EE 101
Final Project: Line Following Car

Zack He
College of Engineering and Mathematical Sciences
The University of Vermont

May 2, 2019

Contents

1 Project description

1.1 Goals

In the manufacturing companies especially the large size products like furniture, the efficiency of the production flow is very important. Moving products by hands during the production line is hazardous and inefficient. Therefore, the line following robot car can be introduced in this condition.

The objective of this project is to create a prototype in the electrical perspective of the real cart. The car needs to move along the track, stop at emergency situations and be controlled from cellphone.

1.2 Functions

In this design, we are going to use a black tape as the track. The robot car needs to move along the tape via two IR sensors at both sides. The IR sensors detect the brightness and generate the analog input which will control the motion of motor.

At the front and back of the car, there are two ultrasonic sensors to detect the distance from other objectives. The cart needs to stop when the distance is less than 10 cm. At that time, LED will blink and a warning message will be sent to user's cellphone.

An LCD is attached to display the motion status of the cart. An button is also attached. The motion status will be sent to cellphone when it is pressed.

Bluetooth technology is used in this project. Cart will go or stop via the command from cellphone. An emergency stop is also added to protect car.

1.3 Input and Output

Input:

- E-Stop (On-off Switch)
- button
- IR sensor X2
- Ultrasonic sensor X2
- Remote (Cellphone via Bluetooth)

Output:

- Motor X2
- LCD
- LED
- Remote (Cellphone via Bluetooth)

1.4 Operation Description

When the power is on, the cart will not move until the command is made from user's cellphone. Only two commands are valid by remote input. "G" is go and "S" is stop. When "G" is typed, the cart should be able to move around the black track automatically. Motion information will display on the LCD or cellphone by pressing the button. When the car is too close to an objective, it will stop until the objective is moved away.

2 Project realization

2.1 Schematics

The schematics and real picture of the cart are shown as Fig 1 and Fig 2.

