Experiment 2 Report

Sai Gaurav Anugole 170070008
Titas Chakraborty 170070019
Jayesh Choudhary 170070038

1 Aim

The aim of this experiment is to design and implement a PID controller for the Spark V robot. The controller would help it traverse on the given path within a given time limit using IR sensors for position data provided on the robot for this purpose.

2 Objective

We have to implement a PID controller on the sensor values which decides the direction of motion : straight, left, right and back and the velocity of the bot in that direction to complete one lap of the track within the time constraint of 30 seconds.

3 Challenges faced

- One of the major challenges was to decide the error function for the PID controller to be implemented. We started with taking 2 different PID controllers with parameters l/c and r/c where l is the left sensor value, r is the right sensor value and c is the central sensor value. However, that proved to be unsuccessful. We tried several other expressions for the error function but they proved to be quite as unsuccessful. Finally, we decided to use a combined error function for l, c and r. That gave us better results than previously achieved.
- However, even with a single error function, we were unable to complete the track. We observed that the bot was turning but with a certain time lag. We reached to a conclusion that there was a feedback delay so we tried to reduce the delay of the loop but it was of no use. Finally, we realized that printing the sensor values was the issue and as soon as we commented that part of the code, the issue was resolved and the bot turned without any time lag.
- To decide whether to take a soft turn or a hard turn, we initially tried to take all turns as hard, believing that the velocity would give the control that the PID needed. However, this turned out to be unsuccessful. Then we decided to threshold for soft and hard turns and control the velocity using PID. Hence, once the final parameters were decided (conditional on the difference of l and r with c), thresholding for soft and hard turns was solved. But this took up a lot of time too since it required excessive trial and error.
- Initially only k_p was introduced and then to optimize the performance, k_d and k_i was introduced. Tuning the values of k_p and k_i took up some time too as they were tuned by trial and error.
- Sometimes, the bot was turning erratically even after doing the things previously mentioned. It turned out that this was due to some imperfections in the track provided. Some parts of the black track were partially white, which caused the bot to misinterpret its position. This problem was solved by using the other track provided, which was uniformly black throughout.

4 Flowchart

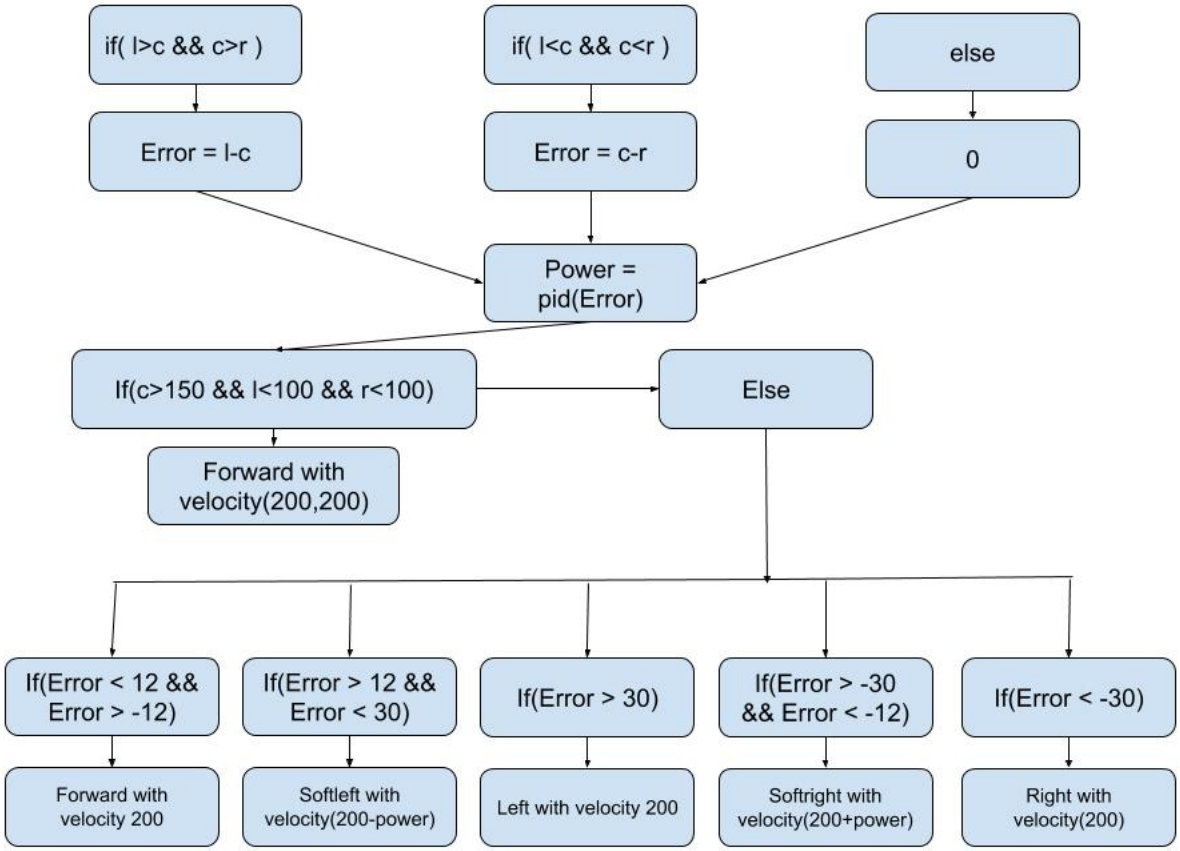

Figure 1: Flow Chart of the conditional circuit within the loop where the value of l,r and c are recursively calculated and motion is determined

5 Results

The values of the PID tuning parameters chosen were:

- $K_P = 4$
- $K_I = 1$
- $K_D = 0.001$
- Time of traversal = 25.8 sec

6 Inference

- We saw that implementing control theory in a real world application is a challenging task due to the many non idealities present in the real world.
- We observed the implementation of PID control in the bot. We also inferred that it is extremely critical to select an appropriate error function.
- We saw the effect of K_p, K_d and K_i in the behaviour of the line follower.
- We also observed that any time delay in the implementation of control problems can cripple the system.