EE4302 Lab 1 Report: Computer-Aided Design of a State-Space Control System

Name: Phang Swee King
Matric No: U066584J
Date: 18th February 2009

1.0 Objective

- Design of a state-feedback controller is considered.
- Desired closed-loop response is given by specifications in the frequency domain.
- A computer-aided design procedure is used to interactively achieve the specifications.

2.0 Equipment

- PCs in the Control & Simulation Laboratory, E4A level3, ECE Dept.
- MATLAB software package.

3.0 Introduction

The use of state-space ideas in control system design allows the control engineer great flexibility in shaping the dynamic response of systems. State-space techniques are powerful and also provide useful insight into the structure of the system under study. However, the calculations involved in design using state-space ideas are often very tedious, and have been one of the drawbacks to widespread use of these techniques. Further, specifications for the desired closed-loop system are often given in the frequency domain (typically bandwidth requirements), and a control engineer using state-space techniques must be able to interactively relate the choice of the state-feedback gains to resulting frequency domain plots.

In recent years, software packages to assist the control engineer have become available, and fairly comprehensive ones are now available for the personal computer environment. In the experiment, we will use the MATLAB package.

3.1 Review of State-Variables and MATLAB

Consider the following plant described in the state-space notation,

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -1.08x_1 - 2.33x_2 + u \\ y &= x_1\end{aligned}$$

The state-variables are $x_1(t)$ and $x_2(t)$; $u(t)$ is the input to the system while $y(t)$ is the output of the system. Both state variables are measurable.

Rewrite the plant in state representation matrices, we have

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -1.08 & -2.33 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$
$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x + [0] u$$

Suppose

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

We are interested to find out the frequency responses of this open-loop system.

The plots of the frequency response from u to x_1 (refer to Appendix A-1) and from u to x_2 (refer to Appendix A-2) can be obtained by using the MATLAB function BODE as shown below:

```
% 3.1 Review of State-Variables and MATLAB
A1 = [0 1;-1.08 -2.33]; B1 = [0;1]; C1 = [1 0]; D1 = 0;
% frequency response plot from u to x1
sys1 = ss(A1,B1,C1,D1); figure(1); bode(sys1);
% frequency response plot from u to x2
% change C2 matrix to [0 1]
C2 = [0 1]; sys2 = ss(A1,B1,C2,D1); figure(2); bode(sys2);
```

Consider the feedback law

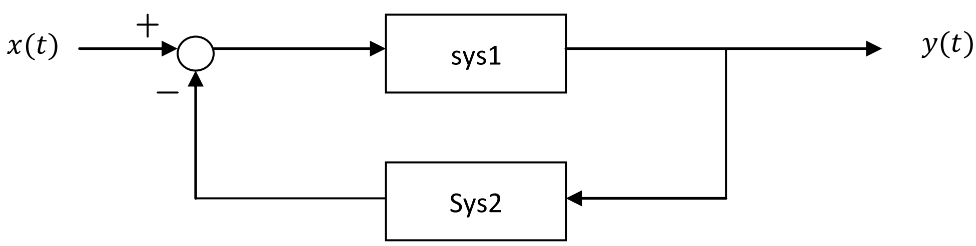
$$u = -k_1 x_1 - k_2 x_2$$

for the particular values of $k_1 = 0.1$ and $k_2 = 0.1$.

The MATLAB function FEEDBACK allows us to obtain the closed-loop system with the following command:

```
sys = feedback(sys1,sys2);
```

It is easier to visualize it with the following diagram:



The frequency response of the closed-loop system (refer to Appendix A-3) can be obtained with the following MATLAB codes:

```
% feedback of the closed loop system
% the output to the control is both x1 and x2, hence C3 = [1 0;0 1]
C3 = [1 0;0 1]; sys3 = ss(A1,B1,C3,D1);
% choosing k1 = 0.1, k2 = 0.1
k1 = 0.1; k2 = 0.1; K = [k1,k2];
A2 = [0 0;0 0]; B2 = [0 0;0 0]; C2 = [0 0]; D2 = K;
sys4 = ss(A2,B2,C2,D2); sys5 = feedback(sys3,sys4);
% feedback response from u to x1 and x2
figure(3); bode(sys5);
```

4.0 Simple State-Feedback Design

Consider the plant described in the previous section. It is desired to design a closed-loop system (Fig. 1) such that the frequency response from the commanded signal, r, to the output, y, has the following specifications:

- Closed-loop bandwidth: Not lower than 5.0 rad/s
- Resonant Peak, M_r : Not larger than 2 dB
- Steady-state gain between r and y: 0 dB.

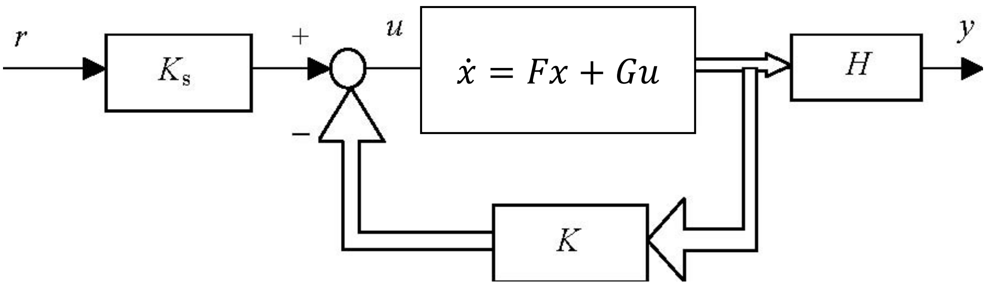


Figure 1

4.1 Design Using Ackermann's Formula

Based on the closed-loop specifications given above, and the relation between location of poles and bandwidth, the desired closed-loop pole locations can be chosen using 3 different methodologies. They are:

1. ITAE Prototype
2. Bessel Prototype
3. Second-Order Dominant Response

After choosing pole locations, the state-feedback gains are calculated by Ackermann's formula to place the poles of the closed-loop at the desired locations.

The MATLAB function ACKER allows us to obtain the state-feedback gains using Ackermann's formula with the following command:

```
K = acker(F,G,P);
```

where P is the pole locations writing in column vector's form.

The scaling gain, K_s is included to give a steady state gain of 1. The steady state is given at the low frequency region of the frequency response of a system. Hence, the following calculation allows us to have a steady state gain of 1 by choosing the appropriate value for K_s:

$$1 = K_s(\text{DC gain of system})$$
$$K_s = \frac{1}{\text{DC gain of system}}$$

MATLAB function DCGAIN will obtain the steady state gain of the system by the following command:

```
K = dcgain(sys);
```

4.1.1 ITAE Prototype Design

Pole locations can be chosen by referring to ITAE Prototype Table as shown in figure 2.

TABLE 6.1
Prototype Response Poles

(a) ITAE transfer functions	k	Pole locations for $\omega_0 = 1$ rad / s[†]
	1	$s + 1$
	2	$s + 0.7071 \pm j0.7071$[‡]
	3	$(s + 0.7081)(s + 0.5210 \pm j1.068)$
	4	$(s + 0.4240 \pm j1.2630)(s + 0.6260 \pm j0.4141)$
	5	$(s + 0.8955)(s + 0.3764 \pm j1.2920)(s + 0.5758 \pm j0.5339)$
	6	$(s + 0.3099 \pm j1.2634)(s + 0.5805 \pm j0.7828)(s + 0.7346 \pm j0.2873)$
(b) Bessel transfer functions	**k**	**Pole locations for $\omega_0 = 1$ rad / s[†]**
	1	$s + 1$
	2	$s + 0.8660 \pm j0.5000)$[‡]
	3	$(s + 0.9420)(s + 0.7455 \pm j0.7112)$
	4	$(s + 0.6573 \pm j0.8302)(s + 0.9047 \pm j0.2711)$
	5	$(s + 0.9264)(s + 0.5906 \pm j0.9072)(s + 0.8516 \pm j0.4427)$
	6	$(s + 0.5385 \pm j0.9617)(s + 0.7998 \pm j0.5622)(s + 0.9093 \pm j0.1856)$

[†]Pole locations for other values of ω_0 can be obtained by substituting s/ω_0 for s everywhere.
[‡]The factors $(s + a + jb)(s + a - jb)$ are written as $(s + a \pm jb)$ to conserve space.

Figure 2

Since the system is a second order system, the pole locations of the system are at $s = -0.7071 \pm j0.7071$ for $\omega_0 = 1$ rad/s

The following MATLAB codes provide the frequency response and step response of the system given the pole locations:

```
% w_0 is varied to satisfy the specifications
w_0 = 1;
P1 = w_0*[-0.7071+0.7071*i;-0.7071-0.7071*i];
% pole placement using Ackermann's Formula
K = acker(A1,B1,P1);
D2 = K;
sys6 = ss(A2,B2,C2,D2);
sys7 = feedback(sys3,sys6);
Ks = 1/dcgain(sys7);
figure(4);
bode(Ks*sys7);
% corresponding step response
figure(5);
step(Ks*sys7);
```

With $\omega_0 = 5$ rad/s, the system failed the given specifications as the closed-loop bandwidth is lower than 5 rad/s (refer to Appendix B-1). Hence, $\omega_0 = 6$ rad/s is used in our ITAE Prototype design (refer to Appendix B-2 for frequency response and step response of the system).

4.1.2 Bessel Prototype Design

Pole locations can be chosen by referring to Bessel Prototype Table as shown in figure 2 above.

Since the system is a second order system, the pole locations of the system are at
$s = -0.8660 \pm j0.5000$ for $\omega_0 = 1$ rad/s

Using the same MATLAB codes by replacing pole locations to:

```
P = w_0*[-0.8660+0.5000*i;-0.8660-0.5000*i];
```

We could obtain the frequency response and step response of the system. After trial of $\omega_0 = 6$ rad/s proves to be unsatisfactory (refer to Appendix B-3), $\omega_0 = 7$ rad/s is used in our final Bessel Prototype design (refer to Appendix B-4 for frequency response and step response of the system).

4.1.3 Second-Order Dominant Response Design

The second-order dominant response allows us to place two poles at the appropriate low frequency such that it dominates the effect of all of the higher frequency poles. Hence a higher order system would behave like a second order system with their two dominating poles. This design is usually used to prevent oscillation in certain application.

First I have chosen the following poles such that they are at low frequencies:

```
P = 6*[-1;-1];
```

However, the frequency response of this system shows that it fails to satisfied the given specification with its small closed-loop bandwidth (refer to Appendix B-5).

Next up, I approximate the system to a first order system with closed-loop bandwidth of 6 rad/s by setting a pole at $s = -6$ and a fast pole:

```
P = [-6;-100];
```

The frequency response of the system satisfied the requirement with a bandwidth of 6 rad/s as predicted. The step response of this system shows no oscillations before it reaches steady state, like a typical first order system (refer to Appendix B-6).

Next, from the typical second order system's equation, the pole locations is given at

$$s = -\zeta\omega_0 \pm \omega_0\sqrt{\zeta^2 - 1}$$

By choosing $\zeta = 0.7071$ (under damping) and $\omega_0 = 6$ rad/s, we obtained the following poles:

```
P = [-4.242+4.242*i;-4.242-4.242*i];
```

It has a bandwidth of 6 rad/s which satisfied the condition. Its step response will oscillate before it reaches the steady state as predicted since the damping ratio is below the value of 1 (refer to Appendix B-7).

Lastly, by choosing $\zeta = 0.5$ (under damping), we obtained the following poles:

```
P = [-3+5.196*i;-3-5.196*i];
```

It satisfied the specifications with some minor changes compare to the previous design. The system has a resonant peak of 1.09dB at 4.83 rad/s (refer to Appendix B-8). The resonant frequency can also be calculated using the following equation:

$$\omega_r = \omega_n\sqrt{1 - 2\zeta^2}\ , \quad \zeta < 0.7071$$

Besides that, since the damping ratio is lower than the previous design, the step response oscillates more before it reaches steady state value compare to the previous design which can be seen clearly in the step response graph.

4.2 Design Using Linear Quadratic Regulator Weightings

The design using Ackermann's formula focuses on choosing precisely the desired closed-loop pole locations. In a system where the dimension is large such a procedure may be problematic as it is not always clear where exactly to place all the closed-loop poles. The Linear Quadratic Regulator (LQR) method adopts a different approach. It focuses on calculating a set of gains to minimize the criterion

$$J = \int_0^\infty (x^T Q x + Ru^2)dt$$

This method requires use of a control system design package like MATLAB using the function

```
[K,S,E] = lqr(F,G,Q,R);
```

where Q and R are the weighting factors.

After choosing the weighting gains, we could use the following MATLAB codes to obtain the frequency response and step response of the system.

```
% Design Using Linear Quadratic Regulator Weightings %
% set everything to gain 1
R = 1; Q = [1 0;0 1];
[K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys18 = ss(A2,B2,C2,D2); sys19 = feedback(sys3,sys18);
Ks = 1/dcgain(sys19);
figure(16); bode(Ks*sys19); figure(17); step(Ks*sys19);
```

The first approach was to set all gains to 1. However, this set of values do not satisfied the specifications (refer to Appendix C-1). Since the error in x_1 is more important to us, the weighting gain for x_1 is increased from 1 to 100, 200 and finally the system satisfied the specifications at the gain of 1000 for x_1 (refer to Appendix C-2, C-3, and C-4).

Next, I have tried to vary the weighting gain of the input u and x_1 to investigate the frequency response of different weighting gain using LQR. Refer to Appendix C for all the frequency and step response.

5.0 State-Feedback Design Including State-Augmentation

In every system, there is possibility of disturbance to the system. The previous state-feedback design using prototype and LQR will not reject a persistent disturbance. Hence, the introduction of integral control will automatically take care of steady state reference signal tracking, and to eliminate disturbance.

Consider the situation in Figure 3. The plant has the state-variable description given in the previous section.

$$
\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -1.08x_1 - 2.33x_2 + u + v \\
y &= x_1
\end{aligned}
$$

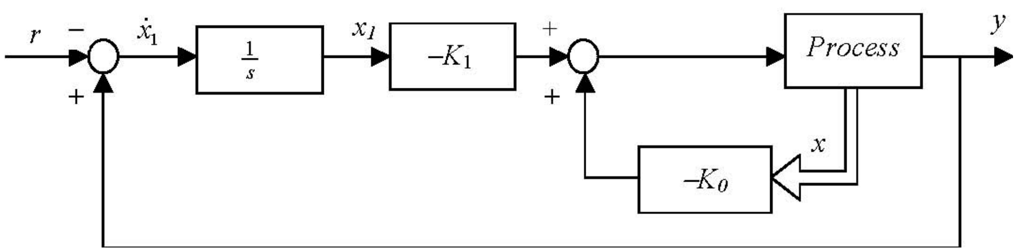


Figure 3

It is now desired to obtain the following frequency specifications between r and y:

- Closed-loop bandwidth: Not lower than 3.5 rad/s
- Resonant Peak, M_r: Not larger than 2 dB
- Steady-state gain between r and y: 0 dB

In addition, the response in y to a step in the unmeasurable disturbance v should be zero in the steady-state.

The augmentation state representation matrices can be written as follow:

$$\begin{bmatrix} \dot{x}_I \\ \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -1.08 & -2.33 \end{bmatrix} \begin{bmatrix} x_I \\ x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} r + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} v$$

$$y = x_1 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_I \\ x_1 \\ x_2 \end{bmatrix}$$

The following MATLAB codes are written to initialize the matrices:

```
F = [0 1 0;0 0 1;0 -1.08 -2.33]; G = [0;0;1]; Gr = [-1;0;0];
Gv = [0;0;1]; H = [0 1 0]; J = 0;
```

5.1 Design Using Ackermann's Formula

The same approaches will be done in this part using

1. ITAE Prototype
2. Bessel Prototype
3. Second-Order Dominant Response

The following MATLAB codes obtained the frequency and step response of the system, the step response of output to the disturbance v and the step response of the control input to the disturbance v :

```
P1 = w_0*[-0.7081;-0.5210+1.068*i;-0.5210-1.068*i]; K =
acker(F,G,P1);
% response of output y to a unit step in r
% assuming v = 0
sys42 = ss(F-G*K,Gr,H,J); figure(40); bode(sys42); figure(41);
step(sys42);
% response of output y to a unit step in the unmeasurable disturbance
v
sys43 = ss(F-G*K,Gv,H,J); figure(42); step(sys43);
% response of input u to a unit step in the unmeasurable disturbance
v
H1 = -K; sys44 = ss(F-G*K,Gv,H1,J); figure(43); step(sys44);
```

5.1.1 ITAE Prototype Design

From the ITAE prototype table in Figure 2 above, the pole location for 3rd order system are:

```
P = w_0*[-0.7081;-0.5210+1.068*i;-0.5210-1.068*i];
```

After trial and error by increasing ω_0 for each trial, $\omega_0 = 4$ rad/s proved to satisfy the specifications (refer to Appendix D-1):.

5.1.2 Bessel Prototype Design

From the Bessel prototype table in Figure 2 above, the pole location for 3rd order system are:

```
P = w_0*[-0.9420;-0.7455+0.7112*i;-0.7455-0.7112*i];
```

After trial and error by increasing ω_0 for each trial, $\omega_0 = 6$ rad/s proved to satisfy the specifications (refer to Appendix D-2).

5.1.3 Second-Order Dominant Response Design

I approximate a first order system by choosing a slow pole together with 2 fast poles for the system as shown below. The step response is expected to behave similar to a first order system (refer to Appendix D-3):

```
P3 = [-4;-100;-100];
```

Next, I approximate the system to a second order under damping system by assigning the damping ratio with a value of 0.7071. The third pole is chosen such that it will not affect the response of the system as shown below (refer to Appendix D-4 for response plots):

```
P4 = [-2.828+2.828*i;-2.828-2.828*i;-100];
```

Finally, I decrease the damping ratio to 0.5000 and expecting a step response with more oscillation (refer to Appendix D-5). The pole locations are as follow:

```
P5 = [-2+3.464*i;-2-3.464*i;-100];
```

5.2 Design Using Linear Quadratic Regulator Weightings

Similar to Part 4.2 above, a set of weighting gains were chosen such that the system satisfy the specifications.

The following MATLAB codes provide us varies responses of the system:

```
% LQR Design
% let everything start from 1
R = 1; Q = [1 0 0;0 1 0;0 0 1]; [K,S,E] = lqr(F,G,Q,R);
% response of output y to a unit step in r
sys48 = ss(F-G*K,Gr,H,J); figure(48); bode(sys48); figure(49);
step(sys48);
% response of output y to a unit step in the unmeasurable disturbance
v
sys49 = ss(F-G*K,Gv,H,J); figure(50); step(sys49);
% response of input u to a unit step in the unmeasurable disturbance
v
H1 = -K; sys50 = ss(F-G*K,Gv,H1,J); figure(51); step(sys50);
```

For the starting, unit gain is used for all the weighting. Clearly the design does not meet the specifications (refer to Appendix E-1).

Gain for x_I is increased since we are only interested in this signal. After trial and error, the following gains satisfied the specifications (refer to Appendix E-2, E-3, E-4):

```
R = 1; Q = [3000 0 0;0 1 0;0 0 1];
```

Next we investigate the effect of weighting more heavily on x_1 instead (refer to Appendix E-5):

```
R = 1; Q = [1 0 0;0 1000 0;0 0 1];
```

Then we investigate the effect of weighting more heavily on x_1 (refer to Appendix E-6):

```
R = 1; Q = [1 0 0;0 1 0;0 0 1000];
```

Finally we investigate the effect of weighting more heavily on u (refer to Appendix E-7):

```
R = 30; Q = [1 0 0;0 1 0;0 0 1];
```

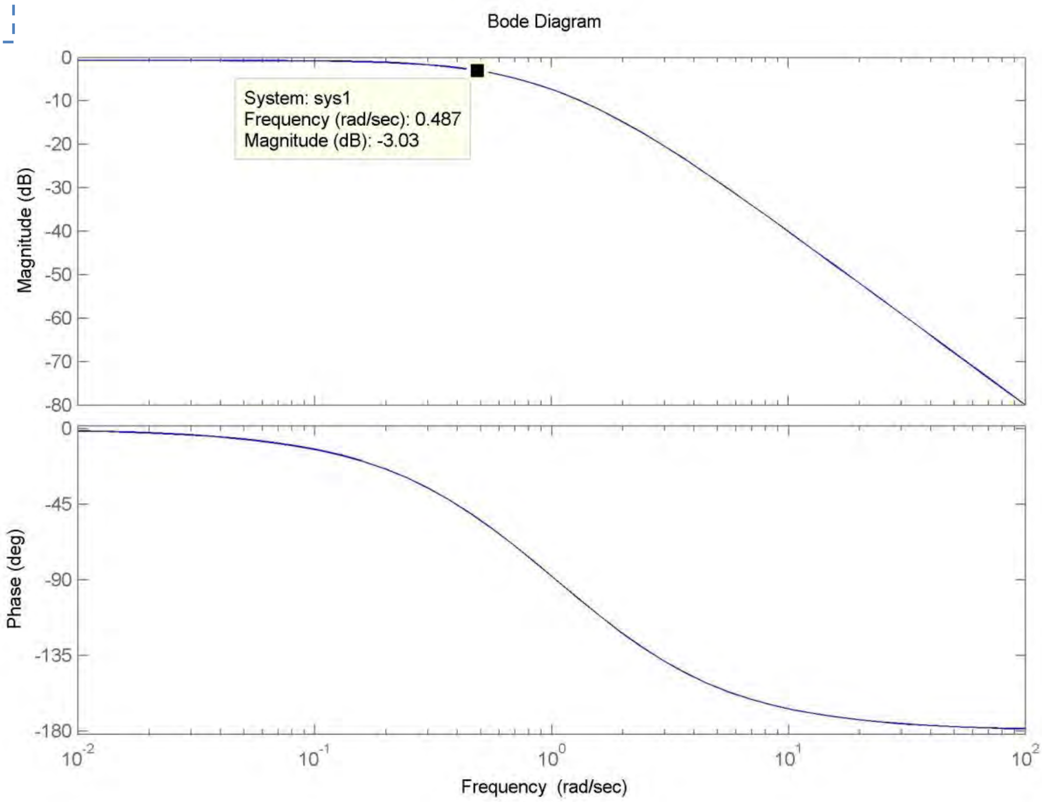
6.0 Conclusion

- Given the closed-loop specifications that is required of the system, the required closed-loop characteristics can be analyzed and evaluated using several approaches, with the aid of computer-based design software such as Matlab.
- The ITAE and LQR methods have been utilized in this experiment. Both can yield the required results in this experiment. The ITAE method has the advantage of simplicity as we are only required to choose the pole location from the ITAE table. The LQR method offers additional flexibility in the design procedure as it allows us to decide on the relative importance of the state-variables and control effort involved.
- Integral control can be introduced to obtain steady-state tracking of a step reference-signal by augmenting the plant state. This design adds the capability of rejecting persistent constant disturbances.

