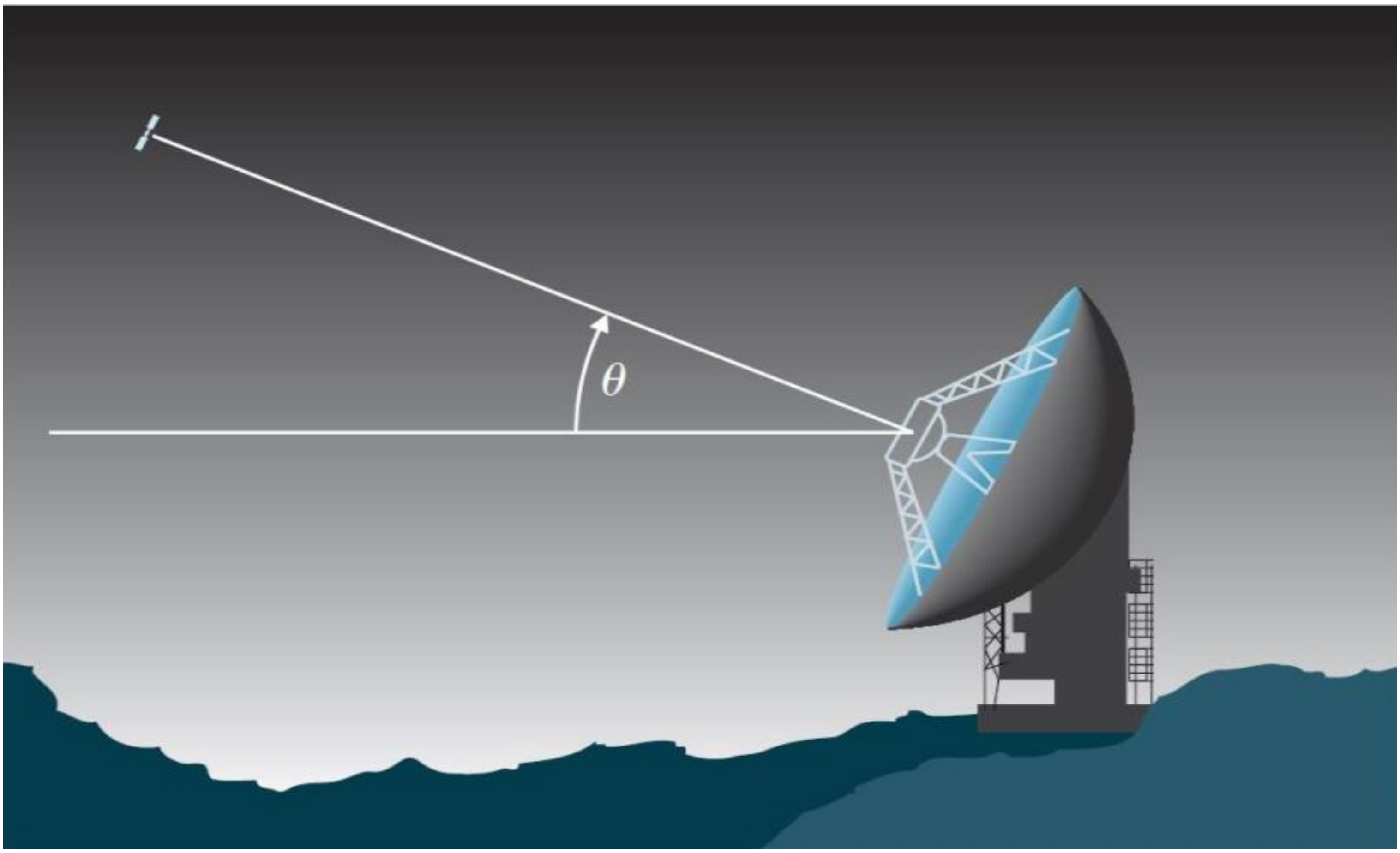
Control



Lab Assignment 02: Design of Satellite-Tracking Antenna

a) Evaluate the closed loop transfer function:

- transfer function $\dfrac{\theta(s)}{\theta r(s)} = \dfrac{\frac{K}{J}}{S^2+\frac{B}{J}S+\frac{K}{J}} = \dfrac{\frac{K}{600{,}000}}{S^2+\frac{1}{30}S+\frac{K}{600{,}000}}$

Assignment_2 Control

$T_c = J\ddot{\theta} + B\dot{\theta}$ & $T_c = K(\theta_r - \theta)$

$\therefore K\theta_r - K\theta = J\ddot{\theta} + B\dot{\theta}$ $\mapsto$ Taking laplace

$\therefore K\theta_r(s) = [J s^2 + Bs + K]\,\theta(s)$

$\therefore TF = \dfrac{\theta(s)}{\theta_r(s)} = \dfrac{K}{J s^2 + B s + K} = \dfrac{\frac{K}{J}}{s^2 + \frac{B}{J}s + \frac{K}{J}}$

& we know that second order has $TF = \dfrac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$

$\therefore \boxed{\omega_n = \sqrt{\frac{K}{J}}}$ & $\boxed{\zeta = \dfrac{\frac{B}{J}}{2\sqrt{\frac{K}{J}}} = \dfrac{B}{2\sqrt{KJ}}}$

$\therefore$ (a) $\boxed{TF = \dfrac{K/600*10^3}{s^2 + \frac{1}{30}s + \frac{K}{600*10^3}}}$ $\rightarrow$ closed loop TF

b) Use MATLAB to generate the state-space representation for the closed-loop sys for K = 1

$$\frac{dx}{dt} = A\,x(t) + B\,\theta r(t) \;\&\; \theta(t) = C\,x(t) + D\,\theta r(t)$$

Using ss(TF) function I can get A &B &C &D from transfer function

```
J = 600000; % kg·m2
B = 20000;  % N·m·sec ==  kg·m2 /sec
```

a)TF = K/(Js^2 + Bs + K)

```
% doc tf
K =1;  % feedback gain
TF = tf(K,[J ,B , K]);
TFOr = tf(K/J,[1 ,B/J , K/J]); %Equivient
```

b)the state-space representation for the closed loop system for K = 1

```
% doc ss
sys_TF = ss(TF)
% dx/dt=A x(t)+B θr(t)
% θ(t)=C x(t)+D θr(t)
```

c)The maximum value of K to have a stable closed loop system:

```
% doc isstable
```

```
sys_TF =

  A =
              x1
   x1   -0.03333
   x2  0.0009766

              x2
   x1  -0.001707
   x2          0

  B =
            u1
   x1  0.03125
   x2        0

  C =
            x1
   y1        0

            x2
   y1  0.05461

  D =
       u1
   y1   0

Continuous-time state-space model.
```

a) Evaluate the closed loop transfer function:

- transfer function $\frac{\theta(s)}{\theta r(s)} = \frac{\frac{K}{J}}{S^2+\frac{B}{J}S+\frac{K}{J}} = \frac{\frac{K}{600{,}000}}{S^2+\frac{1}{30}S+\frac{K}{600{,}000}}$

Assignment_2 Control

$T_c = J\ddot{\theta} + B\dot{\theta}$ & $T_c = K(\theta_r - \theta)$

$\therefore K\theta_r - K\theta = J\ddot{\theta} + B\dot{\theta}$ $\mapsto$ Taking laplace

$\therefore K\theta_r(s) = [J s^2 + B s + K]\,\theta(s)$

$\therefore TF = \frac{\theta(s)}{\theta_r(s)} = \frac{K}{J s^2 + B s + K} = \frac{\frac{K}{J}}{s^2 + \frac{B}{J} s + \frac{K}{J}}$

& we know that second order has $TF = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$

$\therefore \boxed{\omega_n = \sqrt{\frac{K}{J}}}$ & $\boxed{\zeta = \frac{\frac{B}{J}}{2\sqrt{\frac{K}{J}}} = \frac{B}{2\sqrt{KJ}}}$

$\therefore$ ⓐ $\boxed{TF = \frac{K/600\times10^3}{s^2 + \frac{1}{30}s + \frac{K}{600\times10^3}}}$ $\rightarrow$ closed loop TF

b) Use MATLAB to generate the state-space representation for the closed-loop sys for K = 1

$$\frac{dx}{dt} = A\,x(t) + B\,\theta r(t)\ \&\ \theta(t) = C\,x(t) + D\,\theta r(t)$$

Using ss(TF) function I can get A &B &C &D from transfer function

```
J = 600000; % kg·m2
B = 20000;  % N·m·sec ==  kg·m2 /sec
```

a)TF = K/(Js^2 + Bs + K)

```
% doc tf
K =1;  % feedback gain
TF = tf(K,[J ,B , K]);
TFOr = tf(K/J,[1 ,B/J , K/J]); %Equivient
```

b)the state-space representation for the closed loop system for K = 1

```
% doc ss
sys_TF = ss(TF)
% dx/dt=A x(t)+B θr(t)
% θ(t)=C x(t)+D θr(t)
```

c)The maximum value of K to have a stable closed loop system:

```
% doc isstable
```

```
sys_TF =

  A =
              x1
   x1   -0.03333
   x2  0.0009766

              x2
   x1  -0.001707
   x2          0

  B =
            u1
   x1  0.03125
   x2        0

  C =
            x1
   y1        0

            x2
   y1  0.05461

  D =
       u1
   y1   0

Continuous-time state-space model.
```

c) What is the maximum value of K that can be used if you wish to have a stable closed loop system?

The max value is K is infinity this is obtained using Routh-Hurwitz criteria and I will prove it using MATLAB

ⓒ To find Max value I will use Routh Herwitz

$$\begin{array}{c|cc} s^2 & 1 & \frac{K}{J} \\ s & \frac{B}{J} & \\ s^0 & \frac{K}{J} & \end{array}$$

∴ K must be more Than zero

$K > 0$ to be stable

So minimum $K = 0$ but Max is $\boxed{\infty}$

and I prove it by Sumilation.

Prove using MATLAB

c)The maximum value of K to have a stable closed loop system:

```
% doc isstable
K_max =inf;  % feedback gain
TF_max = tf(K_max,[J ,B , K_max]);
stability = isstable(TF_max)
```

```
stability = logical
   1
```

d) What is the maximum value of K that can be used if you wish to have an overshoot Mp<10%?

From maximum overshoot, I can get a damping ratio(ξ) because Mp depends only on zeta (ξ):
And I will prove it using MATLAB

K = 476.9205

ⓓ To find Max value of $\boxed{K}$ @ $M_p < 10\%$.

$M_p = e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}} * 100 = 10$ ∴ $-\zeta\pi = -\ln 0.1 * \sqrt{1-\zeta^2}$

∴ $\zeta = \sqrt{\frac{(-\frac{\ln 0.1}{\pi})^2}{1+(-\frac{\ln 0.1}{\pi})^2}} = 0.5912$ → damping ratio

& $\zeta = \frac{B}{2\sqrt{KJ}}$ ∴ $K = \frac{B^2}{4J\zeta^2} = \frac{20^2 * 10^6}{4 * 10^3 * 600 * 0.5912^2} = 476.92$

∴ $K_{@M_p=10\%} = \cancel{476.8205} = 476.9205$ ~ بدون تقريب

Prove using MATLAB

d)The maximum value of K if you wish to have an overshoot Mp < 10%?

```
% doc stepinfo
tempVar = log(0.1)/pi;
zeta = sqrt(tempVar^2/(1+tempVar^2));
K_Mp_10 = B^2/(4*J*zeta^2) %=476.921 % feedback gain
TF_Mp_10 = tf(K_Mp_10,[J ,B , K_Mp_10]);
sysprop = stepinfo(TF_Mp_10,'RiseTimeThreshold',[0 1]);
sysprop.Overshoot
```

```
K_Mp_10 = 476.9205

ans = 10.0000
```

e) What values of K will provide a rise time of less than 80 sec? (Ignore the Mp constraint.)

Raise time(Tr) = $\frac{\pi - \theta}{Wd}$ (this in case of the rise time from 0 to 100 %). where $\boldsymbol{\theta} = \cos^{-1}\xi$ && **Wd = Wn** $\times \sqrt{1-\xi^2}$ && $\xi = \frac{B}{2\sqrt{KJ}}$

Wn = $\sqrt{\frac{K}{J}}$**. It is difficult to solve exactly so I will use it numerically using MATLAB**

e) $K = ?$ @ $Tr = 80$ ⇝ $Tr = \frac{\pi - \theta}{W_d}$ & $\theta = \cos^{-1}\xi$

$\therefore Tr = \dfrac{\pi - \cos^{-1}\left(\frac{B}{2\sqrt{KJ}}\right)}{\sqrt{\frac{K}{J}}\cdot\sqrt{1 - \frac{B^2}{4KJ}}} = 80 \mapsto$ By nemerical solve

e) we suppose that rise time from 0 → 100%.
so $K = 592.0336$ numerically

Solve using MATLAB

e)The values of K that provides a rise time less than 80 sec.

```
% doc stepinfo
syms K_Tr % solve it numerically
zeta_tr = B/(2*sqrt(K_Tr*J));
Wn = sqrt(K_Tr/J);
Wd = Wn *sqrt(1-zeta_tr^2);
eqn = (pi-acos(zeta_tr))/Wd == 80;
K_tr_80 = double(vpasolve(eqn,K_Tr))
TF_tr_80 = tf(K_tr_80,[J ,B , K_tr_80]);
sysprop_tr = stepinfo(TF_tr_80,'RiseTimeThreshold',[0 1]);
sysprop_tr.RiseTime
```

```
K_tr_80 = 592.0336

ans = 80.0055
```

f) Use MATLAB to plot the step response of the antenna system for K = 200, 400, 1000, and 2000. Find the overshoot and rise time of the four step responses by examining your plots. Do the plots to confirm your calculations in previous parts?

