

Control Systems Lab Manual

Department of Instrumentation Engineering

Jorhat Engineering College

Marks distribution:

Categories		Marks
<i>End semester exam :</i>		
Aim and Theory	:	05
Write the program	:	10
Program execution	:	05
Result and discussion	:	05
Viva (External)	:	10
Total (semester)	:	35
<i>Internal assessment:</i>		
Attendance	:	05
Lab performance	:	05
Lab journal	:	05
Total (IA)	:	15
Total (End semester + AI)	:	50

Course outcomes (CO):

CO1	:	Build in depth knowledge on MATLAB software
CO2	:	Analyze the concept of time response and frequency response of the systems
CO3	:	Design controllers, compensators using MATLAB software and examine their performances
CO4	:	Make use of team-based laboratory activities, students will able to interact with fellow students, and complete assignments

CO-PO mapping:

Course Outcome	Program Outcome												PS01	PS02
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12		
CO1	1	1	1		1							1		
CO2	2	2	2	2	2							1		3
CO3	2	2	2	2	2	1						1		
CO4						1	1	1	2	1		1		

List of the experiments:

Exp. no.	Name of the experiments	Page no.
1.	Familiarization with MATLAB coding language and SIMULINK tool box	4
2.	Develop step and impulse responses of a linear first-order systems	7
3.	Develop step and impulse responses of a linear second-order system	10
4.	Illustrate steady state errors for a type '0', type '1', type '2' systems using MATLAB	13
5.	Construct Root locus plot of a second-order system using MATLAB	15
6.	Experiment with Bode plot of a second-order system using MATLAB	17
7.	Construct Nyquist plot of a second-order system using MATLAB	19
8.	Examine the effect of a forward-path lead compensator of a linear feedback control system	21
9.	Analyze the effect of addition of poles to the forward path transfer function of a closed loop system	25
10.	Analyze the effect of addition of zeros to the forward path transfer function of a closed loop system	27

Exp. 1: Familiarization with MATLAB coding language and SIMULINK tool box

OBJECTIVE : To obtain

- I. Pole, zero, gain values from a given transfer function
- II. Transfer function model from pole, zero, gain values
- III. Pole, zero plot of a transfer function

THEORY:

A transfer function is also known as the network function is a mathematical representation, in terms of spatial or temporal frequency, of the relation between the input and output of a (linear time invariant) system. The transfer function is the ratio of the output Laplace Transform to the input Laplace Transform assuming zero initial conditions. Many important characteristics of dynamic or control systems can be determined from the transfer function. The transfer function is commonly used in the analysis of single-input single-output electronic system, for instance. It is mainly used in signal processing, communication theory, and control theory. The term is often used exclusively to refer to linear time-invariant systems (LTI). In its simplest form for continuous time input signal $x(t)$ and output $y(t)$, the transfer function is the linear mapping of the Laplace transform of the input, $X(s)$, to the output $Y(s)$.

Zeros are the value(s) for z where the numerator of the transfer function equals zero. The complex frequencies that make the overall gain of the filter transfer function zero.

Poles are the value(s) for z where the denominator of the transfer function equals zero. The complex frequencies that make the overall gain of the filter transfer function infinite.

The general procedure to find the transfer function of a linear differential equation from input to output is to take the Laplace Transforms of both sides assuming zero conditions, and to solve for the ratio of the output Laplace over the input Laplace.

The transfer function provides a basis for determining important system response characteristics without solving the complete differential equation. As defined, the transfer function is a rational function in the complex variable 's' that is It is often convenient to factor the polynomials in the numerator and the denominator, and to write the transfer function in terms of those factors:

$$G(s) = \frac{N(s)}{D(s)} = K \frac{(s-z_1)(s-z_2)\dots\dots\dots(s-z_n)}{(s-p_1)(s-p_2)\dots\dots\dots(s-p_m)}$$

where, the numerator and denominator polynomials, $N(s)$ and $D(s)$.

The values of s for which $N(s) = 0$, are known as zeros of the system. i.e; at $s = z_1, z_2, \dots, z_n$.

The values of s for which $D(s) = 0$, are known as poles of the system. i.e; at $s = p_1, p_2, \dots, p_n$.

Example 1: Obtain pole, zero & gain values of a transfer function $G(s) = \frac{s^2+4s+3}{(s+5)(s^2+4s+7)}$. Also obtain

pole zero plot.

```
num = [1 4 3]
den= conv([1 5], [3 4 7])
g = tf (num,den)
[z,p,k] = tf2zp(num,den)
pzmap(g)
```

Output :

Transfer function:
 $s^2 + 4 s + 3$

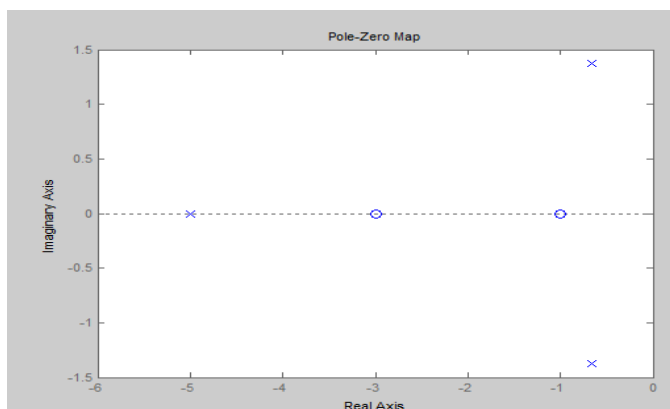
 $3 s^3 + 19 s^2 + 27 s + 35$

z =
 -3
 -1

p =
 -5.0000
 -0.6667 + 1.3744i
 -0.6667 - 1.3744i

k =
 0.3333

Pole Zero Plot:



Example 2: Write matlab code to obtain transfer function of a system from its pole ,zero, gain values. Assume pole locations are -2, 1, zero at -1 and gain is 7.

