Cover sheet for submission of work for assessment

SWINBURNE UNIVERSITY OF TECHNOLOGY

UNIT DETAILS

Unit name	ROBOTIC CONTROL			Class day/time	Wed 8:30am
Unit code	RME30003	Assignment no.	Lab 4	Due date	29/10/2025
Name of lecturer/teacher					
Tutor/marker's name					

Office use only

Faculty or school date stamp

STUDENT(S)

	Family Name(s)	Given Name(s)	Student ID Number(s)
(1)	Kularatne	Rovin	103496754
(2)	Chandrasekara	Nugith	103505197
(3)	Javapro	John	101233221
(4)			
(5)			
(6)			

DECLARATION AND STATEMENT OF AUTHORSHIP

1. I/we have not impersonated, or allowed myself/ourselves to be impersonated by any person for the purposes of this assessment.
2. This assessment is my/our original work and no part of it has been copied from any other source except where due acknowledgement is made.
3. No part of this assessment has been written for me/us by any other person except where such collaboration has been authorised by the lecturer/teacher concerned.
4. I/we have not previously submitted this work for this or any other course/unit.
5. I/we give permission for my/our assessment response to be reproduced, communicated, compared and archived for plagiarism detection, benchmarking or educational purposes.

I/we understand that:

6. Plagiarism is the presentation of the work, idea or creation of another person as though it is your own. It is a form of cheating and is a very serious academic offence that may lead to exclusion from the University. Plagiarised material can be drawn from, and presented in, written, graphic and visual form, including electronic data and oral presentations. Plagiarism occurs when the origin of the material used is not appropriately cited.

Student signature/s

I/we declare that I/we have read and understood the declaration and statement of authorship.

(1)	Rovin Kularatne	(4)	
(2)	Nugith Chandrasekara	(5)	
(3)	John Javapro	(6)	

Further information relating to the penalties for plagiarism, which range from a formal caution to expulsion from the University is contained on the Current Students website at **www.swin.edu.au/student/**

Copies of this form can be downloaded from the Student Forms web page at **www.swinburne.edu.au/studentforms/**

TABLE OF CONTENTS

1. INTRODUCTION

This lab explores digital control analysis using MATLAB and Simulink, focusing on balancing a pendulum with the pole placement method. The experiment shows how digital controllers can be designed, simulated, and applied to control unstable systems.

1.1 BACKGROUND INFORMATION

A pendulum is naturally unstable because it falls over when disturbed unless a control force is applied. In control engineering, it is often used as a classic example to test balancing and stability methods. Digital control systems use computer-based calculations to control system behaviour, while Simulink provides a graphical platform to simulate and test these systems. The pole placement method allows designers to choose specific pole locations to make the system stable and improve performance.

1.2 OBJECTIVES

- To learn how to use MATLAB functions for digital control design.
- To simulate control systems using Simulink.
- To study digital system design using the emulation method.
- To balance a pendulum using the pole placement method.

1.3 EXAMPLE

For example, if the pendulum starts to move slightly from its upright position, the controller calculates and applies the correct control signal to the motor to bring it back to balance. This shows how a properly designed digital controller can stabilize an unstable system in real time.

2. PREDICTIONS

It is expected that the digital controller designed in MATLAB will closely match the behaviour of the continuous controller when an appropriate sampling time is chosen. As the sampling time becomes smaller, the discrete system response should become smoother and more similar to the continuous response, showing less delay and overshoot. However, if the sampling time is too large, the system may become unstable or show larger oscillations. When applying the pole placement method to balance the pendulum using the QUARC model, it is predicted that the controller will stabilize the pendulum in the upright position with minimal overshoot and steady settling, demonstrating effective control through properly placed poles.

3. METHODOLOGY

Part 1: Design a Digital Controller in MATLAB

- Open SISO Tool in MATLAB and import the continuous plant.
- Record the uncompensated step response (overshoot and settling time).

- Convert the system to discrete form with a sampling time of 0.05s.
- Record the discrete step response and note the changes.
- Generate a Simulink block diagram of the discrete system and compare coefficients with calculated results.
- Convert the system back to continuous mode.
- Add the lead compensator $C(s)=4.32\frac{(s+2.08)}{(s+8.40)}$
- Record both continuous and discrete compensated step responses.
- Compare the discrete compensator coefficients with your calculations.

Part 2: Investigate Sampling Time Effects

- Record discrete step responses for sampling times: 0.01s, 0.1s, 0.3s, 0.4s.
- Compare and discuss how sampling time affects stability and response.

Part 3: Pendulum Balancing by Pole Placement Method

- Open and run Servo3_Parameters.m in MATLAB.
- Load the state-feedback gain (K) found earlier into the workspace.
- Open Pole_Placement_Servo3.slx and set HIL preference to Servo_3.
- Build and run the QUARC controller from Simulink.
- Hold the pendulum upright until the controller engages.
- Record the pendulum response from the scopes.
- Analyse the response and system behaviour.
- Stop the controller and power off the QUBE-Servo 3 when finished.

4. CALCULATIONS

This section provides a list of all the calculations done for the lab experiment. The important equations will be discussed in more detail along with explanations as to how they were utilised.

The ZOH equivalent transfer function is an important equation for finding the discrete-time transfer function, $G(z)$, of a continuous-time plant, $G_p(s)$, that is preceded by a Digital-to-Analog Converter (DAC) using a Zero-Order Hold (ZOH). The ZOH holds the digital control signal constant for one sampling period, T. This formula is used to find the plant's $G(z)$ and to find the compensator's $C(z)$.

