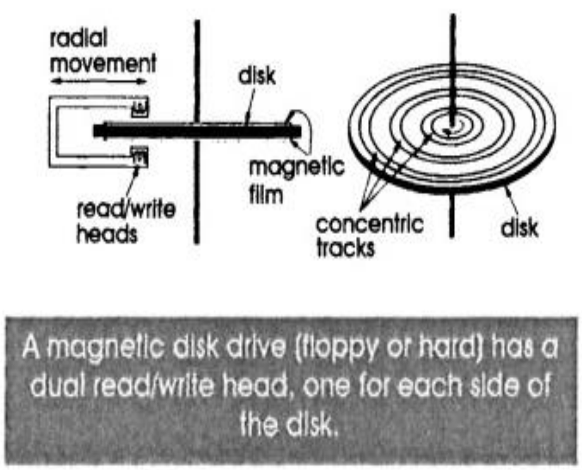
Winter 2021

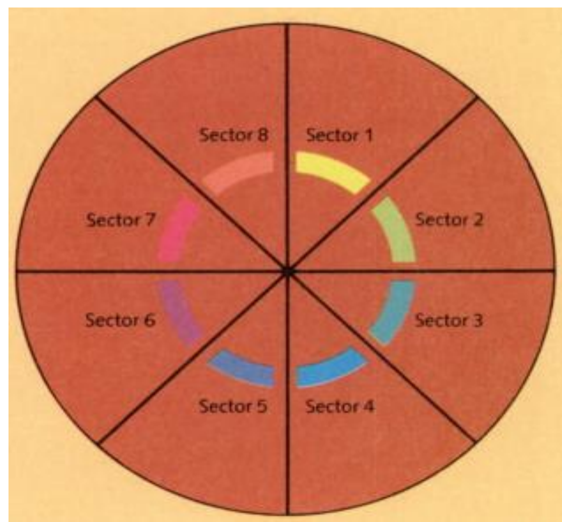
ECE 560 - Modern Control Theory

Floppy Disk Drive Position Control

Introduction:

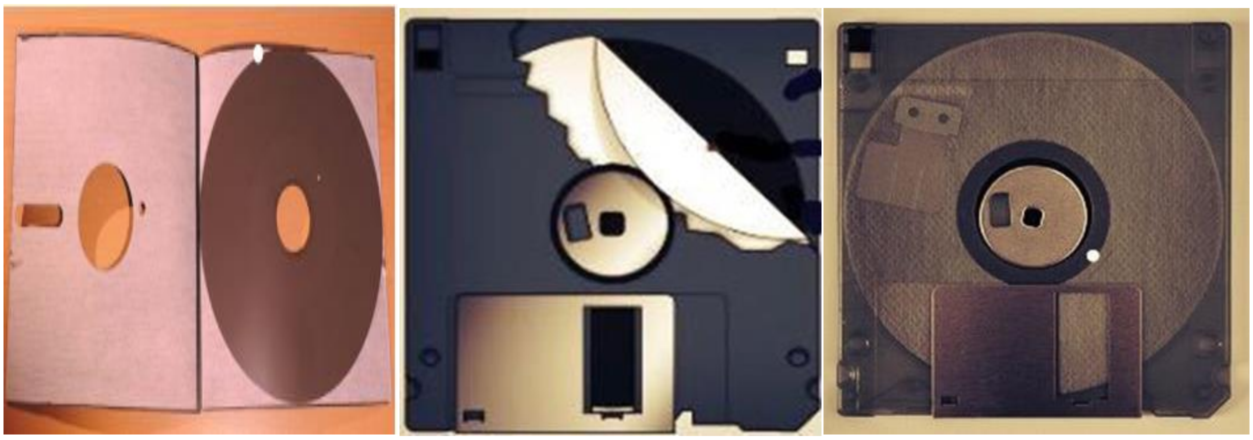
A floppy disk drive is a position control system in which a read-write head is positioned over a magnetic desk and it contains a magnetically coated round plastic medium that has a square plastic cover with a small oblong opening in both sides to allow the drive's heads to read and write data and a large hole in the center to allow the magnetic medium to spin by rotating it from its middle hole. A traditional floppy disk can store up to 1.4 megabytes of data that can be read at rates of several hundred bits per second. The speed of floppy disk must be accelerated up to 360 r.p.m. besides assuring that the magnetic head moved to the correct track and sector.





Inside the cover there are two layers of fabric with the magnetic medium sandwiched in the middle. These layers are designed to decrease friction between the medium and the outer covers besides catch particles of scattered pieces from the disk to prevent them from accumulating on the heads. The cover is made from one-part sheet double folded with flaps glued together. There is LED/photo transistor pair that located near the center of the disk and its function is detecting the index hole once per rotation in the magnetic disk. The LED transistor is useful to provide information about the angular beginning of each track and whether or not the disk is rotating at the accurate speed. Early floppy disks had physical holes for each sector and were named hard sectored disks. Floppy disks became widely in the period between 1980s and

1990s for personal computers uses and for more than two decades, these floppy disks were the primary external writable storage devices used. Also, the floppy disks offered the abilities to distribute software, carry data, and create backups information. Floppy disks were designed to store a computer's operating system. Old days personal computers had an elementary operating system and basic stored in read only memory (ROM) with the option of loading a more advanced operating system from a floppy disk. By the early 1990's the increasing of software size required large packages like Windows or Adobe Photoshop that meant a dozen disks or more. In 1996 there were approximately a five billion standard floppy disks in use then distribution of larger packages was gradually replaced by CD-ROMs, DVDs, and online distribution.



Floppy disk has holes at the bottom left and right that refer whether the disk is write-protected and whether it is high-density. Major issues of the floppy disk is its high sensitivity even though inside a closed plastic covering. The disk medium is very susceptive to dust, condensation, magnetic fields and high temperature degrees. Floppy disks production included an extensive set of warnings that caution users not to expose them to dangerous conditions or removing the disk from the drive while the magnetic media is still spinning because they can damage the disks, drive head, or stored data.

The floppy disk is styled to respond to a command from a computer to position itself at a particular track on the desk. A floppy disk usually differs among three sizes, 8 inches, 5.5 inches and 3.5 inches and it became smaller to match the technology advanced requirement. Advanced floppy disks had four basic components:

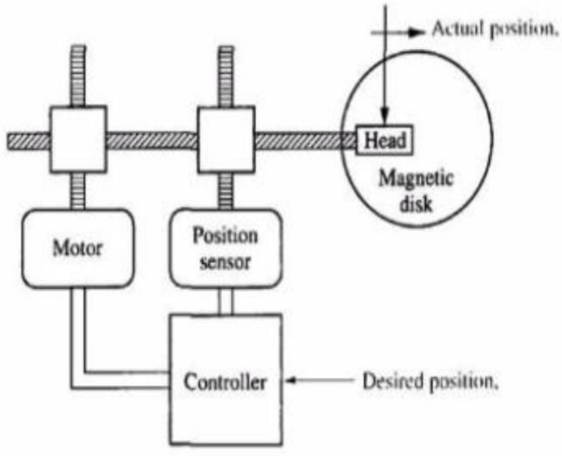
➢ Magnetic read/write heads (one or two).

