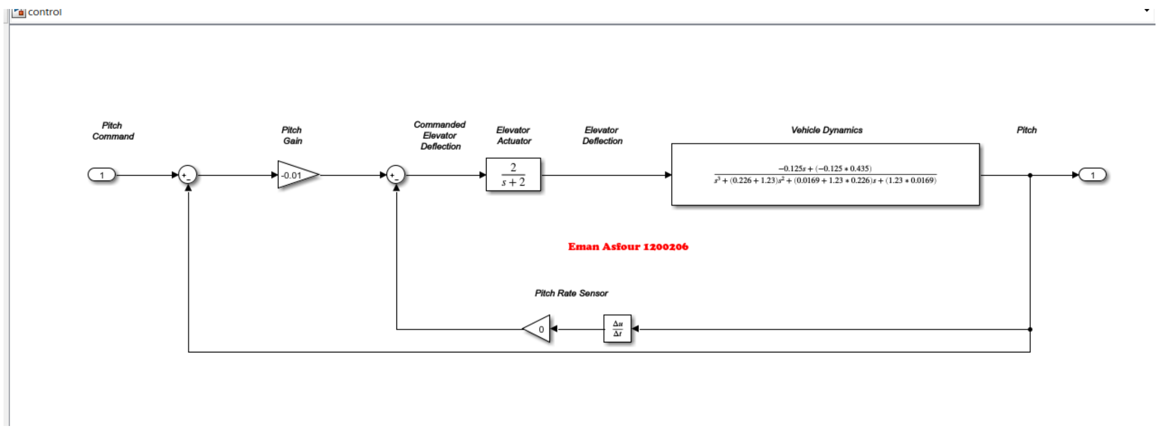
Undershoot: 0

Peak: 1.7491

PeakTime: 3.1365

ans =

0.9751



s =

struct with fields:

RiseTime: 19.8389

SettlingTime: 31.1158

SettlingMin: 0.0231

SettlingMax: 0.0256

Overshoot: 0.5308

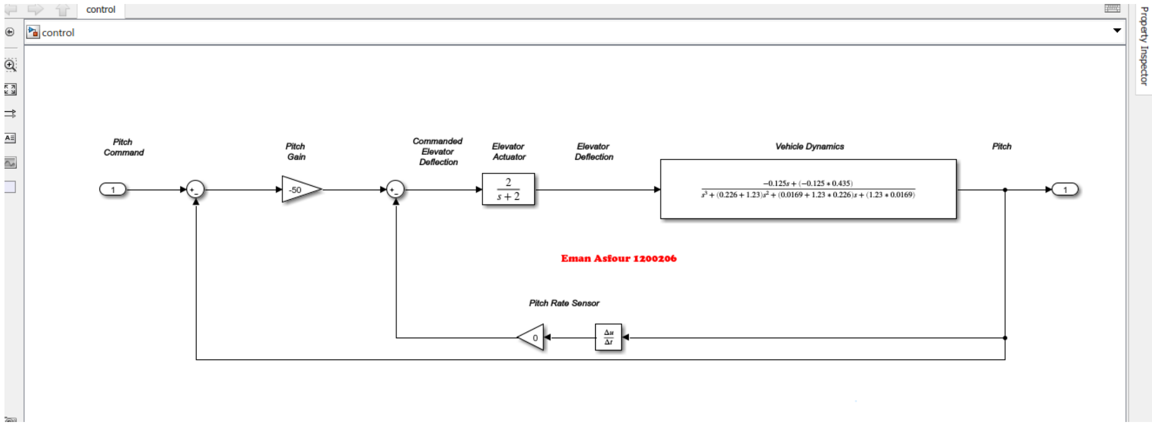
Undershoot: 0

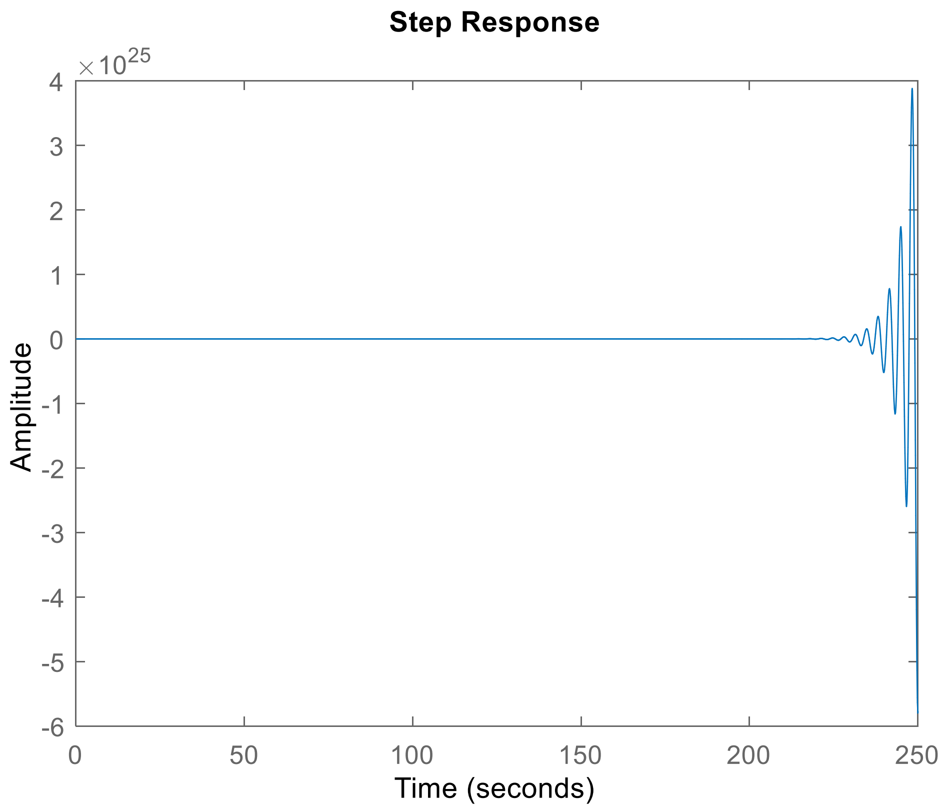
Peak: 0.0256

PeakTime: 45.1095

ans =

0.0255





s =

struct with fields:

RiseTime: NaN

SettlingTime: NaN

SettlingMin: NaN

SettlingMax: NaN

```
        Overshoot: NaN
       Undershoot: NaN
             Peak: Inf
         PeakTime: Inf


ans =

    0.9924
>> Untitled4

A =

   -1.4560   -0.2949   -0.0208    2.0000
    1.0000         0         0         0
         0    1.0000         0         0
         0   -6.2500   -2.7188   -2.0000


B =

     0
     0
     0
   -50


C =
```

```
         0   -0.1250   -0.0544         0


D =

     0


num =

         0         0         0   12.5000    5.4375


den =

    1.0000    3.4560    3.2069   13.1105    5.4791


s =

  struct with fields:

        RiseTime: NaN
    SettlingTime: NaN
     SettlingMin: NaN
     SettlingMax: NaN
       Overshoot: NaN
```

```
      Undershoot: NaN
            Peak: Inf
        PeakTime: Inf


ans =

    0.9924
>>
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
s=stepinfo(closed)
dcgain(closed)
rlocus(closed)
```

Conclusion

In this project, we have chosen to work with a closed-loop system involving an unmanned free-swimming submersible vehicle, specifically focusing on controlling its pitch using Simulink. The initial steps involve deriving the transfer function, incorporating parameters K1 and K2 into the model, and comprehensively understanding how the root locus behaves in response to these parameters. Through practical application and analysis, we aim to gain a deep understanding of how variations in these parameters influence system stability. Moreover, we are actively engaged in drawing and coding in MATLAB to visualize and explore different stability scenarios, ranging from stable oscillatory and stable non-oscillatory to unstable conditions. By delving into these diverse stability states, including the limit of stability, this project provides a holistic learning experience, enabling us to grasp not only theoretical concepts but also develop practical skills in MATLAB for real-world control system applications.

Appendix

```
%eman 1200206
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
closed=tf(num,den)
rlocus(closed)
 %---------------------------------------------------------------------
%eman 1200206
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
closed=tf(num,den)
step(closed)
s=stepinfo(closed)
dcgain(closed)
disp(step_info);
 %---------------------------------------------------------------------
%Eman 1200206
S=[1.25 0.54]
R=[1 3.456 3.207 1.8861 0.584]
[ A D B C ]-tfss(S, R);
s1=transopse(A)
s1=transopse(D)
s1=transopse(B)
%9. Determine the canonical controller state-space representation of the
feedback system
 %---------------------------------------------------------------------
[A,B,C,D]=linmod('control')
[num,den]=ss2tf(A,B,C,D)
s=stepinfo(closed)
dcgain(closed)
rlocus(closed)
```