I will use stepplot() and stepinfo() functions to plot and get maximum overshoot and the raise time

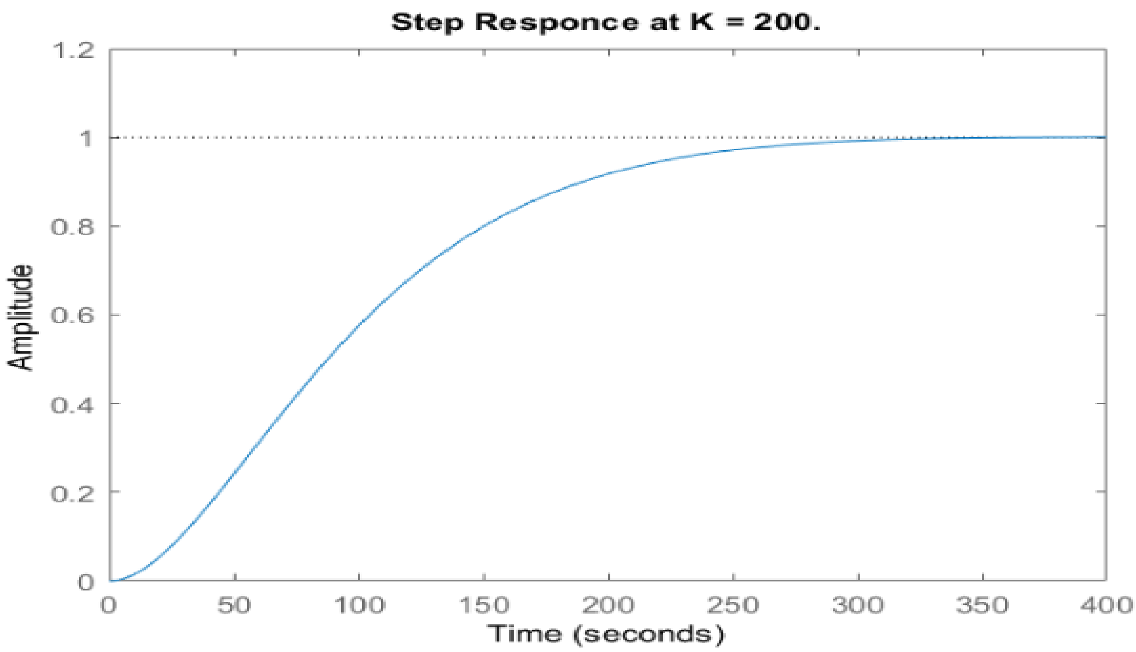
f)Plot the step response of the antenna system for K = 200, 400, 1000, and 2000

```
K_toplot = [200 400 1000 2000];
for i=1:length(K_toplot)
    TF_temp = tf(K_toplot(i),[J ,B , K_toplot(i)]);
    sysprop_temp = stepinfo(TF_temp,'RiseTimeThreshold',[0 1]);
    Mp = sysprop_temp.Overshoot;
    Tr = sysprop_temp.RiseTime;
    figure();
    stepplot(TF_temp);
    title("Step Responce at K = "+num2str(K_toplot(i))+".")
    fprintf("\nAt K = %f\nThe max overshoot = %f\nThe rise time = %f\n",K_toplot(i),Mp,Tr);
end
```

1) The step-response of the antenna system at K = 200

$\xi = \dfrac{B}{2\sqrt{KJ}}$ **= 0.91287 so it is approximately a critically damping**

Mp =0.088930% && Tr =365.082228 sec

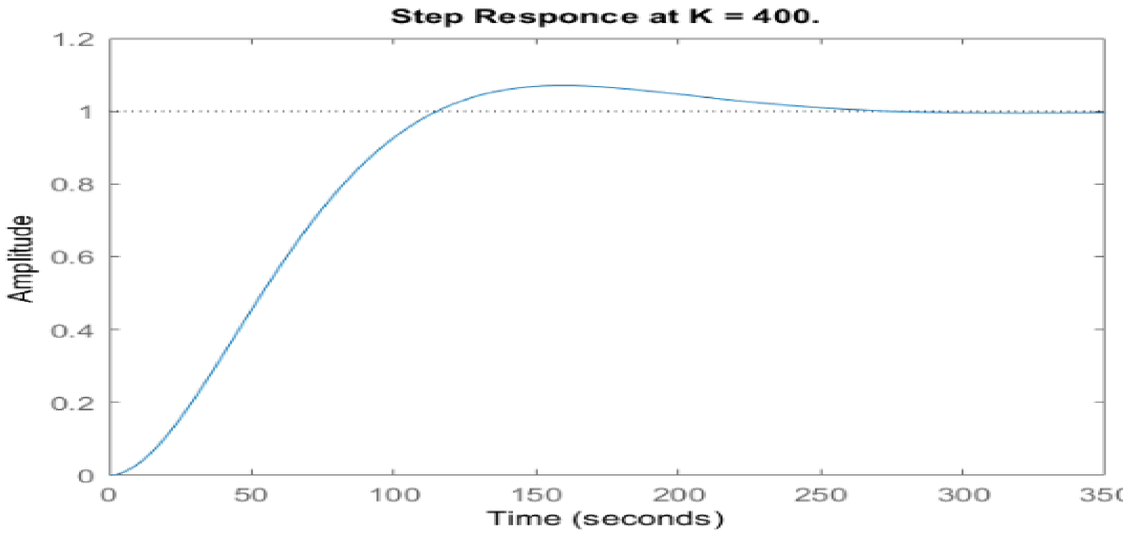



```
At K = 200.000000
The max overshoot = 0.088930
The rise time = 365.082228
```

2) The step-response of the antenna system at K = 400

$\xi = \frac{B}{2\sqrt{KJ}}$ **= 0.6455 so it is an under damping**

Mp =7.026866 % && Tr = 115.261380 sec

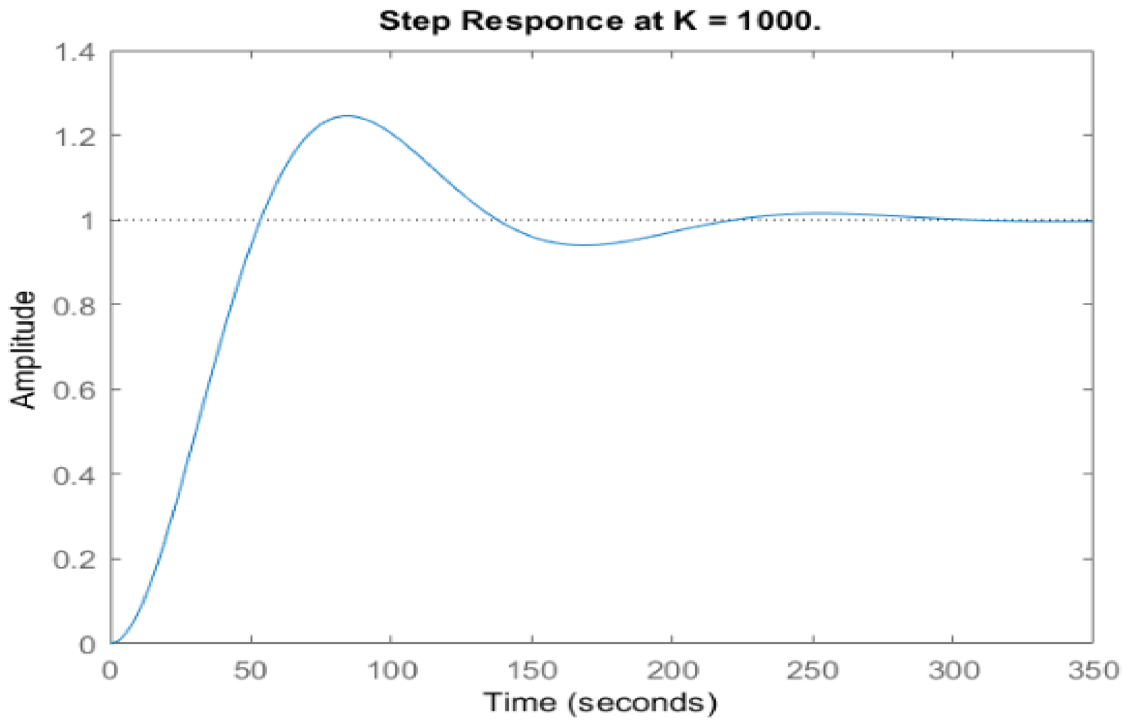



```
At K = 400.000000
The max overshoot = 7.026866
The rise time = 115.261380
```

3) The step-response of the antenna system at K=1000

$\xi = \frac{B}{2\sqrt{KJ}}$ **= 0.40825 so it is an under damping**

Mp =24.5 % && Tr = 53.462051 sec

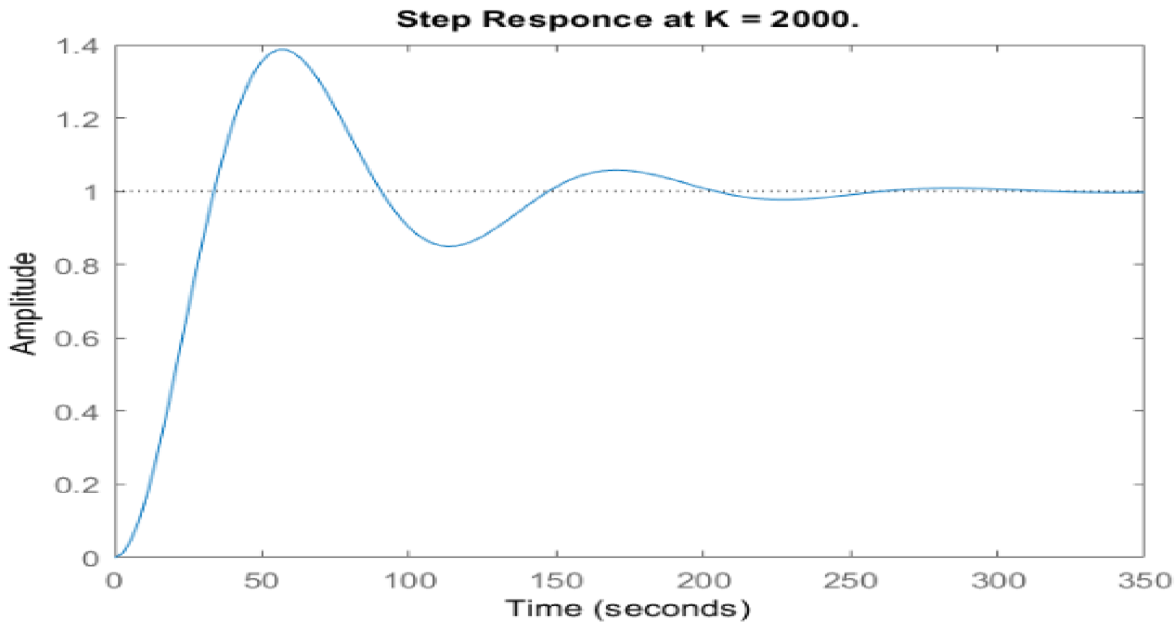



```
At K = 1000.000000
The max overshoot = 24.500456
The rise time = 53.462051
```

4) The step-response of the antenna system at K=2000

$\xi = \frac{B}{2\sqrt{KJ}} = 0.288675$ **so it is an under damping**

Mp =38.69 % && Tr = 33.736331 sec




```
At K = 2000.000000
The max overshoot = 38.691002
The rise time = 33.736331
```

Observation:

As K increases the damping ratio decreases, the maximum overshoot increases and the raising time decreases.
We can see that our calculations in the previous parts is true due to that **Mp= 7% at k = 400** and **Mp= 24,5% at k = 1000**
So at Mp less than 10% (400< k <1000) which is 476.9205
Tr= 115.26 sec at K = 400 and **Tr= 53.46 sec at K = 400**
So at Tr less than 80 sec (400< k <1000) which is 592.0336