Equation 1-ZOH-Equivalent Transfer Function

$$G(z) = Z\left\{\frac{1-e^{-sT}}{s}G_p(s)\right\} = (1-z^{-1})Z\left\{\frac{G_p(s)}{s}\right\}$$

In order to solve the $Z\left\{\frac{G_{p(s)}}{s}\right\}$ or $Z\{\frac{C(s)}{s}\}$ portion of the main ZOH equation, you must first expand the continuous-time function using partial fractions. You break it down into simple terms and then convert each term from the s-domain to the z-domain using these standard transform pairs.

Equation 2- Standard Z-Transform Pairs

$$Z\left\{\frac{1}{s}\right\} = \frac{z}{z-1}$$

$$Z\left\{\frac{1}{s^2}\right\} = \frac{Tz}{(z-1)^2}$$

$$Z\left\{\frac{1}{s+a}\right\} = \frac{z}{z-e^{-aT}}$$

Equation 3 is the standard formula for finding the total closed-loop transfer function, $M(z)$, for a system with unity feedback. $G_{OL}(z)$ represents the entire open-loop transfer function in the forward path.

Equation 3-Closed-Loop Transfer Function-

$$M(z) = \frac{G_{OL}(z)}{1 + G_{OL}(z)}$$

The input's ability to control the system's states is assessed using the controllability matrix (M) using Equation 4. Initially, you compute ($M = [B \quad AB \quad A^2B \quad A^3B]$) for your ($n = 4$) state system. If this matrix has full rank (rank=4), the system is controllable; this may be verified most simply by making sure its determinant is non-zero.

Equation 4-Controllability Matrix

$$M = [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B]$$

Equation 5 converts a time-domain performance requirement (Percent Overshoot, M_p) into a crucial system parameter, the damping ratio (ζ). It is necessary to express M_p as a decimal and this is the first step in determining where you want your dominant poles to be.

Equation 5-Damping ratio from overshoot

$$\zeta = \frac{-\ln(M_p)}{\sqrt{\pi^2 + [\ln(M_p)]^2}}$$

The following formulas relates the settling time (t_s) to the damping ratio (ζ) and natural frequency (ω_n). After finding ζ from the overshoot, this equation is used to solve for ω_n, which determines the speed of the response.

Equation 6-Natural frequency and Settling time

$$t_s = \frac{4}{\zeta\omega_n}$$

$$\omega_n = \frac{4}{\zeta t_s}$$

The standard form of the equation for a pair of complex-conjugate poles is shown below. To determine the exact s-plane locations ($p_{1,2} = -\sigma \pm j\omega_d$) that your dominant poles must have in order to satisfy the specified performance requirements, you use the ζ and ω_n values that were calculated.

Equation 7-Dominant Pole Location

$$p_{1,2} = -\zeta\omega_n \pm j\omega_{n\sqrt{1-\zeta^2}}$$

Upon applying a state-feedback control law ($u = -Kx$), the left side of Equation 8 is the state-space system's true characteristic equation. Using the designated additional poles (p_3, p_4) and your dominant poles (p_1, p_2), the desired characteristic equation is created on the right side. Four linear equations are created to solve for the four unknown gains (k_1, k_2, k_3, k_4) in the feedback matrix K. These equations are created by extending both polynomials and equating the coefficients of like powers of s.

Equation 8-Pole placement characteristic equation

$$|sI - (A - BK)| = (s - p_1)(s - p_2)(s - p_3)(s - p_4)$$

The entire mathematical foundation for the lab's initial analysis is provided by this collection of equations. These procedures successfully model a continuous-time system in the discrete z-domain and construct a state-space controller by positioning poles to satisfy certain transient performance conditions.

1) i) $K_s = 4 \quad T_s = 0.48 \quad T = 0.05$

$$G(z) = Z\left\{G_{ho}(s)G_p(s)\right\} = Z\left\{\frac{1-e^{-sT}}{s}\,\frac{K_s}{s(T_s s+1)}\right\}$$

(use z transform property)

$$G(z) = (1-z^{-1})Z\left\{\frac{G_p(s)}{s}\right\} = (1-z^{-1})\underbrace{Z\left\{\frac{K_s}{s^2(T_s s+1)}\right\}}_{G_1(z)}$$

$G_1(z)$ partial fractions

$$\frac{K_s}{s^2(T_s s+1)} = \frac{A}{s} + \frac{B}{s^2} + \frac{C}{s+(1/T_s)}$$

$$B = \lim_{s\to 0} s^2 G_1 = \frac{4/0.48}{1/0.48} = \underline{\underline{4}}$$

$$A = \lim_{s\to 0} \frac{d}{ds}(s^2 G_1) = \frac{-4/0.48}{(1/0.48)^2} = \underline{\underline{-1.92}}$$

$$C = \lim_{s\to -\frac{1}{T_s}} \left(s+\frac{1}{T_s}\right)(G_1) = \frac{4/0.48}{(-1/0.48)^2} = \underline{\underline{1.92}}$$

