Hardware Implementation of a PID Controller-Based System

Objective:

To design the hardware of the PID Controller-Based System and observe the changes in independent P, I, and D parameters on the closed-loop system response

Apparatus Required:

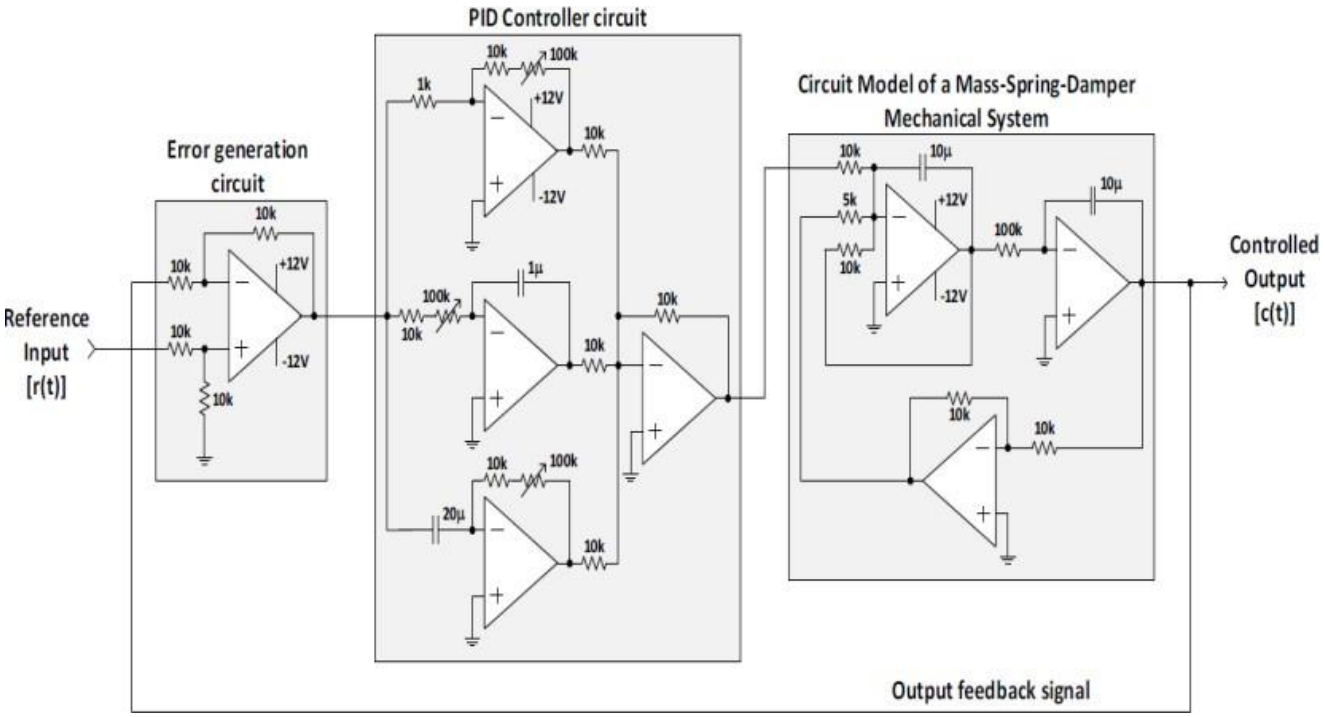
S No	Components	Specifications	Quantity
1	Resistors	10kΩ	15
2	Resistors	1kΩ	1
3	Resistors	100kΩ	1
4	Potentiometers	100kΩ	3
5	IC	LM 324A	3
6	Capacitors	10μF	2
7	Capacitors	1μF	1
8	Capacitors	20μF	1
9	Resistors	5kΩ	1

Theory:

The laboratory experiment presented herein provides an opportunity to model, design, simulate, and implement a complete feedback control system in a very inexpensive way by using only a couple of quad op-amp ICs and a few discrete resistors and capacitors. It includes the design various controller configurations (P, PD, PI, and PID) and investigate the effects of proportional, integral, and derivative gains on the system performance including steady-state error, damping ratio, overshoot, rise time, time constant, settling time, frequency of oscillation, and system stability. This laboratory experiment emphasizes developing an intuitive understanding of PID controller concepts grounded in theory and design.