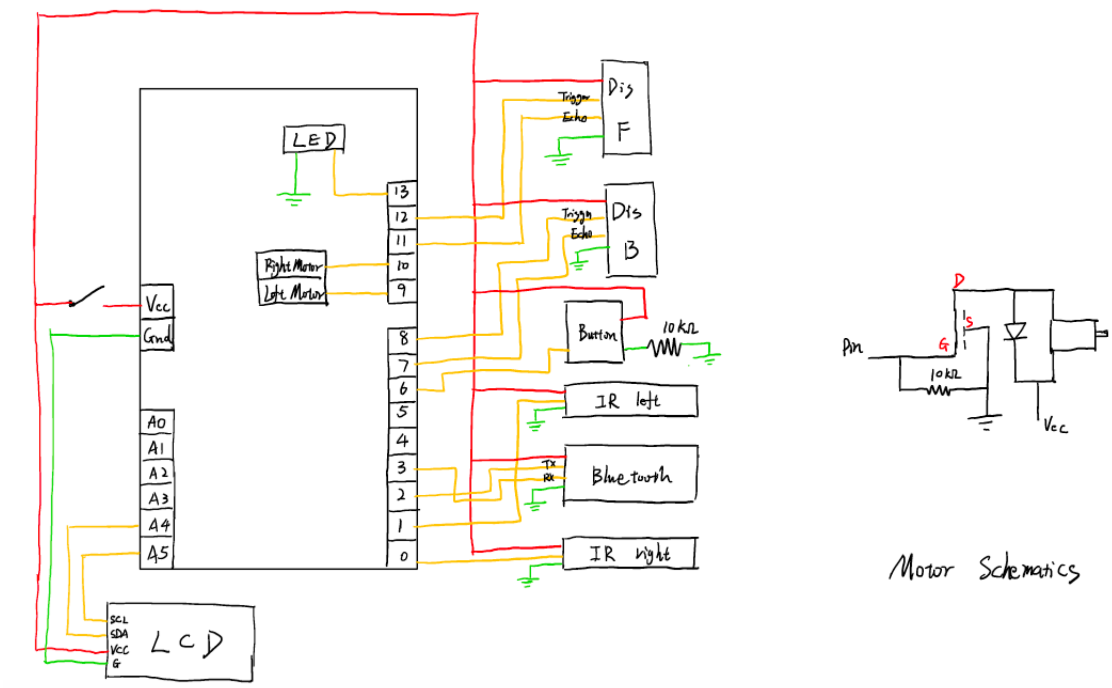

Figure 1: Schematics of Circuit

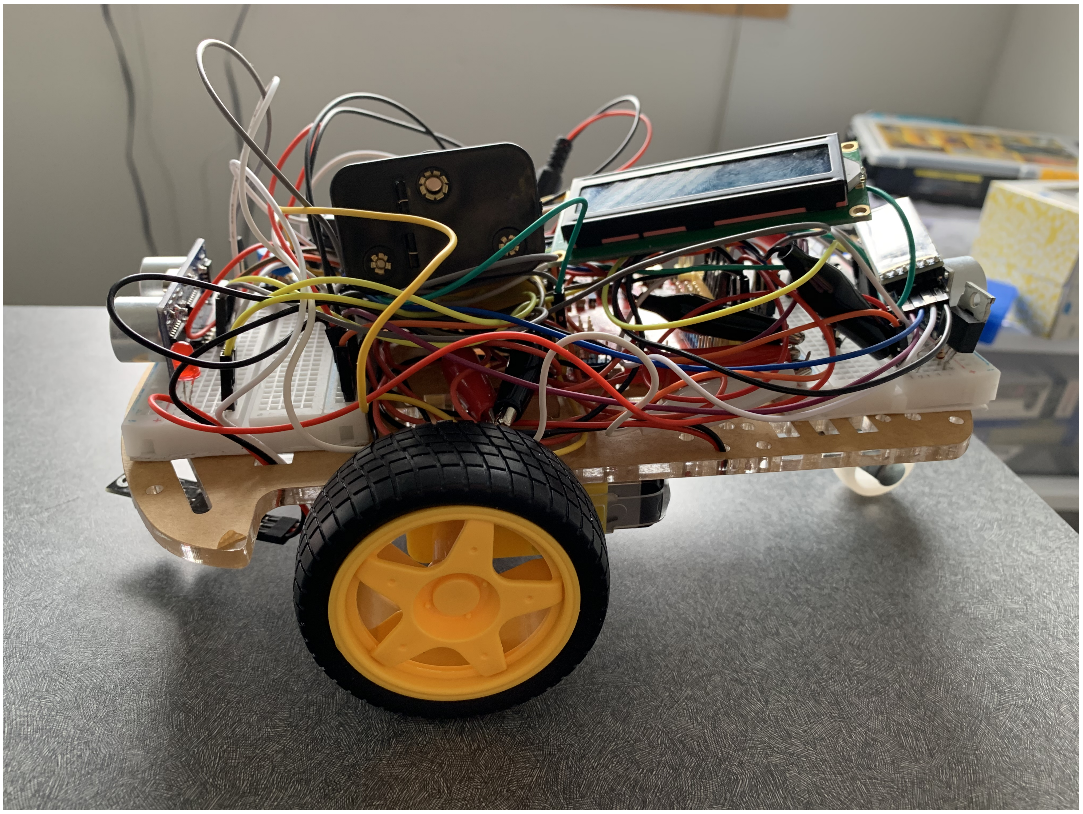

Figure 2: Real Picture of Car

2.2 Arduino Code

{\rtf1\ansi\ansicpg1252\cocoartf1671\cocoasubrtf400 {\fonttbl\f0\fswiss\fcharset0 Helvetica;} {\colortbl;\red255\green255\blue255;} {*\expandedcolortbl;;} \margl1440\margr1440\vieww10800\viewh8400\viewkind0 \pard\tx720\tx1440\tx2160\tx2880\tx3600\tx4320\tx5040\tx5760\tx6480\tx7200\tx7920\tx8640\pardirnatural\partightenfactor0 \f0\fs24 \cf0

```
\
#include <NewPing.h>\
#include <LiquidCrystal_I2C.h>\
#include <Wire.h>\
#include <SoftwareSerial.h>\
#include <BLESerial.h>\
#include <EEPROM.h>\
\
#define IRright 0 //IR sensors\
#define IRleft 4\
\
#define LED 13\
\
#define RX 2 //Bluetooth\
#define TX 3\
\
#define DistFront_Trigger 12 //Distance sensor\
#define DistFront_Echo 11\
#define DistBack_Trigger 8\
#define DistBack_Echo 7\
\
#define RightMotor 10 //motor\
#define LeftMotor 9\
\
#define buttonPin 6\
\
int MAX_DISTANCE = 200;\
int addr = 0;\
\
NewPing sonarF(DistFront_Trigger, DistFront_Echo, MAX_DISTANCE);\
NewPing sonarB(DistBack_Trigger, DistBack_Echo, MAX_DISTANCE);\
\
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE); \
SoftwareSerial BLE(RX,TX);\
\
\
void setup()\{\
  Serial.begin(9600);\
  pinMode(LED, OUTPUT);\
  BLE.begin(9600);\
  lcd.begin(16,1);\
  lcd.backlight();\
\}\
\
void loop()\{\
  int IRR = digitalRead(IRright);\
  int IRL = digitalRead(IRleft);\
  \
  float echotimeF = sonarF.ping();\
  float distanceF = sonarF.convert_cm(echotimeF);\
  float echotimeB = sonarB.ping();\
  float distanceB = sonarB.convert_cm(echotimeB);\
\
  if(BLE.available())\{\
    char BLEinput = BLE.read();\
    if(BLEinput == 'G')\{\
      EEPROM.write(addr,BLEinput);\
    \}\
    if(BLEinput == 'S')\{\
      EEPROM.write(addr,BLEinput);\
    \}\
  \}\
\
  int button = digitalRead(buttonPin);\
\
  char Next = EEPROM.read(addr);\
  if(Next =='G')\{\
      if(((distanceF<10)&&(distanceF>0))||((distanceB<10)&&(distanceB>0)))\{\
        motion('S');\
        lcd.clear();\
        lcd.setCursor(4,0);\
