EE 5351 : CONTROL SYSTEMS DESIGN

LABORATORY 03

NAME	: BALASOORIYA JM
REG No	: EG/2021/4424
GROUP No	: CE 07
DATE	: 04/03/2025

Table 1: Summative Laboratory Form

Semester	05
Module Code	EE 5351
Module Name	Control System Design
Lab Number	03
Lab Name	Laboratory Session-3
Lab Conduction date	05/11/2024
Report Submission date	14/02/2025

List of Figures

List of Tables

CONTENT

01.Observation

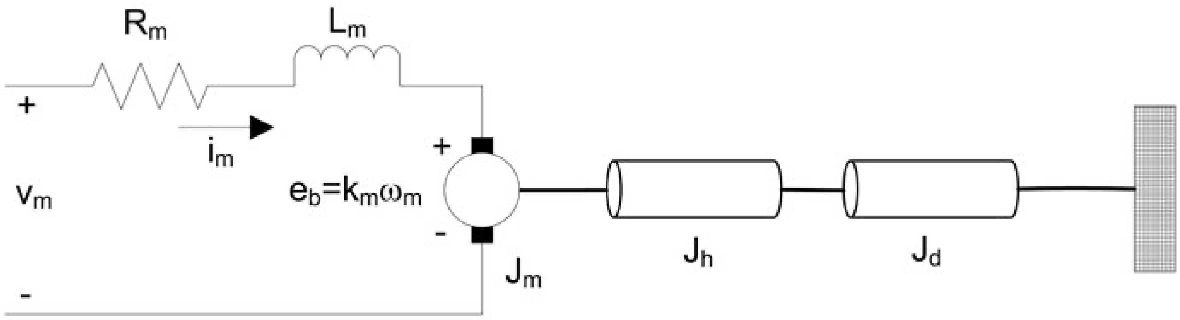


Figure 1: QUBEServo3 DC motor and load

Table 2 : QUBEServo3 parameter

Terminal Resistance (R_m)	8.4Ω
Rotor inductance(L_m)	1.16 mH
Equivalent rotor inertia(J_{eq})	2.09×10^{-5} kgm^2
Torque constant(k_t)	0.042Nm/A
Voltage constant (k_m)	0.042 Nm/A

02.Calculation

Q1)

i) Dynamic Equation for DC motor and load

$$V_m = i_m R_m + L_m \frac{dim}{dt} + e_b$$

$$e_b = k_m \omega_m$$

$$T_m = J_{eq} \frac{d\omega m}{dt}$$

$$T_m = k_t i_m$$

ii) Transfer function

$$Vm(t) = R_m i_m(t) + L_m \frac{di_m(t)}{dt} + k_m \omega_m$$

$transform\ to\ laplace\ domain$

$$Vm(s) = R_m i_m(s) + SL_m i_m(s) + k_m \omega_m$$

$$i_m(t) = \frac{J_{eq}}{k_t} \frac{d\omega_m}{dt}$$

$transform\ to\ laplace\ domain$

$$i_m(s) = \frac{SJ_{eq}}{k_t} \frac{d\omega_m(s)}{dt}$$

$$\frac{\omega_m(s)}{Vm(s)} = \frac{kt}{(SJ_{eq} + km)[Rm + LmS] + kmkt}$$

$$\omega_m(t) = \frac{d\theta_m(t)}{dt}$$

$transform\ to\ laplace\ domain$

$$\omega_m(s) = S\theta_m(s)$$

$$\frac{\Theta m(s)}{Vm(s)} = \frac{kt}{S\{JeqS[Rm + LmS] + kmkt\}}$$

According to the given data rotor inductance is negligible.
Thus:

$$\frac{\Theta m(s)}{Vm(s)} = \frac{kt}{S\{JeqRmS + kmkt\}}$$

$$\frac{\Theta m(s)}{Vm(s)} = \frac{0.042}{1.76 \times 10^{-4} S^2 + 1.764 \times 10^{-3} S}$$

