S E V E N

Steady-State Errors

SOLUTIONS TO CASE STUDIES CHALLENGES

**Antenna Control: Steady-State Error Design via Gain**

**a.** G(s) = . System is Type 1. Step input: e(∞) = 0; Ramp input:   
e(∞) = = = ; Parabolic input: e(∞) = ∞.

**b.**  = = 0.2. Therefore, K = 12.95. Now test the closed-loop transfer function,   
T(s) = , for stability. Using Routh-Hurwitz, the system is stable.

|  |  |  |
| --- | --- | --- |
| s3 | 1 | 198 |
| s2 | 151.32 | 989.25 |
| s1 | 191.46253 | 0 |
| s0 | 989.25 | 0 |

**Video Laser Disc Recorder: Steady-State Error Design via Gain**

**a.** The input, 15t2 , transforms into 30/s3. e(∞) = 30/Ka = 0.005.

Ka = \* K1K2K3 = 6x10-3 K1K2K3. Therefore: e(∞) = 30/Ka = 

= 5x10-3. Therefore K1K2K3 = 106.

**b.** Using K1K2K3 = 106, G(s) = . Therefore, T(s) = .

Making a Routh table,

|  |  |  |
| --- | --- | --- |
| s3 | 1 | 2x105 |
| s2 | 2x104 | 1.2x108 |
| s1 | 194000 | 0 |
| s0 | 120000000 | 0 |

we see that the system is stable.

**c.**

**Program:**

numg=200000\*[1 600];

deng=poly([0 0 -20000]);

G=tf(numg,deng);

'T(s)'

T=feedback(G,1)

poles=pole(T)

**Computer response:**

ans =

T(s)

Transfer function:

200000 s + 1.2e008

------------------------------------

s^3 + 20000 s^2 + 200000 s + 1.2e008

poles =

1.0e+004 \*

-1.9990

-0.0005 + 0.0077i

-0.0005 - 0.0077I

Answers to Review Questions

**1.** Nonlinear, system configuration

**2.** Infinite

**3.** Step(position), ramp(velocity), parabola(acceleration)

**4.** Step(position)-1, ramp(velocity)-2, parabola(acceleration)-3

**5.** Decreases the steady-state error

**6.** Static error coefficient is much greater than unity.

**7.** They are exact reciprocals.

**8.** A test input of a step is used; the system has no integrations in the forward path; the error for a step input is 1/10001.

**9.** The number of pure integrations in the forward path

**10.** Type 0 since there are no poles at the origin

**11.** Minimizes their effect

**12**. If each transfer function has no pure integrations, then the disturbance is minimized by decreasing the plant gain and increasing the controller gain. If any function has an integration then there is no control over its effect through gain adjustment.

**13.** No

**14.** A unity feedback is created by subtracting one from H(s). G(s) with H(s)-1 as feedback form an equivalent forward path transfer function with unity feedback.

**15.** The fractional change in a function caused by a fractional change in a parameter

**16.** Final value theorem and input substitution methods

SOLUTIONS To Problems

**1.**

The system is type 1, so for a step input . For the ramp , , . For the parabolic input .

**2.**

**a**. From the figure .

**b**. Since the system is linear, and because the original input was , the new steady state error is .

**3.**

The system is type 2, , .

**4.**

Reduce the system to an equivalent unit feedback system by first moving 1/s to the left past the summing junction. This move creates a forward path consisting of a parallel pair in cascade with a feedback loop consisting of and . Thus,

Hence the system is Type 1, and the steady-state errors are as follows:

Steady state error for

Steady state error for

Steady state error for

**5.**

System is type 0, so for . For ramp and parabolic inputs .

**6.**



Thus,



**7.**



Therefore, =s2E(s) = s2 =.

**8.**

Since , the system is type 0. Therefore for , ; and for , .

**9.**

**a.**  The closed-loop transfer function is, from which, *n* = and 2*n* = 75. Thus, ** = 0.53 and   
 = 14.01%.**b.**  *Ts* = = = 0.107 second.**c.**  Since system is Type 1, ess for 5*u(t)* is zero.**d.**  Since *Kv* is = 66.67, *ess* = = 0.075.**e.**  ess = ∞, since system is Type 1.

**10.**

. So .**11.**

. Therefore, K = 7291.667.**12.** One way to solve the problem is obtain using any method. Then

.

When =13.3333=.Solving we get

When So

When So

Since , this is a type 0 system.

**13.**

**a.** Collapsing the inner loop and multiplying by 1000/*s* yields the equivalent forward-path transfer

function as,

Hence, the system is Type 1.

**b.** Following the procedure in a., we get that the forward path is

So the system is type 0.

**14.**

The transfer function from command input to error signal can be found using Mason’s rule or any other method:

Letting and by the final value theorem:



1. If  is type 0, it is required that .
2. If  is type 1, it is required that must be type 0, or .
3. If  is type 2, it is required that must be type 0 or 1, or .

**15.**

.

