Introduction to Control Theory

Second Lab Report

Authors	: Alperen ISIK, Hasan Emre AYDIN
Promotion	: EENG 3
Cursus	: EENG
Date	: 09/06/2024

A. Table of Contents

B. Introduction

This laboratory experiment investigates the effects of Proportional (P), Proportional-Integral (PI), and Proportional-Integral-Derivative (PID) controllers on DC motor speed regulation. Utilizing a Data Acquisition Card (DAQ) within a MATLAB and Simulink environment, the study aims to analyse how each controller influences the motor's dynamic responses including overshoot, settling time, and steady-state error.

The setup involves real-time adjustments and monitoring of the motor speed, providing a practical understanding of various control strategies. This report will detail the experimental procedure, present the collected data, and discuss the performance outcomes of different controllers. The hypothesis is that the PID controller, with its combined properties, will offer superior control by optimizing response characteristics.

A. Elements of the considered device:

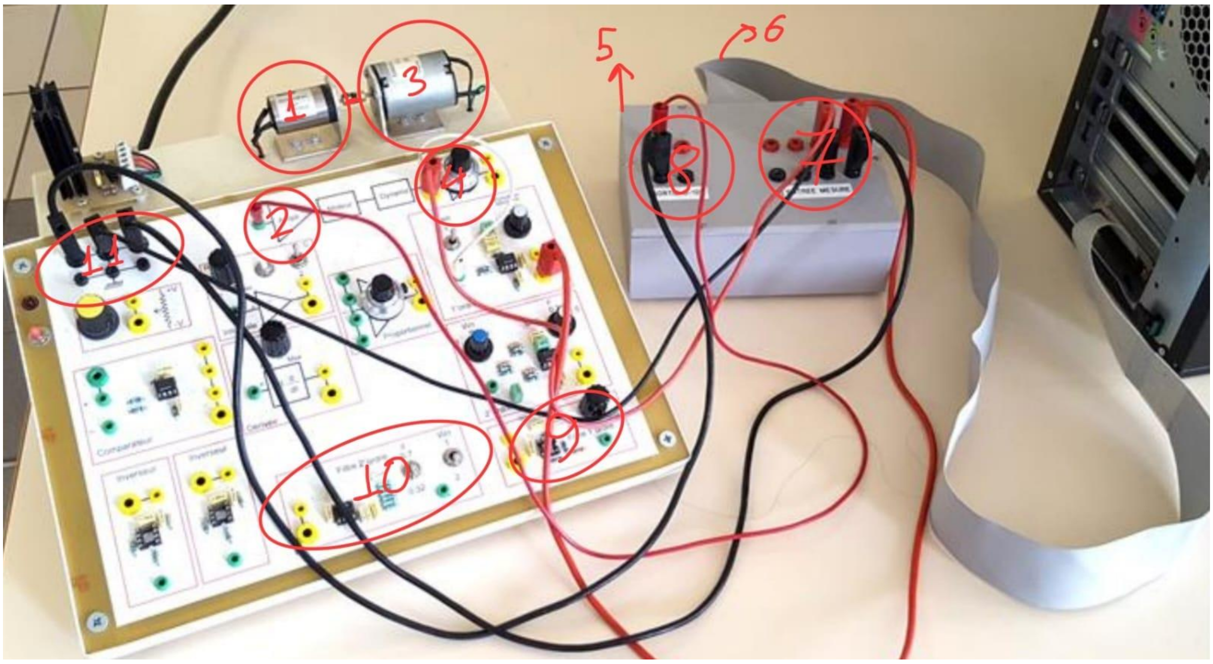


Fig. 1. Experimental device with components

Number	Device name
1	Motor
2	Motor input
3	Speedometer
4	Speedometer output
5	Data acquisition card (DAQ)
6	Card terminal block
7	Card input
8	Card output
9	1rst order filter
10	2^{nd} order filter
11	Ground

Table 1: The LAB process components

B. Functional block diagram

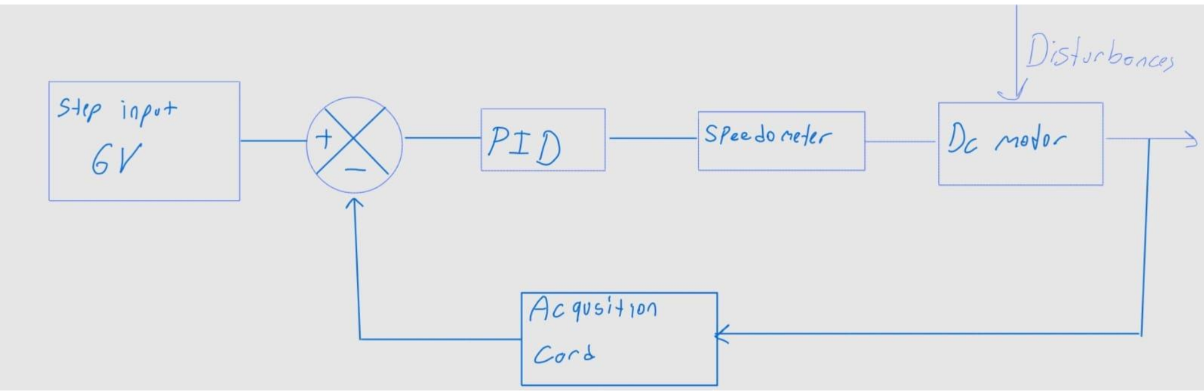


Fig. 2. Sketch of functional block diagram

C.

Q3)

3.1) The Data Acquisition (DAQ) Card is an essential component that measures voltage signals and facilitates the transfer of this data to the computer. It is capable of converting both analogue to digital signals and vice versa. Specifically, it processes the voltage output from a speedometer, typically used in measuring the speed of a DC motor. This input is crucial as it represents the speed data as an analogue voltage signal.

3.2) To ensure accurate and reliable measurements, the input voltage range of the DAQ Card must be carefully regulated. Exceeding the allowable voltage range can lead to potential damage to the card or result in inaccurate readings due to signal interference. It is therefore essential to maintain the voltage within a specified limit to prevent these issues.

3.3) The DAQ Card operates with a sample time of 0.001 seconds, which translates to taking 1000 samples per second. This high sampling rate allows for rapid response and the ability to generate precise data points. This capability is vital for tracing detailed graphs and conducting thorough analysis, ensuring that the fast processes being monitored do not compromise the integrity of the data or the functionality of the DAQ Card.

Q4)

4.1) The Data Acquisition (DAQ) Card is pivotal in both measuring and controlling voltage signals related to motor operations. It receives input voltage signals and converts them, enabling control and visualization through a computer system. The DAQ Card's outputs, which add voltage to the motor's input, are crucial for fine-tuning motor speed, which is directly controlled by these voltage adjustments.