iii) Simulink

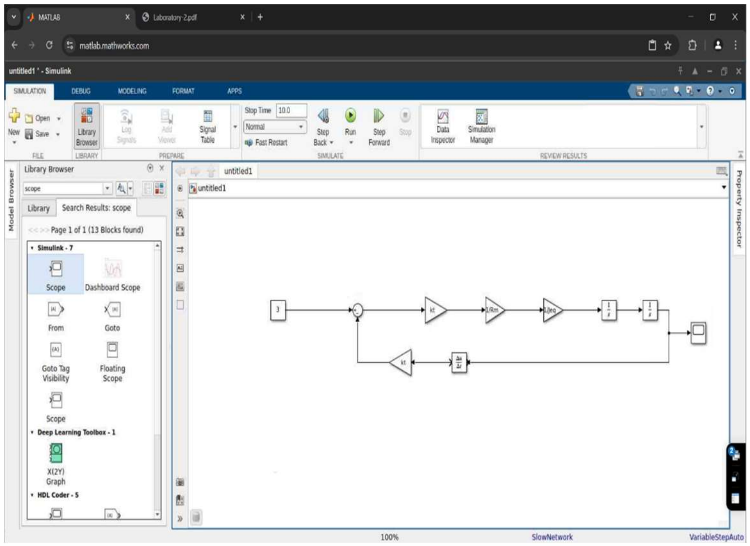


Figure 2: Simulink Q1(III)

iv) Closed loop transfer function

$$G(s) = \frac{\Theta m(s)}{Vm(s)} = \frac{0.042}{1.7556 \times 10^{-4} S^2 + 1.764 \times 10^{-3} S}$$

For closed loop

$$\frac{\theta_m(s)}{\theta_{ref}(s)} = \frac{K_P G(s)}{1 + K_p G(s)}$$

$$\frac{\theta_m(s)}{\theta_{ref}(s)} = \frac{0.042}{1.7556 \times 10^{-4} S^2 + 1.764 \times 10^{-3} S + 0.042}$$

