LINE FOLLOWING CAR FOR AGRICULTURE

by

SHUBHAM KUMAR LALA	18BCE1025
AKSHAT KUMAR	18BCE1033
N. MAHESH	18BCE1256

A project report submitted to

Prof. MUTHULAKSHMI S

SCHOOL OF ELECTRONICS ENGINEERING

in partial fulfilment of the requirements for the course of

CSE2006 – MICROPROCESSOR AND INTERFACING

in

B.Tech. COMPUTER SCIENCE ENGINEERING

Vandalur – Kelambakkam Road

Chennai – 600127

APRIL 2020

BONAFIDE CERTIFICATE

Certified that this project report entitled "**LINE FOLLOWING CAR FOR AGRICULTURE"** is a bonafide work of **SHUBHAM KUMAR LALA– 18BCE1025, AKSHAT KUMAR – 18BCE1033, N. MAHESH 18BCE1256** who carried out the Project work under my supervision and guidance for **CSE2006-Microprocessor and Interfacing.**

PROF. MUTHULAKSHMI S

Associate Professor

School of Electronics Engineering (SENSE), VIT

University, Chennai

Chennai – 600 127.

Shubham Kumar Lala **Akshat Kumar** **N Mahesh**

ABSTRACT

Line following car is the model that follows the line drawn on the floor by detecting the black strip. This model can be combined with agriculture to provide sustainable use of resources such as water, etc. and proper care of the flora and fauna is ensured.

The car is equipped with sensors that detects the environment around it and upon sensing something unusual such as low moisture content carries out the desired activity assigned to it. It is also designed to send information of the environment to the device of the owner. This way the owner ensures that his/her plantations are flourishing under the right condition.

The main objective of this project is to reduce the workload of farmers in addition to providing proper knowledge of the environmental factors that ensures the good yield of the product.

ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our faculty **Prof. MUTHULAKSHMI S.,** School of Electronics Engineering, for her consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work. Her guidance helped us progress with our work leading to successful completion of this project work.

We also thank our friends and classmates who helped us in times of need and gave ideas that never crossed our minds. Due to their dedicated help we were finally able to come up with this innovative idea.

TABLE OF CONTENTS

SERIAL NO.		TITLE	PAGE NO.
		ABSTRACT	3
		ACKNOWLEDGEMENT	4
1		INTRODUCTION	6
	1.1	OBJECTIVES AND GOALS	6
	1.2	APPLICATIONS	6
	1.3	FEATURES	6
2		DESIGN	7-9
	2.1	BLOCK DIAGRAM	7-8
	2.2	HARDWARE ANALYSIS	8
	2.3	SNAPSHOT	9
3		SOFTWARE	10-16
	3.1	CODE AND ANALYSIS	10-14
	3.2	SNAPSHOTS	15-16
4		CONCLUSION AND FUTURE WORK	17
	4.1	CONCLUSION	17
	4.2	FUTURE WORK	17
5		REFERENCES	18

1. INTRODUCTION

1.1 OBJECTIVES AND GOALS

- Design the LINE FOLLOWING CAR
- Detect the presence of the moisture in the soil and manage accordingly.
- Design the automatic irrigation system which has a varying intensity depending upon the intensity of moisture.
- Design the temperature detecting mechanism so that irrigation can be controlled based on temperature also.
- Properly managing the RF sensor so that the car follows the line by detecting the line through RF sensor.

1.2 APPLICATIONS

- To follow the line across the field so that it can sense the moisture and temperature of the all the corners of the field and can carry out the operations as desired.
- To report proper information to the farmer so that it benefits him and leads to proper yield.

