Control Systems

Lab Assessment #6

Date of experiment – 06/10/2023

Group Members: Adarsh Santoria (B21176), Vanshaj Nathani (B20237)

Objective:

To understand the process to control the given dc motor(s) using Arduino and Simulink in open and closed loop form.

Experiment Design:

To accomplish our objective, we are required to practically implement a DC motor speed controller using the specified equipment and MATLAB. The circuit connections have been meticulously set up as follows:

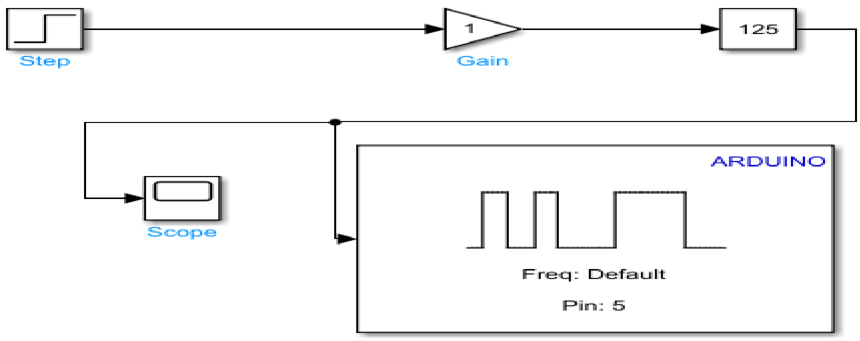
- Arduino Vin is connected to L298N 12V.
- Arduino GND is linked to L298N GND.
- Arduino D2 is interfaced with the Motor Encoder.
- Arduino D5 is connected to L298N IN1.
- L298N M1 is wired to the positive terminal of the motor.
- L298N M2 is attached to the negative terminal of the motor.

Our chosen set point, denoted by the reference value of 125, has been accurately generated using a step signal block followed by a gain operation. Importantly, this value falls within the rated RPM range of the specified motor, ensuring optimal performance and adherence to technical specifications. This consistent selection of the reference value underpins our control strategy in all cases, maintaining a standardized approach throughout the implementation process.

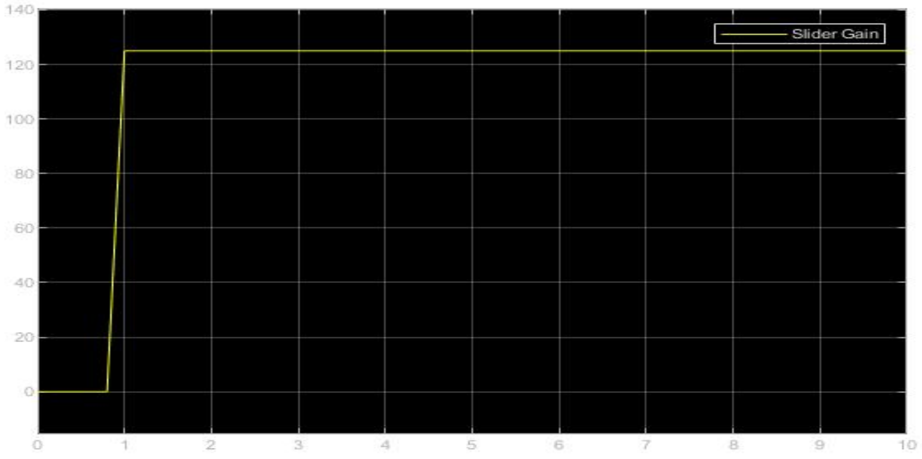
1. **Open loop control test scenario:**

   Here, we are using the digital PWM output of Arduino to provide signal to the motor.

Simulink Model:



Results:



Observations:

- By analyzing the plot, it is evident that our signal starts from zero and eventually reaches the specified value of 125, remaining stable thereafter.
- The nonzero transition time observed in the plot reflects the practical aspect of speed change, accounting for finite acceleration.
- Once the desired RPM is attained, the signal remains constant even when external disturbances are applied.
- It's crucial to note that the provided input only represents the signal we are supplying ; it does not accurately depict the actual speed at which the motor is moving.