MATLAB Code :

```
p= [-2 1]
z= [-1]
```

```

k=7
[num,den]=zp2tf(z',p',k)
g=tf(num,den)

```

Output:

Transfer function:

$$\frac{7s + 7}{s^2 + s - 2}$$

Assignments:

Obtain Pole, zero, gain values of the transfer functions given below. **Also verify your result theoretically**

$$1. G(s) = \frac{1}{s^2+s+4} \quad 2. G(s) = \frac{5}{s^2+9} \quad 3. G(s) = \frac{1}{(s^2+3s+5)(s+3)(s+5)}$$

DISCUSSION:

1. What do you mean by poles?
2. What is the significance of transfer function?

Exp. 2: Develop step and impulse responses of a linear first-order systems

OBJECTIVE : To determine

- I. Step response of 1st order system
- II. Impulse response of 1st order system

THEORY: A first order system is one in which highest power of s in denominator if transfer function defines order of the system.

For first order system,

$$\frac{C(s)}{R(s)} = \frac{1}{sT+1}$$

$$C(s) = \frac{1}{sT+1} R(s) \dots\dots\dots (1)$$

Since the laplace transform of the unit step function is 1/s , substituting R(s)=1/s in equation (1)

$$C(s) = \frac{1}{sT+1} * \frac{1}{s}$$

Expanding C(s) into partial fractions gives ,

$$C(s) = \frac{1}{s} - \frac{1}{sT+1}$$

$$C(s) = \frac{1}{s} - \frac{1}{s + (1/T)} \dots\dots\dots (2)$$

Taking the inverse laplace transform of equation (2), we get

$$C(t) = 1 - e^{-t/T} \quad \text{for } t \geq 0 \dots\dots\dots (3)$$

Equation (3) shows that initially (when t=0), the output c(t) is zero and finally (t→ ∞) e^{-t/T} is zero and the output c(t) becomes unity .

At t = T,

$$C(t) = 1 - e^{-1} = 1 - 0.368 = 0.632$$

That's , the output response has reached 63.2 % of it's final value . T is known as the time constant . Thus , the time constant T is defined as the time required for the output response to attain 63.2% of its final value or steady state value .

Equation (3) shows that the response curve is exponential in nature as shown on figure.

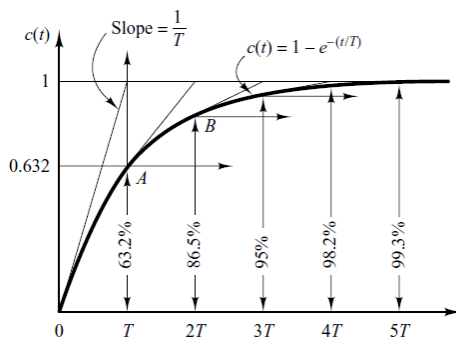


Fig. : Response curve of first order unit step input

Response of the first order system with unit impulse response

For the unit – impulse input

$$R(s) = 1$$

$$C(s) = \frac{1}{sT+1} * R(s) \dots\dots\dots (1)$$

Substituting the value of R(s) = 1 in equation (1), we get

$$C(s) = \frac{1}{sT+1} * 1$$

$$C(s) = \frac{1}{T} * \frac{1}{s+1/T} \dots\dots\dots(2)$$

Taking the inverse laplace transform of the equation of (2) , we get the output response as

$$C(t) = \frac{1}{T} e^{-t/T} \text{ for } t \geq 0 \dots\dots\dots (3)$$

The output response curve shown in the figure

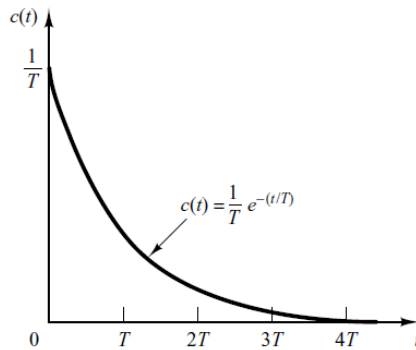


Fig. : Unit impulse response of first order system

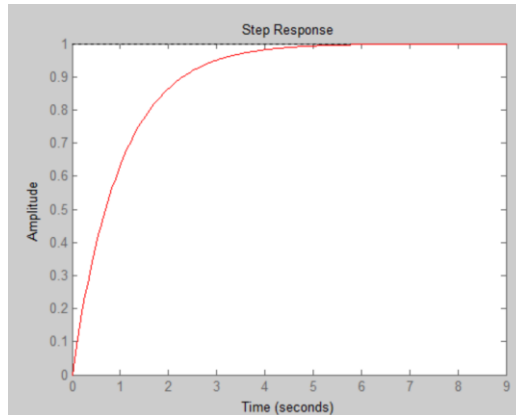
Example 1: Obtain step response of a unity feedback system having forward path transfer function of

$$G(s) = \frac{1}{s}$$

Matlab Code:

```
num = [1];
den = [1 0];
g = tf (num,den);
p = feedback(g,1)
step(p, 'r')
```

Output:

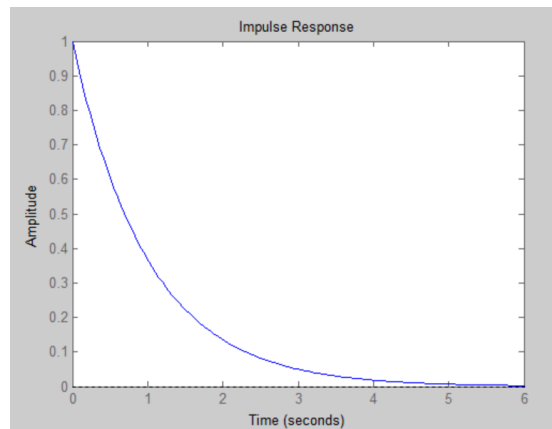


Example 2: Obtain impulse response of a unity feedback system having forward path transfer function of $G(s) = \frac{1}{s}$

Matlab Code :

```
num = [1];
den = [1 0]
g = tf (num,den)
p = feedback(g,1)
impulse(p, 'b')
```

Output:



Assignments:

Obtain step and impulse response of the following systems with unity feedback. Also verify your result theoretically

- i) $G(s) = \frac{2}{s}$
- ii) $G(s) = \frac{5}{s}$

DISCUSSION:

1. Explain why a series RL circuit with high inductance has a slow response?

Exp. 3: Develop step and impulse responses of a linear second-order system

OBJECTIVE : To determine

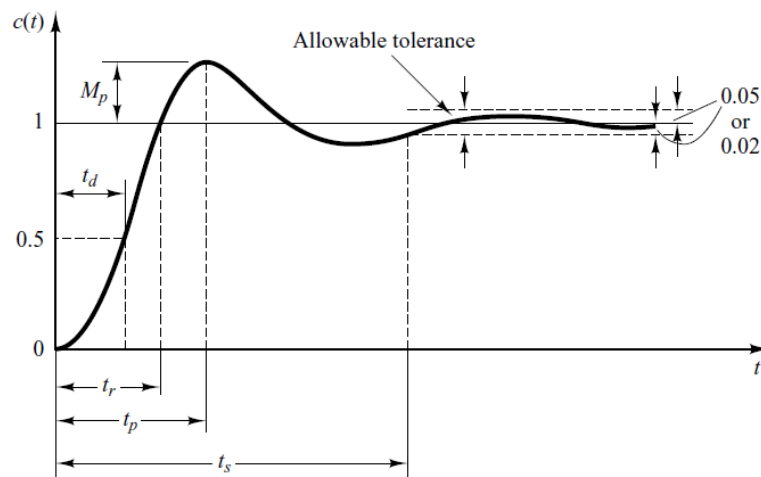
- I. Step response of 2nd order system
- II. Impulse response of 2nd order system

THEORY: The time response has utmost importance for the design and analysis of control systems because these are inherently time domain systems where time is independent variable. During the analysis of response, the variation of output with respect to time can be studied and it is known as time response. To obtain satisfactory performance of the system with respect to time must be within the specified limits. From time response analysis and corresponding results, the stability of system, accuracy of system and complete evaluation can be studied easily. Due to the application of an excitation to a system, the response of the system is known as time response and it is a function of time. The two parts of response of any system:
Transient response; Steady-state response.

Time Response Specification Parameters: The transfer function of a 2-nd order system is generally represented by the following transfer function:

$$\frac{Y(S)}{R(S)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

The dynamic behavior of the second-order system can then be described in terms of two parameters: the damping ratio and the natural frequency. If the damping ratio is between 0 and 1, the system poles are complex conjugates and lie in the left-half s plane. The system is then called **underdamped**, and the transient response is **oscillatory**. If the damping ratio is equal to 1 the system is called **critically damped**, and when the damping ratio is larger than 1 we have **overdamped** system.



Delay time (td)

The delay time is the time required for the response to reach half the final value the very first time.

Rise time (tr)

The rise time is the time required for the response to rise from 10% to 90%, 5% to 95%, or 0% to 100% of its final value. For underdamped second-order systems, the 0% to 100% rise time is normally used. For overdamped systems, the 10% to 90% rise time is commonly used.

Peak time (tp)

The peak time is the time required for the response to reach the first peak of the overshoot.

Maximum (percent) overshoot (Mp)

The maximum overshoot is the maximum peak value of the response curve measured from unity. If the final steady-state value of the response differs from unity, then it is common to use the maximum percent overshoot. It is defined by

$$\frac{y(c_p) - c(\infty)}{\infty} \times 100\%$$

Settling time (ts)

The settling time is the time required for the response curve to reach and stay within a range about the final value of size specified by absolute percentage of the final value (usually 2% or 5%). The settling time is related to the largest time constant of the control system.

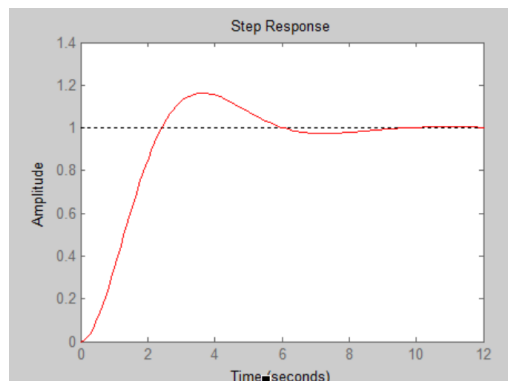
Example 1: Obtain step response of a unity feedback system having forward path transfer function of

$$G(s) = \frac{1}{s(s+1)}$$

Matlab Code:

```
num = [1];
den = [1 1 0];
g = tf (num,den);
p = feedback(g,1)
step(p, 'r');
stepinfo (p)
```

Output:



RiseTime: 1.6390
SettlingTime: 8.0759
SettlingMin: 0.9315
SettlingMax: 1.1629
Overshoot: 16.2929
Undershoot: 0
Peak: 1.1629
PeakTime: 3.5920

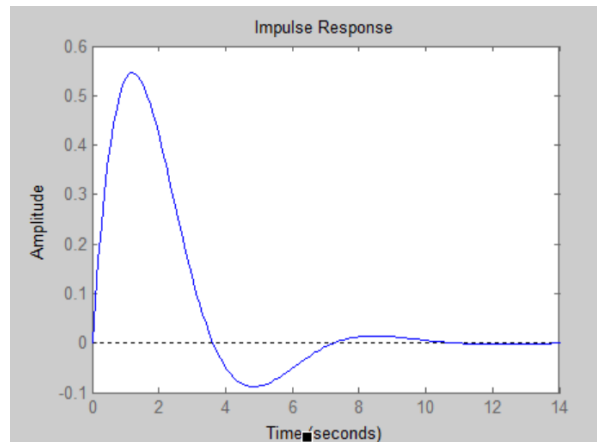
Example 2: Obtain impulse response of a unity feedback system having forward path transfer function

$$\text{of } G(s) = \frac{1}{s^2+s+4}$$

Matlab Code :

```
num = [1];  
den = [1 1 0]  
g = tf (num,den)  
p = feedback(g,1)  
impulse(p, 'b')
```

Output:



Assignments:

Obtain step and impulse response of the following systems with unity feedback connection. Also verify your result theoretically

- i) $G(s) = \frac{2}{s(s+1)}$
- ii) $G(s) = \frac{1}{s(s+10)}$

DISCUSSION:

1. What do you mean by rise time? Derive its expression for a unity feedback 2nd order control system.
2. Why is less overshoot desired for practical systems?