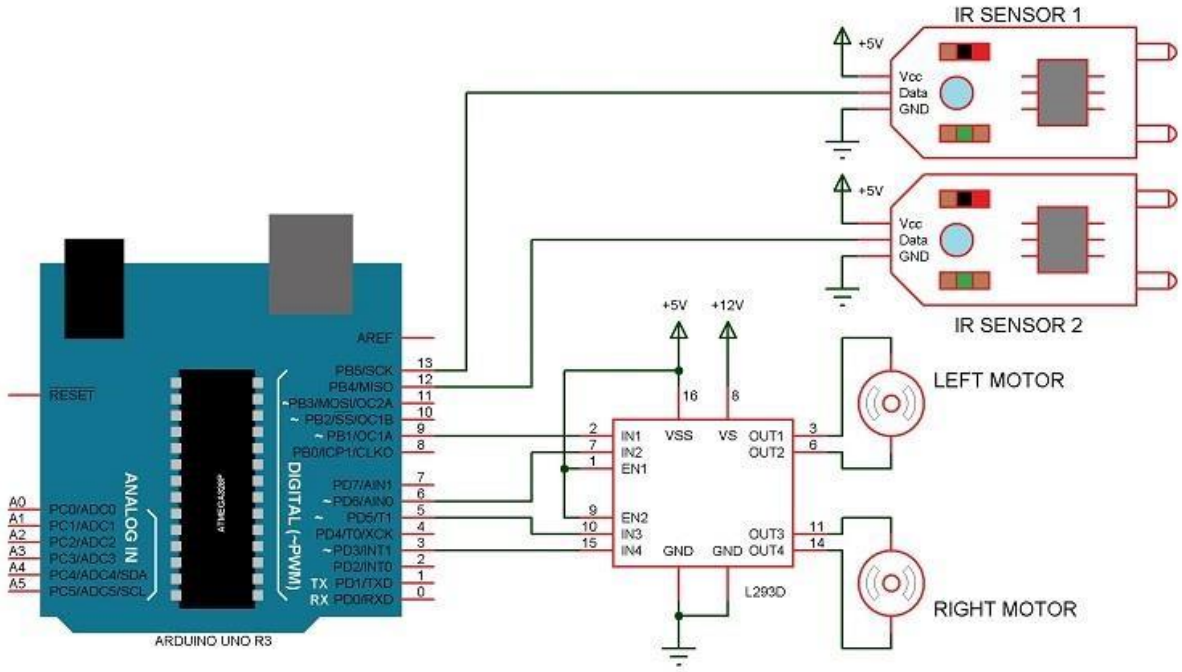
1.3 FEATURES

- To sense the moisture content in the environment.
- To sense the temperature of the environment.
- To follow the line using the RF sensor so that the car can traverse whole field properly.
- To irrigate the field on the basis of the sensed value of the moisture and temperature.

