Hacettepe University

Department of Computer Engineering

BBM434 - Embedded Systems Laboratory

Project Report

Meltem Tokgöz
21527381
Fatmanur Turhan
21527429

June 2019

Contents

1 Introduction

1.1 Overview

In our project, we designed a car that follows the line. Our main goal was to follow the most complex paths quickly and accurately to reach the end point of our car and we achieved this goal. The car detects the path with sensors. Thus, the road is completed without deviating from the line. If the car has successfully reached the end of the road, our system has been successful. This project helped us to understand the technique of driverless vehicles, the logic of my sensors and how the dc motor works. It also helped us understand the logic of technological advances to prevent car accidents.

1.2 Project Features

In our car following line, there are 3 sections as sensor section, driver section and control section we also used 9v battery and powerbank as power source.

1.2.1 Used Equipments

The equipment used in our project are as follows.

- Texas Tiva TM4C Microcentroller
- DC Motor x2
- 3 x Infrared Sensor and Sensor Modüle
- L298N Motor Driver Modüle
- Jumper Cable
- 5v Powerbank
- 9v Duracell Battery
- 9v Battery Bed
- Chasis, wheel, roller coaster
- Screw, Nut
- Drunken-headed wheel
- Double-sided tape
- Road platform

1.2.2 System Logic

The line tracker works as follows: We use the Ir sensor to detect the black line and then send the signal to the TM4C123 Launchpad. The TM4C123 Launchpad then runs the engine with motor driver module according to the output of the sensors.

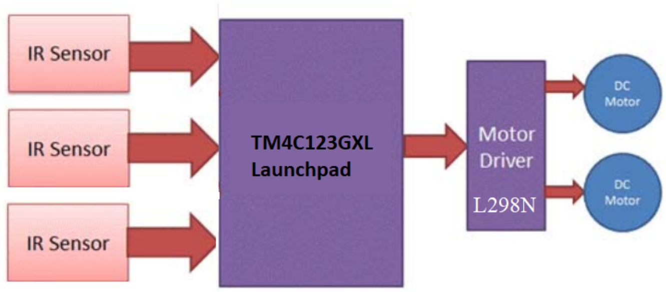

Figure 1: All System Schema

Here in this project we are using three IR sensor modules namely left sensor, right sensor, mid sensor.If the right, left and middle sensors, so all sensors detect black light, our car goes straight ahead. Conversely, if the three sensors detect white light, the car stops and does not go in any direction.

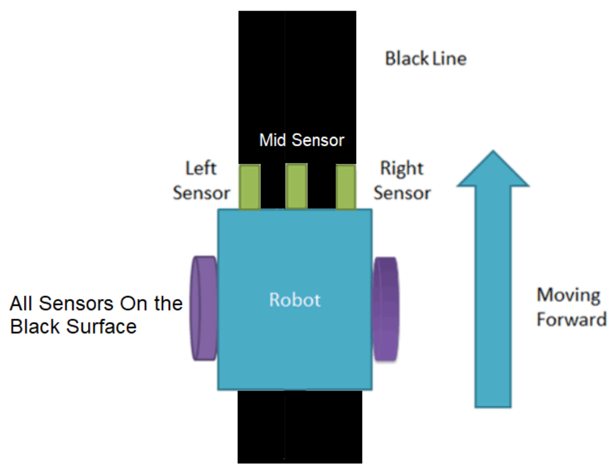

Figure 2: Moving Forward

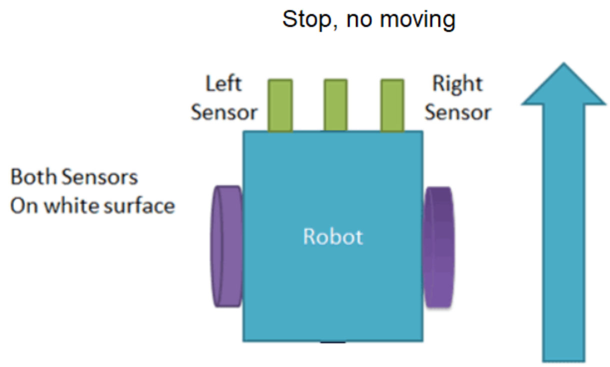

Figure 3: Stop Not Moving

If the two sensors detect black and one sensor detect white, it deviates slightly from the road. To fix this , the motor in the opposite direction of the white sensing sensor is moved and provided to follow the black line.