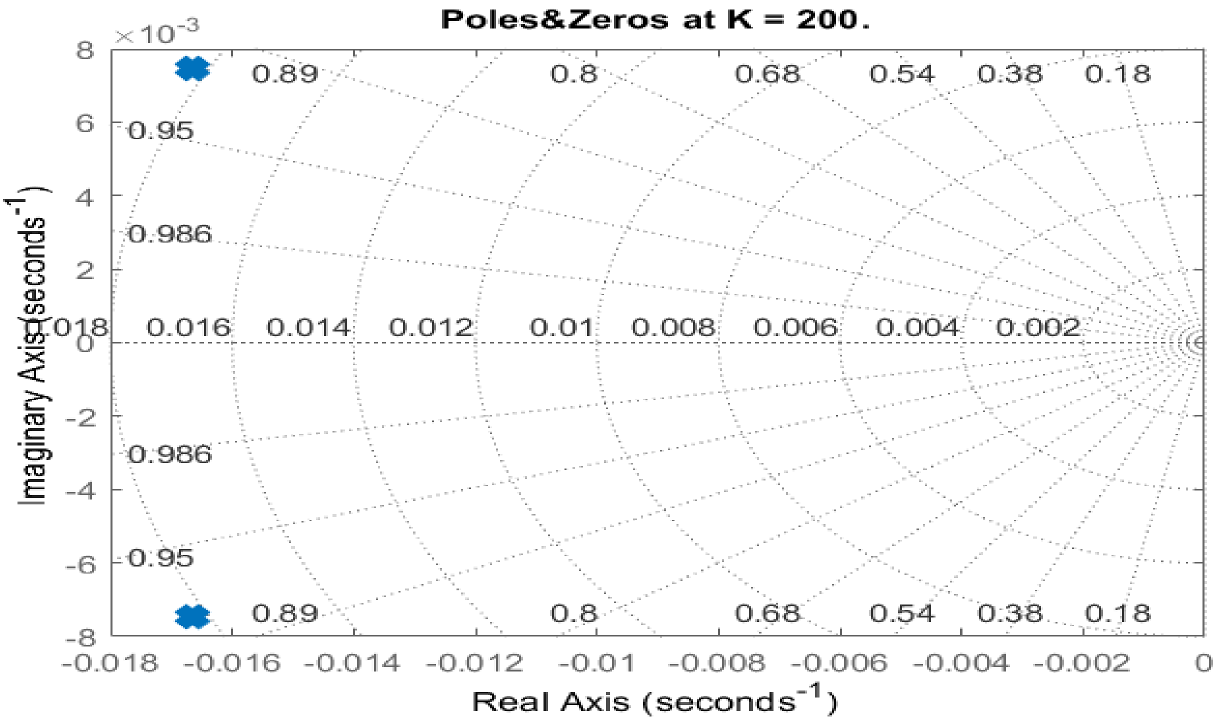
g) Use MATLAB to plot the zeros and poles locations for each value of K in part (e). comment on the effect of K on the closed loop zeros and poles

g. Plot the zeros and poles locations for each value of K.

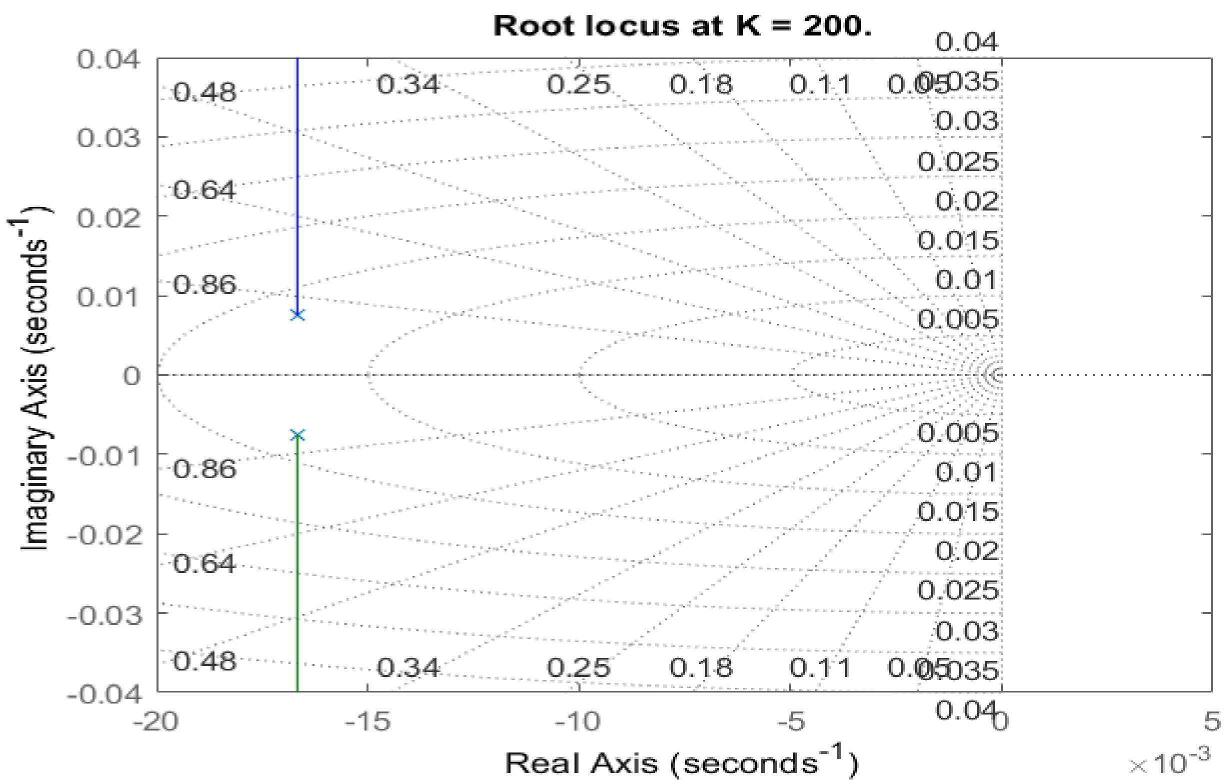
```
K_toplot = [200 400 1000 2000];
for i=1:length(K_toplot)
    TF_temp = tf(K_toplot(i),[J ,B , K_toplot(i)]);
    figure();
    pzplot(TF_temp);
    title("Poles&Zeros at K = "+num2str(K_toplot(i))+".")
    a=findobj(gca,'type','line');
    set(a(2),'linewidth',5,'markersize',10);
    set(a(3),'linewidth',5,'markersize',10);
    grid
end
```

Note: there are zeros at infinity and to show them I will plot the root locus

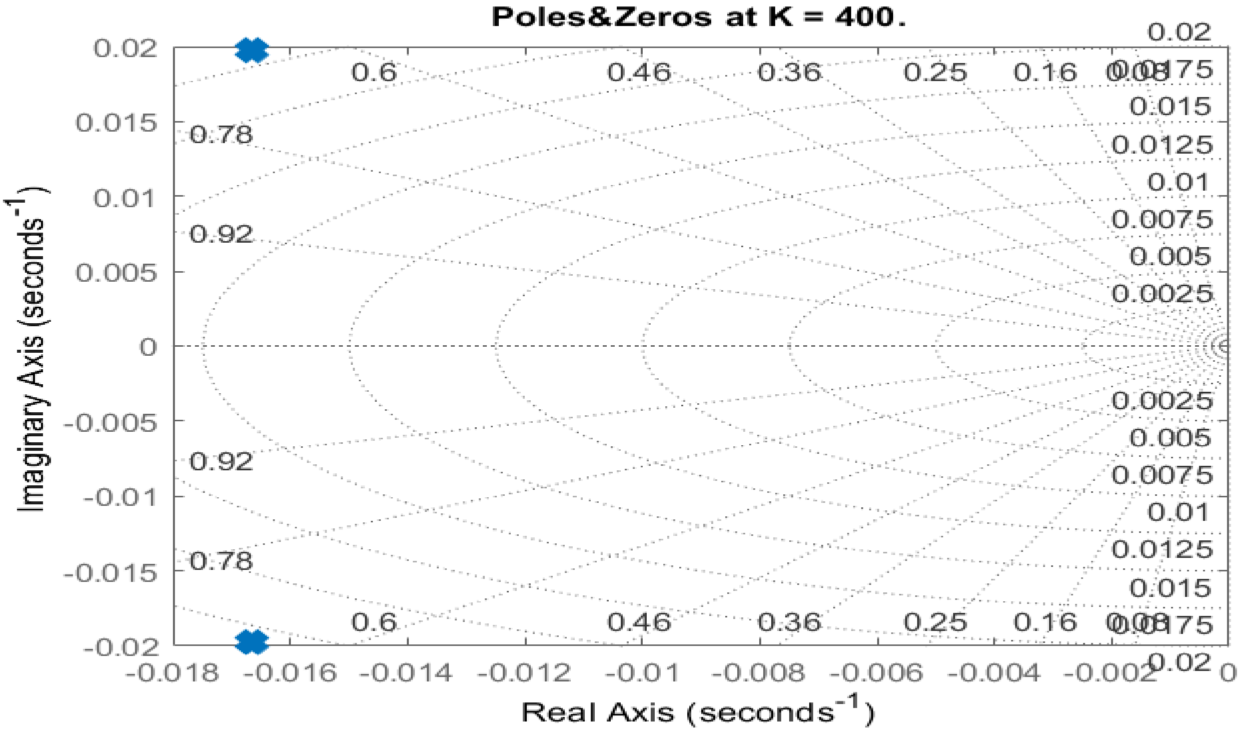
1) The zeros and poles locations at $K = 200$



