CIRCUITS & CONTROLS LABORATORY (SYLLABUS)
V SEMESTER B. E (E&TE)

Sub. Code: 21ET52 **CIE Marks: 50**
Hrs/Week: 2

COURSE OBJECTIVES:

This course will enable students to-

1. Apply mesh and nodal techniques to solve an electrical network.
2. Solve different problems related to Electrical circuits using Network Theorems and Two port network.
3. Understand basics of control systems and design mathematical models using block diagram reduction, SFG, etc.
4. Understand Time domain and Frequency domain analysis.
5. Familiarize with the State Space Model of the system.

Syllabus

1. Verification of Superposition theorem, Maximum Power transfer theorem
2. Verification of Thevenin's, Norton's theorem
3. Speed torque characteristics of i. AC Servomotor ii. DC Servomotors
4. Determination of time response specification of a second order Under damped System, for different damping factors.
5. Determination of frequency response of a second order System
6. Determination of frequency response of a lead lag compensator
7. Using Suitable simulation package study of speed control of DC motor using
   i) Armature control ii) Field control
8. Using suitable simulation package, draw Root locus & Bode plot of the given transfer function.

Demonstration Experiments (For CIE only, not for SEE)

9. Using suitable simulation package, obtain the time response from state model of a system.
10. Implementation of PI, PD Controllers.
11. Implement a PID Controller and hence realize an Error Detector.
12. Demonstrate the effect of PI, PD and PID controller on the system response.

COURSE OUTCOMES (Course Skill Set):

At the end of the course the student will be able to:

COs	Course Outcomes
CO1	**Compute** the circuit analysis techniques to determine circuit parameters, assess stability, and understand the impact of gain on system behaviour
CO2	**Devise** the time response and frequency response of the dynamic systems and the state models to evaluate the system stability and the impact on the system behaviour
CO3	**Assess** the effectiveness of different circuit analysis techniques in solving real-world problems and the stability of systems based on frequency response analysis
CO4	**Design** the circuits using a suitable modern tool individually and **validate** the design

DAYANANDA SAGAR COLLEGE OF ENGINEERING
DEPARTMENT OF ELECTRONICS & TELECOMMUNICATION ENGINEERING
BENGALURU – 560078

DO's

- Adhere and follow timings, proper dress code with appropriate footwear.
- Bags and other personal items must be stored in designated place.
- Come prepare with the viva, procedure, and other details of the experiment.
- Secure long hair, Avoid-loose clothing, Deep neck, and sleeveless dresses.
- Do check for the correct ranges/rating and carry one meter/instrument at a time.
- Inspect all equipment/meters for damage prior to use.
- Conduct the experiments accurately as directed by the teacher.
- Immediately report any sparks/ accidents/ injuries/ any other troublesome incident to the faculty /instructor.
- Handle the apparatus/meters/computers gently and with care.
- In case of an emergency or accident, follow the safety procedure.
- Switch OFF the power supply after completion of experiment.

DONT's

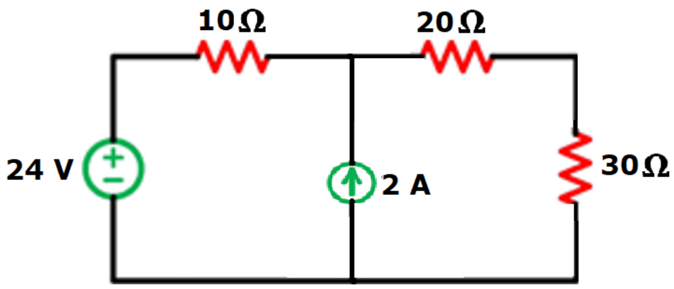
- The use of mobile/ any other personal electronic gadgets is prohibited in the laboratory.
- Do not make noise in the Laboratory & do not sit on experiment table.
- Do not make loose connections and avoid overlapping of wires.
- Don't switch on power supply without prior permission from the concerned staff.
- Never point/touch the CRO/Monitor screen with the tip of the open pen/pencil/any other sharp object.
- Never leave the experiments while in progress.
- Do not insert/use pen drive/any other storage devices into the CPU
- Do not leave the Laboratory without the signature of the concerned staff in observation book.

AIM: To verify Superposition theorem

Software used: MATLAB Simulink

SUPERPOSITION THEOREM: "In a linear network with several independent sources which include equivalent sources due to initial conditions, and linear dependent sources, the overall response in any part of the network is equal to the sum of individual responses due to each independent source, considered separately, with all other independent sources reduced to zero".

Consider the Network given below, verify the superposition theorem for the 20 Ω Resistor



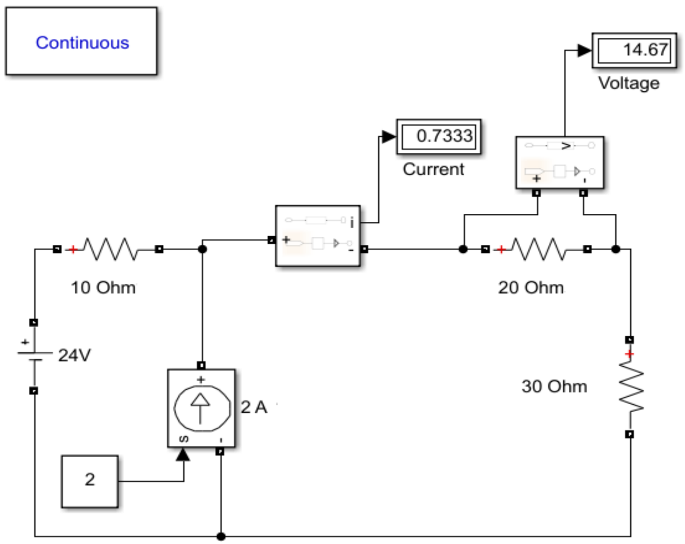
Procedure:

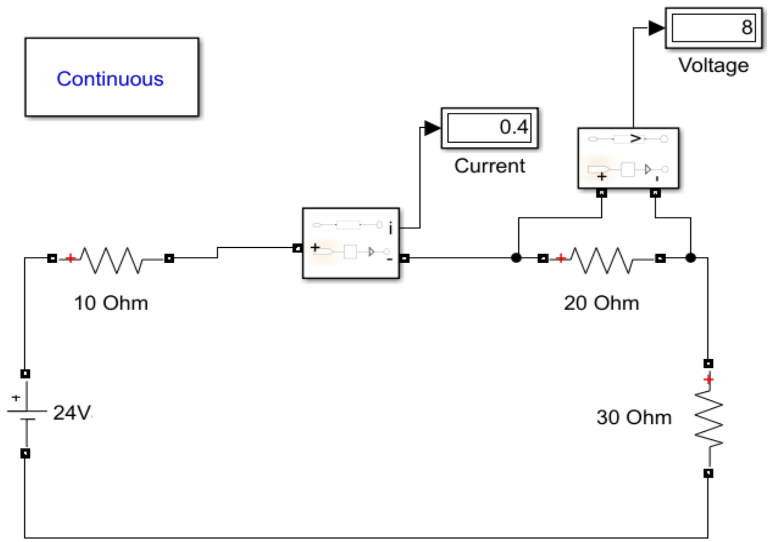
Step 1: Make the connections as shown in the circuit diagram by using MATLAB Simulink and measure the response 'I' in the 20 Ω resistor by considering all the sources 24 V, 2 A in the network

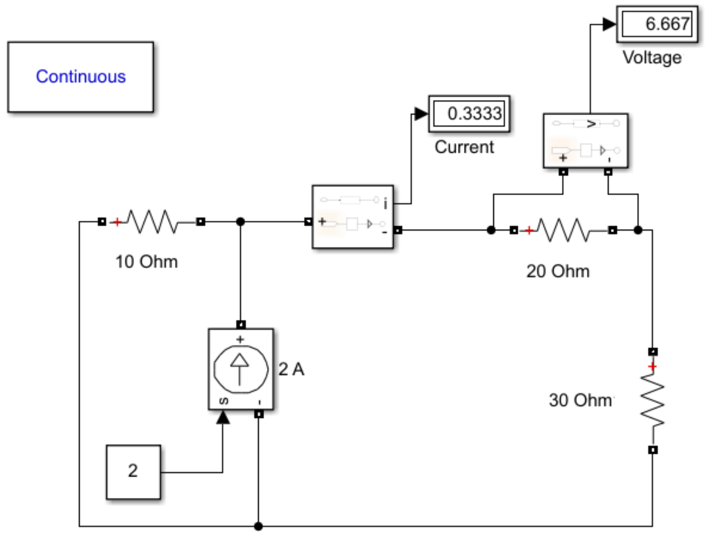
Step 2: Replace the 2 A source with its internal impedances (open circuited), Measure the response 'I_1' in the 20 Ω resistor by considering 24 V source in the network

Step 3: Replace the sources 24 V with its internal impedances (short circuited). Measure the response 'I_2' in the 20 Ω resistor by considering 2 A source in the network.

Step 4: The responses obtained in Step 1 should be equal to the sum of the responses obtained in Steps 2 and 3. $I=I_1+I_2$. Hence Superposition Theorem is verified.


Continuous
14.67
Voltage
0.7333
Current
10 Ohm
20 Ohm
24V
30 Ohm
2 A
2


Continuous
8
Voltage
0.4
Current
10 Ohm
20 Ohm
24V
30 Ohm


Continuous
6.667
Voltage
0.3333
Current
10 Ohm
20 Ohm
2 A
30 Ohm
2

Experiment 1b. Verification of Maximum Power Transfer theorem

AIM: To verify Maximum power Transfer theorem.

Software used: MATLAB Simulink

Maximum Power Transfer Theorem: "In any circuit the maximum power is transferred to the load when the load resistance is equal to the source resistance. The source resistance is equal to the Thevenin's resistance".

Procedure:

Step 1:

1. Make the connections as shown in the circuit diagram by using Multisim/MATLAB Simulink.
2. Measure the Power across the load resistor by considering all the sources in the network.

Step 2: Finding Current through the load $(\boldsymbol{R_L})$

1. Measure the current I through the load (R_L)

Step 3: Finding Voltage across the load $(\boldsymbol{R_L})$

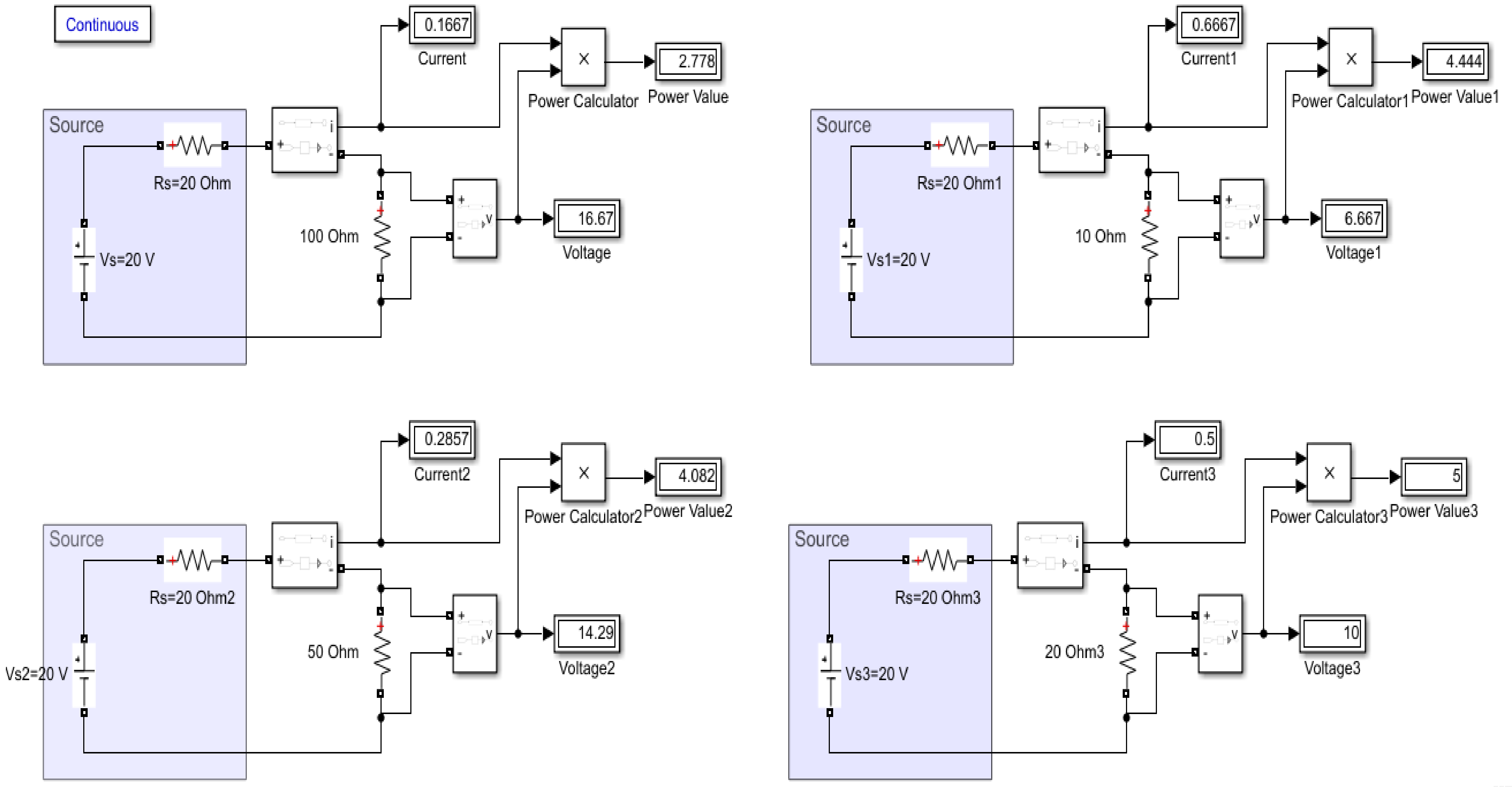
1. Measure the voltage V across the load (R_L)

Step 4: Measuring Power for different Load Resistors

1. Measure the power developed across the load
2. Verify the Maximum power transfer theorem for different values of load resistors (R_L)

Power measured from the above steps results in maximum power dissipation when load $R_L = R_S$

Hence Maximum Power Transfer Theorem is verified


Continuous
0.1667
Current
X
2.778
Power Calculator
Power Value
Source
Rs=20 Ohm
100 Ohm
16.67
Voltage
Vs=20 V
0.6667
Current1
4.444
Power Calculator1
Power Value1
Source
Rs=20 Ohm1
10 Ohm
6.667
Voltage1
Vs1=20 V
0.2857
Current2
4.082
Power Calculator2
Power Value2
Source
Rs=20 Ohm2
50 Ohm
14.29
Voltage2
Vs2=20 V
0.5
Current3
5
Power Calculator3
Power Value3
Source
Rs=20 Ohm3
20 Ohm3
10
Voltage3
Vs3=20 V

Experiment 2a: Verification of Thevenin's Theorem

AIM: To verify Thevenin's theorem

Software Used: MATLAB Simulink

Thevenin's Theorem: "Any two terminal network consisting of linear impedances and generators may be replaced at the two terminals by a single voltage source acting in series with an impedance. The voltage of the equivalent source is the open circuit voltage measured at the terminals of the network and the impedance, known as Thevenin's equivalent impedance, Z_{TH}, is the impedance measured at the terminals with all the independent sources in the network reduced to zero".

