MECHATRONICS CONTROL AND AUTOMATION LAB (MCTA 3104)

EXPERIMENT 3:

MODELLING OF FIRST-ORDER SYSTEMS

USING MATLAB/SIMULINK

(4.5)

INDIVIDUAL REPORT

PROGRAMME: MECHATRONICS ENGINEERING

NUR AYU NAZIERA BINTI ROSLI	2119202
DATE OF EXPERIMENT	28 OCTOBER 2024
DATE OF SUBMISSION	4 NOVEMBER 2024

Contents

Introduction

In this experiment, the modeling and analysis of first-order systems are conducted to deepen the understanding of fundamental control strategies applied in automation. Control systems play a vital role in engineering, particularly in mechatronics, where the goal is to ensure systems operate accurately, efficiently, and within desired parameters. By examining the effects of various control strategies such as Proportional (P), Proportional-Integral (PI), and Proportional-Integral-Derivative (PID), this experiment aims to demonstrate how changes in controller settings impact system behaviour.

The experiment begins with a simple proportional controller setup, analyzing how adjustments to the proportional gain Kp influence system responsiveness, overshoot, and steady-state error. As the experiment progresses, integral control Ki is introduced to observe how accumulated error correction can eliminate steady-state discrepancies, albeit with a potential trade-off of increased oscillatory behaviour if improperly tuned. Finally, derivative control Kd is added to assess its effect in predicting error trends and enhancing system stability by dampening oscillations and reducing overshoot.

By systematically recording and evaluating the step responses under different control configurations, this experiment provides insights into the practical applications and limitations of each control type. The outcomes of this analysis are crucial for engineers seeking to design or refine automated systems that require precise and reliable performance. This hands-on approach not only reinforces theoretical knowledge but also illustrates the importance of tuning control parameters to achieve a balance between speed, accuracy, and stability in real-world applications.

Diagram of The Experiment

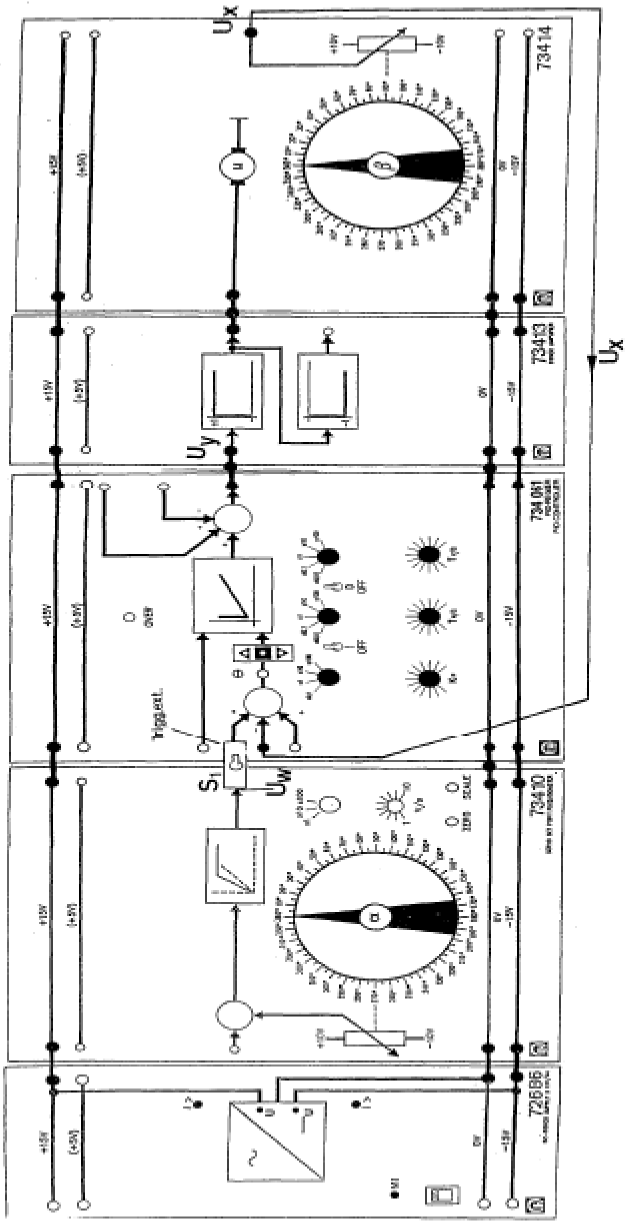


Figure 3: Diagram of Experiment

Experimental Procedure

1. Assemble in accordance with the diagram (Fig.
2. Adjust the setpoint integrator to
3. Adjust the setpoint value to =0
4. Adjust the setpoint value to =90
5. Connect the output of the set point potentiometer directly to the input of the DC
6. amplifier (PID-controller inputs and outputs are disconnected).