4.2) The output of the DAQ Card is presented in an analogy format, mirroring its input characteristics. This ensures that the representation of the signal is accurate, minimizing the risk of damage to the card by adhering to specified voltage limits. These limits are essential to prevent exceeding the card's capacity, which could otherwise result in inaccurate outputs or potential harm to the DAQ Card.

4.3) With a sample time set at 0.001 seconds (or 1000 samples per second), the DAQ Card offers rapid response capabilities that are critical for managing the fast-paced processes of motor control. This high sampling rate allows for precise data collection and fine control over the motor, enhancing both the analysis and adjustment of motor speed.

To visualize and manage these processes effectively, a block diagram was created using Simulink. This tool helps in understanding and manipulating the signal flow and interactions within the system, ensuring that the DAQ Card operates efficiently and within its safe operational parameters.

C. Question 5-6

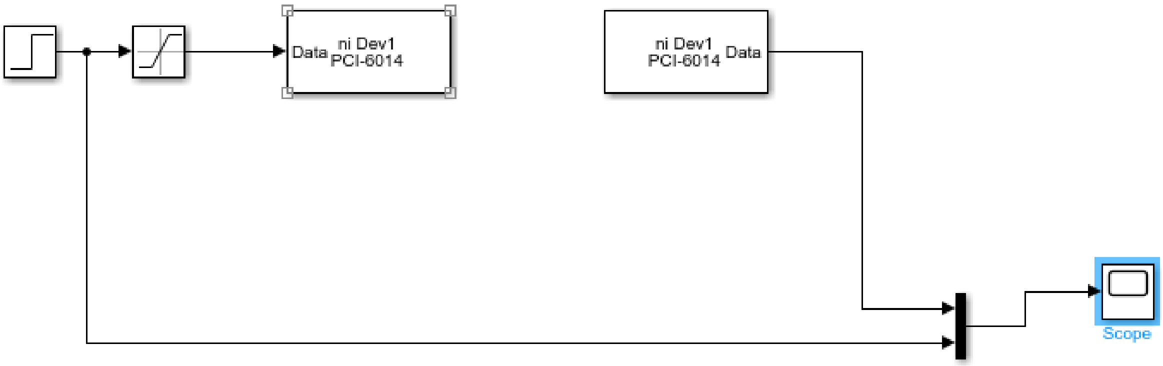


Fig. 3. Simulink design to test the step response of DC Motor

With following parameters:

- Stop time = 3s
- Solver options "Fixed-step"
- Fixed-step size = 0.001s (under "Solver details tab")
- Solver "auto"

Simulation result:

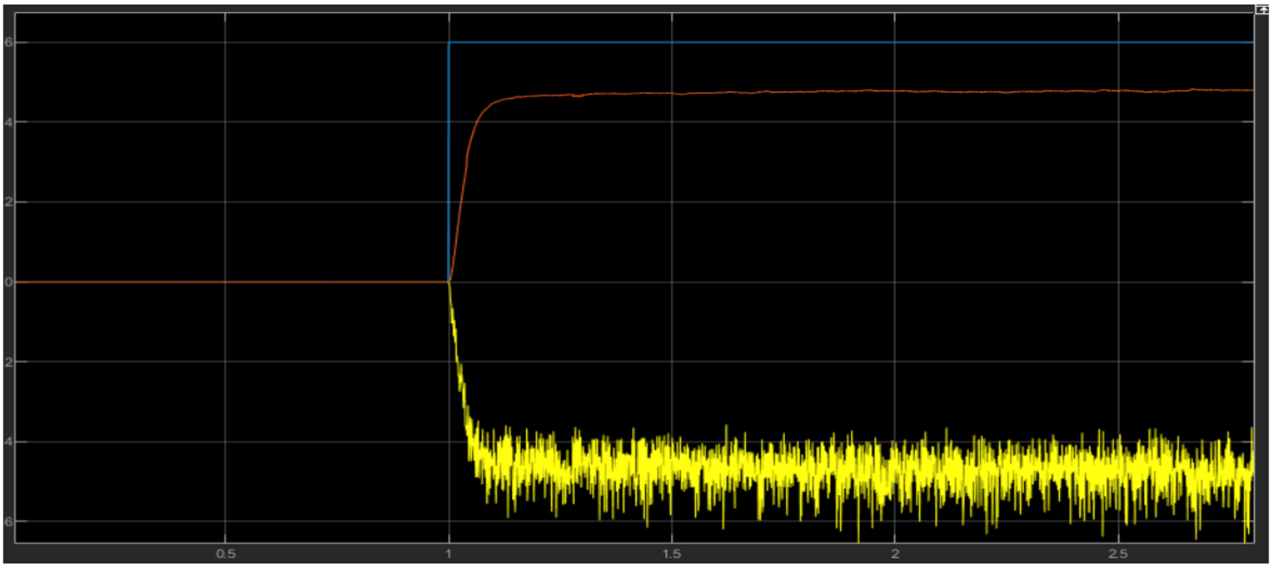


Fig. 4. the unfiltered measure signal & the filtered measure signal vs the input reference

Since question6 including question 5's sketch we added just figure 4.

5.2) An open-loop system is considered stable if it produces a bounded output in response to a bounded input. Despite the output in your graph eventually reaching a steady value, the significant and sustained oscillations indicate instability. In open-loop configurations, the absence of feedback control means there's no mechanism to correct deviations, leading to instability. This system's oscillatory behaviour and lack of damping, evident from the step input response, suggest it may not be suitable for applications requiring precise control and stability.

6.2) In the graph, the blue line represents the reference signal, while the red line indicates the filtered signal, and the yellow line depicts the unfiltered signal. Notably, the blue line consistently remains at a higher level compared to the red line, suggesting that the filtered signal does not fully reach the desired set point indicated by the reference. This discrepancy might be due to a delay or amplitude reduction caused by the filtering process or inherent system characteristics. Meanwhile, the yellow line shows significant fluctuations and much greater volatility compared to the relatively smoother red line, highlighting a high level of noise or interference in the system. This erratic behaviour of the unfiltered signal underlines the effectiveness of the filtering process in mitigating noise, though it also suggests that further tuning of the system or filter might be necessary to ensure the filtered signal adequately matches the reference.

D. Question 7

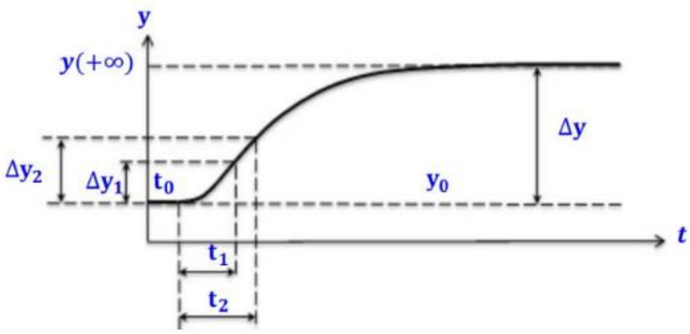
An open-loop system is characterized by a direct and fixed relationship between input and output, where the output is not automatically adjusted based on fluctuations or external disturbances. These systems typically employ a control actuator to directly manage the process without the utilization of feedback loops. This straightforward configuration is beneficial for understanding system dynamics and implementing control strategies without the complexity of feedback mechanisms.