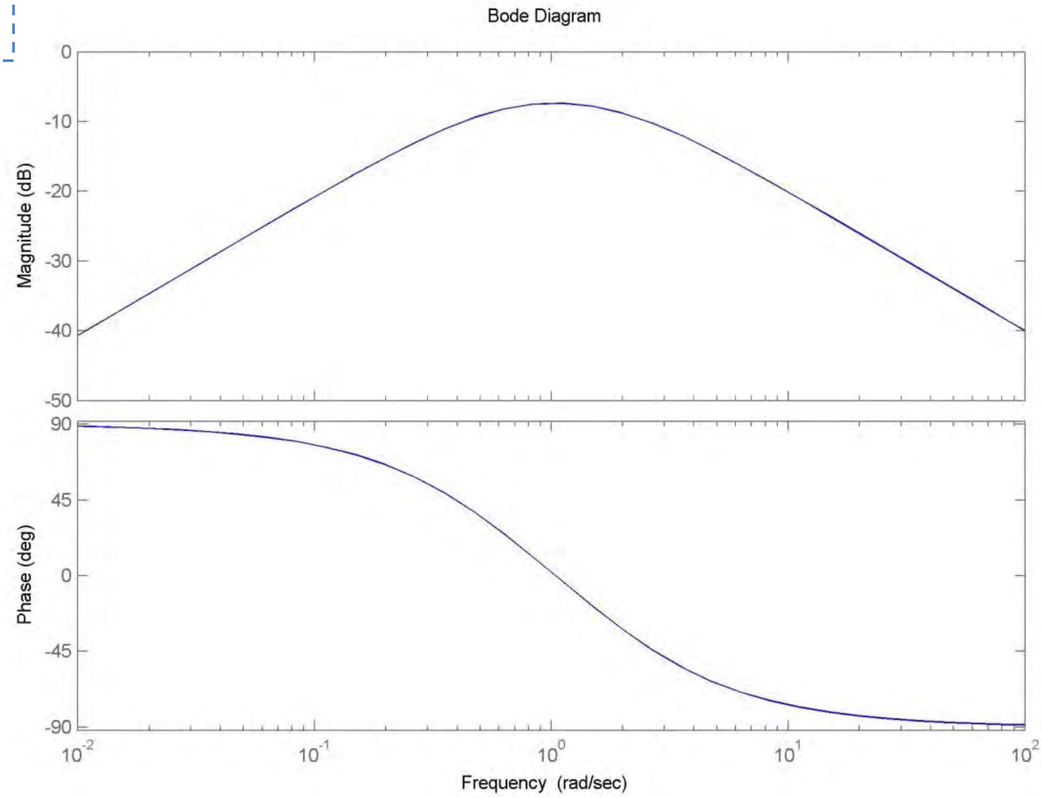
Appendix A

PHANG Swee King
U066584J

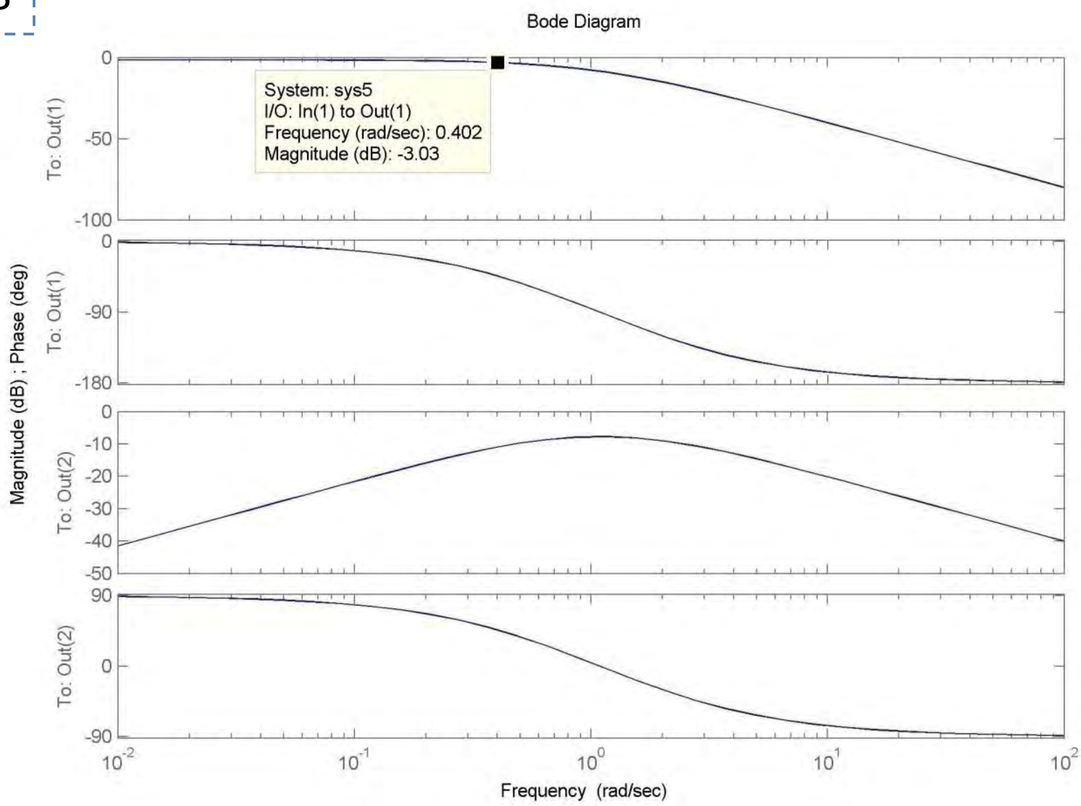
A-1



A-2



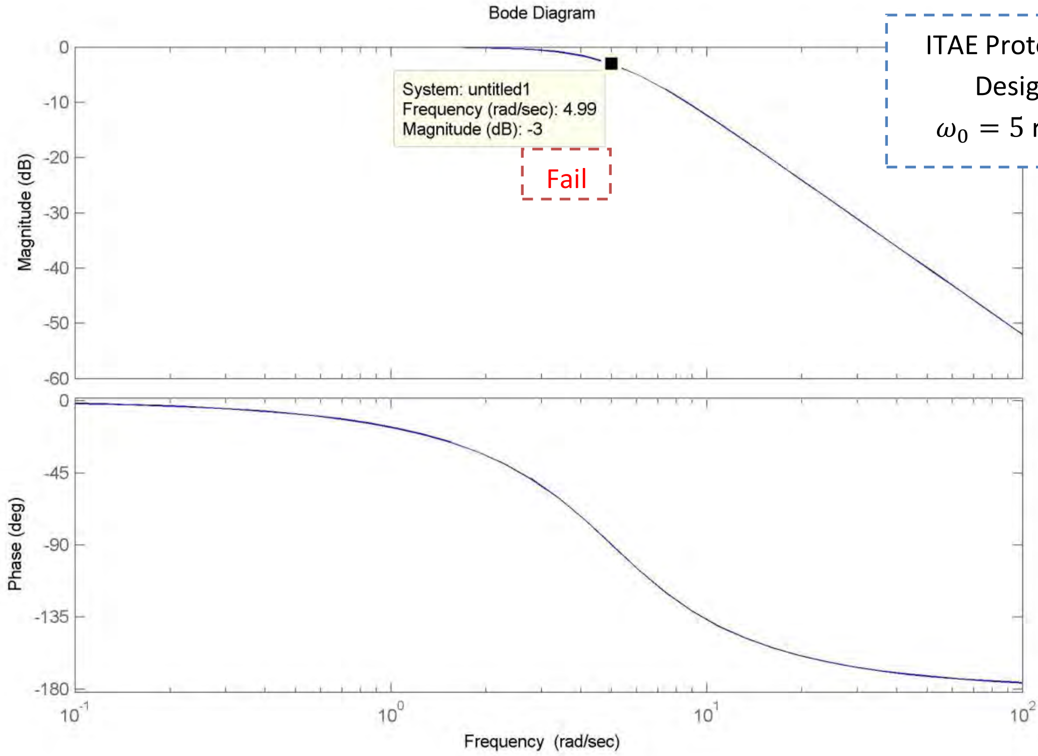
A-3


Bode Diagram
System: sys5
I/O: In(1) to Out(1)
Frequency (rad/sec): 0.402
Magnitude (dB): -3.03
Magnitude (dB) ; Phase (deg)
To: Out(1)
To: Out(2)
Frequency (rad/sec)

Appendix B

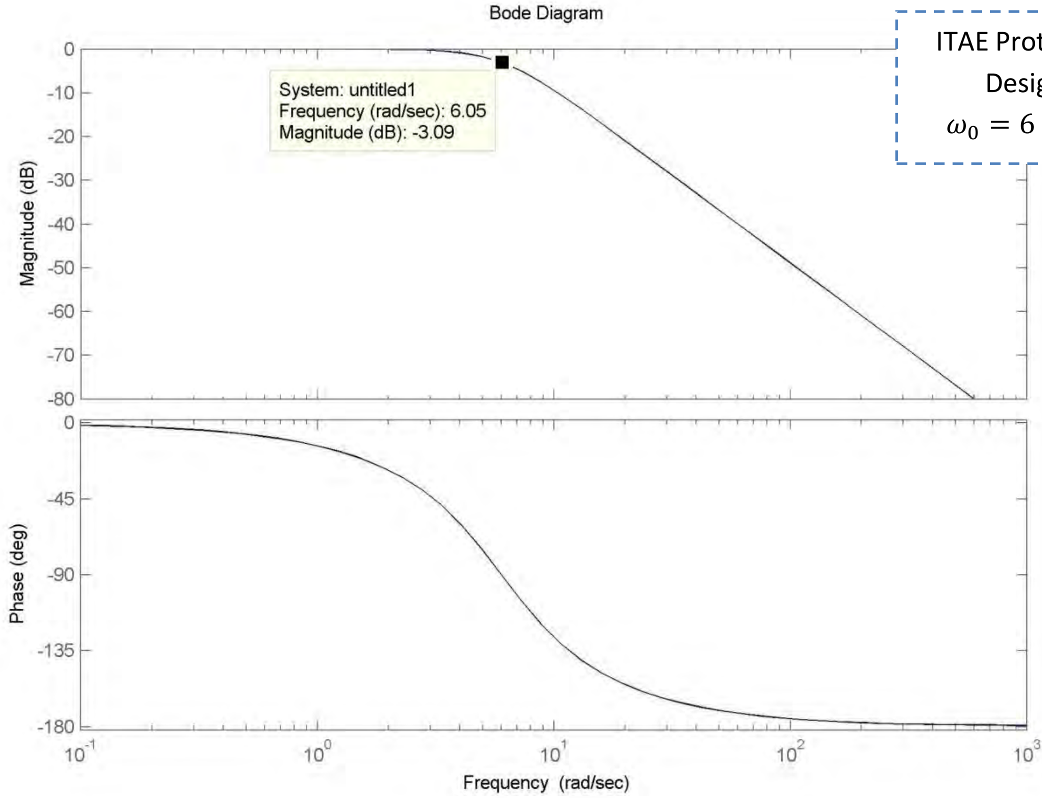
B-1

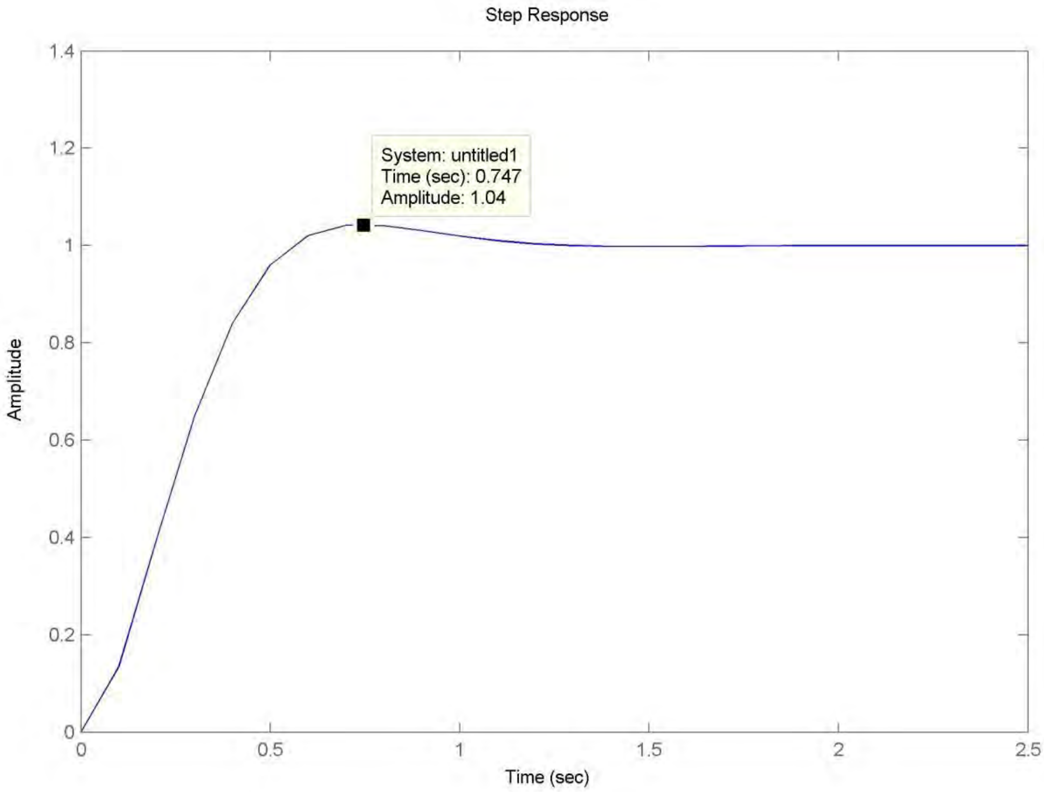
ITAE Prototype Design
$\omega_0 = 5$ rad/s



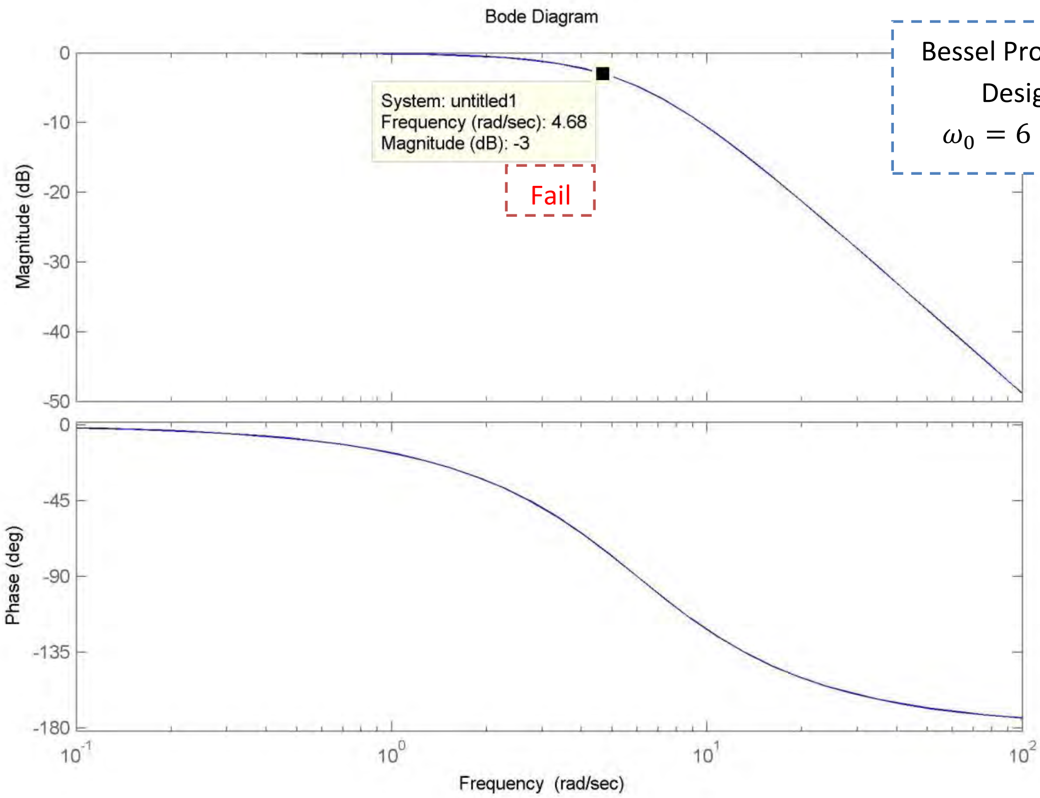
B-2

ITAE Prototype Design
$\omega_0 = 6$ rad/s



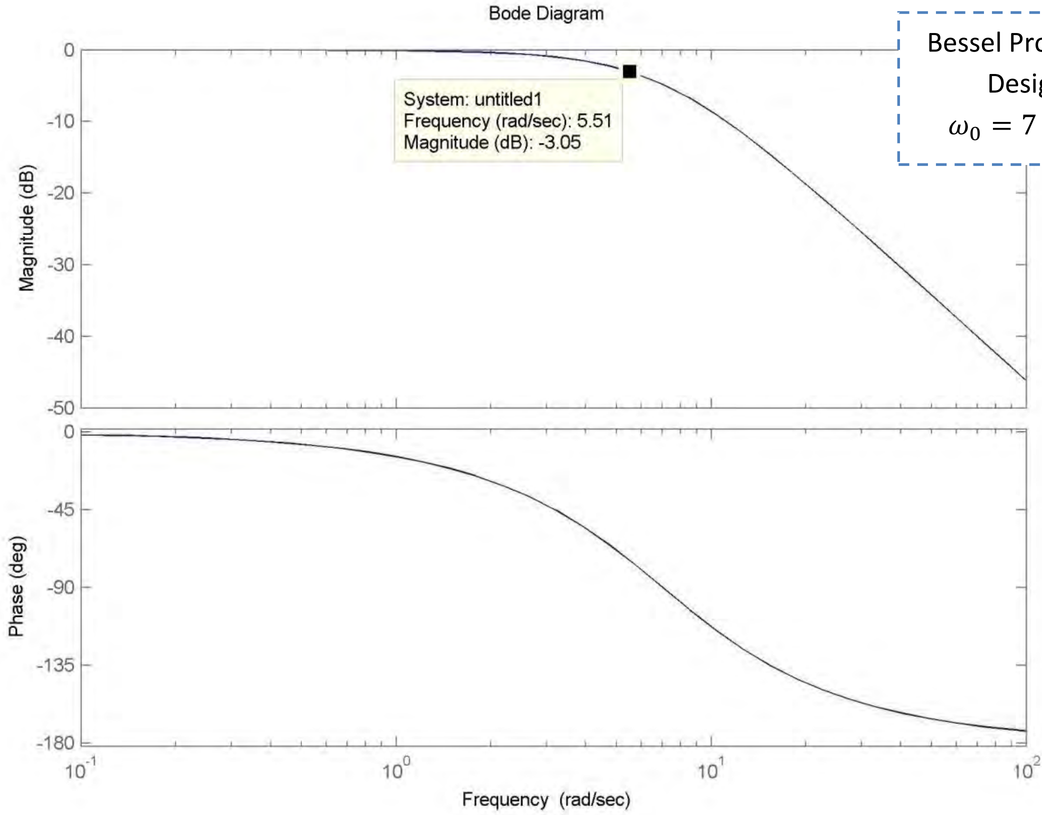


B-3

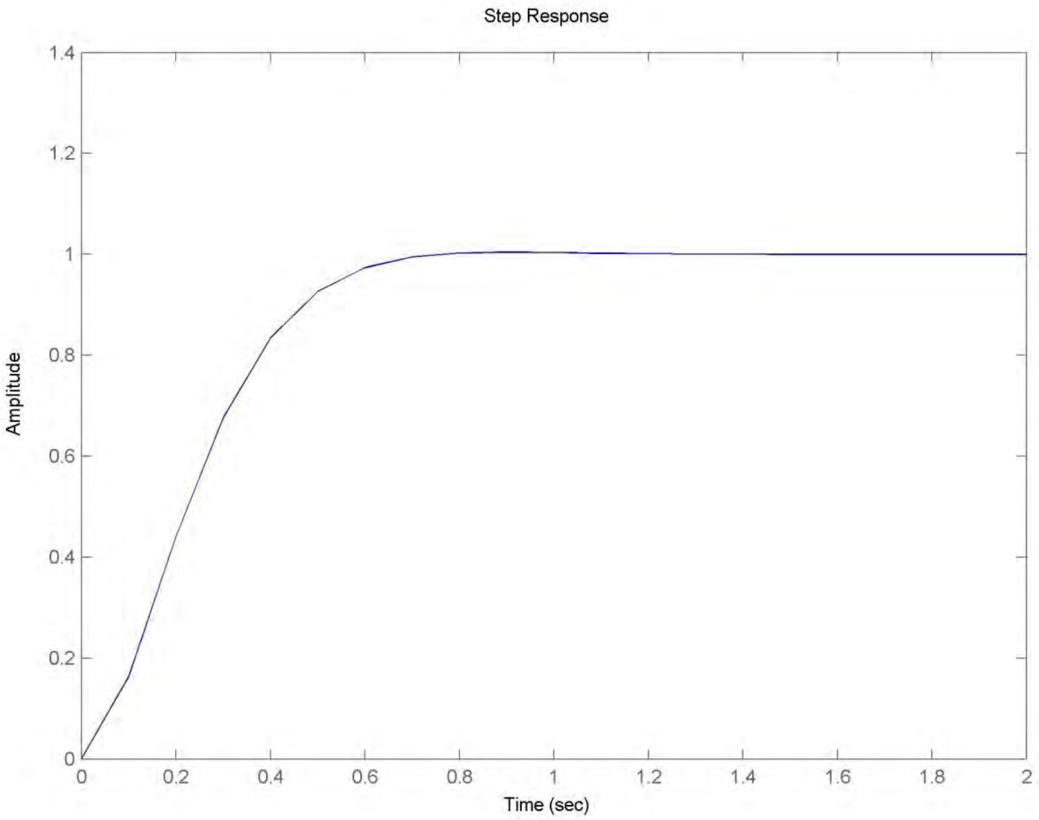


Bessel Prototype Design
$\omega_0 = 6$ rad/s

B-4



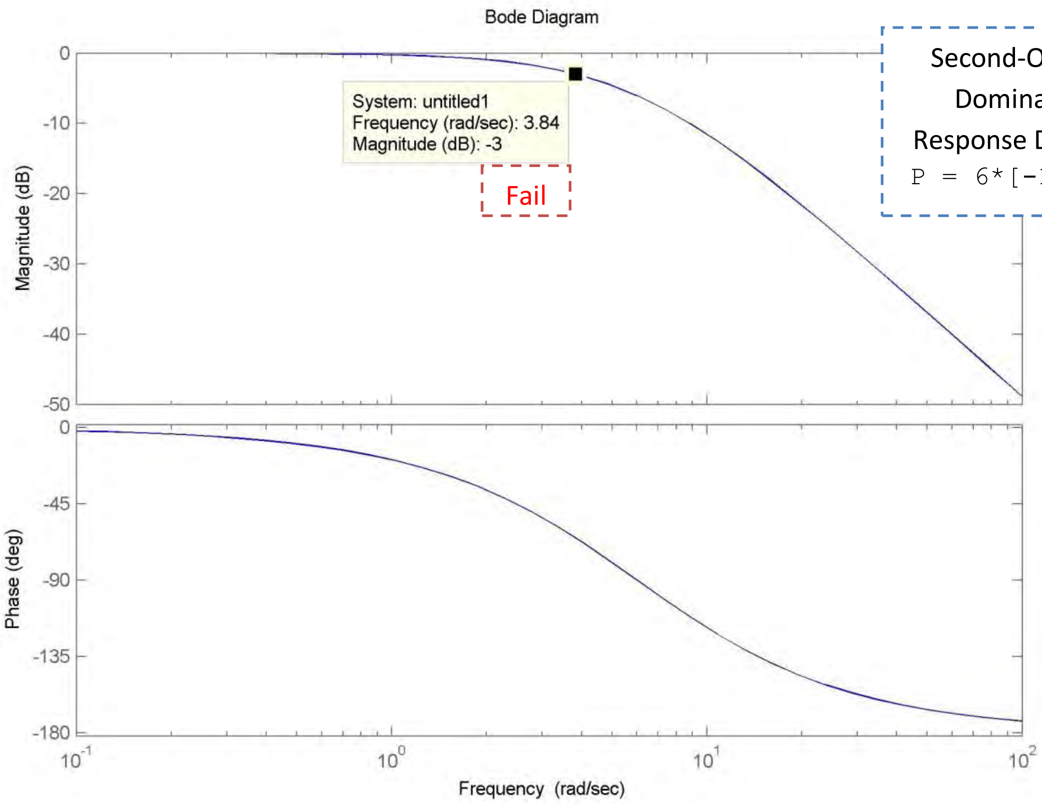
Bessel Prototype Design
$\omega_0 = 7$ rad/s

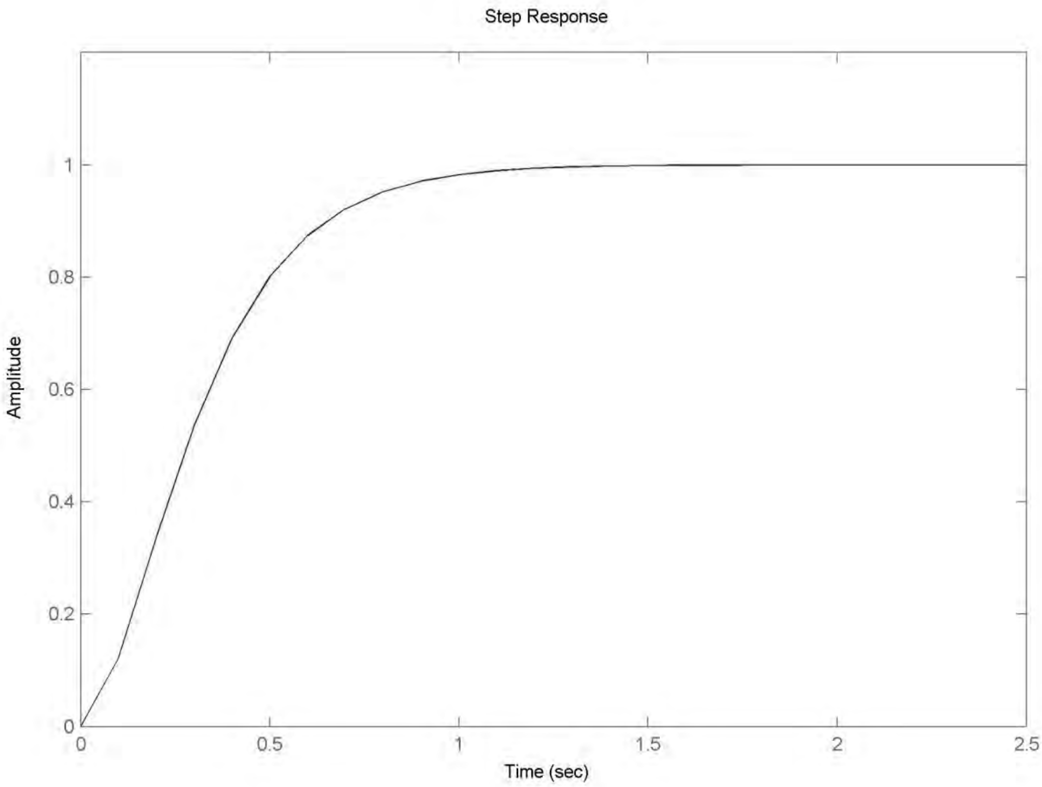


PHANG Swee King
U066584J

B-5

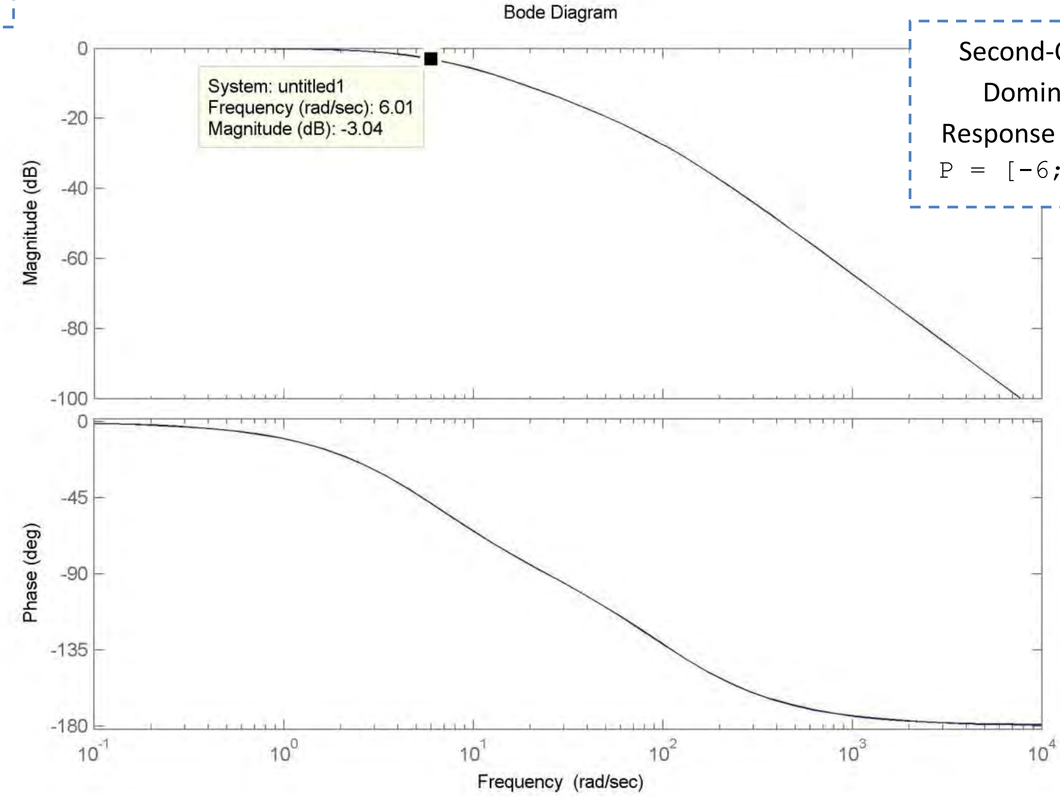
Second-Order Dominant Response Design
`P = 6*[-1;-1]`