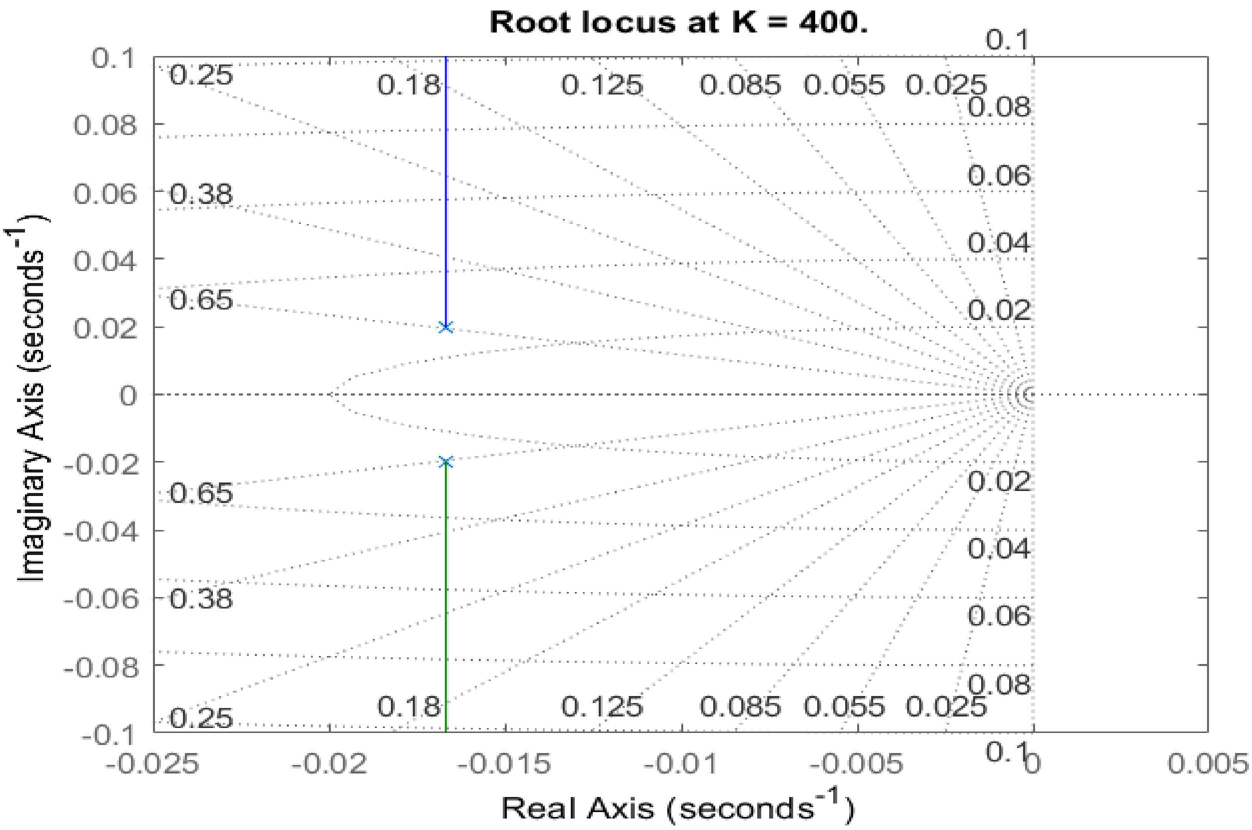
Root Locus



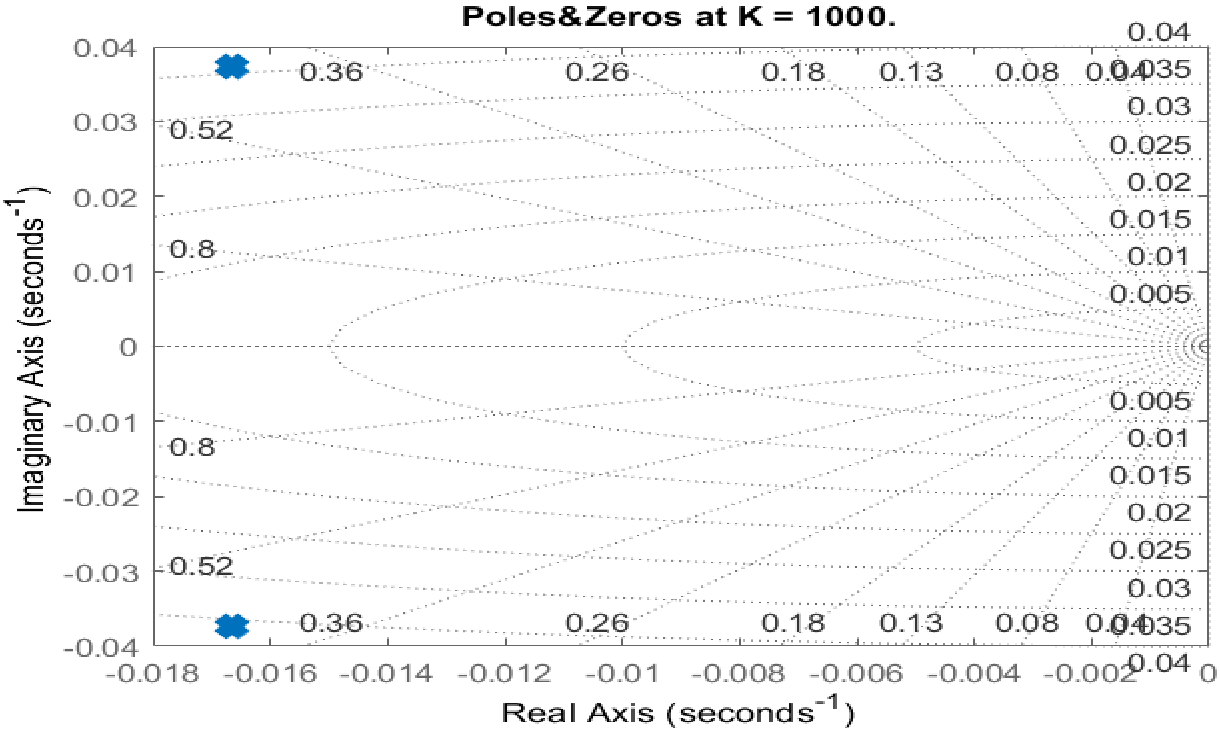
2)The zeros and poles locations at K = 400



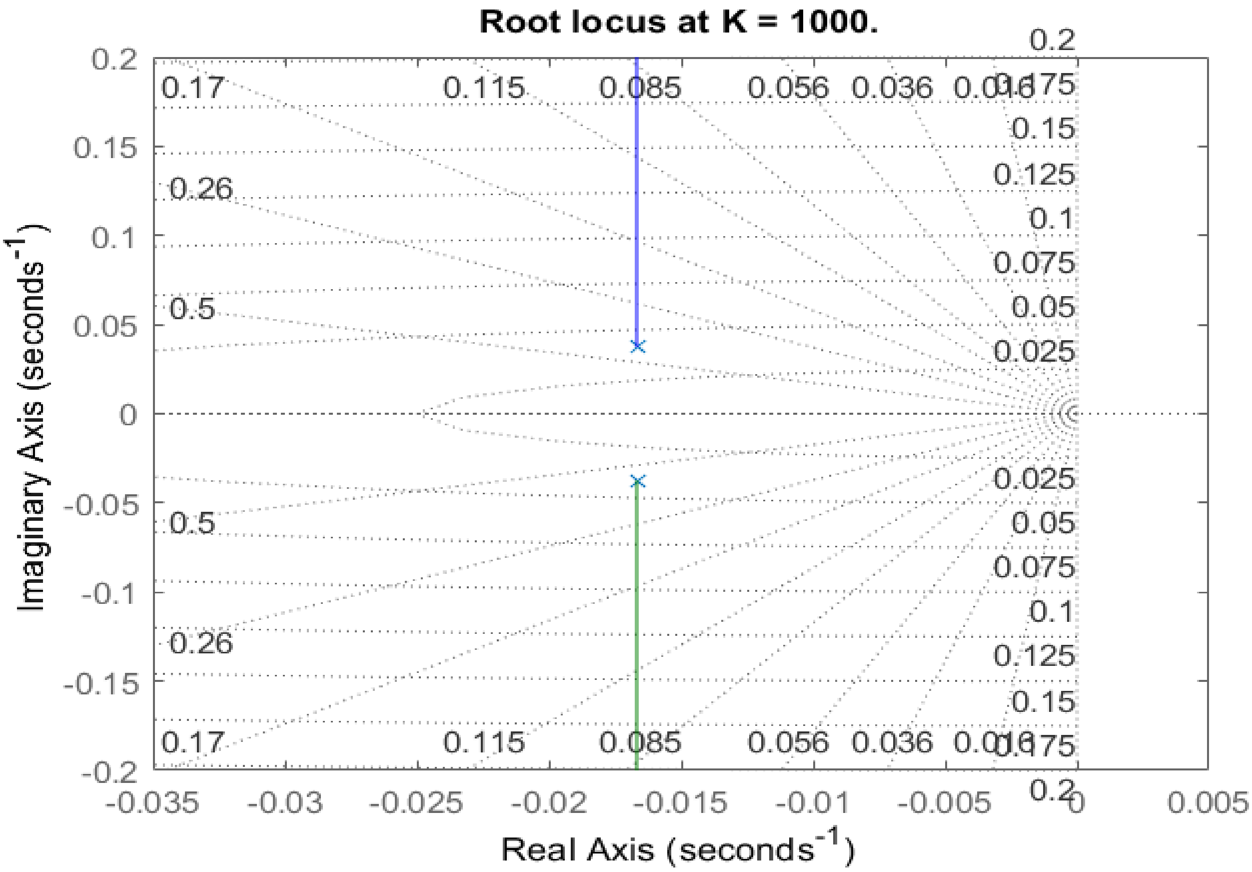
Root Locus



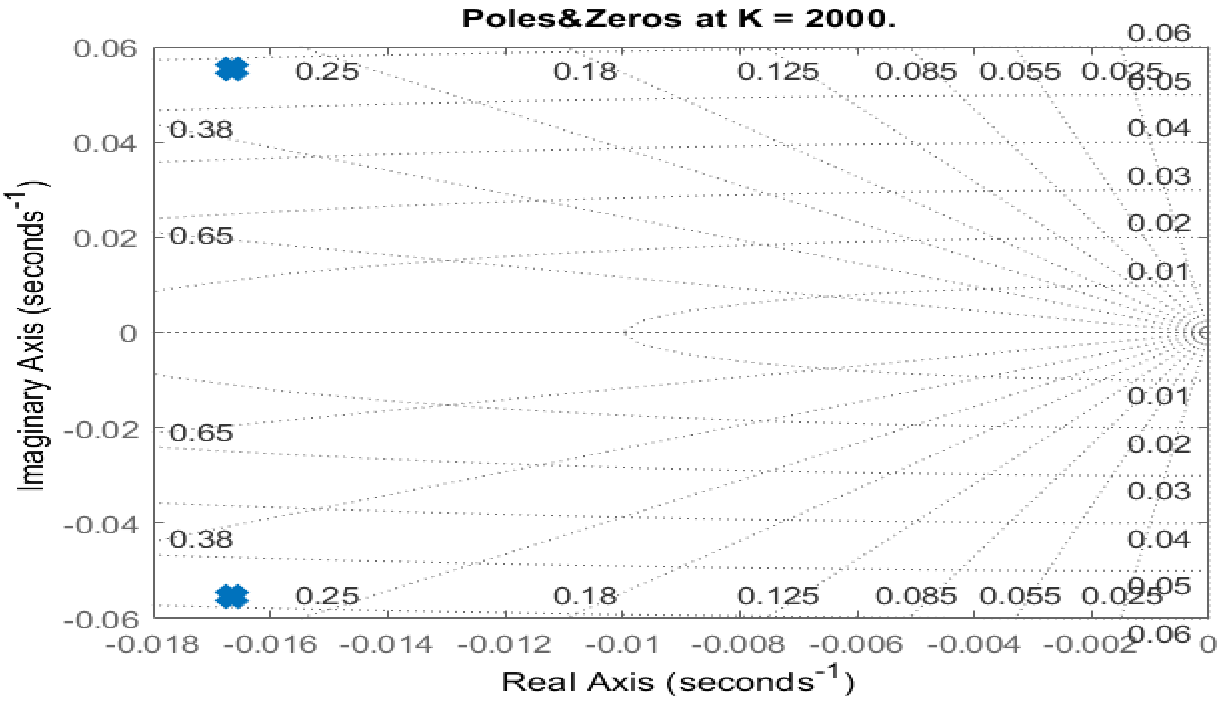
3)The zeros and poles locations at $K = 1000$



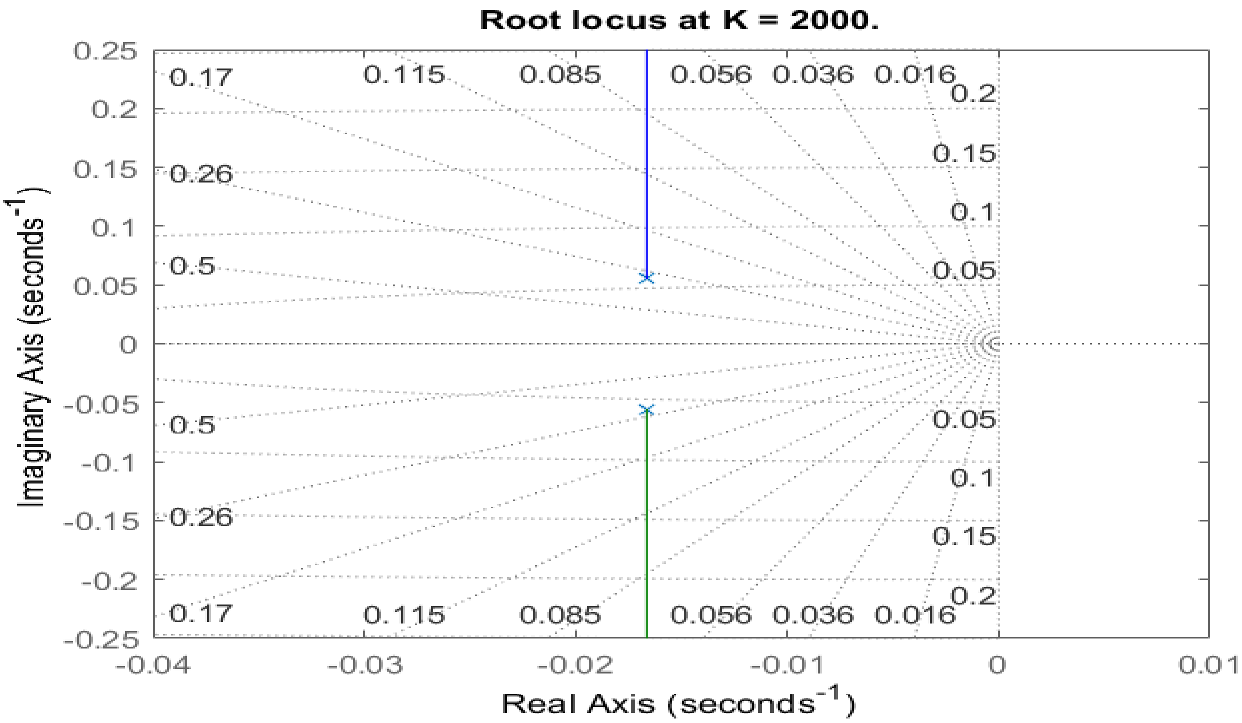
Root Locus



4)The zeros and poles locations at $K = 2000$



Root Locus



Comment

In all cases zeros at infinity but as K increases the poles go deeper to the –ve so the relative stability increases

h) For each value of K in part (e). find the steady-state error.