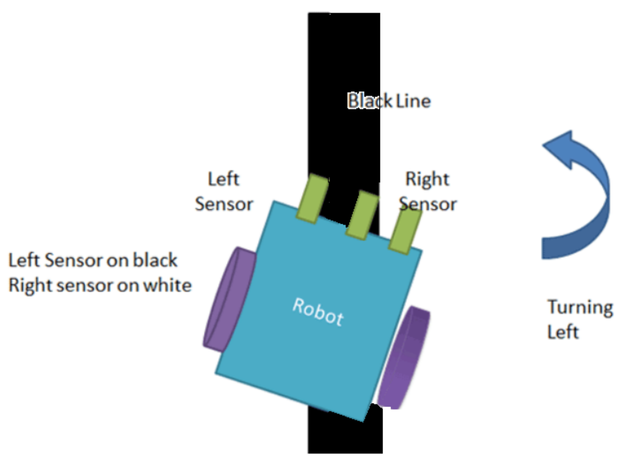

Figure 4: Left Moving

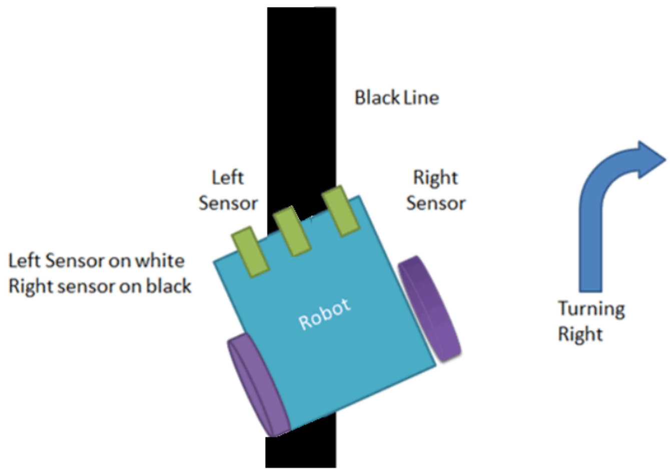

Figure 5: Right Moving

If two sensors detect white, and one sensor detects black, it is highly deviated from the road. To correct this, the motor is more moved further in the opposite direction of the white detection sensor and allowed to follow the black line.

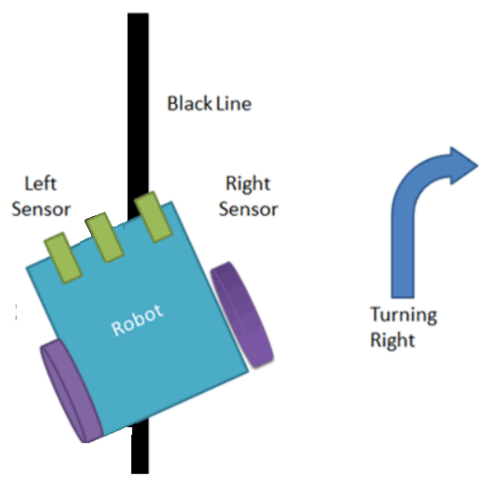

Figure 6: Right Moving

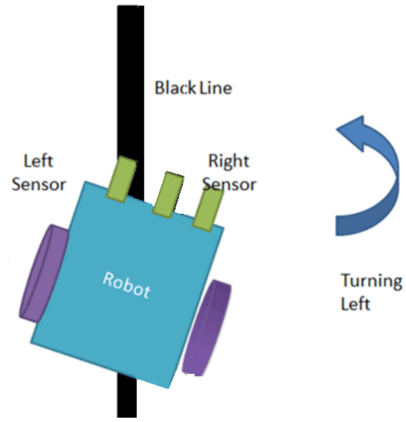

Figure 7: Left Moving

The general working logic of our system is in this way. Now let's examine the software and hardware parts in more detail.

2 Organizations System

Our project consists of two parts: hardware system and software system.

2.1 Hardware System

The hardware system consists of 3 main parts. Firstly, the sensor section is input, the second is the microcontroller section where the information received is evaluated, and the last is the motor section that moves according to the output. Now let's examine them one by one.

2.1.1 IR Sensor Modules

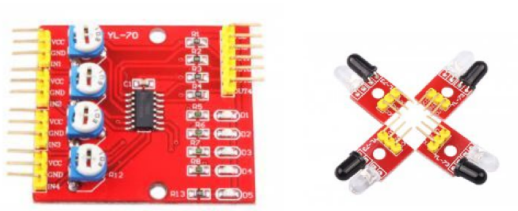

Figure 8: IR Sensor

The 4-track sensor set contains 4 infrared transceiver modules and it is possible to check whether there is a line on the ground or not.We chose to use 3 of the sensors here. Sensitivity and distance adjustment of all sensors can be made from the potentiometers on the control board.

The connection between the control board and the sensors is made by jumper cables. Thanks to the fixing screws on the sensors, the sensors can be used upright or horizontally if desired. We preferred to use upright in our project. There are OUT pins and feed and ground pins for outputs from 3 sensors on the control board. There is also an LED on the control board for each output.Operating Voltage of Sensors: 3,3-5V.