For Type 0, step input: R(s) = , and 

For Type 0, ramp input: R(s) = , and  
 

For Type 0, parabolic input: R(s) = , and 

For Type 1, step input: R(s) = , and 

For Type 1, ramp input: R(s) = , and 

For Type 1, parabolic input: R(s) = , and

For Type 2, step input: R(s) = , and 

For Type 2, ramp input: R(s) = , and 

For Type 2, parabolic input: R(s) = , and 0



**16.**

.

Hence, *Ka* = 180000.

**17.**

Find the equivalent forward loop transfer function for a unit feedback system. . Thus ; from which

**18.**

**a.** e(∞) = = . But, Kv = = 60,000. Hence, K = 10,000. For finite error for a ramp input, n = 1.

**b.** 





**19.**

**a.** Type 0

**b.** E(s) = . Thus, .

**c.** e(∞) = ∞, since the system is Type 0.

**20.**

e(∞) = =  = 0.05. Thus, K = 142.5

**21.**

To meet steady-state error characteristics:



Therefore, K = 92.

To meet the transient requirement: Since T(s) = ,   
n2 = 10 = 2 + K ; 2n = = K+2Solving for ,  = ±1. For  = +1, K = 1.16 and  = 7.76. An alternate solution is  = -1, K = 5.16, and  = 1.74.

**22.**

**a.** System Type = 1

**b.** Assume G(s) = . Therefore, e(∞) = = = 0.01, or = 100.

But, T(s) = = .

Since n = 10, K = 100, and  = 1. Hence, G(s) = .

**c.** 2n =  = 1. Thus,  = .

**23.**

1. Since the steady-state output of the system can follow a ramp input with a finite error, then the system is type-1;
2. The velocity error constant is given by:.

Given that , we have: .

But, , where .

Thus, .

1. But 
2. When *K* = 4 and α = 0.4, , then:

;  rad/sec;

.

**24.**

T(s) = = . Hence, K+ = 2, K = n2 = (12+12) = 2.   
Also, e(∞) = =

**25.**

**a.** For 20% overshoot,  = 0.456. Also, Kv = 1000 = . Since T(s) = , 2n = a, and   
n = . Hence, a = 0.912 . Solving for a and K, K = 831,744, and a = 831.744.

**b.** For 10% overshoot,  = 0.591. Also, = 0.01. Thus, Kv = 100 = . Since T(s) = , 2n = a, and n = . Hence, a = 1.182. Solving for a and K, K = 13971 and a = 139.71.

**26.**

**a.** For the inner loop:

G1(s) = =

Ge(s) = G1(s) =

T(s) = =

**b.** From Ge(s), system is Type 1.

**c.** Since system is Type 1, ess = 0

**d.** ; From Ge(s), Kv =  = . Therefore, ess = = 15.

**e.** Poles of T(s) = -3.0190, -1.3166, 0.3426 ± j0.7762, -0.3495. Therefore, system is unstable and results of (c) and (d) are meaningless

**27.**

The equivalent forward transfer function is . Also . From the problem statement and where . Solving simultaneously for and we get , .

**28.**

Error due only to disturbance: Rearranging the block diagram to show D(s) as the input,



Therefore,

-E(s) = D(s) = D(s)

For D(s) = , eD(∞) = .

Error due only to input: eR(∞) = = = .

Design:

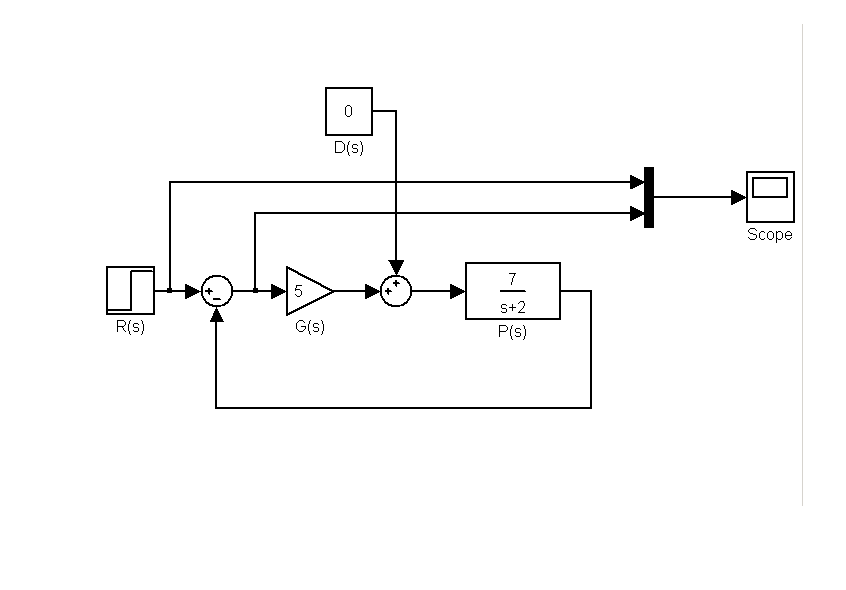
eD(∞) = - 0.00001 = - , or *K*1 = 150,000.