Final expression in partial form

$$\underline{\underline{\frac{-1.92}{s} + \frac{4}{s^2} + \frac{1.92}{s+(1/0.48)}}}$$

$$Z\left\{\frac{1}{s}\right\} = \frac{z}{z-1}$$

$$Z\left\{\frac{1}{s^2}\right\} = \frac{Tz}{(z-1)^2}$$

$$Z\left\{\frac{1}{s+a}\right\} = \frac{z}{z-e^{-aT}}$$

$$G_1(z) = -1.92 \times \frac{z}{z-1} + 4 \times \frac{0.05z}{(z-1)^2} + 1.92 \times \left(\frac{z}{z-e^{-0.05/0.48}}\right)$$

$$= -1.92\frac{z}{z-1} + \frac{0.2z}{(z-1)^2} + 1.92\frac{z}{z-0.9011}$$

$$= \frac{z(0.0101z + 0.0097)}{(z-1)^2(z-0.9011)}$$

$$G(z) = (1-z^{-1})G_1(z)$$

$$G(z) = \frac{z-1}{z}G_1(z)$$

$$= \frac{z-1}{z}\left(\frac{z(0.0101z+0.0097)}{(z-1)^2(z-0.9011)}\right)$$

$$= \frac{0.0101z + 0.0097}{(z-1)(z-0.9011)}$$

$$= \frac{0.0101z + 0.0097}{z^2 - 1.9011z + 0.9011}$$

ii) $$T(z) = \frac{G(z)}{1+G(z)}$$

$$T(z) = \frac{\left(\frac{0.0101z + 0.0097}{z^2 - 1.9011z + 0.9011}\right)}{1 + \left(\frac{0.0101z + 0.0097}{z^2 + 1.9011z + 0.9011}\right)}$$

$$T(z) = \frac{0.0101z + 0.0097}{z^2 - 1.9011z + 0.9011 + 0.0101z + 0.0097}$$

$$T(z) = \frac{0.0101z + 0.0097}{z^2 - 1.891z + 0.9108}$$

$$R(z) = \frac{z}{z-1}$$

$$Y(z) = T(z)\,R(z)$$

$$Y(z) = \frac{0.0101z + 0.0097}{z^2 - 1.891z + 0.9108} \times \frac{z}{z-1}$$

iv) $$\boxed{G_c(s) = 4.32\,\frac{(s+2.083)}{(s+8.4)}}$$

$$\frac{G_c(s)}{s} = \frac{4.32(s+2.083)}{s(s+8.4)} = \frac{A}{s} + \frac{B}{s+8.4}$$

$$A = \lim_{s\to 0} s\left[\frac{4.32(s+2.083)}{s(s+8.4)}\right] = \frac{4.32(2.083)}{8.4} = 1.0713$$

$$B = \lim_{s\to -8.4} (s+8.4) = \frac{4.32(s+2.083)}{s(s+8.4)} = 3.2487$$

$$\frac{G_c(s)}{s} = \frac{1.0713}{s} + \frac{3.2487}{s+8.4}$$

$$Z\left\{\frac{G_c(s)}{s}\right\} = 1.0713\left(\frac{z}{z-1}\right) + 3.2487\left(\frac{z}{z-e^{-8.4T}}\right)$$

$$Z\left\{\frac{G_c(s)}{s}\right\} = \frac{1.0713z}{z-1} + \frac{3.2487z}{z-0.6570}$$

$$C(z) = \frac{z-1}{z}\left(\frac{1.0713z}{z-1} + \frac{3.2487z}{z-0.6570}\right)$$

$$C(z) = 1.0713 + 3.2487 \times \frac{z-1}{z-0.6570}$$

$$C(z) = \frac{(1.0713)(z-0.6570) + 3.2487(z-1)}{z-0.6570}$$

$$C(z) = \frac{1.0713z - 0.7039 + 3.2487z - 3.2487}{z-0.6570}$$

$$C(z) = \frac{4.32z - 3.9526}{z-0.6570}$$

v) • Determine digital plant transfer function and digital compensator

- Calculate open loop " " $L(z) = C(z)\,G(z)$
- Find closed loop " " $T(z) = \frac{L(z)}{1+L(z)}$
- Define input as unit step $R(z) = \frac{z}{z+1}$
- Calculate sys output transform $Y(z) = T(z)\,R(z)$
- Determine output response $y(k)$ by taking inverse Z transform of $Y(z)$

② a) $B = \begin{bmatrix} 0 \\ 0 \\ 56.9837 \\ 56.3211 \end{bmatrix}$

$AB = \begin{bmatrix} 56.9837 \\ 56.3211 \\ -742.31414 \\ -754.83352 \end{bmatrix}$

$A^2B = \begin{bmatrix} -742.31414 \\ -754.83352 \\ 18241.67741 \\ 24737.57326 \end{bmatrix}$

$A^3B = \begin{bmatrix} 18241.67741 \\ 24737.57326 \\ -355726.67890 \\ -446983.88808 \end{bmatrix}$

$M_c = \begin{bmatrix} 0 & 56.98 & -742.34 & 18242.48 \\ 0 & 56.32 & -754.86 & 24738.05 \\ 56.98 & -742.34 & 18242.48 & -355763.5 \\ 56.32 & -754.86 & 24738.05 & -446988.1 \end{bmatrix}$

Determinant is non zero and is rank 4. ∴ system is controllable.