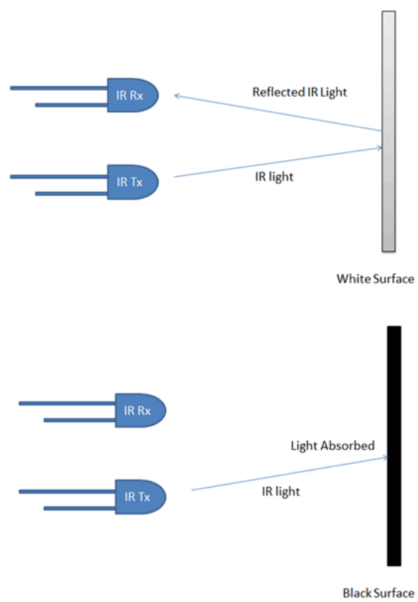

Figure 9: IR Sensor Work Logic

Concept of line follower is related to light. We using the behaviour of light atblack and white surface. When light fall on a white surface it will almost fullreflects and in case of black surface light is absorbed by black surface. Thisexplained behaviour of light is used in this line follower robot.Line follower robot we have used IR Transmitters and IR receivers also calledphoto diodes. They are used for sending and receiving light. IR transmitsinfrared lights. When infrared rays falls on white surface, it's reflected back andcaught by photodiodes which generates some voltage changes. When IR lightfalls on a black surface, light is absorb by the black surface and no rays arereflected back, thus photo diode does not receive any light or rays.

The sensor circuit is as follows.The information received from the sensor is sent to the PB0 PB1 and PB2 pins of the microcontroller via the sensor board.

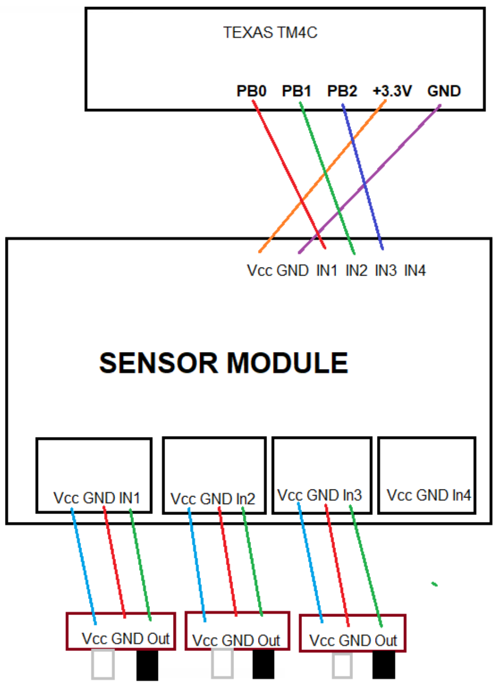

Figure 10: IR Sensor Circuit

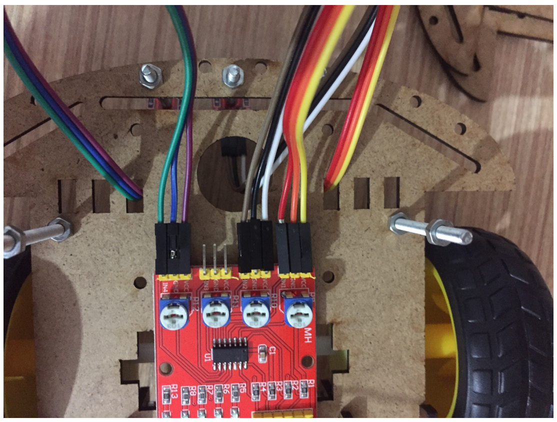

Figure 11: IR Sensor used in Our Project

How the information from the sensor to the microcontroller is processed is explained in detail in the sensor code section of the software system.