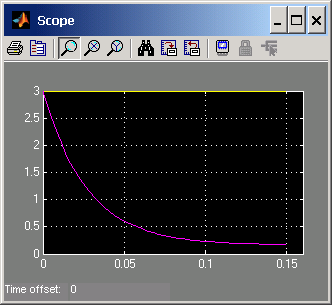
eR(∞) = 0.002 = , or *K*2 = 0.02

**29.**

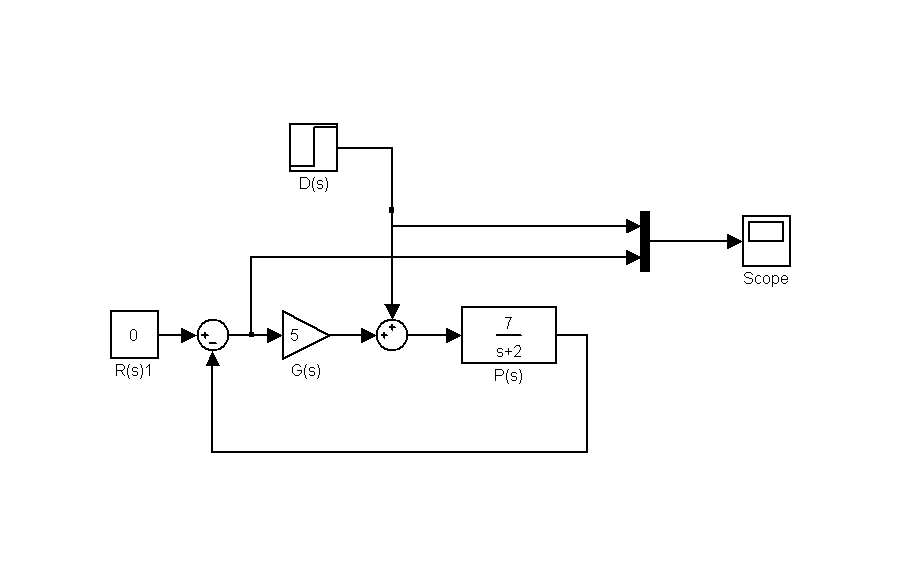
1. The open loop transmission is , so . For a unit step input . Since the input is threefold that we have that 

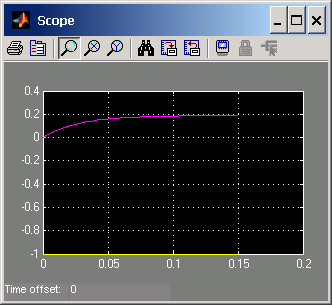
**b.**





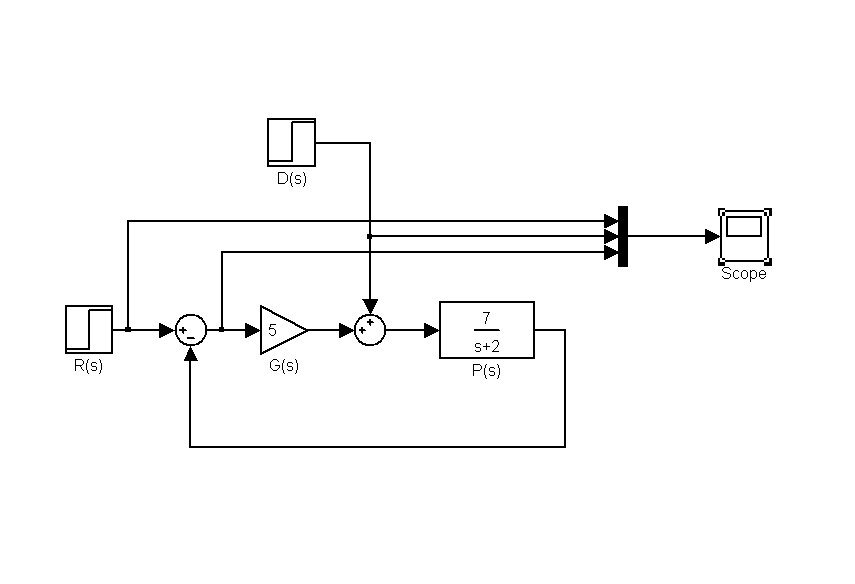
1. The transfer function from disturbance to error signal is  Using the final value theorem 

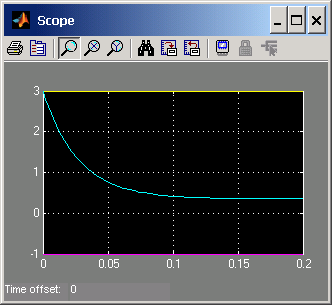




**e**. 

**f**.





**30.**

****

**31.**

**System 1:**

Forming a unity-feedback path, the equivalent unity feedback system has a forward transfer function of



**a.** Type 0 System; **b.** *Kp* = ; **c.** step input; **d.** e(∞) = = 3/4;

**e. .**

**System 2:**

Forming a unity-feedback path, the equivalent unity feedback system has a forward transfer function of



**a.** Type 1 System; **b. **; **c.** ramp input; **d.** ;

**e. .**

**32.**

**System 1.** Push 5 to the right past the summing junction:





Produce a unity-feedback system:





Thus, . *Kp* = . estep = = 0.67, eramp = ∞, eparabola = ∞.