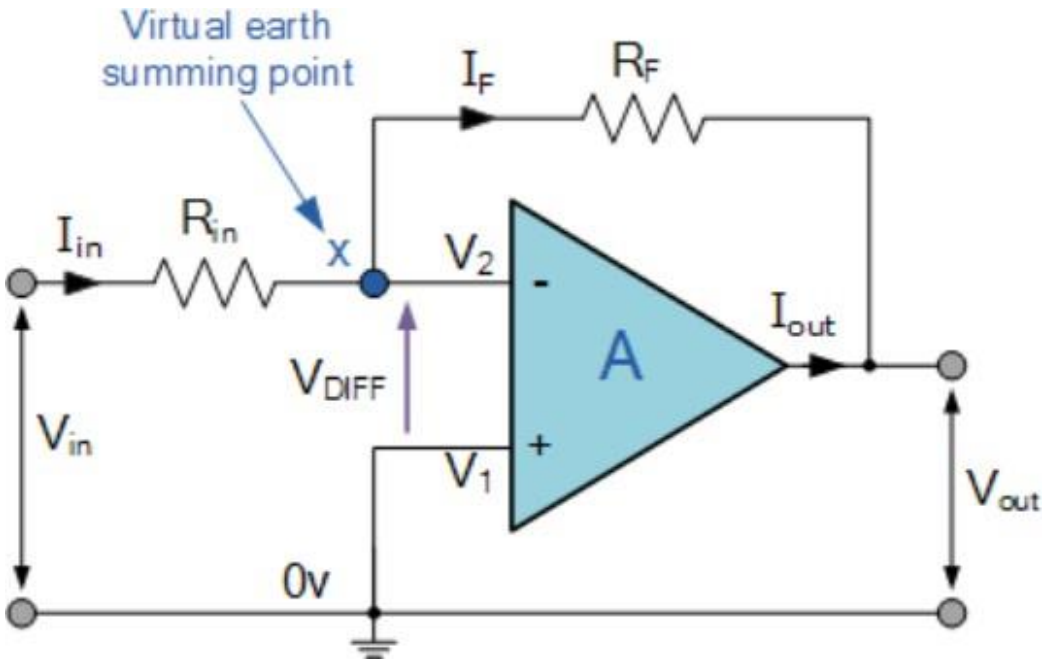
The error signal can be generated by using an op-amp-based difference amplifier circuit and the PID controller can be implemented using an inverting amplifier circuit for proportional gain, an integrator circuit for the integral gain, and a differentiator circuit for the derivative gain. Finally, an op-amp-based summing circuit is used to combine the P, I, and D circuit outputs to create a PID controller. A complete circuit schematic of the mass-spring-damper feedback control system including the plant, error generation, and PID controller is shown in Figure:

Circuit Implementation of Mass-Spring-Damper Closed-Loop Control System:



Once the designed circuit representing the PID-controller-based mass-spring-damper system is prototyped, it can be tested for step response by connecting a low-frequency square wave signal (e.g., 0.5 Hz) as the reference input while monitoring both the reference input and the controlled output using a two-channel oscilloscope to evaluate the system's steady-state and dynamic performance.