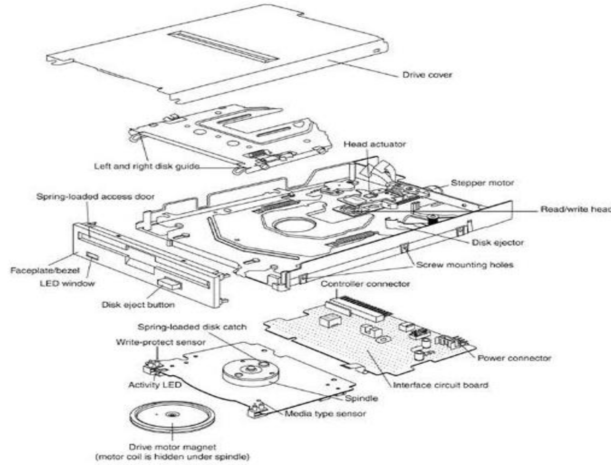
➢ A spindle clamping device that held the disk in place as it was spinning 300 to 360 rotations per minute.

➢ A frame with levers that opened and closed the device.

➢ A circuit board that contained all the electronics.

➢ A physical representation of the system and block diagram are shown in the figure below:

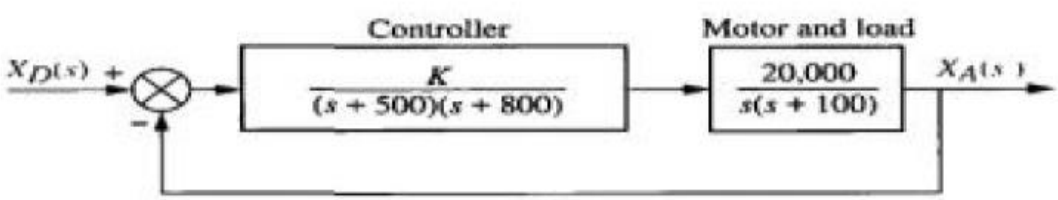




Objective:

The project aims to design a system of floppy disk drive that has main parts:

- Controller.
- Motor and load.



And present the discussion of the following:

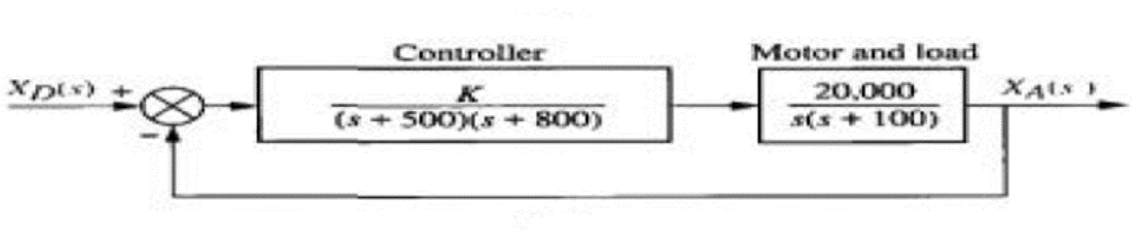
- System modeling (transfer function, state scope, differential equation).
- System stability.
- System controllability.
- System observability.
- Results of percent overshoot and settling time.
- Comparison for different values of K that makes system stable.
- Simulation.

The other sections of project’s report will consider the pros and cons of unity feedback design, pole placement design, and state feedback design. Also, I’ll provide a comparison chart among three designs to show the differences.

Section#1- System modeling:

Transfer Function:

This part will show the transfer function of the following system and provide the step response:



$$TF = \frac{G}{1+HG}$$

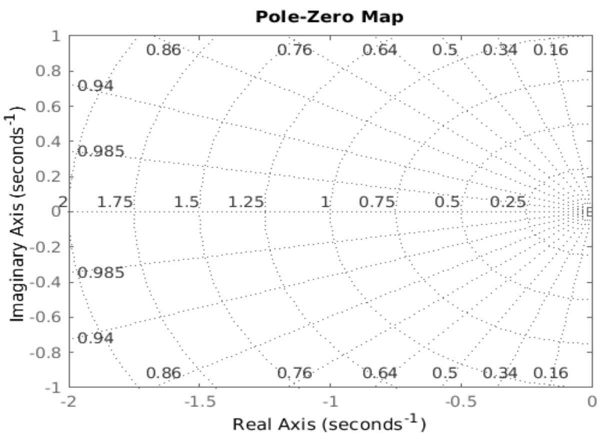
$$TF = \frac{\frac{20000K}{S(S+500)(S+800)(S+100)}}{1+\frac{20000K}{S(S+500)(S+800)(S+100)}}$$

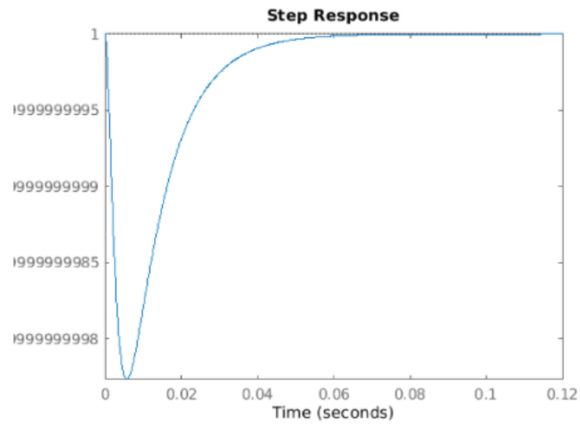
$$TF = \frac{20000K}{S(S+500)(S+800)(S+100)+20000K}$$

$$TF = \frac{20000K}{S\left(S^3+1400S^2+530000S+40000000\right)+20000K}$$

$$TF = \frac{20000K}{S^4+bS^3+cS^2+dS+e};\, e = 20000K$$

```
s = tf('s');
G = (20000*k)/s*(s+500)*(s+800)*(s+100);
k = 50;
T = feedback(G*k,1);
pzmap(T)
grid, axis([-2 0 -1 1])
clf
step (T);
```





Section#2- A, B, C, D Matrix:

The previous section presents the formula of system transfer function therefore, I can find the system matrices:

$$\begin{pmatrix} \dot{X}1 \\ \dot{X}2 \\ \dot{X}3 \\ \dot{X}4 \end{pmatrix} = \begin{pmatrix} -b & -c & -d & -e \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X1 \\ X2 \\ X3 \\ X4 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} U$$

$$Y = \begin{pmatrix} 0 & 0 & 0 & 20000K \end{pmatrix} + \begin{pmatrix} 0 \end{pmatrix} U$$

$$A = \begin{pmatrix} -b & -c & -d & -e \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, B = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 0 & 0 & 20000K \end{pmatrix}, D = \begin{pmatrix} 0 \end{pmatrix}$$

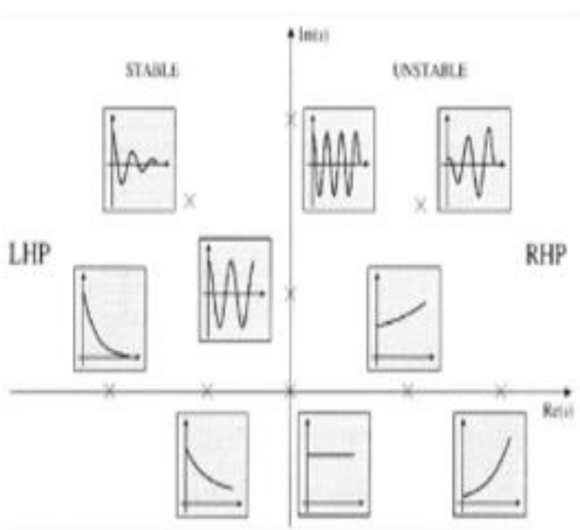
Section 3-Stability:

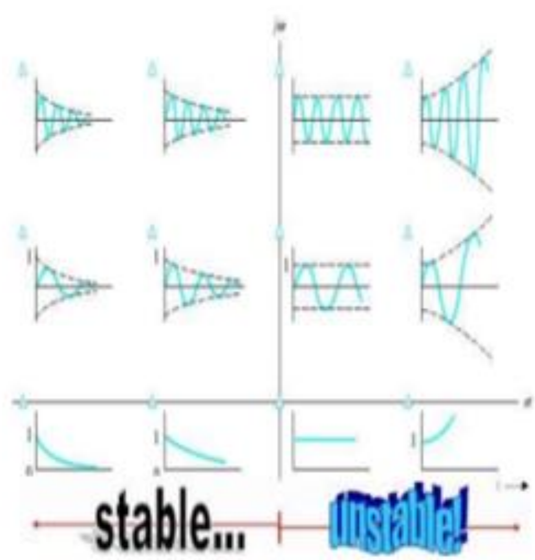
Stability of the control system is ability that enable this system to reach the steady-state and keep on it for a specific input even after few changes in the parameters of the system so under a bounded input the stable system provides a bounded output.