The primary purpose of modelling an open-loop system is to analyse its performance characteristics and to develop a suitable transfer function. The provided model diagram illustrates a typical open-loop setup where the relationship between input and output is modelled to approximate real-world behaviour using mathematical representations.

For a precise understanding and effective control, specific parameters of the system are crucial, namely the steady-state gain (K), the time constant (τ), and the pure time delay (d). These parameters are integral to defining the system's transfer function, which in this context is given by:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{K\,e^{-d\,s}}{1+\tau\,s}$$

Transfer function of the model



Parameters for the Broida method

Variable	Physical meaning
y	The measured value
$U(s)$	The step input
$K = \frac{\Delta y}{\Delta u}$	**The process gain**
$t_1 = t(\Delta y_1) = t(28\%\ \Delta y)$	The time that corresponds to 28% Δy
$t_2 = t(\Delta y_2) = t(40\%\ \Delta y)$	The time that corresponds to 40% Δy
$\tau = 5.5\ (t_2 - t_1)$	**The time constant**
$d = 2.8\ t_1 - 1.8\ t_2$	**The time delay**

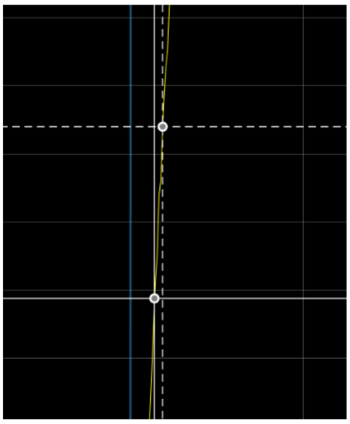
Where:

- *K*K is the steady-state gain calculated as

$$K = \frac{y(\infty)}{u} = \frac{y(\infty)}{6}$$

- *T* is the time constant derived from the response times *t1 and* t2
- *d* is the pure time delay affected by the response dynamics.

The identification and calculation of these parameters are achieved through careful analysis of the system's response to test inputs. Here's how the parameters are specifically calculated:



We position the vertical lines at t11 and t22 where the yellow line intersects the horizontal lines. Consequently, we obtain the following values for t11 and t22.

$$K = \frac{y(\infty) - y(0)}{u(\infty) - u(0)} = \frac{y(\infty)}{u(\infty)} = \quad \begin{matrix} t_1 = 1.02 - 1(\text{time delay}) = \\ t_2 = 1.03 - 1(\text{time delay}) = \end{matrix}$$

3. **Time Constant (T) and Pure Time Delay (d):**

   - T is derived as $T = 5.5 \times (t_1 - t_2) = 5.5 \times (0.02 - 0.03) = -0.055$ seconds
   - d is calculated from $d = 2.8 \times t_1 - 1.8 \times t_2 = 2.8 \times 0.02 - 1.8 \times 0.03 = 0.056 - 0.054 = 0.002$ seconds

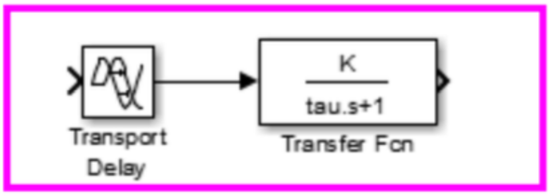
You can see our calculations with MATLAB codes here:

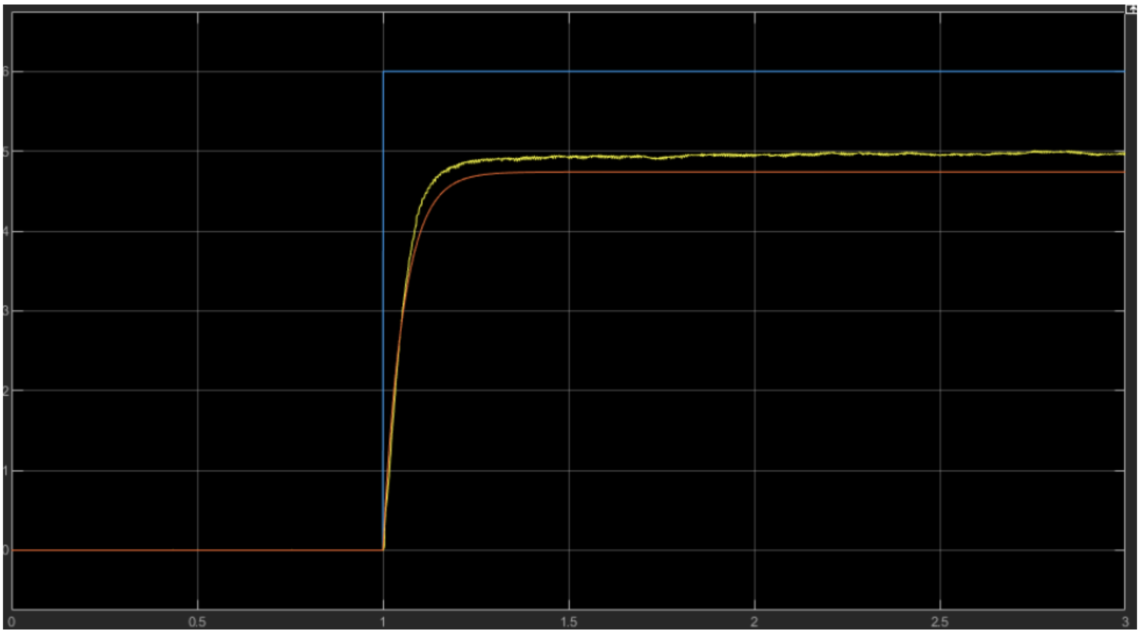
```
u=6
y_f=5.312
k= y_f/u
y_1=0.28*y_f
y_2=0.4*y_f
t1=1.011-1
t2=1.014-1
tau=5.5*(t2-t1)
d=2.8*t1-1.8*t2
dopt=0.002
tauopt=0.022
k_op=k

kp=(100*tauopt)/(286*k_op*dopt)
Ti=1.2*tauopt

kc=(100*tauopt)/(4000*k_op*dopt)
td=0.5*d
```

This model allows the simulation of system behaviour under various scenarios without risking the actual equipment, providing a valuable tool for predicting outcomes following modifications or for system tuning before real-world implementation. Validation of this model involves comparing the output of the real system against the model's output to ensure that they are sufficiently similar, confirming the model's accuracy. This rigorous validation is essential to ensure that the model faithfully represents the dynamics of the physical system it simulates.