Procedure:

Step 1:

1. Make the connections as shown in the circuit diagram by using MATLAB Simulink.
2. Measure the response 'I' in the load resistor by considering all the sources in the network.

Step 2: Finding Thevenin's Resistance (R_{TH})

1. Open the load terminals and replace all the sources with their internal impedances.
2. Measure the impedance across the open circuited terminal which is known as Thevenin's Resistance.

Step 3: Finding Thevenin's Voltage (V_{TH})

1. Open the load terminals and measure the voltage across the open circuited terminals.
2. Measured voltage will be known as Thevenin's Voltage.

Step 4: Thevenin's Equivalent Circuit

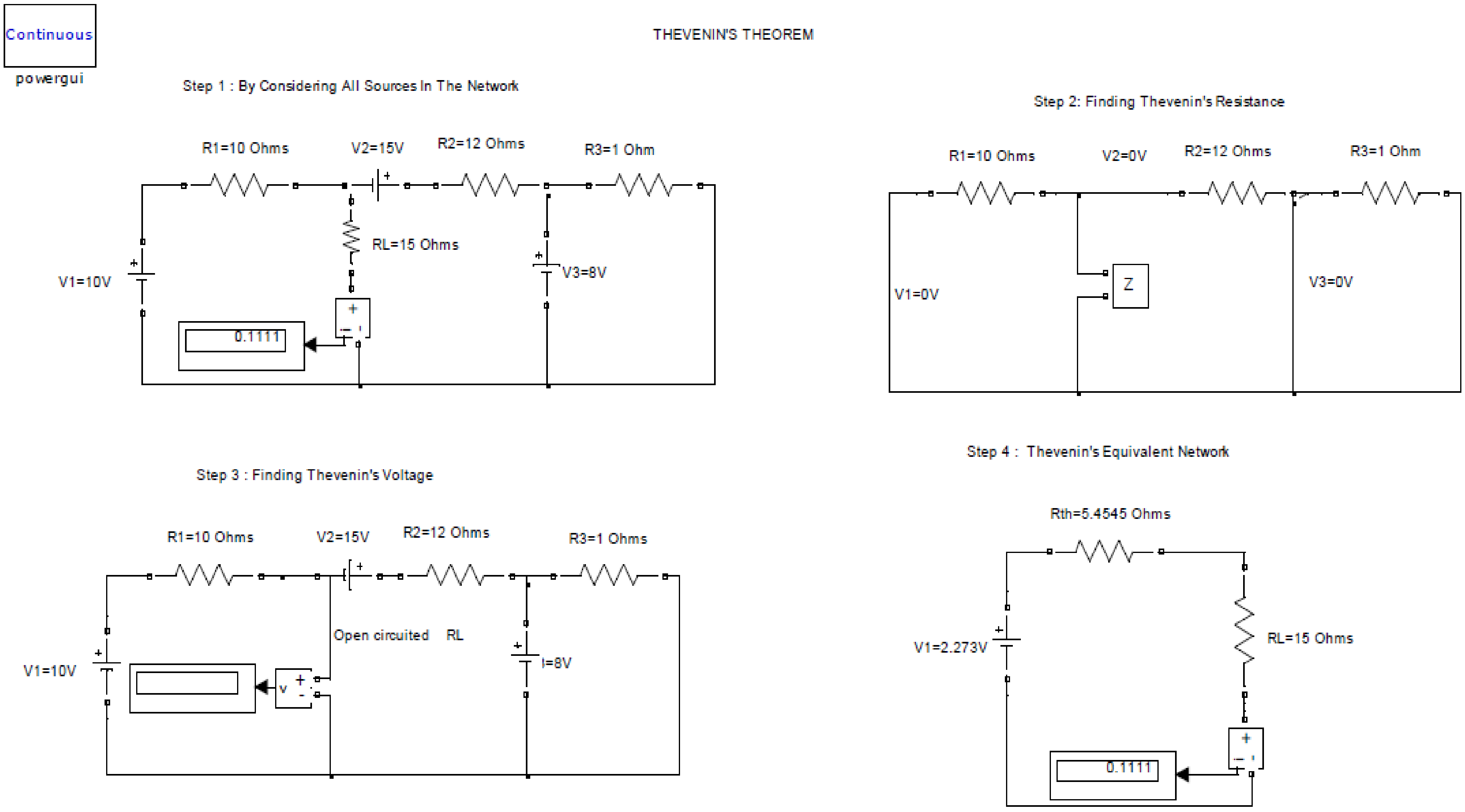
1. V_{TH} and R_{TH} are connected in series with the load.
2. Measure the current through the load resistor

$$I_L = \frac{V_{TH}}{R_{TH} + R_L}$$

Current measured from Thevenin's Equivalent Circuit should be same as current obtained from the actual circuit.

$$I = I_L$$

Hence Thevenin's Theorem is Verified



Open Circuit Voltage Vth = 2.273V
Thevenin's Resistance = 5.4545Ohms
Current through Load Resistor 15 Ohms IL = 0.1111A

With all the sources in the network Current through Load Resistor 15 Ohms : I=0.1111A

I=IL

Hence Thevenin's Theorem is Verified.

Experiment 2b: Verification of Norton’s theorem

AIM: To verify Norton’s theorem

Software used: MATLAB Simulink

Norton’s Theorems: “Any two terminal network consisting of linear impedances and generators may be replaced at its two terminals, by an equivalent network consisting of a single current source in parallel with an impedance. The equivalent current source is the short circuit current measured at the terminals and the equivalent impedance is same as the Thevenin’s equivalent impedance”.

Procedure:

Step 1:

1. Make the connections as shown in the circuit diagram by using MATLAB Simulink.
2. Measure the response ‘I’ in the load resistor by considering all the sources in the network.

Step 2: Finding Norton’s Resistance (R_N)

1. Open the load terminals and replace all the sources with their internal impedances.
2. Measure the impedance across the open circuited terminal, known as Norton’s Resistance (R_N)

Step 3: Finding Norton’s Current (I_N)

1. Short the load terminals and measure the current through the short-circuited terminals.
2. Measured current is be known as Norton’s Current (I_N)

Step 4: Norton’s Equivalent Circuit

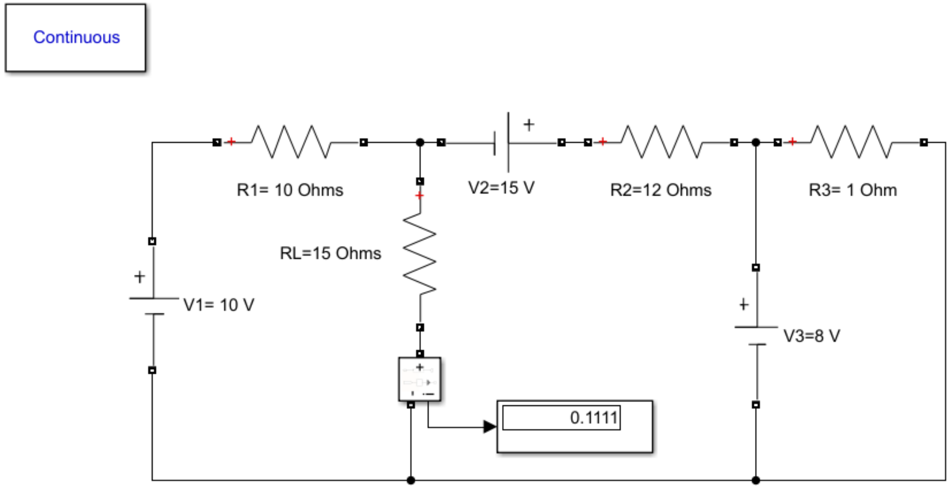
1. R_N and I_N are connected in parallel to the load.
2. Measure the current through the load resistor (R_L)

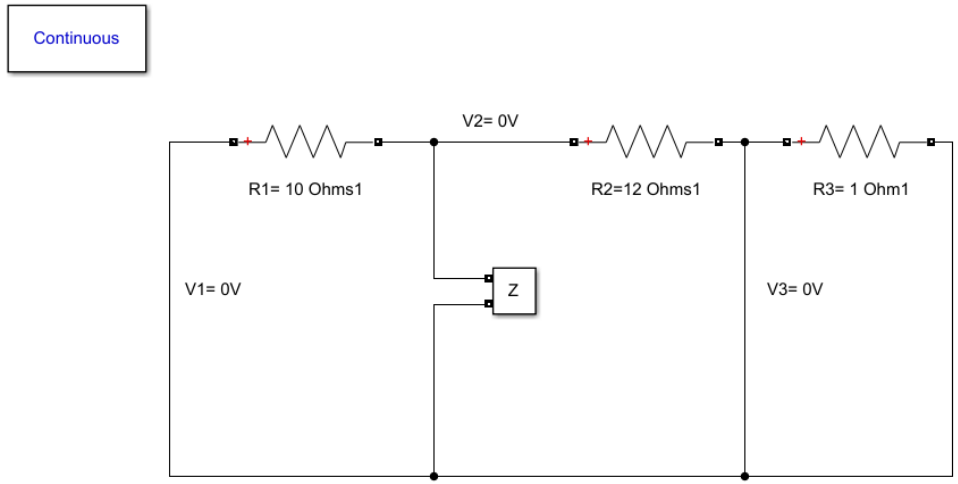
$$I_L = \frac{I_N R_N}{R_N + R_L}$$

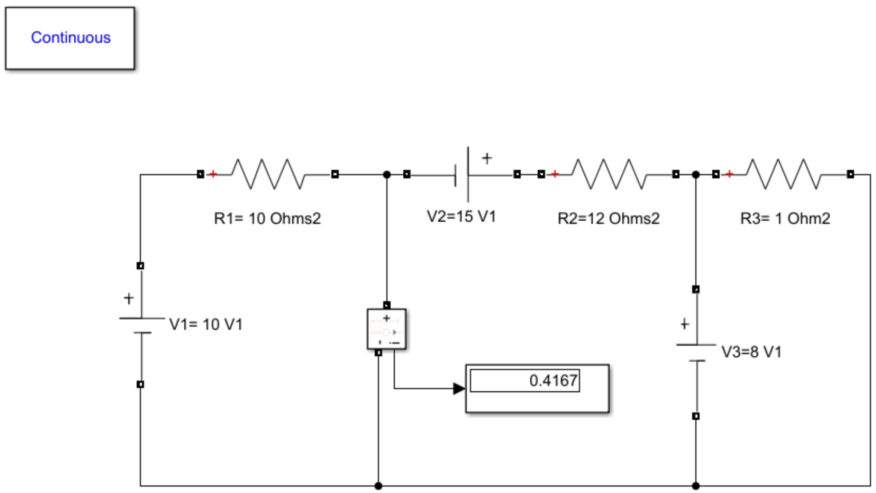
Current measured from Norton’s Equivalent Circuit should be same as current obtained from the actual circuit.

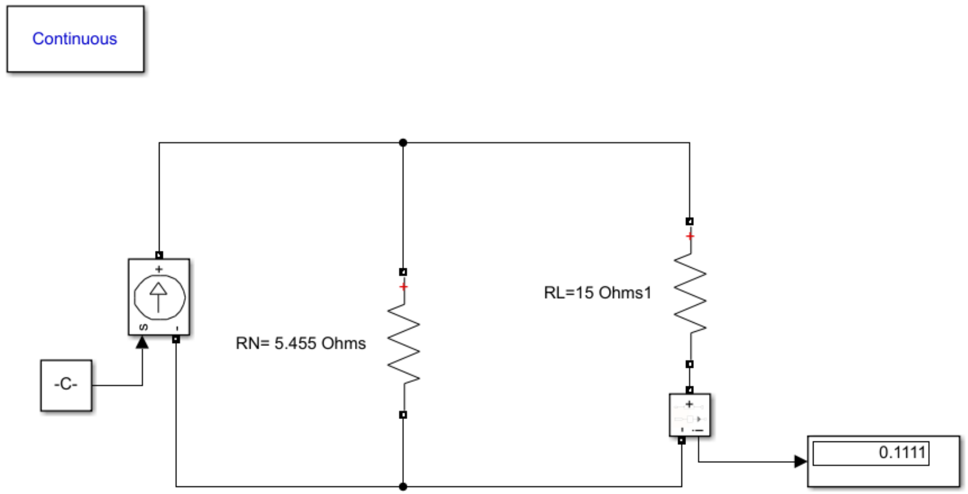
$$I = I_L$$

Hence Norton’s Theorem is Verified.









Norton's Current = 0.4167 A
Norton's Resistance = 5.4545 Ohms
Current through Load Resistor 15 Ohms = 0.1111 A

With all the sources in the Network, Current through Load Resistor 15 Ohms= 0.1111 A

Hence Norton's Theorem is verified

Experiment 3: Speed torque characteristics of i) AC Servomotor ii) DC Servomotors

AIM: To obtain speed torque characteristics of AC and DC Servomotor

THEORY:

A servo motor is a special type of electric motor which is used as a rotary or linear actuator for precise control of angular or linear position, velocity, acceleration and many other closed-loop position control applications. Servo motors are designed with long rotor length and small diameter. Thus, due to low inertia, the servo motors have a high-speed response. A typical servo motor consists of a motor, feedback system, controller, etc. Where, it uses the position feedback to control the speed and final position of the motor. There are two types of servo motors, AC servos and DC servos. The main difference between the two motors is their source of power. AC servo motors rely on an electric outlet, rather than batteries like DC servo motors. While DC servo motor performance is dependent only on voltage, AC servo motors are dependent on both frequency and voltage. Due to complexity of the power supply, AC servo motors can handle high surges, which is why they are often used in industrial machinery.

Parameters	AC Servo Motor	DC Servo Motor
Definition	A servo mechanism in which a two-phase or three-phase induction motor is used	A servo mechanism which consists of a DC motor
Speed	Adaptable to high-speed operating conditions	Adaptable to the limited speed working conditions.
Torque	Develop high torque	Develop a limited torque
Operation	AC servo motors have smooth, stable and less noise operation	DC servo motors have less stable and noisy operation
Efficiency	Less efficient, ranging from 5%-20%.	Highly efficient
Maintenance	Require less maintenance	Require frequent maintenance due to presence of commutator and brushes
Weight	Lighter in weight, smaller in size	Heavier, larger in size
Practical applications	Suitable for Low power Applications (0.5-100 W) like robotics, machine tools, semiconductor devices, aircrafts, etc.	Suitable for High power Applications like computers, prime movers, controlled machineries, etc.