        lcd.print("WARNING!");\
        BLE.println("WARNING!");\
        digitalWrite(LED,HIGH);\
        delay(50);\
        digitalWrite(LED,LOW);\
        delay(50);\
      \} \
      else\{\
        if((IRR==0)&&(IRL==0))\{\
          motion('F');\
```

```
          lcd.clear();\
          lcd.setCursor(0,0);\
          lcd.print("Move");\
          lcd.setCursor(5,0);\
          lcd.print("Farward");\
          if(button==1)\{\
            BLE.println("The cart is moving forward!");\
          \}\
        \}\
        if((IRR==1)&&(IRL==0))\{\
          motion('R');\
          lcd.clear();\
          lcd.setCursor(0,0);\
          lcd.print("Turning");\
          lcd.setCursor(8,0);\
          lcd.print("Right");\
          if(button==1)\{\
            BLE.println("The cart is turning right!");\
          \}\
        \}\
        if((IRR==0)&&(IRL==1))\{\
          motion('L');\
          lcd.clear();\
          lcd.setCursor(0,0);\
          lcd.print("Turning");\
          lcd.setCursor(8,0);\
          lcd.print("Left");\
          if(button==1)\{\
            BLE.println("The cart is turning left!");\
          \}\
        \}\
        if((IRR==1)&&(IRL==1))\{\
          motion('S');\
          lcd.clear();\
          lcd.setCursor(6,0);\
          lcd.print("Stop");\
          if(button==1)\{\
            BLE.println("The cart is stopping!");\
          \}\
        \}\
      \}\
    \}\
   if(Next =='S')\{\
    motion('S');\
    lcd.clear();\
    lcd.setCursor(6,0);\
    lcd.print("Stop");\
    if(button==1)\{\
            BLE.println("The cart is stopping!");\
          \}\
   \}\
\
\}\
    \
\
void motion(char m)\{\
  switch(m)\{\
    case 'F':analogWrite(RightMotor,75);analogWrite(LeftMotor,75);break;\
    case 'L':analogWrite(RightMotor,150);analogWrite(LeftMotor,LOW);break;\
    case 'R':analogWrite(RightMotor,LOW);analogWrite(LeftMotor,150);break;\
    case 'S':analogWrite(RightMotor,LOW);analogWrite(LeftMotor,LOW);break;\
  \}\
\}\
\

}
```

2.3 SW List

The software used for coding is Arduino. The libraries used list in following.