E. Question 8

We created a new block diagram using the previously determined model. For defining the Simulink block G(s), we utilized:





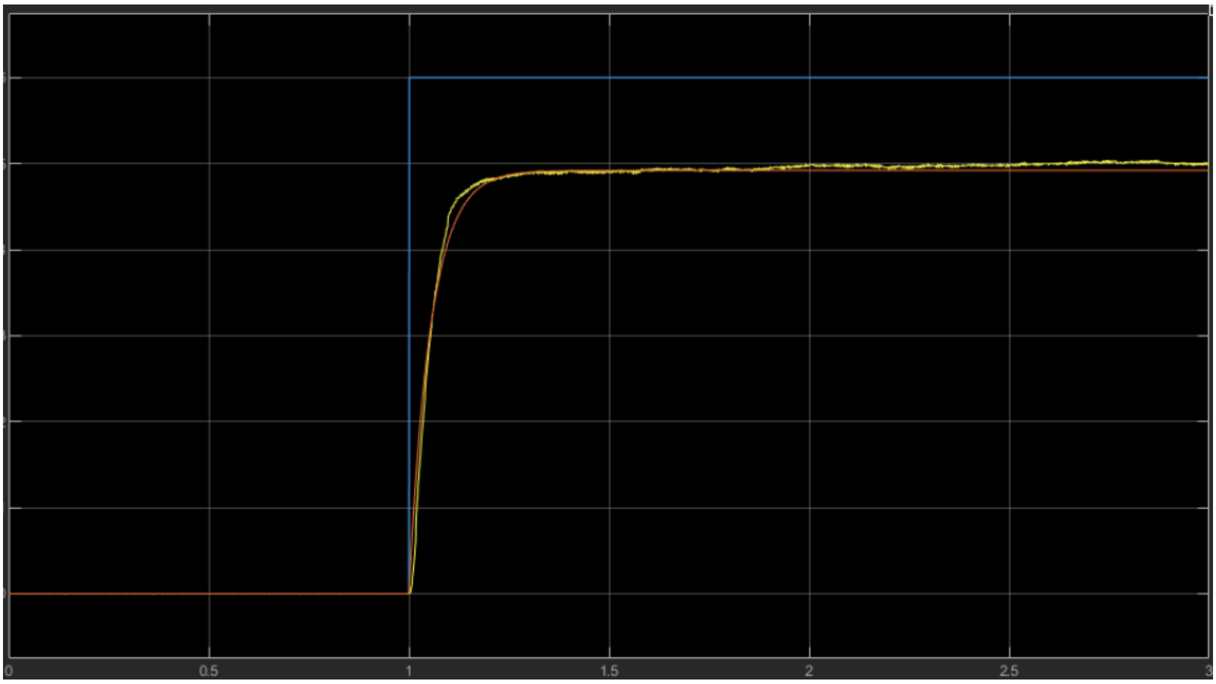
In evaluating the DC motor system using the Broida model and real system configurations, both are integrated within a unitary feedback loop aimed at controlling motor speed to reach a target value. The performance and accuracy of the model are validated through the comparison of experimental and simulated outputs, where the red and yellow curves, representing the real system and the model respectively, demonstrate a high degree of superposition.

To achieve this, a controller block, which could be a Proportional (P), Proportional-Integral (PI), or Proportional-Integral-Derivative (PID), is employed in both setups. The parameters for these controllers are empirically tuned to optimize system response. The initial setup in the Simulink model uses a transfer function (G(s)), which initially exhibited a slight discrepancy, noted as a delta of around 0.2V between the experimental curve (yellow) and the model (orange). By adjusting the gain (K) from 0.79 to 0.82, the curves' alignment improved, indicating an optimal transfer function configuration.

This process of empirical tuning and curve fitting ensures that both the theoretical model and the actual system perform closely, allowing for reliable control over the motor speed with the adjusted parameters providing the necessary corrections to match the simulated outcomes with the experimental data. This methodical approach confirms the validity of the Broida model in real-world applications, establishing a robust framework for further fine-tuning and

$$G_{Broida\ optimal}(s) = \frac{Y(s)}{U(s)} =$$

performance evaluation.



PB=100/Kc

Proposer	Type*1	Control mode	PB	Optimum setting*2 T_I	T_D	Optimum condition*3
Ziegler Nichols (1942)	A,B	P	$100\ K_pL/T$	—	—	25 % damping
		PI	$110\ K_pL/T$	$3.3\ L$	—	
		PID	$83\ K_pL/T$	$2\ L$	$0.5\ L$	
Takahashi	B	P	$110\ K_pL/T$	—	—	Minimum control area
		PI	$110\ K_pL/T$	$3.3\ L$	—	
		PID	$77\ K_pL/T$	$2.2\ L$	$0.45\ L$	
Chien Hrones Reswick	A	P	$333\ K_pL/T$	—	—	No overshoot and minimum response time
		PI	$286\ K_pL/T$	$1.2\ T$	—	
		PID	$167\ K_pL/T$	T	$0.5\ L$	
Chien Hrones Reswick	A	P	$143\ K_pL/T$	—	—	20 % overshoot and minimum response time
		PI	$167\ K_pL/T$	T	—	
		PID	$105\ K_pL/T$	$1.35\ T$	$0.47\ L$	
Chien Hrones Reswick	B	P	$333\ K_pL/T$	—	—	No overshoot and minimum response time
		PI	$167\ K_pL/T$	$4\ L$	—	
		PID	$105\ K_pL/T$	$2.4\ L$	$0.4\ L$	
Chien Hrones Reswick	B	P	$143\ K_pL/T$	—	—	20 % overshoot and minimum response time
		PI	$143\ K_pL/T$	$2.3\ L$	—	
		PID	$83\ K_pL/T$	$2\ L$	$0.42\ L$	
Fujii Yoshikawa	A	P	$100\ K_pL/T$	—	—	Minimum control area
		PI $L/T \le 1$	$167\ K_pL/(T+L)$	$T+L$	—	
		PI $L/T \ge 1$	$250\ K_pL/(T+2\ L)$	$2\ L$	—	
		PID $L/T \le 1$	$133\ K_pL/(T+(1/3)L)$	$0.5(T+L)$	$0.125(T+L)$	
		PID $L/T \ge 1$	$200\ K_pL/(T+L)$	L	$0.25\ L$	

(Notes) *1 Type A: for setpoint change; Type B: for disturbance.
*2 T, L, K_p: obtained from transient response.