Speed: The speed of a motor is defined as the rate at which the motor rotates and is measured in revolutions per minute, or RPM.

Torque: The torque output of a motor is the amount of rotational force that the motor develops and is measured in Newton-meters, or Nm. Torque is the twisting force of the motor.

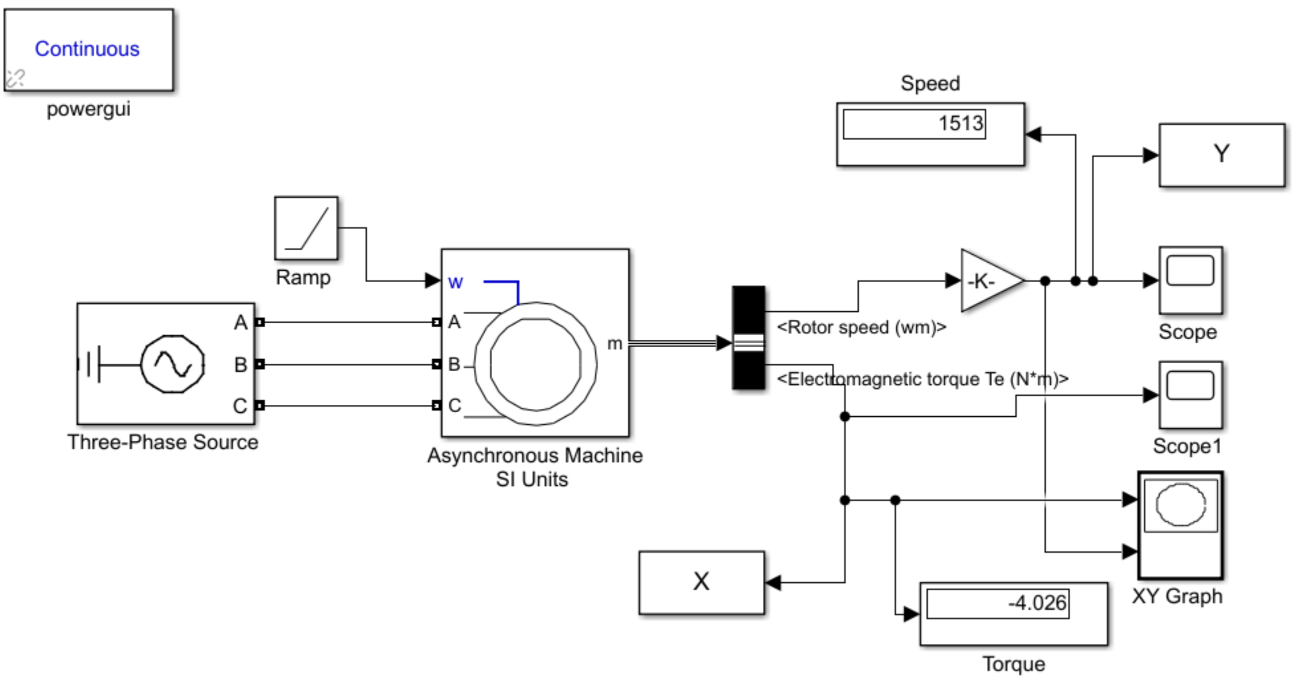
The Torque and Speed relationship is defined by the formula:

$$Mechanical\ power = \ Speed\ \times Torque$$

The torque and speed relationship is inversely proportional since the rated output power of a motor is a fixed value. As output speed increases, the available output torque decreases proportionately. As the output torque increases, the output speed decreases proportionately. Although speed governs the maximum speed of an electric motor, having more torque enables the system to reach top speed in less time.

AC SERVOMOTOR:

SIMULINK IMPLEMENTATION BLOCK DIAGRAM:

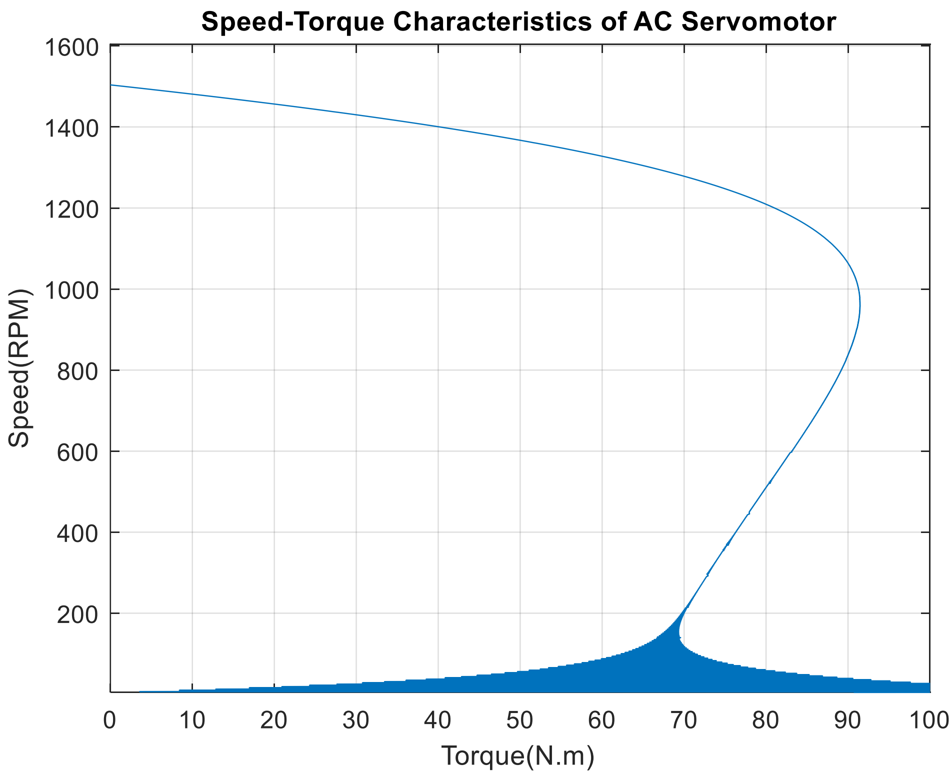


SPEED-TORQUE CHARACTERISTICS OF AC SERVOMOTOR

PROCEDURE:

- 3 phase induction motor is modelled using Asynchronous Machine SI Units with the specifications: Squirrel-cage Rotor with 5.4 HP, (4KW), 400 V, 50 Hz and 1430 RPM, Mechanical input is chosen as Speed w, Reference frame: Stationary
- It is connected to a 3-phase source, values are changed to match the specifications of the motor
- Mechanical input in terms of Speed is provided using Ramp signal of maximum Slope 16 V and start time 0.1 seconds
- Speed is converted to RPM using: $RPM = \left(\frac{rad}{\sec}\right) * \frac{60}{2\pi}$
- Bus Selector is added at the output of the AC Motor to select 2 outputs: Speed and Torque
- The values of Speed (RPM) and Torque (Nm) are displayed on the Scope.
- Speed and Torque are observed using XY Graph
- Plot the Speed-Torque Characteristics using MATLAB Command Window: Plot(X,Y)
- Label the figure accordingly

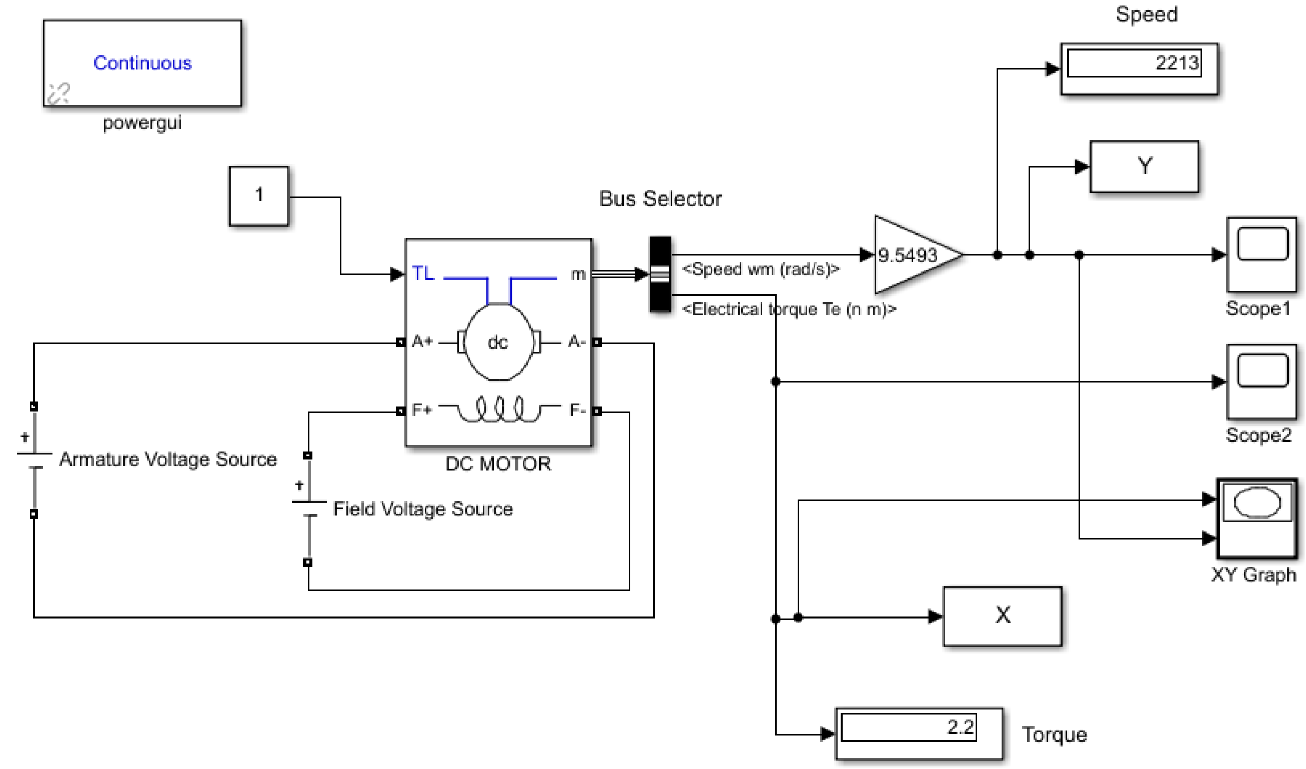
RESULT:



Observation: The Speed of a DC Servomotor decreases linearly with the increase in the Torque.

DC SERVOMOTOR:

SIMULINK IMPLEMENTATION BLOCK DIAGRAM:

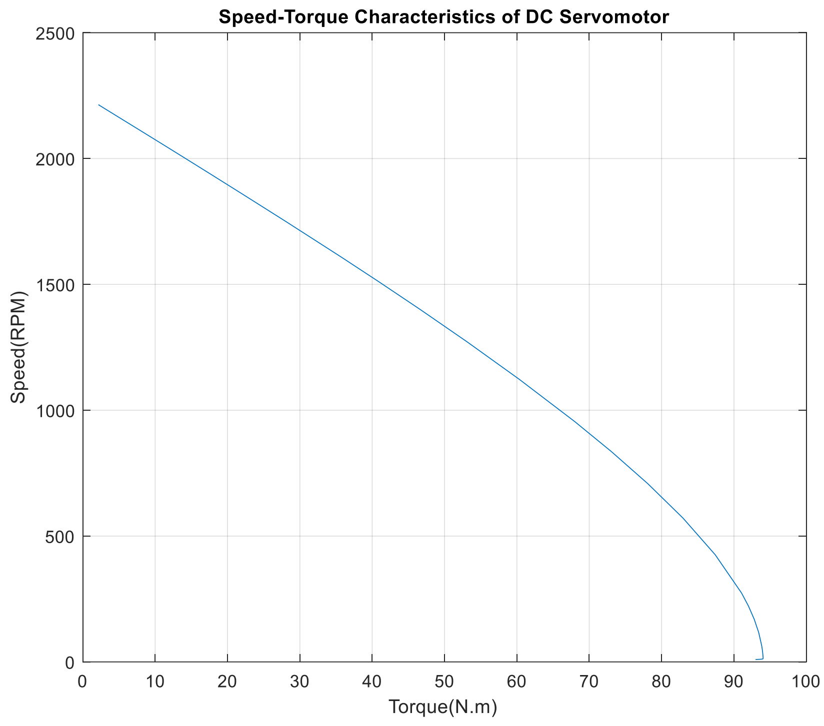


SPEED-TORQUE CHARACTERISTICS OF DC MOTOR

PROCEDURE:

- Set up the block diagram using SIMULINK
- DC Machine with 5HP, Armature Voltage 240 V and Field Voltage 300 V is modelled as a DC servomotor
- Mechanical input in terms of Speed is provided using unit step signal
- Bus Selector is added at the output of the DC Motor to select 2 outputs: Speed and Electrical Torque
- Speed is converted to RPM using: $RPM = \left(\frac{rad}{\sec}\right) * \frac{60}{2\pi}$
- The values of Speed (RPM) and Electrical Torque (Nm) are displayed on the Scope.
- Speed and Electrical Torque are plotted using XY Graph
- Plot the Speed-Torque Characteristics using MATLAB Command Window: Plot(X,Y)
- Label the figure accordingly

RESULT:



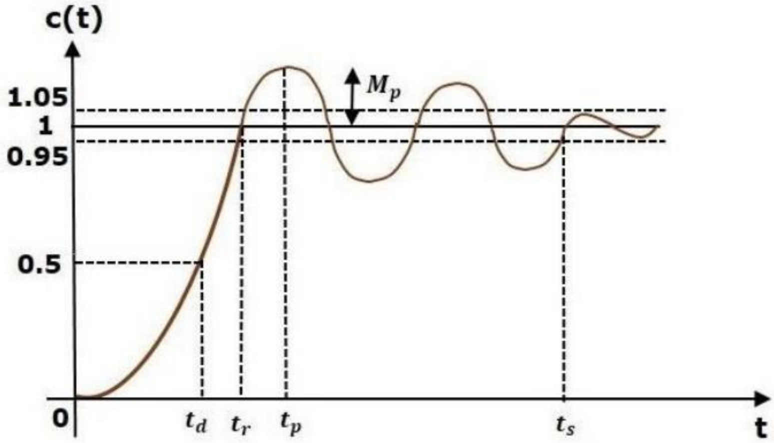
Observation: The Speed of a DC Servomotor decreases linearly with the increase in the Torque

Experiment 4: Determination of time response specification of a second order under damped System, for different damping factors.

AIM: To obtain time response of a second order system in case of under damped systems.

THEORY:

The time response of control system consists of two parts. Transient response and steady state response. Most of the control systems use time as its independent variable. All the time domain specifications are represented in the figure. The response up to the settling time is known as transient response and the response after the settling time is known as steady state response.



Analysis of response means to see the variation of output with respect to time. The output of the system takes some finite time to reach to its final value. Every system has a tendency to oppose the oscillatory behaviour of the system which is called damping. The damping is measured by a factor called damping ratio (ξ) of the system.