v) Simulink

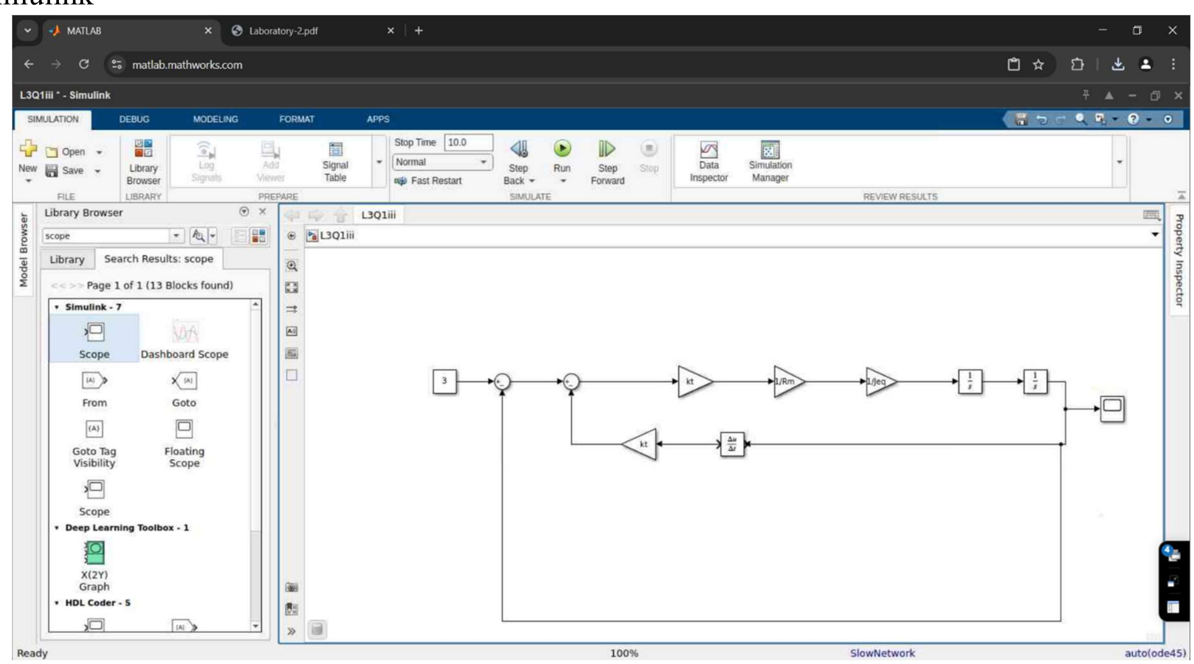


Figure 3: Simulink Q1(V)

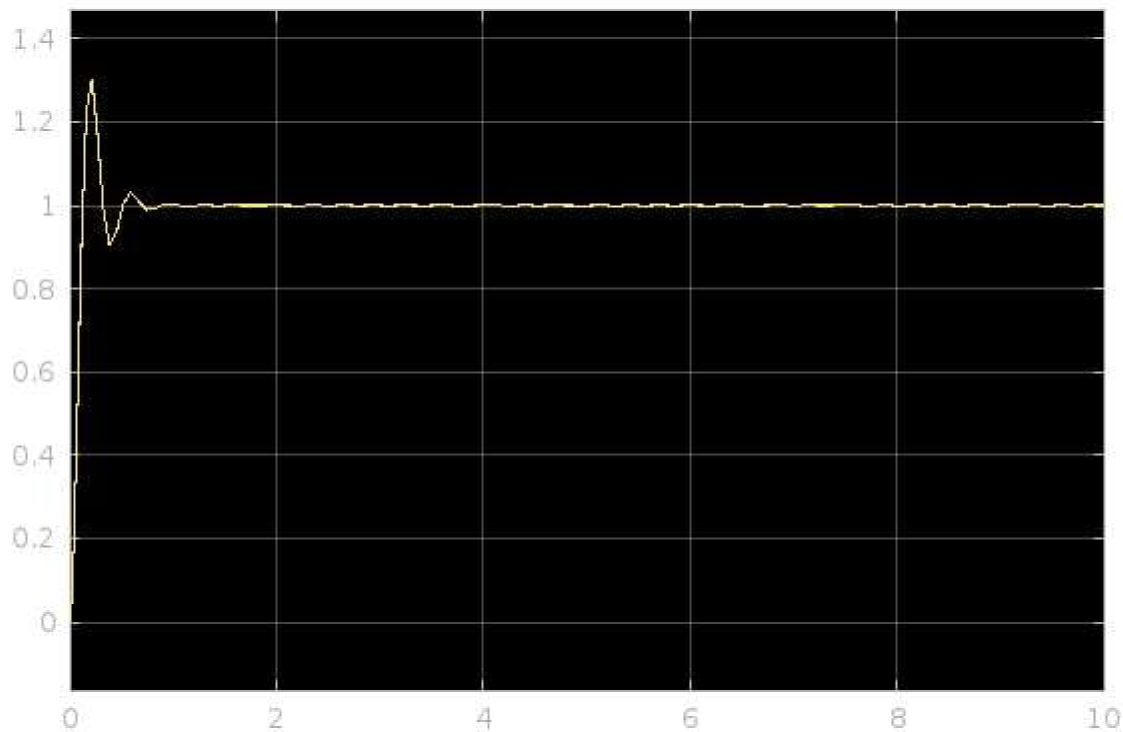


Figure 4: Output of Scope (Q1-V)