Exp. 4: Illustrate steady state errors for a type ‘0’, type ‘1’, type ‘2’ systems using MATLAB

OBJECTIVE : To determine

- I. Step response of Type ‘0’ system
- II. Impulse response of Type ‘0’ system
- III. Step response of Type ‘1’ system
- IV. Impulse response of Type ‘1’ system
- V. Step response of Type ‘2’ system
- VI. Impulse response of Type ‘2’ system

THEORY:

A real time system can be expressed by its transfer function. Based on presence of poles at origin of s plane , transfer functions can be classified as Type ‘0’, Type ‘1’, Type ‘2’ Systems. Let open loop transfer function a system is expressed as:

$$G(s)H(s) = \frac{k(1+sT_a)(1+sT_b)}{s^N(1+sT_1)(1+sT_2)} \tag{i}$$

From equation (i) it is clear that ‘N’ determines the number of poles at origin.

For a Type ‘0’ system, N = 0

For a Type ‘1’ system, N = 1

For a Type ‘2’ system, N = 2

⋮
⋮
⋮
⋮

For a Type ‘N’ system, N = N

The steady state error can be found out by the following equation

$$e_{ss} = \lim_{s \rightarrow \infty} \frac{sR(s)}{1 + G(s)H(s)}$$

Type	Step Input		Ramp input		Parabolic input	
	K _P	e _{ss}	K _V	e _{ss}	K _a	e _{ss}
0	K	$\frac{1}{1+K}$	0	∞	0	∞
1	∞	0	K	$\frac{1}{K}$	0	∞
2	∞	0	∞	0	K	$\frac{1}{K}$

Example 1: Obtain step response of a Type '0' system having forward path transfer function of $G(s) =$

$$\frac{1}{s^2+s+4}$$

Matlab Code :

```
num = [1];  
den = [1 1 4]  
g = tf (num,den)  
p = feedback(g,1)  
step(p, 'r')
```

Example 2: Obtain step response of a Type '1' system having forward path transfer function of $G(s) =$

$$\frac{1}{s(s^2+s+4)}$$

Matlab Code :

```
num = [1];  
den = [1 1 4 0];  
g = tf (num,den)  
p = feedback(g,1)  
step(p, 'r')
```

Example 3: Obtain Step response of a Type '2' system having forward path transfer function of $G(s) =$

$$\frac{1}{s^2(s^2+s+4)}$$

Matlab Code :

```
num = [1];  
den = [1 1 4 0 0]  
g = tf (num,den)  
p = feedback(g,1)  
step(t, 'r')
```

DISCUSSION:

- 1. What would be steady state error for a type 1 system if unit ramp input is applied?**

Exp. 5: Construct Root locus plot of a second-order system using MATLAB

OBJECTIVE : To determine Root Locus plot of a 2nd order system

THEORY: rlocus computes the Evans root locus of a SISO open-loop model. The root locus gives the closed-loop pole trajectories as a function of the feedback gain k (assuming negative feedback). Root loci are used to study the effects of varying feedback gains on closed-loop pole locations. In turn, these locations provide indirect information on the time and frequency responses. rlocus(sys) calculates and plots the root locus of the open-loop SISO model sys. This function can be applied to any of the following feedback loops by setting sys appropriately. If sys has transfer function $G(s) = \frac{N(s)}{D(s)}$, The closed-loop poles are the roots of $d(s) + k*n(s)=0$

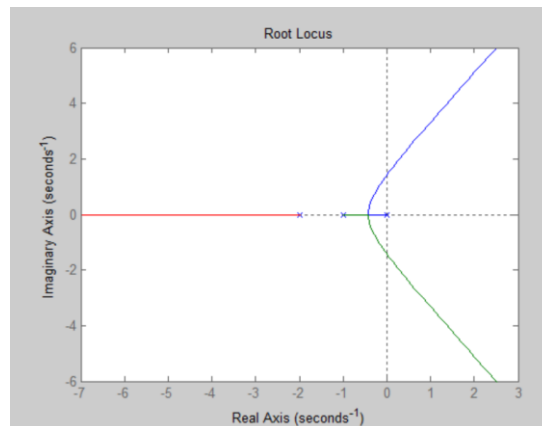
Example 1: Obtain Root Locus Plot of a system having forward path transfer function of

$$G(s) = \frac{1}{s(s+1)(s+1)}$$

Matlab Code:

```
g = tf (1 , [1 0]) * tf(1, [1 1]) * tf (1, [1 2]);  
rlocus(g)
```

Output :

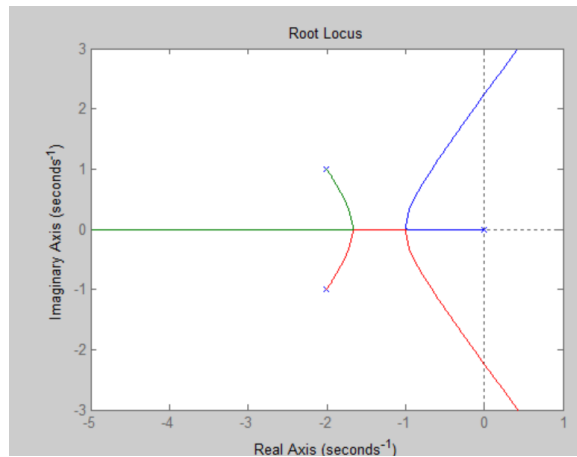


Example 2: Obtain Root Locus Plot of a system having forward path transfer function of

Matlab Code :

```
g = tf (1 , [1 0]) * tf(1, [1 4 5])  
rlocus(g);
```

Output :



Assignment:

Obtain Root Locus Plot of the following transfer function. Also verify your result theoretically.

i)
$$G(s) = \frac{1}{s(s+2)(s^2+2s+5)}$$

ii)
$$G(s) = \frac{1}{s(s+1)(s^2+4s+5)}$$

DISCUSSION:

1. What is relative stability? How can you measure relative stability using Root locus?
2. What do you mean by break away point?
3. What is asymptotic line?