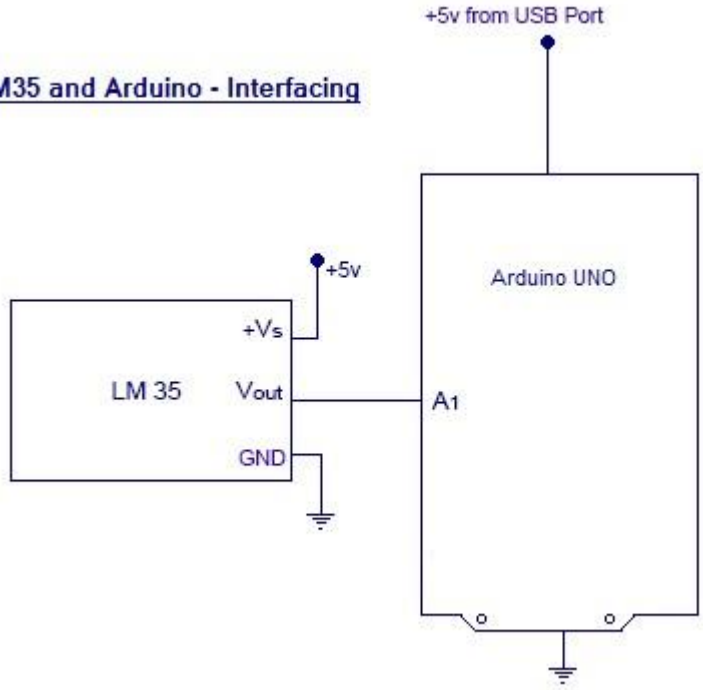
2. DESIGN

2.1 BLOCK DIAGRAM

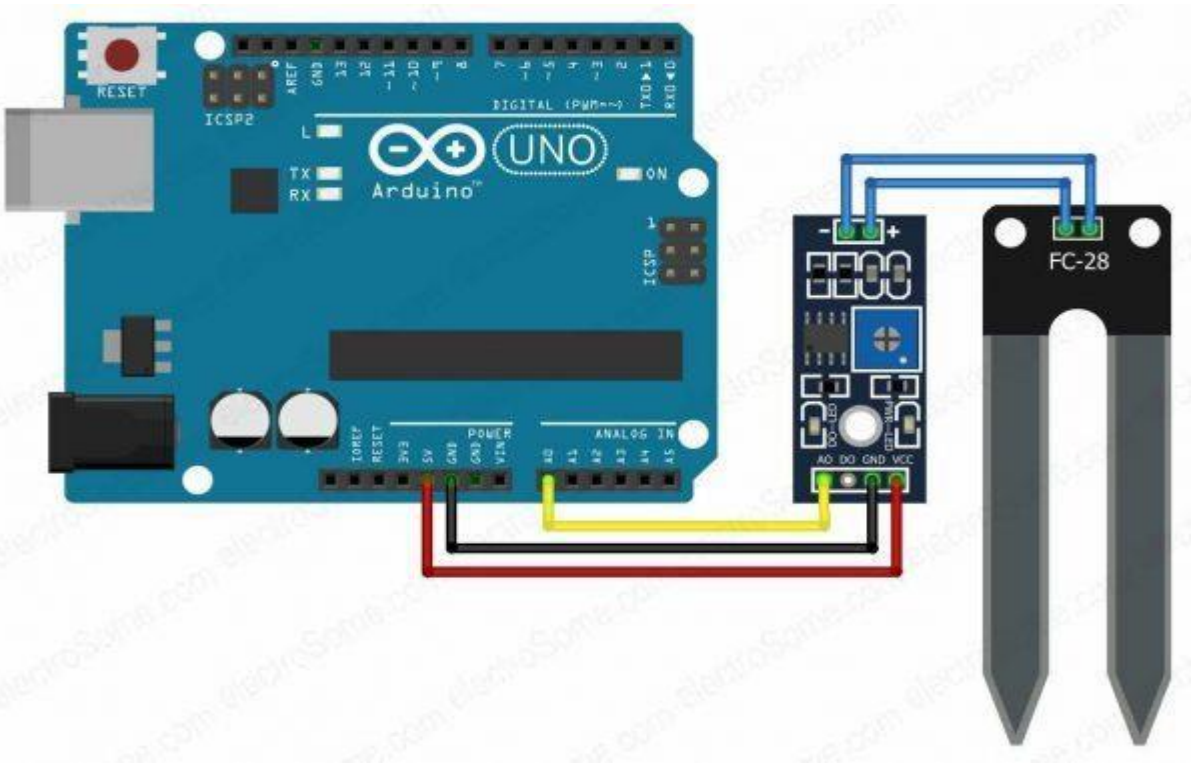
- Line following car



- LM35 and Arduino - Interfacing



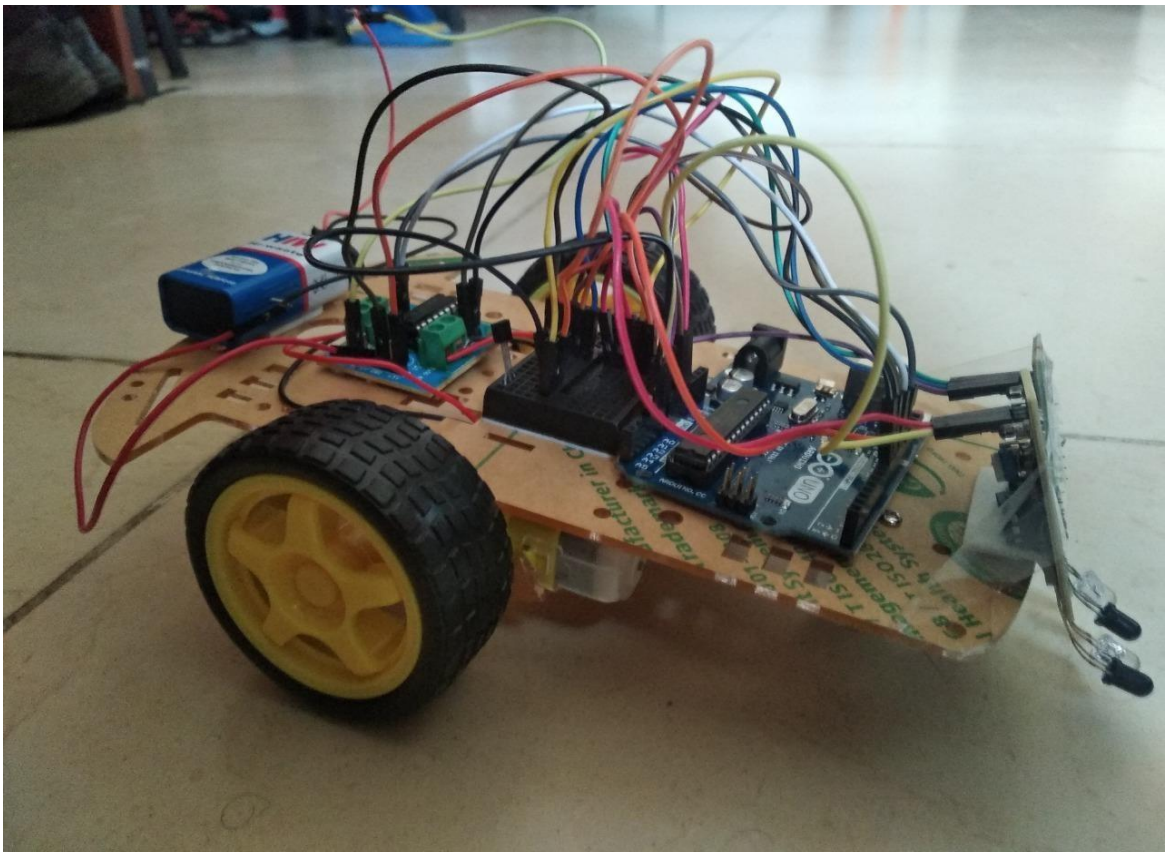
- Interfacing soil moisture module FC-28