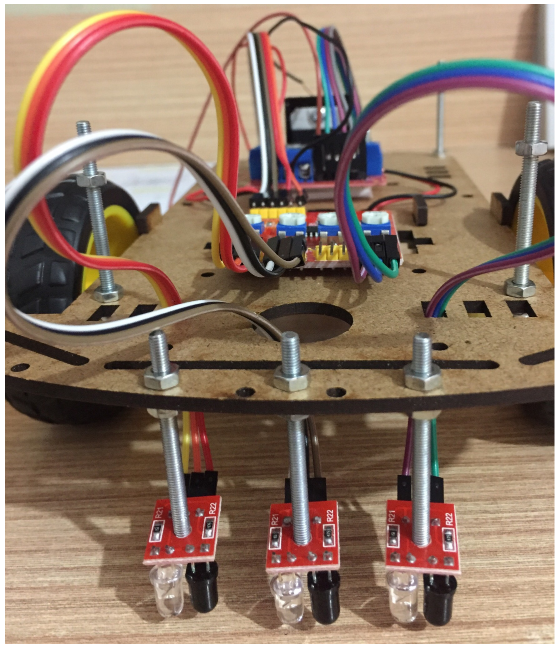

Figure 12: IR Sensor used in Our Project

2.1.2 Motor Driver Module

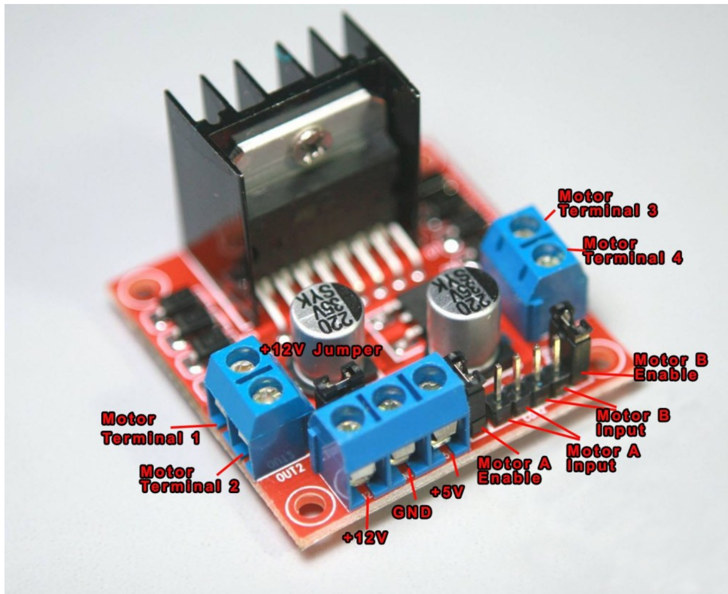

Figure 13: L298N Motor Driver Module

- On the left side, we use OUT1 and OUT2 to connect the DC motor, and on the right side the OUT3 and OUT4 port for the DC motor, from here we connect with DC motors.

- ENA and ENB activate pins by connecting ENA to high or 5V; Activates the OUT1 and OUT2 ports. If we connect the ENA pin to low or ground (ie 0), it will disable OUT1 and OUT2. Similarly, it applies to ENB and OUT3 and OUT4.

- The pins between IN1 and IN4 pins are the input pins that we connect to the texas microcontroller. We have connected to pins PA3, PA4, PA5 and PA6 respectively.Thus, we can control the direction of the motor according to the information we receive from the sensor through these pins.

- When we want to reverse the direction of rotation of the motor, it is enough for IN3 and IN4 to reverse the polarity.The same goes for IN1 and IN2.

- By applying PWM signal to ENA and ENB, we are able to control the speed of the motors. We have connected the ENA port to the texas microcontroller PA2 and the ENB port to the PA7 pin.

- We fed the card with a 9v voltage from the battery.We connected the 5V terminal to the Vbus on the texas microcontroller to provide power.

The movement of the motors according to the values of the pins of the motor driver is as follows.

Motor A truth table			
ENA	**IN1**	**IN2**	**Description**
0	N/A	N/A	Motor A is off
1	0	0	Motor A is stopped (brakes)
1	0	1	Motor A is on and turning backwards
1	1	0	Motor A is on and turning forwards
1	1	1	Motor A is stopped (brakes)

Motor B truth table			
ENB	**IN3**	**IN4**	**Description**
0	N/A	N/A	Motor B is off
1	0	0	Motor B is stopped (brakes)
1	0	1	Motor B is on and turning backwards
1	1	0	Motor B is on and turning forwards
1	1	1	Motor B is stopped (brakes)

Figure 14: DC Motors Status

How the information from the sensor to the microcontroller is processed is explained in detail in the motor code section of the software system.

The engine circuit is as follows. The information sent to pins PA2, PA3, PA4, PA5, PA6, PA7 of the microcontroller is transferred to the DC motors via the motor driver board and the direction or the operation of the motor is determined.