F. Question 9 - Control

9.2)

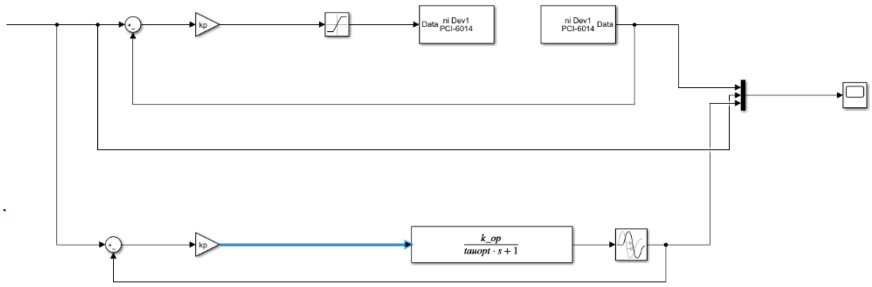


Fig.7: Simulation sketch with Kc

In figure 7 we added a proportional controller and results shown below:

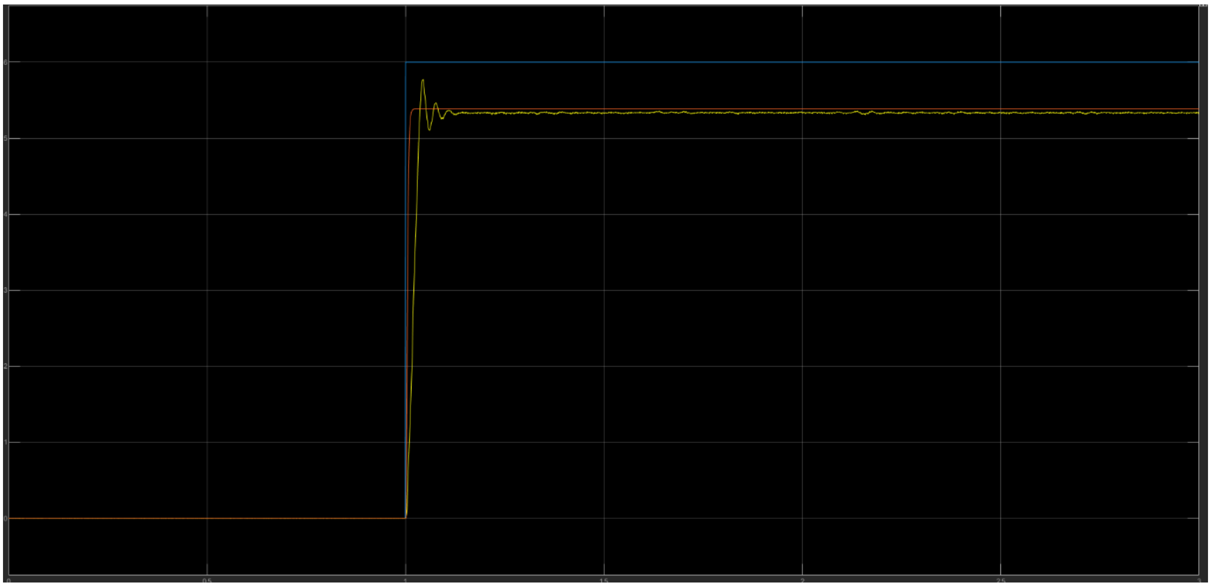


Fig. 8. Simulation graph with Kc

In the provided graph, the blue line represents the step input, while the yellow line depicts the step response of the system with a proportional controller (Kc) added for unitary feedback.

Impact of the Proportional Controller:

1. Immediate Response: The yellow line shows a rapid rise in response to the step input, indicative of the direct influence of the proportional control. This rapid response is a desirable attribute in control systems where quick system responsiveness is critical.

2. Overshoot: After the initial spike, the system overshoots the desired set point (as indicated by the blue line). This overshoot is a common characteristic in proportional control systems, particularly when high gain values are used. It demonstrates the controller's aggressiveness in attempting to reach the set point.

3. Settling and Stability: Post-overshoot, the yellow line exhibits slight oscillations before settling close to the step input level. The minor oscillations around the set point suggest that

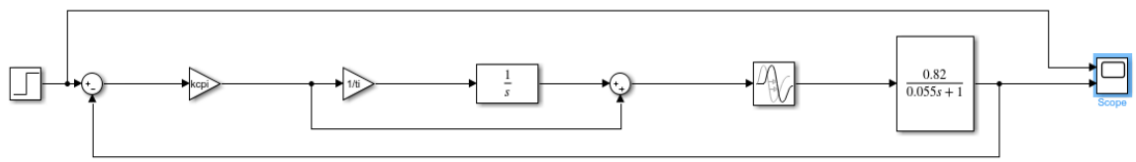
while the controller stabilizes the output, it does so with minor fluctuations, which are typical in systems without damping influence from integral or derivative actions.

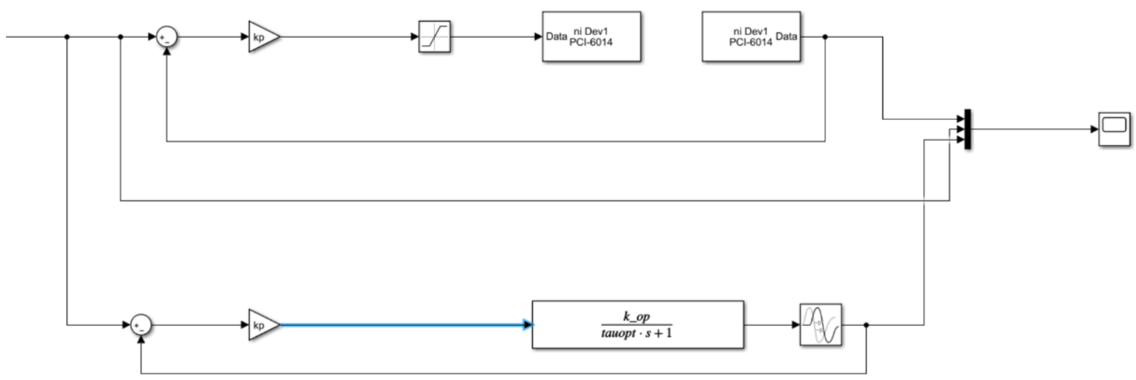
The proportional controller effectively brings the system output to the vicinity of the desired set point quickly, which is advantageous for systems where speed of response is critical. However, the overshoot and minor oscillations indicate that while the proportional control improves response time, it may require tuning to minimize overshoot and enhance stability. This could involve adjusting the gain or incorporating additional PI or PID elements to refine the control strategy for smoother convergence to the set point without oscillations.