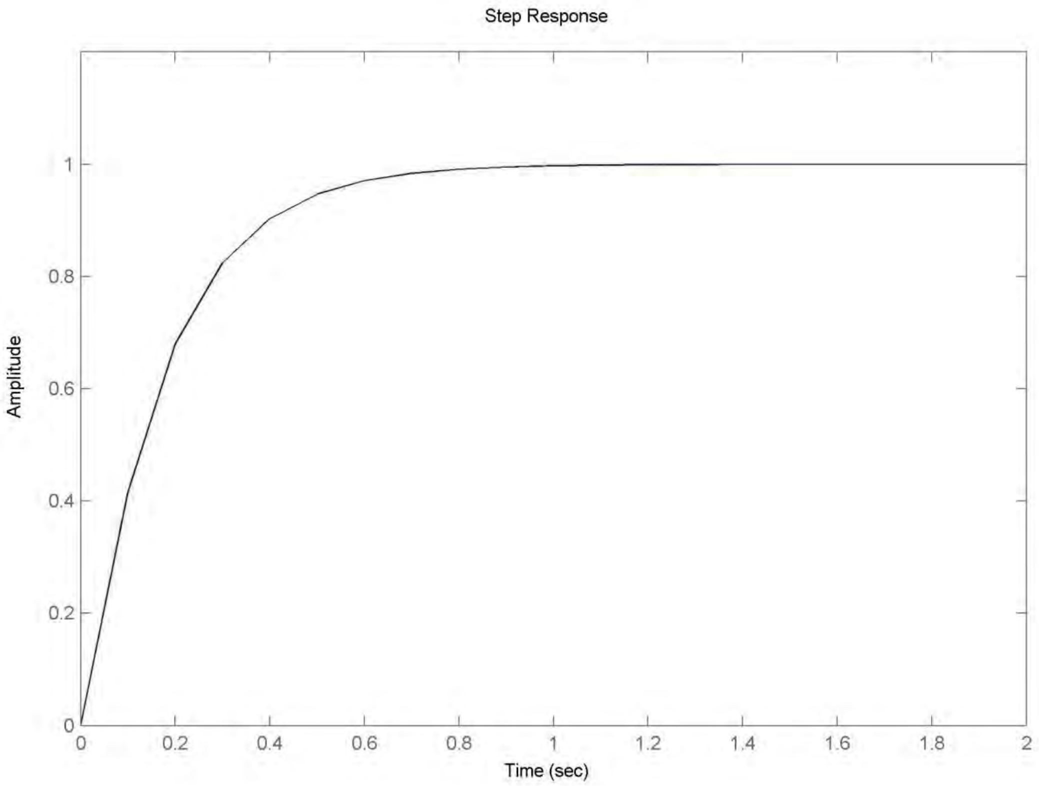




B-6

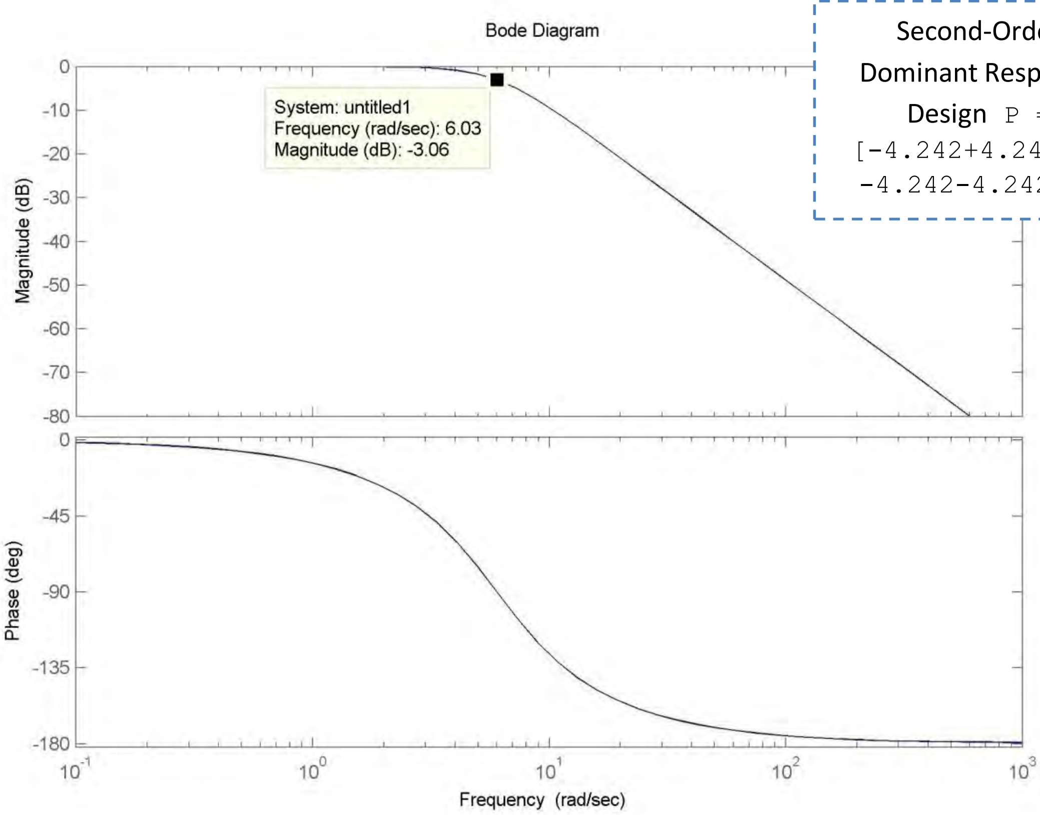
Second-Order Dominant Response Design
`P = [-6;-100]`

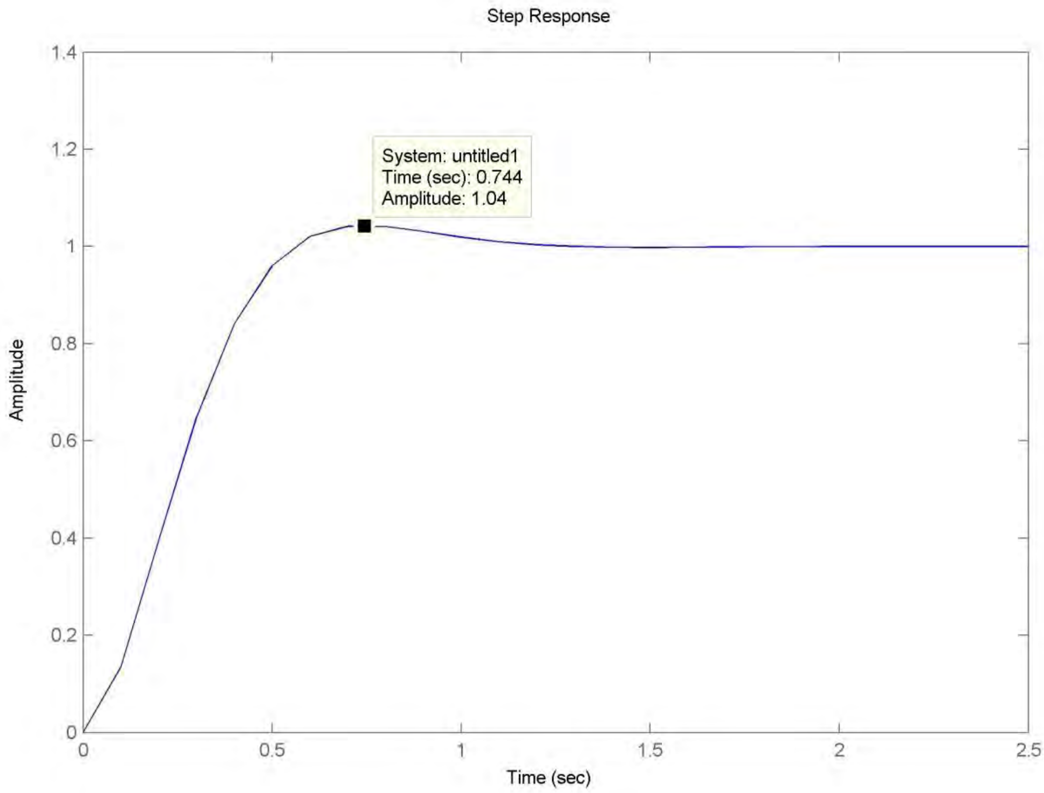




B-7

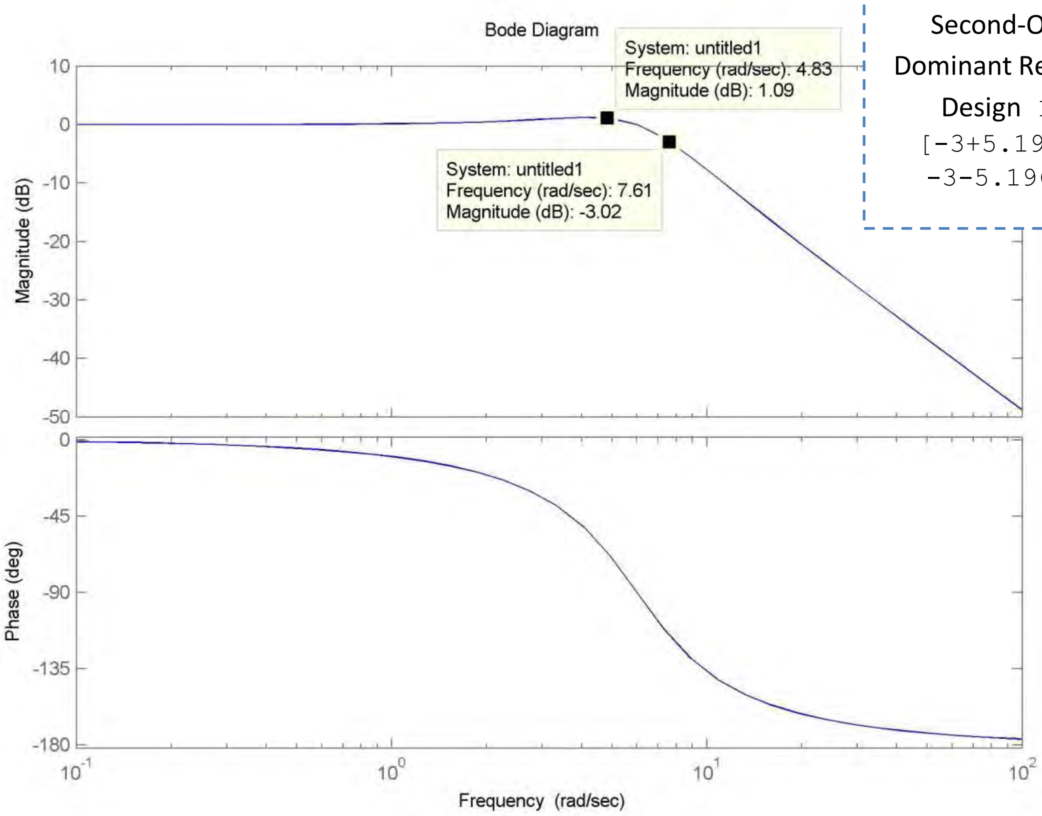
Second-Order Dominant Response Design `P = [-4.242+4.242*i; -4.242-4.242*i]`

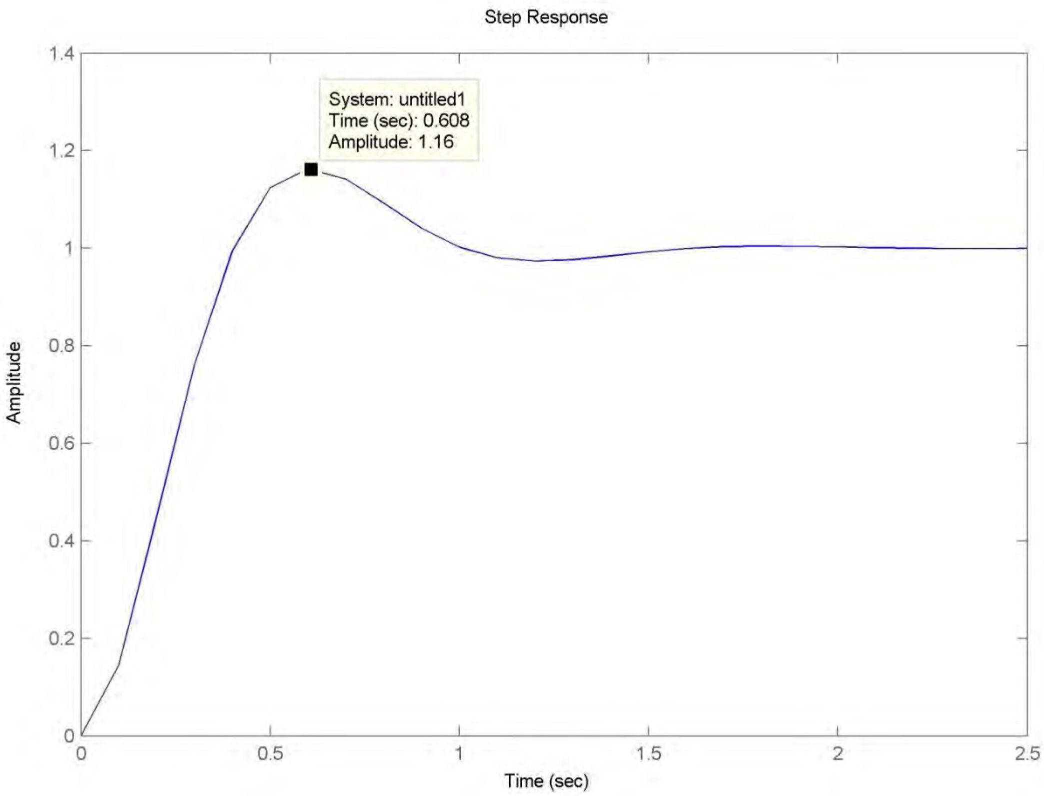




B-8

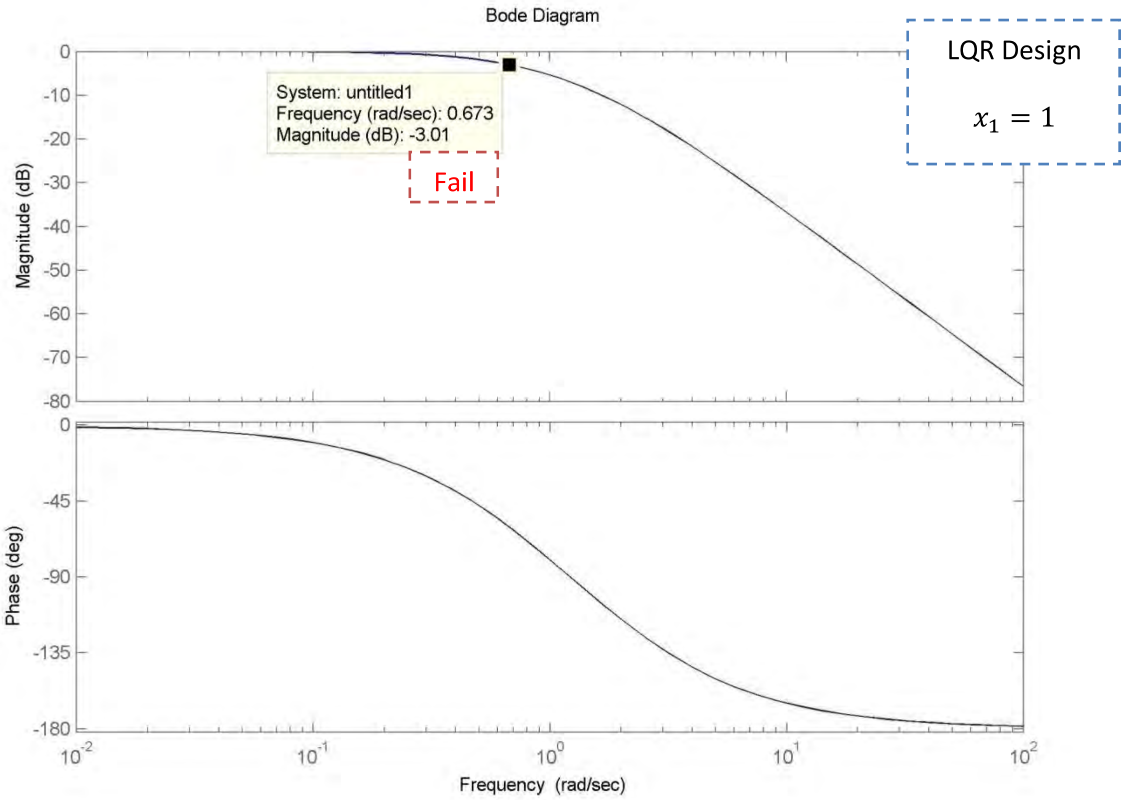
Second-Order Dominant Response Design `P = [-3+5.196*i; -3-5.196*i]`