- If the damping is very high then there will not be any oscillations in the output. The output is purely exponential. Such system is called an over damped system ($1 < \xi < \infty$)
- If the damping is less compared to over damped case, the system is called a critically damped system ($\xi = 1$)
- If the damping is very less, the system is called under damped system($0 < \xi < 1$)
- With no damping system is undamped. ($\xi = 0$)

Time domain specifications:

- Damped Natural Frequency: $w_d = w_n\sqrt{1 - \xi^2}$
- Rise Time: It is the time required for the response to rise from 0 to 100% of the final value for the underdamped system $t_r = \frac{\pi - \theta}{w_n\sqrt{1-\xi^2}}$; $\theta = cos^{-1}(\xi)$
- Peak time: It is the time required for the response to reach the peak of time response or the peak overshoot $t_p = \frac{\pi}{w_n\sqrt{1-\xi^2}}$

- Settling Time for 2% tolerance band: It is the time required for the response to reach and stay within a specified tolerance band (2% or 5%) of its final value $t_s = \frac{4}{\xi w_n}$
- Percent Overshoot: $M_p = e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}}$

```
% Second-Order Underdamped System Example

% Given System
wn = 10; % Natural Frequency
zeta = [0.1  0.5 0.9]; % Damping Ratio

% Calculate Time Response Specification for Different Damping Factors
for i = 1:length(zeta)
  theta(i)= acos(zeta(i));
  wd = wn*sqrt(1-zeta(i)^2);    % Calculate Damped Natural Frequency

  % Calculate Time Response Specifications
   tr = (pi-theta(i))/wd;
   tp= pi/wd;
   ts = 4/(zeta(i)*wn);
   Mp = exp(-zeta(i)*pi/sqrt(1-zeta(i)^2));


   % Display Results
   fprintf('Damping Ratio = %.1f\n', zeta(i));
   fprintf('Damped Natural Frequency = %.2f rad/s\n', wd);
   fprintf('Rise Time = %.2f s\n', tr);
   fprintf('Peak Time = %.2f s\n', tp);
   fprintf('Settling Time = %.2f s\n', ts);
   fprintf('Percent Overshoot = %.2f %%\n\n', Mp*100);
end
```

RESULT:

Damping Ratio = 0.1

Damped Natural Frequency = 9.95 rad/s

Rise Time = 0.17 s

Peak Time = 0.32 s

Settling Time = 4.00 s

Percent Overshoot = 72.92 %

Damping Ratio = 0.5

Damped Natural Frequency = 8.66 rad/s

Rise Time = 0.24 s

Peak Time = 0.36 s

Settling Time = 0.80 s

Percent Overshoot = 16.30 %

Damping Ratio = 0.9

Damped Natural Frequency = 4.36 rad/s

Rise Time = 0.62 s

Peak Time = 0.72 s

Settling Time = 0.44 s

Percent Overshoot = 0.15 %

Experiment 5: Determination of frequency response of a second order System

AIM: To study the Frequency response of a second order system

THEORY: The frequency response of a second-order system describes how the system responds to different frequencies in the input signal. It provides information about the system's magnitude and phase shift characteristics at different frequencies. The frequency response is obtained by analyzing the system's transfer function or by applying a sinusoidal input signal with varying frequencies and measuring the system's output.

Consider the transfer function of the second order closed loop control system as,

$$T(s) = \frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\delta\omega_n s + \omega_n^2}$$

Substitute, s = jω in the above equation.

$$T(j\omega) = \frac{\omega_n^2}{(j\omega)^2 + 2\delta\omega_n(j\omega) + \omega_n^2}$$

$$\Rightarrow T(j\omega) = \frac{\omega_n^2}{-\omega^2 + 2j\delta\omega\omega_n + \omega_n^2} = \frac{\omega_n^2}{\omega_n^2\left(1 - \frac{\omega^2}{\omega_n^2} + \frac{2j\delta\omega}{\omega_n}\right)}$$

$$\Rightarrow T(j\omega) = \frac{1}{\left(1 - \frac{\omega^2}{\omega_n^2}\right) + j\left(\frac{2\delta\omega}{\omega_n}\right)}$$

Let, ω/ω_n = u Substitute this value in the above equation.

$$T(j\omega) = \frac{1}{(1-u^2) + j(2\delta u)}$$

Magnitude of T(jω) is -

$$M = |T(j\omega)| = \frac{1}{\sqrt{(1-u^2)^2 + (2\delta u)^2}}$$

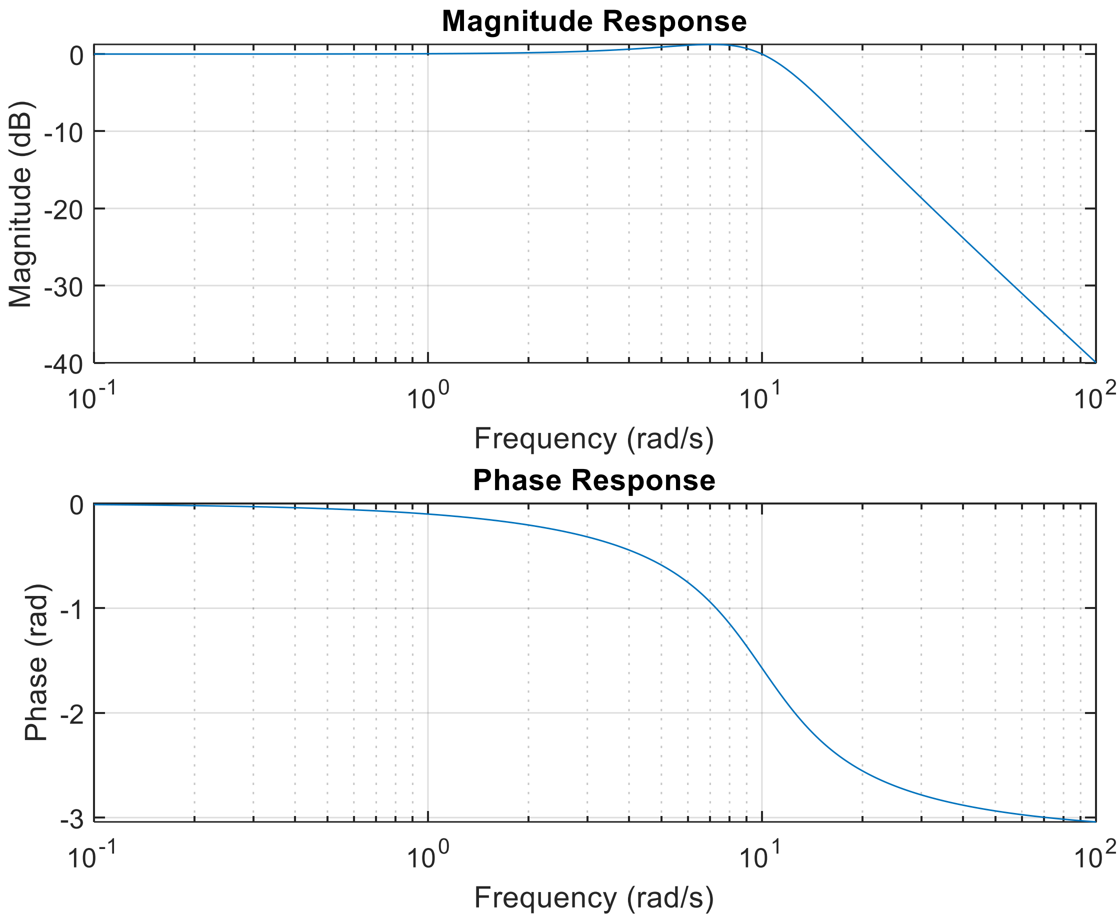
Phase of T(jω) is -

$$\angle T(j\omega) = -tan^{-1}\left(\frac{2\delta u}{1-u^2}\right)$$

```
% Frequency Response of Second-Order System
clc;
clear all;
close all;

% Given System
wn = 10; % Natural Frequency
zeta = 0.5; % Damping Ratio
 % Define Transfer Function
```

```
num = wn^2;
den = [1 2*zeta*wn wn^2];
sys = tf(num, den);
w = logspace(-1, 2, 1000);     % Define Frequency Range
% Calculate Frequency Response
h=freqs(num,den,w);
mag=20*log10(abs(h));
phase=angle(h);

% Plot Frequency Response
subplot(2,1,1);
semilogx(w,mag);
grid on;
title('Magnitude Response');
xlabel('Frequency rad/s)');
ylabel('Magnitude (dB)');
subplot(2,1,2);
semilogx(w, phase);
grid on;
title('Phase Response');
xlabel('Frequency (rad/s)');
ylabel('Phase(rad)');
```

RESULT:



Experiment 6: Determination of frequency response of a lead lag compensator

AIM: To design the Lead-lag compensator for given transfer function $G(s) = \frac{k}{s(s+1)(s+2)}$ with unity feedback system having the following specification: Velocity error constant $k_v = 10sec^{-1}$ and Phase margin ≥ $\mathbf{50}$°

THEORY:

In control systems engineering, compensators are devices or components used to modify the response characteristics of a system. Their primary purpose is to improve the performance of a control system by adjusting the dynamic behavior of the system to meet certain design specifications. Compensators are employed to achieve desired system characteristics such as stability, transient response, and steady-state accuracy.

There are two main types of compensators: lead compensators and lag compensators.

1. Lead Compensator:
   - A lead compensator is designed to increase the phase margin of a system, which helps improve the stability of the system.
   - It introduces a phase lead in the system response, making the system respond more quickly to changes in the input signal.
   - Lead compensators are often used to enhance the transient response of a system.

2. Lag Compensator:
   - A lag compensator is designed to decrease the phase margin of a system, which can be useful for improving the steady-state accuracy of the system.
   - It introduces a phase lag in the system response, making the system respond more slowly to changes in the input signal.
   - Lag compensators are commonly employed to reduce steady-state error in the system.

Compensators are typically implemented using electrical circuits or software algorithms in digital control systems. The design of compensators involves analyzing the open-loop and closed-loop transfer functions of the system and using techniques such as frequency response analysis and root locus analysis. The goal is to achieve a stable and well-performing closed-loop system with desired characteristics. Compensators are essential components in control systems that help shape the dynamic behavior of the system to meet specific performance requirements. They play a crucial role in achieving stability, improving transient response, and minimizing steady-state errors in control systems. Both types of Compensators can be subsequently implemented to achieve combined advantages of both, resulting in Lead-Lag Compensator

Solution

Step 1.Find the value of k for the uncompensated system the given transfer function

$$G(s) = \frac{k}{s(s+1)(s+2)} \text{ having unity feedback } H(s) = 1 \quad k_v = 10\text{sec}^{-1}$$

$$k_v = Lt_{s\to 0} SG(s)H(s)$$
$$10 = \frac{k}{2} \Rightarrow k = 20$$

Step 2. Draw the bode plot for the uncompensated system and we need to find the value of phase margin (γ)

$$\text{The transfer function } G(s) = \frac{20}{s(s+1)(s+2)}$$

Frequency domain transfer function $s = j\omega$

$$G(j\omega) = \frac{20}{j\omega(jw+1)(j\omega+2)}$$

$$= \frac{10}{s(s+1)(0.5s+1)}$$
$$= \frac{10}{j\omega(j\omega+1)(0.5j\omega+1)}$$

The PM for uncompensated system

$$PM = -28°$$

Lead lag compensator unstable (given$PM \geq 50°$)

Step 3:- select the new PM for the compensated system

$$\gamma_n = \gamma_d + \epsilon$$

$$\gamma_d = desired\ PM\ (gien\ in\ the\ Pbm)$$
$$\epsilon = small\ value\ for\ correction$$

The new PM for the compensated system

$$\gamma_n = \gamma_d + \epsilon$$
$$= 50° + 5°$$
$$\gamma_n = 55°$$

Step 4: Find the new ω_{gc} which corresponds to $\emptyset_{gcn} = \gamma_n - 180°$

$$= 55° - 180° = -125°$$

The new gain crossover frequency $\omega_{gcn} = 0.416 rad / sec$ from bode plot.

To find β for lag compensator (Ag=27dB from graph)

$$\beta = 10^{Ag/20}$$
$$= 10^{27/20}$$
$$\beta = 22.387$$

Step 5: To obtain the transfer function of Lead-lag compensator.

Zero of lag compensator $z_c = \frac{\omega_{gcn}}{8}$

WKT $\omega_{gcn} = 0.416$

$$T_1 = \frac{8}{\omega_{gcn}} = \frac{8}{0 \cdot 416} = 19.23$$

Lag compensator
Transfer function

$$G_o(s) = \frac{(1 + sT)}{(1 + s\beta T)}$$

$$G_o(s) = \frac{(1 + 19.23s)}{(1 + s(22.387)(19.23))}$$

$$G_0(s) = \frac{(1+19.23s)}{(1+430.5s)} \text{ TF of Lag compensator}$$

The relation between $\alpha \& \beta$

$$\alpha = \frac{20}{\beta} = \frac{20}{22 \cdot 387} = 0.8983$$

Based on α need to find the given by formula

$$-20\, log \frac{1}{\sqrt{\alpha}} = -20 \log \frac{1}{\sqrt{0.8933}} = -0.49$$

Corresponding frequency from bode plot $\omega_m = 2 \cdot 91 rad / sec$

$$T_2 = \frac{1}{\omega_m \sqrt{\alpha}}$$

$$= \frac{1}{2.91\sqrt{0.8933}}$$

$$T_2 = 0.3636$$

The lead compensator $G_{0lead}(s) = \frac{(1+ST_2)}{(1+s\alpha T_2)}$

$$= \frac{(1+S(0.3636))}{(1+s(0.8933)(0.3636))}$$

$$= \frac{(1+0 \cdot 3636s)}{(1+0 \cdot 324s)}$$

Step 6:- OL transfer function of lead lag compensator

$$G_c(S) = G(S) \times G_{Lag}(s) \times G_{Lead}(s)$$

$$= \frac{10}{s(s+1)(1+0.5s)} \times \frac{(1+19.23^s)}{(1+430.5s)} \times \frac{(1+0.3636s)}{(1+0.324s)}$$

Step 7:- obtain the bode plot from compensated OLTF

$$G_c(s) = \frac{10}{s(s+1)(1+0.5s)} \times \frac{(1+19.23\ \ s)}{(1+430.5s)} \times \frac{(1+0.3636s)}{(1+0.324s)}$$

MATLAB CODE:

```
clc;
clear all;
close all;
num=[20];
den=[1 3 2 0];
G=tf(num,den);
figure(1);
bode(num,den);
title('Bode Plot for Uncompensated System G(s)=20/S(S+1)(S+2)')
grid;

[Gm,Pm,Wcg,Wcp]=margin(num,den);
GmdB=20*log10(Gm);
disp('Results of Uncompensated System:');
fprintf('Gain Margin is: %0.2f dB\n', GmdB);
```