7. What does control mean?

Control refers to the process of manipulating the input of a system to achieve the desired output. It involves maintaining the stability and performance of a system within certain acceptable limits by adjusting parameters based on feedback.

P Controller

1. Make the connections with the PID controller block added and adjust Kp=1.0;

2. Record the values for Table 1.

Kp	2	3	4	5	10	100	1000
	95˚	93˚	92˚	103˚	100˚	102˚	100˚

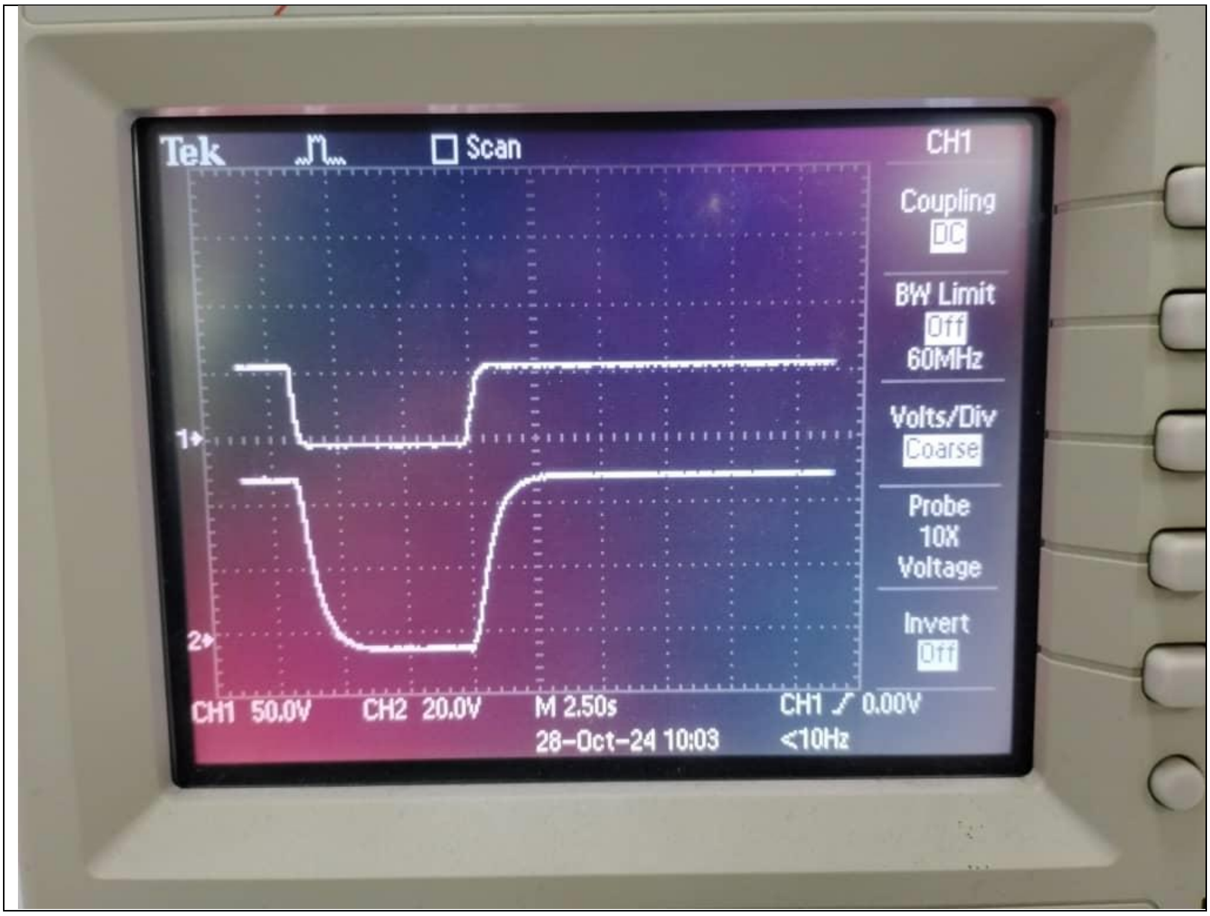
3. Start with Kp=1*1, 2*1, 3*1. (Bottom gain adjuster is a multiplier)

4. Then adjust the potentiometer to Kp=5*1 and Kp=1

5. Subsequently, set Kp=10*100. Explain the results!

As the proportional gain Kp increases, the system's response becomes more aggressive, which typically leads to faster response times. However, excessively high Kp values can lead to system instability, oscillations, or overshoot. For example, low Kp values result in a slower, more stable response but may not reach the setpoint effectively. Thus, a high Kp values like 100 or 1000 can cause significant overshoot or even make the system oscillate without settling.

