Istanbul Medipol University
School of Engineering and Natural Sciences

Introduction to CoE and EEE – Fall 2019
Project Final Report

Team Members: Burak Gödek, Enes Uğuz, Hasan Şenyurt, Gülsüm İkbal Avşar, Ozan Halis İlter
Group Name: Saturn
Project Leader: Ozan Halis Ilter

Introduction, Blocks and Their Functionality

When looking at the block diagram, it is understood that the operations are collected in the CPU and these processes are transmitted to the car. CPU is the brain of the car; it makes computations to let the car stay in the middle of the road. Besides, the CPU sends commands to the output. Motors take that information and move the car forward. A certain code system is required to perform these operations, so the group has come together, and the racecourse of the autonomous car has been examined. Using this data, it was determined how the car will accelerate and decelerate. In addition, it was discussed how many cms before the vehicle should turn when the sensor detected lights and what speed the vehicle should have. After performing these steps, the code of the actions required in the project was written.

Each group member in the group expressed their opinion, and a joint decision was made. We have decided every member to focus on software because the hardware part is almost completed. We also developed two mathematical models.

Let us suppose X is the distance to the center of the road, R is the distance to the right lane line, L is the distance to the left lane line. X must be equal to (R-L) / 2. Our goal is to reduce X to 0. If X is less than 0 means that the vehicle should move in the left direction, and if X is greater than 0 means that the vehicle should move in the right direction. We can use the arithmetic mean of X and (R-L) / 2 If a problem occurs, it will not affect that much. Also, sometimes sensors can only show impossible values for just one second. We can set a limit to prevent this. For example, if the difference between the two values is higher than the small value, use the small value. It would save time for the sensors to return to normal.

It can be written an algorithm that increases the speed in a short time. For example, we can use $f(x)=x^2$ function. X is time and f(x) is the speed. Let us say 100 is the full speed and 1 is the lowest speed. Each second (or a smaller time unit) it will speed up exponentially until it reaches to the 10 because we set the full limit is 100. On the other hand, we can use the reverse of it to decrease the speed in a short while. For example, we can decrease the value of X in 10 seconds to stop the car.

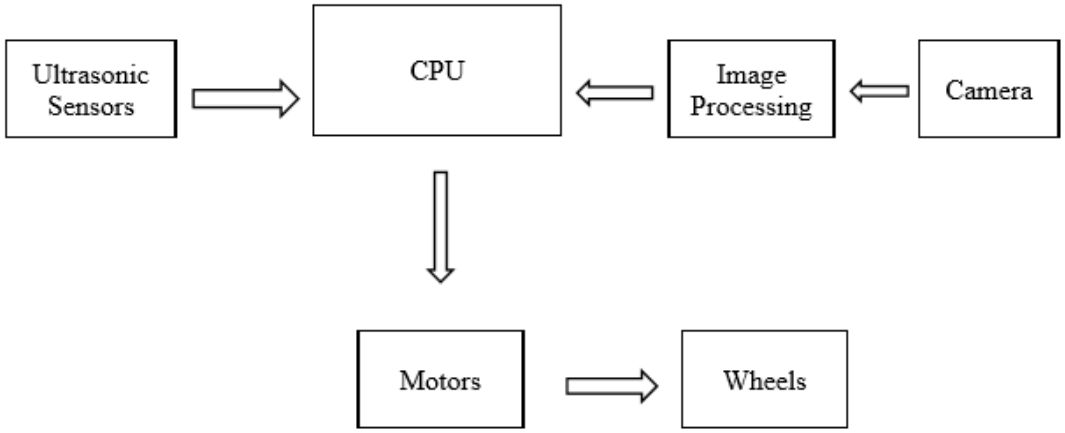

Figure -1: Block Diagram and Block Interfaces

There are two cases for the line follow. Firstly, if the car is placed middle of the road at the beginning of the race, the line follower sensor is got involved. For the second case, if the car is placed between the middle line and side of the road, the mathematical model which is mentioned above about line following is got involved. After the middle line is found using the mathematical model, the line following system will be maintained with the line sensor.