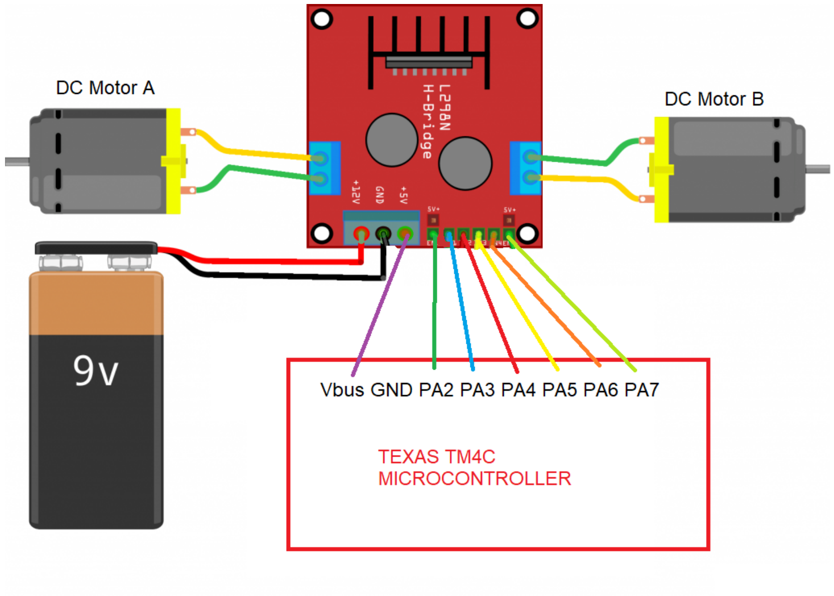

Figure 15: L298N Motor Driver Module

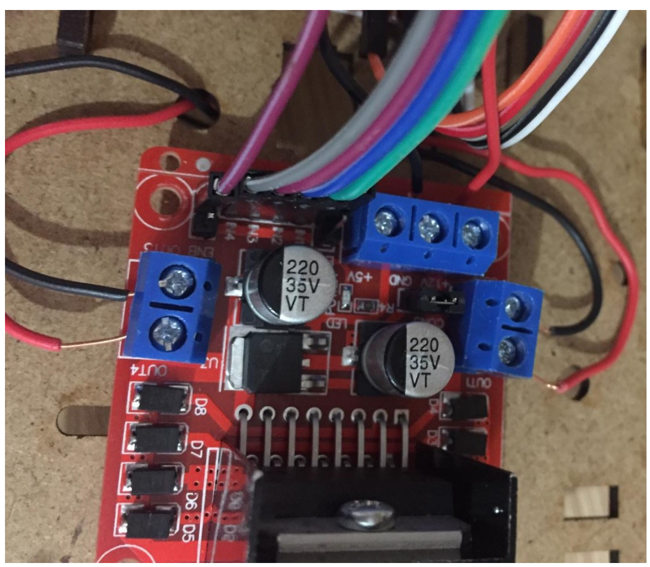

Figure 16: L298N Motor Driver Module

2.1.3 TM4C123GXL LaunchPad from Texas Instruments

The TM4C from the Texas Tiva c series is used as microcontroller. We load the code we write and control the hardware according to this code.The first thing is the brain of the robot, which is the Tiva Microcontroller. This is what will be programmed in order to make all the components work together.TM4C123 Launchpad is used for controlling the whole process of line follower robot. The outputs of comparators are connected to digital pins of TM4C123Launchpad. TM4C123 Launchpad reads these signals and send commands to driver circuit to drive line follower.

The pins A 2-7 and for the Vbus motor B0-4, Vcc, Gnd pins are used for the sensor of the microcontroller.We give power to microcontroller using powerbank, powerbank output is 5v.

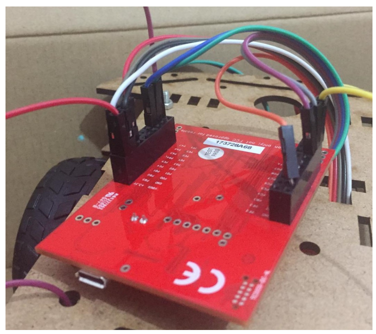

Figure 17: TM4C123GXL LaunchPad from Texas Instruments

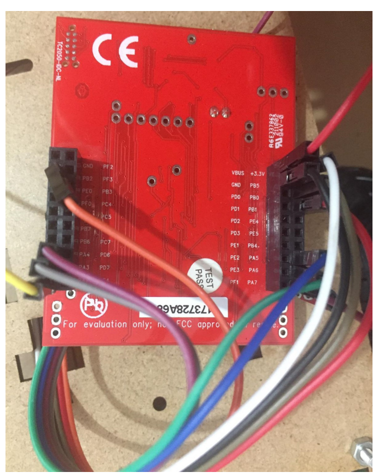

Figure 18: TM4C123GXL LaunchPad from Texas Instruments

2.1.4 Hardware Circuit

All our circuit diagrams and connections are shown in the figure below.