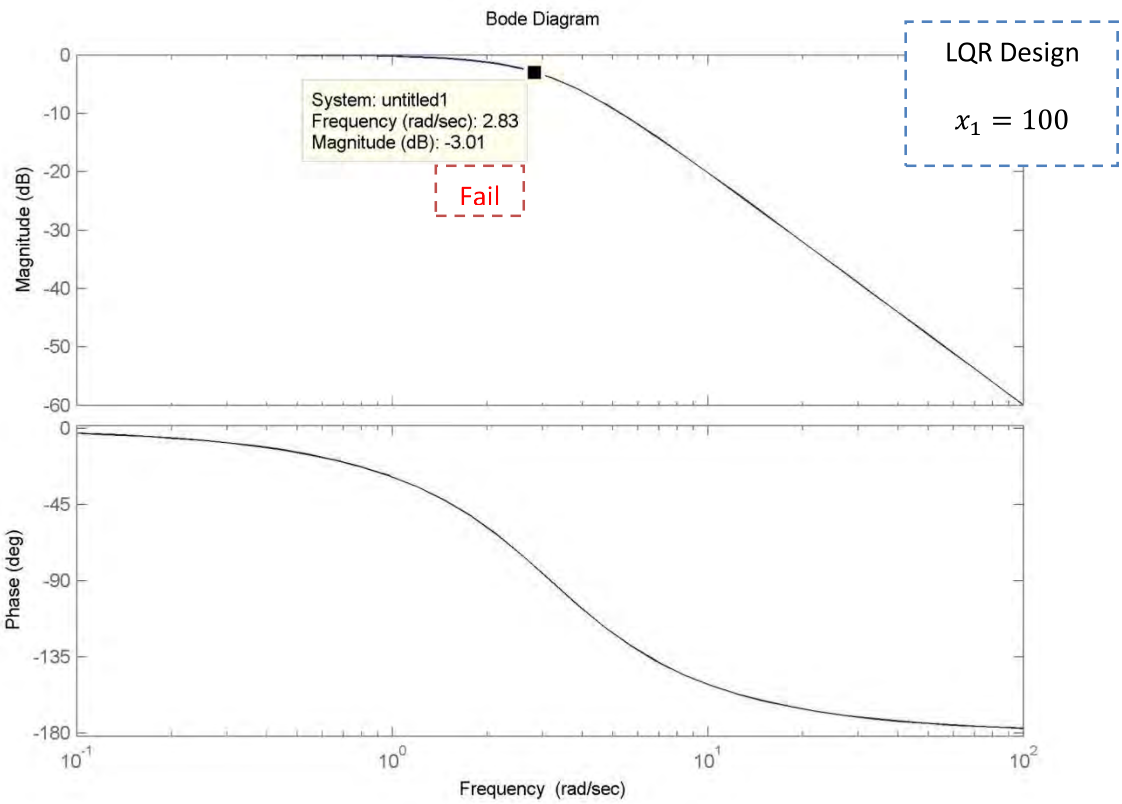


Appendix C

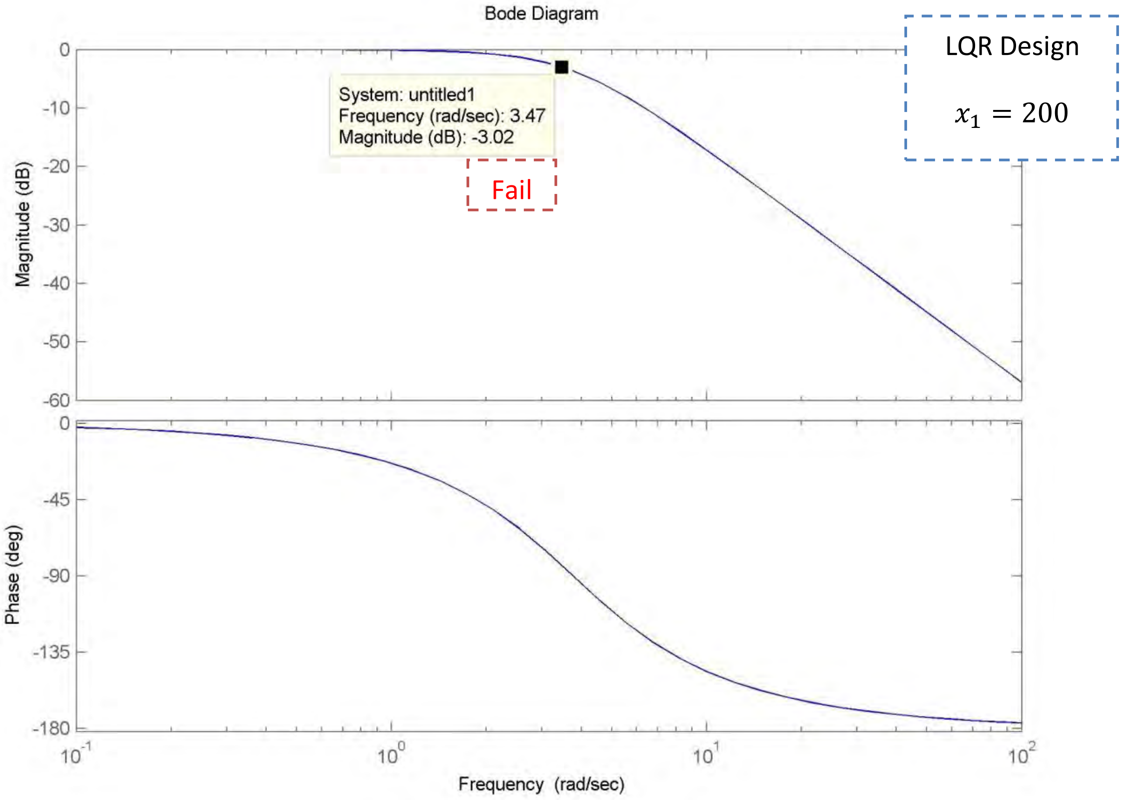
C-1



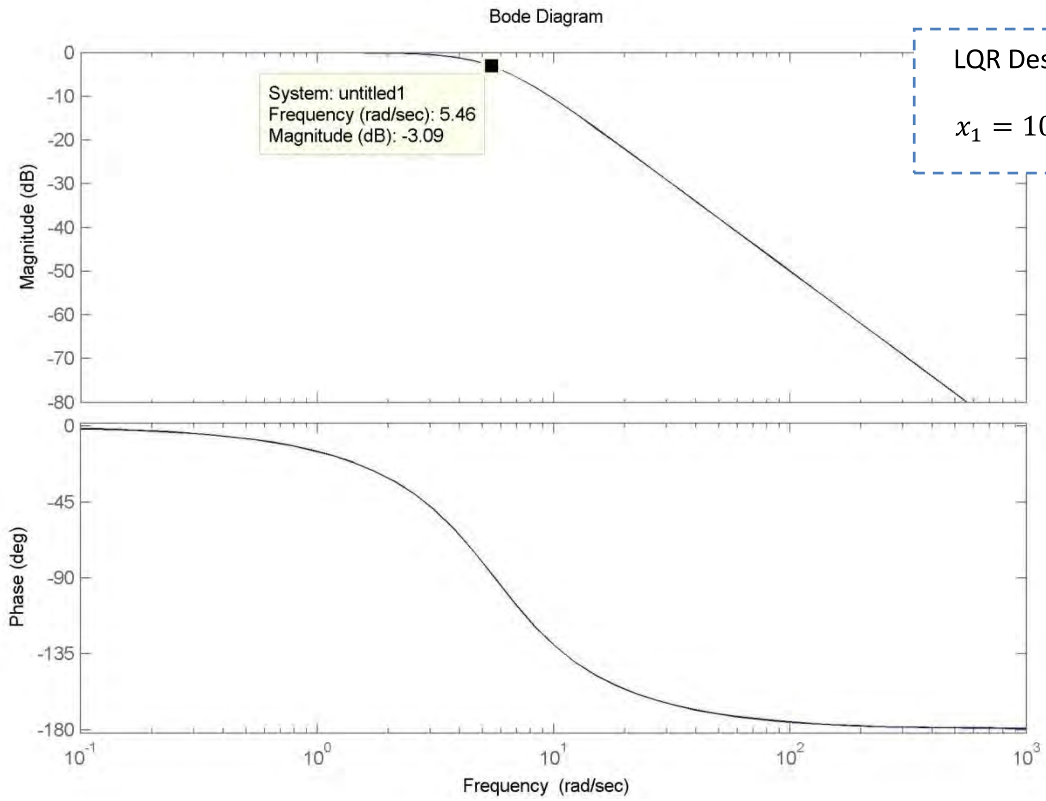
C-2



C-3

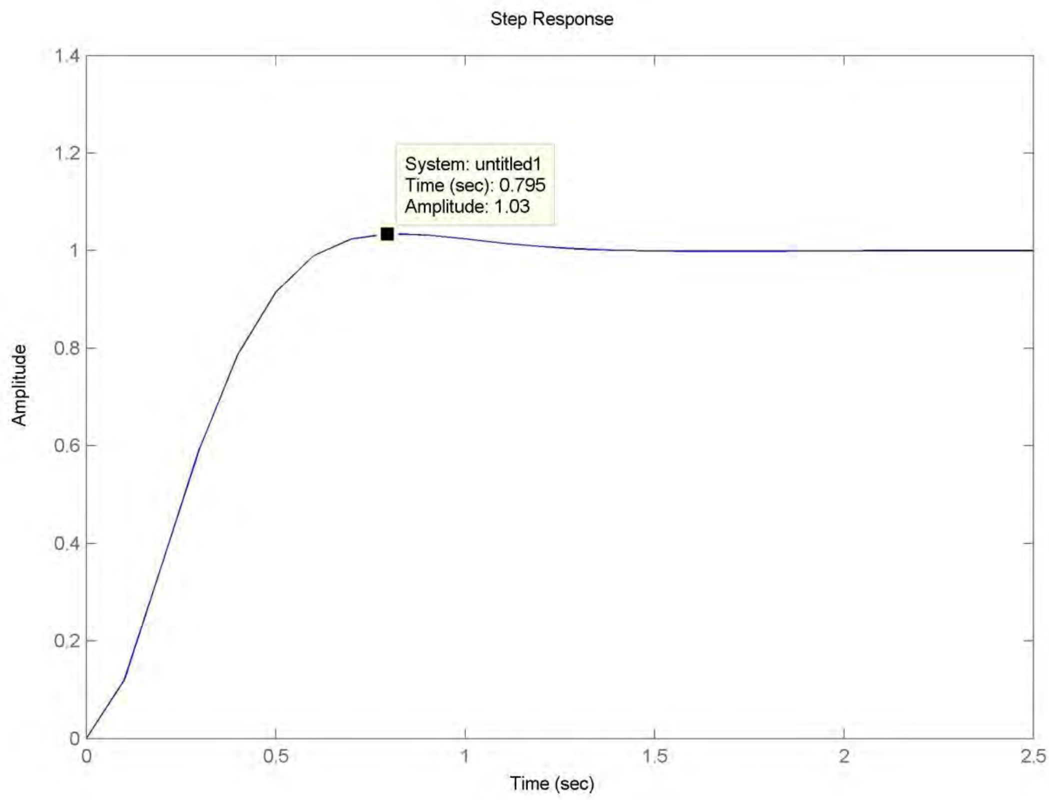

Bode Diagram
Magnitude (dB)
Phase (deg)
Frequency (rad/sec)
System: untitled1
Frequency (rad/sec): 3.47
Magnitude (dB): -3.02
Fail
LQR Design
$x_1 = 200$

C-4



LQR Design

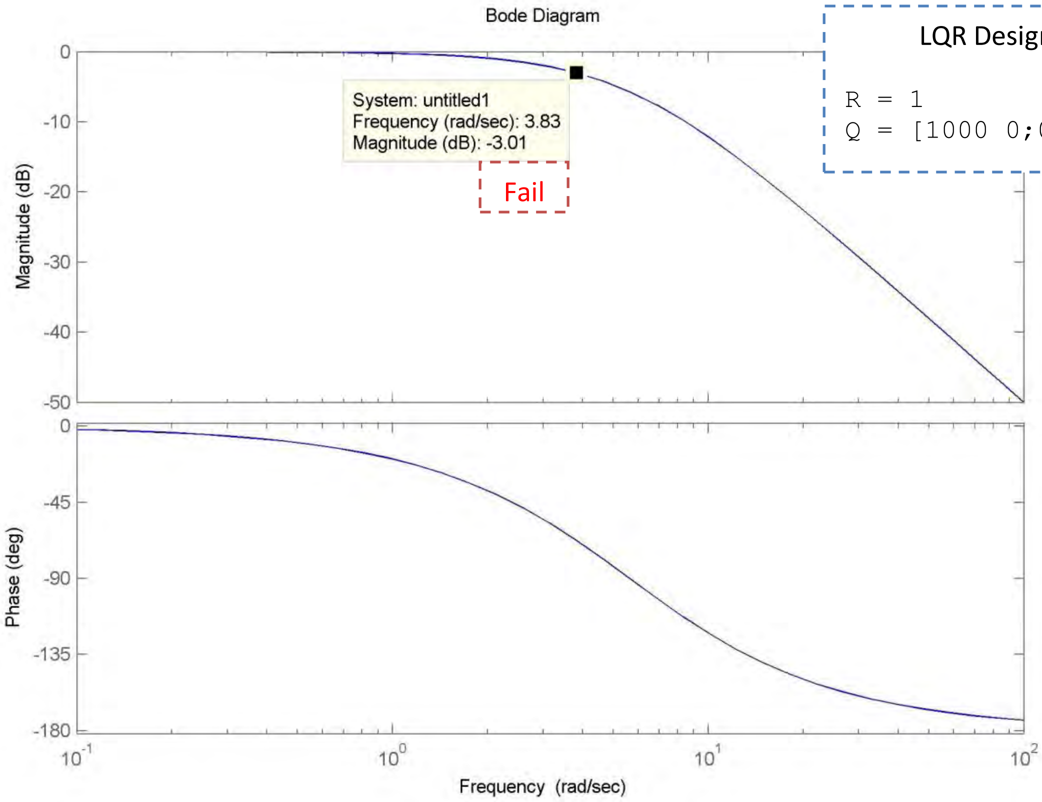
$x_1 = 1000$



C-5

LQR Design

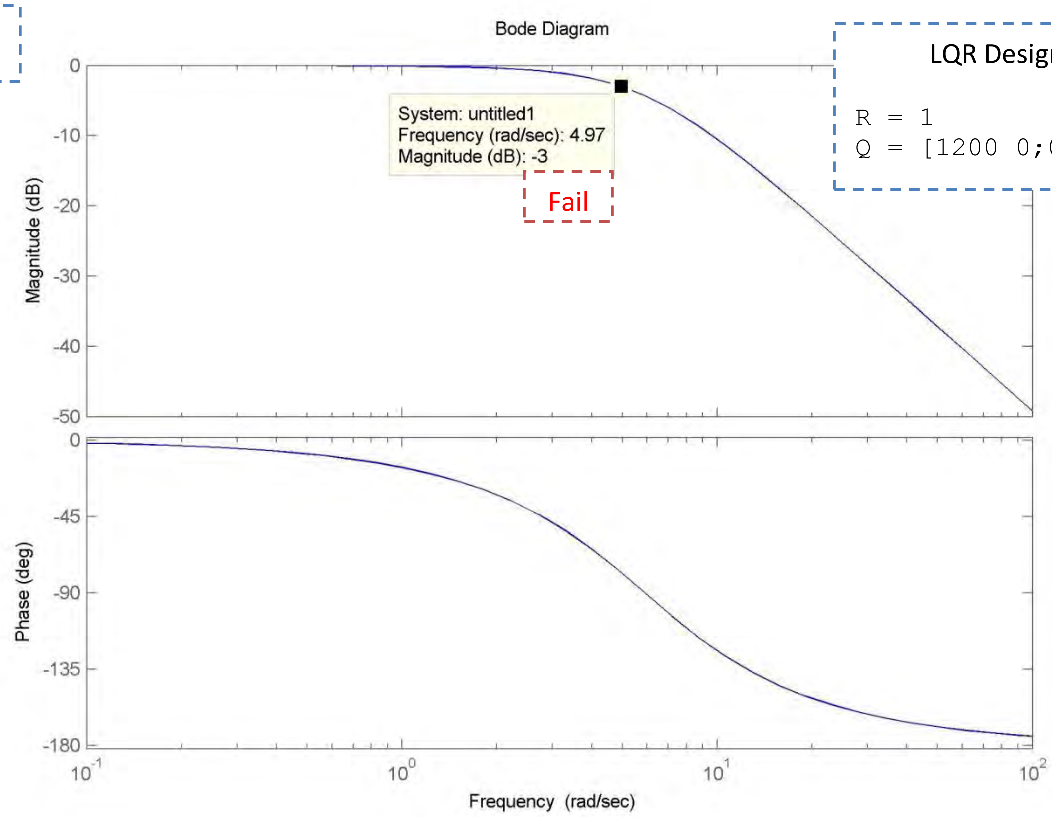
```
R = 1
Q = [1000 0;0 50]
```



C-6

LQR Design

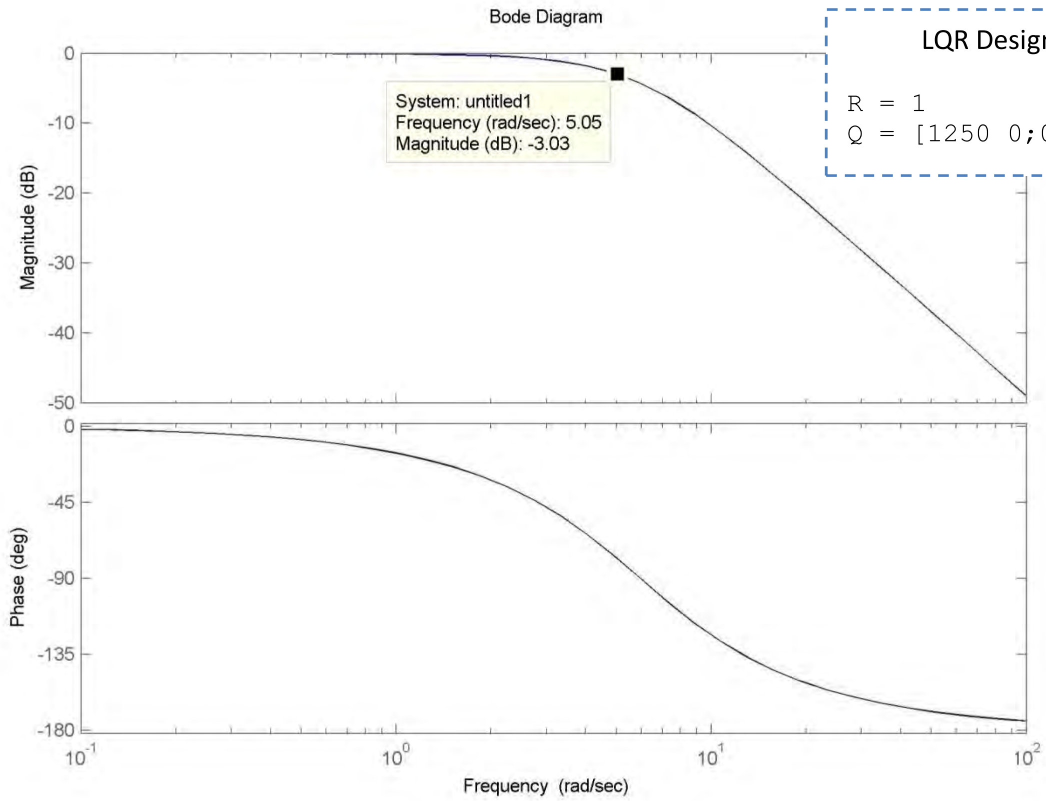
```
R = 1
Q = [1200 0;0 20]
```

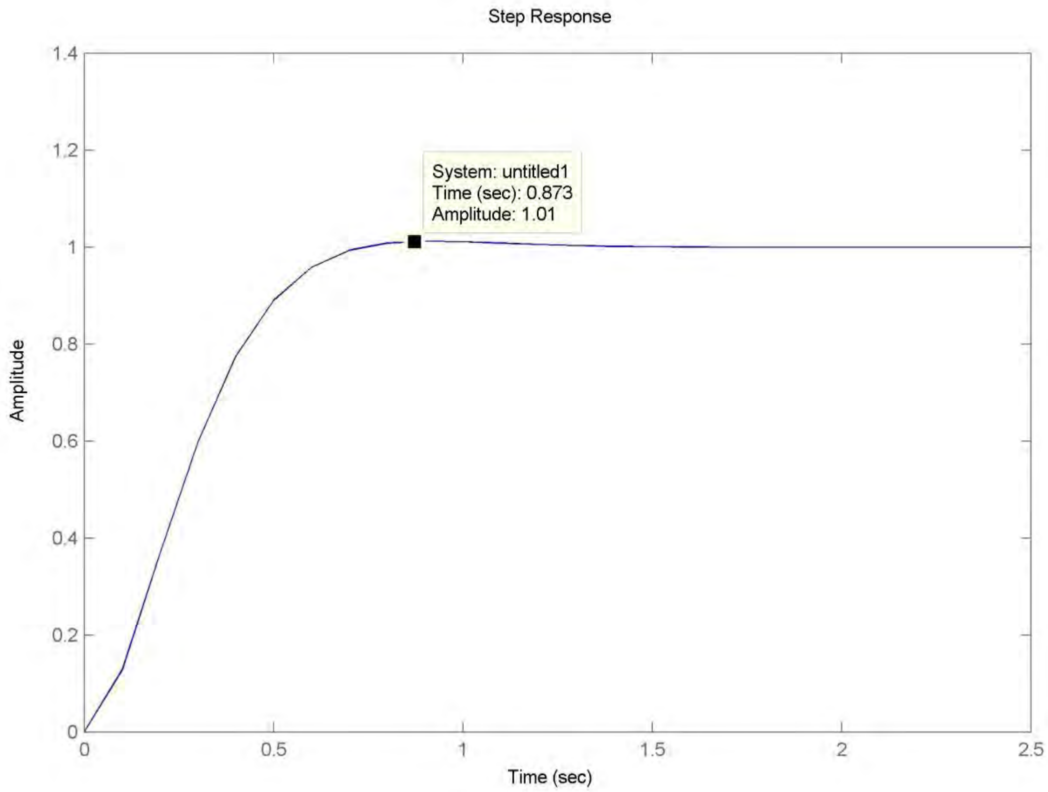


C-7

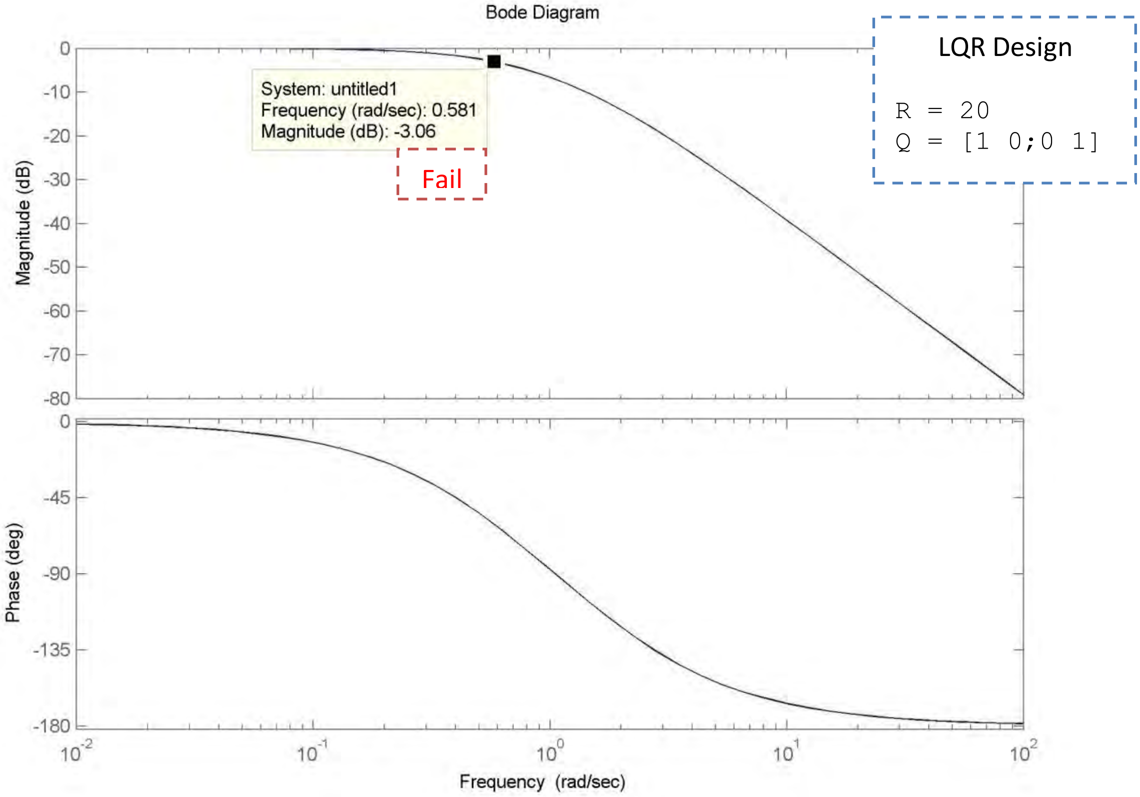
LQR Design

```
R = 1
Q = [1250 0;0 20]
```

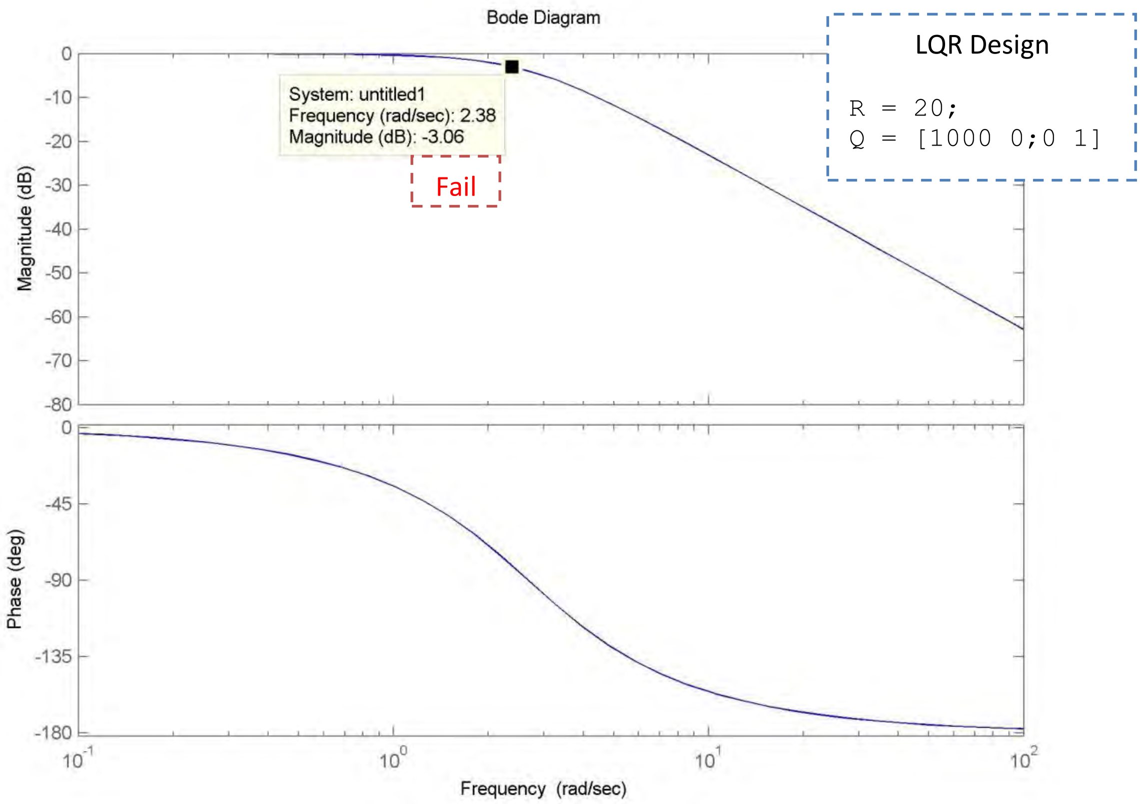




C-8

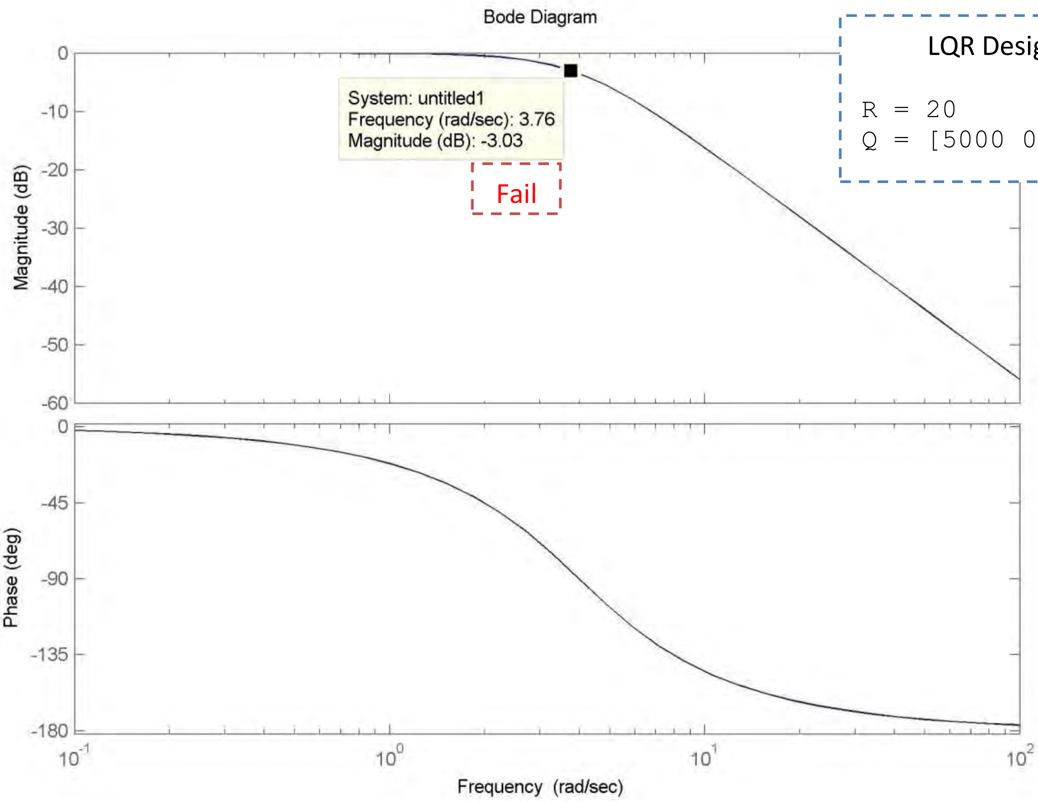

Bode Diagram
System: untitled1
Frequency (rad/sec): 0.581
Magnitude (dB): -3.06
Fail
LQR Design
R = 20
Q = [1 0;0 1]
Magnitude (dB)
Phase (deg)
Frequency (rad/sec)

C-9


Bode Diagram
System: untitled1
Frequency (rad/sec): 2.38
Magnitude (dB): -3.06
Fail
LQR Design
R = 20;
Q = [1000 0;0 1]
Magnitude (dB)
Phase (deg)
Frequency (rad/sec)