Stability is an important factor in a control system because if the system is unstable, the output of this system can continue increasing to infinity even though the input to the system is finite, so this critical situation can cause multiple of concern problems and the final results will be disappointed with a failure in function besides sometimes high cost.

Stability of any control system can be divided among three catalogs:

- Absolutely stable system: the system is stable for each component value of system; in other words, all poles of the system are located in the left half of s plane.
- Conditionally stable system: the system is stable for a specific components value.
- Marginally stable system: the system is stable when an output signal is related directly to constant amplitude and constant frequency of vibrations for an input; in other words, when the system has two poles are located on the imaginary axis.





Routh-Hurwitz Stability Method:

To discuss the stability of the following transfer function of floppy disk drive according to Routh-Hurwitz method, I provided the table that shows the Routh array:

$$aS^4 + bS^3 + cS^2 + dS + e$$

s^4	a	c	e
s^3	b	d	0
s^2	$\frac{bc-ad}{b}=g$	$\frac{de-0}{d}=e$	0
s^1	$\frac{gd-be}{g}=h$	0	0
s^0	$\frac{he-0}{h}=e$	0	0

Finding Range Value of K for Stability:

s^4	**1**	**530000**	**20000K**
s^3	**1400**	**40000000**	**0**
s^2	**501428.6**	**20000K**	**0**
s^1	**39999941.9-55.8K**	**0**	**0**
s^0	**20000K= e**	**0**	**0**

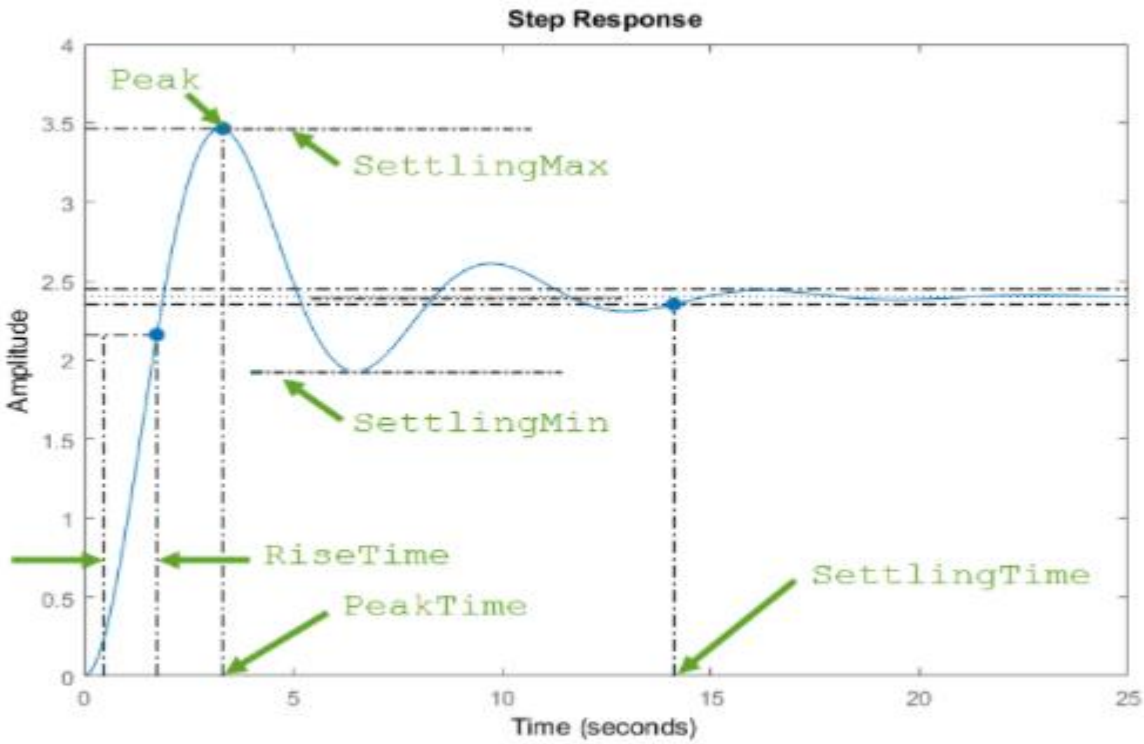
The previous results show that system is stable when e>0→20,000K>0→K>0.

Also, 39999941.9-55.8K>0→39999941.9>55.8K→K>716,844.837.

So, I can conclude from previous results that the acceptable range for K values in my design can be expressed as: [0<K<716,844.837] to keep the system stable.

Section#4-Design for Settling Time To=50msec:

Settling Time is the time after the response arrives to its steady-state condition with value approximately equal to 98% of its final value and this time period nearly equal to four times of time constant of a signal. Also, settling time is needed to enable the signal output to reach and stay within a given error band following some input stimulus.

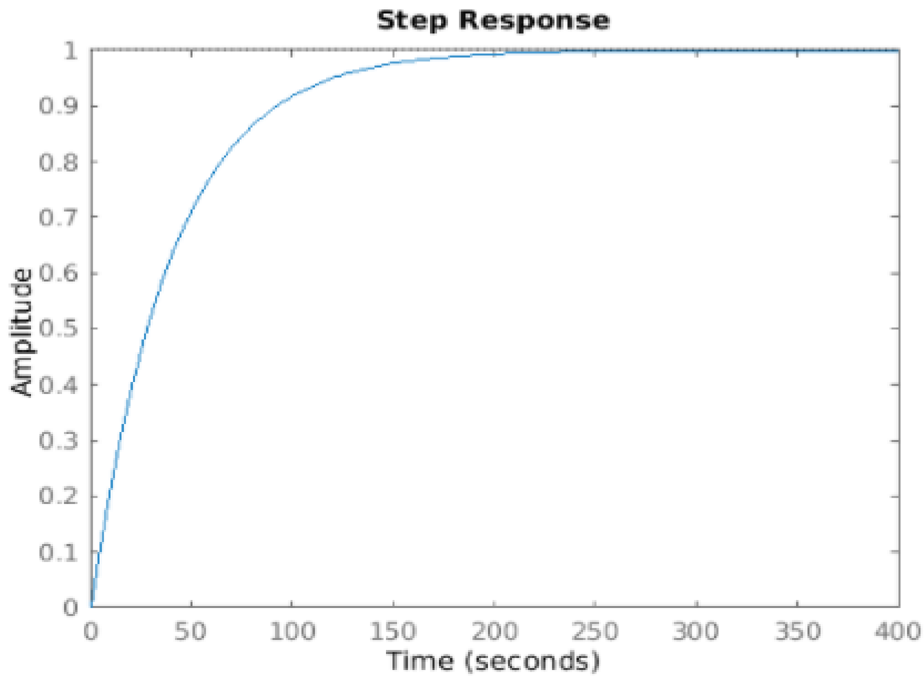


the expiration of settling time can be given as:

$$t_s = \frac{4}{\zeta \omega_n}$$

settling time is considered an important parameter for robust performance of control systems to produce an accurate data, therefore the signal output must settle before converting the data from one state to another one. Moreover, settling time is usually not an easy parameter to measure. In my design, I choose settling time equal to 50msec and I calculate the value of K that satisfies with this condition.

```
sys = tf([20000*k],[1 1400 530000 40000000 20000*k]);
k=50;
step(sys)
S = stepinfo(sys)
```