Checking for stability, from first block diagram above, T(s) = . The system is stable.

**System 2.** Push 20 to the right past the summing junction and push 10 to the left past the pickoff point:





Produce a unity-feedback system:





Thus, . *Kp* = .

estep = = -20, eramp = ∞, eparabola = ∞.

Checking for stability, from first block diagram above, .

Therefore, system is stable and steady-state error calculations are valid. 

**33.**

Produce a unity-feedback system:



Thus, Ge(s) = = . Error = 0.001 = .

Therefore, Kp = 999 = . Hence, K = 1.001001.

Check stability: Using original block diagram, T(s) = = .

Making a Routh table:

|  |  |  |
| --- | --- | --- |
| s3 | 1 | K |
| s2 | 2 | K |
| s1 |  | 0 |
| s0 | K | 0 |

Therefore, system is stable and steady-state error calculations are valid.

**34.**

**Program:**

K=10

numg1=K\*poly([-1 -2]);deng1=poly([0 0 -4 -5 -6]);

'G1(s)='

G1=tf(numg1,deng1)

numh1=[1 6];denh1=poly([-8 -9]);

'H1(s)='

H1=tf(numh1,denh1)

'H2(s)=H1-1'

H2=H1-1

%Form Ge(s)=G1(s)/(1+G1(s)H2(s)

'Ge(s)=G1(s)/(1+G1(s)H2(s))'

Ge=feedback(G1,H2)

%Test system stability

'T(s)=Ge(s)/(1+Ge(s))'

T=feedback(Ge,1)

pole(T)

Kp=dcgain(Ge)

'sGe(s)'

sGe=tf([1 0],1)\*Ge;

sGe=minreal(sGe)

Kv=dcgain(sGe)

's^2Ge(s)'

s2Ge=tf([1 0],1)\*sGe;

s2Ge=minreal(s2Ge)

Ka=dcgain(s2Ge)

essstep=30/(1+Kp)

essramp=30/Kv

essparabola=60/Ka

K=1E6

numg1=K\*poly([-1 -2]);deng1=poly([0 0 -4 -5 -6]);

'G1(s)='

G1=tf(numg1,deng1)

numh1=[1 6];denh1=poly([-8 -9]);

'H1(s)='

H1=tf(numh1,denh1)

'H2(s)=H1-1'

H2=H1-1

%Form Ge(s)=G1(s)/(1+G1(s)H2(s)

'Ge(s)=G1(s)/(1+G1(s)H2(s))'

Ge=feedback(G1,H2)

%Test system stability

'T(s)=Ge(s)/(1+Ge(s))'

T=feedback(Ge,1)

pole(T)

Kp=dcgain(Ge)

'sGe(s)'

sGe=tf([1 0],1)\*Ge;

sGe=minreal(sGe)

Kv=dcgain(sGe)

's^2Ge(s)'

s2Ge=tf([1 0],1)\*sGe;

s2Ge=minreal(s2Ge)

Ka=dcgain(s2Ge)

essstep=30/(1+Kp)

essramp=30/Kv

essparabola=60/Ka

**Computer response:**

K =

10

ans =

G1(s)=

Transfer function:

10 s^2 + 30 s + 20

-------------------------------

s^5 + 15 s^4 + 74 s^3 + 120 s^2

ans =

H1(s)=

Transfer function:

s + 6

---------------

s^2 + 17 s + 72

ans =

H2(s)=H1-1

Transfer function:

-s^2 - 16 s - 66

----------------

s^2 + 17 s + 72

ans =

Ge(s)=G1(s)/(1+G1(s)H2(s))

Transfer function:

10 s^4 + 200 s^3 + 1250 s^2 + 2500 s + 1440

-----------------------------------------------------------------------

s^7 + 32 s^6 + 401 s^5 + 2448 s^4 + 7178 s^3 + 7480 s^2 - 2300 s - 1320

ans =

T(s)=Ge(s)/(1+Ge(s))

Transfer function:

10 s^4 + 200 s^3 + 1250 s^2 + 2500 s + 1440

---------------------------------------------------------------------

s^7 + 32 s^6 + 401 s^5 + 2458 s^4 + 7378 s^3 + 8730 s^2 + 200 s + 120

ans =

-8.5901 + 0.3993i

-8.5901 - 0.3993i

-6.0000

-4.4042 + 0.1165i

-4.4042 - 0.1165i

-0.0057 + 0.1179i

-0.0057 - 0.1179i

Kp =

-1.0909

ans =

sGe(s)

Transfer function:

10 s^5 + 200 s^4 + 1250 s^3 + 2500 s^2 + 1440 s

-----------------------------------------------------------------------

s^7 + 32 s^6 + 401 s^5 + 2448 s^4 + 7178 s^3 + 7480 s^2 - 2300 s - 1320

Kv =

0

ans =

s^2Ge(s)

Transfer function:

10 s^6 + 200 s^5 + 1250 s^4 + 2500 s^3 + 1440 s^2

-----------------------------------------------------------------------

s^7 + 32 s^6 + 401 s^5 + 2448 s^4 + 7178 s^3 + 7480 s^2 - 2300 s - 1320

Ka =

0

essstep =

-330.0000

essramp =

Inf

essparabola =

Inf

K =

1000000

ans =

G1(s)=

Transfer function:

1e006 s^2 + 3e006 s + 2e006

-------------------------------

s^5 + 15 s^4 + 74 s^3 + 120 s^2

ans =

H1(s)=

Transfer function:

s + 6

---------------

s^2 + 17 s + 72

ans =

H2(s)=H1-1

Transfer function:

-s^2 - 16 s - 66

----------------

s^2 + 17 s + 72

ans =

Ge(s)=G1(s)/(1+G1(s)H2(s))

Transfer function:

1e006 s^4 + 2e007 s^3 + 1.25e008 s^2 + 2.5e008 s + 1.44e008

--------------------------------------------------------------------------------

s^7 + 32 s^6 + 401 s^5 - 997542 s^4 - 1.899e007 s^3 - 1.16e008 s^2 - 2.3e008 s

- 1.32e008

ans =

T(s)=Ge(s)/(1+Ge(s))

Transfer function:

1e006 s^4 + 2e007 s^3 + 1.25e008 s^2 + 2.5e008 s + 1.44e008

--------------------------------------------------------------------------------

s^7 + 32 s^6 + 401 s^5 + 2458 s^4 + 1.007e006 s^3 + 9.009e006 s^2 + 2e007 s

+ 1.2e007

ans =

-28.2460 +22.2384i

-28.2460 -22.2384i

16.7458 +22.2084i

16.7458 -22.2084i

-6.0000

-1.9990

-1.0007

Kp =

-1.0909

ans =

sGe(s)

Transfer function:

1e006 s^5 + 2e007 s^4 + 1.25e008 s^3 + 2.5e008 s^2 + 1.44e008 s

--------------------------------------------------------------------------------

s^7 + 32 s^6 + 401 s^5 - 9.975e005 s^4 - 1.899e007 s^3 - 1.16e008 s^2

- 2.3e008 s - 1.32e008

Kv =

0

ans =

s^2Ge(s)

Transfer function:

1e006 s^6 + 2e007 s^5 + 1.25e008 s^4 + 2.5e008 s^3 + 1.44e008 s^2

--------------------------------------------------------------------------------

s^7 + 32 s^6 + 401 s^5 - 9.975e005 s^4 - 1.899e007 s^3 - 1.16e008 s^2

- 2.3e008 s - 1.32e008

Ka =

0

essstep =

-330.0000

essramp =

Inf

essparabola =

Inf

**35.**

* 1. Mason’s rule can be used to find the open loop transfer from input to output:

Only one forward path, 

Three touching loops, , , 

; 

. Letting 



Since the system is not unity feedback, we calculate





The system is type 0.

* 1. For a step input we calculate 

Then 

**36.**

Y(s) = R(s) +

E(s) = R(s) - Y(s) = R(s) - R(s) -

= R(s) - D(s)

Thus,

e(∞) = limsE(s) = lim

**37.**

**a.** E(s) = R(s) - C(s). But, C(s) = [R(s) - C(s)H(s)]G1(s)G2(s) + D(s). Solving for C(s),

C(s) = +

Substituting into E(s),

E(s) = R(s) - D(s)

**b.** For R(s) = D(s) = ,

e(∞) = limsE(s) = 1 - -

* 1. Zero error if G1(s) and/or G2(s) is Type 1. Also, H(s) is Type 0 with unity dc gain.

**38.**

First find the forward transfer function of an equivalent unity feedback system.



Thus, 

Finding the sensitivity of e(∞), we get: *Se:a* = = = .

The following MATLAB M-file was written and used to plot the sensitivity, *e*, as a function of the parameter *a*. The graph obtained is shown below.

a = 0 : 0.1 : 10;

e = (a-1).^-1;

plot (a, e,'LineWidth',2)

grid

title 'Sensitivity to Parameter a'

xlabel 'a'

ylabel 'Sensitivity of errror, e, to a'



**39.**

From Eq. (~~7.657.~~7.70),

e(∞) = 1 - lim- lim=

Sensitivity to K1:

Se:K1 = = - = - = - 0.833

Sensitivity to K2:

Se:K2 = = ­= = - 0.89

**40.**

1. Using Mason’s rule:

; Loops  and , no non-touching loops. 



1. For a unit step input,



1. For a unit ramp input,



1. The system is type 0.

****

**41.**

1. Following Figure P7.26, the transfer function from to e is given by:

For we have that in steady state

It can be seen from this expression that if is type 1 or larger .

1. From Figure P7.26:

The error is now defined as

=

In steady state this expression becomes:

It can be seen in this equation that the steady state error cannot be made zero.