b) $M_p = 6.81\%$ $\quad t_s = 1.54s$

$$\xi = \frac{-\ln(M_P)}{\sqrt{\pi^2 + \ln(M_P)^2}} = \frac{-\ln(0.0681)}{\sqrt{\pi^2 + \ln(0.0681)^2}} = \frac{2.687}{\sqrt{9.870 + 7.220}} \approx 0.65$$

$$W_n = \frac{4}{\xi t_s} = \frac{4}{0.65 \times 1.54} = 4 \text{ rad/s}$$

c) $P_{1,2} = -\xi W_n \pm j W_n \sqrt{1-\xi^2}$

$P_{1,2} = -0.65 \times 4 \pm j\, 4\sqrt{1-0.65^2}$

$P_{1,2} = -2.6 \pm j3.04$

d) $(s-P_1)(s-P_2)(s-P_3)(s-P_4) = 0$

$(s-(-2.6+j3.04))(s-(-2.6-j3.04))(s+40)(s+45) = 0$

$((s+2.6)^2 + 3.04^2)(s^2+85s+1800) = 0$

$(s^2+5.2s+6.76+9.242)(s^2+85s+1800) = 0$

$(s^2+5.2s+16.002)(s^2+85s+1800) = 0$

$s^4 + 90.18s + 2256.19s^2 + 10659.85s + 28602 = 0$

$|sI - (A-Bk)| = 0$

$$A - Bk = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -56.98k_1 & 152.01-56.98k_2 & -12.53-56.98k_3 & -0.5-56.98k_4 \\ -56.32k_1 & 264.31-56.32k_2 & -12.36-56.32k_3 & -0.87-56.32k_4 \end{bmatrix}$$

Determinant of $sI - (A \cdot Bk)$

s^3 : $13.4 + 56.98k_3 + 56.32k_2 = 90.18$

s^2 : $157.1 + 705.8k_2 + 700.1k_3 + 12.39k_4 = 2256.19$

s^1 : $40081.9 + 1506k_1 + 8663k_2 + 12.53k_3 = 10659.85$

s^0 : $15063k_2 - 8663k_1 = 28602$

Solve for $k_{1,2,3,4}$ and substitute to matrix $k = [k_1 \ k_2 \ k_3 \ k_4]$

$[-3.31 \quad 0.11 \quad 0.99 \quad 0.38]$

5. PRACTICAL RESULTS

This section provides the results obtained from the practical procedures of this experiment. The first part of the lab focuses on analysing the digital controllers using MATLAB, while the second section involves balancing a pendulum using pole placement method via SIMULINK/QUARC model.

Figure 1 below shows the step response for a continuous feedback system with no compensator added. Figure 2 shows the same system without the compensator, but it is converted to a discrete system with a zero-order hold method and sampling rate of 0.05s.

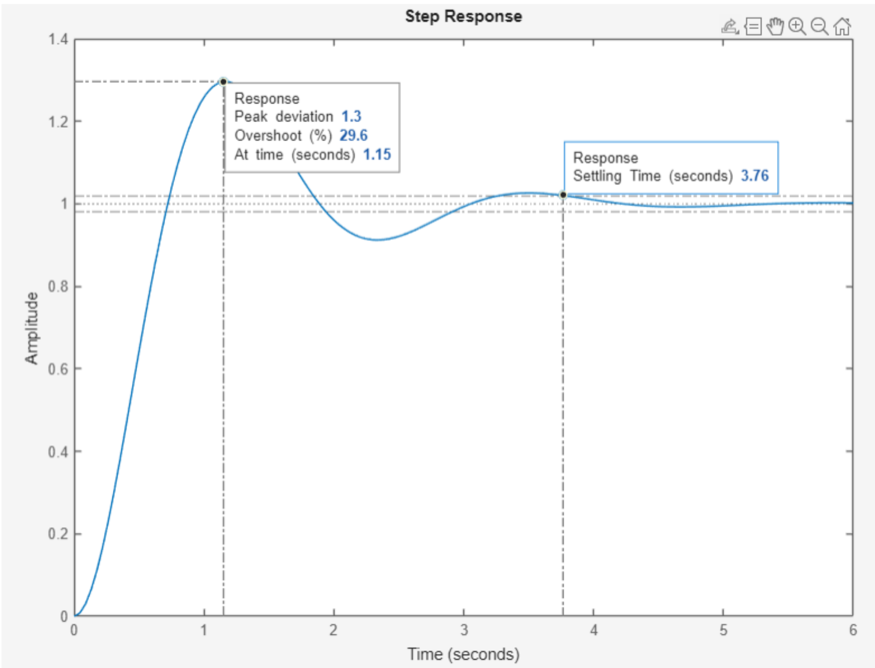

Figure 1-Continuous feedback system without C(s)