Here I am so confused so I will assume two cases:

Case 1 It is an unity feedback so G(s) = $\frac{K}{JS^2+B\,S}$ then I can compute Kp & Kv &Ka then the steady-state error.

```
syms S
K_toplot = [200 400 1000 2000];
for i=1:length(K_toplot)
    G_s = K_toplot(i)/(J*S^2 + B*S);
    Kp = limit(G_s,S,0)    % Position constant, Kp
    Kv = limit(S*G_s,S,0) % Velocity constant, Kv
    Ka = limit(S^2*G_s,S,0)% Acceleration constant, Ka
    fprintf("The steady-state error at K = %d",K_toplot(i))
    steadyStateError_P = 1/(1+Kp)
    steadyStateError_V = 1/Kv
    steadyStateError_A = 1/Ka
end
```

Case 2 I will calculate it from the step response and that means that I deal with it as Kv because the input is the unit step

h)for each value of K in part (e). find the steady state error

```
K_toplot = [200 400 1000 2000];
for i=1:length(K_toplot)
    TF_temp = tf(K_toplot(i),[J ,B , K_toplot(i)]);
    [y,t]=step(TF_temp); %get the response of the system to a step with amplitude S
    sserror=abs(1-y(end)) %get the steady state error
end
```

1)Steady-state error at K = 200

Kp = NaN

Kv =

$\frac{1}{100}$

Ka = 0

The steady-state error at K = 200

steadyStateError_P = NaN

steadyStateError_V = 100

steadyStateError_A = ∞

At case 2: steady-state error = 5.14 $\times$ 10^{-4}

2)Steady-state error at K = 400

Kp = NaN

Kv =

$\frac{1}{50}$

Ka = 0

The steady-state error at K = 400

steadyStateError_P = NaN

steadyStateError_V = 50

steadyStateError_A = ∞

At case 2: steady-state error = 0.0049

3)Steady-state error at K = 1000

Kp = NaN

Kv =

$\frac{1}{20}$

Ka = 0

The steady-state error at K = 1000

steadyStateError_P = NaN

steadyStateError_V = 20

steadyStateError_A = ∞

At case 2: steady-state error = 0.0033

4)Steady-state error at K = 2000

Kp = NaN

Kv =

$\frac{1}{10}$

Ka = 0

The steady-state error at K = 2000

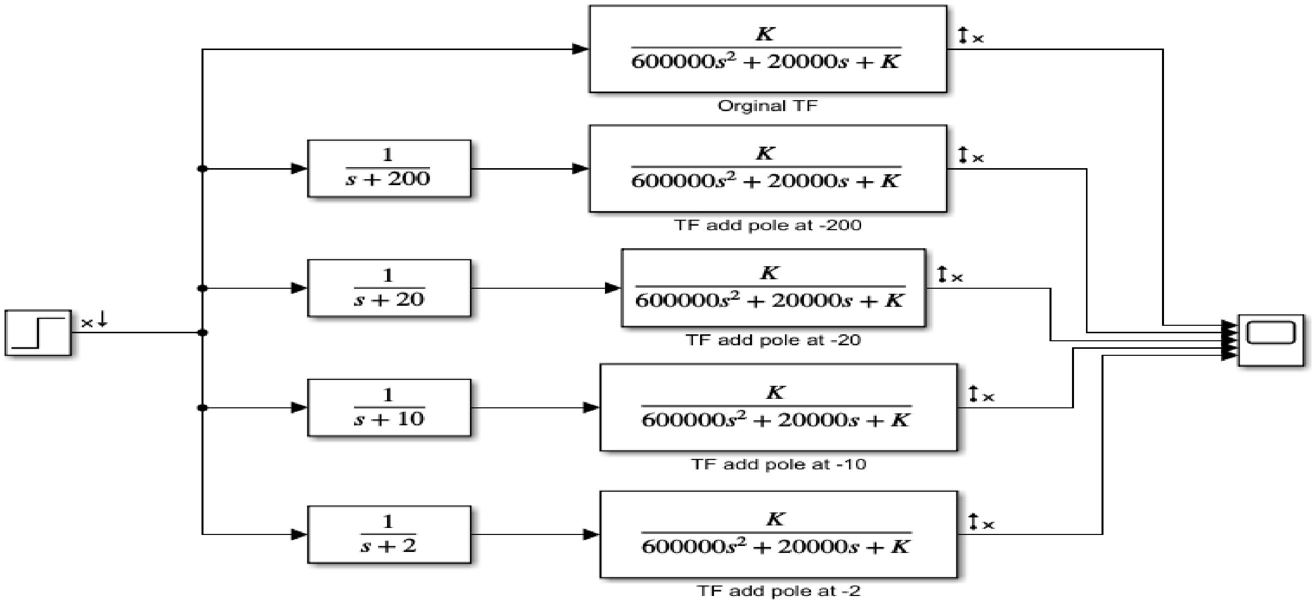
steadyStateError_P = NaN

steadyStateError_V = 10

steadyStateError_A = ∞

At case 2: steady-state error = 8.7153×10^{-4}

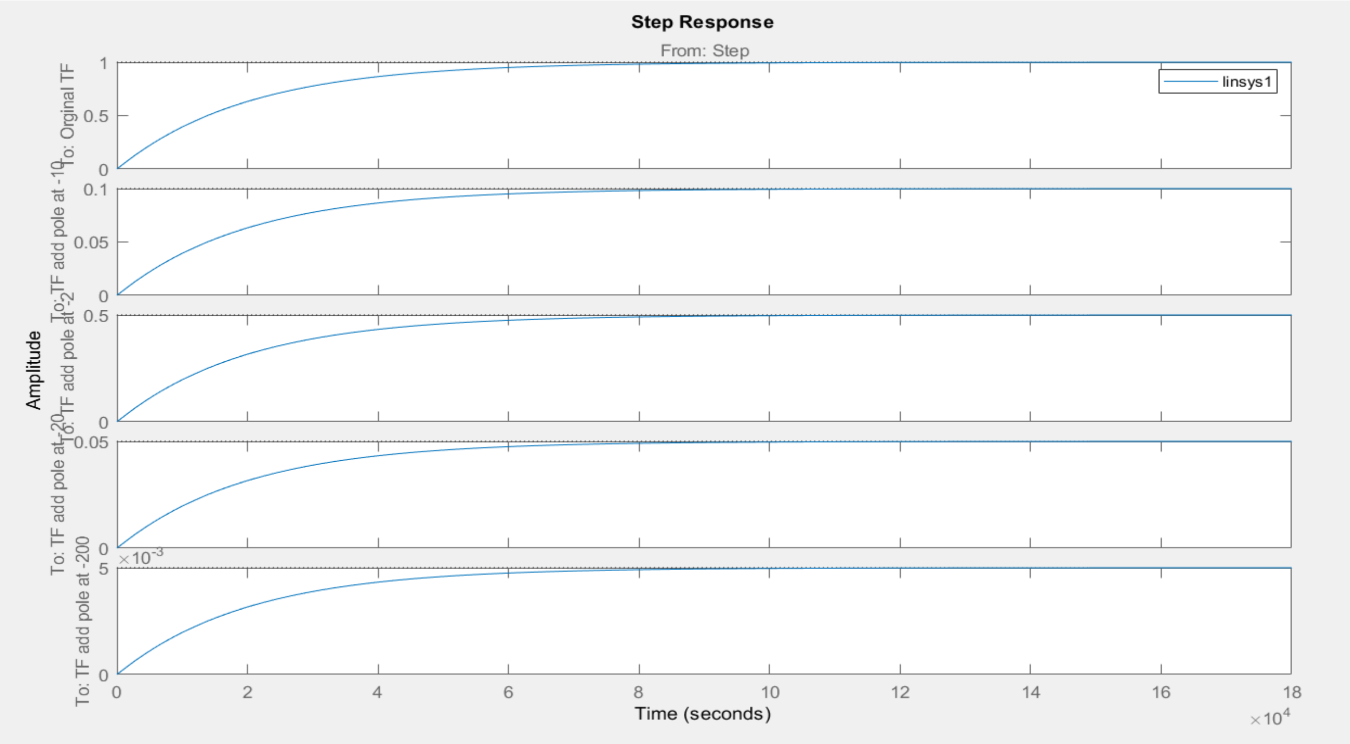
i) Using Simulink, add a pole to the second-order system and plot the step responses of the system when the higher-order pole is nonexistent, at −200; − 20; − 10, and −2. Make your plots on a single graph, using the Simulink LTI Viewer. Normalize all plots to a steady-state value of unity. Record percent overshoot, settling time, peak time, and rise time for each response.



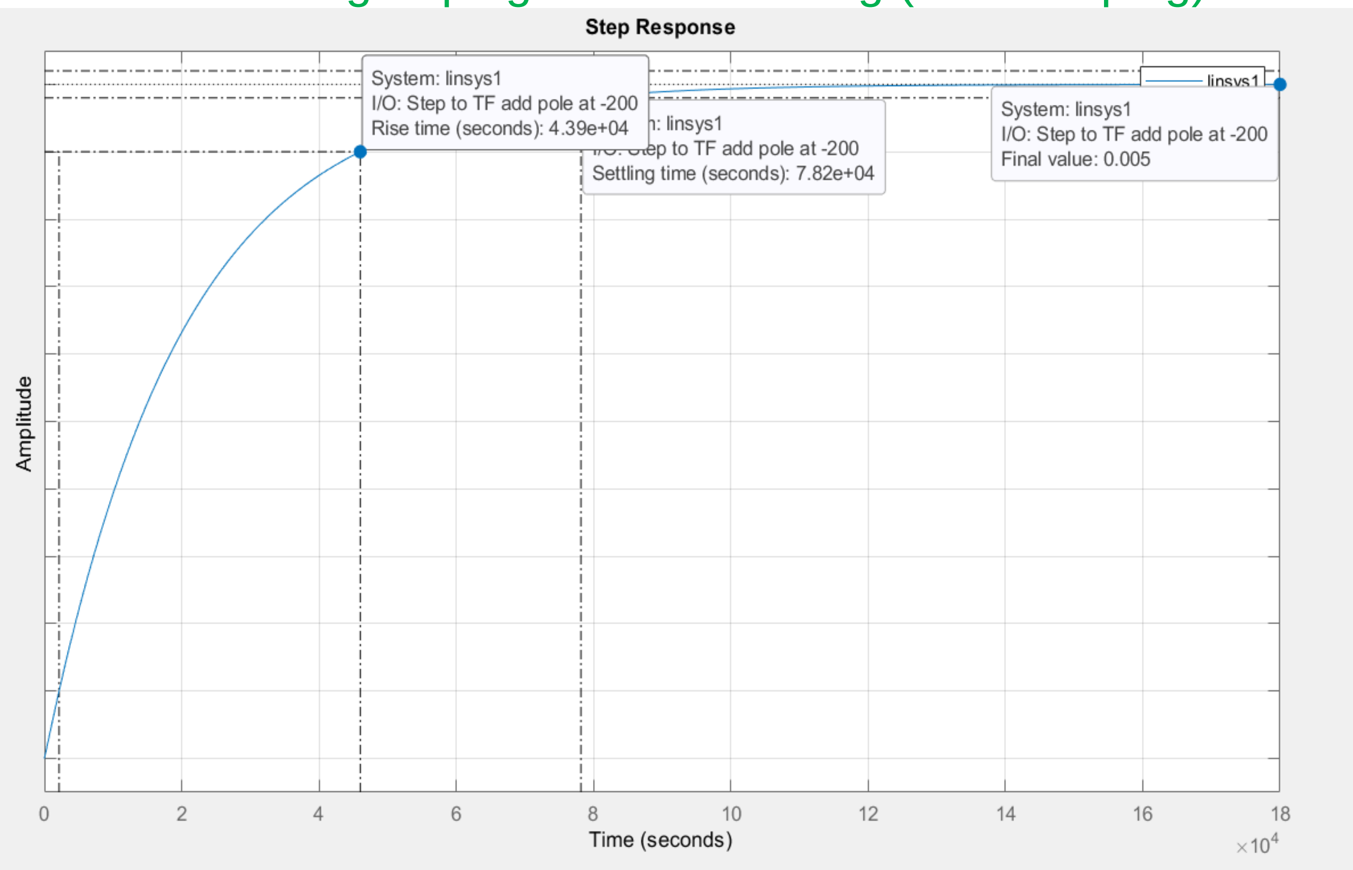
I don't know what is the values of K to use here but I will use 3 values K = 1 (overdamping), K = 166.6667(critically damping), K = 2000 (under damping)

Note: at all values of K the step response is the same because the system's poles before adding any poles are too small about $\frac{-1}{30}$ so poles at -200, -20, -10, and -2 are relatively larger than the original sys poles so they consider as dominant poles and the system still second order.