Exp. 6: Experiment with Bode plot of a second-order system using MATLAB

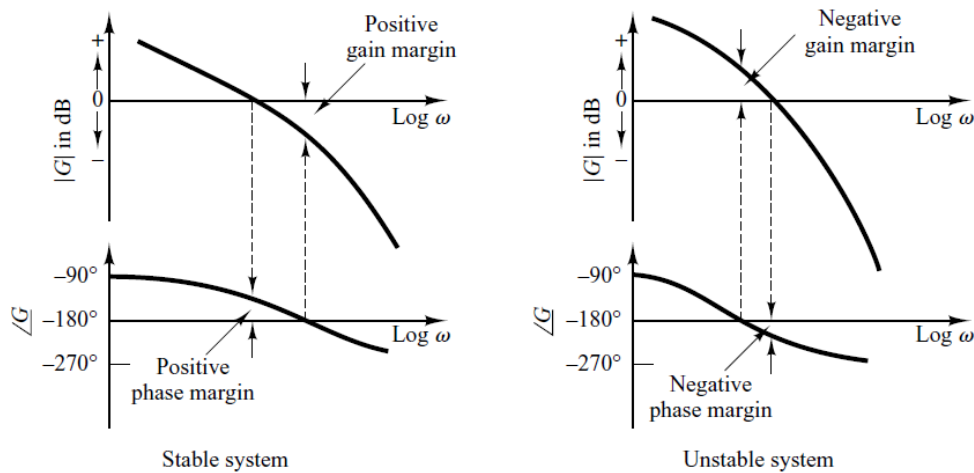
OBJECTIVE : To determine

- I. Bode plot of a 2nd order system
- II. Frequency domain specification parameters

THEORY: The frequency response method may be less intuitive than other methods you have studied previously. However, it has certain advantages, especially in real-life situations such as modeling transfer functions from physical data. The frequency response of a system can be viewed two different ways: via the Bode plot or via the Nyquist diagram. Both methods display the same information; the difference lies in the way the information is presented. We will explore both methods during this lab exercise. The frequency response is a representation of the system's response to sinusoidal inputs at varying frequencies. The output of a linear system to a sinusoidal input is a sinusoid of the same frequency but with a different magnitude and phase. The frequency response is defined as the magnitude and phase differences between the input and output sinusoids. In this lab, we will see how we can use the open-loop frequency response of a system to predict its behavior in closedloop. To plot the frequency response, we create a vector of frequencies (varying between zero or "DC" and infinity i.e., a higher value) and compute the value of the plant transfer function at those frequencies. If $G(s)$ is the open loop transfer function of a system and ω is the frequency vector, we then plot $G(j\omega)$ vs. ω . Since $G(j\omega)$ is a complex number, we can plot both its magnitude and phase (the Bode plot) or its position in the complex plane (the Nyquist plot).

The gain margin is defined as the change in open loop gain required to make the system unstable. Systems with greater gain margins can withstand greater changes in system parameters before becoming unstable in closed loop.

The phase margin is defined as the change in open loop phase shift required to make a closed loop system unstable.



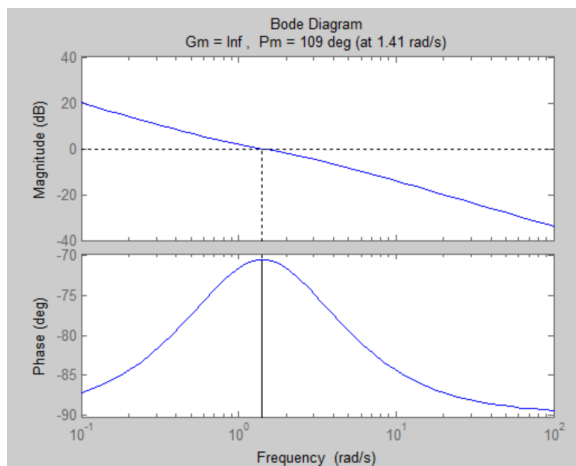
Example 1: Obtain Bode Plot of the system having forward path transfer function of

$$G(s) = \frac{s + 1}{s(0.5s + 1)}$$

Matlab Code:

```
num = [1 1]
den = conv([1 0], [.5 1]);
g = tf(num, den);
bode(g);
margin(g);
```

Output :



Assignments:

Obtain bode diagram of the following transfer function. Also verify your result theoretically.

$$G(s) = \frac{1}{s(s + 1)(5s + 1)}$$

DISCUSSION:

1. What do you mean by GM & PM?
2. How GM & PM affects system?
3. What is gain cross over frequency?

Exp. 7: Construct Nyquist plot of a second-order system using MATLAB

OBJECTIVE : To determine nyquist plot of a 2nd order system

THEORY: A stability test for time invariant linear systems can also be derived in the frequency domain. It is known as Nyquist stability criterion. It is based on the complex analysis result known as Cauchy's principle of argument. Note that the system transfer function is a complex function. By applying Cauchy's principle of argument to the open-loop system transfer function, we will get information about stability of the closed-loop system transfer function and arrive at the Nyquist stability criterion (Nyquist, 1932). The importance of Nyquist stability lies in the fact that it can also be used to determine the relative degree of system stability by producing the so-called phase and gain stability margins. These stability margins are needed for frequency domain controller design techniques.

We present only the essence of the Nyquist stability criterion and define the phase and gain stability margins. The Nyquist method is used for studying the stability of linear systems with pure time delay. For a SISO feedback system the closed-loop transfer function is given by:

$$T(s) = \frac{G(s)}{1 + G(s)H(s)}$$

Since the system poles are determined as those values at which its transfer function becomes infinity, it follows that the closed-loop system poles are obtained by solving the following equation.

In the following we consider the complex function

$$D(s) = 1 + G(s)H(s)$$

whose zeros are the closed-loop poles of the transfer function. In addition, it is easy to see that the poles of $D(s)$ are the zeros of $T(s)$. At the same time the poles of are the open-loop control system poles since they are contributed by the poles of which can be considered as the open-loop control system transfer function—obtained when the feedback loop is open at some point. The Nyquist stability test is obtained by applying the Cauchy principle of argument to the complex function. First, we state Cauchy's principle of argument.

Let $F(s)$ be an analytic function in a closed region of the complex plane except at a finite number of points. It is also assumed that $F(s)$ is analytic at every point on the contour. Then, as travels around the contour in the -plane in the clockwise direction, the function $F(s)$ encircles the origin in the plane in the same direction N times, given by $N = Z - P$

Where Z and P stand for the number of zeros and poles (including their multiplicities) of the function inside the $F(s)$ contour.

