

Owen Richards
CS3501: Embedded Circuits
12/12/2021

Line Follower Challenge Report

The overall strategy for the line follower challenge was to get as many sensor inputs using the widest range of sensors so that the robot can move around the track as quick as possible. The two main optimizations that I used to make the robot quicker was to have a variable set to the previous state so that when the robot gets lost it can look at the previous state it was in and do a certain function depending on what was the last state. This is a sort of memory that the robot used so that it wouldn't do the same function if lost, but instead do a more optimal action. The second optimization that I did was making the reflectance read value to return the whole range of the sensor. This made it so that depending how far left or right of the center the robot was, the robot would increase the motor speed on one side of the robot to slightly turn back to the center of the track. This solved the problem of constantly stopping and rotating to recenter then running the go straight command. Ultimately, having these different motor speeds for different reflectance read values made it so that the robot moves smoother and faster.

The issue that arose when having two finite state machines interacting with one another, was that the program could get stuck in the i2c driver. This issue arose because when reading the sensor values in the while loop, the program would miss a read and then get stuck in the i2c driver because of missing this read. To solve this issue, I added a counter so that the program would read the sensor values only after the counter reaches a certain number. The two finite state machines interacted with one another by first updating the `liftedUp` and `StopSlowNormalLight` variables in the sensor finite state machine. Then these two variables will be used in the normal fsm so that if `liftedUp` was updated to 1 the robot would stop since its been lifted. The same premise applies for `StopSlowNormalLight` where depending the value of this variable, the robot would either stop, move slower or go at normal speed.

Other than these changes, all robot functions remained the same as specified in previous reports.