Circuit Diagram and working of an inverting amplifier:

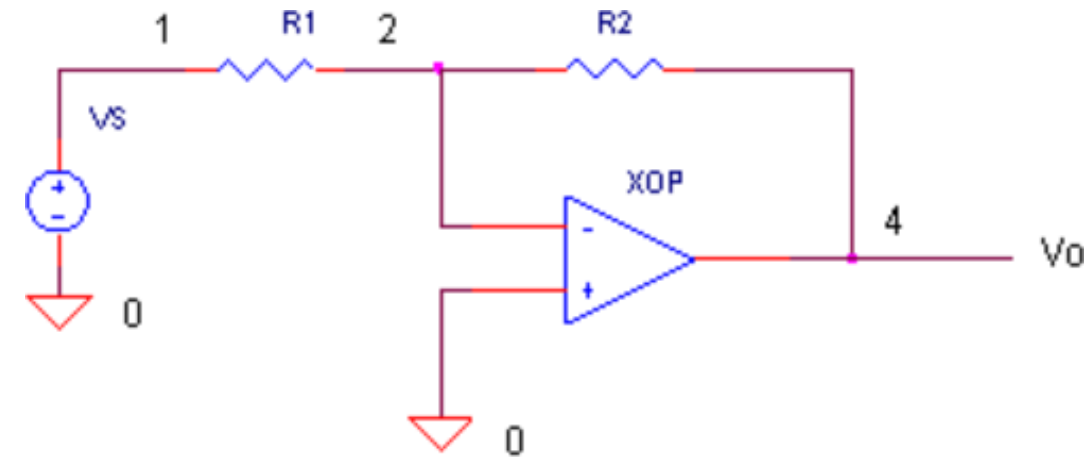


In this Inverting Amplifier circuit, the operational amplifier is connected with feed-back to produce a closed-loop operation. When dealing with operational amplifiers there are two very important rules to remember about inverting amplifiers, these are: "No current flows into the input terminal" and that ***V1 always equals V2***.

Then, the **Closed-Loop Voltage Gain** of an Inverting Amplifier is given as.

$$\text{Gain (Av)} = \frac{V_{out}}{V_{in}} = -\frac{R_f}{R_{in}}$$

For determining Kp through the op-amp function:

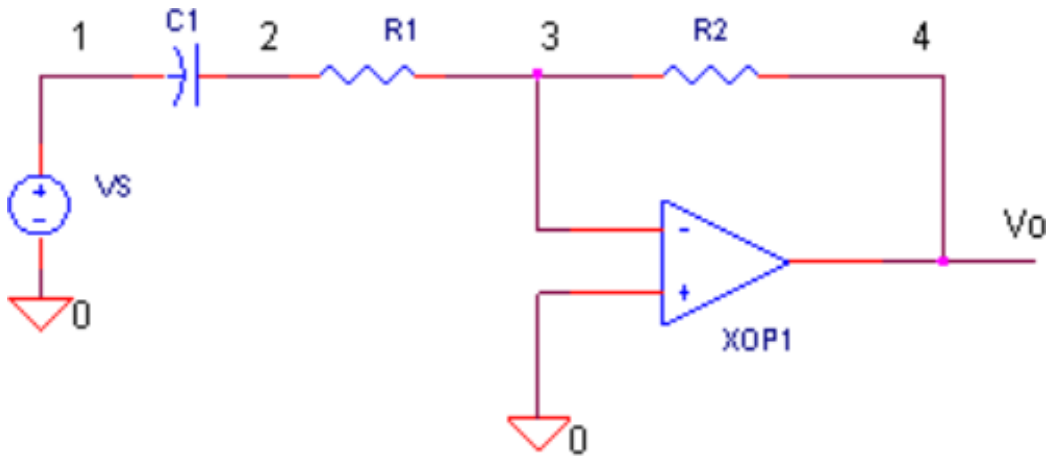


Kp can be determined by the voltage gain happening in the circuit:

$$Vo = -Vs \times \frac{R2}{R1}$$

Kp=R2/R1

For determining Kd through the op-amp function:



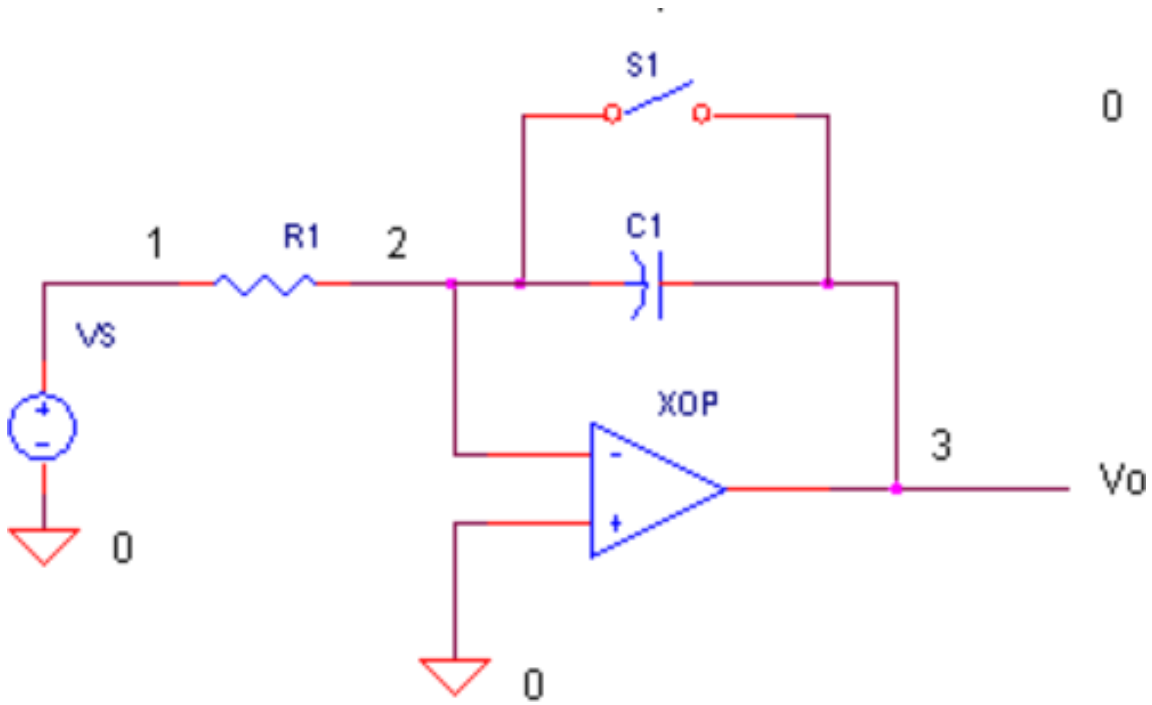
Kd can be determined by the voltage gain happening in the circuit:

$$i = C1 \cdot dVS / dt$$

$$Vo = - C1 \cdot R2 \cdot dVS / dt$$

Kd=C1*R2

For determining Ki through the op-amp function:



From the figure, K_i can be determined by the output voltage:

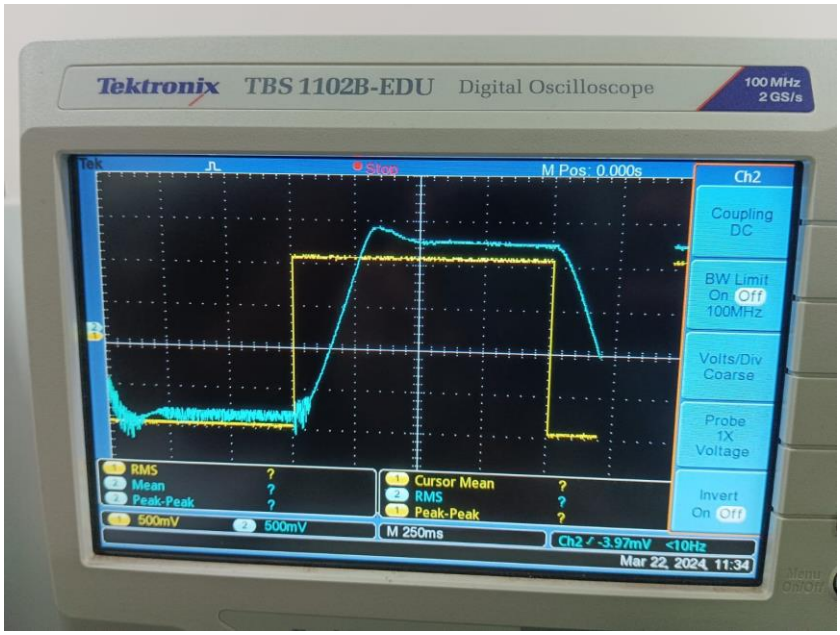
$$Vo = -\frac{1}{C1}\int_0^T \frac{Vs}{R1}\,dt$$