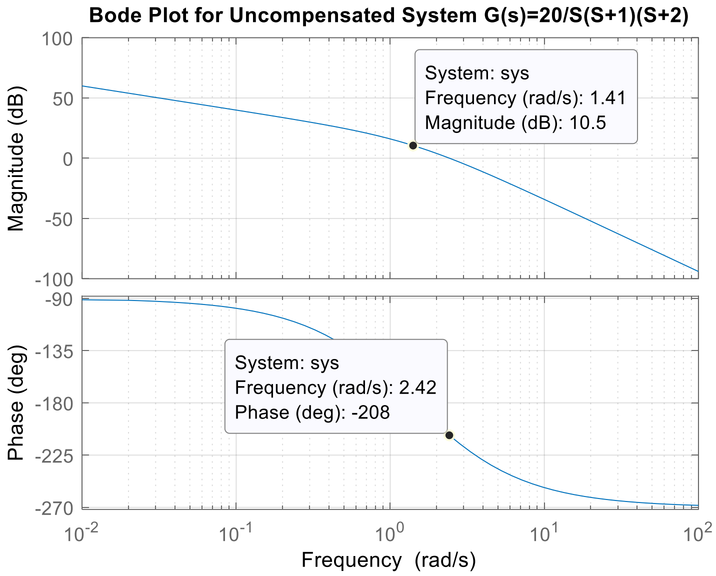
```
fprintf('Phase Margin is: %0.2f degs\n',Pm );
fprintf('Gain Crossover Frequency is: %0.2f rad/sec\n',Wcg );
fprintf('Phase Crossover Frequency is: %0.2f rad/sec\n',Wcp);

W=logspace(-1,1,100)';

%% Bode Plot for Lag Section
[mag,ph]=bode(G,W);
ph=reshape(ph,100,1);
mag=reshape(mag,100,1);
PM=-180+50+5;
Wg=interp1(ph,W,PM);
beta=interp1(ph,mag,PM);
tau=8/Wg;
disp('The Transfer Function of the Lag Section:')
D=tf([tau 1],[beta*tau 1])

%% Bode Plot for Lead section
alpha=20/beta;
mag=20*log10(mag);
Gm=-20*log10(1/sqrt(alpha));
Wm=interp1(mag,W,-20*log10(1/sqrt(alpha)));
tau=1/(Wm*sqrt(alpha));
disp('The Transfer Function of the Lead Section:')
E=tf([tau 1],[alpha*tau 1])

%% Bode Plot of Lead-Lag Compensator
disp('The Transfer Function of the Lead-Lag Compensator:')
Gc1=D*E*G
figure(2);
bode(Gc1);
title('Bode Plot for the Lead-Lag compensated System')
grid;
[Gm1,Pm1,Wcg1,Wcp1]=margin(Gc1);
```

```
Gm1dB=20*log10(Gm1);
disp('Results of Lead-Lag Compensated System:');
fprintf('Gain Margin is: %0.2f dB\n', Gm1dB);
fprintf('Phase Margin is: %0.2f degs\n',Pm1 );
fprintf('Gain Crossover Frequency is: %0.2f rad/sec\n',Wcg1 );
fprintf('Phase Crossover Frequency is: %0.2f rad/sec\n',Wcp1);
```

RESULT:

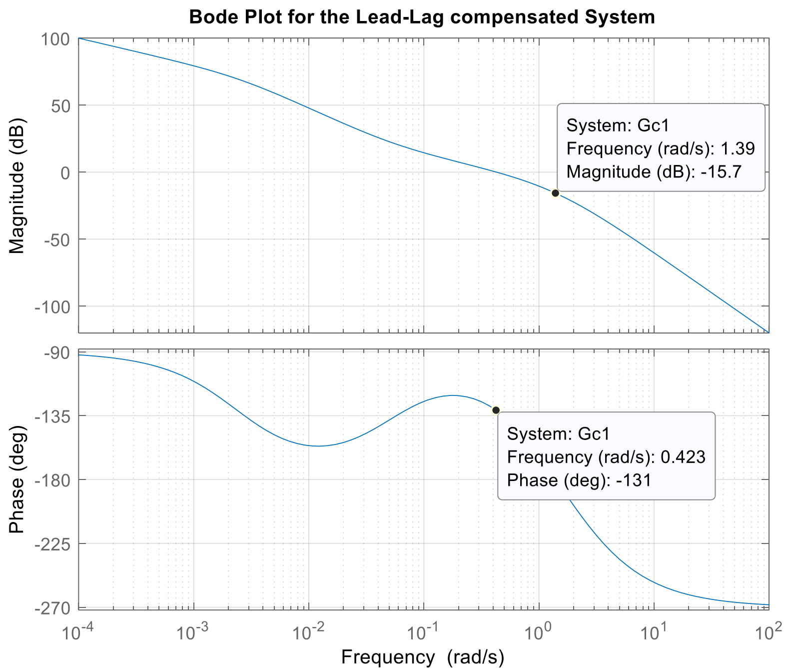
```
Results of Uncompensated System:
Gain Margin is: -10.46 dB
Phase Margin is: -28.08 degs
Gain Crossover Frequency is: 1.41 rad/sec
Phase Crossover Frequency is: 2.43 rad/sec

The Transfer Function of the Lag Section:
D =  18.84 s + 1
      -----------
     399.4 s + 1
  Continuous-time transfer function.

The Transfer Function of the Lead Section:
E =  0.4195 s + 1
     ------------
     0.3957 s + 1
 Continuous-time transfer function.

The Transfer Function of the Lead-Lag Compensator:
Gc1 =       158 s^2 + 385.1 s + 20
       -----------------------------------------------
       158 s^5 + 873.9 s^4 + 1516 s^3 + 802.6 s^2 + 2 s
 Continuous-time transfer function.

Results of Lead-Lag Compensated System:
Gain Margin is: 15.74 dB
Phase Margin is: 48.58 degs
Gain Crossover Frequency is: 1.40 rad/sec
```

Phase Crossover Frequency is: 0.43 rad/sec



The Frequency response of the uncompensated system shows:

- Gain Margin=0dB-Magnitude(dB)=-10.5 dB
- Phase Margin=Phase(deg)-180^0=-28 degs



The Frequency response of the Lead-Lag Compensated system shows:

- Gain Margin=0dB-Magnitude(dB)=15.7 dB
- Phase Margin=Phase(deg)-180^0=49 degs

Experiment 8: Using Suitable simulation package study of speed control of DC motor using

i) Armature control ii) Field control

AIM: To study the speed regulation of DC Motor using Armature Control and Field Control

THEORY: Many applications require a DC motor's speed to adjust, maximizing machine function and performance. Doing so intentionally and as necessary requires speed control. Operators can do this manually or rely on automated technology devices. Speed control of a DC motor differs from speed regulation, which is keeping a continuous speed despite load variances.

Speed controllers come in two primary forms — armature controls and field controls. Changes in the terminal voltage or external resistance impact function as Armature Controls. Conversely, changing the magnetic flux is a method of Field Control.

Armature Control for DC Motors:

With armature control the voltage is varied using several methods. One way is by implementing armature resistance, which involves connecting a variable resistance in series to the circuit of the armature. Once resistance has been increased, the current flow through the circuit is reduced and the armature voltage drop is less than the line voltage. This in turn reduces the motor speed in proportion to the voltage that’s being applied. The armature resistance control method is used in applications that require motor speed variation for shorter periods of time, not continuously.

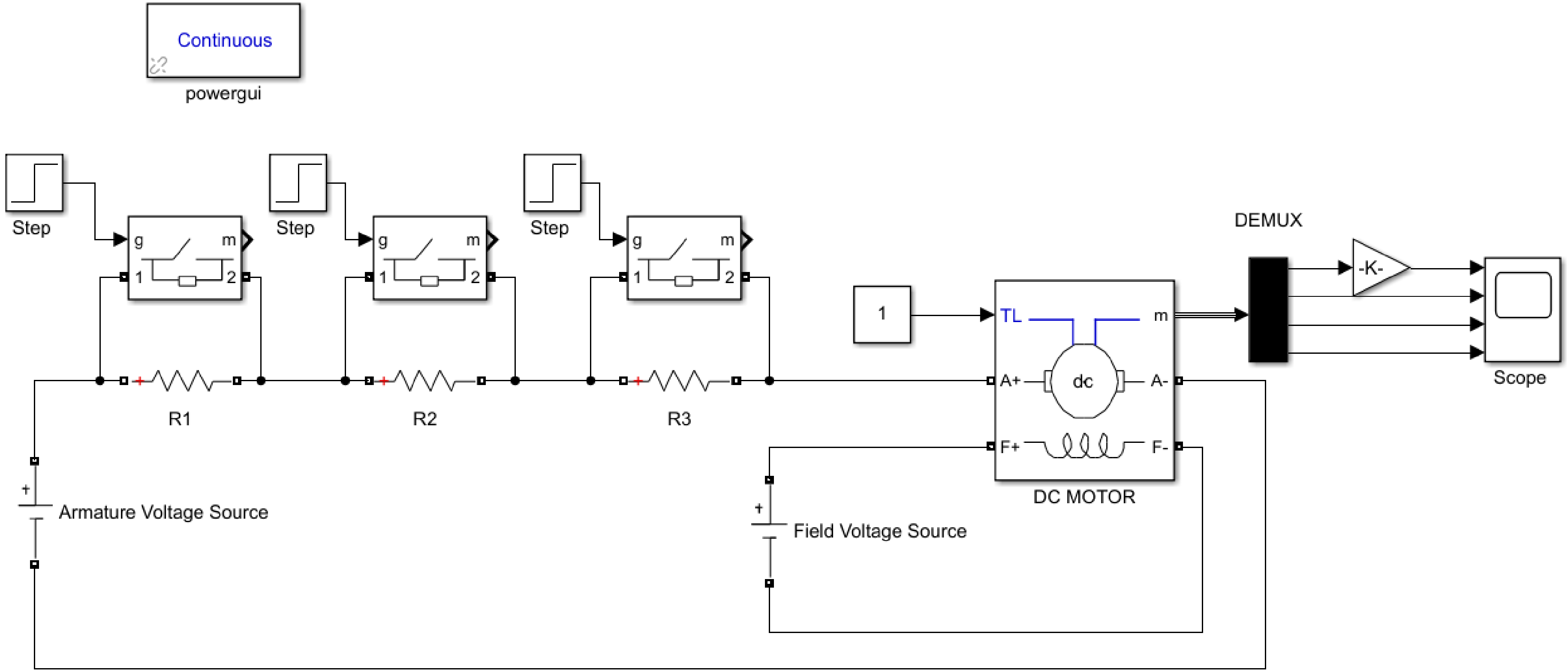
Field Control Method:

When using the field control method for DC motors, the field is weakened to increase the speed or it can be strengthened to reduce the motor’s speed. Attaining speeds that are above the rated speed can be achieved by providing variable resistance in series to the field circuit, varying the reluctance of the magnetic circuit, or by varying the applied voltage of the motor to the field circuit (with constant voltage being supplied to the armature circuit).

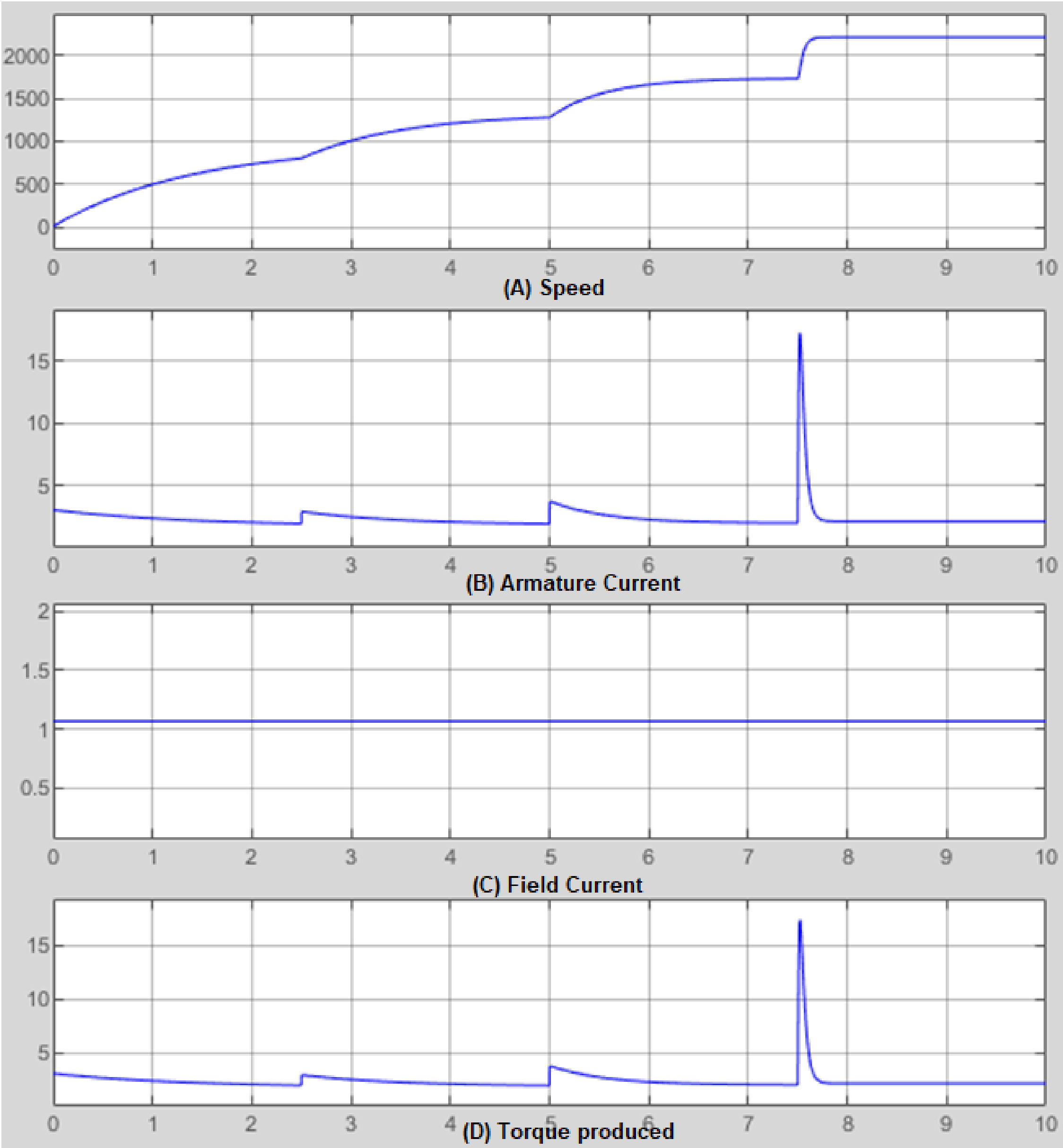
SIMULATION CIRCUIT:

- Armature Voltage Source= 240 V
- Field Voltage Source= 300 V
- Gain $K = {}^{60}/_{2\pi}$ to convert the speed in rad/sec to RPM