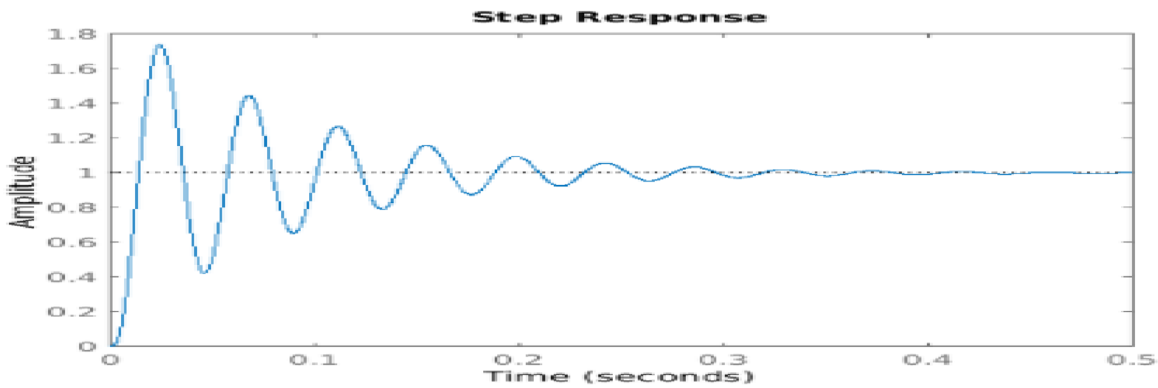



```
S = struct with fields:

        RiseTime: 87.8513
    SettlingTime: 156.4441
     SettlingMin: 0.9045
     SettlingMax: 1.0000
       Overshoot: 0
      Undershoot: 0
            Peak: 1.0000
        PeakTime: 421.6938
```

For different value of K, I got the following results:

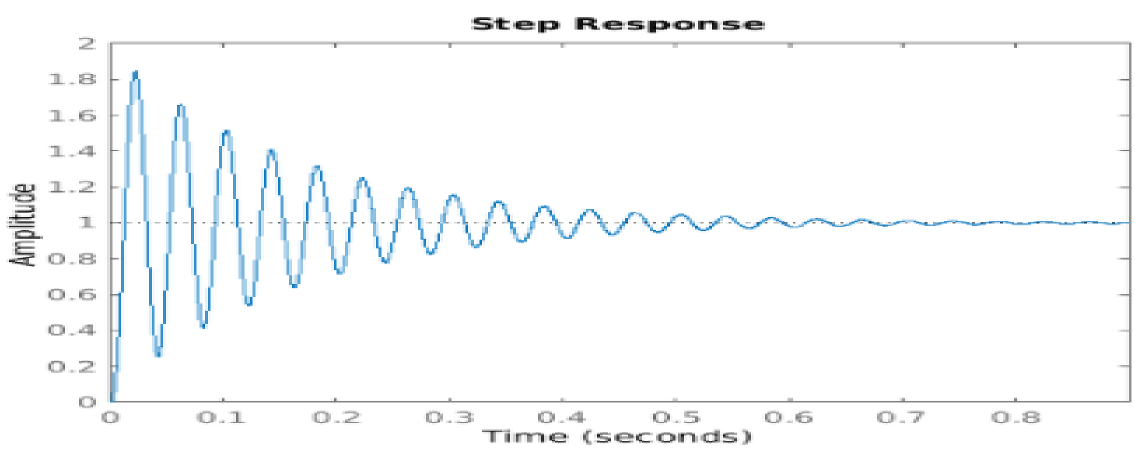
```
K=500000;
a=[20000*K];
b=[1 1400 530000 40000000 20000*K];
sys=tf(a,b);
step(sys);
S=stepinfo(sys)
```




```
S = struct with fields:

        RiseTime: 0.0081
    SettlingTime: 0.3317
     SettlingMin: 0.4239
     SettlingMax: 1.7427
       Overshoot: 74.2659
      Undershoot: 0
            Peak: 1.7427
        PeakTime: 0.0246
```

```
K=600000;
a=[20000*K];
b=[1 1400 530000 40000000 20000*K];
sys=tf(a,b);
step(sys);
S=stepinfo(sys)
```

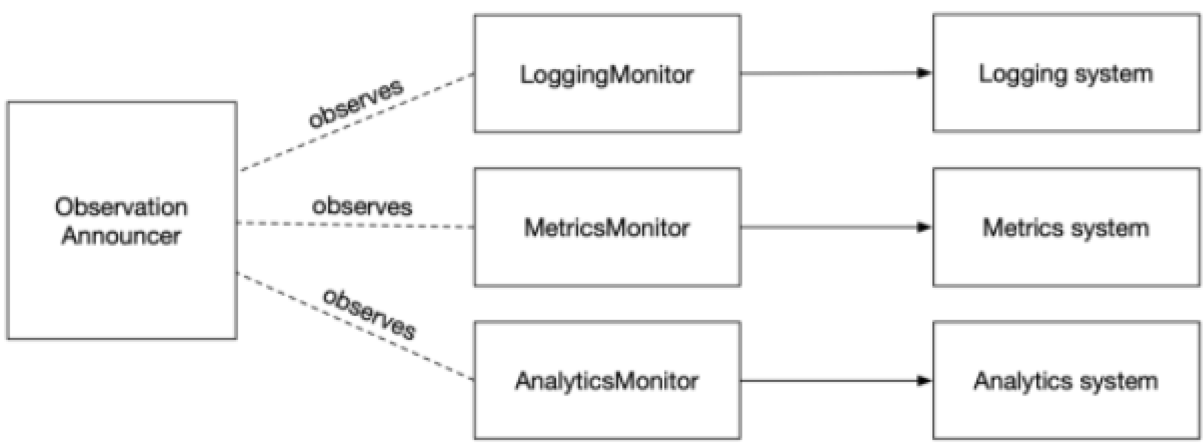



```
S = struct with fields:

        RiseTime: 0.0073
    SettlingTime: 0.6276
     SettlingMin: 0.2526
     SettlingMax: 1.8439
       Overshoot: 84.3901
      Undershoot: 0
            Peak: 1.8439
        PeakTime: 0.0228
```

Section#5-Observability:

Observability is the capability of the system to define its internal states by observing the behavior of signal output during a limited time interval when the input is applied to the system, then the system can supply all important data that are required to estimate all particular states of the system . Observability is a necessary property of the control system because it confirms the expected approach of the control system. On the other hand, if a state is not observable then the controller will have no ability to decide its behavior from by dealing with the available system output and this case will lead to unstable system. Observability is helpful in the complicated system that can include many parts and many stages because it isn't only give the ability to figure out the problem, it allows to go deeply in analysis and understanding of any abnormal behavior that can exist in the system and affect the total function of the configuration.