The above result can be also written as

$$\text{Arg}(F(s)) = 2\pi N$$

Nyquist Criterion:

It states that the number of unstable closed-loop poles is equal to the number of unstable open-loop poles plus the number of encirclements of the origin of the Nyquist plot of the complex function $D(s)$. This can be easily justified by applying Cauchy's principle of argument to the function $D(s)$ with the -plane contour. Note that and represent the numbers of zeros and poles, respectively, of in the unstable part of the complex plane. At

the same time, the zeros of $D(s)$ are the closed-loop system poles, and the poles of $D(s)$ are the open-loop system poles (closed-loop zeros).

The above criterion can be slightly simplified if instead of plotting the function $D(s) = 1 + G(s)H(s)$

, we plot only the function $G(s)H(s)$ and count encirclement of the Nyquist plot of around the point $(-1+j0)$, so that the modified Nyquist criterion has the following form. The number of unstable closed-loop poles (Z) is equal to the number of unstable open-loop poles (P) plus the number of encirclements (N) of the point $(-1+j0)$,

$$Z=N+P$$

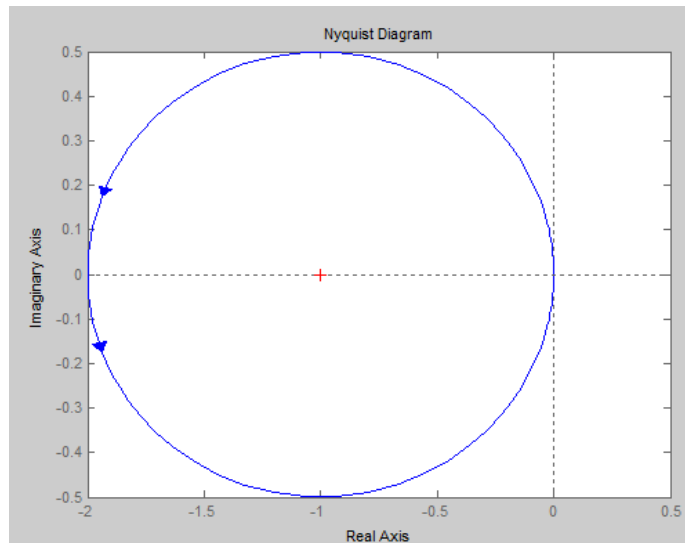
Example : Obtain Nyquist Plot of a system having forward path transfer function of

$$G(s) = \frac{s + 2}{(s + 1)(s - 1)}$$

Matlab Code:

```
num = [1 2]
den = conv([1 1], [1 -1])
g = tf(num,den);
nyquist(g)
```

Output :



Assignment:

Obtain Nyquist Plot of the following transfer function. Also verify your result theoretically.

$$G(s) = \frac{s + 1}{(s + 1)(s + 3)}$$

DISCUSSION:

1. What is nyquist contour?
2. What is polar plot? How it differs from Nyquist plot?

Exp. 8: Examine the effect of a forward-path lead compensator of a linear feedback control system

Objective:

1. To measure the gain and phase margins of an uncompensated linear 3rd order, type-1 system from its Bode plot.
2. To study the bode plot of a suggested forward-path lead compensator.
3. To recalculate the gain and phase margins of the compensated system.
4. To compare the step responses of the unity-feedback uncompensated and compensated systems.

Familiarization with the system:

The original or uncompensated open-loop system is of 3rd-order, type-1 variety and, therefore, when operated with unity negative feedback will have a zero steady-state error for a step input, irrespective of the value of the gain K . However, since the velocity error coefficient K_v of the system equals K (prove this), the steady-state error corresponding to a unit ramp input is $1/K$. To attain a given value of this error, say

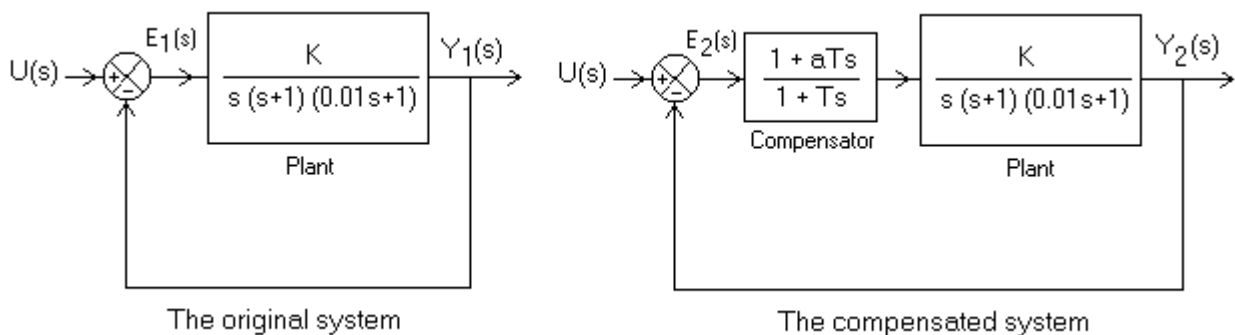


Fig. 1

Fig. 2

0.01, a K of 100 is required. Using the Routh-Hurwitz criterion, the value of the critical gain K_c , for which the system is critically stable, can be determined. It will be seen that since K_c is very close to 100, the closed-loop system will exhibit very poor relative stability. It is manifested in the form of very low *gain margin* (GM) and *phase margin* (PM) in the frequency domain and also in the form of a highly oscillatory step response in the time domain.

One popular way of alleviating the stability problem, without compromising the given velocity error specification, is to add a lead compensator block with a transfer function $G_c(s) = \frac{1+aTs}{1+Ts}$, $a > 1$, before the plant. Verify that this modification does not change the value of K_v , and thus the value of the steady state velocity error. There are suitable design methods to determine values of 'a' and 'T', the two parameters of the compensator, in order that given GM and PM specifications are met. Without going into the design techniques, we will use $a = 10$ and $T = 0.015$ (second) for this experiment.

Procedure:

1. Click on the MATLAB icon on the desktop to invoke the MATLAB program. In command window, go to file | new | m-file to open the MATLAB editor/debugger. Copy the first nine lines of the sample program given at the end of this section. Use the file menu of the editor to save the code with a filename `expt2.m`. Now the program can be executed by going to tools | run in the editor window. By placing the crosshair at appropriate positions measure the values of gain crossover frequency, phase margin, phase crossover frequency and gain margin of the uncompensated system.