1. Speed regulation using Armature control



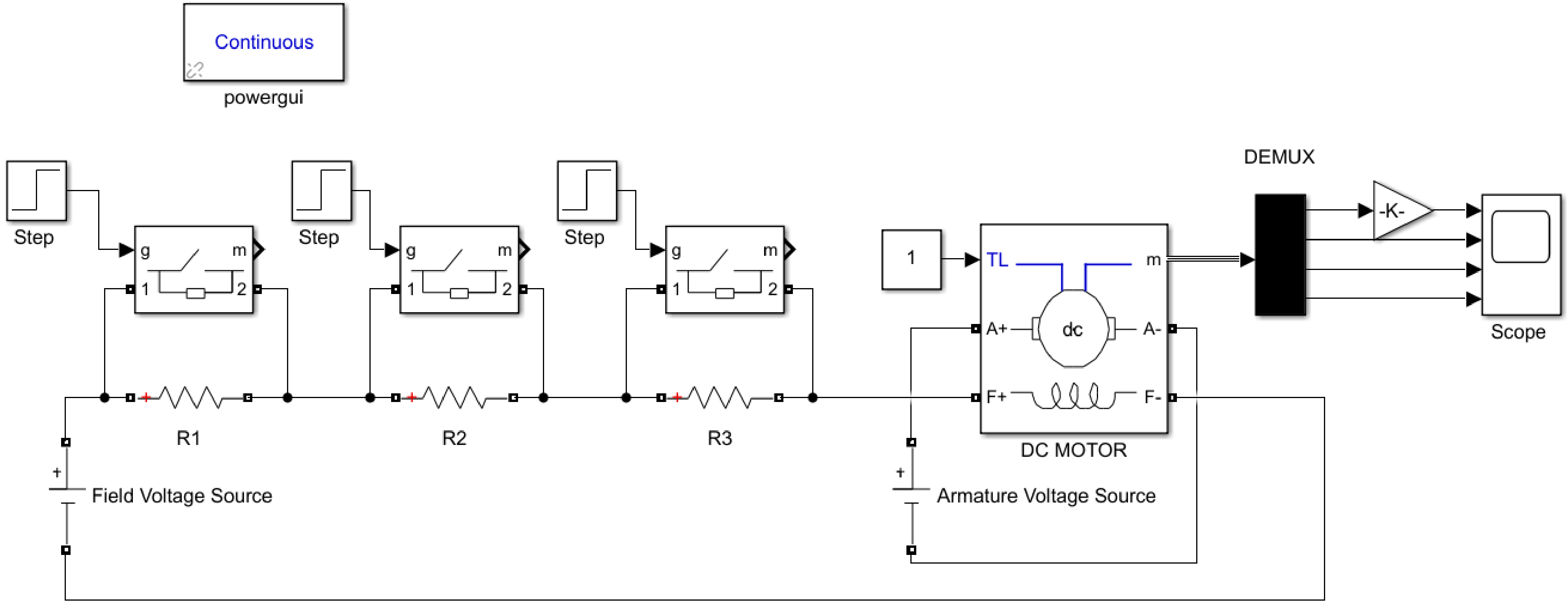
- $R_1 = R_2 = R_3 = 25\Omega$
- Ideal Switches are used to cut off resistors from the circuit
- Step Inputs are turned on every 2.5 seconds



Speed Regulation using Armature Control

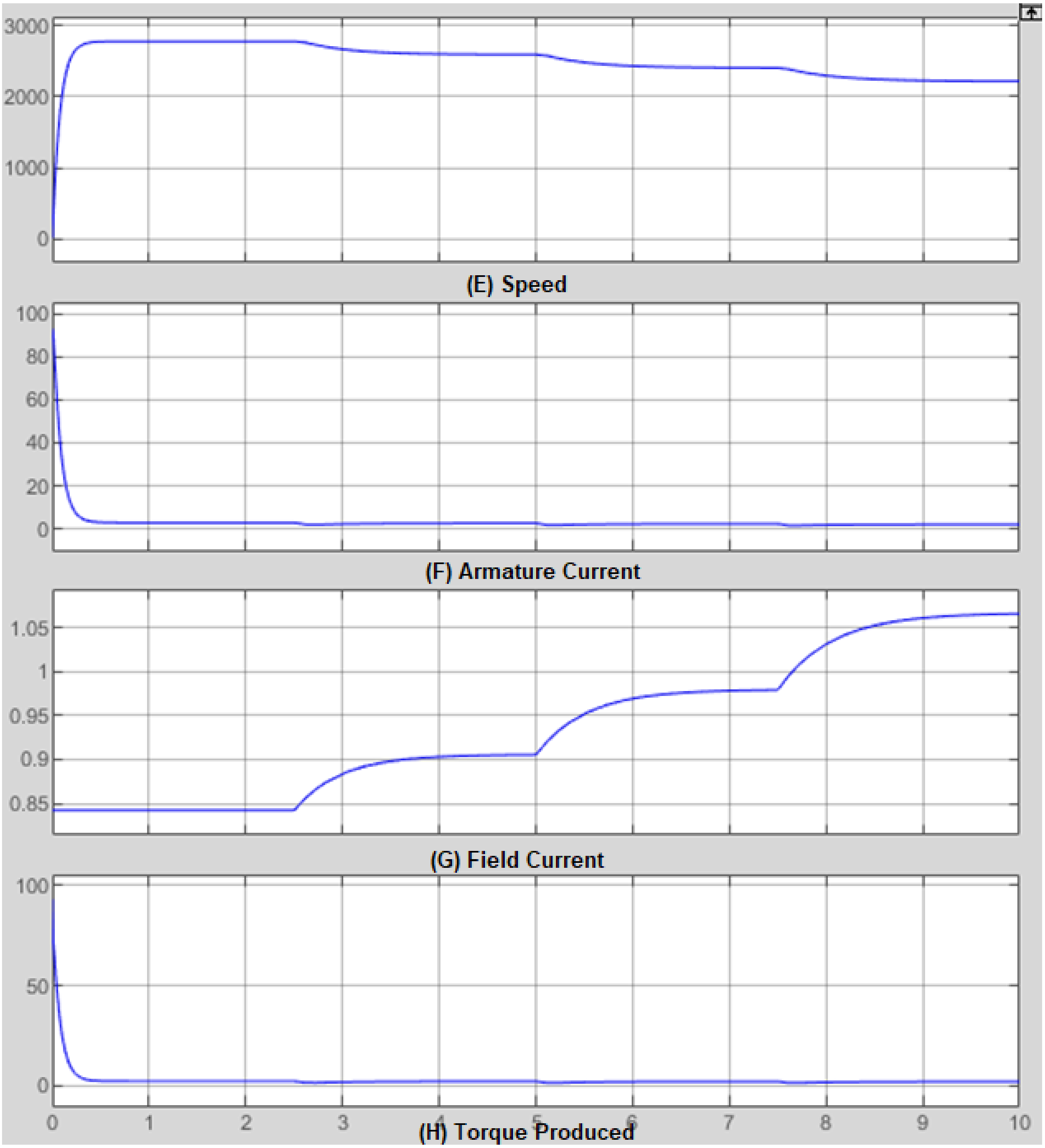
- Figure A shows the increase in Speed. Every 2.5 seconds one resistance will be cut-off causing a rise in the speed. After 7.5 seconds all resistors are cut-off and the speed is maximum over 2000 RPM.
- Figure B shows the sudden increase and drop in the Armature Current as each Resistor is turned off every 2.5 seconds and a sudden surge is observed when all three resistors are turned off after 7.5 seconds.
- Figure C shows that Field Current remains constant and unaffected by the change in the Armature Current.
- Figure D shows that Torque produced follows the pattern seen in Armature Current.

2. Speed regulation using Field Control



- Figure E shows the decrease in Speed. Initially the Speed is maximum over 2000 RPM. Every 2.5 seconds one resistance will be cut-off causing a decrease in the speed. After 7.5 seconds all resistors are cut-off and the speed is minimum over 2000 RPM.

- Figure F shows that Armature Current is initially high and then drops. Figure G shows the current in Field winding. Initially the current is low due to the three resistors connected in series, causing the speed to be the maximum. Turning off resistors causes increase in the current and lowering of speed. Figure H shows that Torque produced follows the pattern seen in Armature Current.



Speed Regulation using Field Control

Experiment 8: Using suitable simulation package, draw Root locus & Bode plot of the given Transfer function

AIM: To plot the Root Locus and Bode plot of the Transfer function $T(s) = \frac{s+2}{s^2+4s+3}$

THEORY:

Root locus is a graphical method in which the movement of poles in the s-plane can be located when a gain of the system is varied from 0 to infinity. The advantages of root locus are that it enhances system designing with better accuracy, helps to determine the gain margin, relative stability, phase margin, and the system's settling time, helps in analyzing the performance of the control system.

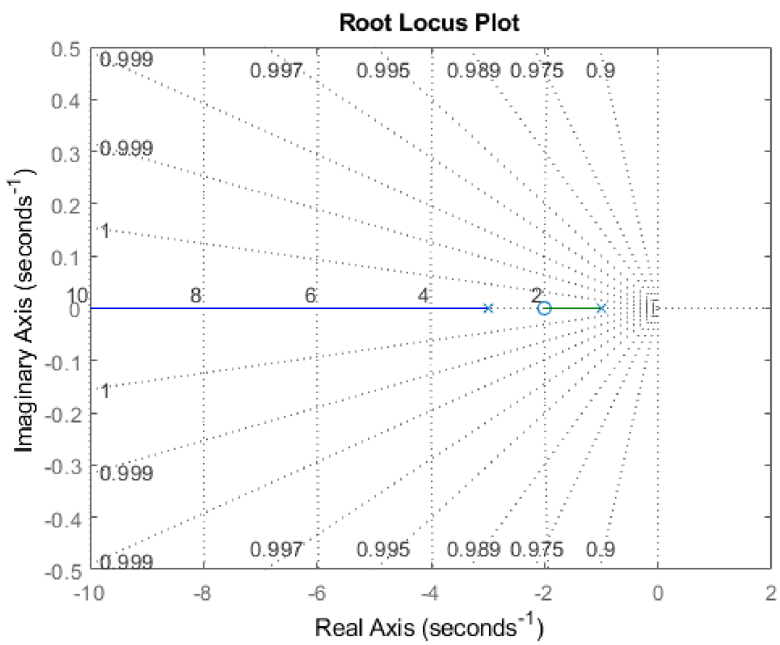
Bode Plots show the frequency response, that is changes in magnitude and phase as a function of frequency. The advantage of using Bode plot is that they provide a straightforward and common way of describing the frequency response of a linear time invariant system. This is done on semi-log scale plots: Magnitude or ‘gain’ plot in dB and Phase plot in degrees. The information in a Bode plot can be used to quantify the stability of a feedback system by using phase and gain margins

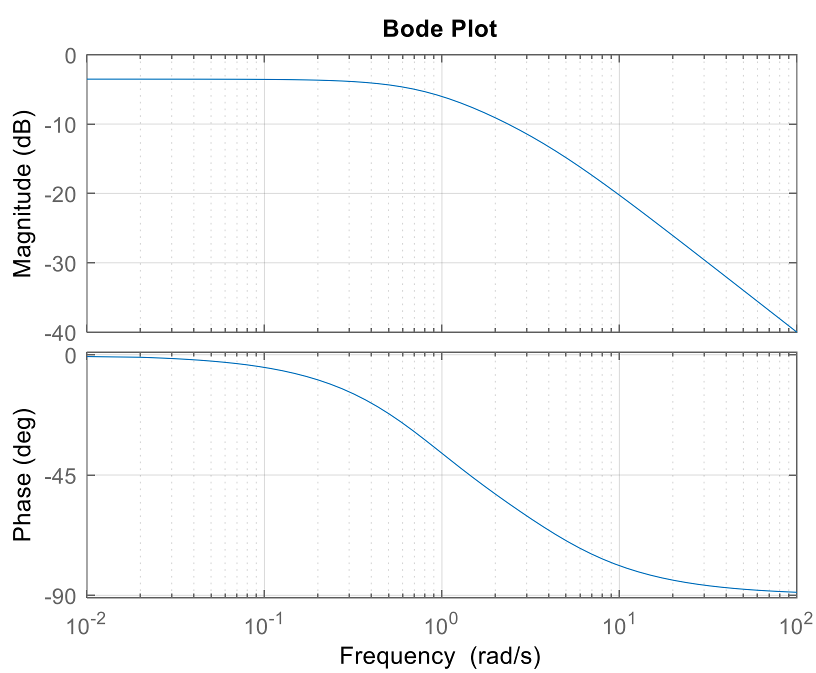
- **Phase margin** is measured at the frequency where gain equals 0 dB, also referred to as the “crossover frequency”. Phase margin is a measure of the distance from the measured phase to a phase shift of -180°.
- **Gain margin** is measured at the frequency where the phase shift equals -180°. Gain margin indicates the distance, in dB, from the measured gain to a gain of 0 dB.

```
% Given Transfer Function
num = [1 2];
den = [1 4 3];
sys = tf(num, den);   % Transfer function ‘sys’ using the numerator and denominator coefficients

% Root Locus Plot
figure;
rlocus(sys);   % ‘rlocus’ function to plot the root locus of the system
grid on;
title('Root Locus Plot');

% Bode Plot
figure;
```

```
bode(sys);
grid on;
title('Bode Plot');  % 'bode' function to plot the Bode plot of the system
```

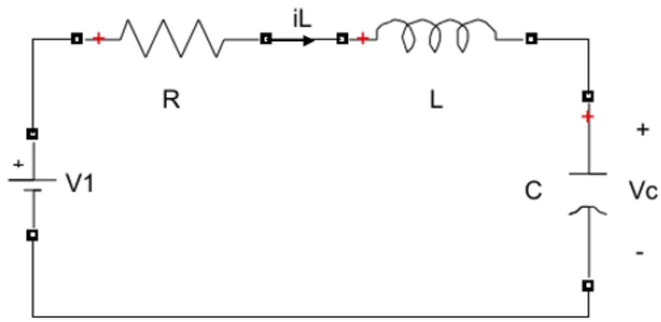
RESULT:





Experiment 9: Using suitable simulation package, obtain the time response from state model of a system.