PHANG Swee King
U066584J

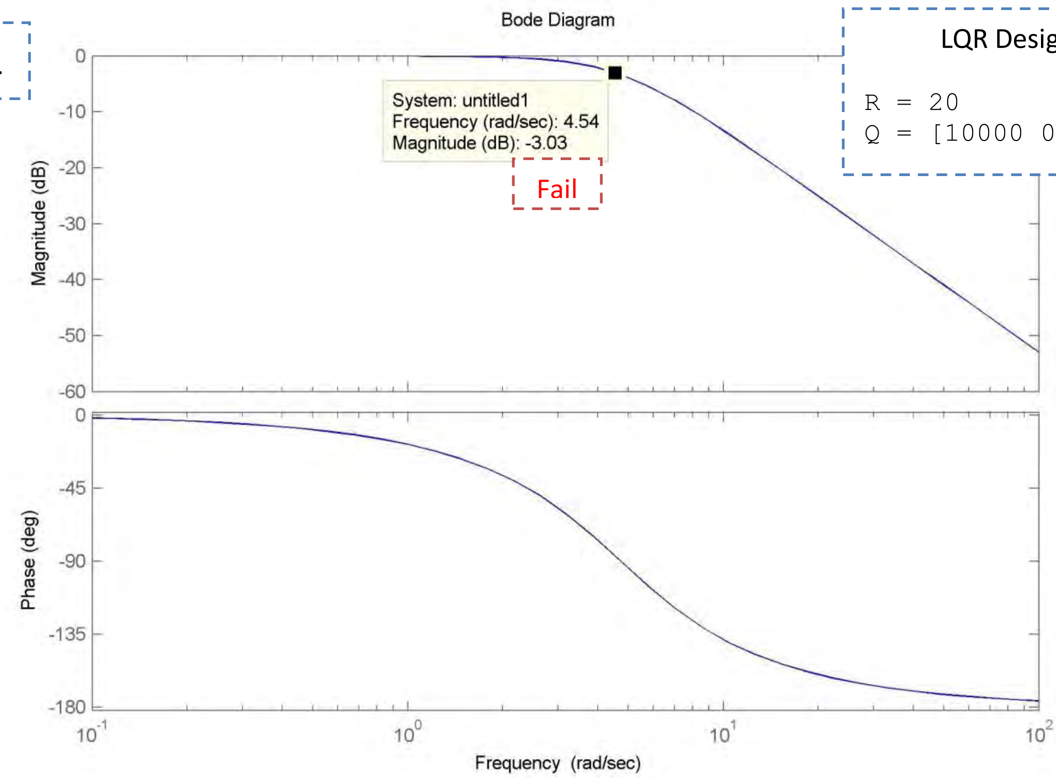
C-10



LQR Design

```
R = 20
Q = [5000 0;0 1]
```

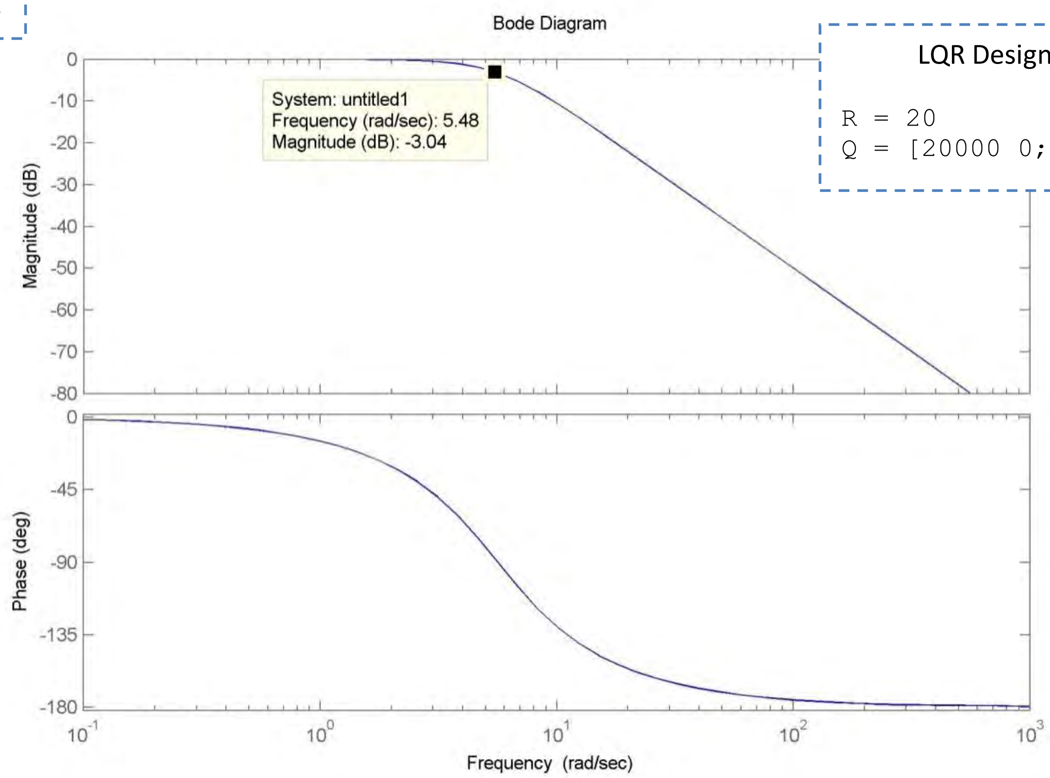
C-11



LQR Design

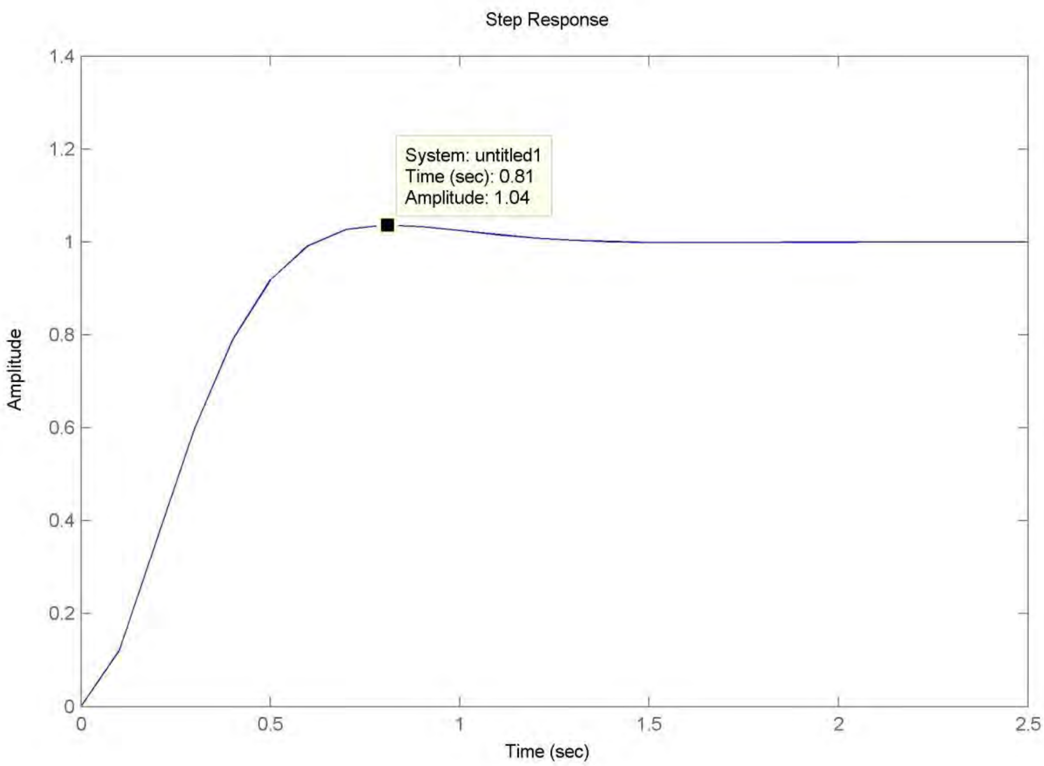
```
R = 20
Q = [10000 0;0 1]
```

C-12

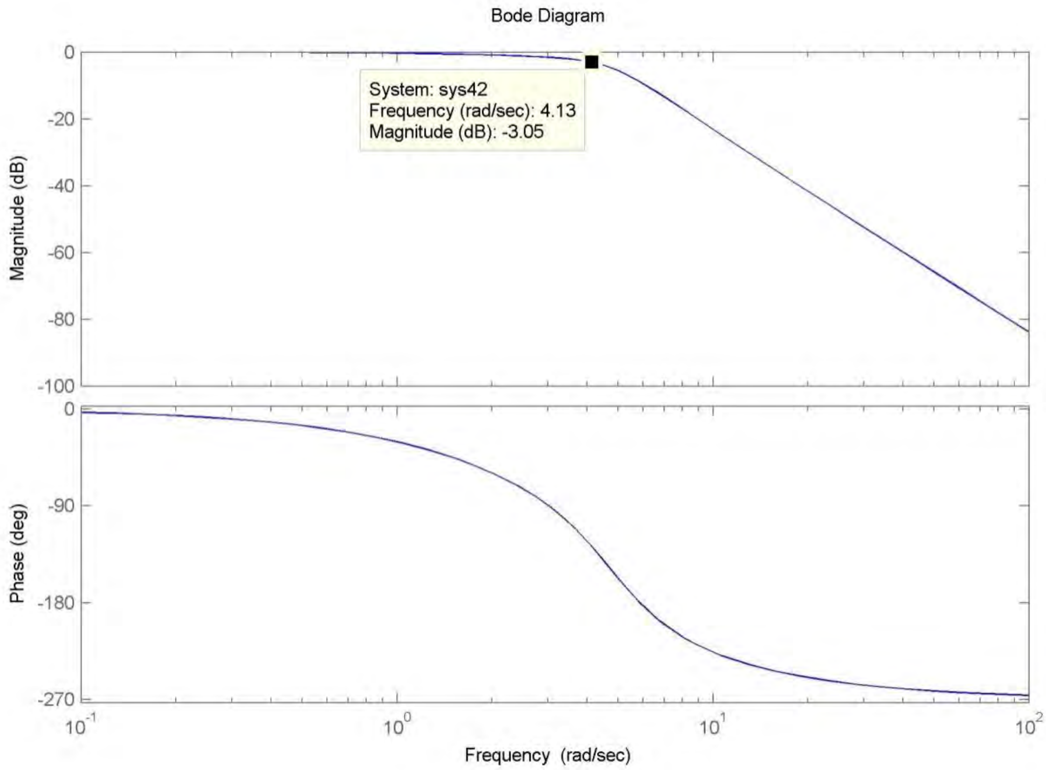


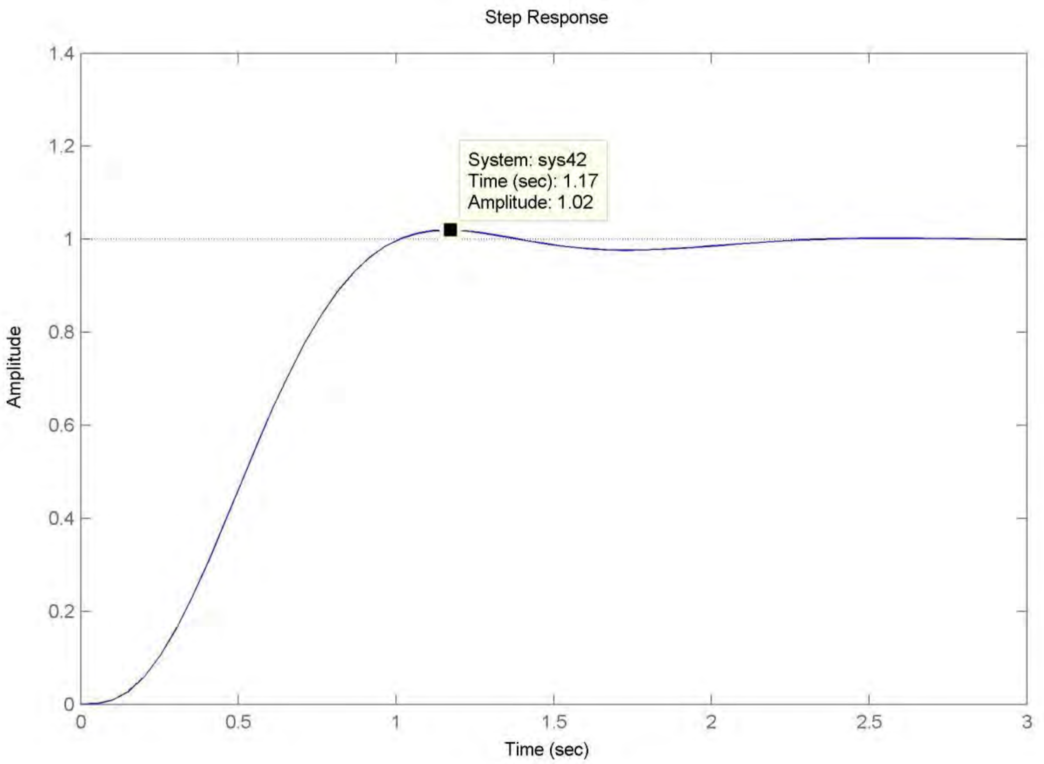
LQR Design

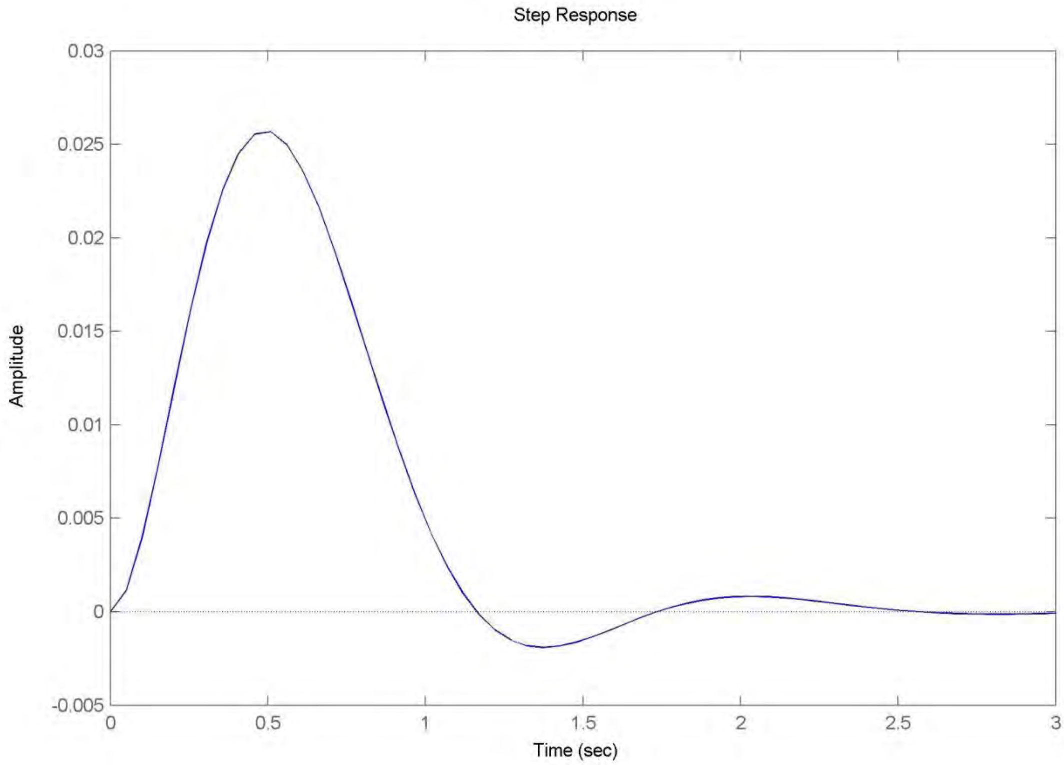
```
R = 20
Q = [20000 0;0 1]
```

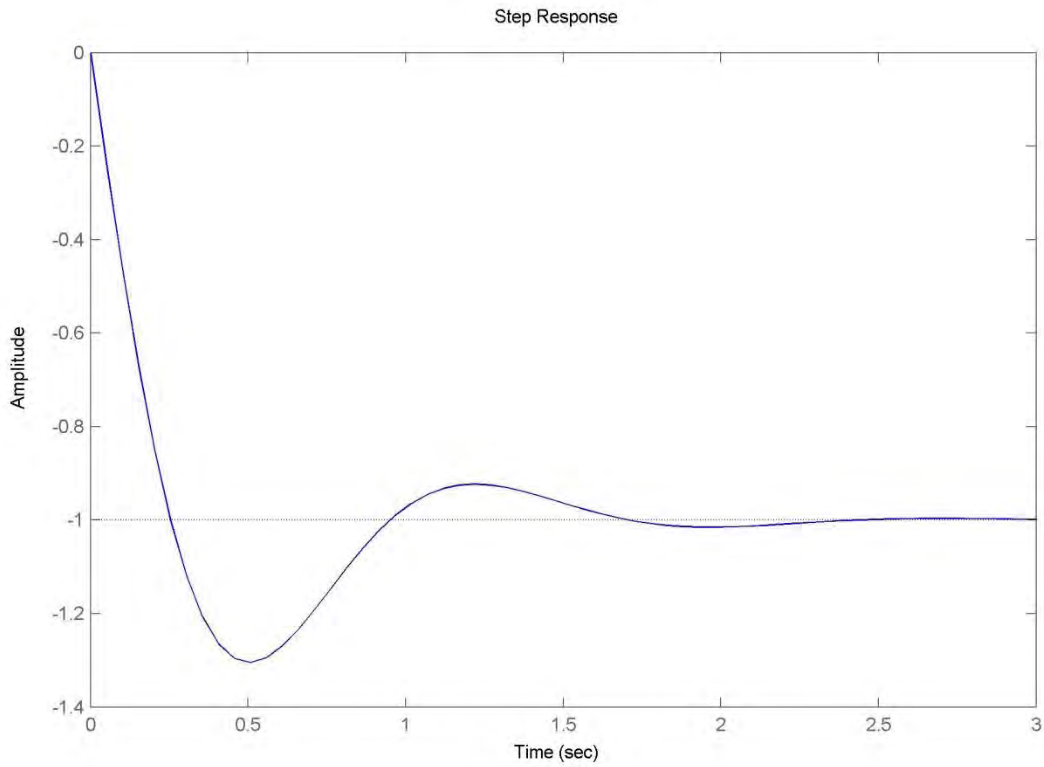


D-1

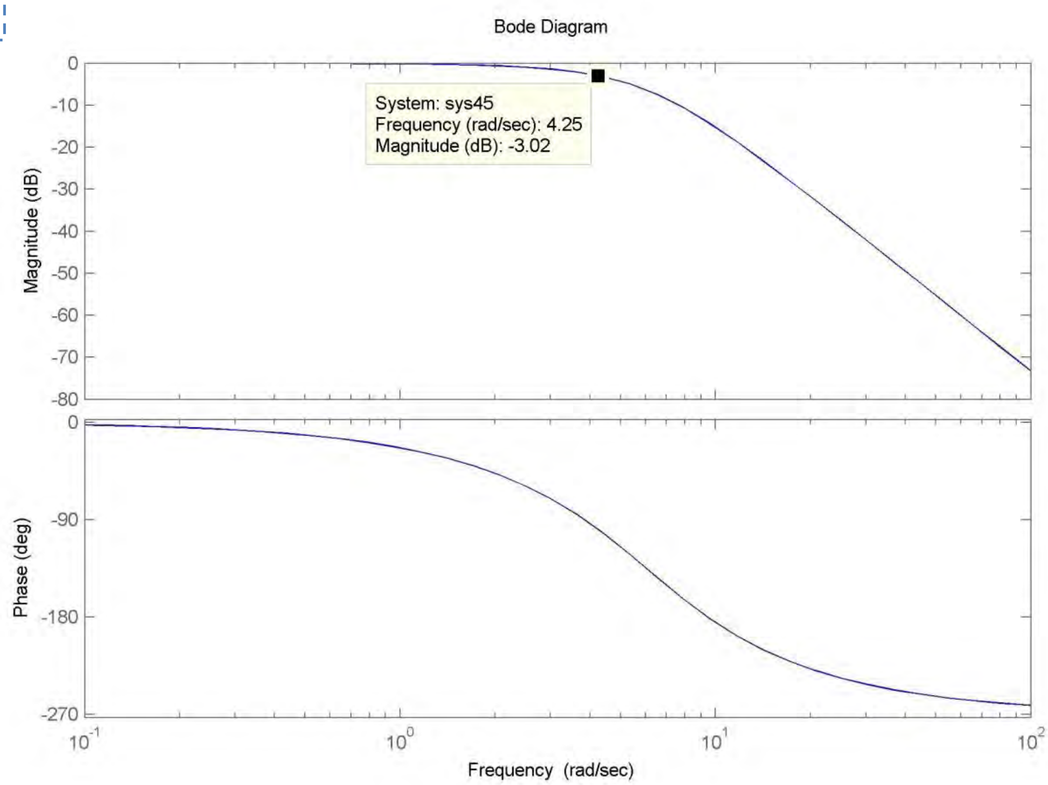

Bode Diagram
System: sys42
Frequency (rad/sec): 4.13
Magnitude (dB): -3.05
Magnitude (dB)
0
-20
-40
-60
-80
-100
Phase (deg)
0
-90
-180
-270
10^{-1}
10^{0}
10^{1}
10^{2}
Frequency (rad/sec)

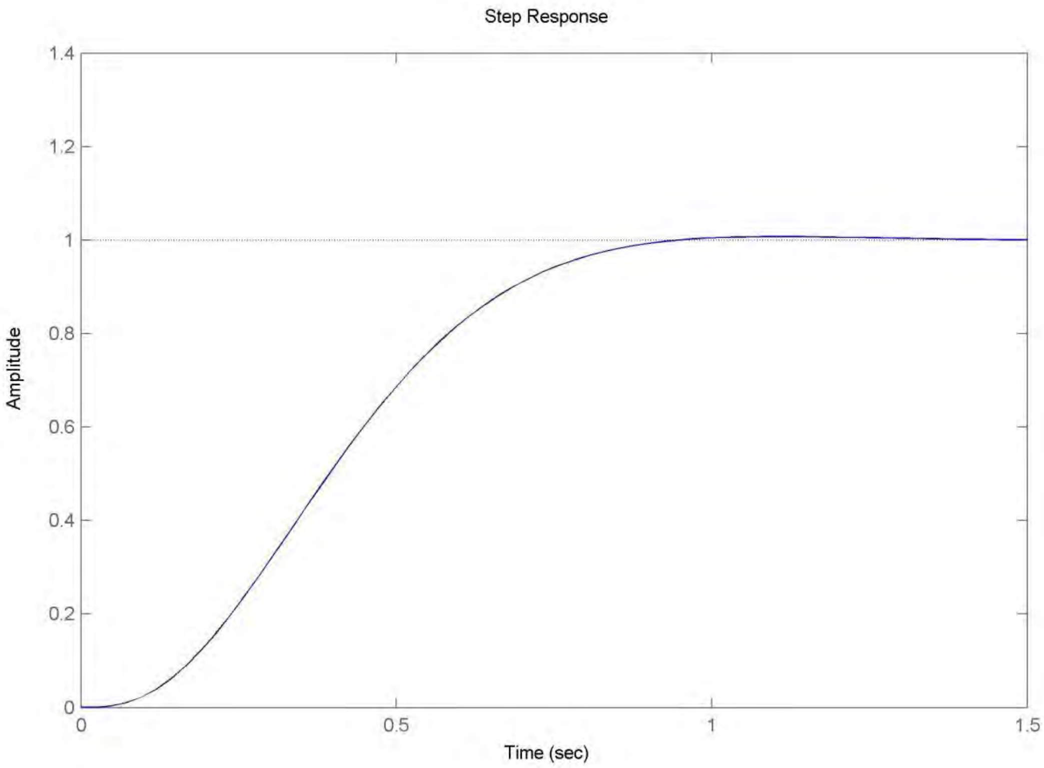

Step Response
System: sys42
Time (sec): 1.17
Amplitude: 1.02
Amplitude
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0.5
1
1.5
2
2.5
3
Time (sec)

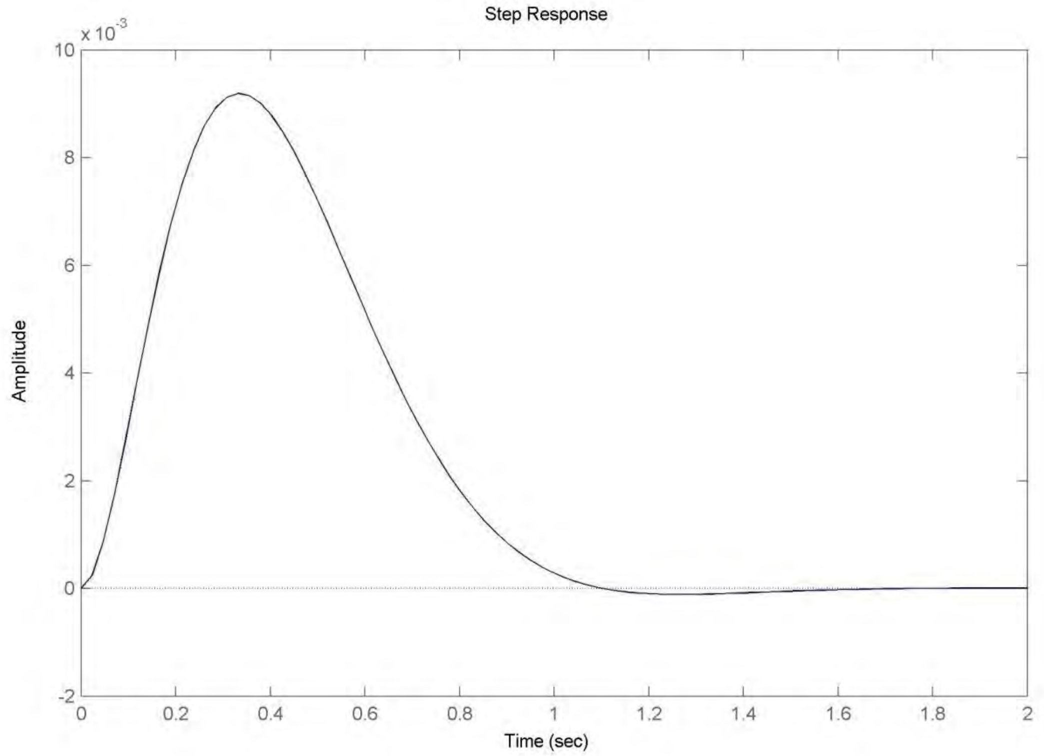

Step Response
Amplitude
Time (sec)
0.03
0.025
0.02
0.015
0.01
0.005
0
-0.005
0
0.5
1
1.5
2
2.5
3