Reduce the plant gain K to 50 ($= 100/2$) now. Observe that the phase-vs-frequency plot has remained unchanged, but the gain-vs-frequency plot has shifted downwards by 6 dB ($= 20 \log 2$). Again measure the values of gain crossover frequency, phase margin, phase crossover frequency and gain margin. Record the data in the present and the previous paragraph in a single table.

2. Restore the value of K to 100 now. Copy nine more lines of the sample program. Run the program and make a free-hand sketch of the Bode plot of the lead compensator. Record the values of very low frequency gain, very high frequency gain, maximum phase lead ϕ_m and the corresponding angular frequency ω_m .
3. Extend the program upto line 26 and run it. Observe the Bode plot of the compensated system and measure the values of gain crossover frequency, phase margin, phase crossover frequency and gain margin.
4. Copy the remaining lines of the sample program and deactivate all the *ginput* and *pause* commands. Observe the unit step responses of the original and the compensated systems. Approximately measure the maximum peak overshoots of the two systems.
5. Insert *printsys/ tf* commands at suitable places to record the closed-loop transfer functions of the original and compensated systems. Observe that the order of the system has increased after compensation and yet the system has better relative stability.

MATLAB code:

```
clc;
clear all;
close all;

% Expt 6 - Control Lab % (Line 1)

nump = [100]; % Specifies the plant TF numerator
denp = [0.01 1.01 1 0]; % Specifies the plant TF denominator
p1 = tf (nump, denp) % TF of the plant
[GM, PM, wcg, wcp] = margin (p1) % Margins of the plant
w = logspace (-1, 4, 100);
bode (nump, denp, w); % Bode plot the plant
grid on;
ginput % (Line 9)
pause;

numc = [0.15 1]; % Numerator of the compensator
denc = [0.015 1]; % Denominator of the compensator
p2 = tf (numc, denc) % TF of the compensator
[GM, PM, wcg, wcp] = margin (p2) % Margins of the compensator
figure (2)
bode (numc, denc, w); % Bode plot the compensator
grid on;
ginput % (Line 18)
pause;

% The open-loop cascaded system
[num, den] = series (numc, denc, nump, denp);
p3 = tf(num, den)

% Margins of the open-loop cascaded system
[GM, PM, wcg, wcp] = margin (p3)
figure (3)
bode (num, den, w);
grid on;
ginput % (Line 26)
pause;

% The uncompensated closed-loop system (Fig. 1)
[n1, d1] = feedback (nump, denp, 1, 1, -1);

% TF of the uncompensated closed-loop system
p4 = tf (n1, d1)
t= 0: 0.01: 1.5;

% Step response of uncompensated closed-loop system
y1 = step (n1, d1, t);
```

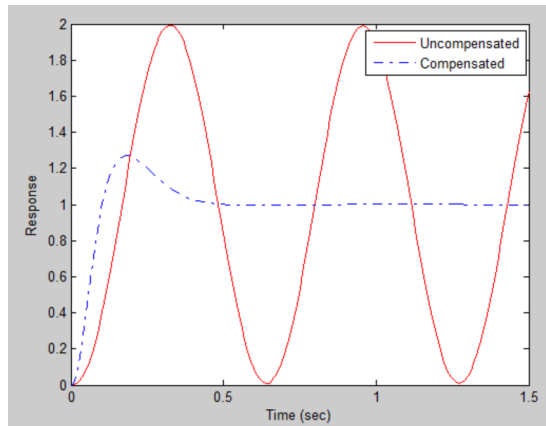
```

% The compensated closed-loop system (Fig. 2)
[n2, d2] = feedback (num, den, 1, 1, -1);

% TF of the compensated closed-loop system
p5= tf (n2, d2)

% Step response of compensated closed-loop system
y2 = step (n2, d2, t);
figure (4)

% Plot step responses of uncompensated and compensated closed-loop
systems
plot (t, y1, 'r-', t, y2, '-.')
```



Report:

1. Reproduce all the data in tabular form and the free-hand sketch of the Bode plot of the compensator obtained in step 2 of procedure. Compare the theoretical values of ϕ_m and ω_m with the experimentally obtained ones.
2. Show that for $a < 1$, a compensator with a transfer function $G_c(s) = (1+aTs)/(1+Ts)$ always produces a phase lag, independent of the frequency. What should be a suitable name for such a compensator?
3. Consider an operational amplifier-based, unity-gain, inverting amplifier with both input and feedback resistors of value R . Connect capacitors C_1 and C_2 across the input and feedback resistors, respectively. Derive the transfer function $E_o(s)/E_i(s)$ and show that, barring the negative sign, it realizes $G_c(s)$. Express 'a' and 'T' in terms of the component values.

Exp. 9: Analyze the effect of addition of poles to the forward path transfer function of a closed loop system

OBJECTIVE : To study effect of addition of poles to forward path of a closed loop system.

The forward path transfer function of general second order system is given by,

$$G(s) = k_c \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

Addition of pole to forward path transfer function:

When we add a pole, the transfer function becomes,

$$G_1(s) = k_c \frac{\omega_n^2}{(s^2 + 2\xi\omega_n s + \omega_n^2)(s + a)}$$

$$Y_1(s) = k_c \frac{\omega_n^2}{(s^2 + 2\xi\omega_n s + \omega_n^2)(s + a)} U(s)$$

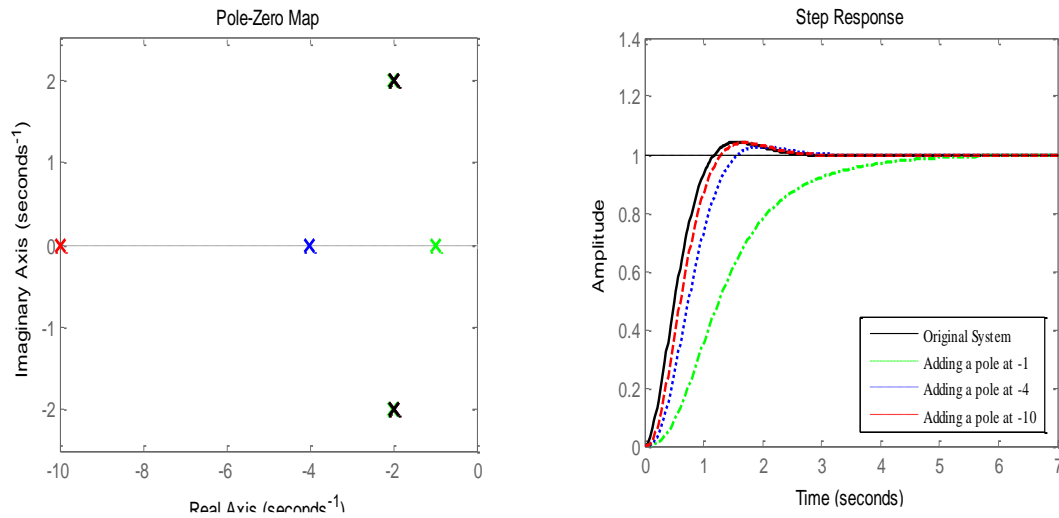
$$Y_1(s) = k_c \frac{\omega_n^2}{(s^2 + 2\xi\omega_n s + \omega_n^2)(s + a)} \frac{1}{s}$$