AIM: To develop a state model for the given Electrical Network and obtain the time response



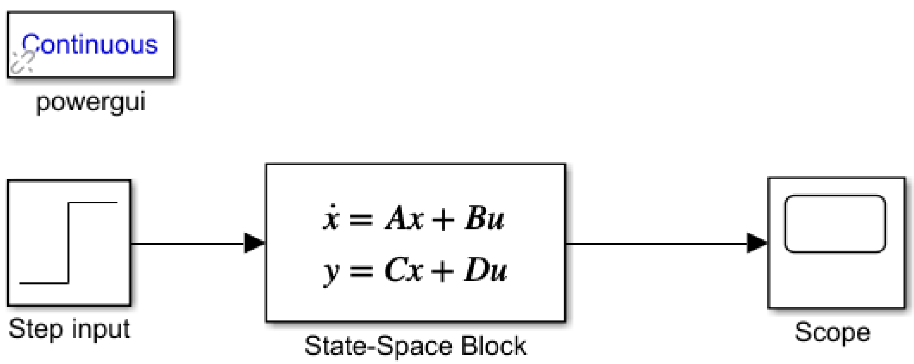
Where, $L = 10H$, $C = 0.2\ F$ and $R = 6\Omega$

THEORY: For the circuit given the state equations are:

$$\begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1/C \\ -1/L & -R/L \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1/L \end{bmatrix} [u]$$

$$y = [0 \quad 1] \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix} + [0][V_1]$$

SIMULINK MODEL:



% Given State Model

L=10;

Cap=0.2;

R=6;

A=[0 1/Cap; -1/L -R/L];

B=[0; 1/L];

C=[0 1];

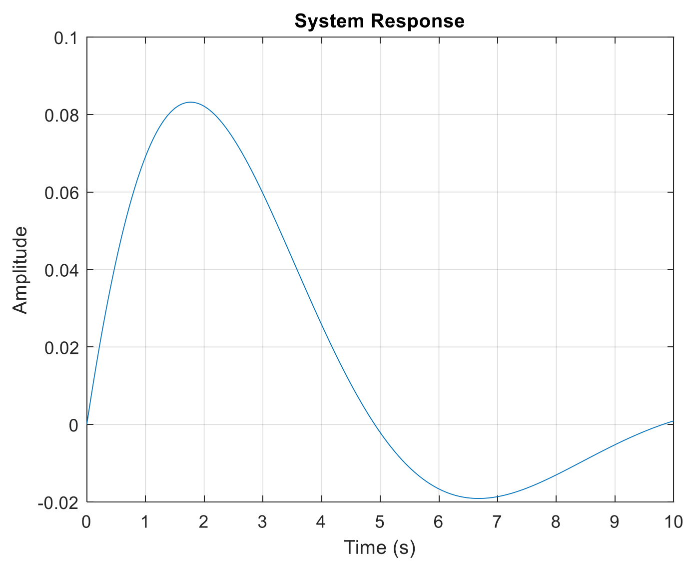
D=[0];

sys = ss(A, B, C, D);

% Define Input Signal

```
t = linspace(0, 10, 1000);
u=ones(1,length(t))

% Simulate System Response
[y, t, x] = lsim(sys, u, t);

% Plot System Response
figure;
plot(t, y);
grid on;
title('System Response');
xlabel('Time (s)');
ylabel('Amplitude');
```

RESULT:



Time Response from State Model of a system

The Time response using State Model of a system, excited by a Step Input has been observed and the Maximum Peak Amplitude is found to be 0.0832 within 2 seconds of the excitation.

Experiment 10: Demonstrate the implementation of PI and PD controller

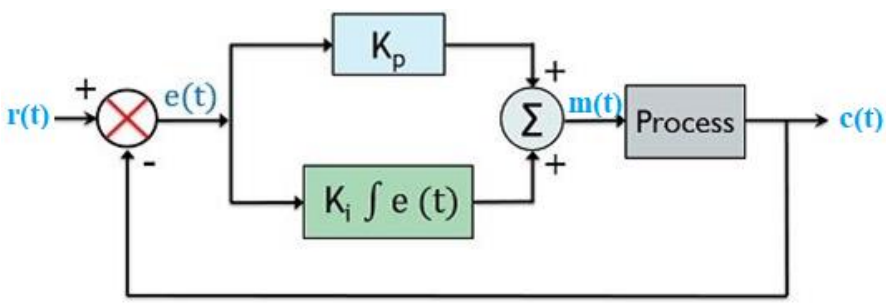
AIM: To study the effect of P, PI, PD and PID controller on the step response of a feedback control system.

THEORY:

1. **Proportional Integral (PI) Controller:** Proportional Integral controller sometimes also known as **proportional plus integral (PI) controllers**. It is a type of controller formed by combining proportional and integral control action. Thus, it is named as PI controller. In the proportional-integral controller, the control action of both proportional, as well as the integral controller, is utilized. This combination of two different controllers produces a more efficient controller. In this case, the control signal shows proportionality with both the error signal as well as with integral of the error signal. Mathematical representation of proportional plus integral controller is given as:

$$m(t) = K_p\, e(t) + K_i \int e(t)$$

The figure below represents the block diagram of the system with PI controller:



In order to have the transfer function of the controller, we need to consider the Laplace transform of the above equation, so it is given as:

$$M(s) = K_p\, E(s) + K_i \frac{E(s)}{s}$$

Taking the common term i.e., E(s) out, we will get

$$M(s) = E(s)\, [\, K_p + \frac{K_i}{s}\,]$$

The error signal will act as input that will cause variation in the output of the controller,

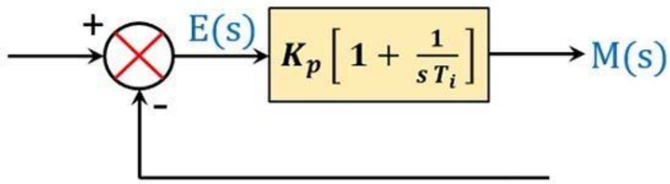
$$\frac{M(s)}{E(s)} = K_p + \frac{K_i}{s}$$

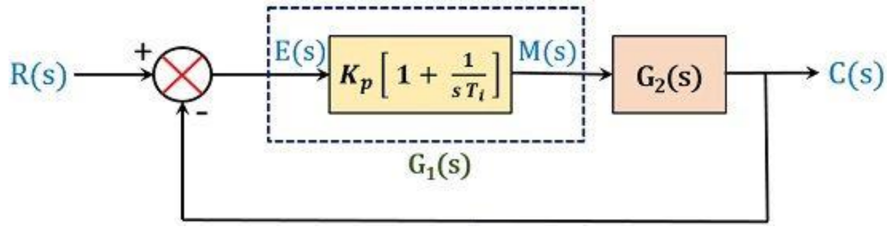
$$\frac{M(s)}{E(s)} = K_p \left[1 + \frac{K_i}{K_p s} \right]$$

$$\frac{M(s)}{E(s)} = K_p\left[1 + \frac{1}{T_i s}\right]$$

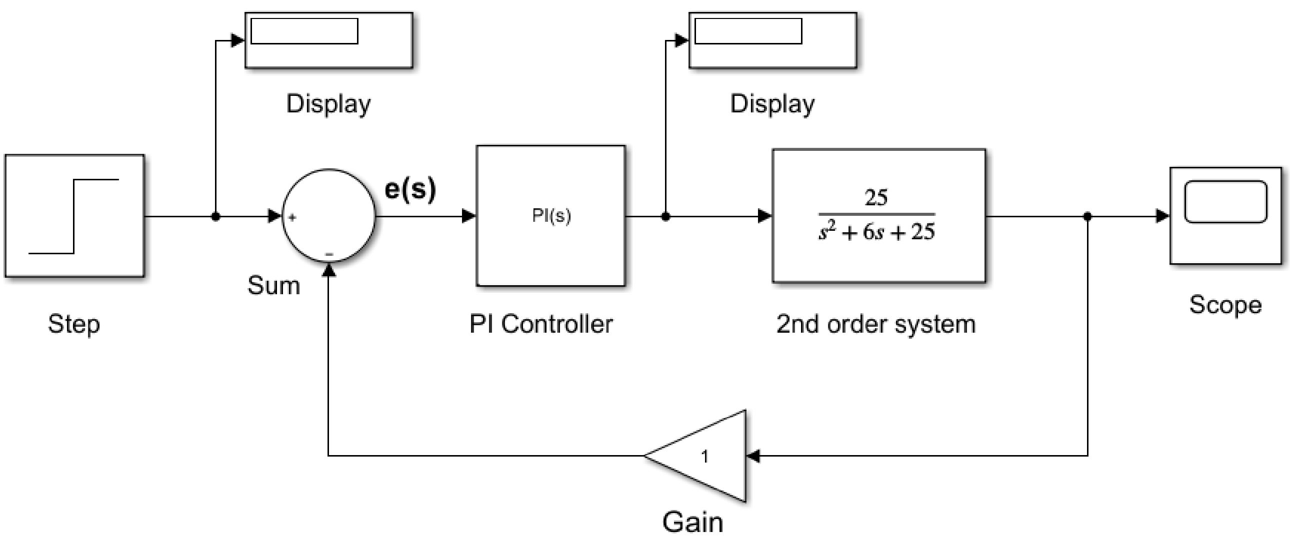
This equation represents the gain of the PI controller, where $T_i = K_p/K_i$

So, the block diagram of a PI controller is given as:





SIMULINK IMPLEMENTATION:



IMPLEMENTATION OF PI CONTROLLER

Proportional Derivative (PD) Controller

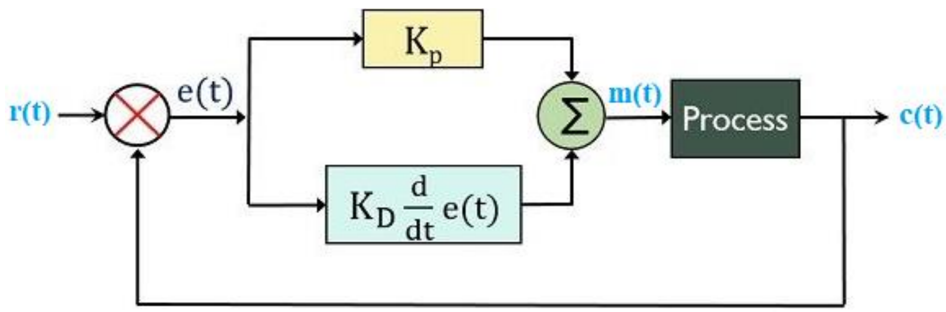
A type of controller in a control system whose output varies in proportion to the error signal as well as with the derivative of the error signal is known as the proportional derivative controller. It is also known as a proportional plus derivative controller or PI controllers.

This type of controller provides combined action of both proportional and derivative control action. Presence of two distinct control action generates a more precise system.

For the PD controller, the output is given as:

$$m(t) = K_p\, e(t) + K_D \frac{d}{dt} e(t)$$

The block diagram of a control system comprising of PD controller is given below:



K_p is the constant of proportionality of error signal,

K_D is the constant of proportionality of derivative of the error signal.

Laplace transform of the above equation,

$$M(s) = K_P\, E(s) + K_D\, sE(s)$$

$$M(s) = E(s)\, [K_p + sK_D]$$

Transfer function is given as:

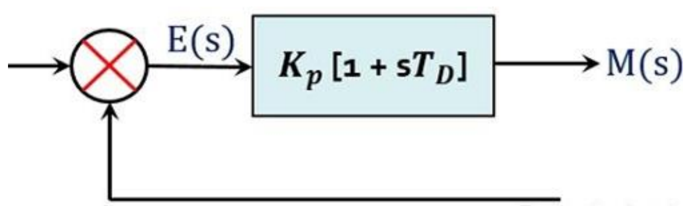
$$\frac{M(s)}{E(s)} = K_p + sK_D$$

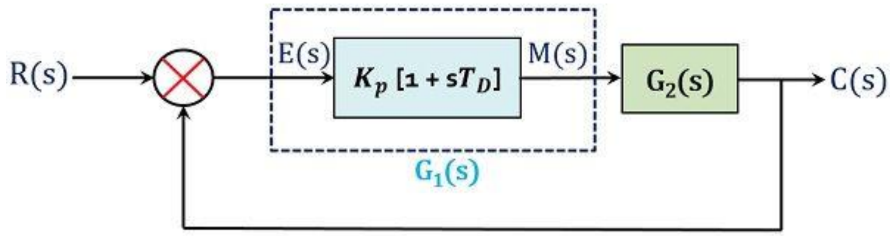
$$\frac{M(s)}{E(s)} = K_p[1 + s\frac{K_D}{K_p}]$$

$$\frac{M(s)}{E(s)} = K_p[1 + sT_D]$$

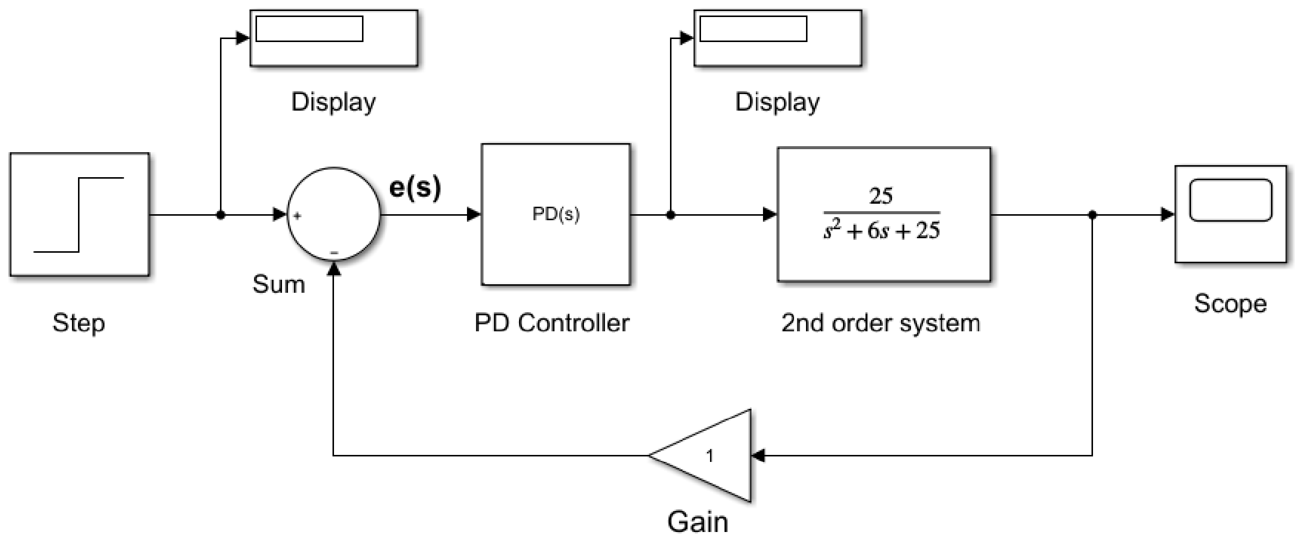
This is defined as the gain of the PD controller: $T_D = K_D/K_p$

PD controller with gain is represented as:





SIMULINK IMPLEMENTATION



IMPLEMENTATION OF PD CONTROLLER

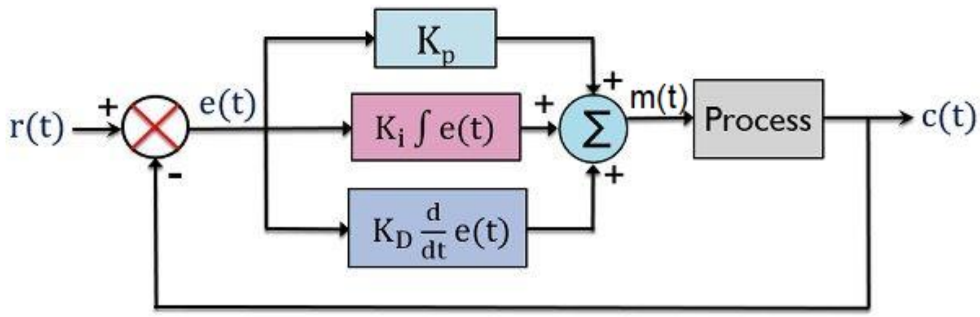
Experiment 11: Implement a PID Controller and hence realize an Error Detector

AIM: To implement the PID controller and demonstrate its application as an Error Detector

THEORY: Proportional Integral Derivative (PID) Controller: It is a controller in which the output of the controller varies in proportion with the error signal, integral of the error signal and derivative of the error signal is known as the Proportional Integral Derivative Controller. Proportional plus integral plus derivative controller is sometimes referred as a 3-mode controller, as it combines the controlling action of proportional, integral as well as derivative controller altogether. To eliminate the respective disadvantages of PI and PD controllers, PID controllers are used. Hence, a PID controller produces a system, that provides increased stability with a reduction in the steady-state error. The combination of all the three types of control action improves the overall performance of the control system, in order to provide the desired output in an effective manner. The output of a PID controller is given as:

$$m(t) = K_p\, e(t) + K_i \int e(t) + K_d \frac{d}{dt} e(t)$$

Block diagram of the PID controller:



Here K_p is proportionality constant for the error signal,

K_i is the proportionality constant for integral of the error signal and

K_D is the proportionality constant for the derivative of the error signal.