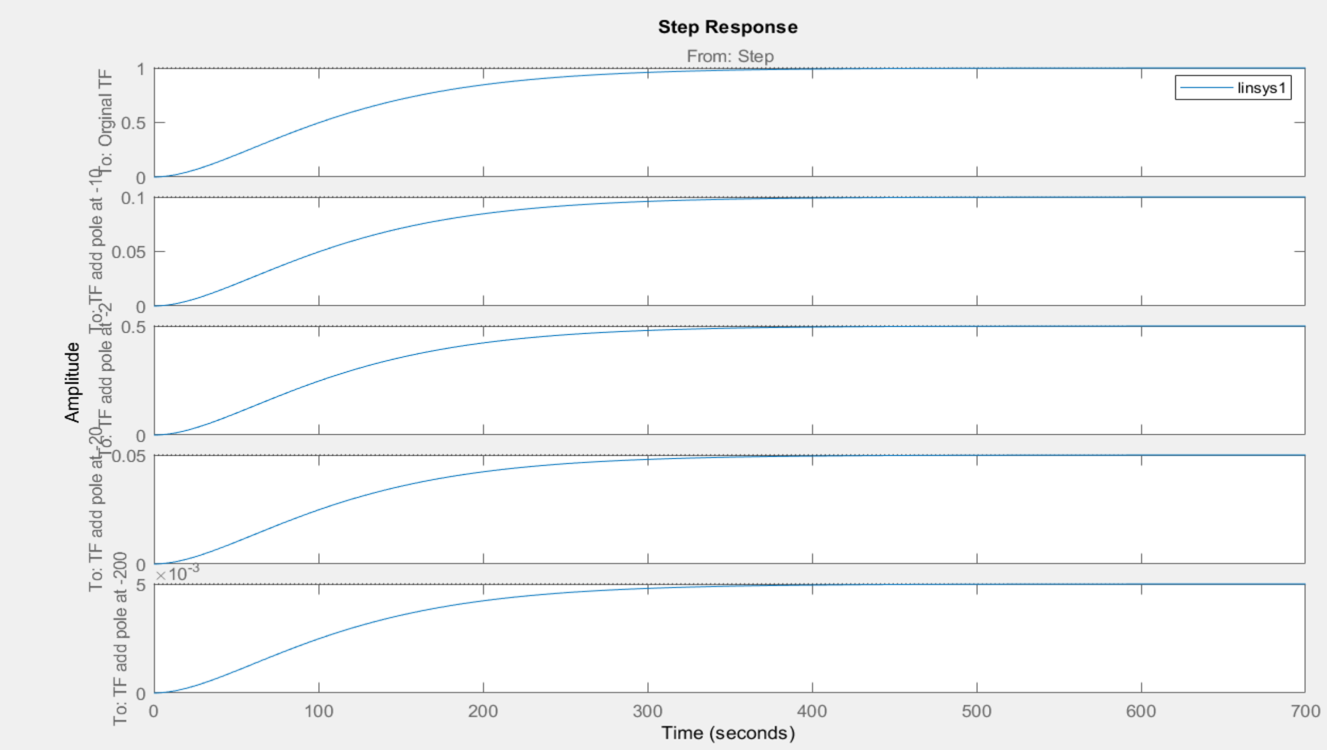
1)At $K = 1$ before grouping and normalizing (overdamping)



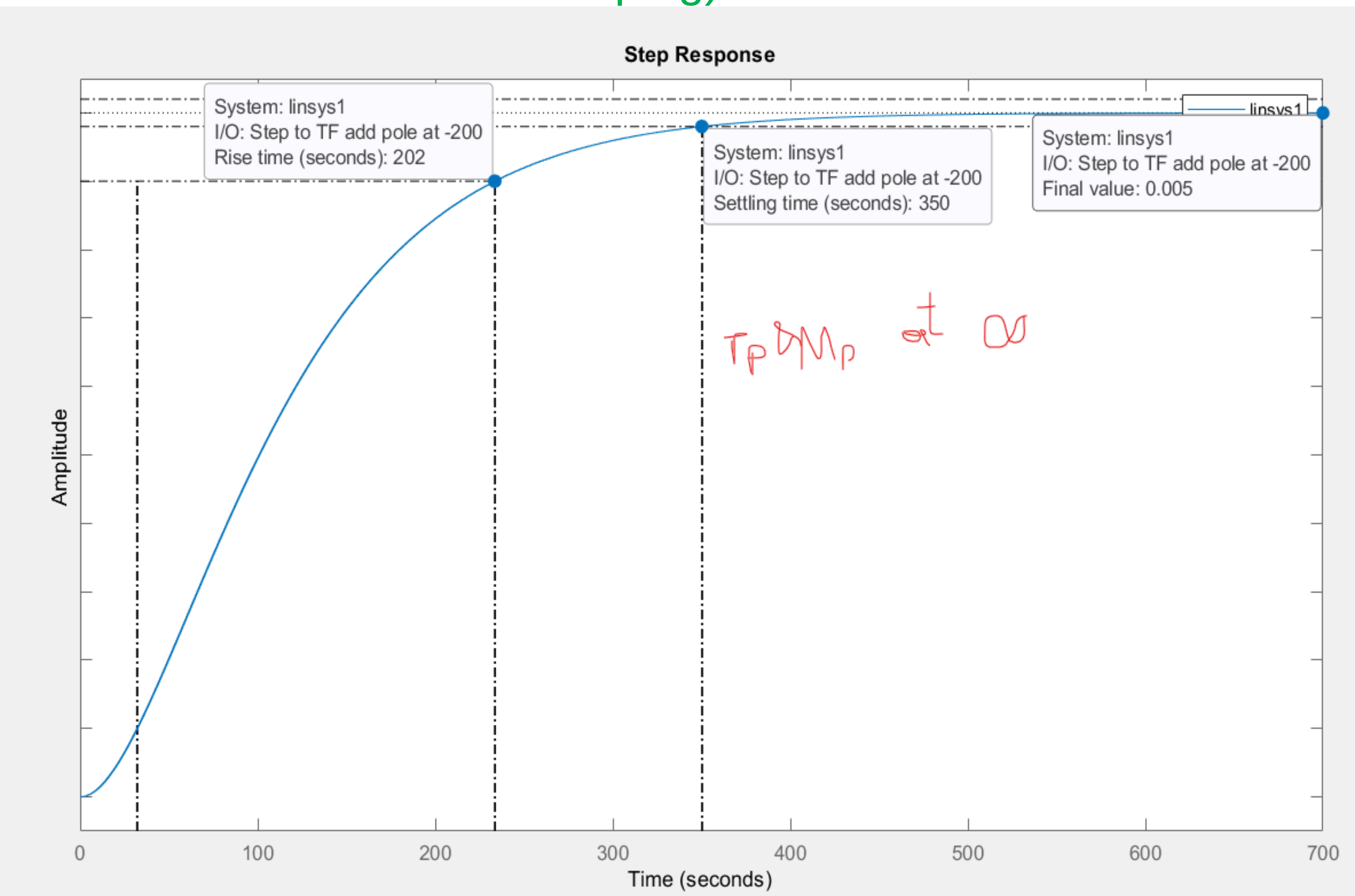
At $K = 1$ After grouping and normalizing (overdamping)



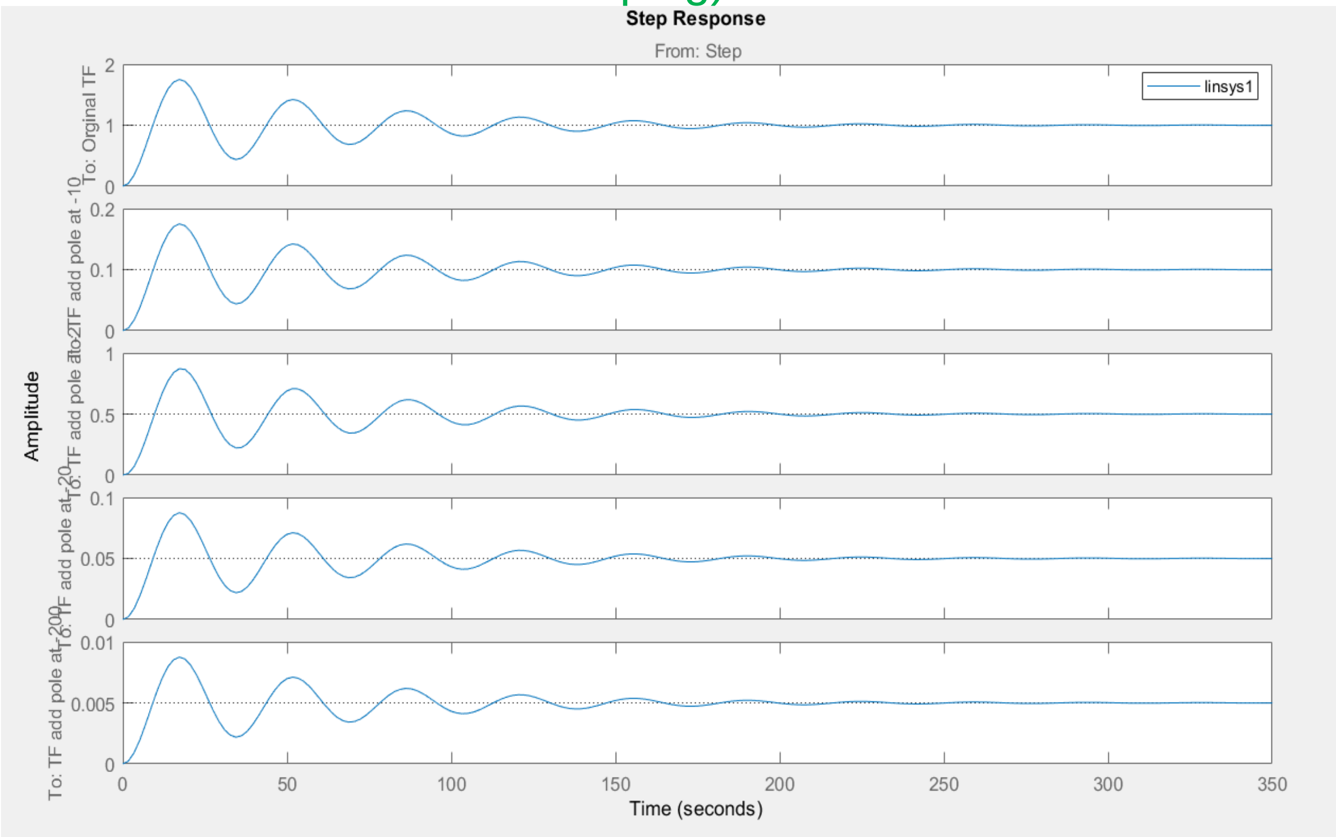
2)At K = 166.6667 before grouping and normalizing (critical damping)



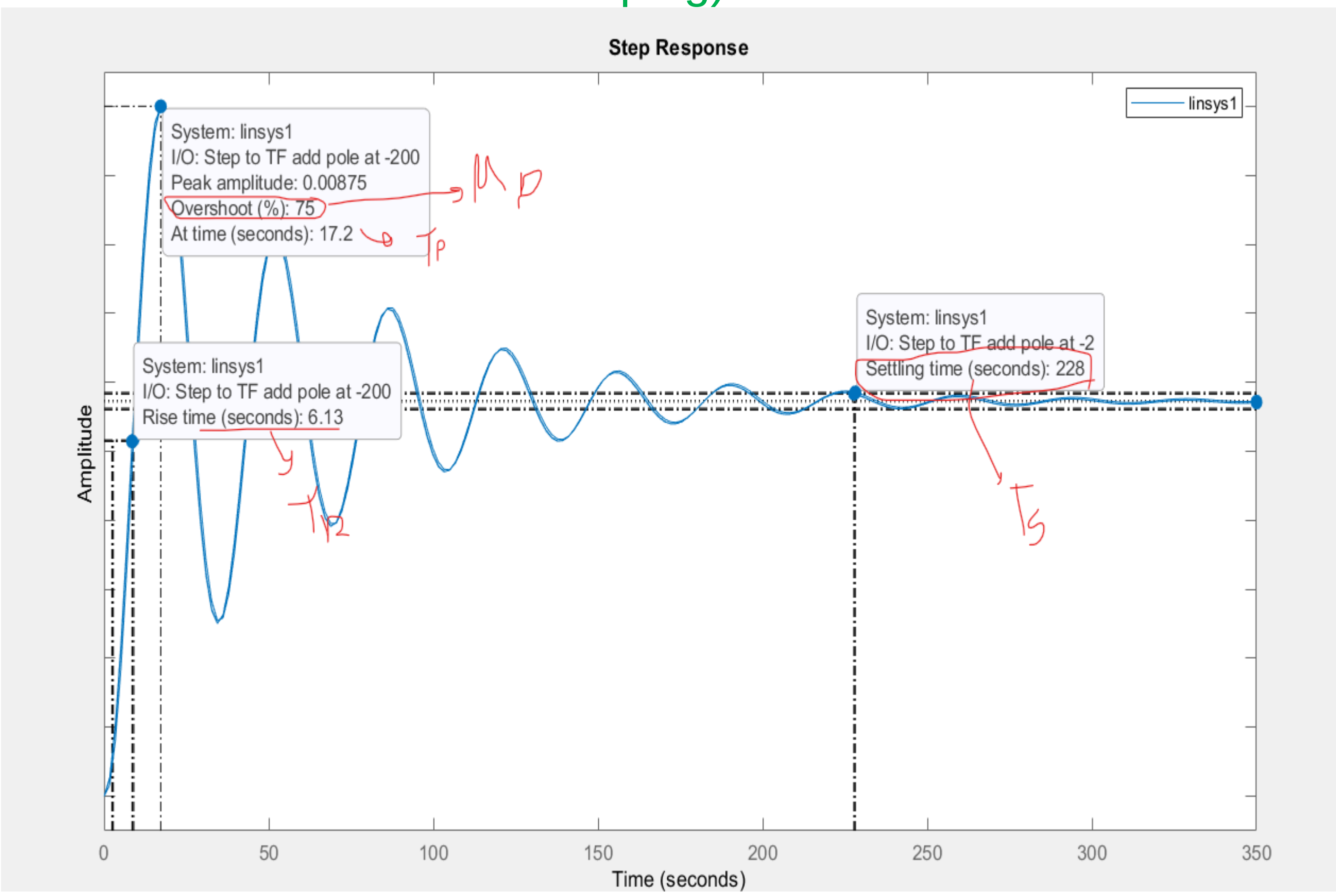
At K = 166.6667 After grouping and normalizing (critical damping)