The modularity was implemented for each step of the autonomous car project process. The camera which is esp32 cam was activated by using Arduino. The ultrasonic sensor provides to measure the distance between car and obstacle, and the car stops before the collision. The code for this system was written by using Arduino. Also, the line following system was applied like mentioned above. After these steps were completed separately, every part was combined. Modularity was applied in this way.

The project software structure was completed by combining different techniques. In the beginning, the camera module which is esp 32 cam was arranged with Arduino Uno. The spectrum was seen anymore. After then, it was searched how the image processing would be. OpenCV library in Python and the spectrum which is received from esp 32 cam was combined for image processing.

Firstly, for each sign color, the RGB values were specified. It was required to determine the color of the sign. For instance, when the green object is shown to the camera, the RGB value of the object is intensive between the green RGB interval. The object color recognizing was completed in this way. After the camera recognizes the object's color, the if-else statements

were written. According to the color which was seen by the camera, the turning or stopping instructions were adjusted using mBlock application. When the code determined the color of the sign, the mBLock commands got involved. However, the mBlock application was running with the keyboard. The code which allows Python to press the keyboard itself was added. Thus, when the color was detected, the button was pressed depend on color's command so turning 90°. The wheel which is on the turning side was slowing down, and the turning was completed successfully.

The other issue was the timing of turning. After the color was detected and if-else statements were started to run, the determined intensive color's area was arranged in RGB as white which has RGB value 255. Also, the other color's area was arranged RGB value 0 which is black. The density of 255 RGB values was counted in this part of code. The increment of density of 255 RGB values means that the car is getting closer to the sign. After the density reached a certain rate, the mBLock commands were became involved.