Laplace transform of the above equation,

$$M(s) = K_p\, E(s) + \frac{K_i}{s}\, E(s) + K_d s\, E(s)$$

$$M(s) = K_p\, E(s) + \frac{K_i}{s}\, E(s) \ + K_d s\, E(s)$$

$$M(s) = E(s)[K_p + \frac{K_i}{s} + K_d s]$$

Transfer of the controller will be given as,

$$\frac{M(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d s$$

$$\frac{M(s)}{E(s)} = K_p[1 + \frac{K_i}{K_p s} + \frac{K_d s}{K_p}]$$

Considering, $T_i = K_p/K_i$ and $T_D = \frac{K_D}{K_p}$,

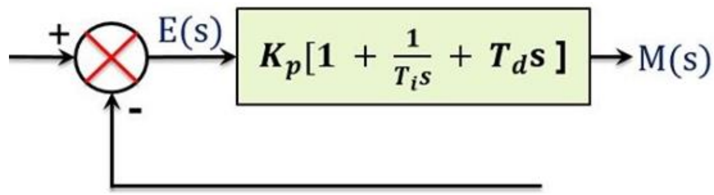
$$\frac{M(s)}{E(s)} = K_p[1 + \frac{1}{T_i s} + T_d s\,]$$

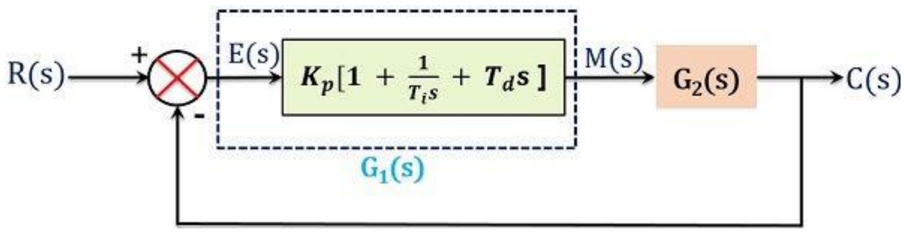
$$\frac{M(s)}{E(s)} = \frac{K_p S + K_p T_d S^2 + \frac{K_p}{T_i}}{S}$$

$$\frac{M(s)}{E(s)} = \frac{K_p S + K_d S^2 + K_i}{S}$$

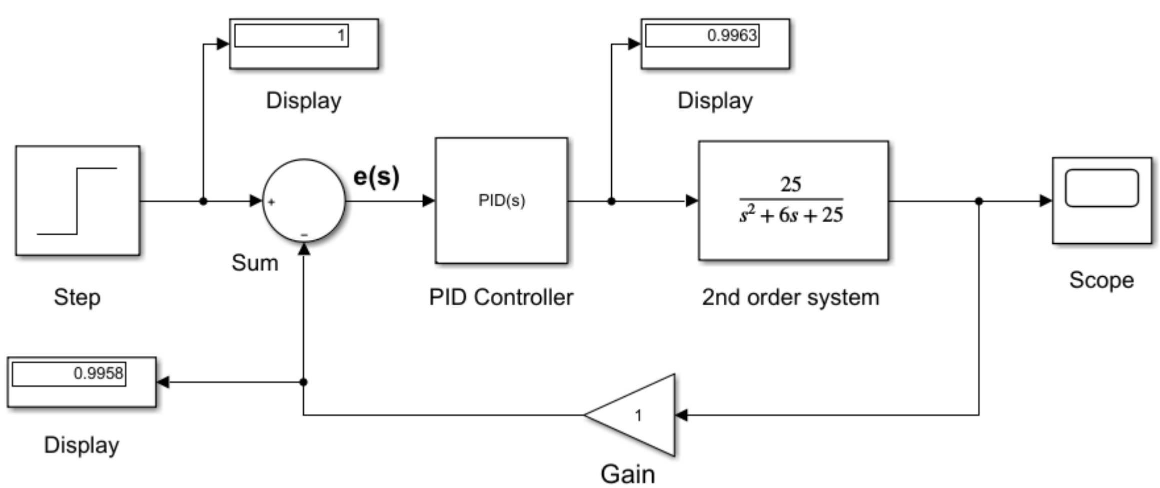
This is the simplified transfer function of the PID controller.

The PID controller in the form of block diagram with gain is represented below:





SIMULINK IMPLEMENTATION



ERROR DETECTOR USING PID CONTROLLER

Experiment 12: Demonstrate the effect of PI, PD and PID controller on the system response

AIM: To study the behavior of PI, PD and PID controller in response to step input using SIMULINK

THEORY:

Proportional Integral (PI) Controller: In the proportional-integral controller, the control action of both proportional, as well as the integral controller, is utilized. In this case, the control signal shows proportionality with both the error signal as well as with integral of the error signal. Mathematical representation of proportional plus integral controller is given as:

$$m(t) = K_p\, e(t) + K_i \int e(t)$$

Here K_p is proportionality constant for the error signal,

K_i is the proportionality constant for integral of the error signal

Proportional Derivative (PD) Controller: This type of controller provides combined action of both proportional and derivative control action. Presence of two distinct control action generates a more precise system. For the PD controller, the output is given as:

$$m(t) = K_p\, e(t) + K_D \frac{d}{dt} e(t)$$

K_p is the constant of proportionality of error signal,

K_D is the constant of proportionality of derivative of the error signal.

Proportional Integral Derivative (PID) Controller: Proportional plus integral plus derivative controller is combines the controlling action of proportional, integral as well as derivative controller altogether. The combination of all the three types of control action improves the overall performance of the control system, in order to provide the desired output in an effective manner. The output of a PID controller is given as:

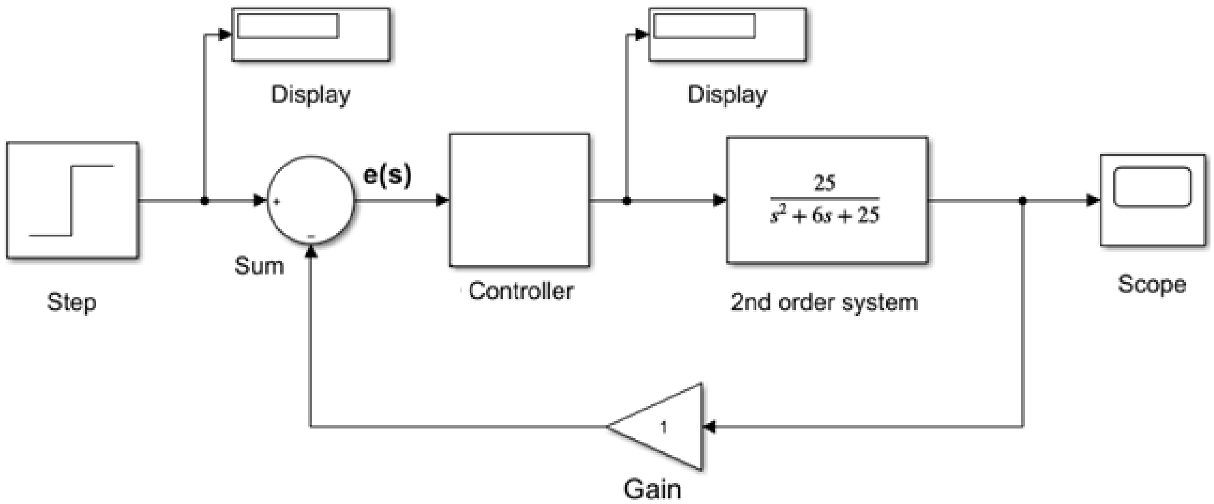
$$m(t) = K_p\, e(t) + K_i \int e(t) + K_d \frac{d}{dt} e(t)$$

Here K_p is proportionality constant for the error signal,

K_i is the proportionality constant for integral of the error signal and

K_D is the proportionality constant for the derivative of the error signal.

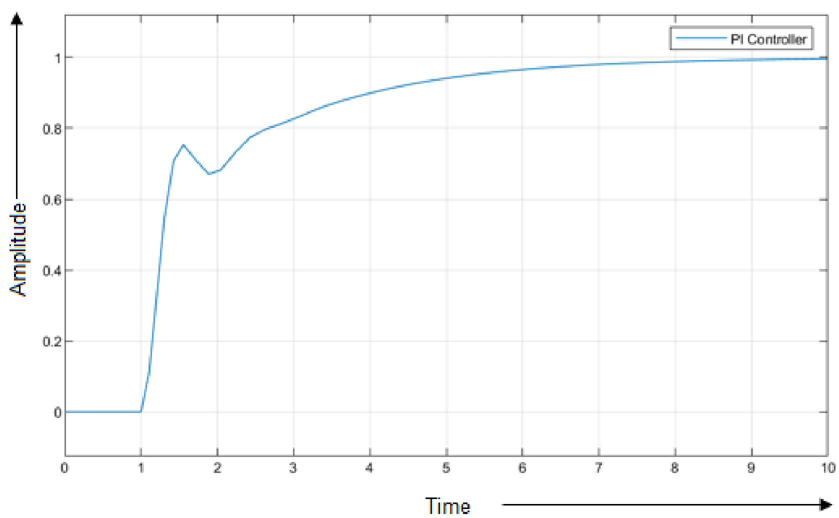
BLOCK DIAGRAM FOR SIMULINK IMPLEMENTATION:



PROCEDURE:

- The block diagram is assimilated using SIMULINK
- Step input is provided, starting at 0 seconds
- The Controller Block is modelled as PI, PD and PID controller separately
- Individual values of Step input and Controller output are observed using Displays
- Response of the different controllers with respect to time are observed on the Scope

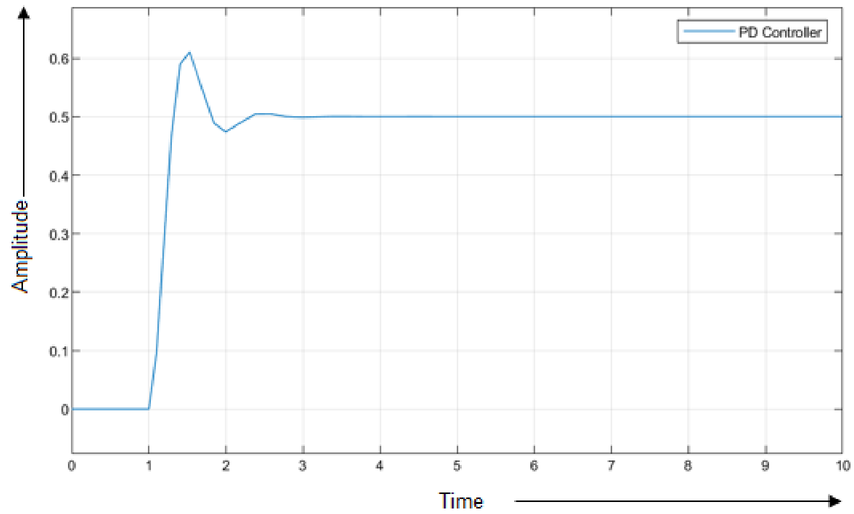
1. PI Controller



Step response using PI Controller

The step response of the PI Controller shows that the transient response is triggered after 1 second of the stimulation and reaches steady state within 8 seconds of the stimulation. Maximum steady state Amplitude achieved is equal to the Amplitude of the Step Input applied.

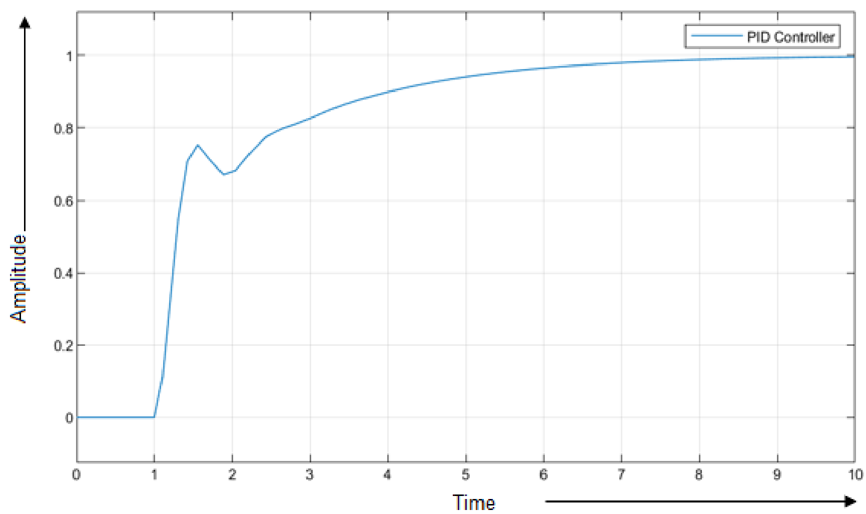
2. PD controller



Step response using PD Controller

The step response of the PD Controller shows that the transient response is triggered after 1 second and reaches steady state within 3 seconds of the stimulation. Maximum Transient Amplitude achieved is around 80% of the Step Input applied. Steady State amplitude is approximately 50% of the applied input.

3. PID Controller



Step response using PID Controller

The step response of the PI Controller shows that the transient response is triggered after 1 second and reaches steady state within 8 seconds of the stimulation. Maximum steady state Amplitude achieved is equal to the Amplitude of the Step Input applied.