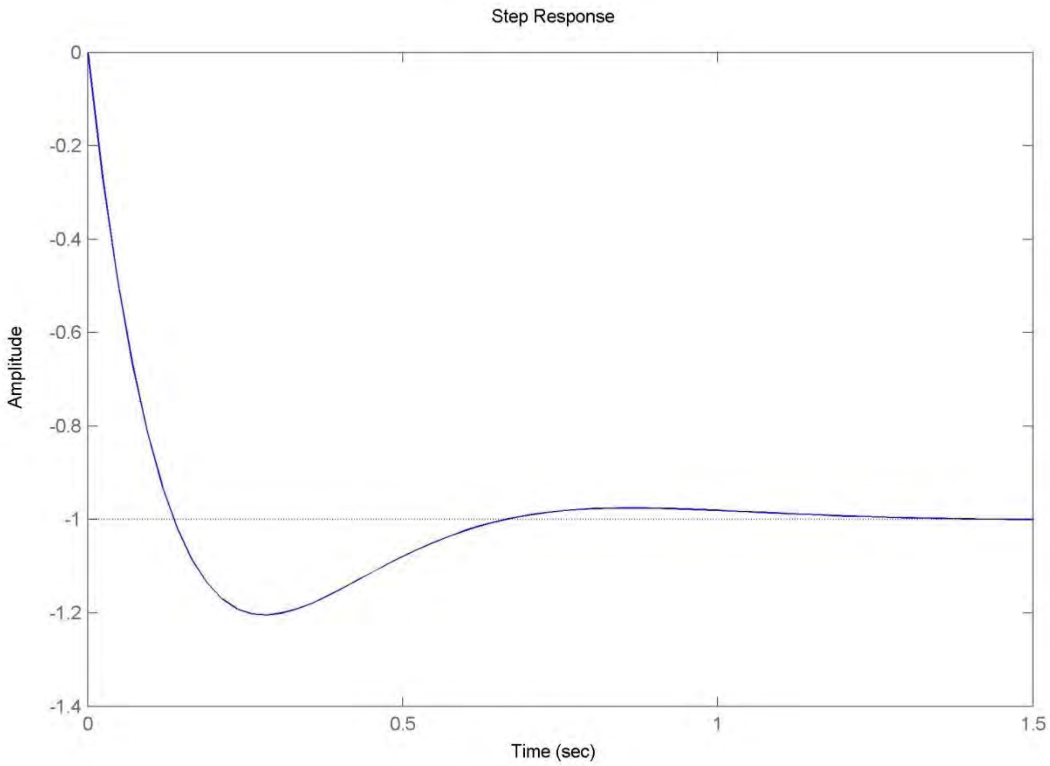

Step Response
Amplitude
Time (sec)
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
0
0.5
1
1.5
2
2.5
3

D-2


Bode Diagram
System: sys45
Frequency (rad/sec): 4.25
Magnitude (dB): -3.02
Magnitude (dB)
0
-10
-20
-30
-40
-50
-60
-70
-80
Phase (deg)
0
-90
-180
-270
10^{-1}
10^{0}
10^{1}
10^{2}
Frequency (rad/sec)


Step Response
Amplitude
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
0.5
1
1.5
Time (sec)

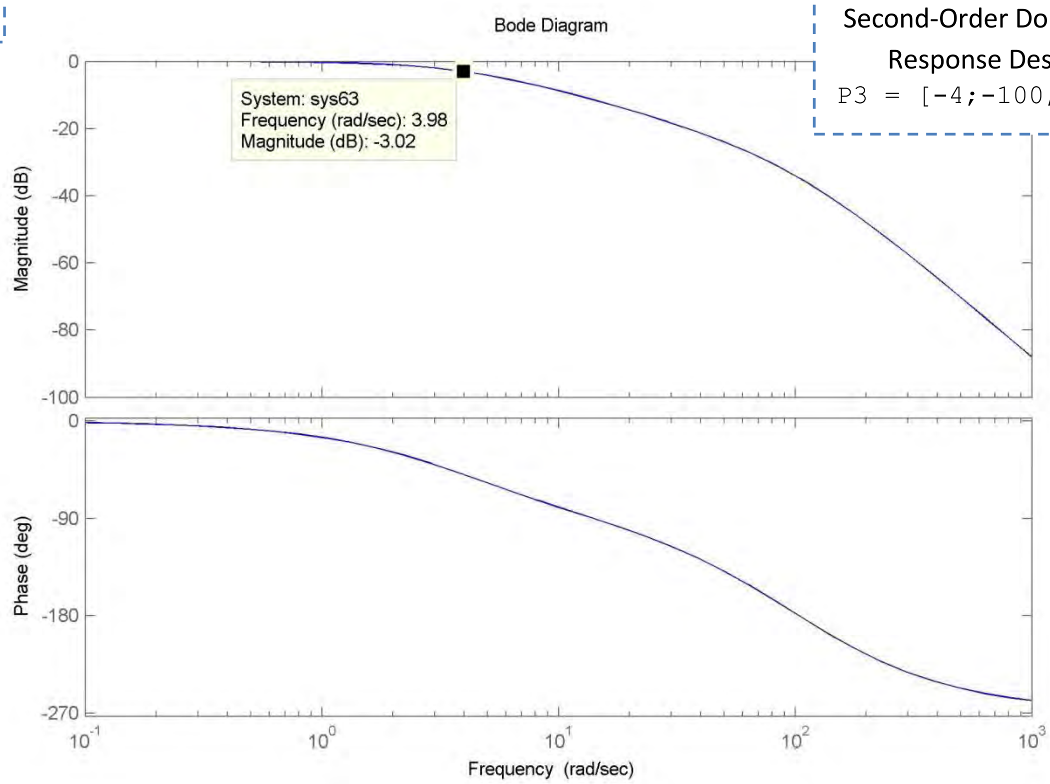

Step Response
x 10^-3
10
8
6
4
2
0
-2
Amplitude
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
Time (sec)

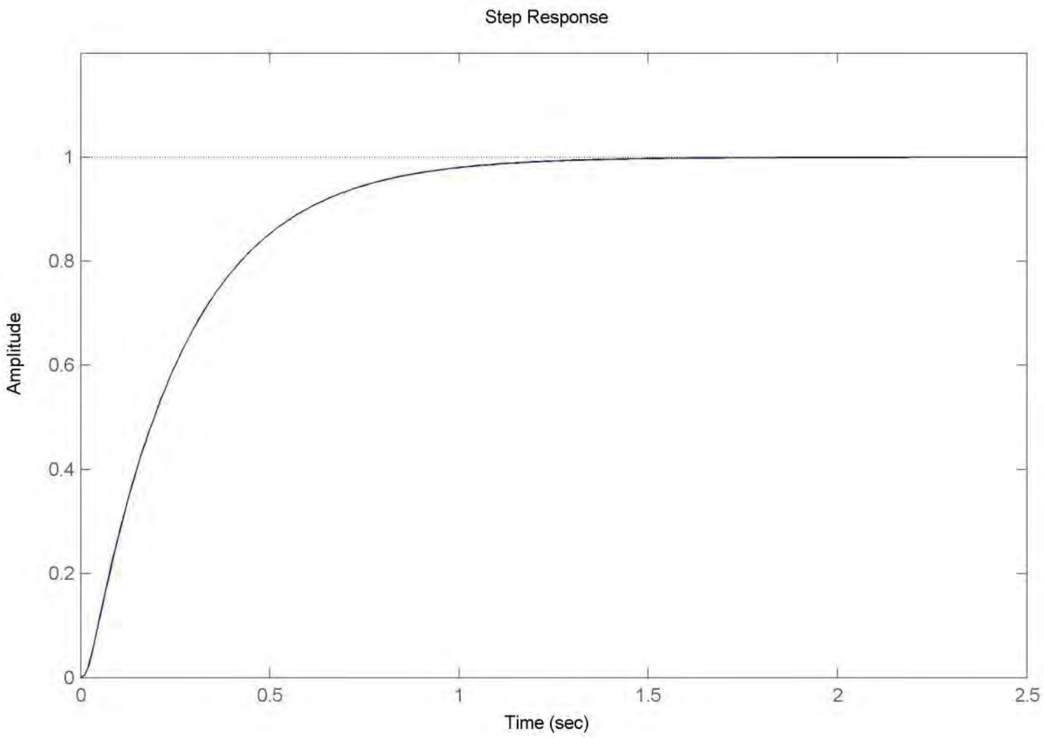

Step Response
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
Amplitude
0
0.5
1
1.5
Time (sec)

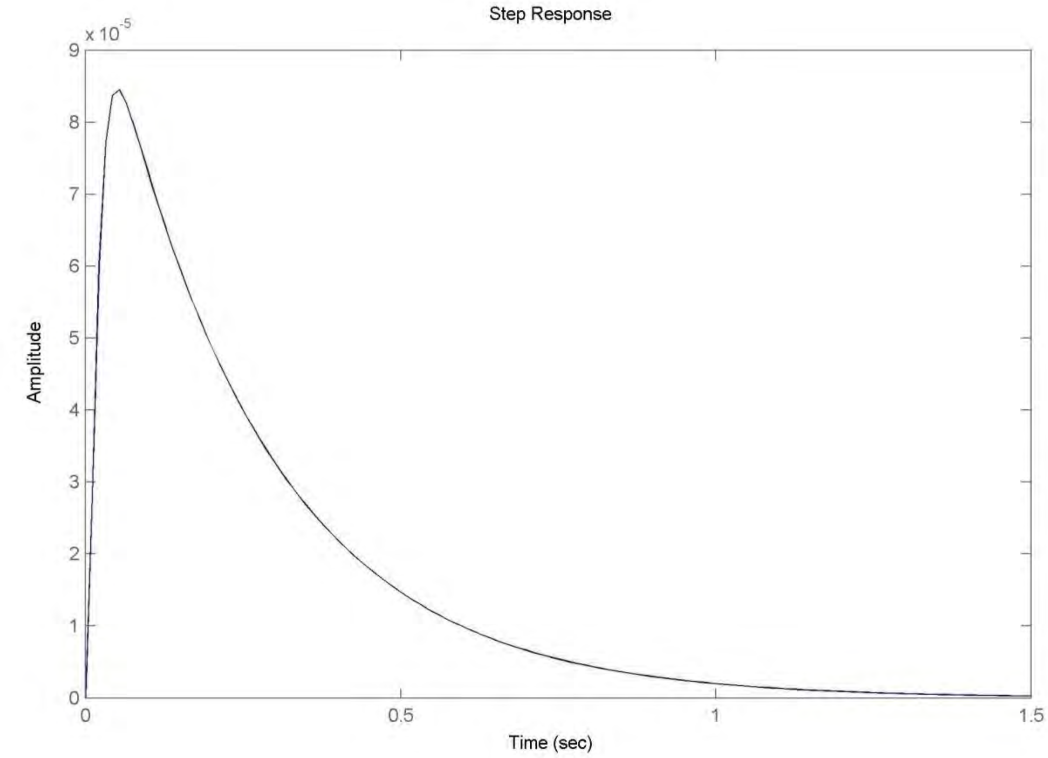
D-3

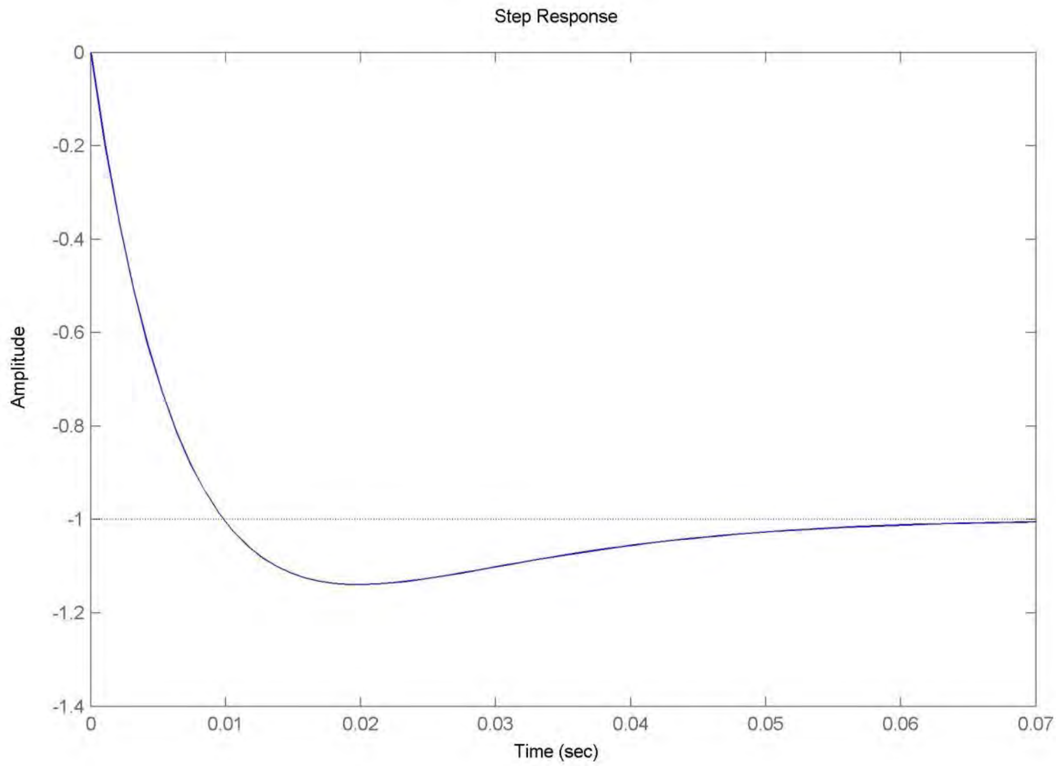
Second-Order Dominant Response Design

`P3 = [-4;-100;-100]`





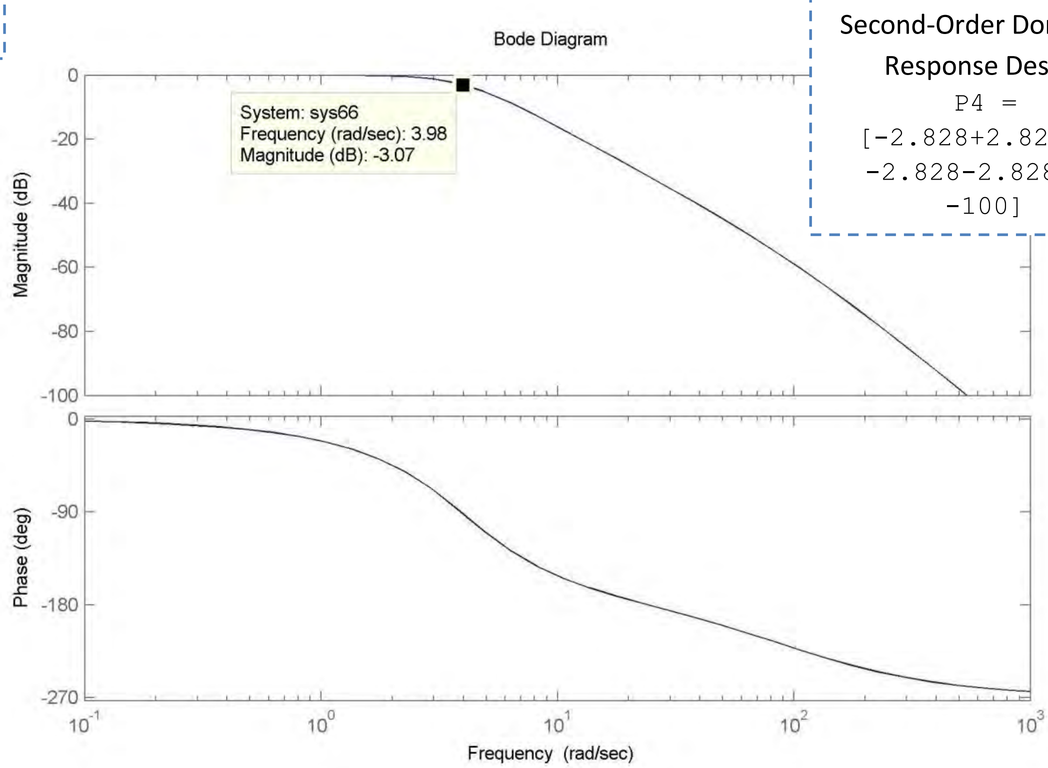

Step Response
x 10^-5
Amplitude
Time (sec)
0
1
2
3
4
5
6
7
8
9
0
0.5
1
1.5

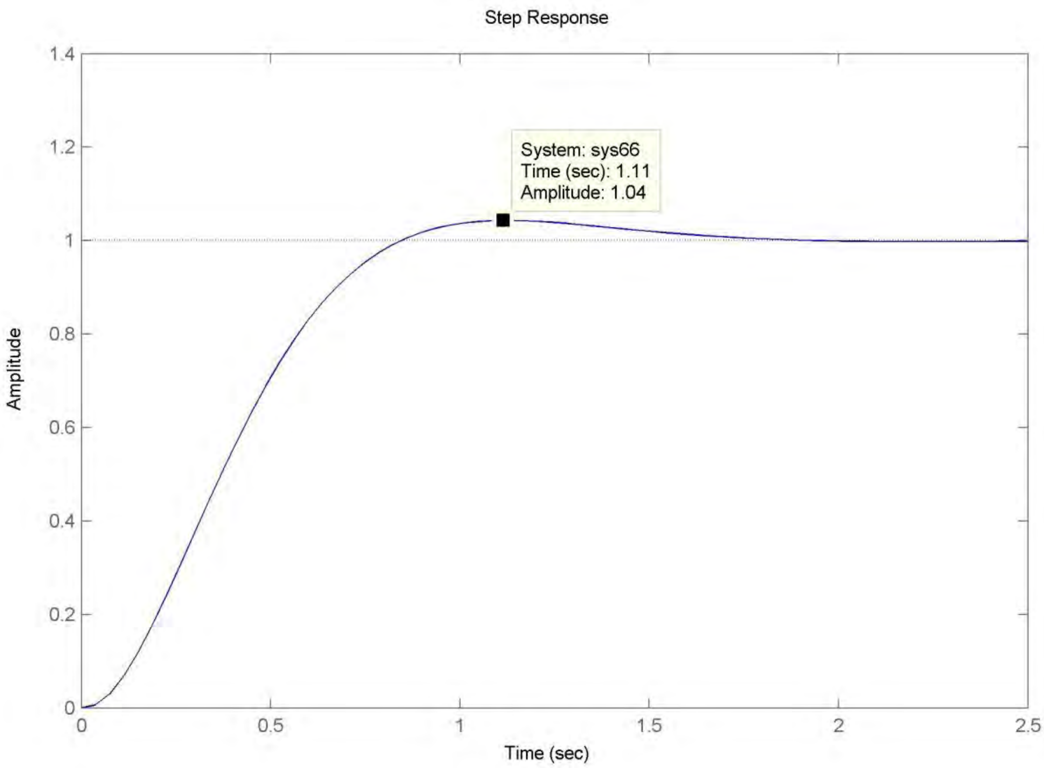

Step Response
Amplitude
Time (sec)
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07

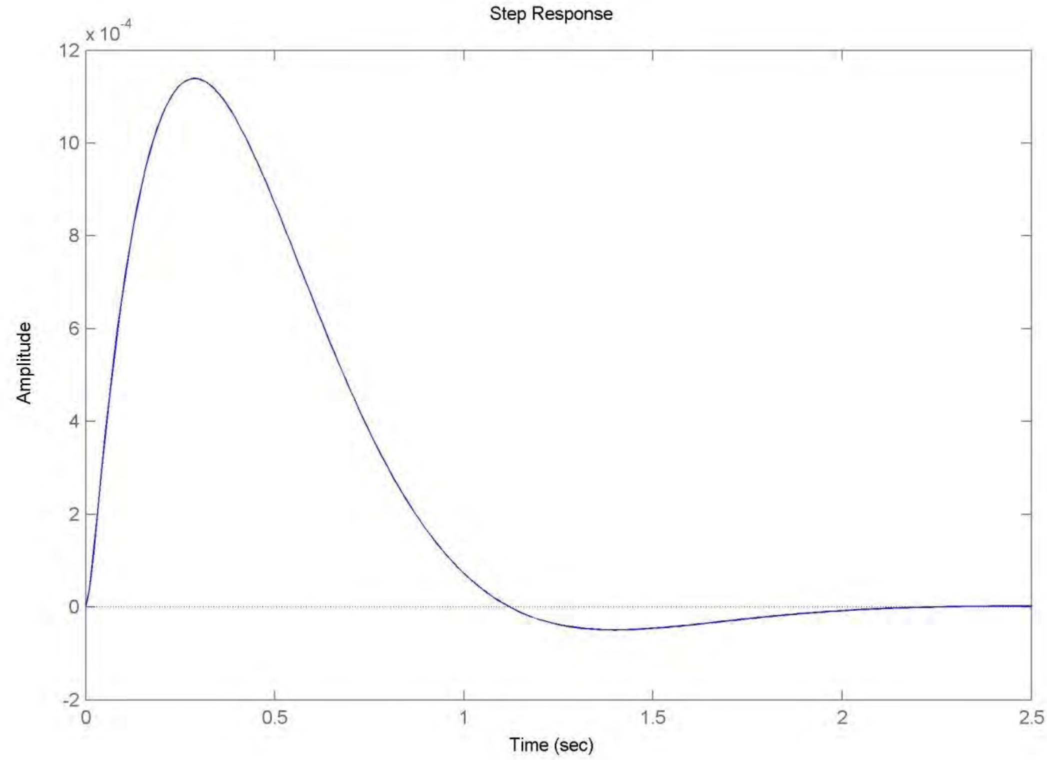
D-4

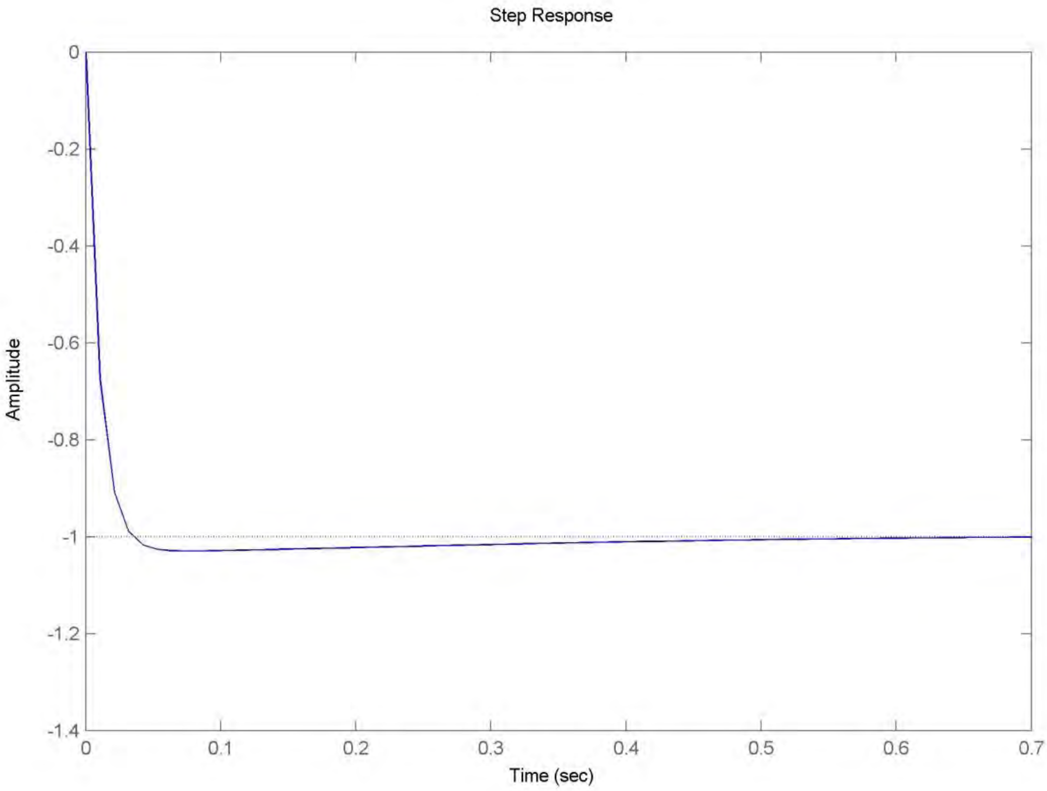
Second-Order Dominant Response Design

```
P4 =
[-2.828+2.828*i;
-2.828-2.828*i;
-100]
```





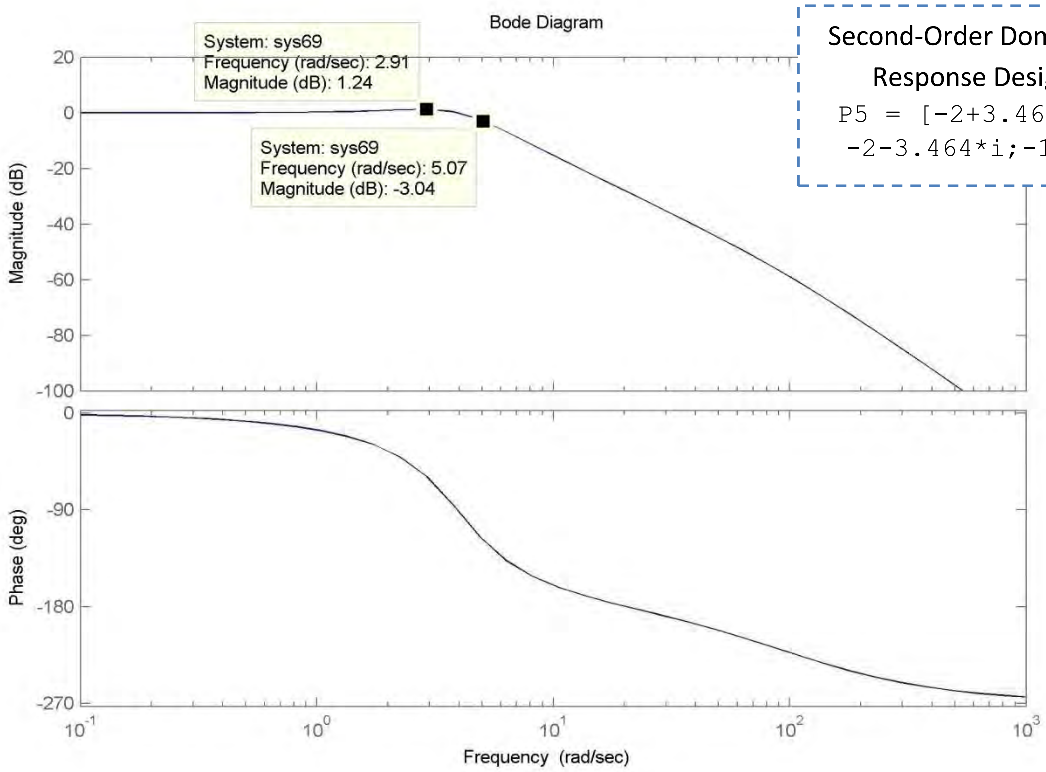

Step Response
x 10^-4
12
10
8
6
4
2
0
-2
Amplitude
0
0.5
1
1.5
2
2.5
Time (sec)