6. How does the control performance?



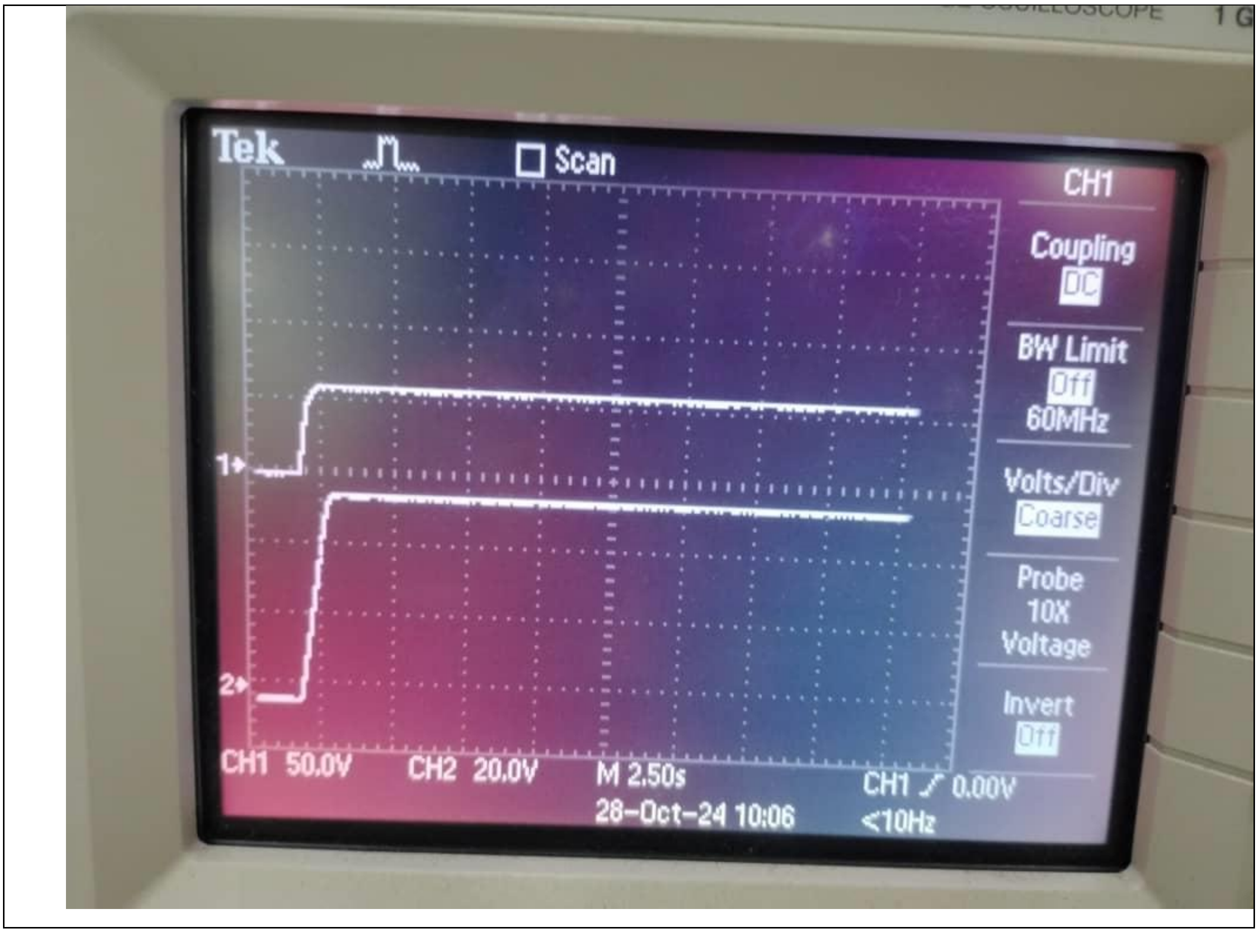
PI Controller

1. Adjust Kp = 1.0; Ki=off; Kd=off. S1 = off and the adjusting disc of the servo to = 0.

2. Connect S1 to "on". Now set Ki=0.5*1 and turn the I_{off} switch up.

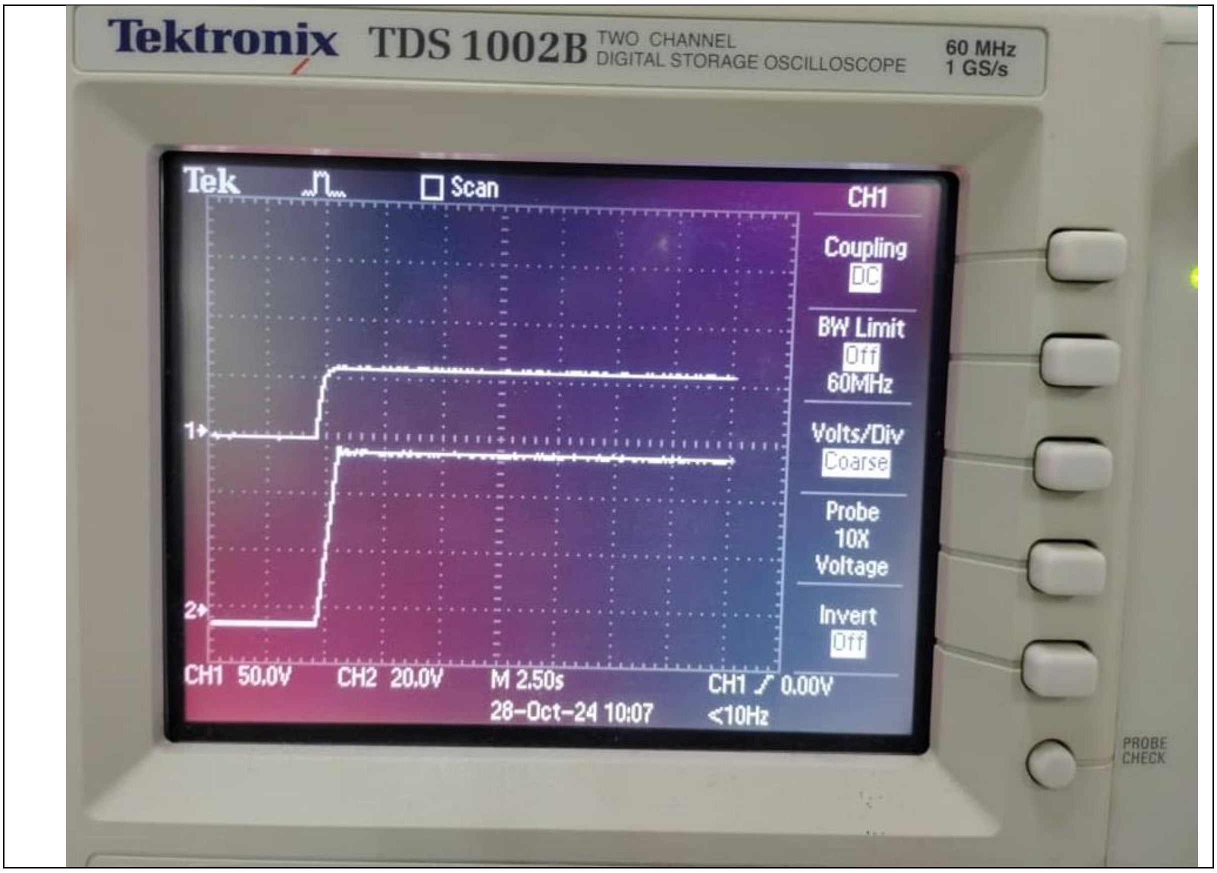
3. Record the step response of the control loop with the help of the oscilloscope, explain how the integral control operates?

The integral part with Ki of a PI Controller accumulates the error over time and adjusts the output to eliminate steady-state errors. This means it helps the system reach the setpoint more precisely by considering past errors and correcting them, reducing residual error.