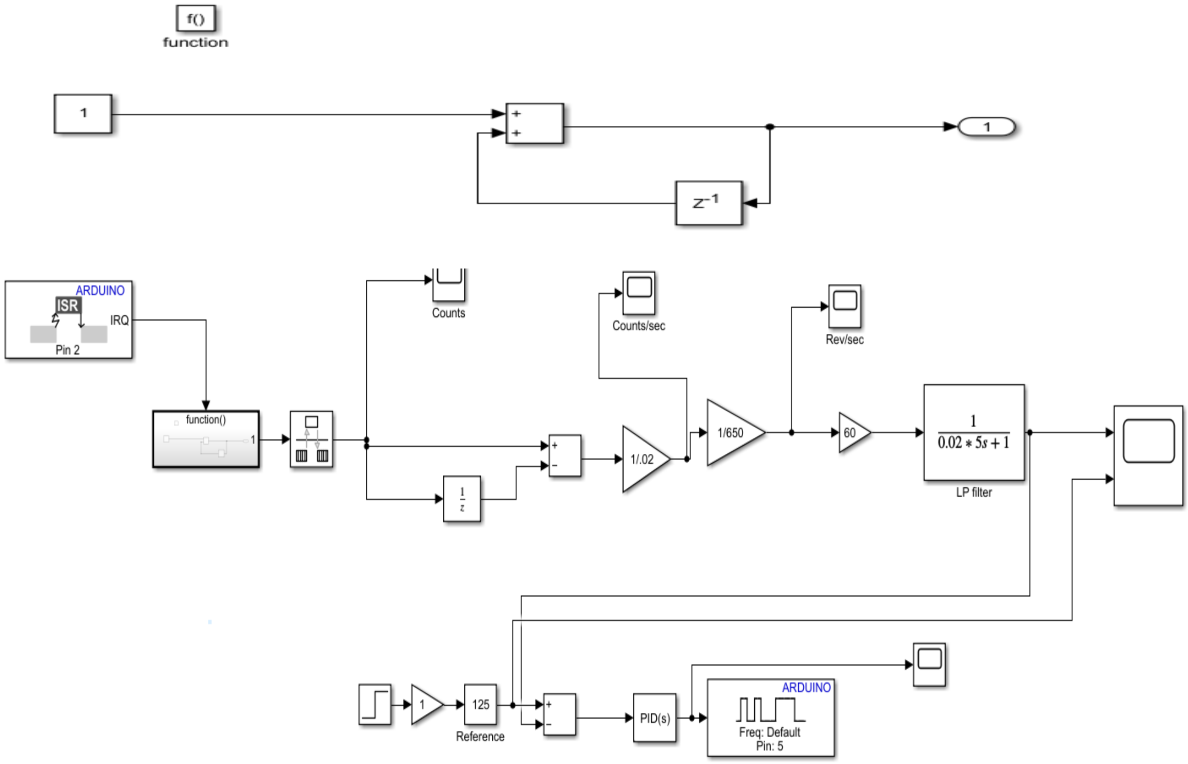
2. Closed loop control test scenario:

In the closed-loop configuration, we must accurately sense the motor's actual speed and utilize it as feedback for the controller.

Tuned Parameters:

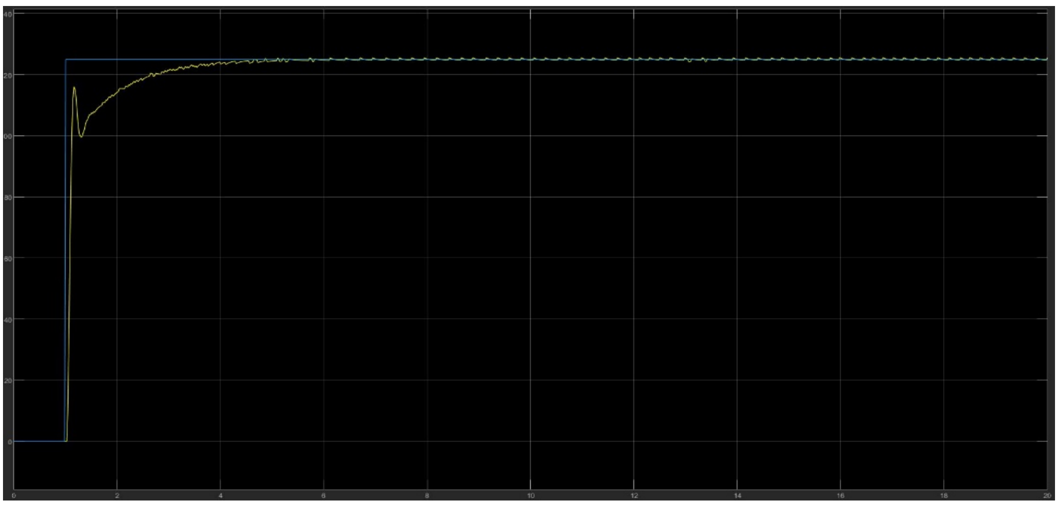
S.No	Controller	Parameter	Value
1	Proportional	Kp	13
2	Proportional Integrator	Kp	2
3	Proportional Integrator	Ki	3

Simulink Model:

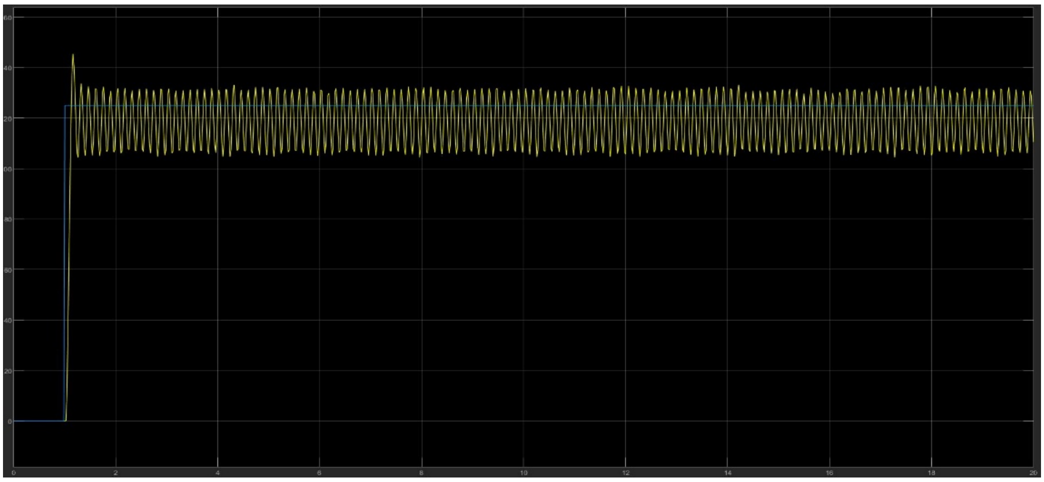


Results:

1. PI controller



2. Proportional controller:



Observations:

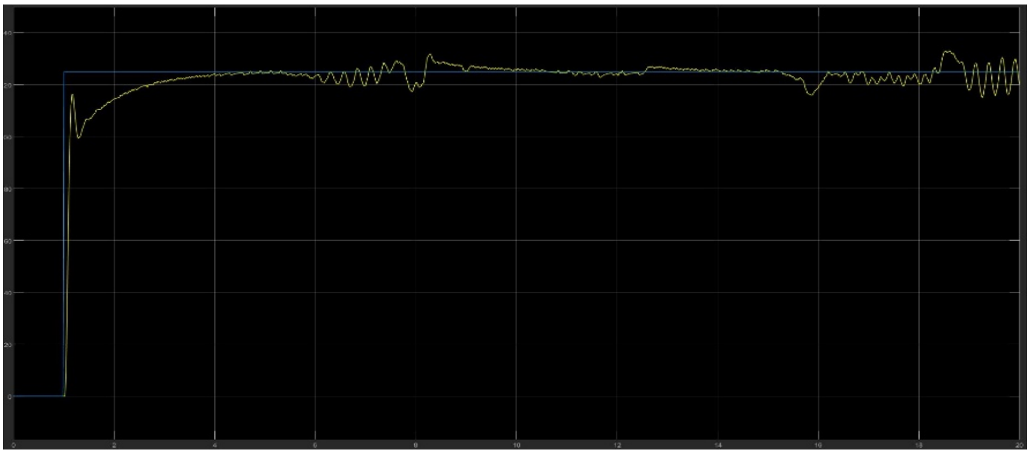
- Both PI and P controllers maintain stable average values consistent with the set reference, ensuring system stability.
- Proportional controller exhibits notable oscillations, while the PI controller shows minimal, insignificant deviations, indicating a more stable performance.
- The amplitude fluctuates by 9-12% in the Proportional controller, contrasting with the PI controller, which maintains a steady amplitude.
- Simple Proportional controller achieves stability faster (around 1.2 seconds) compared to PI controller (approximately 5.4 seconds), highlighting differences in response time.
- An intriguing observation in the Proportional controller case is the asymmetry between the reference line and oscillations, suggesting a challenge in precisely matching the average value of oscillations with the set reference.; it does not accurately depict the actual speed at which the motor is moving.

3. The disturbance:

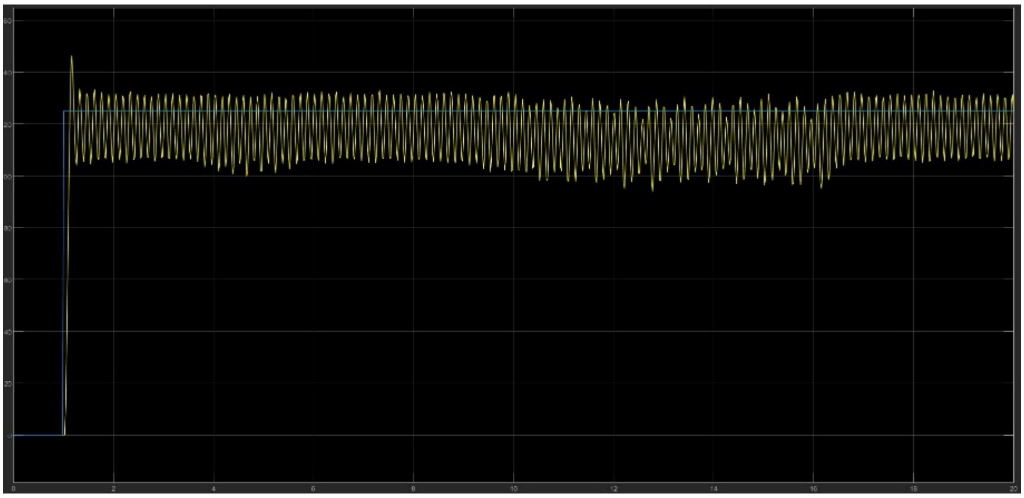
Disturbance has been applied by providing some torque to the shaft of the motor.
In case of open loop, there is no meaning to analyse the disturbance’s response as we are not monitoring.
Simulation model will be same as previous experiment.

Results:

1. PI controller



2. Proportional controller:



Observations:

- On application of the disturbance we can see that our signal diverts until the effect of the disturbance is significant and later settles to the stabilised reference values (back to oscillations in the Proportional controller case).
- It apparently seems that the effect on the PI is more than the P case.
- But it is not the truth. In both the cases the effect is same, but due to the oscillations in the Prop case where the disturbance range is affected by the oscillations range, the significant difference is not visible.

Improvements and Learnings:

1. Refining the filter to eliminate false or unnecessary oscillations.
2. Evaluating the error and implementing substantial adjustments when stabilising the system from disturbances.