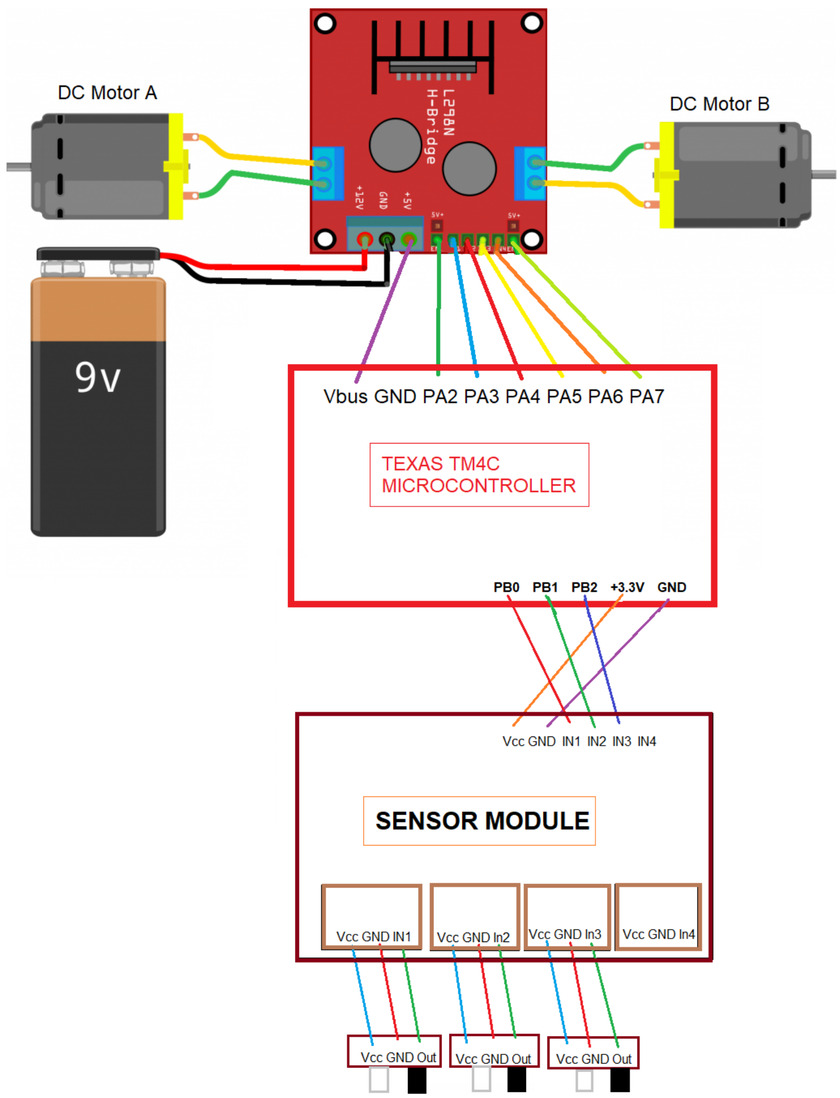

Figure 19: System Hardware Circuit

2.2 Software System

As explained in the Hardware section, we used 2 DC motors and 3 sensors in this project. In the software phase of the project, information from the sensors was read as input value and commands were sent to the motors according to this information and microcontroller was used to get information from the sensor.

2.2.1 Sensor Code

The B port of the microcontroller was used to record the information from the sensors. The PB0 pin holds information from the right sensor, the PB1 pin holds information from the middle sensor, and the PB2 pin holds information from the left sensor. If a sensor is on a black floor, it records pin 1 and a white floor.

There are 6 situations for sensors according to our platform where we will operate our robot. These ;

1. All sensors see the black floor: robot must forward to straight.

The PB0, PB1 and PB2 pins all hold value of 1.

2. The left sensor and the middle sensor see the black floor, the right white floor:

PB0 pin holds 0, PB1 and PB2 pins hold value of 1.

3. The left sensor sees the white floor, the middle sensor and the right sensor see the black floor:

PB0 pin and PB1 pin hold value of 1, PB2 pin holds value of 0.

4. The left sensor sees the black floor, the right sensor and the middle sensor see the white floor:

PB0 pin and PB1 pin hold value of 0, PB2 pin holds value of 0.

5. The left sensor and the middle sensor see the white floor, the right sensor sees the white floor:

PB0 pin holds value of 1, PB1 and PB2 pins hold value of 0.

6. All sensors see the white floor:

PB0, PB1 and PB2 pins all hold value of 0.

```
#define BBB  0x07
#define BBW  0x06
#define WBB  0x03
#define BWW  0x04
#define WWB  0x01
#define WWW  0x00
```

Figure 20: Define Direction Pins

2.2.2 Motor Code

Direction commands are sent to the motor driver with information from the sensor. The A port of our microcontroller was used to give these commands to the motor drive. There are two motors, right and left. The A pins of the motor driver for the left motor and B pins of the motor driver for the right motor were used.

For the left-hand motor, PA2 pin represents enableA, PA3 pin backwardA, and PA4 pin forwardA. For the right motor, the PA5 pin represents forwardB, the PA6 pin backwardB, and the PA7 pin enableB.

With this information, FORWARD, LEFT, RIGHT and STOP values are calculated as follows:

```
// B --> right motor
// A --> left motor
//pin7=enableB, pin6=backwardB, pin5=forwardB, pin4=forwardA, pin3=backwardA, pin2=enableA
#define FORWARD  0xB4 //1011 0100
#define LEFT  0xAC   //1010 1100
#define RIGHT  0xD4   //1101 0100
#define STOP  0x00   //0000 0000
```

Figure 21: Motor Code

If the 3 sensors are on the black floor, both motors will go forward, so the moveForward () function will be called.

If only the right sensor is on the white floor or the right and middle sensor are on the white floor, the robot will turn to the left. For the turn left, the right motor must be go forward and the left motor must be go backward. The moveLeft () function must be called for this.

If only the left sensor is on the white floor or the left and middle sensor are on the white floor, the robot will turn to the right. For the turn right, the left motor must be go forward and the right motor must be go backward. The moveRight () function must be called for this.

If all sensors are on a white floor, both motors must stop. The enable pins and the direction pins must have a value of 0. The Stop () function must be called for this.

```c
/*********************** MOTOR FUNCTIONS ************************/
void moveForward (void){
  GPIO_PORTA_DATA_R = FORWARD;
}
void moveLeft (void){
  GPIO_PORTA_DATA_R = LEFT;
}
void moveRight (void) {
 GPIO_PORTA_DATA_R = RIGHT;
}
void Stop (void){
 GPIO_PORTA_DATA_R = STOP;
}
```

Figure 22: Motor Functions

2.2.3 Main Program

Continuous state control is performed in the main function with a while infinite loop.

If the information from the sensor is BBB (black-black-black), moveForward() should be called.

If the information from the sensor is BBW (black-black-white), moveLeft() should be called.

If the information from the sensor is WBB (white-black-black), moveRight() should be called.

If the information from the sensor is BWW (black-white-white), moveLeft() should be called.

If the information from the sensor is WWB (white-white-black), moveRight() should be called.