Q2)

i) Plot root locus

Matlab code

```
% Given parameters
Rm = 8.4;         % Terminal resistance (Ohm)
Lm = 1.16e-3;     % Rotor inductance (H)
Jeq = 2.09e-5;    % Equivalent rotor inertia (kg.m^2)
kt = 0.042;       % Torque constant (Nm/A)
km = 0.042;       % Voltage constant (V/rad/s)

% Transfer function G(s) = kt / (Jeq * Rm * s^2 + (kt * km) * s)
s = tf('s');
G = kt / (Jeq * Rm * s^2 + (kt * km) * s);

% Closed-loop transfer function with unity feedback
T = feedback(G, 1);

% Plot root locus of the closed-loop system
figure;
rlocus(T);
grid on;
title('Root Locus of the Closed-Loop System');
xlabel('Real Axis');
ylabel('Imaginary Axis');
```

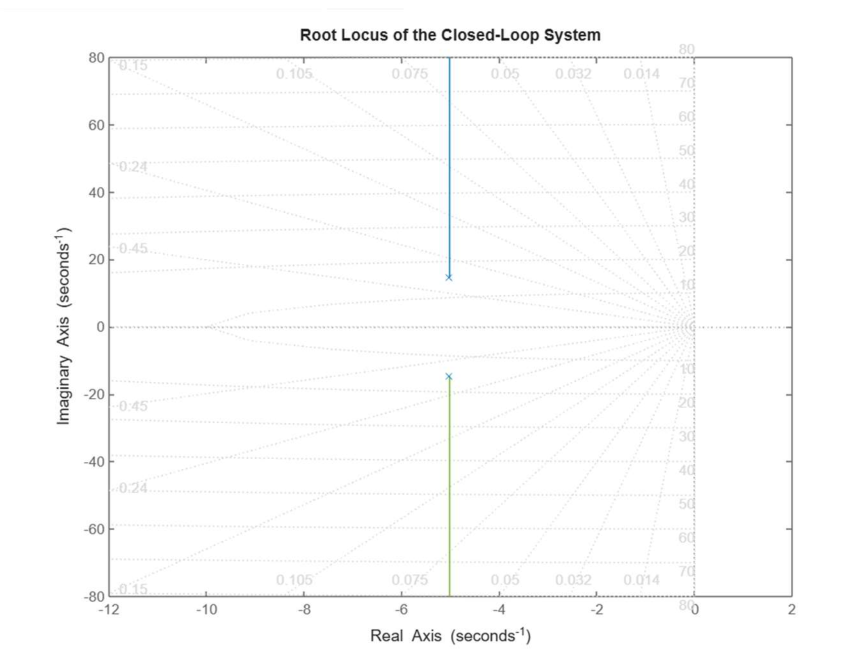


Figure 5: Root Locus of DC Motor

ii) Calculate ω

When ζ = 0.5;

Then Sin θ = 0.5'

$$S^2 + 2\zeta\omega_n S + \omega_n{}^2 = 0$$

$$2\zeta\omega_n = \frac{k_t k_m}{j_{eq} R_m}$$

$$2 \times 0.5 \times \omega_n = \frac{0.042 \times 0.042}{(2.09 \times 10^{-5})(8.4)}$$

$$\omega_n = 10.048\, rad/s$$

iii) Design compensator

Matlab code

```
% Given motor parameters

Rm = 8.4;        % Terminal resistance (Ohm)
Lm = 1.16e-3;    % Rotor inductance (H)
Jeq = 2.09e-5;   % Equivalent rotor inertia (kg.m^2)
kt = 0.042;      % Torque constant (Nm/A)
km = 0.042;      % Voltage constant (V/rad/s)

% Define s as a Laplace variable
s = tf('s');

% Open-loop transfer function G(s)
G = kt / (Jeq * Rm * s^2 + (kt * km) * s);

% Desired increase in omega_n by 10%
omega_n_old = sqrt(kt / (Jeq * Rm));  % Original natural frequency
omega_n_new = 1.1 * omega_n_old;      % 10% increase

% Lead compensator design
z = 8;   % Zero of the compensator
p = 40;  % Pole of the compensator
Kc = 1;  % Compensator gain

% Lead compensator transfer function
Gc = Kc * (s + z) / (s + p);

% Compensated open-loop transfer function
G_comp = Gc * G;

% Plot root locus for the compensated system
figure;
rlocus(G_comp);
grid on;
title('Root Locus with Lead Compensator');
xlabel('Real Axis');
ylabel('Imaginary Axis');
```