**42.**

**a.** The system’s type is 0 because the system shows a nonzero steady state error for a step input

**b.**

**c.** Since the system is type 0, for a ramp input .

**43.**

**a.** We will calculate the steady state error from so we start by calculating the system’s closed loop transfer function. From Mason’s Rule:

Applying the Final Value Theorem

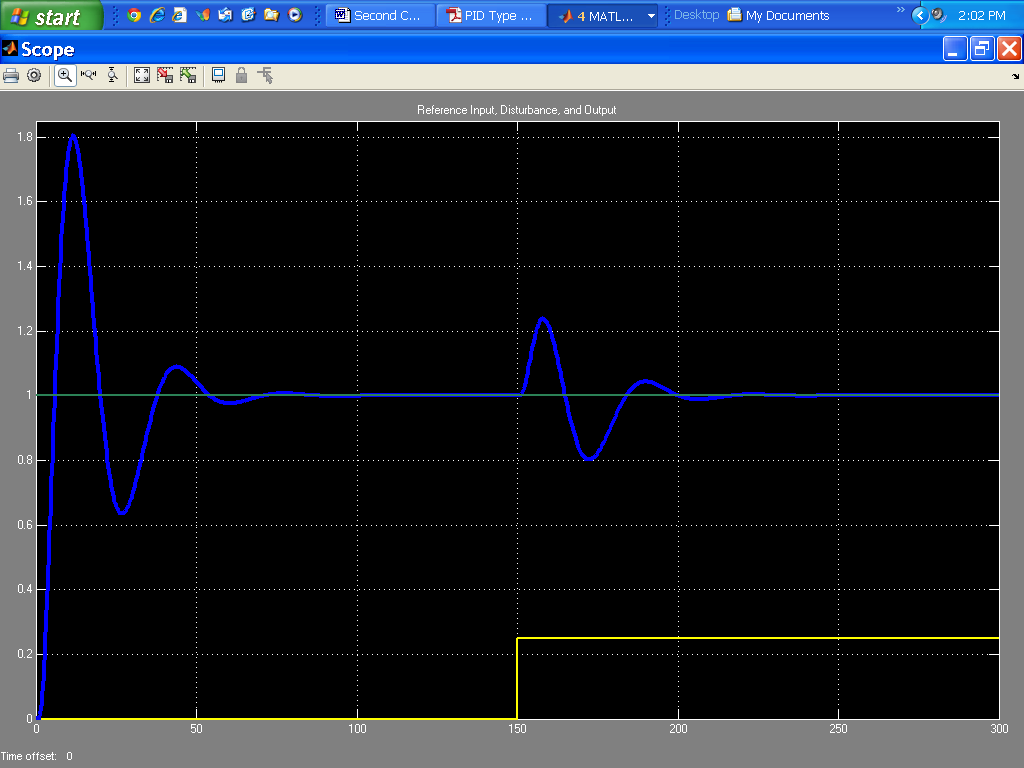
**b.** Since the steady state error to a step input is non-zero the system is type 0

**44.**

The Simulink model of this system and its step response (displayed in blue) from t = 0 to 300 seconds are shown below. Here, the reference input, *r*(*t*), is a unit step, *u*(*t*), applied at t = 0 (displayed in green), and the disturbance is *d(t)* = 0*.*25 *u*(*t*) applied at t = 150 seconds (displayed in yellow). As could be seen from this plot, the steady-state errors due to a step reference input, *r*(*t*) *= u(t)*, and due to a disturbance, *d(t)* = 0*.*25 *u*(*t*), are both equal to zero.

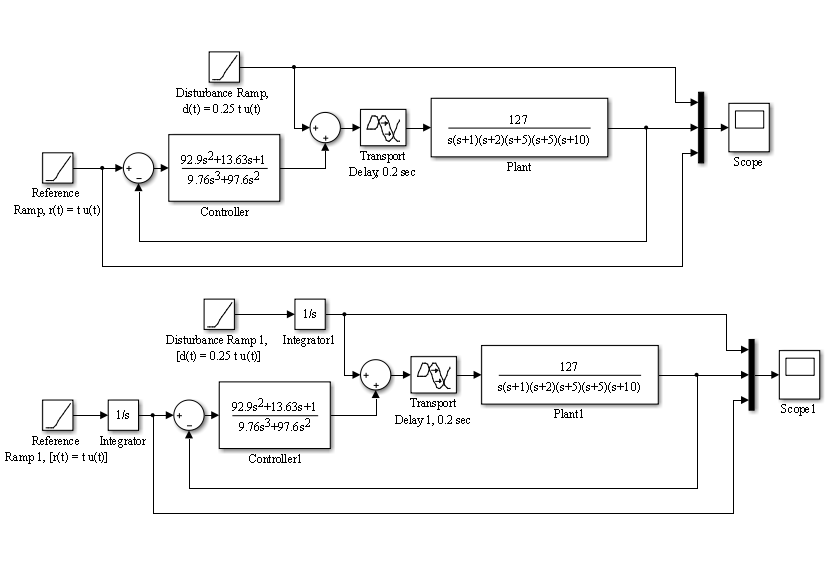
It should be noted, however, that the relative stability of this Type 3 system is poor, since it exhibits a high percent overshoot, % OS ≅ 80 %, due to the unit-step reference input.