2.2 HARDWARE ANALYSIS

1. Arduino Uno
2. Jumper cables (Female-Female, Male-Female)
3. Copper wires
4. LM-35 Temperature sensor module: Operating voltage: 4V-30V, Output signal: Analog output voltage which is proportional to temperature in degree Celsius, Range: −55°C to 150°C
5. Breadboard
6. Car Chassis
7. 4x Plastic tyres
8. 2x L293D Motor Driver Module: Supply-Voltage: 5 V, VSS Power Supply: 36 V, Separate Input-Logic Supply, Highest Output Current 1.2 A Per Channel.
9. 2x IR Sensor Module: Operating Voltage: 3.3V – 5.0V, Detection range: 2cm – 30cm (Adjustable using potentiometer), Current Consumption: at 3.3V: ~23 mA, at 5.0V: ~43 mA, On board Obstacle Detection LED indicator.
10. Soil moisture sensor: FC-28, Operating Voltage: 3.3V – 5.0, Output Voltage: 0 – 4.2V, Input Current: 35mA, Output signal: Both Analog and Digital

2.3 SNAPSHOT



3. SOFTWARE

3.1 CODE AND EXPLANATION

```
float tempc; //variable to store temperature in degree Celsius

float tempf; //variable to store temperature in Fahreinheit

float vout; //temporary variable to hold sensor reading

void setup()
{

pinMode(A3,INPUT); // Initialize pin A3 for soil moisture sensor

pinMode(A0,INPUT); // Initialize Pin A0 as Input Pin for IR Sensor 1

pinMode(A1,INPUT); // Initialize Pin A1 as Input Pin for IR Sensor 2

pinMode(8,OUTPUT); // Initialize Pin 8 as Output Pin for Motor Output 2

pinMode(9,OUTPUT); // Initialize Pin 9 as Output Pin for Motor Output 2

pinMode(10,OUTPUT); // Initialize Pin 10 as Output Pin for Motor Output
1
```