If the information from the sensor is WWW (white-white-white), the Stop() function should be called.

```c
int main ()
{
  volatile unsigned delay_clk;
  SYSCTL_RCGCGPIO_R |= 0x03;  // PORTA_CLK_EN + PORTB_CLK_EN
  delay_clk         = SYSCTL_RCGCGPIO_R;

  GPIO_PortA_Init();
  GPIO_PortB_Init();

  while(1)
  {
    if(GPIO_PORTB_DATA_R == BBB){         //forward --> black-black-black
        moveForward();
    }
    else if(GPIO_PORTB_DATA_R == BBW ){   //left  --> black-black-white
        moveLeft();
    }
    else if(GPIO_PORTB_DATA_R == WBB ){   //right --> white-black-black
        moveRight();
    }
    else if(GPIO_PORTB_DATA_R == BWW ){   //left --> black-white-white
        moveLeft();
    }
    else if(GPIO_PORTB_DATA_R == WWB ){   //right --> white-white-black
        moveRight();
    }
    else if(GPIO_PORTB_DATA_R == WWW ){   //stop --> white-white-white
        Stop();
    }
  }
}
```

Figure 23: Main Function

2.2.4 Software Logical Schema

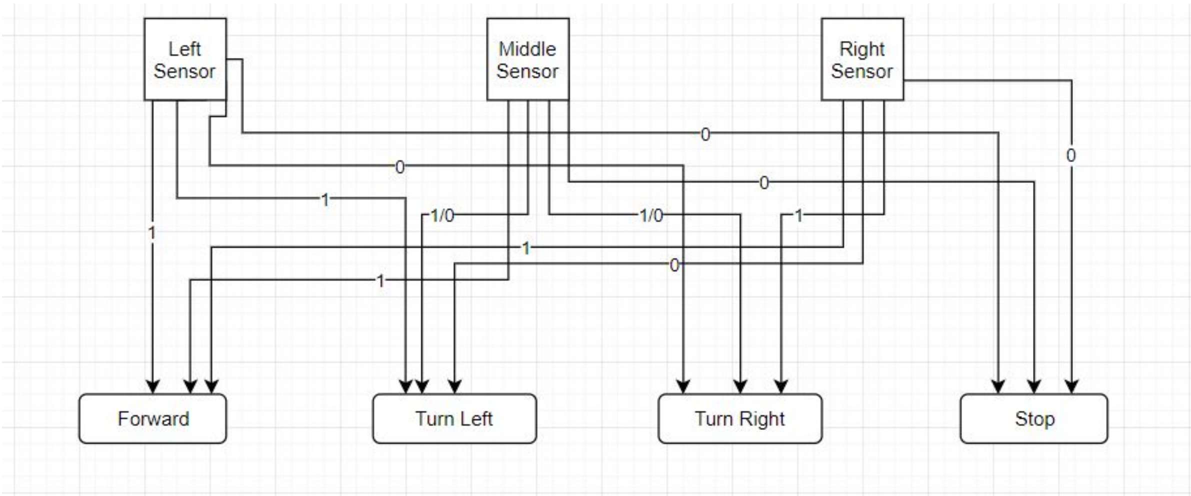

Figure 24: Software Logical Schema

3 Project Video and Image

3.1 Youtube Link

You can watch the video of our project by clicking this link (demo).