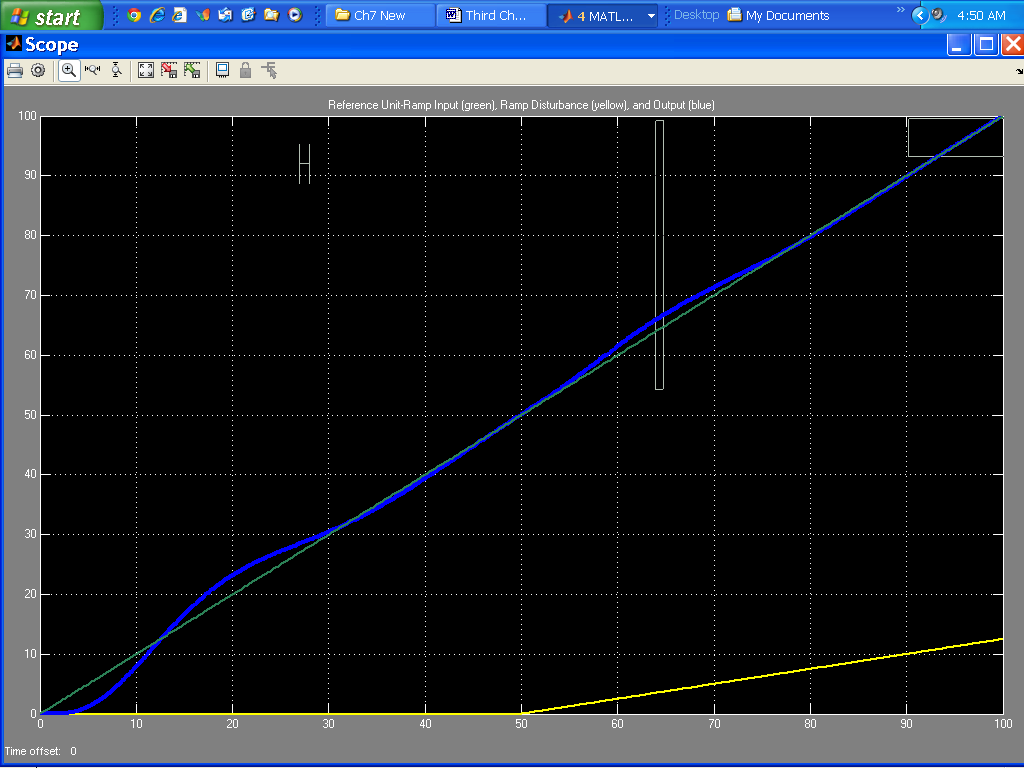


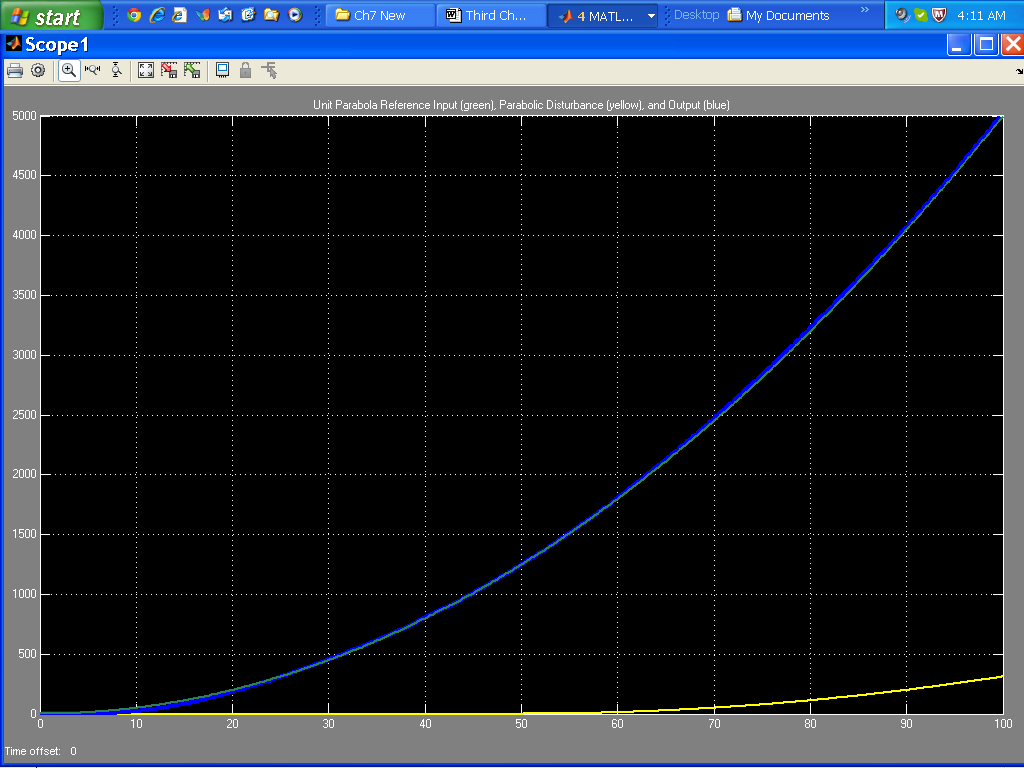


**45.**

The modified Simulink models of this system and their responses (displayed in blue) from t = 0 to 100 seconds are shown below. In the top model, the reference input, *r*(*t*), is a unit ramp, *t u*(*t*), applied at t = 0 (displayed in green), and the disturbance is *d(t)* = 0*.*25 *t u*(*t*) applied at t = 50 sec (displayed in yellow). As could be seen from this plot, the steady-state position errors due to the unit-ramp reference input and the 0*.*25 *t u*(*t*) disturbance ramp are equal to zero.







**SOLUTIONS TO DESIGN PROBLEMS**

**46.**

The force error is the actuating signal. The equivalent forward-path transfer function is . The feedback is . Using Eq. (7.72) . Applying the final value theorem, . Thus, *K*2 < 0.1*Ke.* Since the closed-loop system is second-order with positive coefficients, the system is always stable.

**47.**

**a.** The minimum steady-state error occurs for a maximum setting of gain, *K*. The maximum *K* possible is determined by the maximum gain for stability. The block diagram for the system is shown below.



Pushing the input transducer to the right past the summing junction and finding the closed-loop transfer function, we get



Forming a Routh table,

|  |  |  |
| --- | --- | --- |
| s3 | 1 | 50 |
| s2 | 14 | 3*K*+100 |
| s1 |  | 0 |
| s0 | 3*K*+100 | 0 |

The s1 row says -∞ < K < 200. The s0 row says  < *K.* Thus for stability,

 < *K* < 200. Hence, the maximum value of *K* is 200.

**b.** . Hence, **.**

**c.** Step input.

**48.**

1. The open loop transmission is . The system is type 2.
2. The Transfer function from disturbance to error signal is



Using the Final value theorem



1. We calculate  so . So we get 
2. The system’s characteristic equation is  or

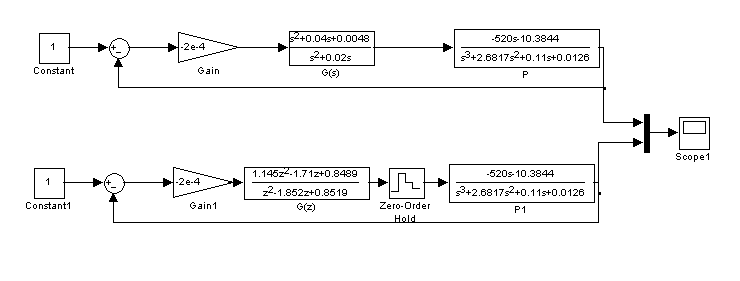