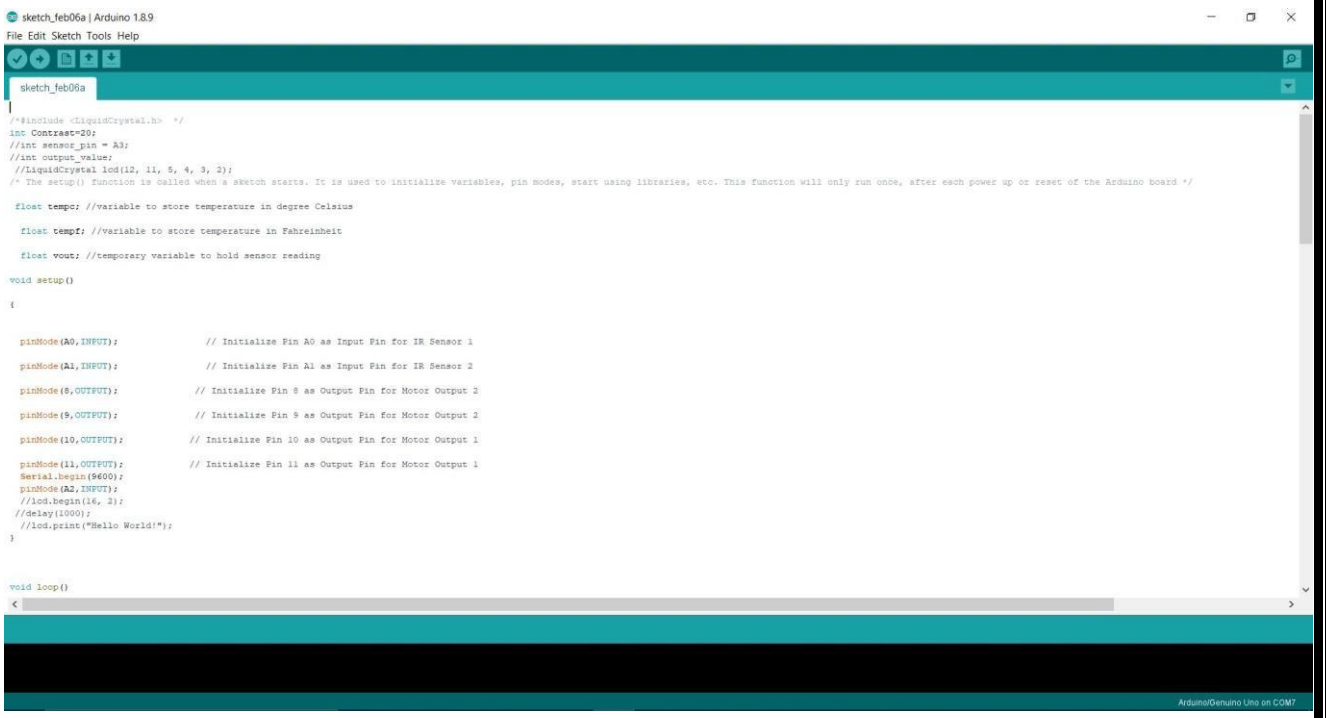
```
  pinMode(11,OUTPUT); // Initialize Pin 11 as Output Pin for Motor Output
1

  Serial.begin(9600); // Initialize Serial monitor for output

  pinMode(A2,INPUT); // Initialize pin A2 for temperature sensor module
  }

void loop() // put your main code here, to run repeatedly:

{
  vout=analogRead(A2); //Reading the value from temperature sensor

  output_value= analogRead(sensor_pin); //Read soil moisture value

  output_value = map(output_value,550,0,0,100); //Analog to readable value

  Serial.print("Mositure : ");

  Serial.print(output_value);

  Serial.println("%");

  delay(1000); // Print and delay the monitor
```