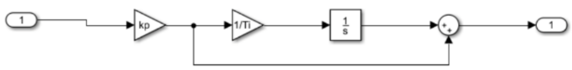
Question 10

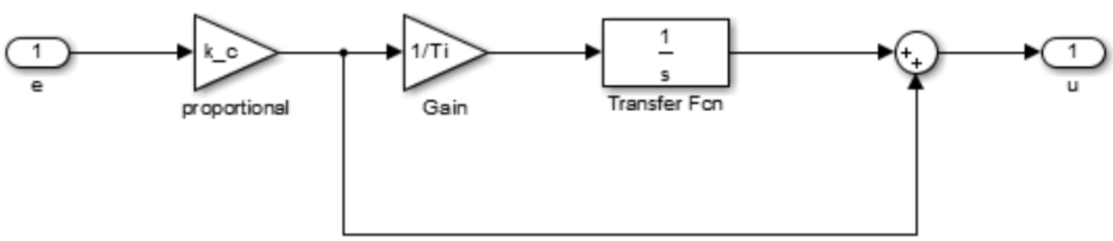
$$PI(s) = k_c\left(1 + \frac{1}{T_i s}\right)$$

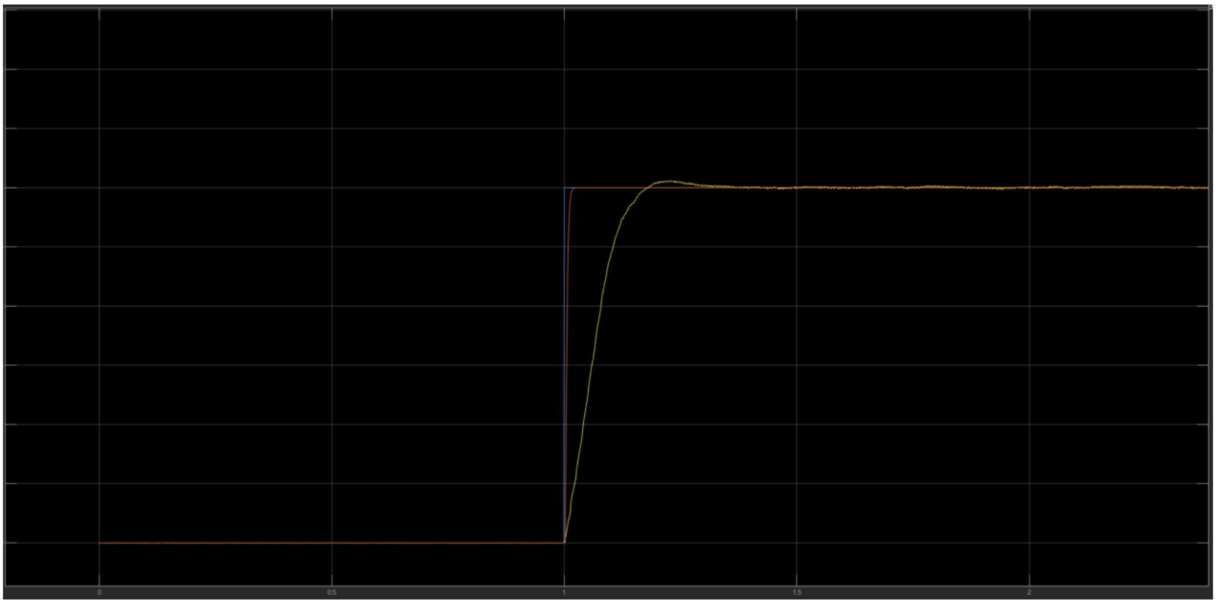
		k_c	T_i	Optimum conditions
Hrones -Type A	PI	$k_c = \dfrac{100\,\tau}{286\,K\,d}$	$T_i = 1.2\,\tau$	✓ No overshoot ✓ Minimum response time