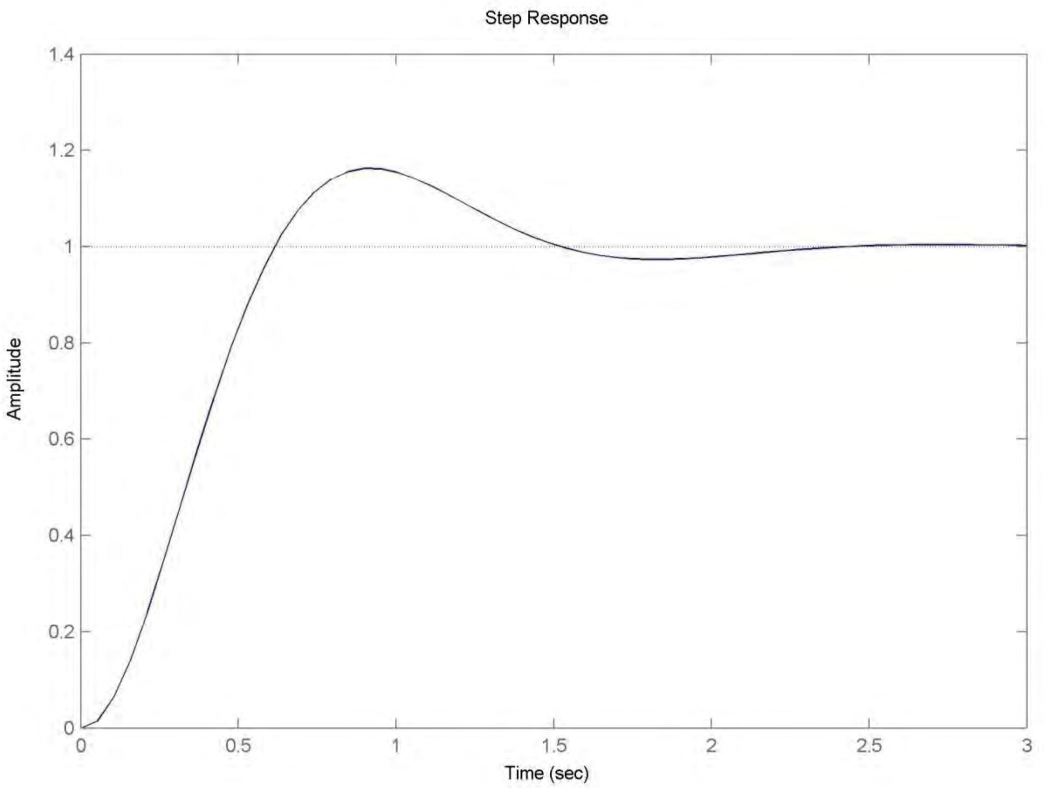

Step Response
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
Amplitude
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
Time (sec)

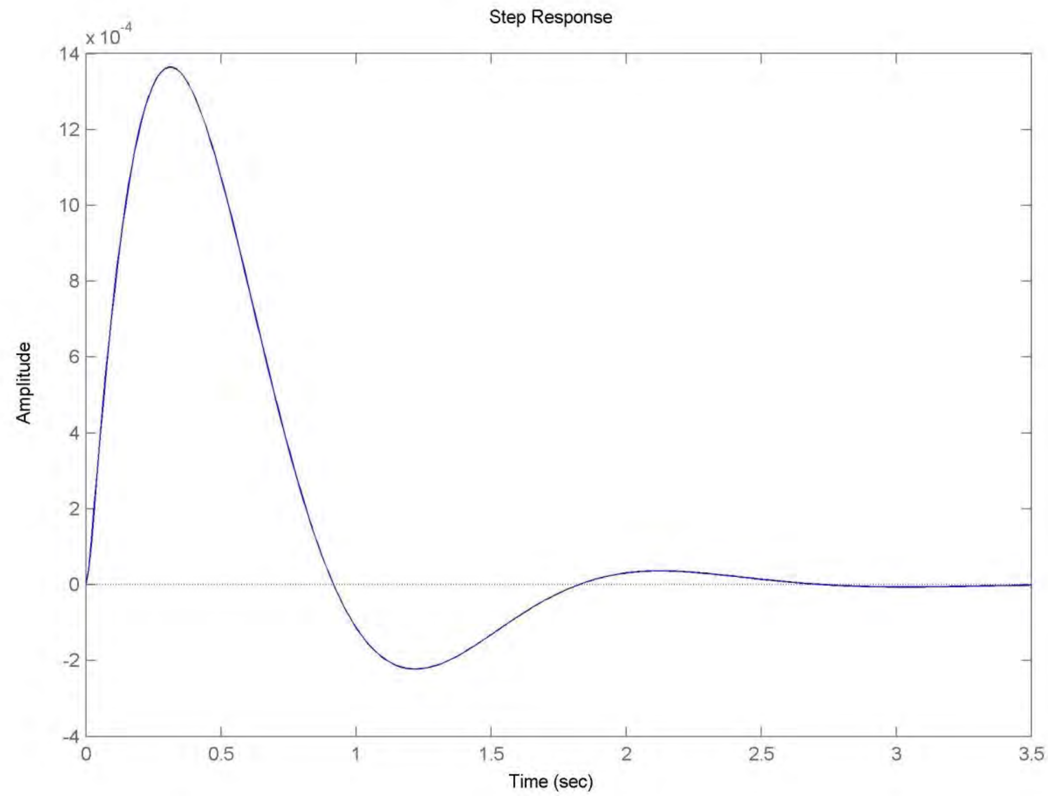
D-5

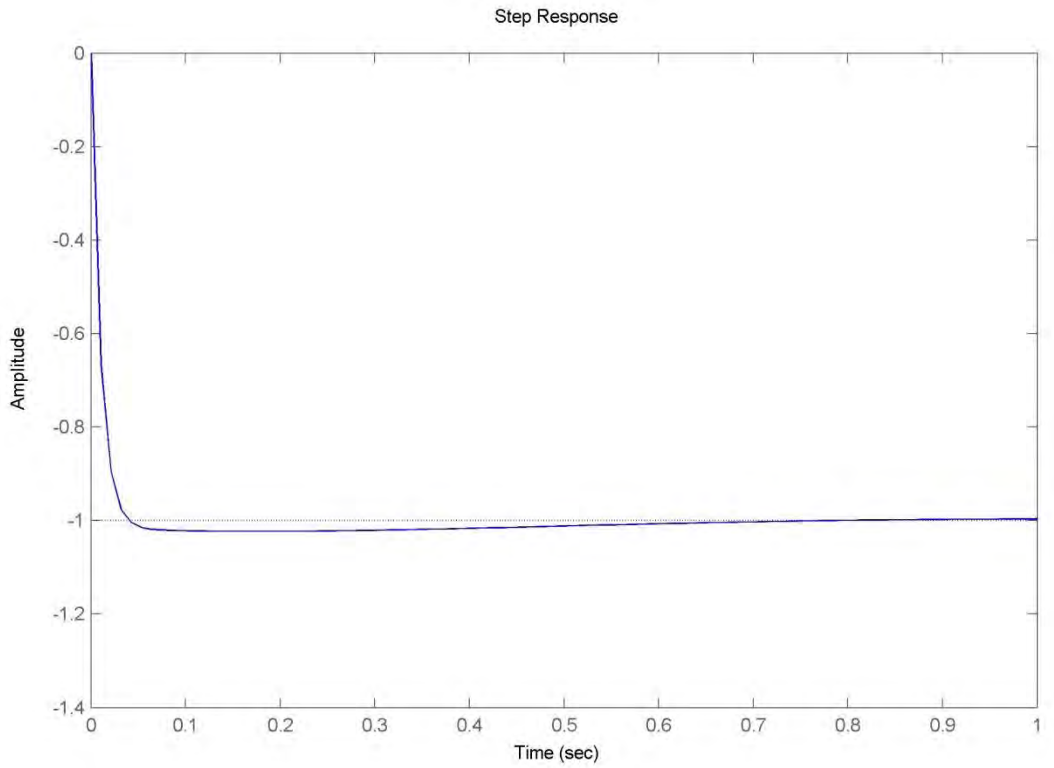
Second-Order Dominant Response Design

`P5 = [-2+3.464*i; -2-3.464*i;-100]`



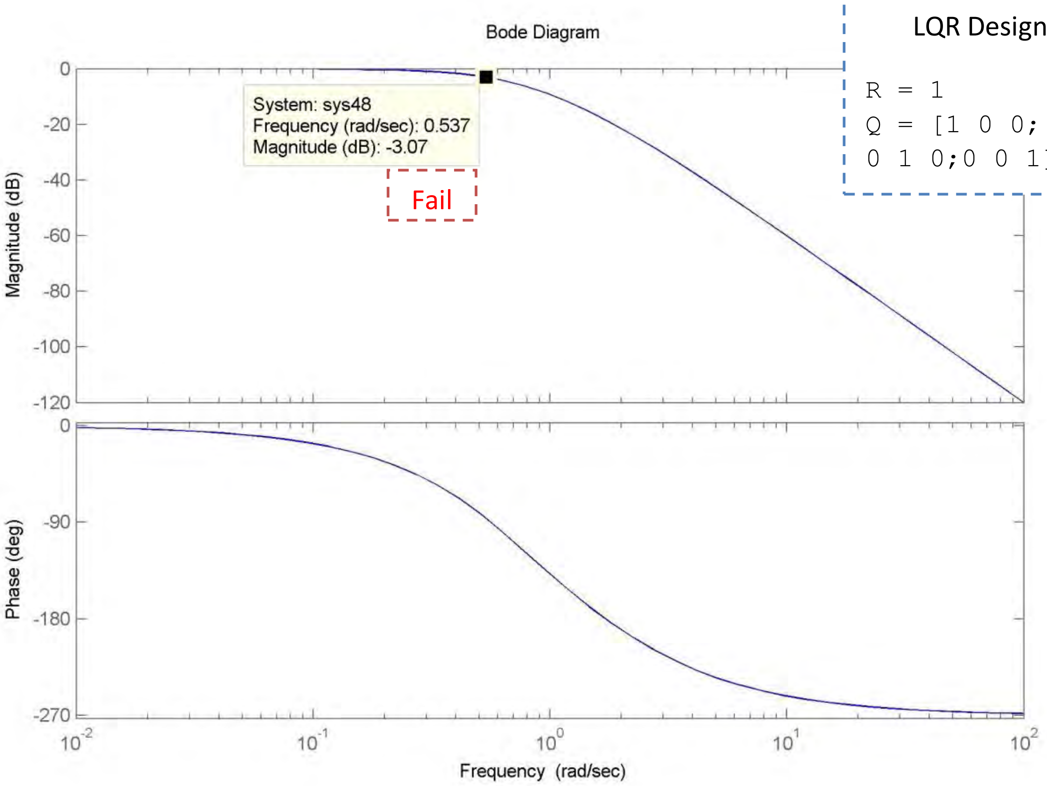



Step Response
x 10^-4
14
12
10
8
6
4
2
0
-2
-4
Amplitude
0
0.5
1
1.5
2
2.5
3
3.5
Time (sec)


Step Response
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
Amplitude
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Time (sec)

Appendix E

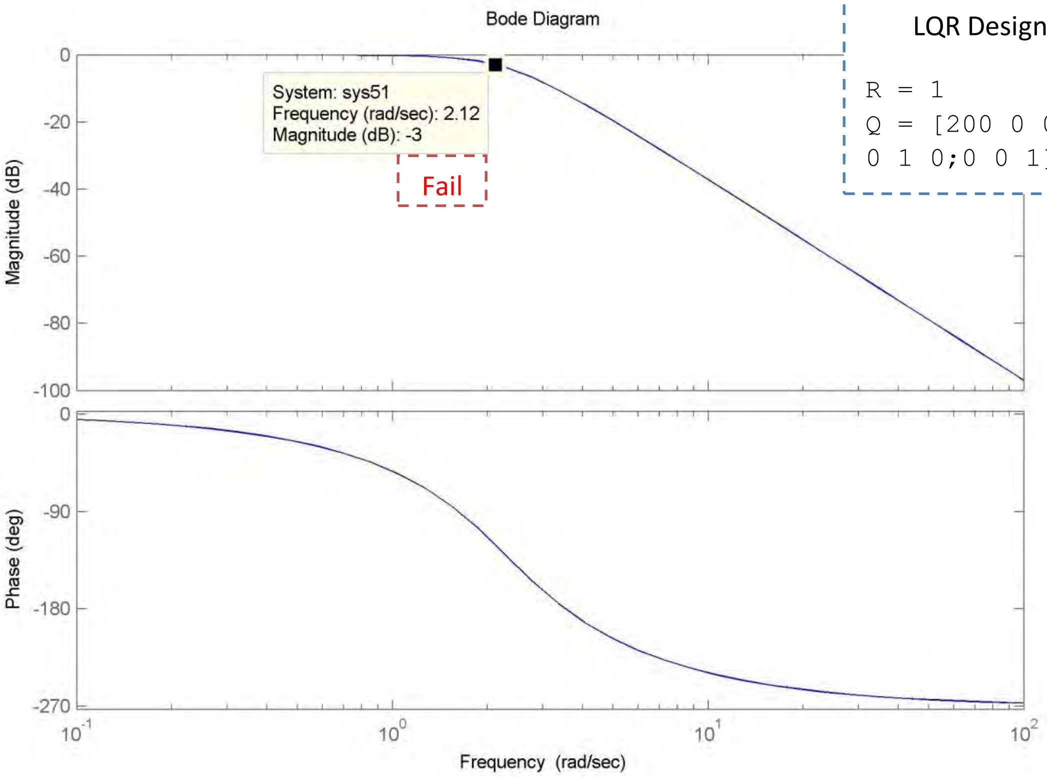
E-1



LQR Design

R = 1
Q = [1 0 0;
0 1 0;0 0 1]

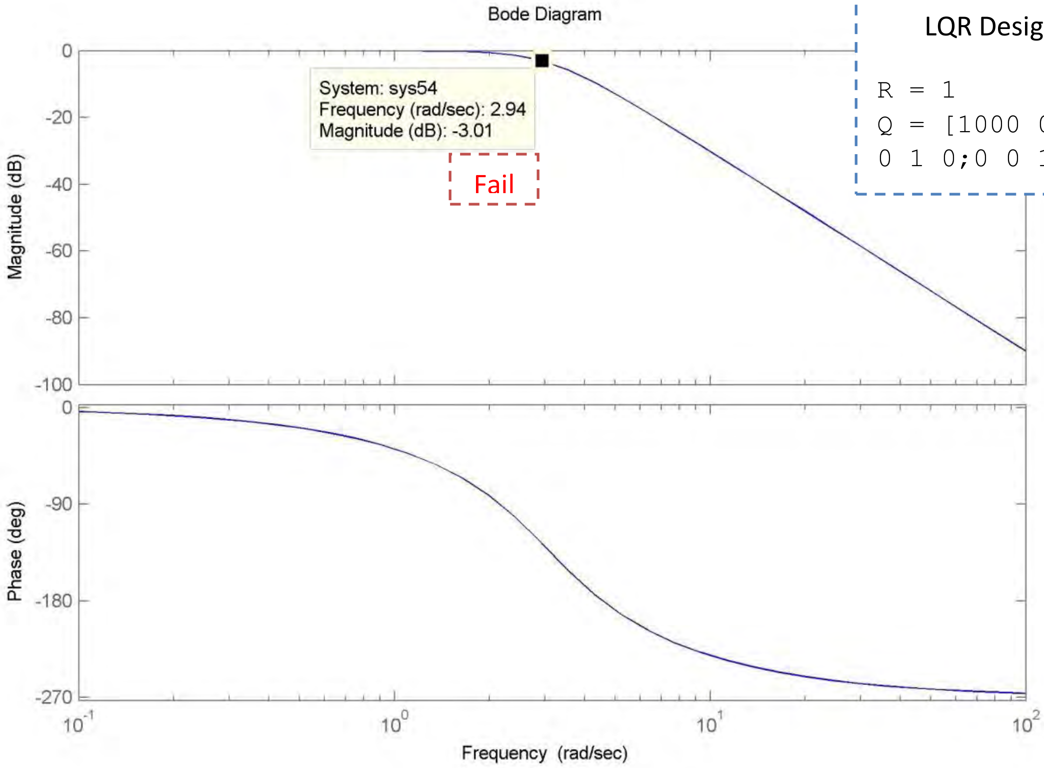
E-2



LQR Design

R = 1
Q = [200 0 0;
0 1 0;0 0 1]

E-3



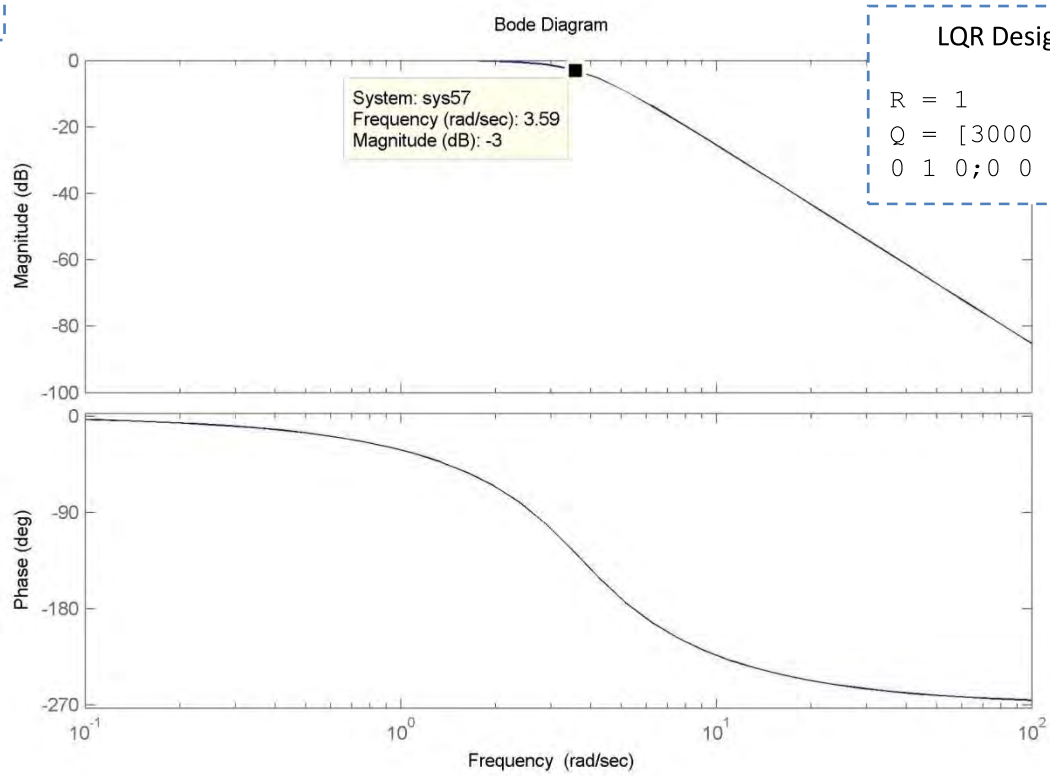
LQR Design

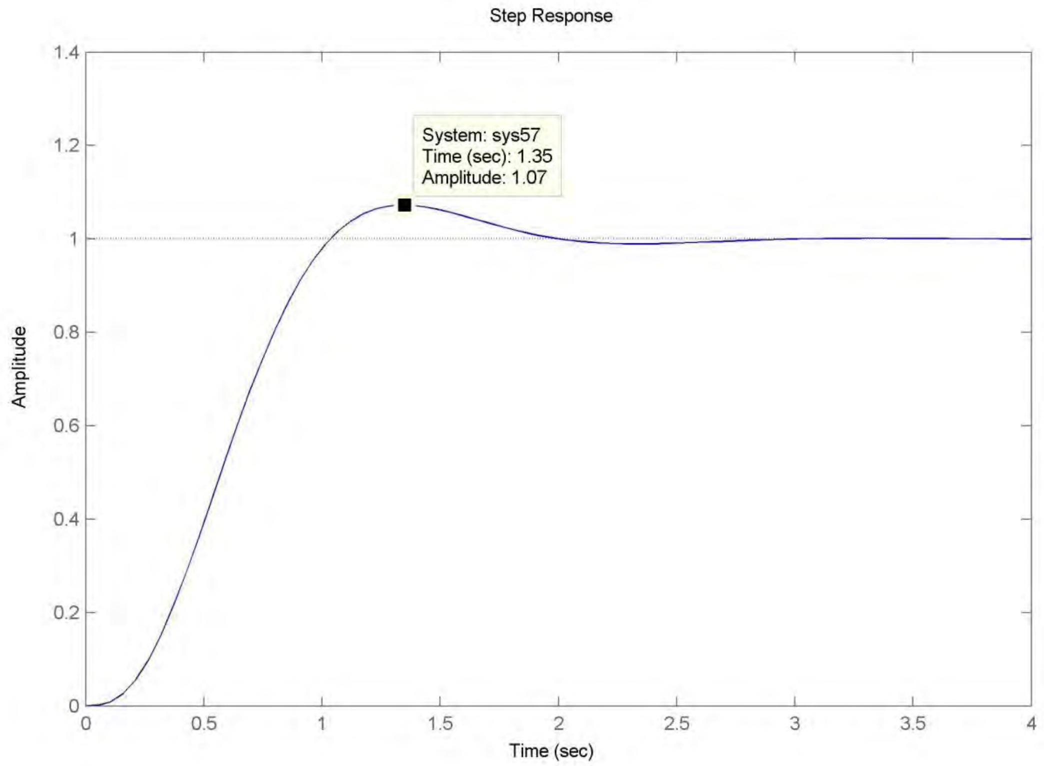
```
R = 1
Q = [1000 0 0;
0 1 0;0 0 1]
```

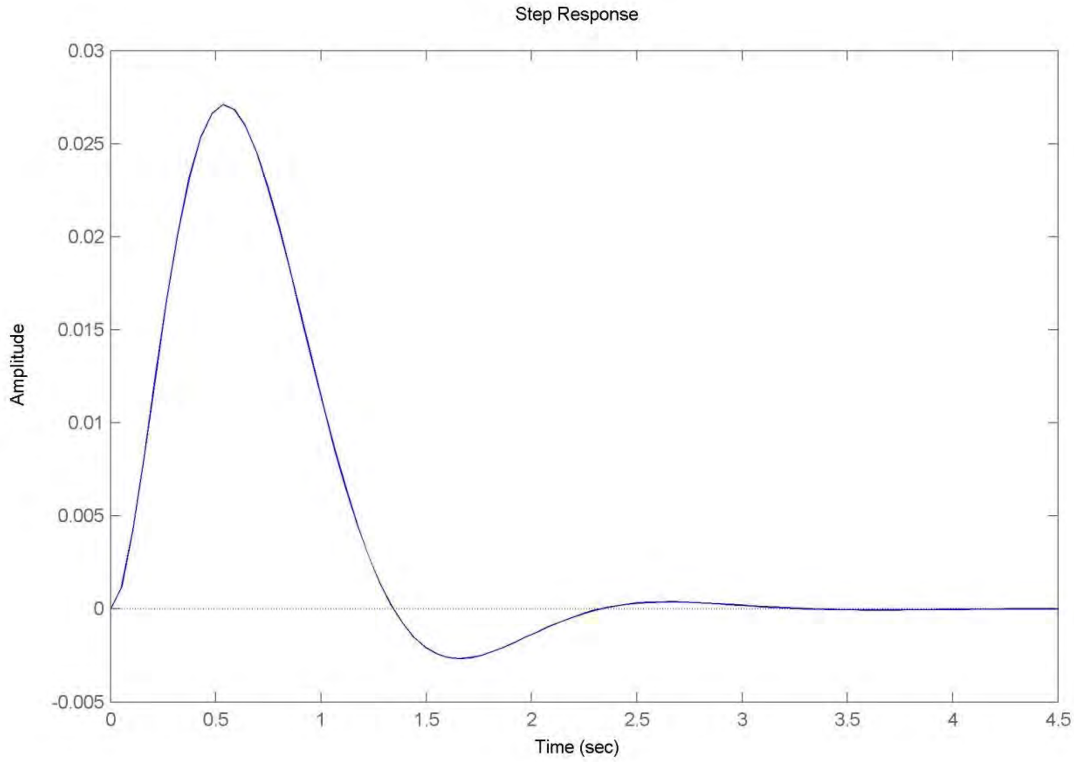
E-4

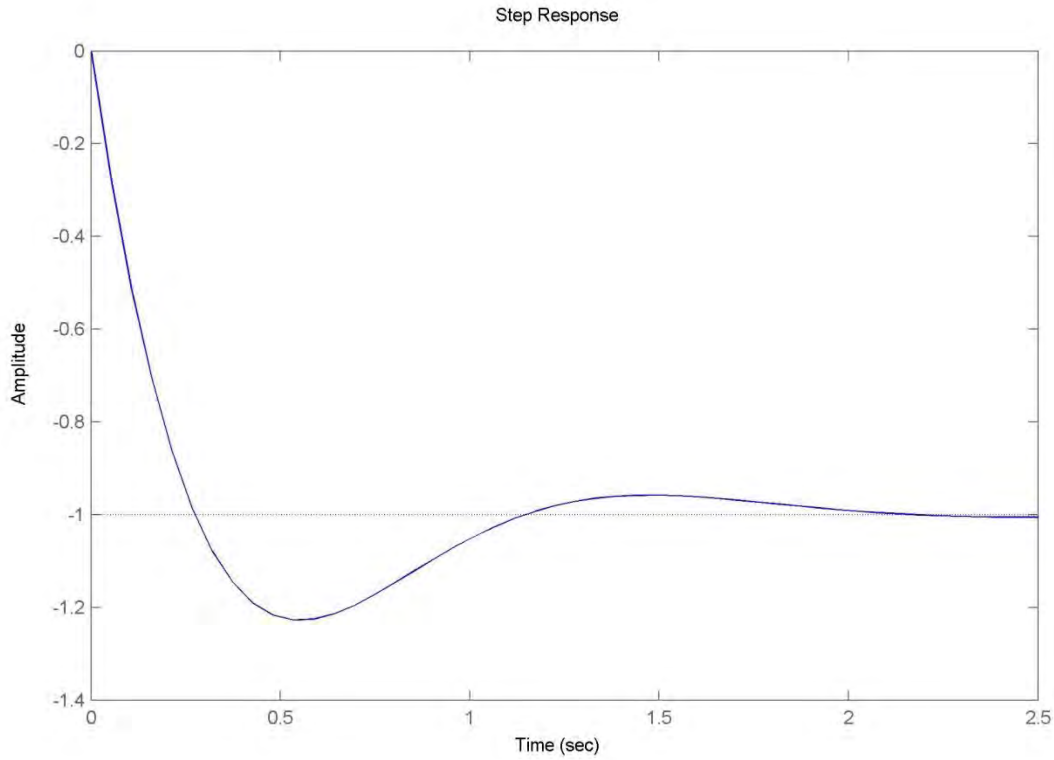
LQR Design

```
R = 1
Q = [3000 0 0;
0 1 0;0 0 1]
```