. The Routh array is:

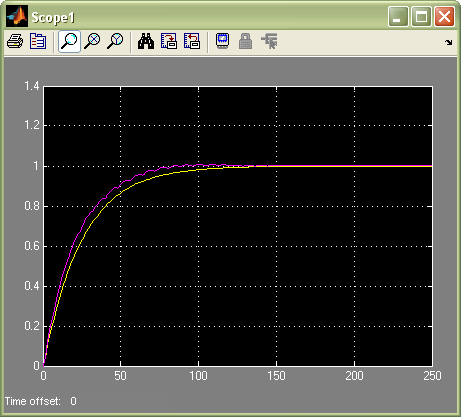
|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
| 1 |  |  |

The dominant requirement is given by the third row 

**49.**

1. A bode plot of the open loop transmission shows that the open loop transfer function has a crossover frequency of . A convenient range for sampling periods is . T=8 days fall within range.
2. We substitute into we get 





**50.**

**a.** When the speed controller is configured as a proportional controller, the forward-path transfer function of this system is:

 (1)

For the steady-state error for a unit-step input, *r(t)* = *u(t*), to be equal to 1%:

 (2)

From equation (2), we get: , which yields: *KP*SC = 85.9.

**b.** When the speed controller is configured as a proportional plus integral controller, the forward-path transfer function of the system becomes:

 (3)

For the steady-state error for a unit-ramp input, *r(t)* = *t u(t*), to be equal to 2.5%:

 (4)

From equation (4), we get: , which yields: *KI*SC = 34.7.

**c.** We’ll start by finding *G1(s)*, the equivalent transfer function of the parallel combination, representing the torque and speed controllers, shown in Figure P7.35:

 (5)

Given that the equivalent transfer function of the car is: , we apply equation 7.62\* of the text taking into consideration that the disturbance here is a step with a magnitude equal to 83.7:



**51.**

1. The system is type 0 so . It follows that . So . From Problem 76, Chapter 6, the system is closed-loop stable for *K* < 9.63, so this steady-state error is not achievable.
2. Due to stability constraints, the minimum steady state error for a unit step input is achievable when *K* = 9.63. or 12.92%
3. For zero steady-state error to a step input, the system must be augmented to Type 1. The simplest compensator that can be used to achieve this is an integrator, namely .