Control Systems

Lab Assessment #7

Date of experiment – 20/10/2023

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

Objective:

To incorporate additional sensor data to adjust the reference on-the-fly with respect to the closed loop test scenario in lab exercise 6 (whose objective was to understand the process to control the given dc motor using Arduino and Simulink in open and closed loop form).

Experiment Design:

For completing our objective we need to implement practically the given use case for the smart dc motor speed controller which can adjust the reference on the fly based on the nearby environment (based on sensor data).

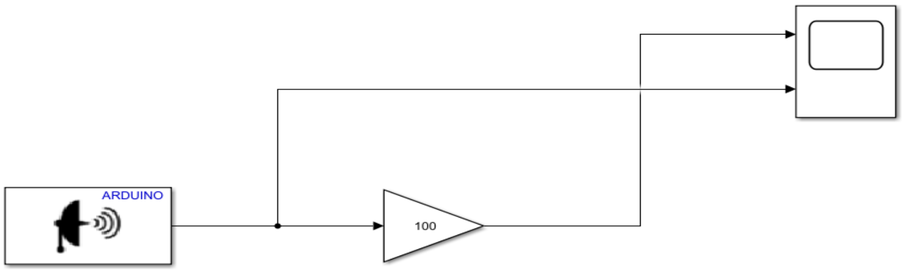
The circuit connections are made as follows:

S.No	From	To
1	Arduino Vin	L298N 12V
2	Arduino GND	L298N GND
3	Arduino D2	Motor Encoder
4	Arduino D5	L298N IN1
5	L298N M1	Motor +ve
6	L298N M2	Motor -ve
7	Ultrasonic Trig	Arduino D3
8	Ultrasonic Echo	Arduino D4

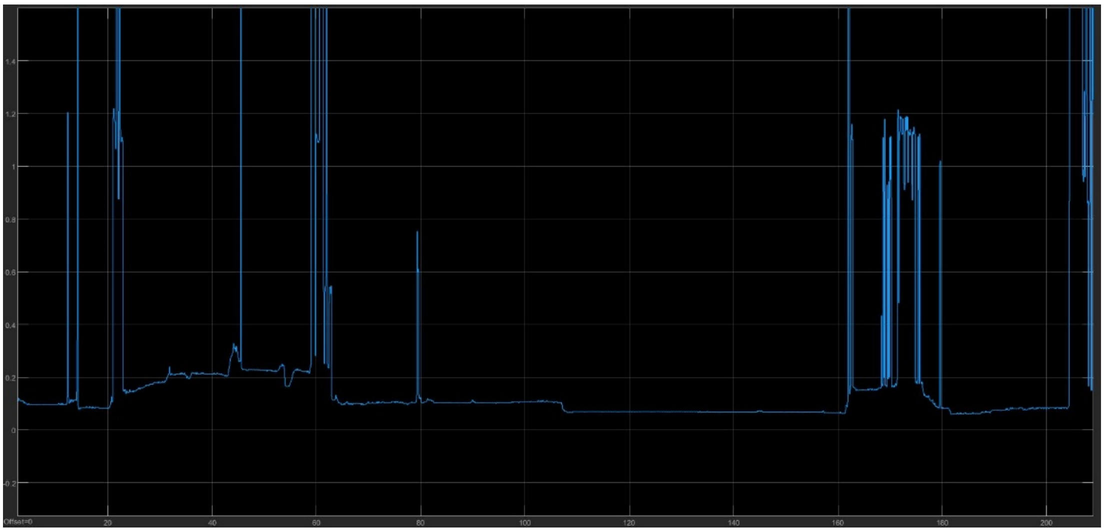
Set point: As a reference, we are using a value of 125. The signal with this value has been achieved using the step signal block followed by a gain. As this value falls under the rated rpm of the given motor, there is no issue. In all the cases this value is chosen as the reference. So, the task is to sense the distance, depending on the range change and adjust the speed to the respective limit.