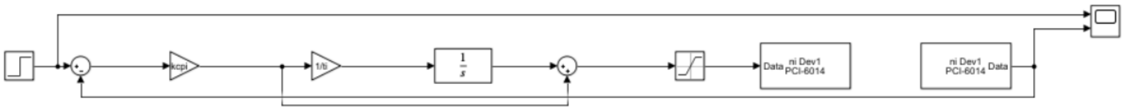


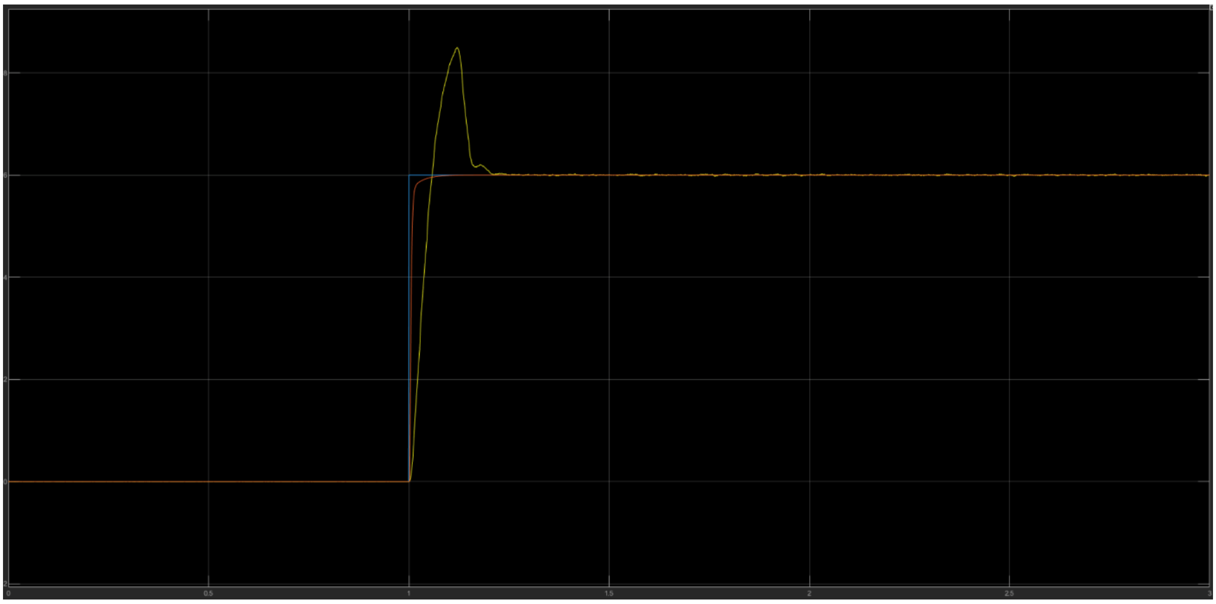




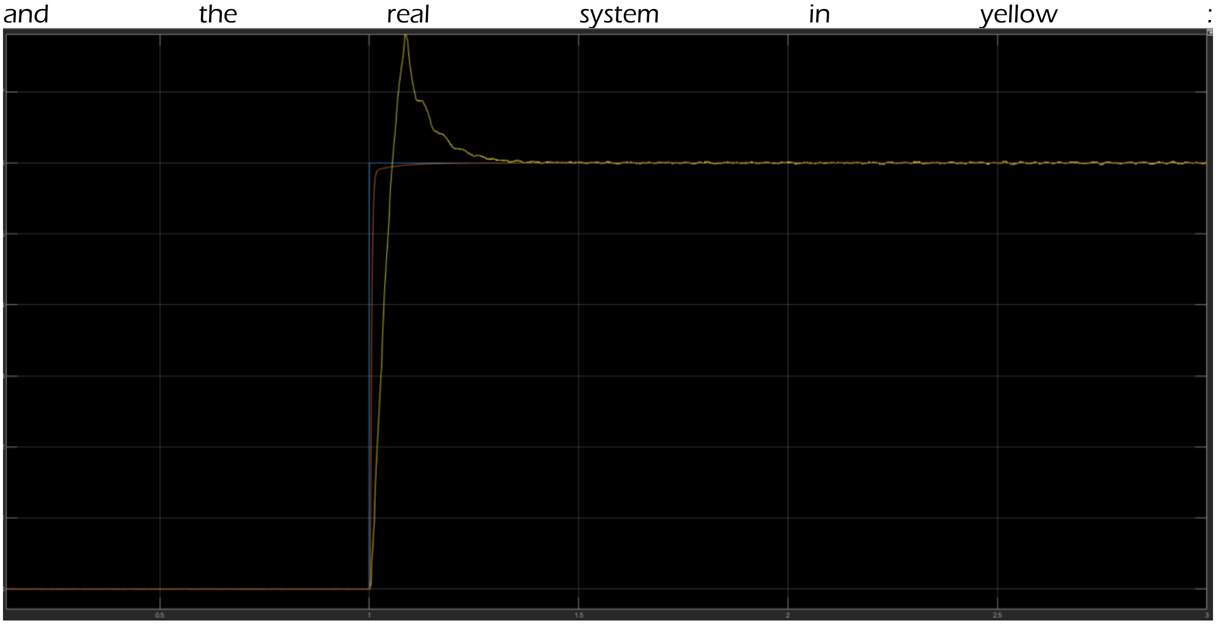
Looking at the graph provided, it is evident that the model meets the required specifications, which include a response time of 0.2 seconds or less, no overshoot, no steady-state error, and a gain of K=1.

G. Question 11

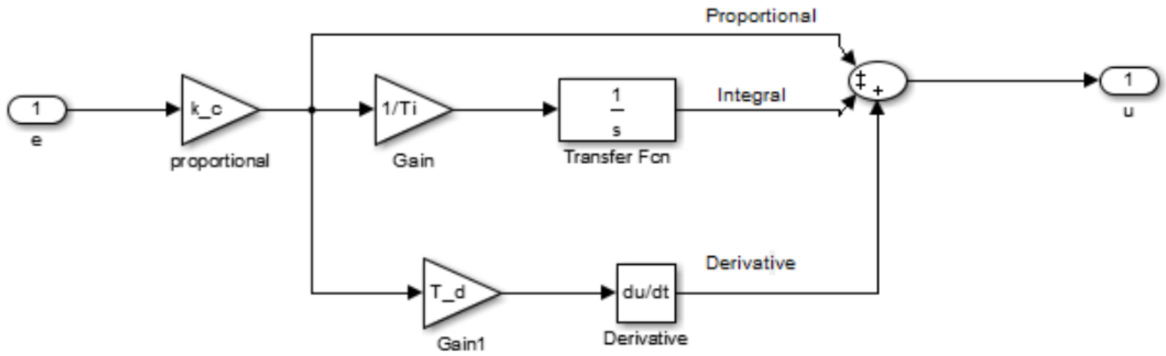




Discussing the outcomes with the actual model, we observe a substantial overshoot in the real system. This likely results from the controller's integral coefficient, which excessively adjusts the voltage sent to the motor. This is illustrated in the comparison of the two curves below, with the model shown in orange and the real system in yellow :



H. Question 12 and 13



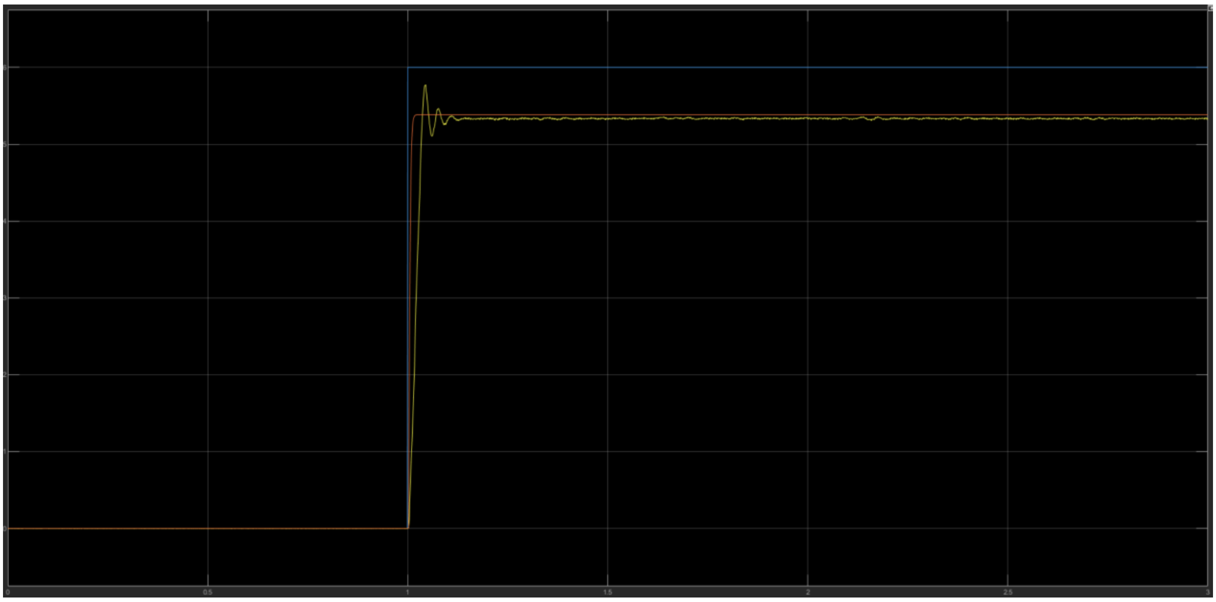
		k_c	T_i	T_d	Optimum conditions
Reswick-Type A	PID	$k_c = \dfrac{100\ \tau}{167\ K\ d}$	$T_i = \tau$	$T_d = 0.5\ d$	✓ No overshoot ✓ Minimum response time