$$Y_1(s) = \frac{A}{s + a} + \frac{B}{s} + \frac{C(s + 2\xi\omega_n)}{s(s^2 + 2\xi\omega_n s + \omega_n^2)} + \frac{D\omega_n}{s(s^2 + 2\xi\omega_n s + \omega_n^2)}$$

$$y_1(t) = Ae^{-at} + Bu(t) + Ce^{-\xi\omega_n t} \cos(\omega_d t) + De^{-\xi\omega_n t} \sin(\omega_d t)$$

Example: Consider a closed loop system having transfer function of $G(s) = \frac{1}{s^2+4s+8}$. Write a MATLAB code to show the effect of addition of poles at -1, -4, -10.

$$G(s) = \frac{1}{s^2 + 4s + 8} \frac{1}{(T_p s + 1)}$$



Observation Table:

Tp	tr	Tp	Ts	%Mp
1				
4				
10				

DISCUSSION:

1. Discuss the effect of adding poles to overall transfer function on system response curve.

Exp. 10: Analyze the effect of addition of zeros to the forward path transfer function of a closed loop system

OBJECTIVE : To study effect of addition of zeros to forward path of a closed loop system.

THEORY: The forward path transfer function of general second order system is given by,

$$G(s) = k_c \frac{\omega_n^2}{s^2 + 2\xi\omega_n + \omega_n^2}$$

Addition of zero to forward path transfer function:

When we add a zero the forward path transfer function becomes,

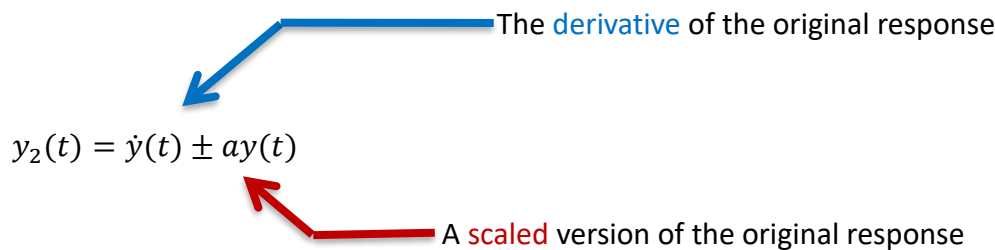
$$G_2(s) = \frac{Y_2(s)}{U(s)} = k_c \frac{\omega_n^2(s \pm a)}{(s^2 + 2\xi\omega_n + \omega_n^2)}$$

$$G_2(s) = \frac{Y_2(s)}{U(s)} = s \frac{k_c \omega_n^2}{(s^2 + 2\xi\omega_n + \omega_n^2)} \pm a \frac{k_c \omega_n^2}{(s^2 + 2\xi\omega_n + \omega_n^2)}$$

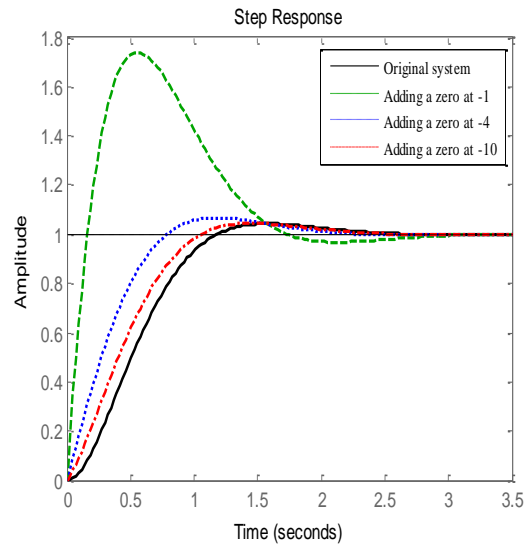
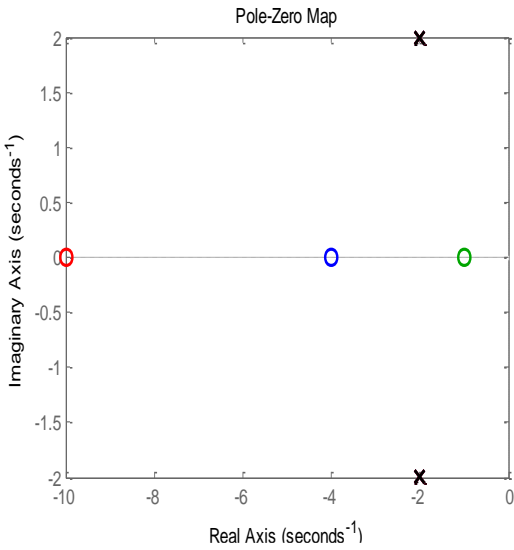
$$G_2(s) = \frac{Y_2(s)}{U(s)} = sG(s) \pm aG(s)$$

$$Y_2(s) = sG(s)U(s) \pm aG(s)U(s)$$

$$Y_2(s) = sY(s) \pm aY(s)$$



Example: Consider a closed loop system having the transfer function of $G(s) = \frac{1}{s^2+4s+8}$. Write a MATLAB code to show the effect of addition of zeros at -1, -4, -10.



Observation Table:

Tz	Tr	Tp	Ts	%Mp
1				
4				
10				

DISCUSSION:

Discuss the effect of adding zeros to overall transfer function on system response curve.