Ki=1/(C1*R1)

Observations:

Effects of independent P, I, and D parameters on closed-loop system response:

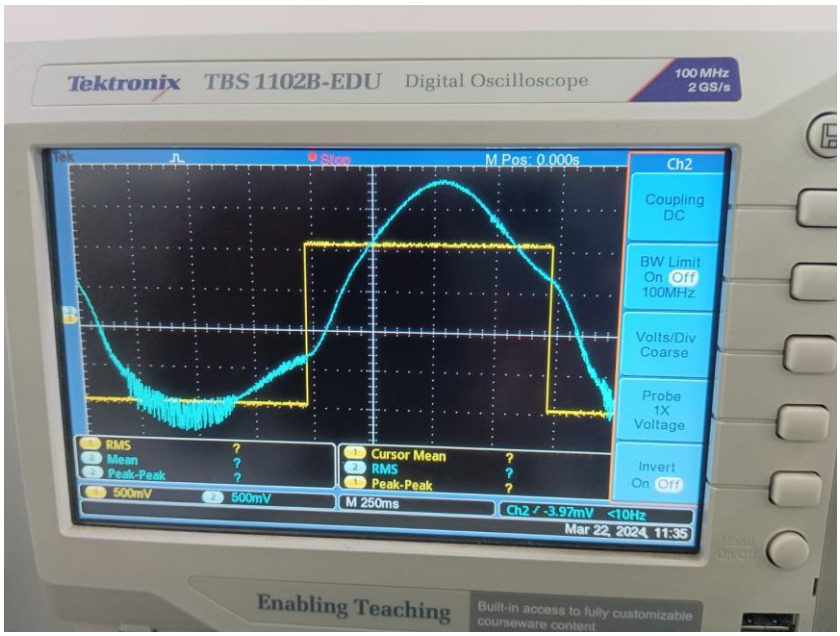
	Rise Time	*Overshoot*	*Settling Time*	*Steady State Error*	*Stability*
Increasing Kp	*decrease*	*increase*	*increase*	*small change*	*decrease*
Increasing Ki	*decrease*	*increase*	*increase*	*decrease*	*decrease*
Increasing Kd	*increase*	*decrease*	*decrease*	*very small change*	*increase*



This is the initial graph that was observed after all the connections were made.

a) ***On increasing Kp***: A significant decrease is observed in the rise time while the overshoot increases gradually. The settling time and steady-state error are observed to have changed. The stability of the system decreases because, in the case of proportional gain the magnitude of correction signal m is proportional to the error e. So, if K is increased, the sensitivity of the controller to error is increased but the stability is impaired.

Here, the very first graph is only **Kp1**.