In my design the calculation of observability is shown below to confirm also the system stability

$$O = \begin{pmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{pmatrix} : A = \begin{pmatrix} -b & -c & -d & -e \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad \& \quad C = \begin{pmatrix} 0 & 0 & 0 & 20000K \end{pmatrix}$$

$$O = \begin{pmatrix} 0 & 0 & 0 & 20000K \\ 0 & 0 & 20000K & 0 \\ 0 & 20000K & 0 & 0 \\ 20000K & 0 & 0 & 0 \end{pmatrix}$$

$$\det(O) = \det\begin{pmatrix} 0 & 0 & 0 & 20000K \\ 0 & 0 & 20000K & 0 \\ 0 & 20000K & 0 & 0 \\ 20000K & 0 & 0 & 0 \end{pmatrix} = 1.6E17K^4$$

Since Det(O) isn't equal to zero, so my system is observable.

```
K=50;
A=[-1400 -530000 -40000000 -20000*K; 1 0 0 0; 0 1 0 0; 0 0 1 0]
C=[0 0 0 20000*K]
O=[C; C*A; C*A*A; C*A*A*A]
D=det(O)
rref(O)
rank(O)
```

```
A = 4×4
       -1400     -530000   -40000000    -1000000
           1           0           0           0
           0           1           0           0
           0           0           1           0

C = 1×4
           0           0           0     1000000

O = 4×4
           0           0           0     1000000
           0           0     1000000           0
           0     1000000           0           0
     1000000           0           0           0

D = 1.0000e+24

ans = 4×4
     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1

ans = 4
```

Section#7-Controllability:

Controllability is the capability to control all the states of the system and it is significant characteristic that present the expectation about any achievable control performance. It depends on the possibility of forcing the system to a certain state by application of a control input to reach a specific state of the output after supplying a requirement of initial state of the input through a particular time. On the other hand, if the uncontrollable part is stable, then the unstable part can be stabilized by using a feedback over the controllable states. Controllability is a property of the coupling between the input and the state and thus involves the matrices A and B. The failure of achieving a standard controllable procedure may lead to highly cost losses and production difficulties because the system can face many problems that are difficult or impossible to fix.

In my design the calculation of controllability is shown below to confirm also the system stability:

$$C = \begin{pmatrix} B & AB & A^2B & A^3B \end{pmatrix} : A = \begin{pmatrix} -b & -c & -d & -e \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad \& \quad B = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & -b & b^2 - c & -b^3 + 2bc - d \\ 0 & 1 & -b & b^2 - c \\ 0 & 0 & -b & -b \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & -1400 & 1430000 & -1300000000 \\ 0 & 1 & -1400 & 1430000 \\ 0 & 0 & 1 & -1400 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\det(C) = \det\begin{pmatrix} 1 & -1400 & 1430000 & 1300000000 \\ 0 & 1 & -1400 & 1430000 \\ 0 & 0 & 1 & -1400 \\ 0 & 0 & 0 & 1 \end{pmatrix} = 1$$

Since Det(O) isn't equal to zero, so my system is controllable.

```
K=50;
A=[-1400 -530000 -40000000 -20000*K; 1 0 0 0; 0 1 0 0; 0 0 1 0]
B=[1;0;0;0]
C=ctrb(A,B)
D=det(C)
rref(C)
rank(C)
```

```
A = 4×4
       -1400     -530000   -40000000    -1000000
           1           0           0           0
           0           1           0           0
           0           0           1           0

B = 4×1
     1
     0
     0
     0

C = 4×4
10^9 ×
     0.0000   -0.0000    0.0014   -1.3000
          0    0.0000   -0.0000    0.0014
          0         0    0.0000   -0.0000
          0         0         0    0.0000


D = 1
ans = 4×4
       1      0      0      0
       0      1      0      0
       0      0      1      0
       0      0      0      1

ans = 3
```

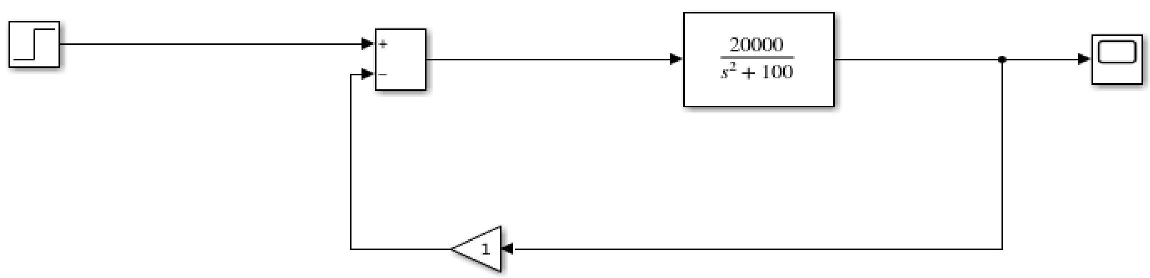
Section#8-Unity Feedback design:

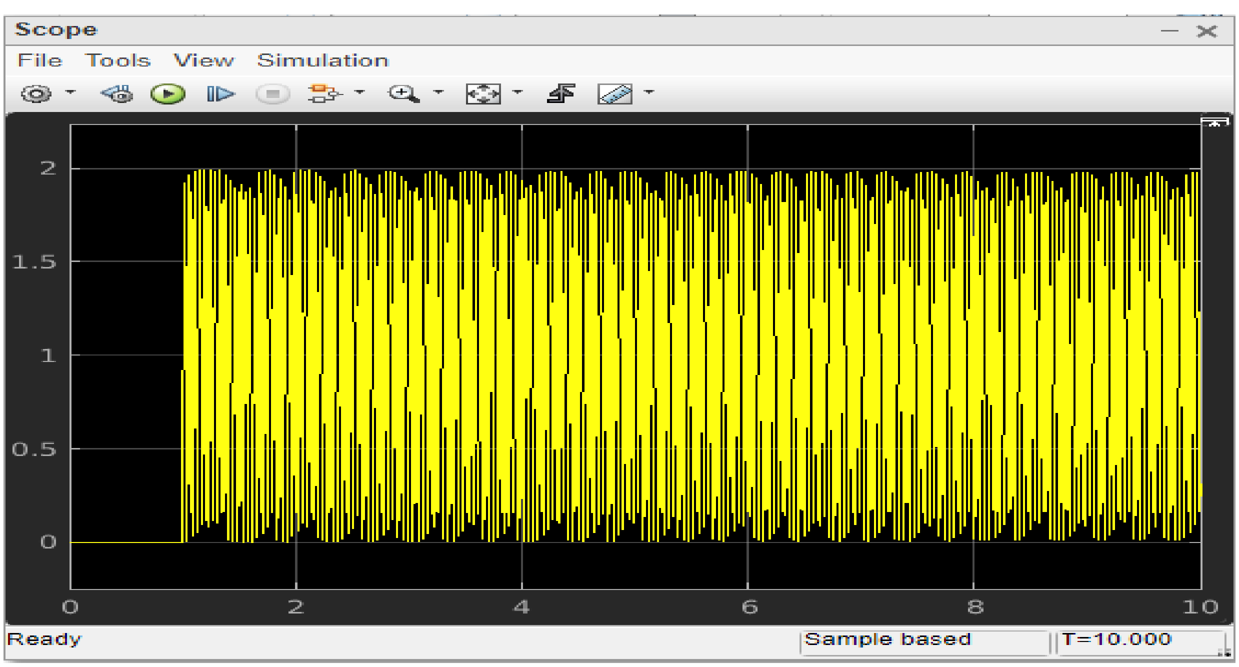
Unity feedback loop is dynamic way to overcome the system parameter changes and a solution to reduce the impact of these changes on the system output. Unity feedback is a closed loop system configuration that has many advantages:

- Opposing the disturbance signals and noises that affect the system output.
- Developing the system performance in the case of model unreliability.
- Helping in stabilize an unstable plant.