1. Fetching the sonar data:

Model:



Results:



Inferences:

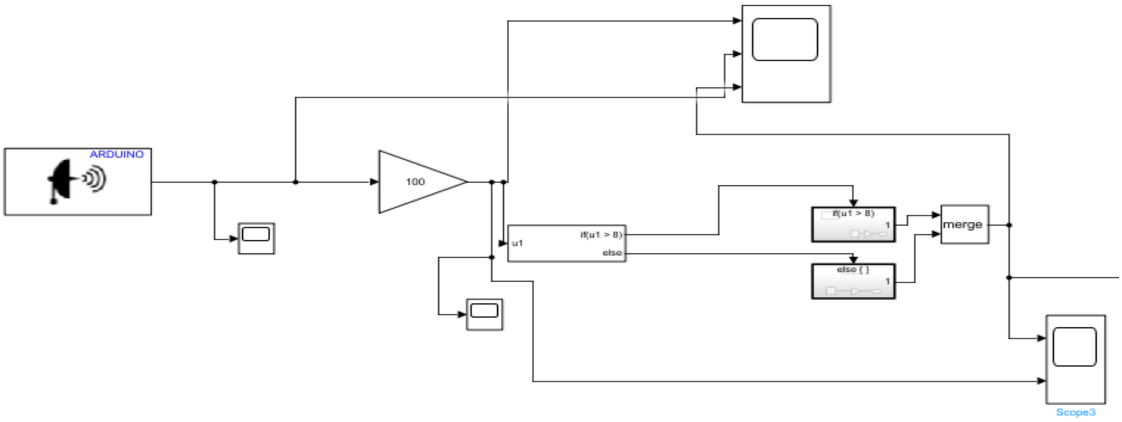
- From the plot we can see that the output varies as the distance changes.
- The sudden spiking is due to the scenarios where the object is being detected at a large distance (far obstacles).

2. Threshold setting and operating speed:

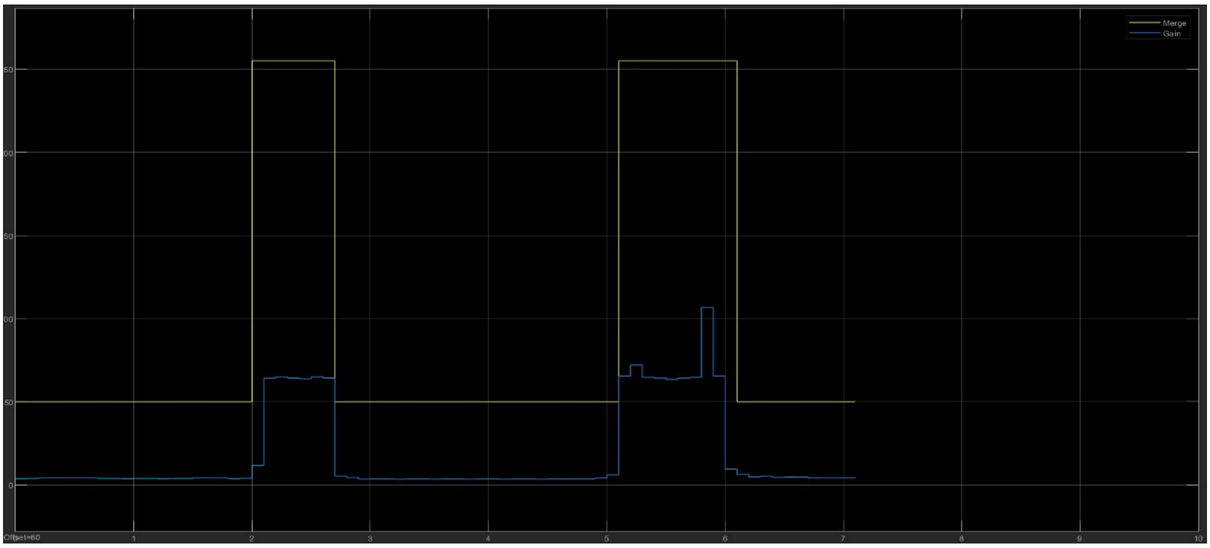
Let us continue with taking a threshold at 8 cm and operating speeds as:

- 255 rpm if the object is far than 8cm.
- 50 rpm if the object is closer than 8cm.

Model:



Results:

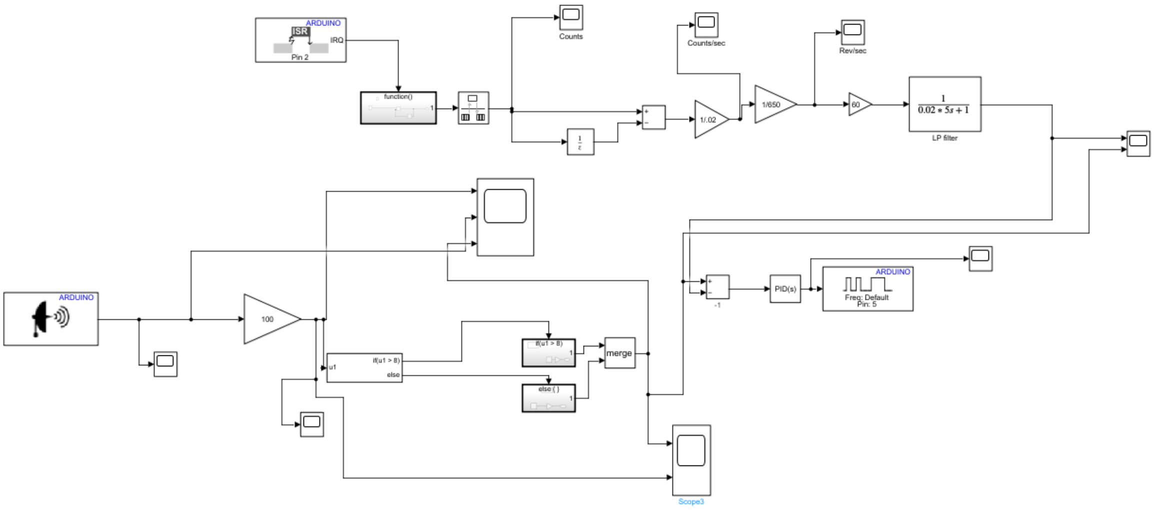


Inferences:

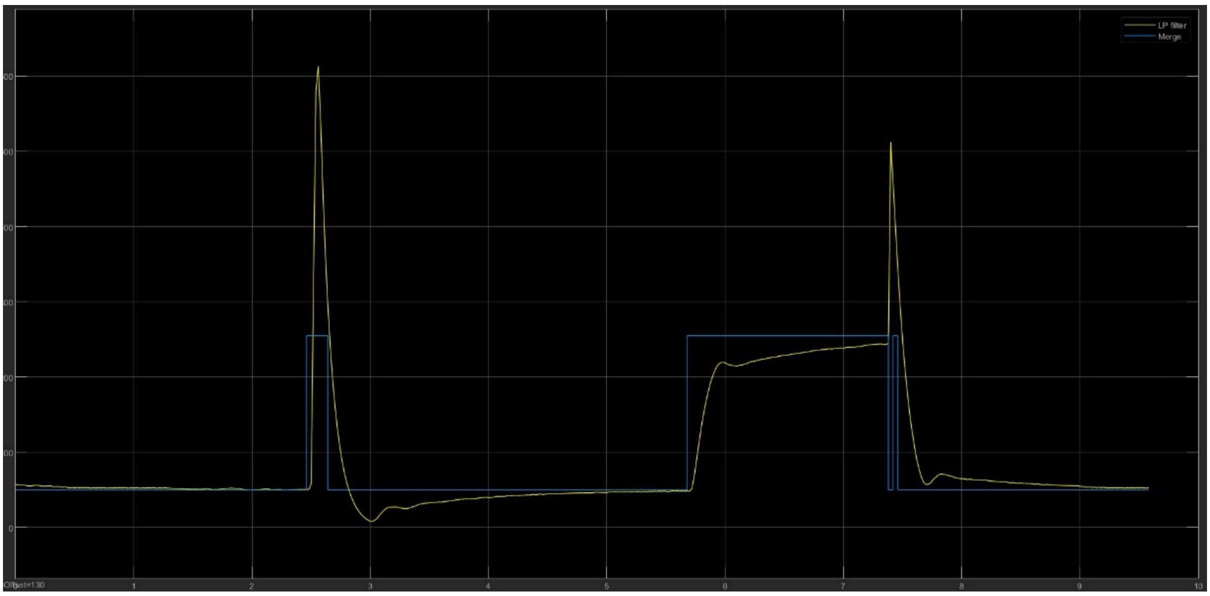
- From the plots, we can observe that the reference operating speed changes as the distance varies.
- The blue line indicates the distance in cm.
- The yellow plot represents the reference operating speeds in rpm.
- The functionality and the logic are working fine.

3. The Simulink model for adjusting:

Model:



Results:



Inferences:

- Now we need a final model which shuffles the operating speed according to the distance (wrt to the 8cm threshold).
- And also, we need a PI control which stabilises the speed to the required operating speed.
- From the plot we can clearly observe the logic and speed stabilisation.
- We used KI as 3 and KP as 2 for this optimal behaviour.
- Note: The plot shown is a zoomed/scaled version.

Improvements and Learnings:

1. We used a 2-level switching here which can be extended to the multi-level switching.
2. Also, we may develop a function for smooth detections which helps in smooth avoidance.