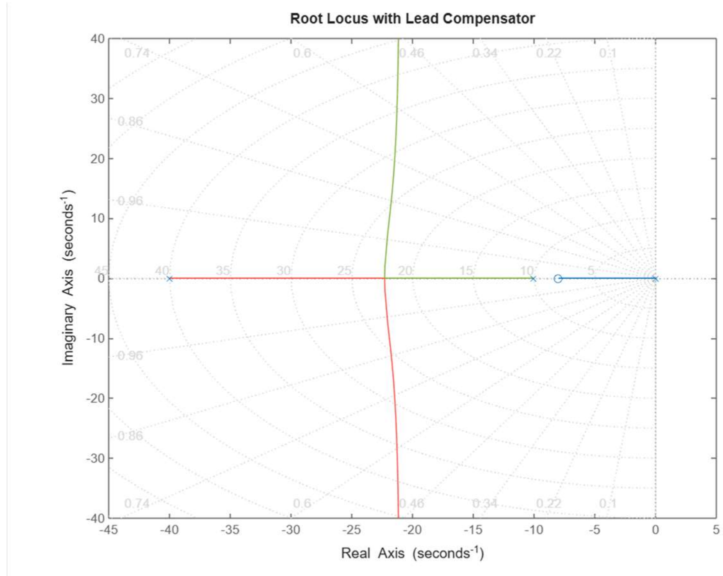
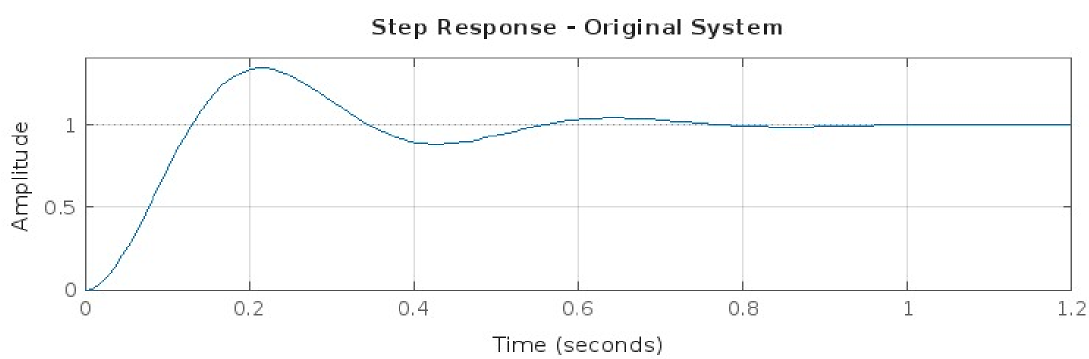


Figure 6: Root Locus after compensation

iv) Plot time domain response



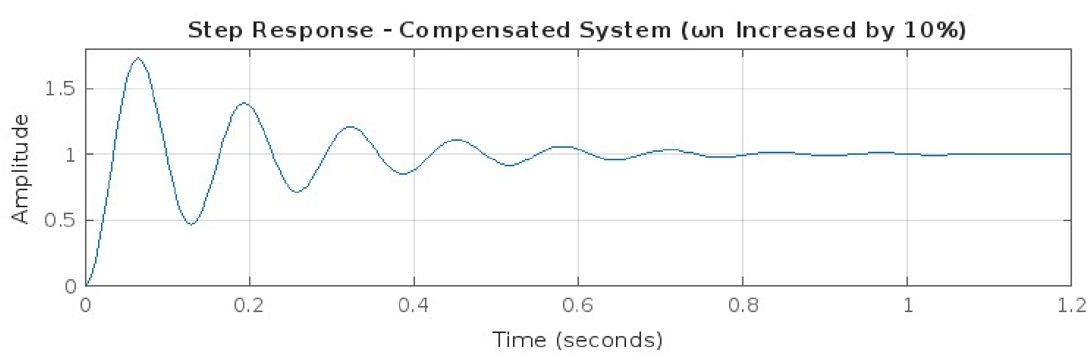


Figure 7 : Step Response before and after the compensator

Q3)