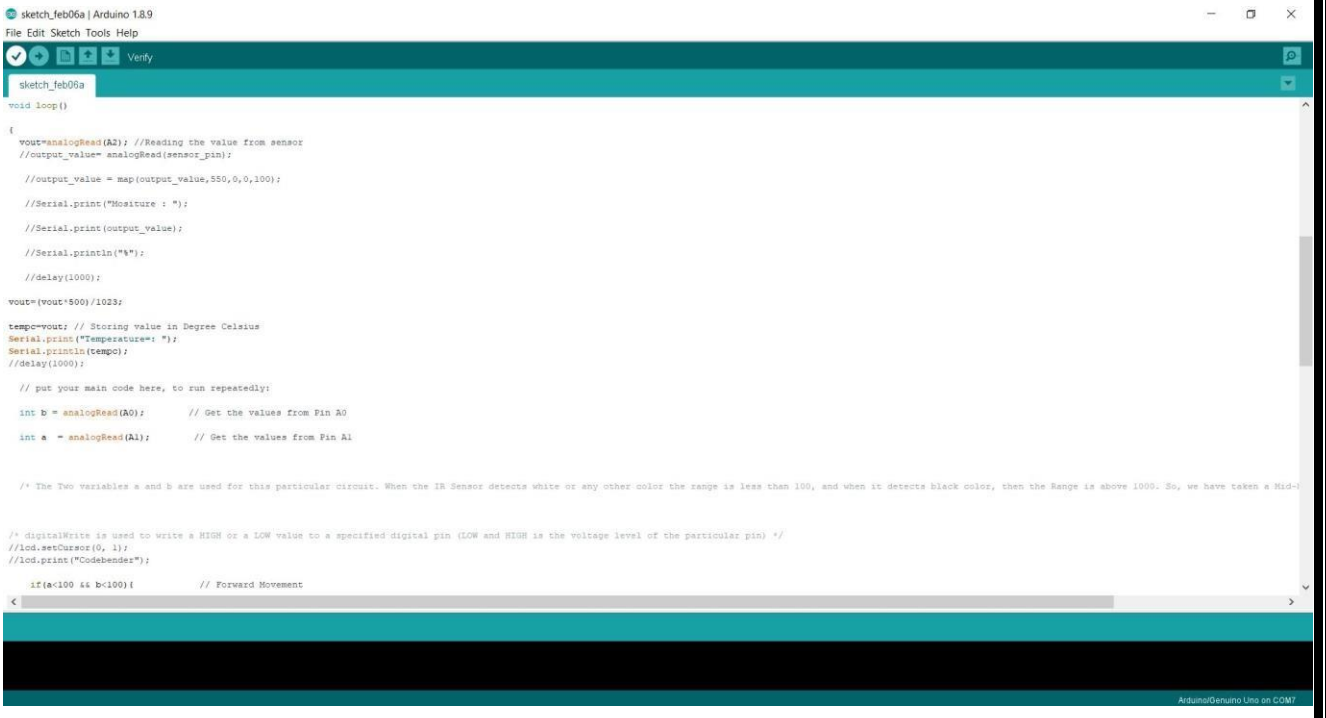
```
vout=(vout*500)/1023; // Converting analog to readable value

tempc=vout; // Storing value in Degree Celsius

Serial.print("Temperature=: ");

Serial.println(tempc);

delay(1000); // Print and delay monitor

int b = analogRead(A0);        // Get the values from Pin A0

int a  = analogRead(A1);        // Get the values from Pin A1

 /* The Two variables a and b are used for this particular circuit. When the
IR Sensor detects white or any other color the range is less than 100,       and
when it detects black color, then the Range is above 1000. So, we have taken
a Mid-Point Parameter with range of 100. */

/* digitalWrite is used to write a HIGH or a LOW value to a specified digital
pin (LOW and HIGH is the voltage level of the particular pin) */

   if(a<100 && b<100){           // Forward Movement
digitalWrite(8,HIGH);          // Pin 8 is High for Motor Output 2
digitalWrite(9,LOW);           // Pin 9 is Low     digitalWrite(10,HIGH);
// Pin 10 is High for Motor Output 1
```