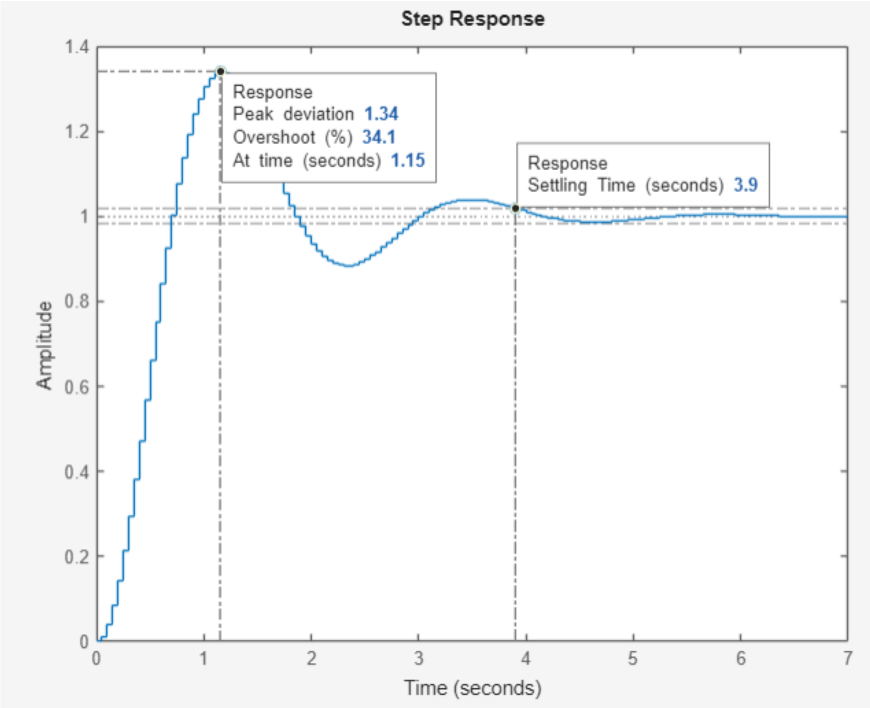

Figure 2- Discrete Continuous feedback system without C(s) with 0.05s sampling time

Now, Figure 3 shows the step response for a continuous feedback system with a lead compensator from the previous laboratory added. Figure 4 shows the same system as before, but it is converted to a discrete system with a zero-order hold method and sampling rate of 0.05s.

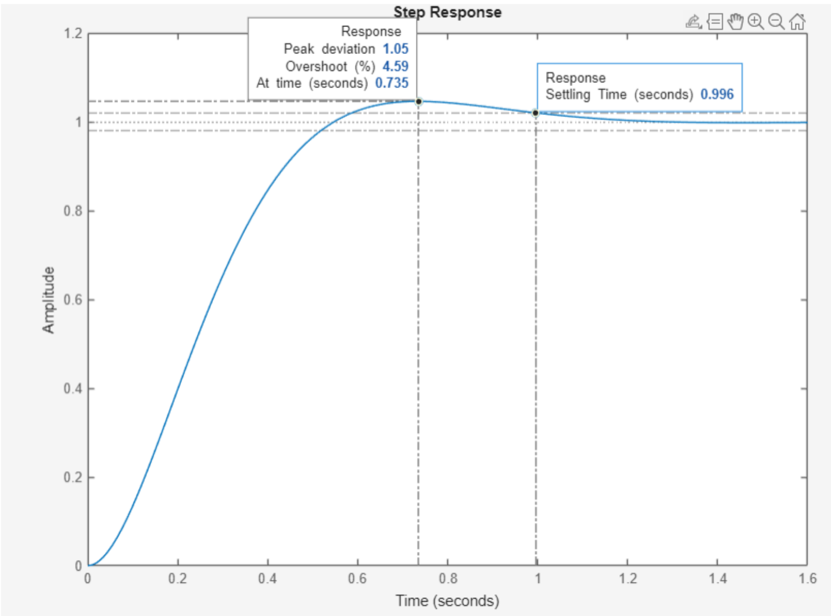

Figure 3-Continuous feedback system with C(s)

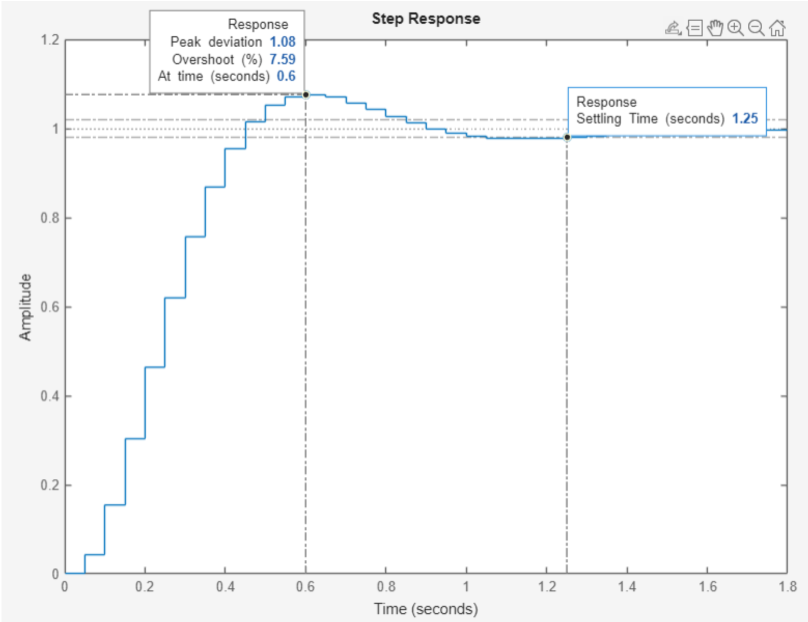

Figure 4-Discrete Continuous feedback system with C(s) with 0.05s sampling time

Figures 5 and 6 show the aforementioned compensated system, with a discrete system step response for the sampling time set to 0.01s and 0.1s respectively. As shown in these figures, the accuracy of the step response decreases as the sampling time increases. Figures 7 and 8 show the same system, but with a discrete system step response for the sampling time set to 0.3s and 0.4s respectively. Here, the sampling time is too high and results in a invalid step response for the discrete digitised system.