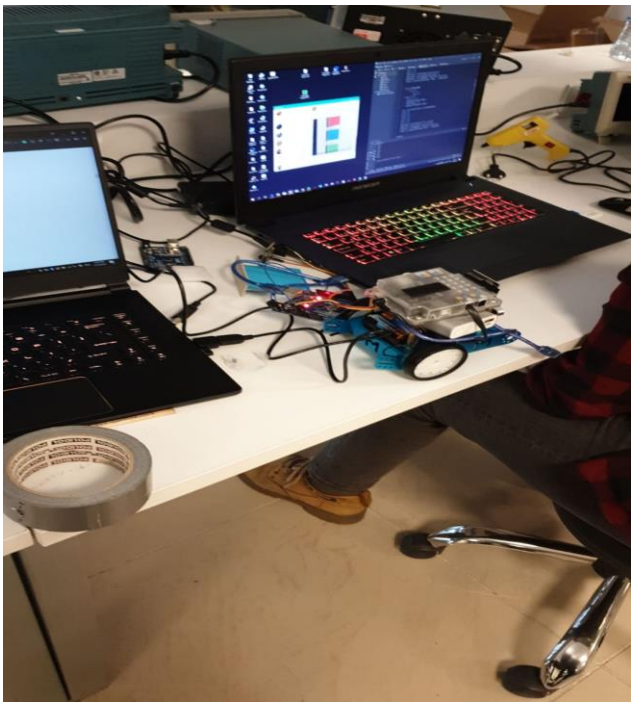

Figure -2: Color Recognition

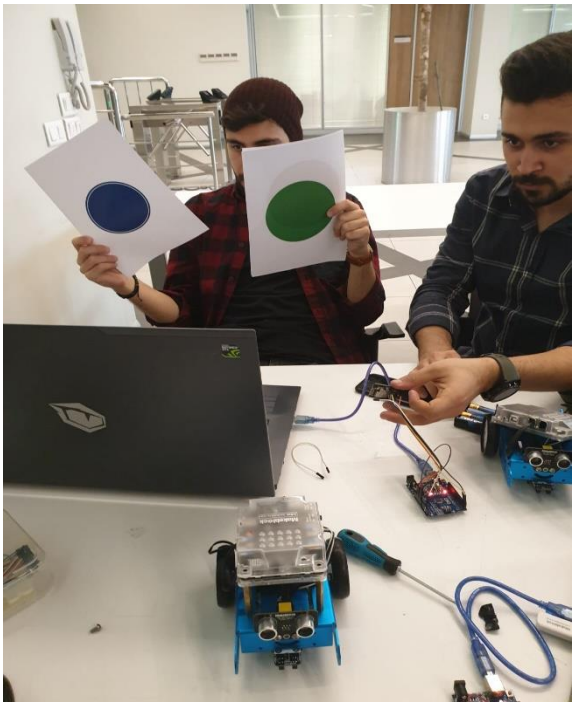

Figure -3: Color Recognition Test

Simulation/Demo Results

In the simulation, the autonomous car moves very smoothly. Turning, slowing down and speeding up timing is smooth and good. Setting the place of the autonomous car to the middle of the road in simulation is provided by side sensors. When the autonomous car must turn right or left, it stops, turns and speeds up again. Shortly, the turning system was like that and actually it is not realistic a little bit. After that, turning system was changed into smoother one. The car turns arc-shaped with 90 degrees. All simulation process was done with a computer camera. Finally, all computer camera information was transferred to the esp-32 camera.

Side sensors were removed and a line-follower sensor was used instead of. The line-follower sensor is under the car and it follows the specific lines.

In demo results, the road and curves were not like the car's logic. The car was prepared to move fast and turn arc-shaped. So, the speed of the car was decreased a bit and prepared for this road conditions. The autonomous car's logic was all correct. When it saw green, it turned right. When it saw blue, it turned left. Finally, when it saw red, it stopped. Generally, the demo result was successful.

The biggest challenge of the project was connecting the camera to autonomous car. Because the camera was given separately from the autonomous car and there is no cable-line between them. So, the problem was solved with mBlock program and wi-fi module of the car. There was another big challenge which that esp-32 did not work well. Its connection is not good with the computer and sometimes its fps dropped and froze. This was a hardware problem, not software. So, it was solved too.

Another challenge of the project was setting the place of the car to the middle of the road. The first idea which is using side-sensor did not work well because there were no walls at sides of the road. So, sensors cannot see anything and they cannot command the car that must stop. Instead of side-sensors, the line-follower sensor was used.

The group got full point on results because of the algorithm and teamwork. Every team member did their job well. The algorithm was written with the mid-report. The group leader got the responsibility well and gave the missions fair to members. The algorithm was the key of the success because there were a lot of problems that must be solved quickly. Time and information were limited, but the group did not give up and get the success.