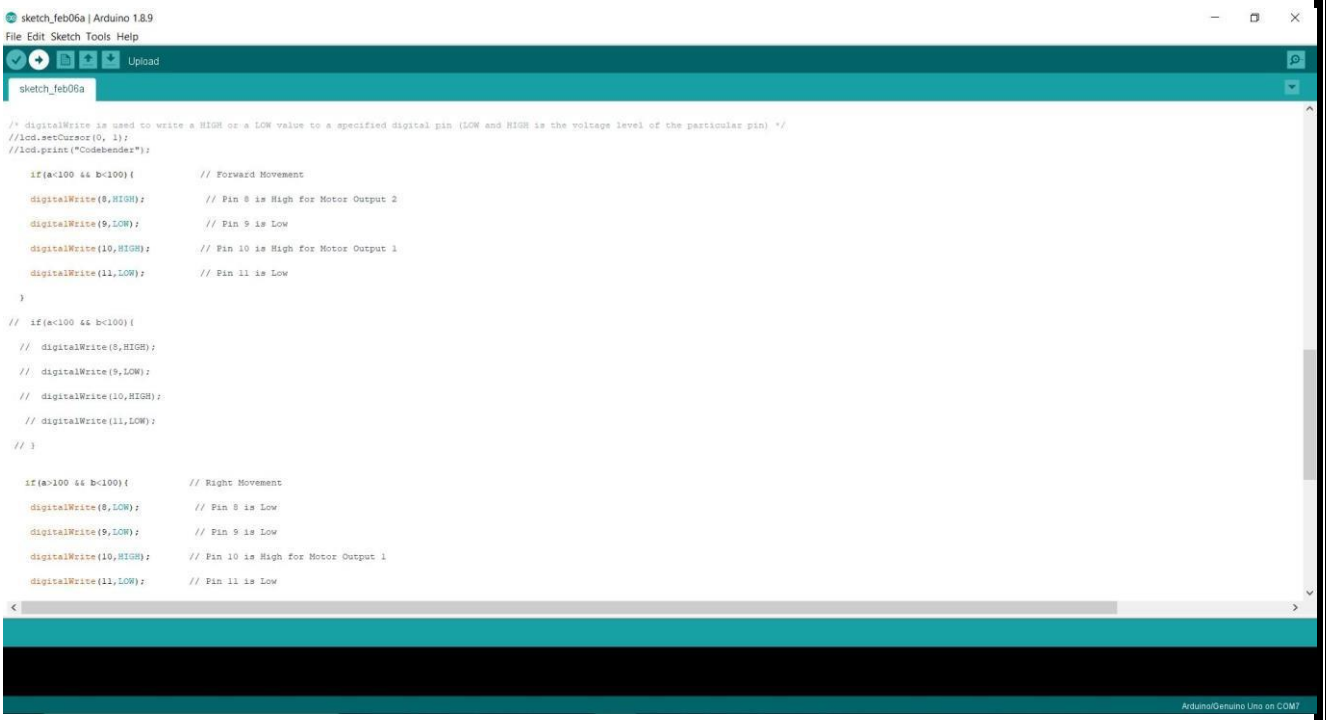
```
   digitalWrite(11,LOW);          // Pin 11 is Low
  }

  if(a>100 && b<100){            // Right Movement

   digitalWrite(8,LOW);          // Pin 8 is Low

   digitalWrite(9,LOW);          // Pin 9 is Low

   digitalWrite(10,HIGH);        // Pin 10 is High for Motor Output 1

   digitalWrite(11,LOW);         // Pin 11 is Low

}
   if(a<100 && b>100){           // Left Movement

   digitalWrite(8,HIGH);         // Pin 8 is High for Motor Output 2

   digitalWrite(9,LOW);          // Pin 9 is Low

   digitalWrite(10,LOW);          // Pin 10 is Low      digitalWrite(11,LOW);
// Pin 11 is Low

  }
```

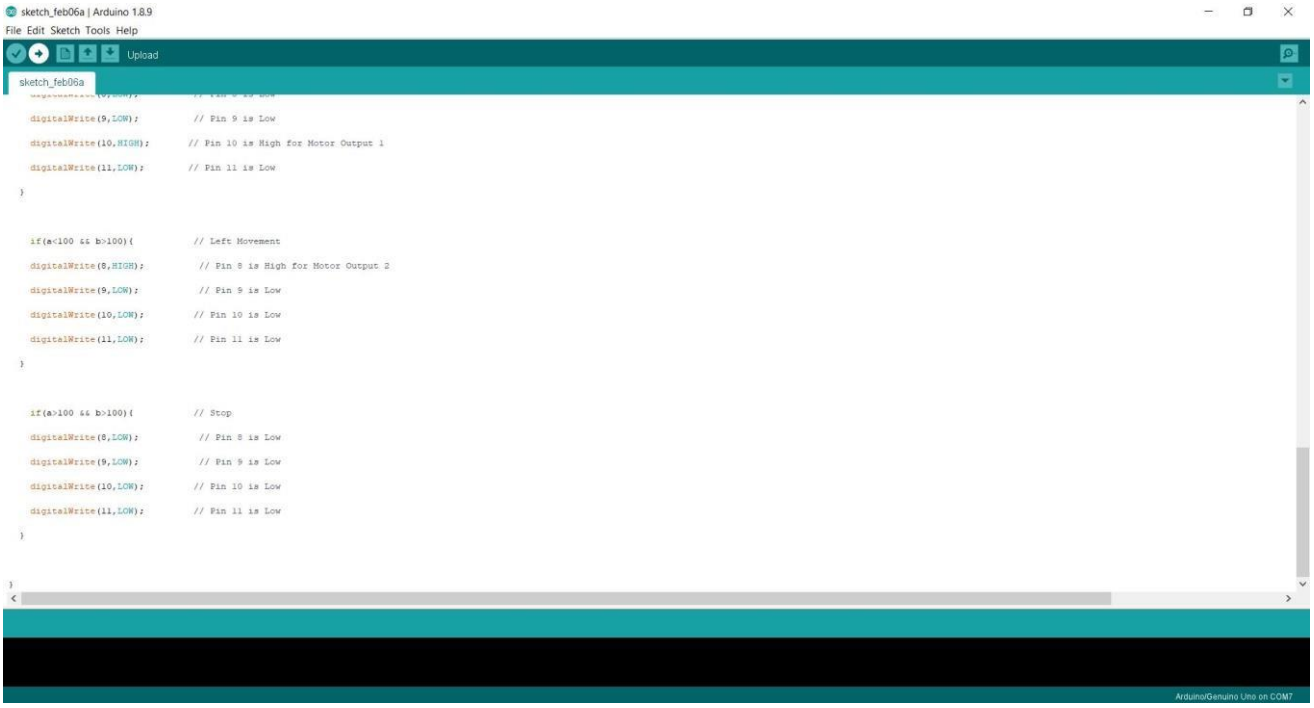
```
    if(a>100 && b>100){          // Stop

    digitalWrite(8,LOW);         // Pin 8 is Low

    digitalWrite(9,LOW);         // Pin 9 is Low

    digitalWrite(10,LOW);        // Pin 10 is Low

    digitalWrite(11,LOW);        // Pin 11 is Low

  }
}
```