I used the following unity feedback design and showed its simulation:

$$TF = \frac{G}{1 + GH}$$





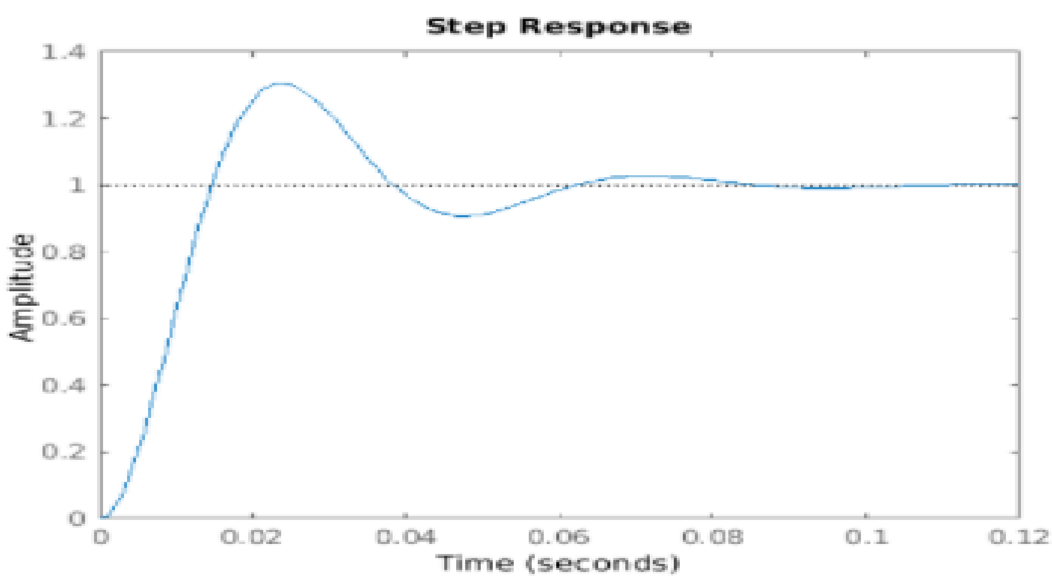
```
a=[20000]
b=[1 100 20000]
sys=tf(a,b)
step(sys)
s=stepinfo(sys)
```

```
a = 20000
b = 1×3
          1         100       20000

sys =

       20000
  ------------------
  s^2 + 100 s + 20000

Continuous-time transfer function.
```




```
s = struct with fields:
        RiseTime: 0.0099
    SettlingTime: 0.0774
     SettlingMin: 0.9071
     SettlingMax: 1.3049
       Overshoot: 30.4890
      Undershoot: 0
            Peak: 1.3049
        PeakTime: 0.0239
```

Section#9-Pole Placement and Modeling design:

Pole placement is a design method that specify the locations of the closed loop system poles to be on the complex plane by setting a controller gain K. The pole placement method is comparable to the root locus method and the difference between two methods can be concluded by using the root-locus design requires to place only the dominant closed loop poles at the desired locations while in the pole placement method all closed-loop poles are placed at desired locations.

I used the following design where the following matrices are plotted in the equation shape:

|SI-A+KC| where K=[K1,K2],

$$A = \begin{pmatrix} -100 & 1 \\ -20000 & 0 \end{pmatrix} \quad B = \begin{pmatrix} 0 \\ 20000 \end{pmatrix} \quad C = \begin{pmatrix} 1 & 0 \end{pmatrix}$$

|SI-A+KC|= $S^2 + (100 + K1)S + 20000 + K2$

$$G(s) = \frac{20000}{S^2 + 100S + 20000}$$

$$TF = \frac{G}{1+GH} = \frac{20000}{S^2 + 100S + 40000}$$

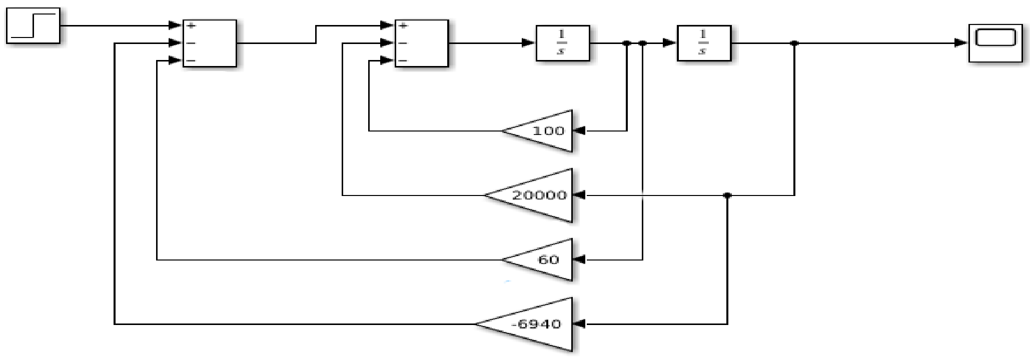
Let settling time=0.05sec, Ɛ=0.7→

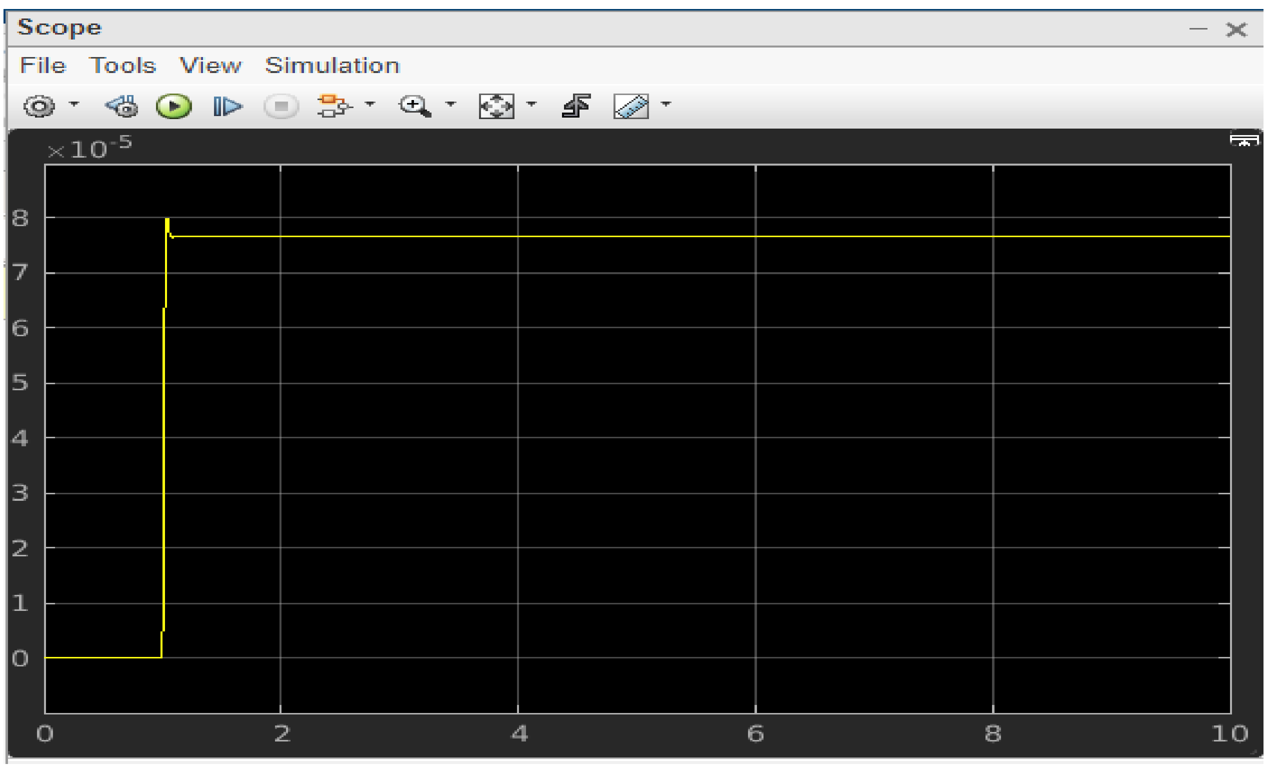
$$t_s = \frac{4}{\zeta\omega_n} \rightarrow \quad Wn = \frac{4}{\xi Ts} = \frac{4}{0.7x0.05} = 114.28 \frac{rad}{sec}$$