- NewPing.h
- LiquidCrystal_I2C.h
- Wire.h
- SoftwareSerial.h
- BLESerial.h
- EEPROM.h

2.4 Project Plan vs Actual

First test is to figure out the reliability of ultrasonic sensor. We need to figure out what is the real distance when the cart stops. We can place an objective in front of the cart. The cart should stop automatically. We measure the distance between the distance sensor and the objective. Compare this value to the value we set. Calculate the percent error.

$$PercentError = \frac{|d_{actual} - d_{theoretical}|}{d_{theoretical}} \tag{1}$$

We repeat this measurement for 10 times and find the average percent error. The design is considered to pass the test if the average percent error is less than %10.

This test performed pretty well and all results are recorded. Percent error is calculated as well.

Second test is to find the possibility that car can move through the track with 30 cm radius. We placed car on the track and let it go through the track. If it passes the turn, it is considered "Pass". If it is out of the track, it is "Fail". We recorded the results for 10 trials and calculate the possibility. The design is considered to pass the test if the possibility is larger than 80%.

Initially, we wanted to find the minimum turning radius of the car. For this test, we can use tape to make several circle with different radius. We place the cart on these track to find the minimum turning radius that the cart still running. However, it is hard to make a circle with desired radius by tapes. It requires a lot of space and resources. Therefore, we decided to do the test for possibility of one specific radius instead of finding minimum radius.

2.5 Test Results

The method of first test was discussed above and shown as Fig 3. The result for this one is shown as Table 1

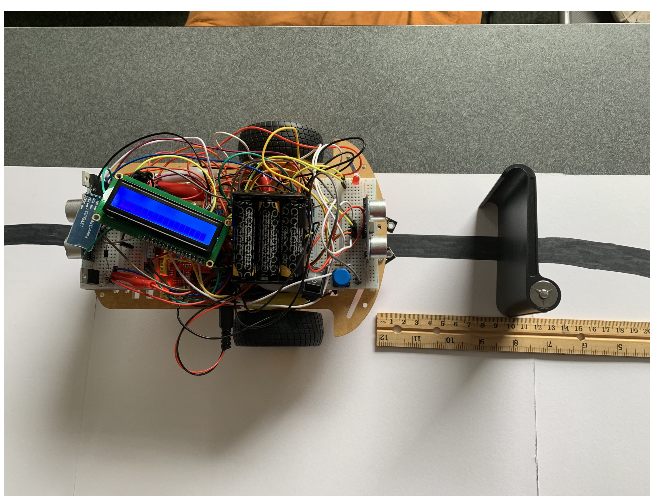

Figure 3: Method for Test 1

Trial	Distance (cm)	Theoretical Distance (cm)	Percent Error(%)
1	9.4	10	6
2	9.5	10	5
3	9.4	10	6
4	9.6	10	4
5	9.3	10	7
6	9.7	10	3
7	9.4	10	6
8	9.5	10	5
9	9.5	10	5
10	9.5	10	5
		Average Percent Error	5.2

Table 1: Result for Test 1

The percent error for test 1 is 5.2% which is less than 10%. We consider it passes the test. The result is factual because the car has inertia when it stops moving. Besides, the sensor need time to react as well.

The result of test 2 is shown as Table 2.

Trail	Radius[cm]	Result
1	30	Pass
2	30	Pass
3	30	Fail
4	30	Pass
5	30	Pass
6	30	Pass
7	30	Pass
8	30	Fail
9	30	Pass
10	30	Fail
	Possibility	70%

Table 2: Result for Test 2

The design failed this test because of following reason. The car's speed is a critical factor. If we increase the motor speed, the car is easier to move out of the track. While we decrease the speed, the torque decrease as well, the friction between wheels and ground is high enough to drag the car.

3 Project conclusion

3.1 Delivery on Requirements

From the delivery on requirement, all input and output requirements are met in this project. The basic functionality is achieved as well. The prototype of car, code and schematics are all provided in this project.

3.2 Difficulties

There are a lot difficulties during this project. First of all, the wire arrangement is messy. The space of car board is limited and there are a lot of inputs and outputs required in this project which make the car looking indecent. Secondly, we used MOSFET to control the DC motor which didn't perform well. The speed of motor is hard to control and it affected the motion of car. Lastly, the wire used in Makerspace Kit is so loose that the sometimes we did not notice them when they are disconnected.

3.3 Recommendations

For the improvement about further study, I highly recommend that the PCB (Printed Circuit Board) and motor driver can be provided in class. Besides, the wire welding should be introduced in this course as well. The wire arrangement is important for future electrical design.