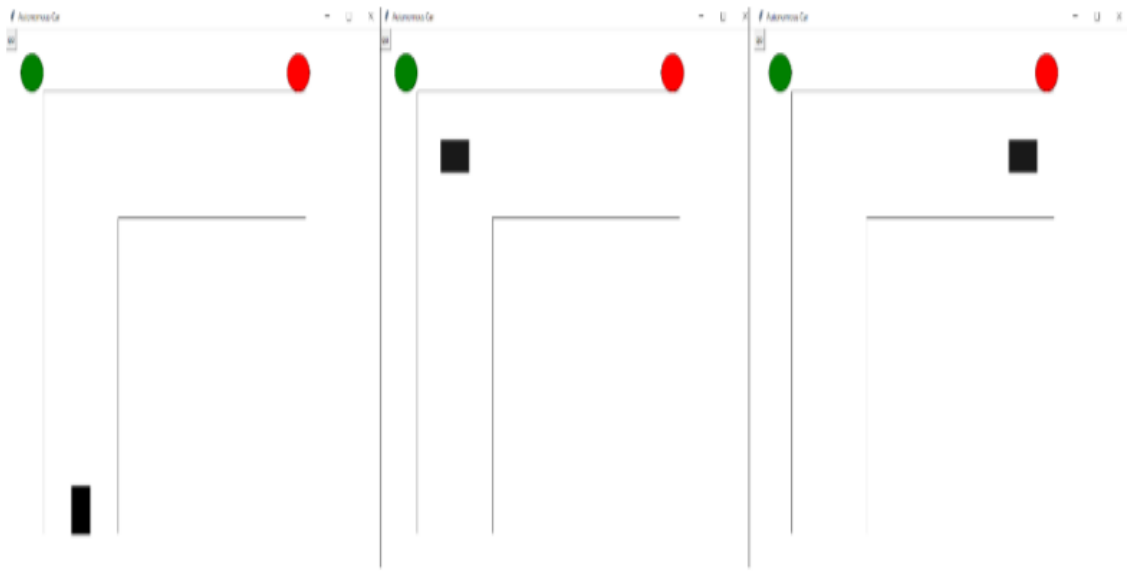

Figure -4: The Simulation Program Written in Python

Conclusion

In general, the hardware was obtained and the codes for moving the car, line follow, activating the camera by using Arduino and connecting the camera to the autonomous car by using the mBlock program were written. Algorithms to move the car were also written and the sensors were used. Simulations of the car moving, turning, speeding up and slowing down have been done. Shortly, this project includes the successes which are using robots, developing the algorithm, connecting different parts, learning the things that has been never seen and using time and skills very well.

To improve the design, put together the information gathered and start analyzing and optimizing core problems like the cost, strength, size, weight, reliability and performance. Also, a good way is to produce Prototypes to find the best solution for problems. Our design was good because we fit a lot of things in the car which are power bank, esp-32 camera and its Arduino card and batteries. The car's top side was raised and created space for camera and power bank. The exterior design was good but the car was heavy because of the power bank. Maybe some batteries can be replaced instead of the power bank. For the functional parts, instead of the esp-32 camera, telephone camera or some other cameras must be used absolutely because our biggest problems were because of the esp-32 camera.

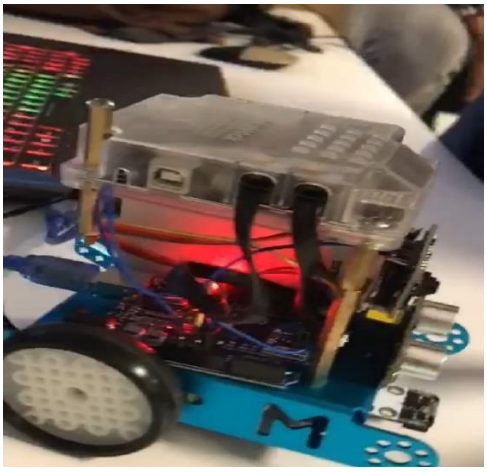

Figure -5: Exterior Design of Car (Right)

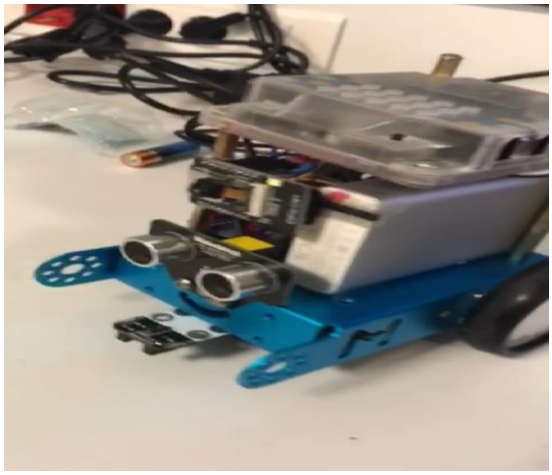

Figure -6: Exterior Design of Car (Left)

As engineering candidates, this project helped us to learn how to solve problems in a structured way. It also helped us to learn how to use Arduino and the mBlock programs to move the car, turn the car, activate the camera, connect the camera to the autonomous car or to set clear priorities. As a team it also helps learning how to use the knowledge and skills each team member has, to execute the project effectively and efficiently. This project helped to learn how to clarify roles for each of the team members and work as a team more effectively. It also helped us to learn how to communicate and listen to the opinions, thoughts and ideas of each member and to make decisions.

(The video link for the design of the Car : https://youtu.be/jYRtgT2dxTg)