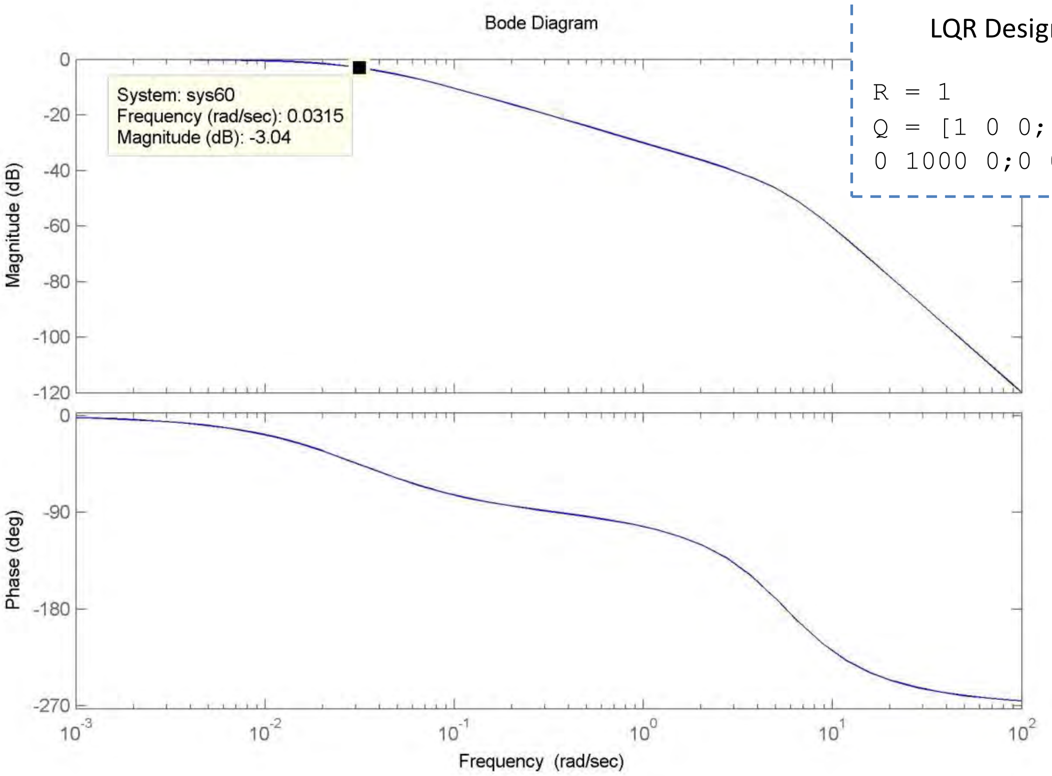





Step Response
0.03
0.025
0.02
0.015
0.01
0.005
0
-0.005
Amplitude
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
Time (sec)


Step Response
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
Amplitude
0
0.5
1
1.5
2
2.5
Time (sec)

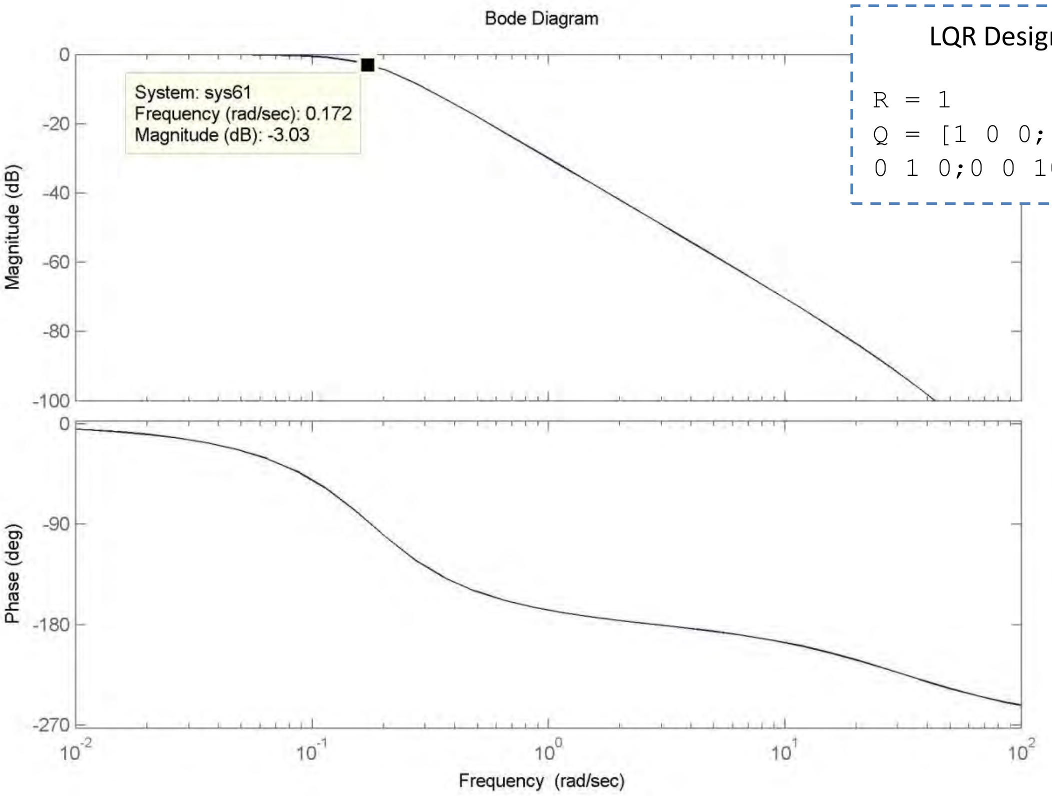
E-5



LQR Design

```
R = 1
Q = [1 0 0;
0 1000 0;0 0 1]
```

E-6

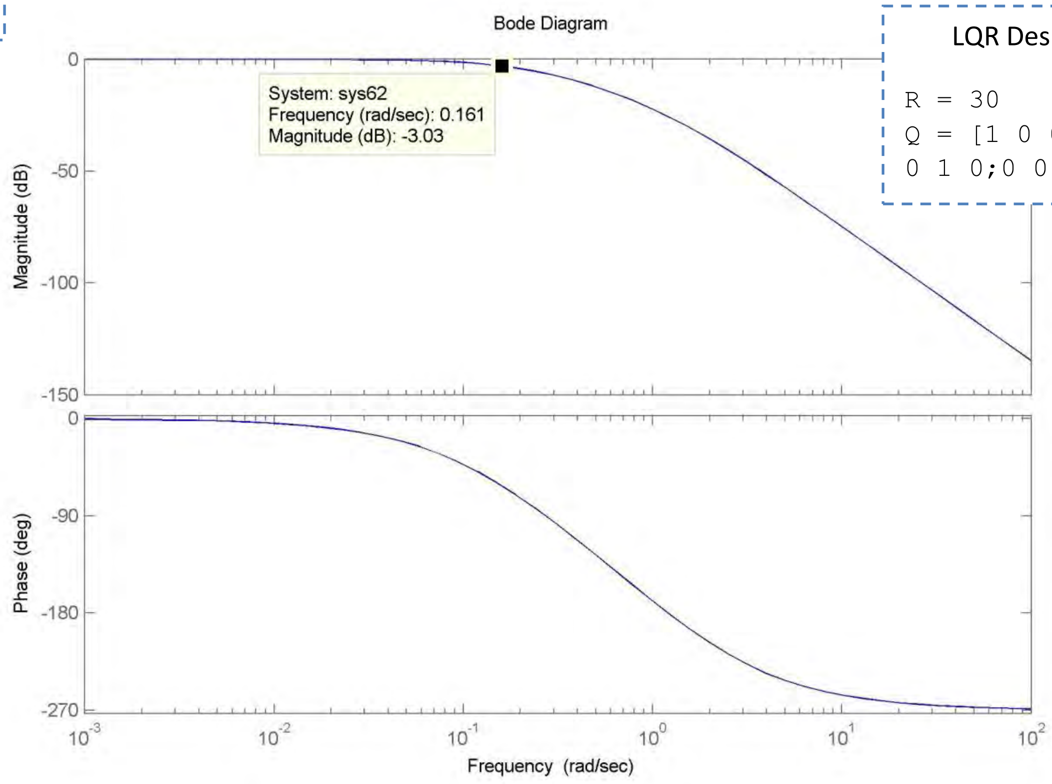


LQR Design

```
R = 1
Q = [1 0 0;
0 1 0;0 0 1000]
```

PHANG Swee King
U066584J

E-7



LQR Design

```
R = 30
Q = [1 0 0;
0 1 0;0 0 1]
```

Complete Lab 1 MATLAB Listing

```
% 3.1 Review of State-Variables and MATLAB
A1 = [0 1;-1.08 -2.33]; B1 = [0;1]; C1 = [1 0]; D1 = 0;
% frequency response plot from u to x1
sys1 = ss(A1,B1,C1,D1); figure(1); bode(sys1);
% frequency response plot from u to x2
% change C2 matrix to [0 1]
C2 = [0 1]; sys2 = ss(A1,B1,C2,D1); figure(2); bode(sys2);

% feedback of the closed loop system the output to the control is both x1
% and x2, hence C3 = [1 0;0 1]
C3 = [1 0;0 1]; sys3 = ss(A1,B1,C3,D1);
% choosing k1 = 0.1, k2 = 0.1
k1 = 0.1; k2 = 0.1;K = [k1,k2]; A2 = [0 0;0 0]; B2 = [0 0;0 0]; C2 = [0 0];
D2 = K; sys4 = ss(A2,B2,C2,D2);
% feedback response from u to x1 and x2
figure(3); sys5 = feedback(sys3,sys4); bode(sys5);

% 4.0 Simple State-Feedback Design
% 4.1 Design Using Ackermann's Formula
%---------------------------------------------%
%                ITAE Design                  %
%---------------------------------------------%

% let W0 = 6 since must be higher than 5 rad/s
P1 = 6*[-0.7071+0.7071*i;-0.7071-0.7071*i];
% pole placement using Ackermann's Formula
K = acker(A1,B1,P1); D2 = K; sys6 = ss(A2,B2,C2,D2);
sys7 = feedback(sys3,sys6); Ks = 1/dcgain(sys7); figure(4); bode(Ks*sys7);
% corresponding step response
figure(5); step(Ks*sys7);

%---------------------------------------------%
%               Bessel Design                 %
%---------------------------------------------%

% let W0 = 6 since must be higher than 5 rad/s
P2 = 7*[-0.8660+0.5000*i;-0.8660-0.5000*i];
% pole placement using Ackermann's Formula
K = acker(A1,B1,P2); D2 = K; sys8 = ss(A2,B2,C2,D2);
sys9 = feedback(sys3,sys8); Ks = 1/dcgain(sys9); figure(6); bode(Ks*sys9);
% corresponding step response
figure(7); step(Ks*sys9);

%---------------------------------------------%
%          2nd Order Dominant Poles           %
%---------------------------------------------%

% poles of my own selection(trial and error #1)
P3 = 6*[-1;-1];
% pole placement using Ackermann's Formula
K = acker(A1,B1,P3); D2 = K; sys10 = ss(A2,B2,C2,D2);
sys11 = feedback(sys3,sys10); Ks =
1/dcgain(sys11);figure(8);bode(Ks*sys11);
% corresponding step response
figure(9); step(Ks*sys11);

% poles of my own selection(trial and error #2)
```

```
P4 = [-6;-100];
% pole placement using Ackermann's Formula
K = acker(A1,B1,P4); D2 = K; sys12 = ss(A2,B2,C2,D2);
sys13 = feedback(sys3,sys12); Ks = 1/dcgain(sys13); figure(10);
bode(Ks*sys13);
% corresponding step response
figure(11); step(Ks*sys13);

% poles of my own selection(trial and error #3)
P5 = [-4.242+4.242*i;-4.242-4.242*i];
% pole placement using Ackermann's Formula
K = acker(A1,B1,P5); D2 = K; sys14 = ss(A2,B2,C2,D2);
sys15 = feedback(sys3,sys14); Ks = 1/dcgain(sys15); figure(12);
bode(Ks*sys15);
% corresponding step response
figure(13); step(Ks*sys15);

% poles of my own selection(trial and error #4)
P6 = [-3+5.196*i;-3-5.196*i];
% pole placement using Ackermann's Formula
K = acker(A1,B1,P6); D2 = K; sys16 = ss(A2,B2,C2,D2);
sys17 = feedback(sys3,sys16); Ks = 1/dcgain(sys17);
figure(14); bode(Ks*sys17);
% corresponding step response
figure(15); step(Ks*sys17);

% 4.2
%----------------------------------------------------%
% Design Using Linear Quadratic Regulator Weightings %
%----------------------------------------------------%
% set everything to gain 1
R = 1; Q = [1 0;0 1]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys18 = ss(A2,B2,C2,D2); sys19 = feedback(sys3,sys18); Ks =
1/dcgain(sys19); figure(16); bode(Ks*sys19); figure(17); step(Ks*sys19);

% set weight of x1 to be higher (100x1 + 1x2)
R = 1; Q = [100 0;0 1]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys20 = ss(A2,B2,C2,D2); sys21 = feedback(sys3,sys20); Ks =
1/dcgain(sys21);
figure(18); bode(Ks*sys21); figure(19); step(Ks*sys21);

% set weight of x1 to be even higher (200x1 + 1x2)
R = 1; Q = [200 0;0 1]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys22 = ss(A2,B2,C2,D2);sys23 = feedback(sys3,sys22); Ks = 1/dcgain(sys23);
figure(20); bode(Ks*sys23); figure(21); step(Ks*sys23);

% set weight of x1 to be much higher (1000x1 + 1x2)
R = 1; Q = [1000 0;0 1]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys24 = ss(A2,B2,C2,D2); sys25 = feedback(sys3,sys24); Ks =
1/dcgain(sys25);
figure(22); bode(Ks*sys25); figure(23); step(Ks*sys25);

% set weight of x2 to be higher (1000x1 + 50x2)
R = 1; Q = [1000 0;0 50]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys26 = ss(A2,B2,C2,D2); sys27 = feedback(sys3,sys26); Ks =
1/dcgain(sys27);
figure(24); bode(Ks*sys27); figure(25); step(Ks*sys27);

% control both the weight of x1 and x2 to meet specification
```

```
R = 1; Q = [1200 0;0 20]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys28 = ss(A2,B2,C2,D2); sys29 = feedback(sys3,sys28); Ks =
1/dcgain(sys29);
figure(26); bode(Ks*sys29); figure(27); step(Ks*sys29);

% control both the weight of x1 and x2 to meet specification
R = 1; Q = [1250 0;0 20]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys30 = ss(A2,B2,C2,D2); sys31 = feedback(sys3,sys30); Ks =
1/dcgain(sys31);
figure(28); bode(Ks*sys31); figure(29); step(Ks*sys31);

% control R (input to the system)
R = 20; Q = [1 0;0 1]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys32 = ss(A2,B2,C2,D2); sys33 = feedback(sys3,sys32); Ks =
1/dcgain(sys33);
figure(30); bode(Ks*sys33); figure(31); step(Ks*sys33);

% control R and Q to meet specification
R = 20; Q = [1000 0;0 1]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys34 = ss(A2,B2,C2,D2); sys35 = feedback(sys3,sys34); Ks =
1/dcgain(sys35);
figure(32); bode(Ks*sys35); figure(33); step(Ks*sys35);

% control R and Q to meet specification
R = 20; Q = [5000 0;0 1]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys36 = ss(A2,B2,C2,D2); sys37 = feedback(sys3,sys36); Ks =
1/dcgain(sys37);
figure(34); bode(Ks*sys37); figure(35); step(Ks*sys37);

% control R and Q to meet specification
R = 20; Q = [10000 0;0 1]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys38 = ss(A2,B2,C2,D2); sys39 = feedback(sys3,sys38); Ks =
1/dcgain(sys39);
figure(36); bode(Ks*sys39); figure(37); step(Ks*sys39);

% control R and Q to meet specification
R = 20; Q = [20000 0;0 1]; [K,S,E] = lqr(A1,B1,Q,R); D2 = K;
sys40 = ss(A2,B2,C2,D2); sys41 = feedback(sys3,sys40); Ks =
1/dcgain(sys41);
figure(38); bode(Ks*sys41); figure(39); step(Ks*sys41);

% 5.0
%----------------------------------------%
%    State-Feedback Design Including     %
%          State-Augmentation            %
%----------------------------------------%

F = [0 1 0;0 0 1; 0 -1.08 -2.33]; G = [0;0;1]; Gr = [-1;0;0]; Gv = [0;0;1];
H = [0 1 0]; J = 0;

%-------------------------------------%
%             ITAE Design             %
%-------------------------------------%
% use W0 = 4 since bandwidth must be greater than 3.5

P1 = 4*[-0.7081;-0.5210+1.068*i;-0.5210-1.068*i]; K = acker(F,G,P1);
% response of output y to a unit step in r
% assuming v = 0
sys42 = ss(F-G*K,Gr,H,J); figure(40); bode(sys42); figure(41); step(sys42);
```

```
% response of output y to a unit step in the unmeasurable disturbance v
sys43 = ss(F-G*K,Gv,H,J); figure(42); step(sys43);
% response of input u to a unit step in the unmeasurable disturbance v
H1 = -K; sys44 = ss(F-G*K,Gv,H1,J); figure(43); step(sys44);

%-------------------------------------%
%            Bassel Design            %
%-------------------------------------%
% use W0 = 6 by trial and error

P2 = 6*[-0.9420;-0.7455+0.7112*i;-0.7455-0.7112*i]; K = acker(F,G,P2);
% response of output y to a unit step in r
% assuming v = 0
sys45 = ss(F-G*K,Gr,H,J); figure(44); bode(sys45); figure(45); step(sys45);
% response of output y to a unit step in the unmeasurable disturbance v
sys46 = ss(F-G*K,Gv,H,J); figure(46); step(sys46);
% response of input u to a unit step in the unmeasurable disturbance v
H2 = -K; sys47 = ss(F-G*K,Gv,H2,J); figure(47); step(sys47);

%-------------------------------------%
%              LQR Design             %
%-------------------------------------%
% let everything start from 1
R = 1; Q = [1 0 0;0 1 0;0 0 1]; [K,S,E] = lqr(F,G,Q,R);
% response of output y to a unit step in r
sys48 = ss(F-G*K,Gr,H,J); figure(48); bode(sys48); figure(49); step(sys48);
% response of output y to a unit step in the unmeasurable disturbance v
sys49 = ss(F-G*K,Gv,H,J); figure(50); step(sys49);
% response of input u to a unit step in the unmeasurable disturbance v
H1 = -K; sys50 = ss(F-G*K,Gv,H1,J); figure(51); step(sys50);

% increase xI weight
R = 1; Q = [200 0 0;0 1 0;0 0 1]; [K,S,E] = lqr(F,G,Q,R);
% response of output y to a unit step in r
sys51 = ss(F-G*K,Gr,H,J); figure(52); bode(sys51); figure(53); step(sys51);
% response of output y to a unit step in the unmeasurable disturbance v
sys52 = ss(F-G*K,Gv,H,J); figure(54); step(sys52);
% response of input u to a unit step in the unmeasurable disturbance v
H2 = -K; sys53 = ss(F-G*K,Gv,H2,J); figure(55); step(sys53);

% increase xI weight
R = 1; Q = [1000 0 0;0 1 0;0 0 1]; [K,S,E] = lqr(F,G,Q,R);
% response of output y to a unit step in r
sys54 = ss(F-G*K,Gr,H,J); figure(56); bode(sys54); figure(57); step(sys54);
% response of output y to a unit step in the unmeasurable disturbance v
sys55 = ss(F-G*K,Gv,H,J); figure(58); step(sys55);
% response of input u to a unit step in the unmeasurable disturbance v
H3 = -K; sys56 = ss(F-G*K,Gv,H3,J); figure(59); step(sys56);

% increase xI weight
R = 1; Q = [3000 0 0;0 1 0;0 0 1]; [K,S,E] = lqr(F,G,Q,R);
% response of output y to a unit step in r
sys57 = ss(F-G*K,Gr,H,J); figure(60); bode(sys57); figure(61); step(sys57);
% response of output y to a unit step in the unmeasurable disturbance v
sys58 = ss(F-G*K,Gv,H,J); figure(62); step(sys58);
% response of input u to a unit step in the unmeasurable disturbance v
H3 = -K; sys59 = ss(F-G*K,Gv,H3,J); figure(63); step(sys59);

% the effect of weighting more heavily on x1
Q = [1 0 0;0 1000 0;0 0 1]; [K,S,E] = lqr(F,G,Q,R);
```

```
% response of output y to a unit step in r
sys60 = ss(F-G*K,Gr,H,J); figure(64); bode(sys60);

% the effect of weighting more heavily on x2
Q = [1 0 0;0 1 0;0 0 1000]; [K,S,E] = lqr(F,G,Q,R);
% response of output y to a unit step in r
sys61 = ss(F-G*K,Gr,H,J); figure(65); bode(sys61);

% the effect of weighting more heavily on input
R = 30; Q = [1 0 0;0 1 0;0 0 1]; [K,S,E] = lqr(F,G,Q,R);
% response of output y to a unit step in r
sys62 = ss(F-G*K,Gr,H,J); figure(66); bode(sys62);

%---------------------------------------------%
%          2nd Order Dominant Poles           %
%---------------------------------------------%
% random choosing a set of poles location #1
P3 = [-4;-100;-100]; K = acker(F,G,P3);
% response of output y to a unit step in r
% assuming v = 0
sys63 = ss(F-G*K,Gr,H,J); figure(67); bode(sys63); figure(68); step(sys63);
% response of output y to a unit step in the unmeasurable disturbance v
sys64 = ss(F-G*K,Gv,H,J); figure(69); step(sys64);
% response of input u to a unit step in the unmeasurable disturbance v
H3 = -K; sys65 = ss(F-G*K,Gv,H3,J); figure(70); step(sys65);

% random choosing a set of poles location #2
P4 = [-2.828+2.828*i;-2.828-2.828*i;-100]; K = acker(F,G,P4);
% response of output y to a unit step in r
% assuming v = 0
sys66 = ss(F-G*K,Gr,H,J); figure(71); bode(sys66); figure(72); step(sys66);
% response of output y to a unit step in the unmeasurable disturbance v
sys67 = ss(F-G*K,Gv,H,J); figure(73); step(sys67);
% response of input u to a unit step in the unmeasurable disturbance v
H4 = -K; sys68 = ss(F-G*K,Gv,H4,J); figure(74); step(sys68);

% random choosing a set of poles location #3
P5 = [-2+3.464*i;-2-3.464*i;-100]; K = acker(F,G,P5);
% response of output y to a unit step in r
% assuming v = 0
sys69 = ss(F-G*K,Gr,H,J); figure(75); bode(sys69); figure(76); step(sys69);
% response of output y to a unit step in the unmeasurable disturbance v
sys70 = ss(F-G*K,Gv,H,J); figure(77); step(sys70);
% response of input u to a unit step in the unmeasurable disturbance v
H5 = -K; sys71 = ss(F-G*K,Gv,H5,J); figure(78); step(sys71);
```