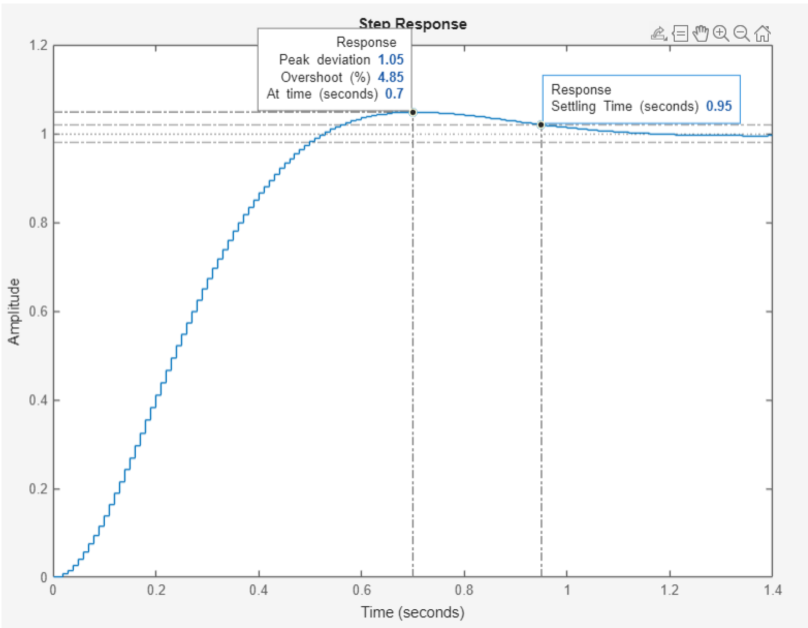

Figure 5-Discrete system step response for the sampling time equal to 0.01s

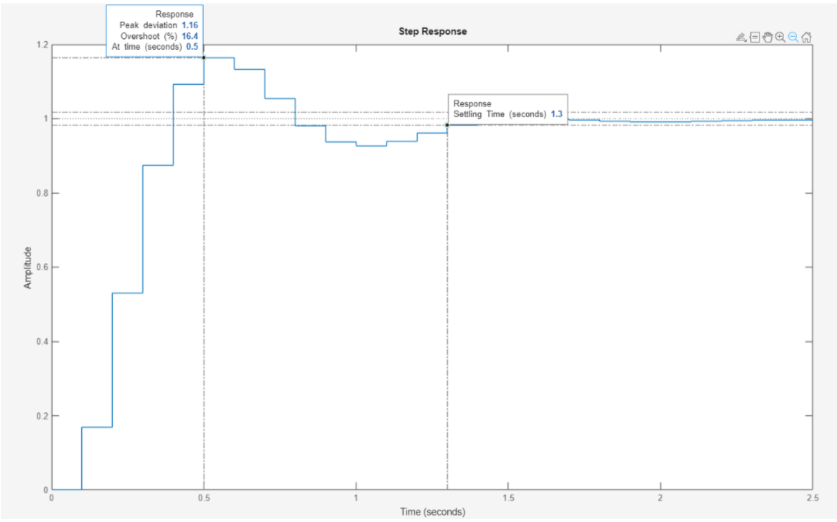

Figure 6-Discrete system step response for the sampling time equal to 0.1s

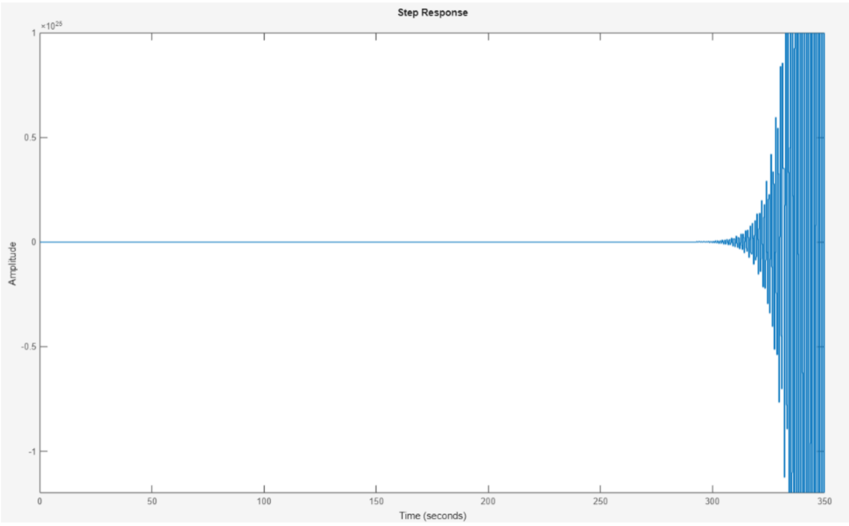

Figure 7-Discrete system step response for the sampling time equal to 0.3s

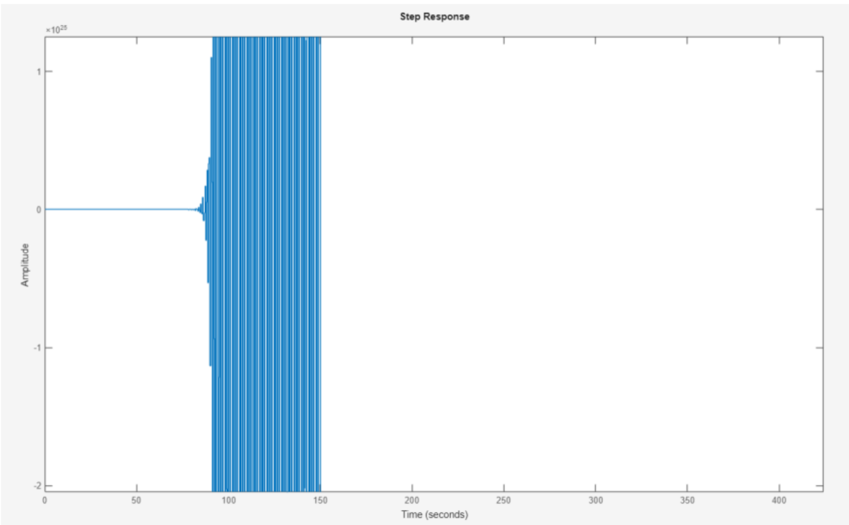

Figure 8-Discrete system step response for the sampling time equal to 0.4s

Figure 11 shows the full step response for the pendulum SIMULINK/QUARC experiment. Figure 9 shows how the V_{ss} (Steady State) was obtained (61.51), whereas Figure 10 shows how the V_{peak} was obtained (63.92). Using this information, the overshoot for the system was calculated to be approximately 3.92%.