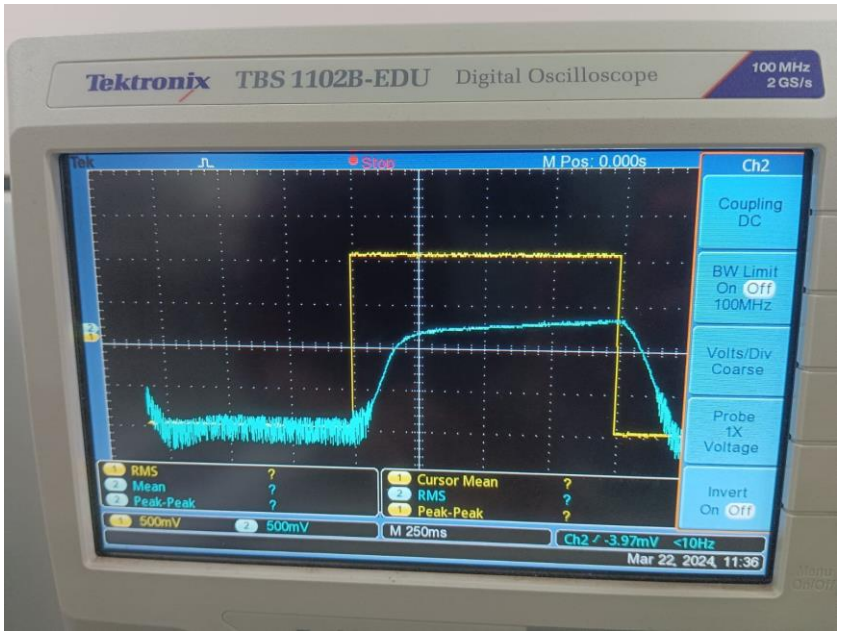


Kp2

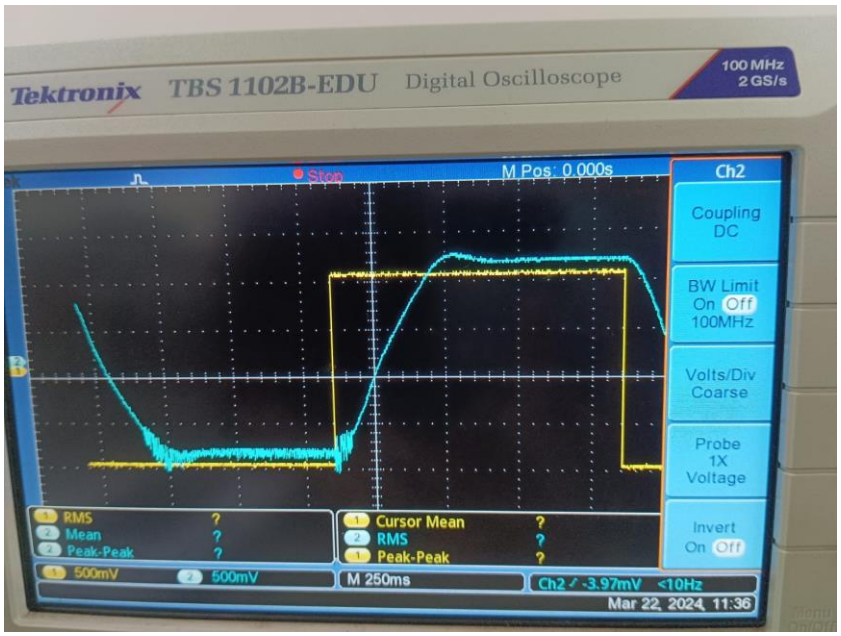
The relation between the values of Kp is:

$$Kp2 > Kp1$$

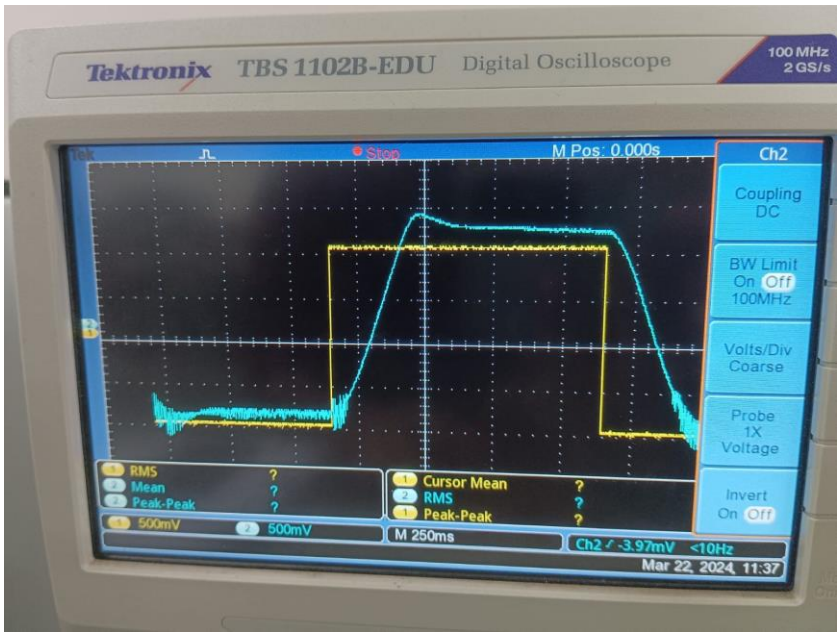
b) ***On increasing Ki*** : The rise time in this case decreases but steady-state error and stability decrease due to a lowering in the speed of the transient response. While the overshoot and settling time increase.