$$PID(s) = k_c\left(1 + \frac{1}{T_i s} + T_d s\right)$$

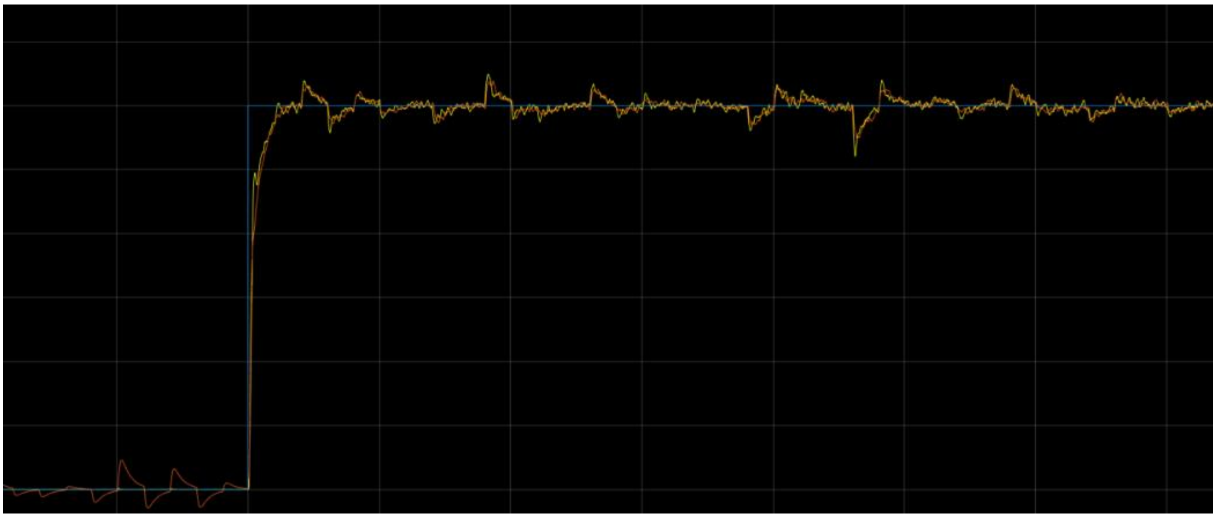
In our laboratory exercises, we focused on controlling the speed of a DC motor using the Broida method and PID controllers, tested under both theoretical and real system conditions with MATLAB and Simulink.

We employed the Broida method for its effectiveness in open-loop stable processes, which allowed us to identify and validate the characteristics of the DC motor system through careful comparison of output responses with the real system's behaviour.

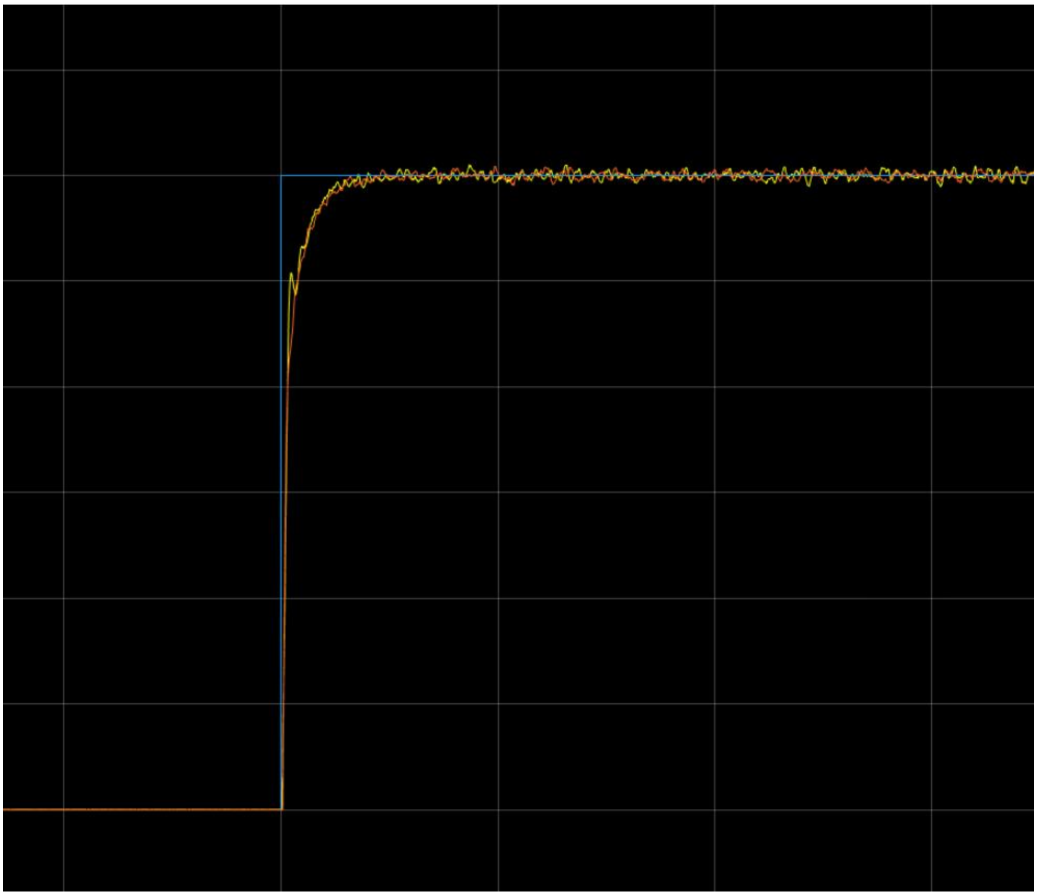
In the control phase, we explored various controller blocks—Proportional (P), Proportional-Integral (PI), and Proportional-Derivative-Integral (PID). Our findings indicated that the PI controller was most effective, especially after introducing disturbances which initially caused massive overshoots. However, both the theoretical model and the real system eventually stabilized at the desired voltage.



The PI controller's superiority was evident as it handled system dynamics more smoothly compared to the PID controller, which introduced excessive oscillations at peak voltage and was deemed unnecessary due to the lack of value added by the derivative component.



The experiments demonstrated that while the model responded quicker and with fewer disturbances, both setups stabilized comparably. We concluded that the use of fixed controller values provided more reliable testing conditions than random values, improving our ability to control the DC motor's speed accurately.



Overall, these laboratory sessions were crucial in enhancing our practical understanding of system control, proving educational and essential for our academic progression. The hands-on experience reinforced our theoretical knowledge, preparing us effectively for our finals.

I. Conclusion & Perspectives

In this laboratory exercise, we explored the effectiveness of Proportional (P), Proportional-Integral (PI), and Proportional-Integral-Derivative (PID) controllers in regulating the speed of a DC motor within a MATLAB and Simulink environment. Our findings revealed that the PID controller, with its combined control actions, significantly enhances system responsiveness, stability, and accuracy compared to P and PI controllers alone.

The practical application of these controllers demonstrated their potential in fine-tuning system responses to achieve desired performance levels, particularly in reducing overshoot and stabilizing the motor speed. This experiment not only reinforced our theoretical understanding of control systems but also provided valuable hands-on experience in optimizing complex control environments.

This study underscores the importance of precise controller settings and system feedback in achieving optimal performance in control systems. Moving forward, these insights will guide more advanced applications and improvements in control strategy implementation.