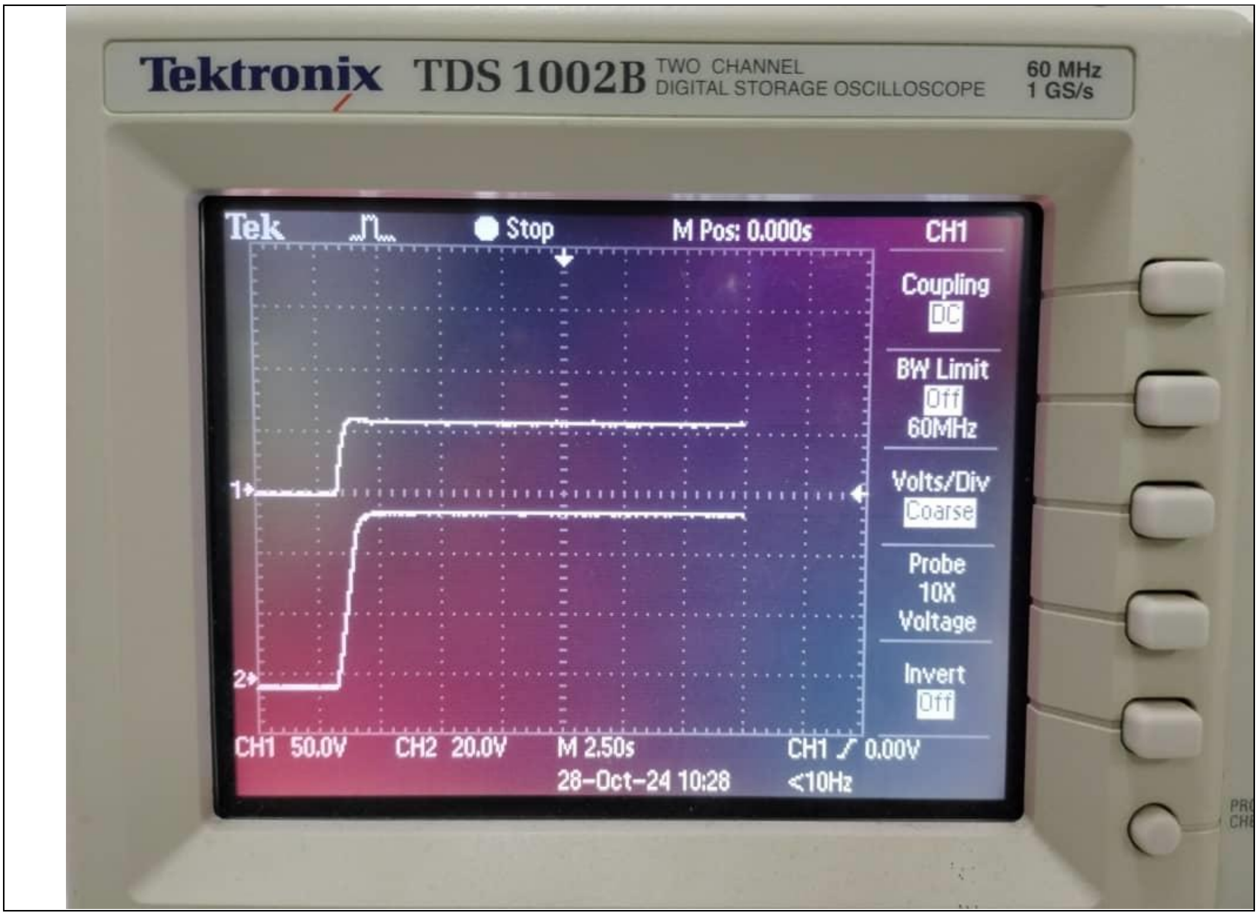
4. Make the following adjustments at the controller: Kp=200, Ki=0.5.

5. Record the step response with the oscilloscope.



6. Make the following adjustments (PID Controller): Kp=200, Ki=0.5, Kd= 0.05.

7. Record the step response.



8. Do the same with the following adjustments: Kp=20, Ki=50, Kd=0.2.

9. How can the overshoot be reduced?

To reduce overshoot, there are three ways. First is to decrease Kp which will reduce the aggressive response. Second is increase Kd to add more damping. Lastly, fine-tune the Ki to avoid excessive correction and oscillation.

Conclusion

Based on the data and results observed in this experiment, it is evident that different control configurations significantly impact system behaviour. The P Controller provided a quick response but introduced steady-state errors and potential overshoot as Kp increased. The PI Controller improved accuracy by eliminating steady-state errors through integral action, although it introduced a risk of oscillations if not properly tuned. The PID Controller, with proportional, integral, and derivative actions, delivered the most balanced performance, minimizing both overshoot and settling time by predicting and adjusting to changes in the error trend. This combination of controls demonstrated that fine-tuning the Kp, Ki, and Kd parameters is crucial for achieving optimal system stability and performance.