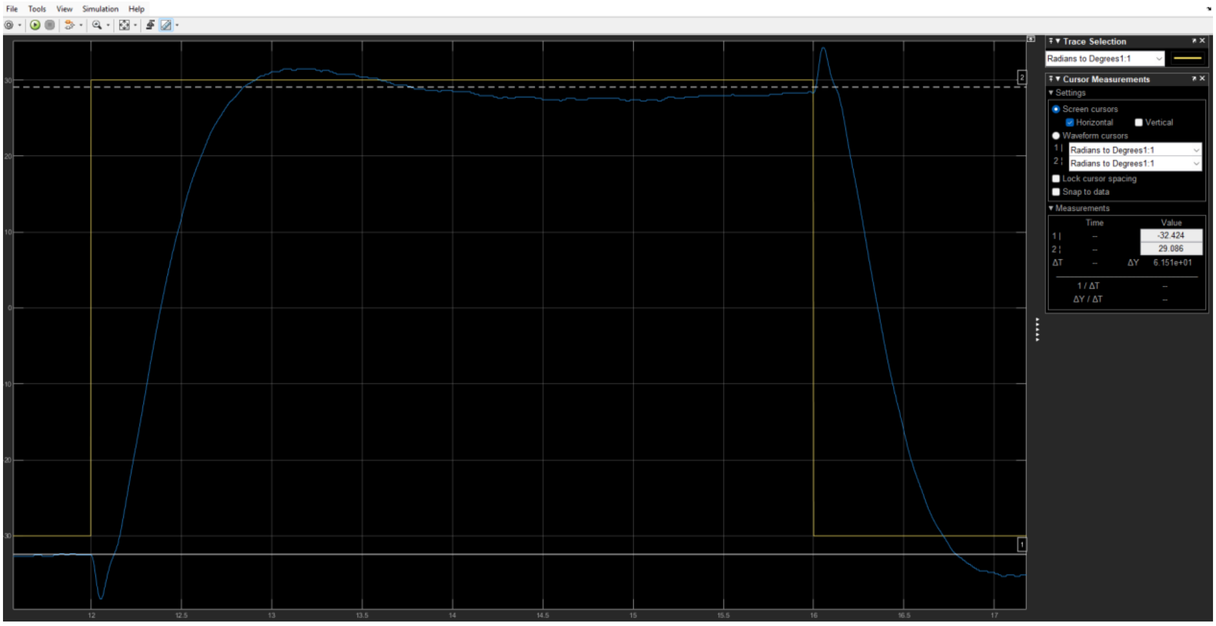

Figure 9-Steady state obtained from the system

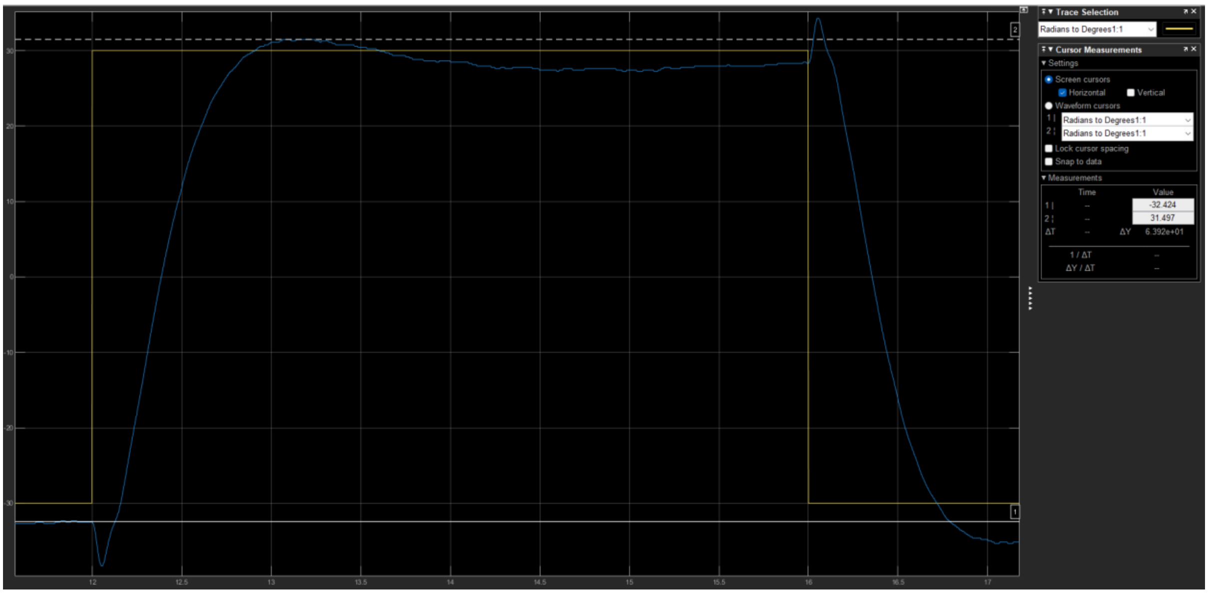

Figure 10-V peak obtained from the system

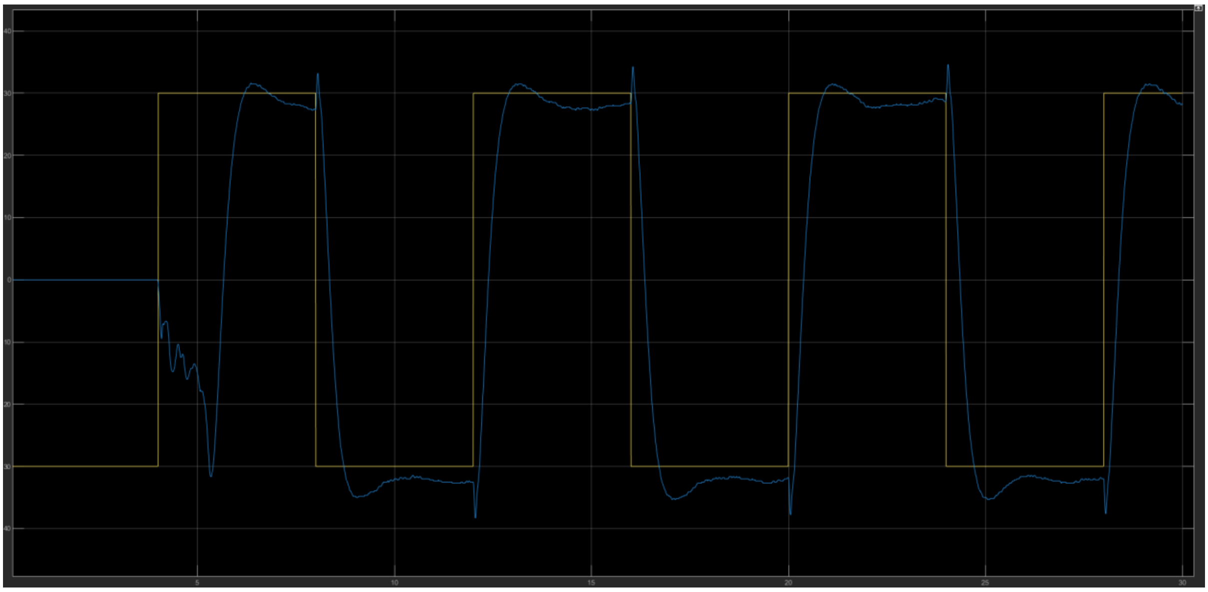

Figure 11-Section of the step response of the pendulum balancing system

6. ANALYSIS AND DISCUSSION

This section will provide a more in-depth analysis of the results obtained from the experiments conducted in this lab with MATLAB in the primary part of the lab, and SIMULINK/QUARC for the second part of the lab.

The experimental findings illustrate important digital control concepts. Because of the intrinsic delay of the Zero-Order Hold, digitisation introduces slight increases in overshoot and settling time, as seen by a comparison of the continuous (as shown in Figure 1 and Figure 3) and discrete (as shown in Figure 2 and Figure 4) responses with a sampling time T=0.05s. The efficiency of the lead compensator is confirmed even after digitisation, as the discrete compensated system shown in Figure 4 still performs noticeably better than the uncompensated system shown in Figure 2, attaining much faster settling and smaller overshoot.

The investigation into sampling time also emphasizes its importance. The output for T=0.01s (Figure 5) closely mirrors the continuous compensated response (Figure 3), confirming that when the sampling time is smaller the approximation is much better. However, when T rises to 0.1s (Figure 6), the output worsens, characterized by larger overshoots and oscillations. Still, for T=0.3s (Figure 7) and T=0.4s (Figure 8), the output is unstable, exhibiting greater oscillations, since sampling periods beyond a threshold lead to instability in discrete system characteristics.

The pendulum balance experiment has successfully validated the pole placement technique. The controller, using the calculated feedback gain K, controlled the inherently unstable pendulum to remain upright. The recorded response (Figure 11) shows the system tracking the reference square wave, and the measured overshoot (approximately 3.92%) indicates good transient performance achieved by placing the closed-loop poles at the desired locations.

7. CONCLUSION

This experiment successfully met its objectives. Familiarity with MATLAB functions for digital control was gained through the digitization of continuous systems and compensators using the ZOH emulation method. Simulink was effectively used to simulate these systems and observe the impact of varying sampling times on performance and stability, demonstrating that sufficiently small sampling periods are crucial for accurate discrete representation and system stability. Finally, the pole placement method was successfully applied using the SIMULINK/QUARC model to design and implement a state-feedback controller, which effectively balanced the physical QUBE-Servo 3 pendulum. The practical results validated the theoretical calculations and demonstrated the effectiveness of digital control techniques for stabilizing unstable systems and achieving desired performance specifications.