i) Design compensator

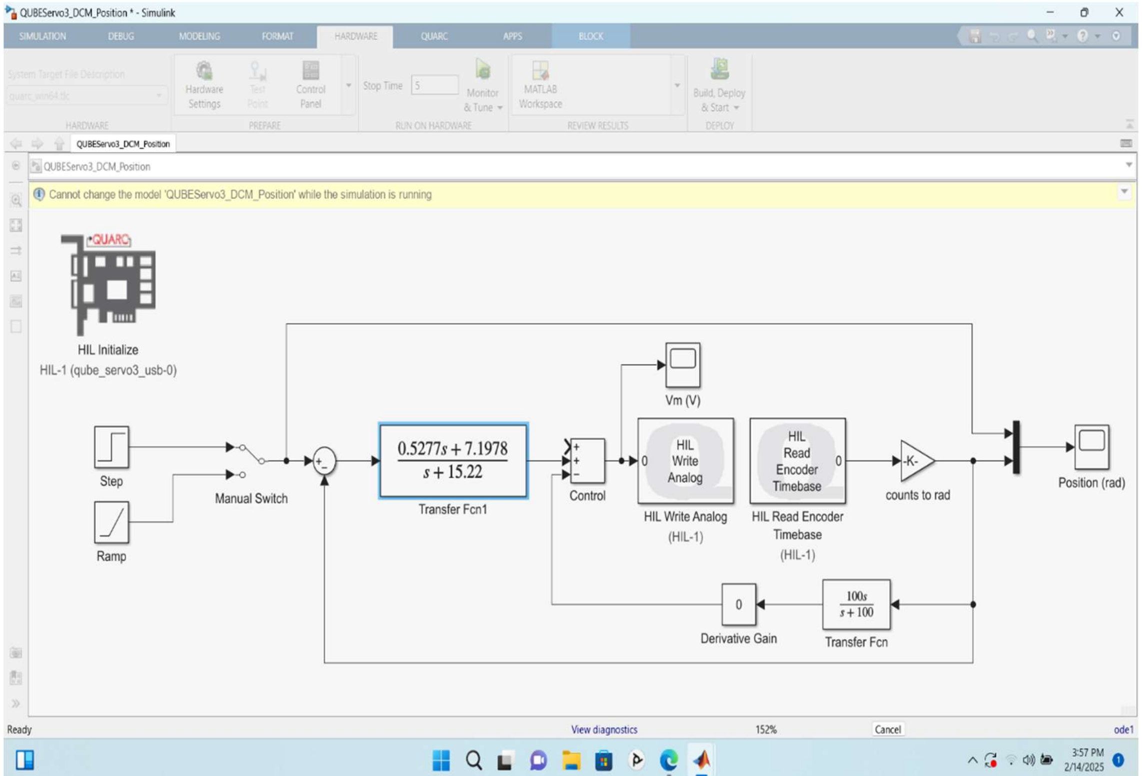


Figure 8 : Simulink for compensator

ii) Plot time response

Figure 9: Time Response

References

[1] "Mathworks," [Online]. Available: https://www.mathworks.com/help/control/index.html.

[2] "Tutorialspoint," [Online]. Available: https://www.tutorialspoint.com/control_systems/index.htm.

[3] "Control Tutorial," [Online]. Available: https://ctms.engin.umich.edu/CTMS/index.php?aux=Home.