Plug the previous value of Wn in the following equation:

$$S^2 + 2\zeta WnS + Wn^2 = S^2 + 160S + 13060 = S^2 + (100 + K1)S + (20000 + K2)$$

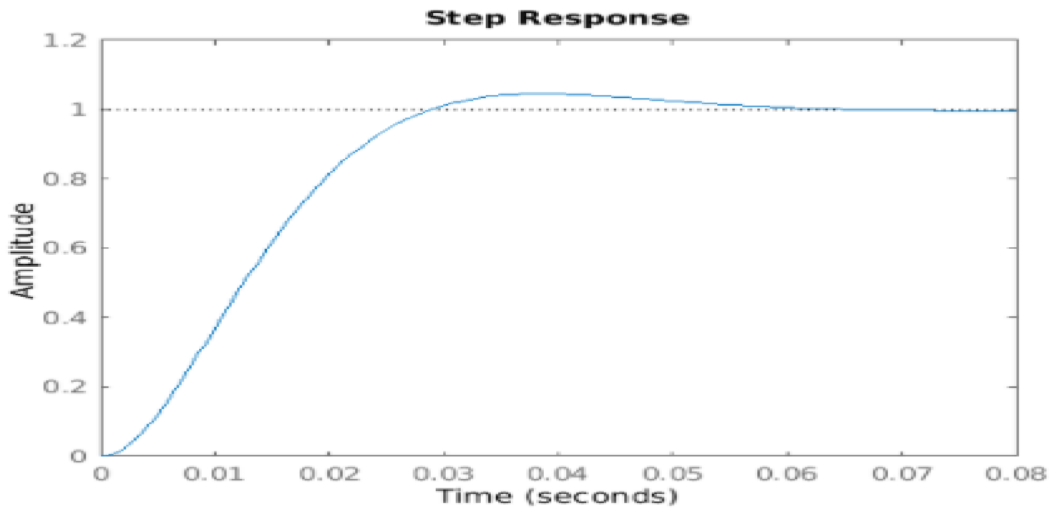
160=100+K1→K1=60. And 13060=20000+K2→K2=-6940.






```
k1=60;
K2=-6940;
a=[20000 + K2]
b=[1 100+k1 20000 + K2]
sys=tf(a,b);
step(sys)
s=stepinfo(sys)
```

```
a = 13060
b = 1×3
          1          160        13060
```




```
s = struct with fields:
        RiseTime: 0.0186
    SettlingTime: 0.0523
     SettlingMin: 0.9001
     SettlingMax: 1.0460
       Overshoot: 4.5973
      Undershoot: 0
            Peak: 1.0460
        PeakTime: 0.0386
```

Section#10-State feedback design:

State feedback in control system is an effective method to reframe the internal behavior of a system. The advantages of feedback in most cases make it as major concept in the design of control system. The state feedback design method is depending on the pole placement method and the quadratic optimal regulator method where we place closed loop poles at desired locations. State feedback is used to obtain the desired pole locations of the closed loop transfer function. State feedback's objective is distributing the roots of the characteristic equation where the transient performance meets the desired response.

Advantages of State Feedback Design:

- Eliminate many disturbances and noise signals from outside the system.
- Reduce the change in the system's performance of the system according to parameter variations.
- Decrease the steady state error of the system to make the response more stable.
- Adjust the behavior of transient signal to be as expected in the system design.
- Comparison between results of feedback and the desired state enable to decide the next steps of corrective actions.

Disadvantages of State Feedback Design:

- Probability of having a complicated system because of using many components, such as sensors and error detectors.
- Probability of decreasing the overall gain of the system requires to be compensated in the design.
- Probability to have unstable system even though the comparable open loop system is stable.
- Probability to have a change in an Output that can affect the system input and needs an error detector to figure out the undesired case.

For my state feedback design, I used the following system parameters:

$$G(s) = \frac{20000}{S^2 + 100S + 20000}$$

$$TF = \frac{G}{1+GH} \quad = \frac{20000}{S^2 + 100S + 40000}$$

$$A = \begin{pmatrix} -100 & 1 \\ 20000 & 0 \end{pmatrix}, B = \begin{pmatrix} 0 \\ 20000 \end{pmatrix}, C = \begin{pmatrix} 1 & 0 \end{pmatrix}, G = \begin{pmatrix} G1 & G2 \end{pmatrix}$$

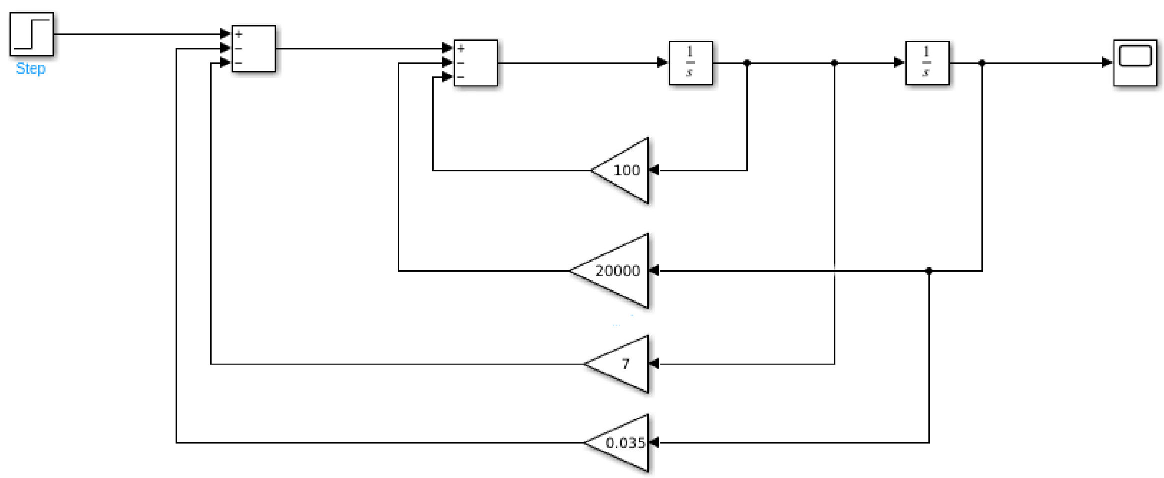
|SI-A+BG|= $S^2 + (20000G2 + 100)S + (20000G1 + 20000)$