3.2 Images

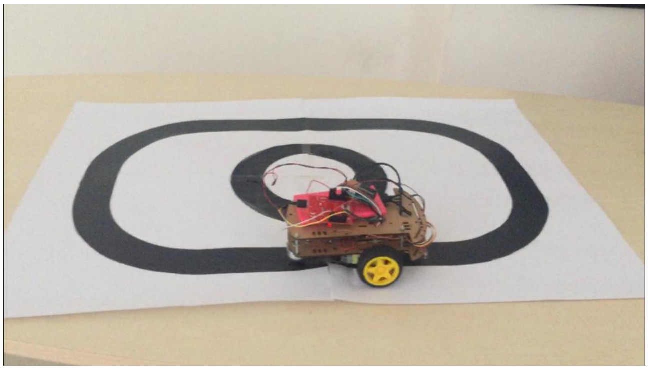

Figure 25: Line Following Car

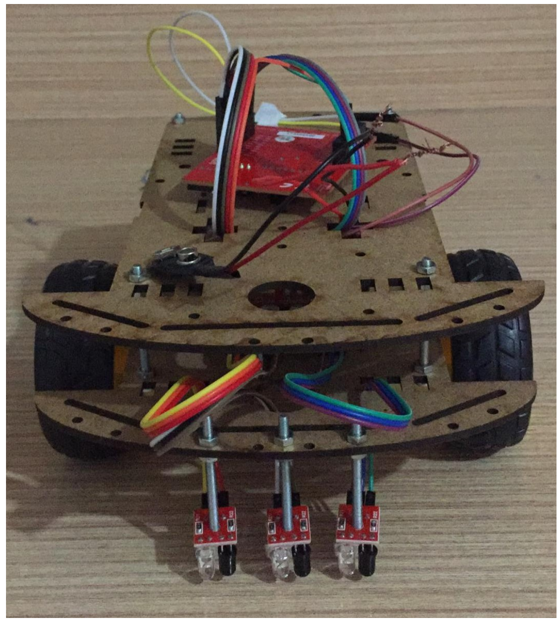

Figure 26: Line Following Car

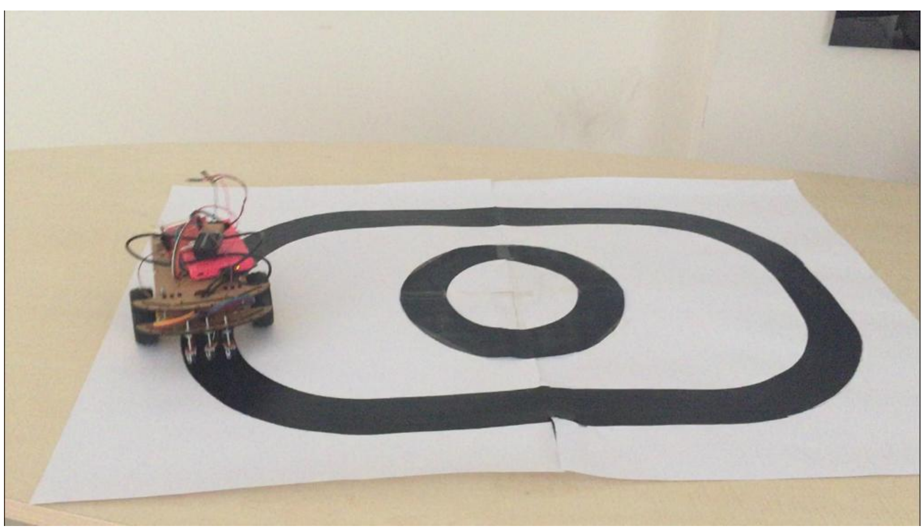

Figure 27: Line Following Car

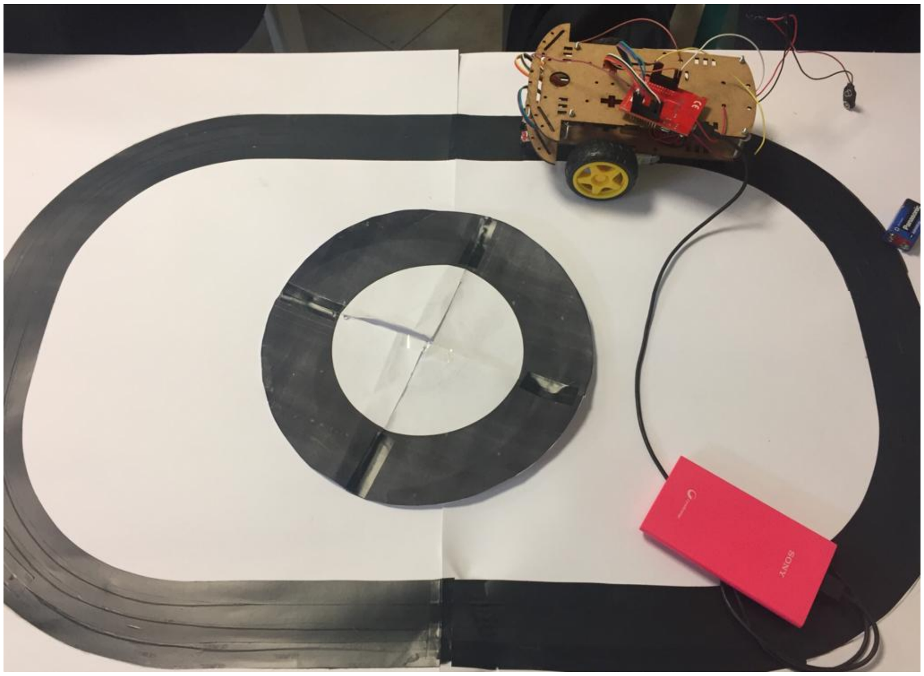

Figure 28: Line Following Car

4 Extras

4.1 Responsibilities

In all construction phases of the project, the group members thought together, decided together and implemented them together. Firstly, sample projects were researched and it was decided to make a robot following the line. As a result of the researches, the necessary materials were determined and taken.

When the project was started, two group members were met at every free time. Firstly, hardware planning was done. Then, the motors were started first and then the sensors were added to the circuit and the software stages were completed. The platform was prepared for the necessary tests. In this platform, the movements of the robot were tested and necessary arrangements were made and the robot was made ready.

4.2 Budget

- Texas Tiva TM4C Microcentroller 140 TL
- DC Motor x2 16 TL
- 3 x Infrared Sensor and Sensor Modüle 23 TL

- L298N Motor Driver Modüle 13 TL
- Jumper Cabled 4 TL
- 5v Powerbank
- 9v Duracell Battery 20 TL
- 9v Battery Bed 2 TL
- Chasis, wheel, roller coaster 3 TL
- Screw, nut x30 20 TL
- Drunken-headed wheel 5 TL
- Double-sided tape 4 TL
- Road platform 8 TL