Ki1



Ki2

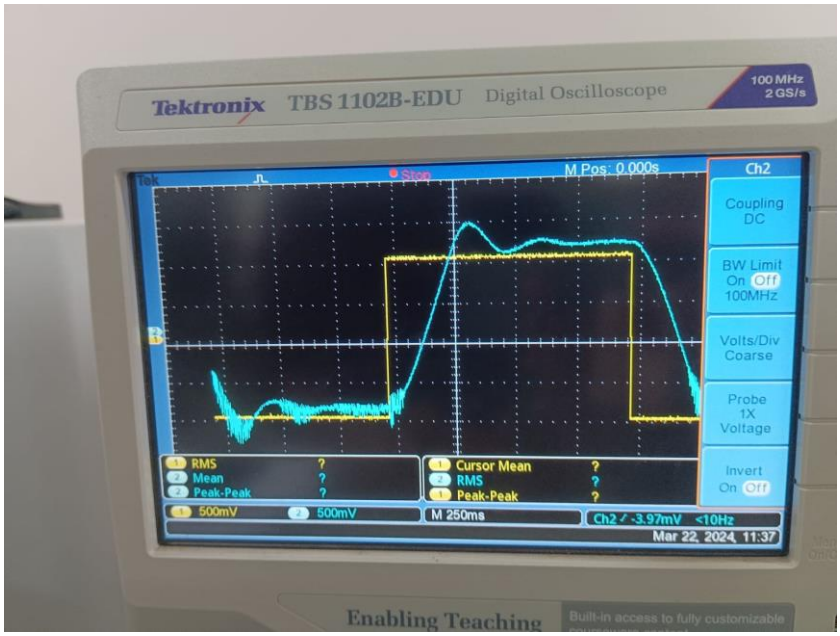


Ki3

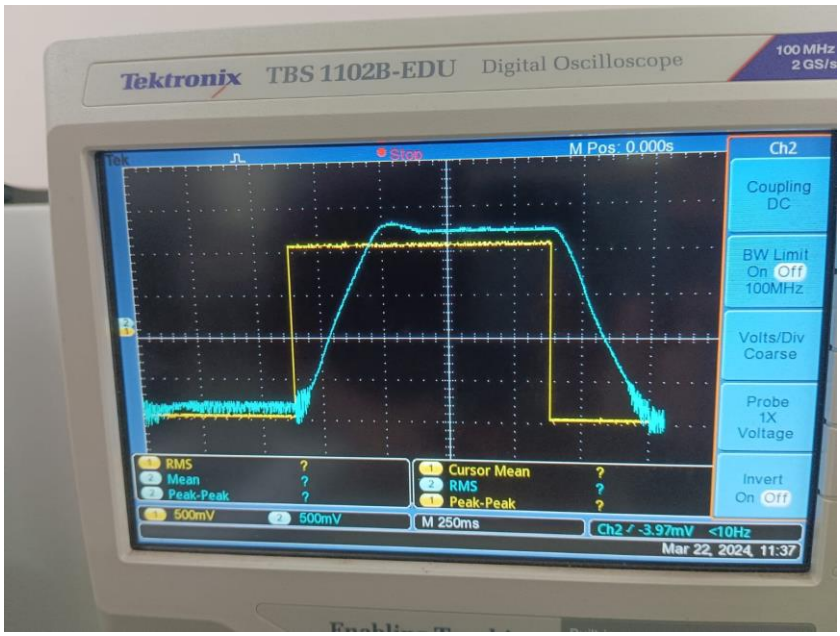
The relation between the values of Ki is:

$$Ki3 > Ki2 > Ki1$$

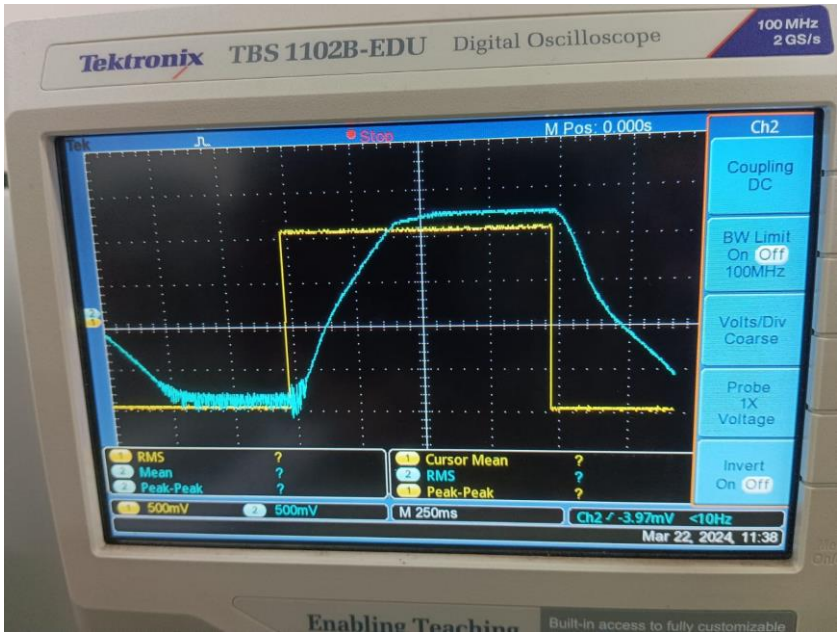
c) ***On increasing Kd*** : There is an increase observed in rise time due to a decreased number of oscillations. Overshoot and settling time decrease gradually but very small changes are visible for steady-state error. The stability of the system is enhanced because of the decrease in the remaining errors of the system.



Kd1



Kd2



Kd3

The relation between the values of Kd is:

$$Kd3 > Kd2 > Kd1$$

Conclusions:

1. The simulation results demonstrated that variations in P, I, and D gains have a significant effect on the system's steady-state error. Fine-tuning these gains allows for minimizing or eliminating steady-state error, ensuring the system reaches the desired output accurately.

2. The study revealed a clear correlation between the choice of gains and overshoot. Adjusting P, I, and D gains allows for controlling overshoot, preventing excessive oscillations, and enhancing system stability.

3. Testing the hardware implementation under different operating conditions, such as varying loads or disturbances, can demonstrate the robustness of the PID controller. A robust controller should maintain stable performance despite external factors.

4. The concept of the right way to connect potentiometers is very important for a clean output.

5. The process of implementing the PID controller in hardware may reveal certain challenges such as noise interference, sensor limitations, or computational constraints. Understanding these challenges can lead to improvements in future iterations of the hardware design.