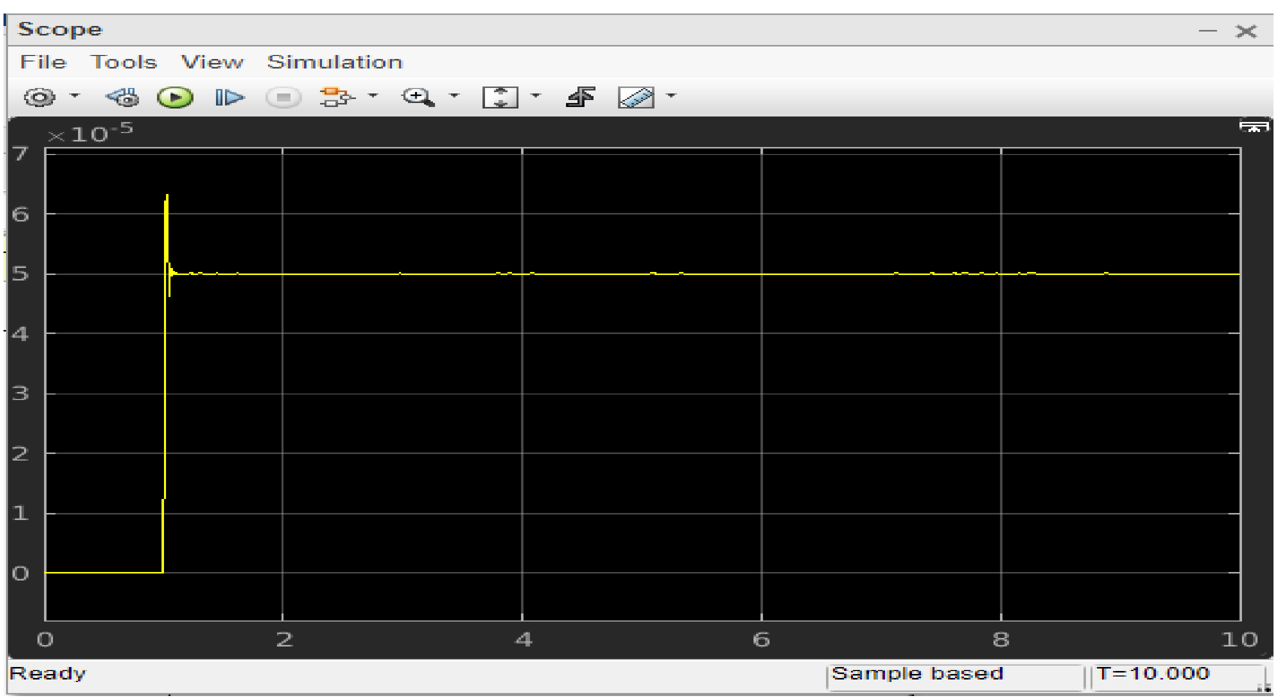
Let settling time=0.01sec, Ɛ=1→ $t_s = \frac{4}{\zeta\omega_n}$ → $Wn = \frac{4}{Tsx\zeta} = \frac{4}{0.01} = 400\ \frac{rad}{sec}$

Plug the previous value of Wn in the following equation:

$$S^2 + 2\zeta WnS + Wn^2 = S^2 + 800S + 160000 \quad = S^2 + (20000G2 + 100)S + (20000G1 + 20000)$$

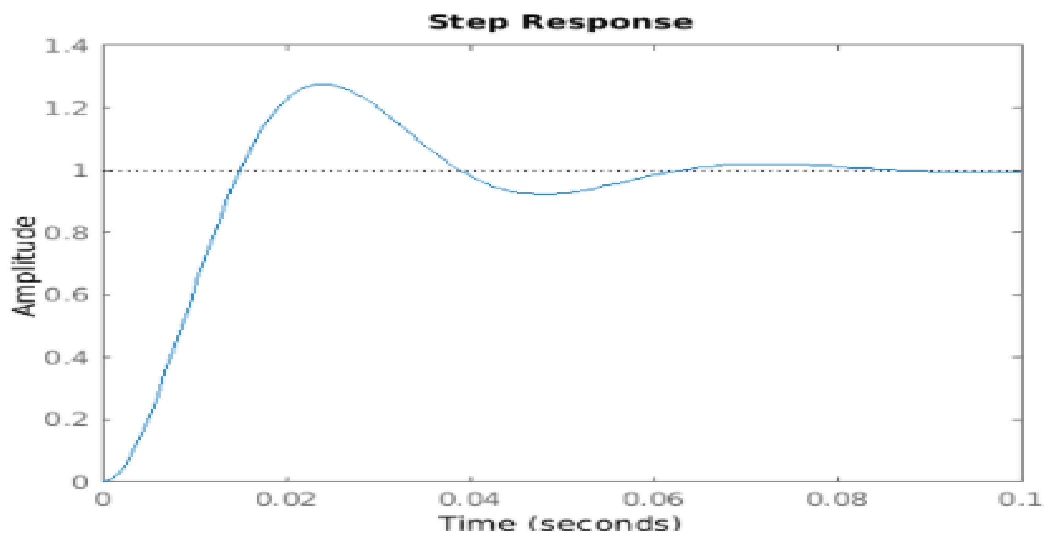
800=20000G2+100→G2=0.035 and 160000. And 160000=20000G1+20000→G1=7.





```
G1=7;
G2=0.035;
a=[20000+G2]
b=[1 100+G1 20000+G2]
sys=tf(a,b);
step(sys)
s=stepinfo(sys)
```

```
a = 2.0000e+04
b = 1×3
10^4 ×
    0.0001    0.0107    2.0000
```




```
s = struct with fields:
        RiseTime: 0.0101
    SettlingTime: 0.0745
     SettlingMin: 0.9198
     SettlingMax: 1.2769
       Overshoot: 27.6929
      Undershoot: 0
            Peak: 1.2769
        PeakTime: 0.0241
```

Section#11-Qualitative Discussion:

In this section, I provided a comparison among K-value design, unity feedback design, pole placement design, and state feedback design that showed the pros and cons for each one to achieve an accurate configuration that offer a robust system.

System Design	Overshoot	Settling Time	Rise Time	Peak Time
K-Value	74.3	0.3317	0.0081	0.0246
Unity Feedback	30.5	0.08	0.009	0.024
Pole Placement	4.6	0.05	0.02	0.04
State Feedback	27.7	0.07	0.01	0.04

The conclusion of my project prove that in engineering there is no design is optimal because each design has some advantages and disadvantages, so if the system requires less overshoot then the good choice will be pole placement design or if the system seeks less settling time then pole placement design, state feedback design, and unity feedback design have almost the same values with little differences. Moreover, the best values of rise time can be obtained from either K-value design or unity feedback design.

Section#12-References:

- Ogata, K. (2016). *Modern control engineering*. Delhi, India: Pearson.
- Manke, B. S. (2003). *Linear control systems: A textbook for engineering students*. Delhi, India: Khanna.
- Chen, C. (2009). *Linear System Theory and Design: International third edition*. New York, New York: Oxford University Press.
- Kailath, T. (1980). *Linear systems*. Englewood Cliffs, NJ, NJ: Prentice-Hall.
- Bhavesh, Bhagyarekha, & Mark. (2011, November 19). Controllability and observability, kalman's test, Gilberts Test, Jordan canonical form. Retrieved April 26, 2021, from https://www.circuitstoday.com/controllability-and-observability
- Lecture Notes, ECE-560, Winter 2021.