3.2 SNAPSHOTS






sketch_feb06a | Arduino 1.8.9
File Edit Sketch Tools Help
Upload
sketch_feb06a
/* digitalWrite is used to write a HIGH or a LOW value to a specified digital pin (LOW and HIGH is the voltage level of the particular pin) */
//lcd.setCursor(0, 1);
//lcd.print("Codebender");
if(a<100 && b<100){ // Forward Movement
digitalWrite(8,HIGH); // Pin 8 is High for Motor Output 2
digitalWrite(9,LOW); // Pin 9 is Low
digitalWrite(10,HIGH); // Pin 10 is High for Motor Output 1
digitalWrite(11,LOW); // Pin 11 is Low
}
// if(a<100 && b<100){
// digitalWrite(8,HIGH);
// digitalWrite(9,LOW);
// digitalWrite(10,HIGH);
// digitalWrite(11,LOW);
// }
if(a>100 && b<100){ // Right Movement
digitalWrite(8,LOW); // Pin 8 is Low
digitalWrite(9,LOW); // Pin 9 is Low
digitalWrite(10,HIGH); // Pin 10 is High for Motor Output 1
digitalWrite(11,LOW); // Pin 11 is Low
Arduino/Genuino Uno on COM7


sketch_feb06a | Arduino 1.8.9
File Edit Sketch Tools Help
Upload
sketch_feb06a
digitalWrite(9,LOW); // Pin 9 is Low
digitalWrite(10,HIGH); // Pin 10 is High for Motor Output 1
digitalWrite(11,LOW); // Pin 11 is Low
}
if(a<100 && b>100){ // Left Movement
digitalWrite(8,HIGH); // Pin 8 is High for Motor Output 2
digitalWrite(9,LOW); // Pin 9 is Low
digitalWrite(10,LOW); // Pin 10 is Low
digitalWrite(11,LOW); // Pin 11 is Low
}
if(a>100 && b>100){ // Stop
digitalWrite(8,LOW); // Pin 8 is Low
digitalWrite(9,LOW); // Pin 9 is Low
digitalWrite(10,LOW); // Pin 10 is Low
digitalWrite(11,LOW); // Pin 11 is Low
}
}
Arduino/Genuino Uno on COM7

4. CONCLUSION AND FUTURE WORK

4.1 CONCLUSION

Technology can serve a greater purpose by making them handle the redundant tasks which we humans repeat. For the farmers, they need to keep a constant check on the environment on which their crops are growing. Our device makes that task a bit easier. So now they need not go to their land and test each and every time.

By reducing their burden of checking their environment on which crops are growing, they can focus on other works like marketing, future planning etc.

4.2 FUTURE WORK

- An involvement of mobile application which will be installed in user's phone. The application will uniquely identify the controlling device and store the values in database for future reference.
- An additional variable will be added which can uniquely identify the place or the relative distance from where the measurements that were taken.

5. REFERENCES

https://www.electronicshub.org/arduino-line-follower-robot/

http://www.circuitstoday.com/lm35-and-arduino-interfacing

https://electrosome.com/interfacing-soil-moisture-sensor-arduino/