The Sea Otters

Mohammed Alghoribi: Report editor
Michaela DiVito: Report author
Jackson Hopper: Code editor
Noah Schliesman: Technical Lead

ENGR 102 Electromechanical System Design
Dr. Nadieh Moghadam
9 December 2021

Detailed Timeline

We planned to write and modify existing code for Project 1 on 11/18. By the end of class on 11/23 we planned to be done with the code for both Project 1 and Project 2. On 11/30 and 12/2 we plan to revise our code until the car meets the requirements for the project. If we have time on 12/2, we will attempt to incorporate the optional elements from Project 2 into our project. We will get started on writing the report right away, and continue adding details as we go along. We will not be able to finish the report until 12/7, when we are able to demonstrate the capabilities of the car to the class.

Car Performance

Efficiency Analysis and Score (out of 10)

1. Completing each project request within 3 minutes: 10. Our team completed all project tasks well within the allotted time.
2. Completing all requests in order: 10. We tested our criteria in order, starting with Project 1 Line Tracker, then Project 1 Obstacle Avoidance, then finally Project 2.
3. Line tracking task working as specified: 10. The line tracking task correctly allowed the car to follow the line, and move backwards if it got off course until it could sense the line again.
4. Obstacle avoiding task working as specified: 9. Once we reattached the Ultrasonic Sensor in the correct orientation, the car functioned perfectly. Once the car sensed the obstacle in the way it would turn according to where the obstacle was in relation to the car.

Technical Achievement

Key Features of Robot Car

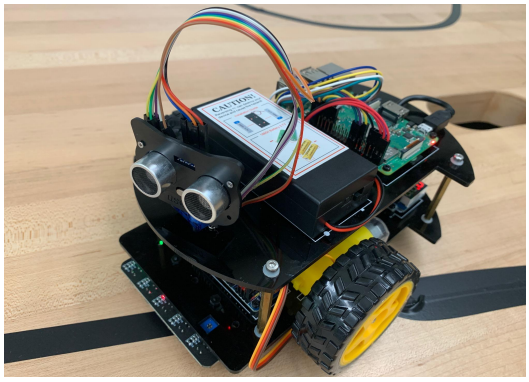

Figure 1: The entire robot car. Here we can see the Line Tracker and Ultrasonic sensor in context with the rest of the vehicle.

Our Raspberry pi robot car is unique because it can take input from the user to drive at a specified speed. We ask the user to input a number 1-3 and depending on the value given, it will change the speed that the car operates at. Our car also has two different drive modes in

which the user can also choose from, Line tracker or Obstacle avoidance. These two modes have a different goal in mind giving the user the option of completing two separate tasks with the same robot. Using the speed given to the Raspberry pi from the user the robot can go forward at that specified speed and depending on the mission selected by the user, can either follow a line precisely or look ahead of itself and avoid running into obstacles. Our car is also able to have both line tracking and obstacle avoiding capabilities simultaneously. It will avoid objects, but if none are present, it will follow the path laid out by the black line thanks to the line tracker.

Figure 2: Line Tracker sensor attached to the robot car, seen here tracking the line on the table.

Figure 3: Ultrasonic sensor is attached to the robot car.

```
//1. "Car speed is taken from the user"
int SPEED = -1; //sets SPEED to an invalid condition
while((SPEED < 1) || (SPEED > 10)) { //validator pattern checks to see if i
    printf("\nEnter an integer between 1 to 3 to specify car speed:\n");
    scanf("%d",&SPEED); //takes the input from the user
    if (SPEED == 1) { //rescale speed so motors have enough torque
        SPEED = 7;
    }
    else if (SPEED == 2) {
        SPEED = 8;
    }
    else if (SPEED == 3) {
        SPEED = 9;
    }

}
SPEED *= 400; //Scalar such that Max Speed ~4000
```

Figure 4: Speed and mode input based on validator pattern. .

A pivotal aspect of the code that we touched on was edge cases. In the code shown in Figure 4, we checked to see if the inputs for speed and mode were within the constraints we asked

for. If they weren't, we used a validator pattern to ensure that the user would be prompted until they imputed a valid value. We also took a user input from 1 to 3 and scaled it according to the motor speeds that would effectively move the car.

Design Process

An important step we used in the engineering design process was building a prototype. We had to build the car and connect the hardware correctly in order to ensure that the car would be as effective as possible based on our design parameters. We also had to plan out the code and compose it in an effective way. Another important step from the design process was testing and evaluating the prototype. We needed to make sure that the car would meet the specified criteria set by the instructor. We did this by running the newly written code and checking that the car would act as expected. Lastly, we needed to improve and redesign as needed. We did not expect our code to run perfectly the first time. We were constantly debugging our code and checking over our work.

Challenges

- We wanted to make our robot unique from other groups but seeing how we all had the same goal in mind this was a challenge. We overcame this by using solely our group's creativity to code so that our solutions would be unique and organically invented. This also meant that we needed to find our own solutions to problems that were unique to our group.
- Trying to keep to our timeline and get our goals done by certain dates was much more challenging than we anticipated. The code instructions looked deceivingly simple so we had a false sense of security. We overcame this challenge by working over Thanksgiving break and delegating tasks to each other.
- Writing our code in a consistent and easy to follow manner to make it intelligible to readers. We attempted to avoid adding extraneous lines in our code since we want it to do the operations requested in this project in the most clear and shortest way possible.
- Another challenge was getting our team all together at the same time. The week of Thanksgiving break we were missing members due to travel plans. After the break we were missing people since class had been canceled. We overcame this challenge by communicating outside of class and working diligently whenever we could all be in the same room at the same time. This also meant that we really needed to make our time in class as productive as possible.

Lessons for Future

- We did not think coding would be as time intensive as it was. We were not able to complete the code for Project 1 by the end of class on 11/18. We also weren't able to complete the code for Project 2 on 11/23 because we needed to catch up on Project 1. Next time we would try to be more thoughtful when it comes to planning out the timeline.
- We want to instead work together on the code instead of trying to divide and conquer so that we are most efficient as a team. When one of our group members was missing, we had to find ways to still be productive.
- Learning more advanced coding techniques. We completed our project based on the materials covered in ENGR 102 class and using other high-level sources to code may be helpful in this regard.

Entry for Resume

Electromechanical System Design Course Project

- Constructed Raspberry Pi circuit boards and software.
- Built a robot car and continuously made modifications to its design in order to accommodate new features.
- Wrote and modified existing code to achieve specific parameters.
- Worked together as a team and communicated effectively to accomplish a final goal.