3)At K = 2000 before grouping and normalizing (under damping)



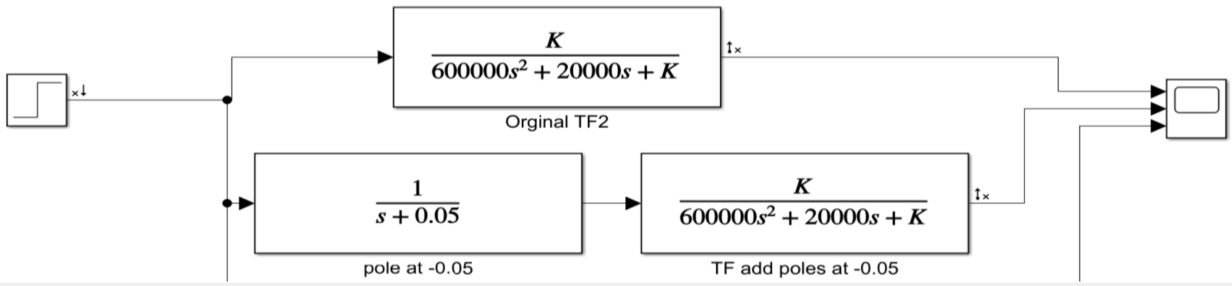
At K=2000 After grouping and normalizing (under damping)

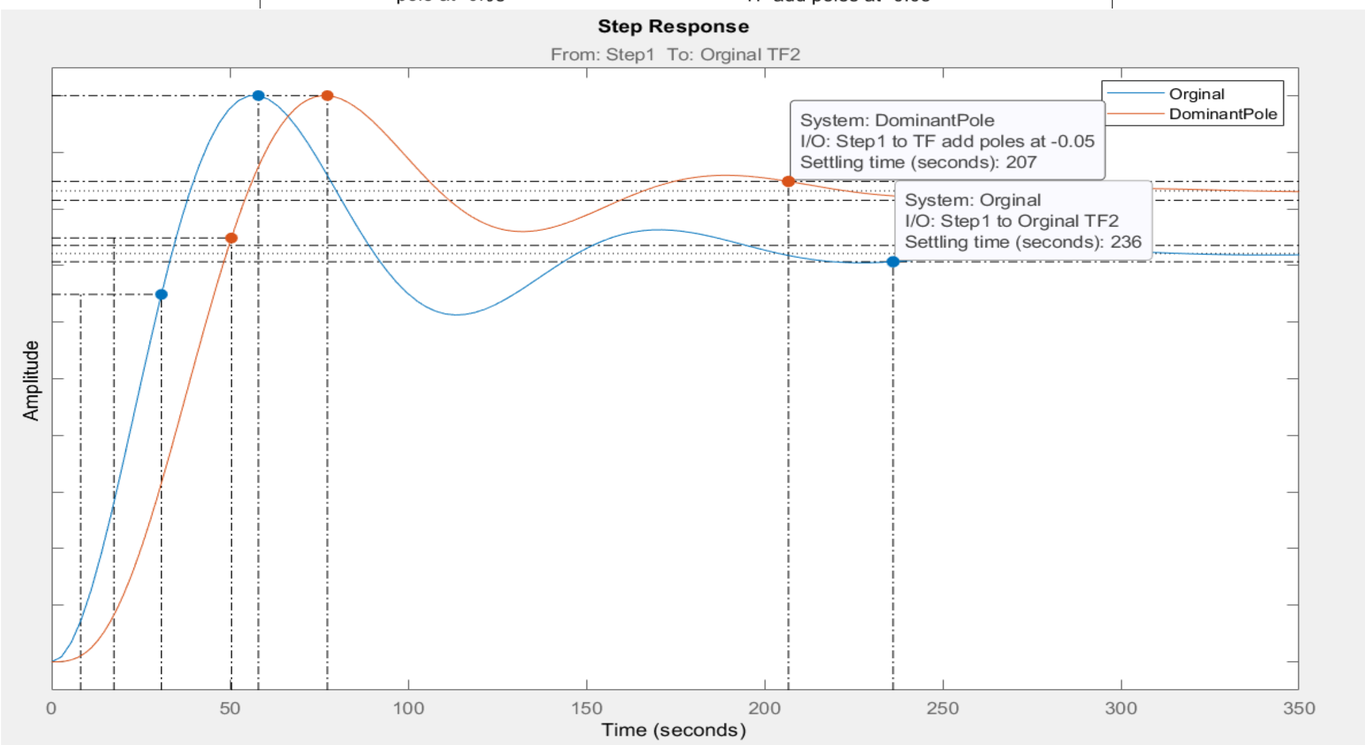


j) Discuss the effect upon the transient response of the proximity of a higher-order pole to the second-order system.

As I have said above the poles at -200, -20, -10, and -2 are relatively larger than the system poles to all positive values of K which is approximately $\frac{-1}{30}$ so the system will not affect by these poles and the system acts as the second order, not the third order **But** if I put K = 2000 then the original system poles will be

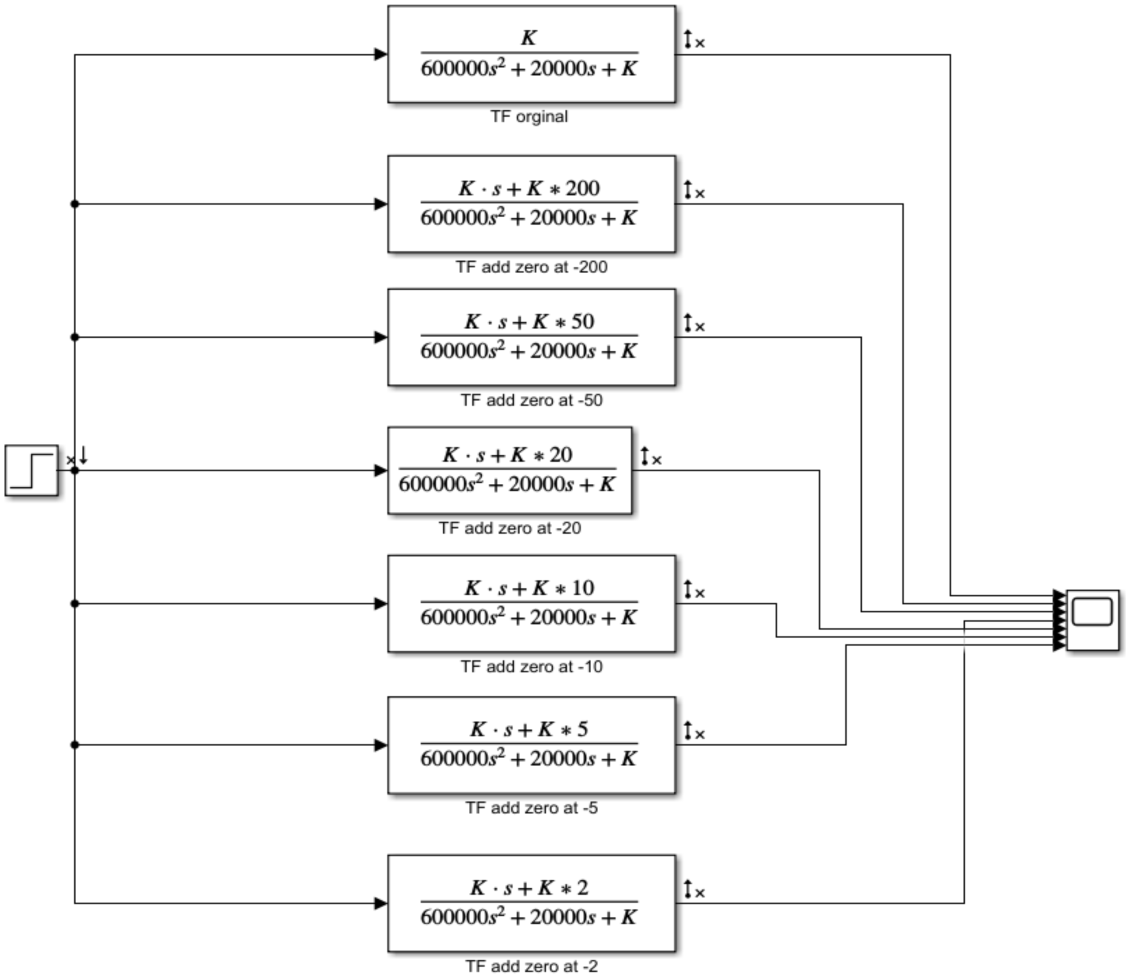
S1= $\frac{-1}{60}$ +$\frac{\sqrt{11}}{60}$ i && S1= $\frac{-1}{60}$ -$\frac{\sqrt{11}}{60}$ i, then I will add a dominant pole say at $\frac{-1}{20}$ = -0.05



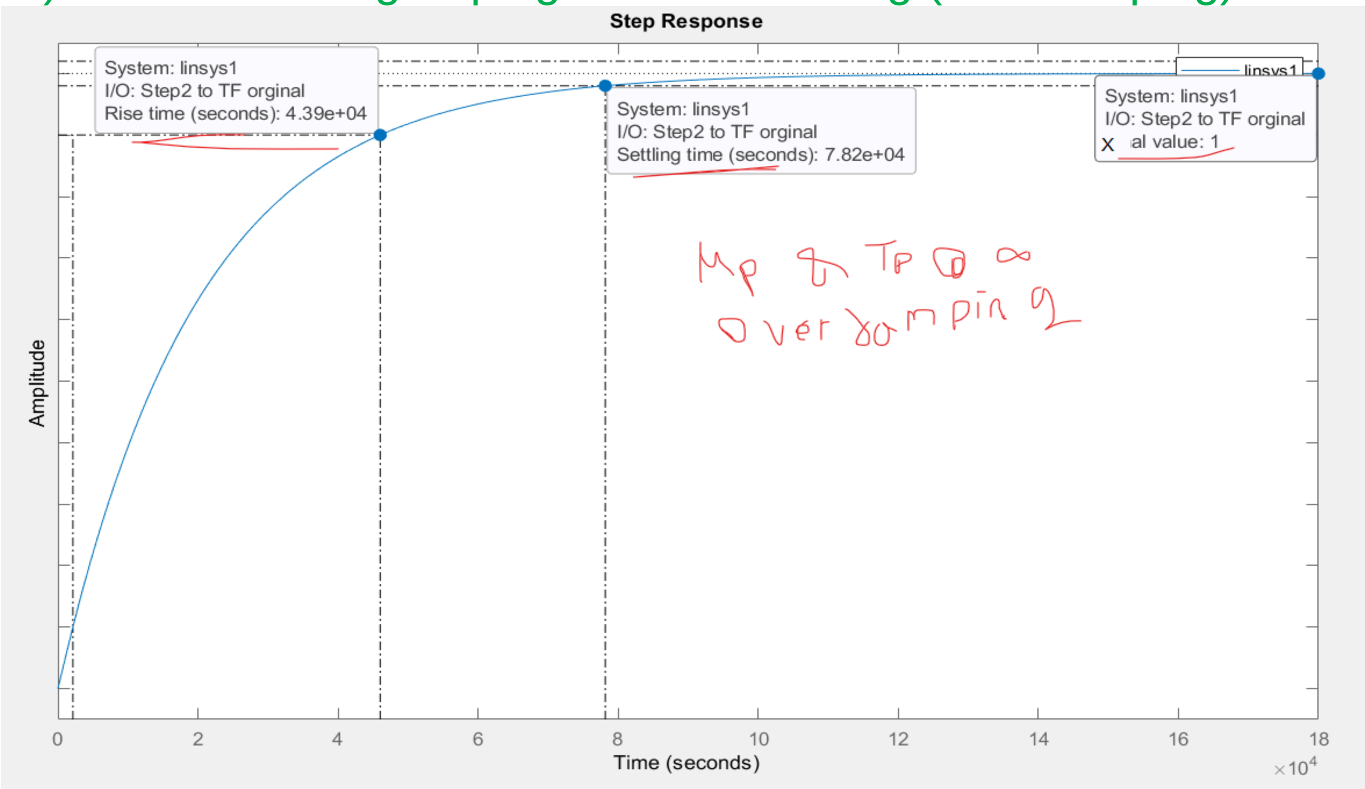


Adding a dominant pole increases peak time and raise time (slows the response of the system) and decreases the maximum overshoot, If the added pole is positive or in the same location as the system becomes unstable.

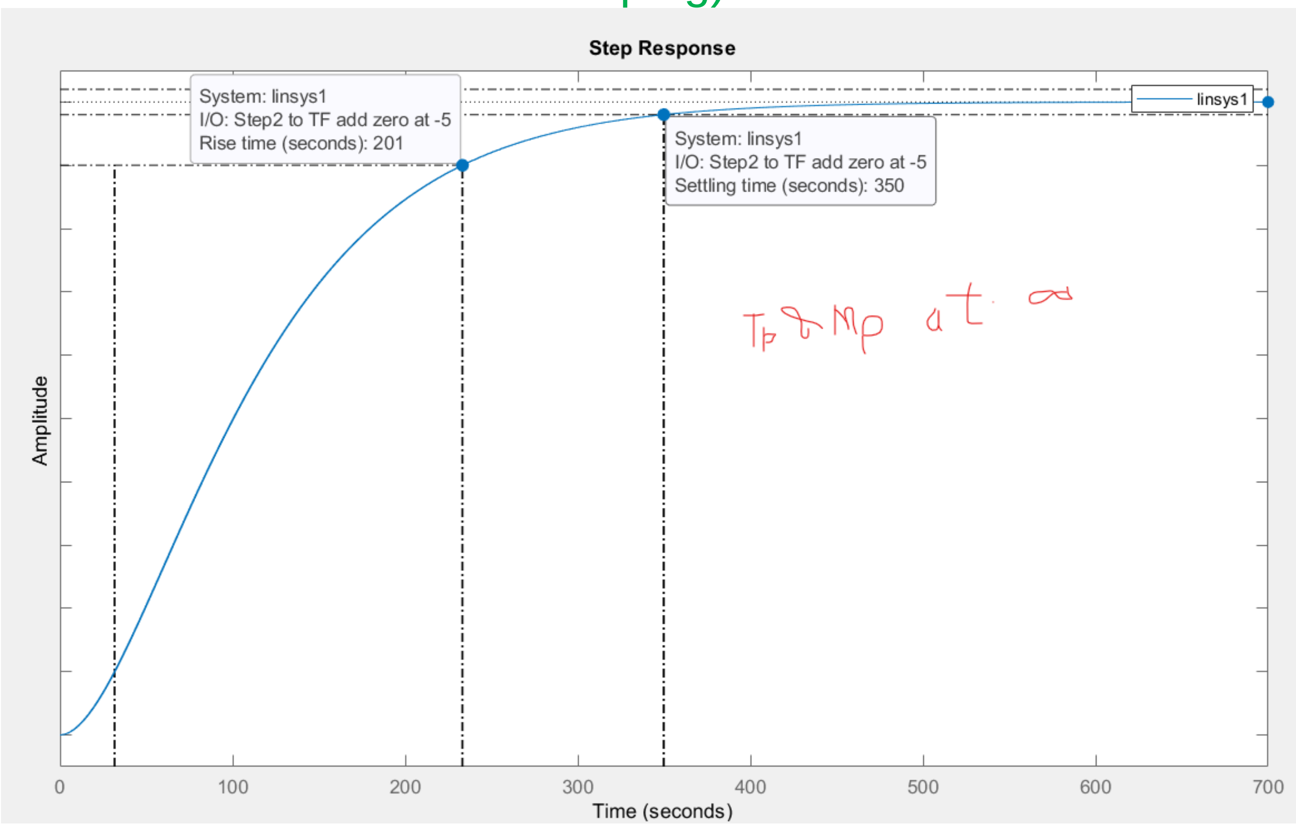
k) Using Simulink, add a zero to the second-order system and plot the step responses of the system when the zero is nonexistent, at −200; −50; −20; −10; −5, and −2. Make your plots on a single graph, using the Simulink LTI Viewer. Normalize all plots to a steady-state value of unity. Record percent overshoot, settling time, peak time, and rise time for each response



1) At K = 1 After grouping and normalizing (overdamping)

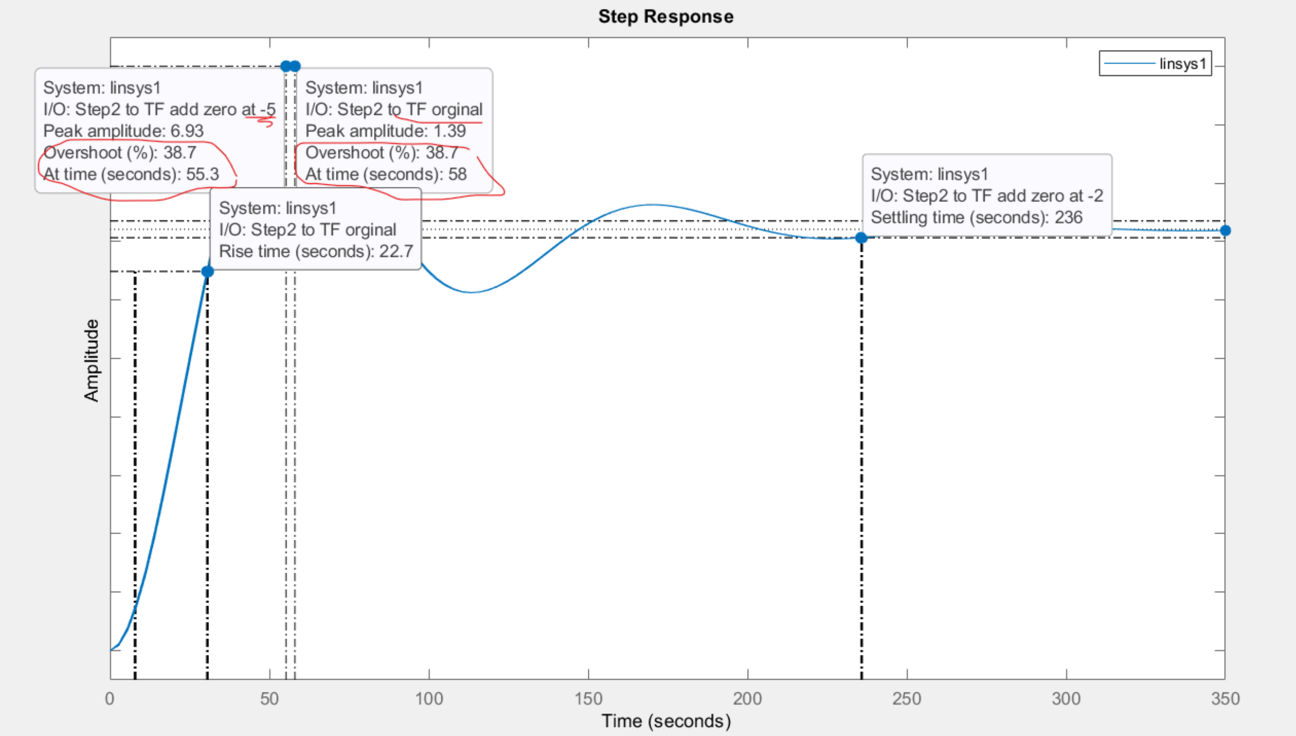


2)At K = 166.6667 after grouping and normalizing (critical damping)



Adding zeros doesn’t change anything here because zeros are so far from the poles.

3)At K=2000 After grouping and normalizing (under damping)



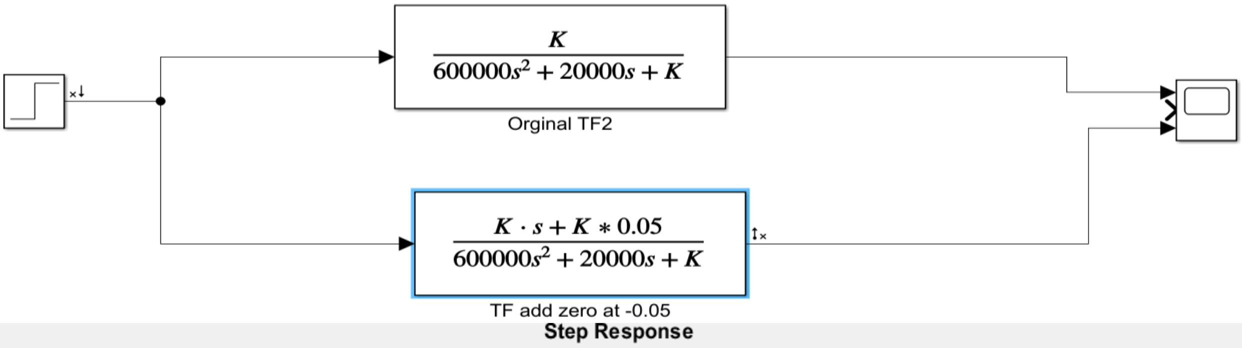
Note: There is a difference in peak time and maximum overshoot in the case of zeros at -2 and -5 because it's relativity close to poles so adding zero near to poles decreases the peak time so the system becomes faster and the maximum overshoot increases

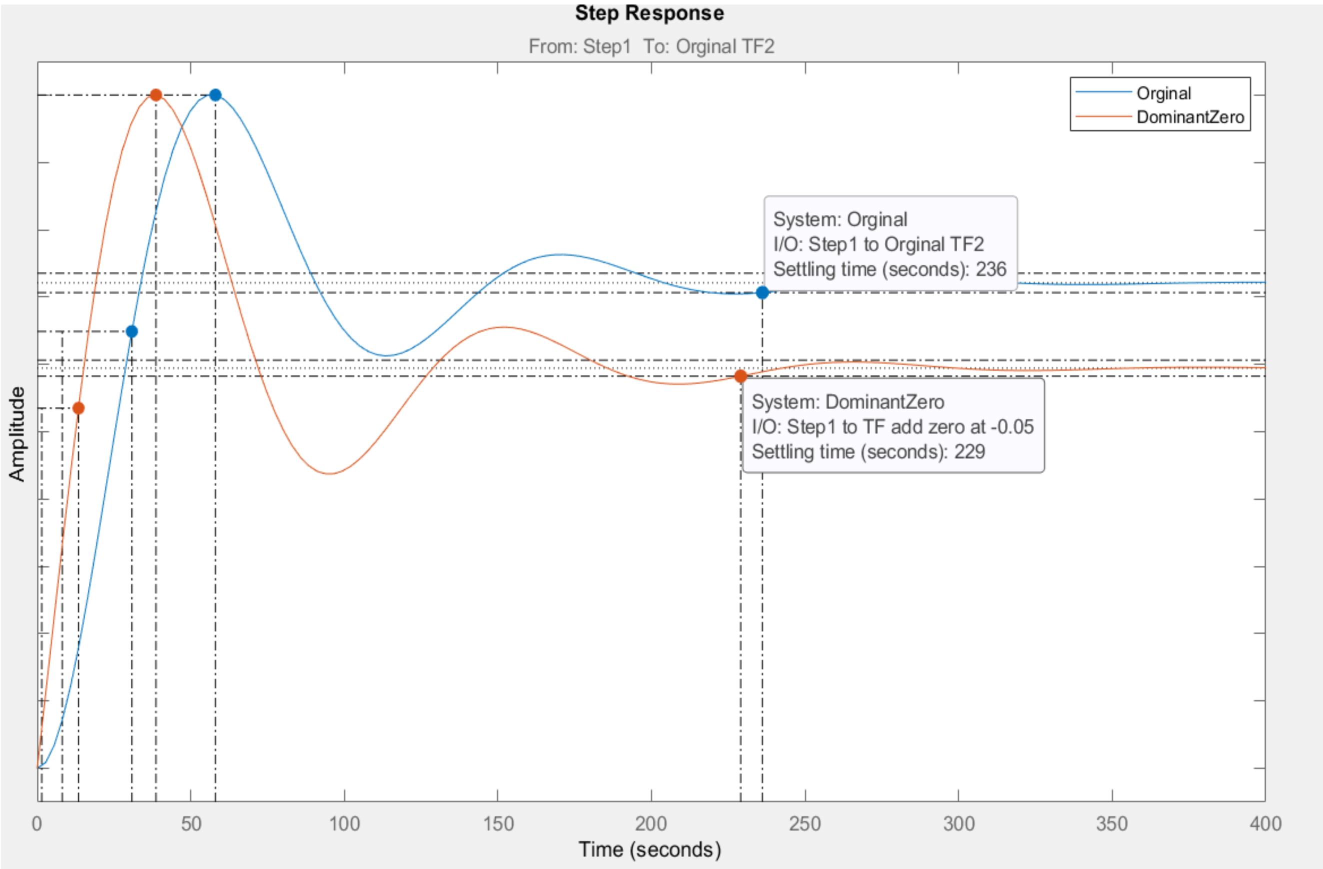
l) Discuss the effect upon the transient response of the proximity of a zero to the dominant second-order pole pair:

The zeros at -200, -50, -20, and -10 is relatively far away from the poles so the system doesn't affect by them but it has a little change when zero at -5 and -2.

But if I put K = 2000 then the original system poles will be

S1= $\frac{-1}{60}$ +$\frac{\sqrt{11}}{60}$ i && S1= $\frac{-1}{60}$ -$\frac{\sqrt{11}}{60}$ i, then I will add a dominant zero say at $\frac{-1}{20}$ = -0.05 which means it is so close to poles then





Adding dominant zero to the system decreases the peak and the raise time (fasts the response of the system) and increases the maximum overshoot.