Ahsanullah University of Science & Technology

Department of Computer Science & Engineering

Line Follower Robot

CSE 3216

Microcontroller Based System Design Lab

Submitted By:

Md. Reasad Zaman	16.01.04.004
Abu Hasanat Jaman	16.01.04.005
Bayezid Hasan	16.01.04.006
Pushpal Kumar Paul	16.01.04.018

Date of Submission: **17 April, 2019**

Introduction:

Line follower is a machine that can follow a path. The path can be visible like a black line on a white surface. Sensing a line and maneuvering the robot to stay on course, while constantly correcting wrong moves using feedback from the sensor forms a simple yet effective system. It can be used in automobile, industrial automation and guidance etc.

Equipment:

- **Arduino Uno:** The Arduino UNO is an open-source microcontroller board based on the Microchip ATmega328P microcontroller. The microcontroller is programmed to make the robot move forward, turn left or right. Output of the Arduino is fed to motor driver which drives the motor. It gets signals from infrared sensors and sonar sensors and drives motors according to the sensor inputs.

- **L293D Motor Driver Shield:** This motor driver shield is placed directly on Arduino board. Arduino gives signals according to sensor inputs which drives the motor driver. 4 gear motor is connected with the motor driver shield to drive the robot.

- **DC Geared Motor:** 4 dc geared motor of 100 rpm is used for the robot. These motors are connected with L293D motor driver which is responsible to drive the motors.

- **Robot Chassis:** Chassis is used for giving a complete structure to our robot.

- **IR Sensor Module:** The sensor module has a pair of infrared transmitting and receiving tube, when detecting direction meets with obstacles (reflecting surface), reflected infrared receiving by the tube. This sensor can differentiate between black and white colors and gives a signal to the Arduino corresponding to it.

- **Sonar Sensor:** This sensor is used to find an obstacle within the range of 12cm in front of the robot. If any obstacle found within the range the robot will turn around to find a new path or return to the previous path.

- **Power Supply:** We used a 5V power bank as the power supply which is used to drive both Arduino and motor driver shield.

- **Connecting Wires:** Wires are used to connect different components with each other.

- **Double Sided Tape:** Double sided tape is used to place the components on the robot chassis.

Software:

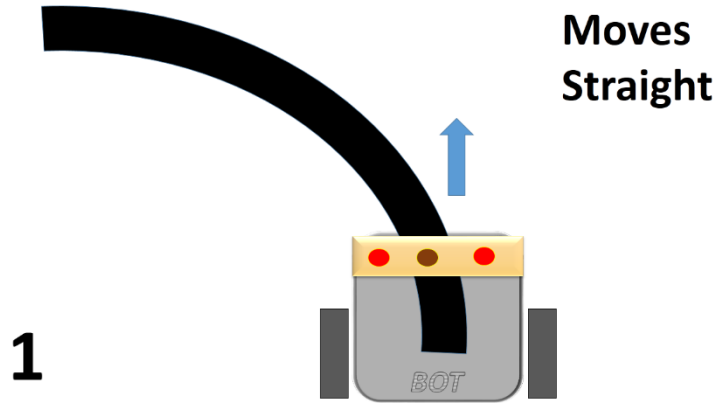
- **Arduino IDE:** Arduino IDE is used for coding, debugging and uploading code to the board.

Features:

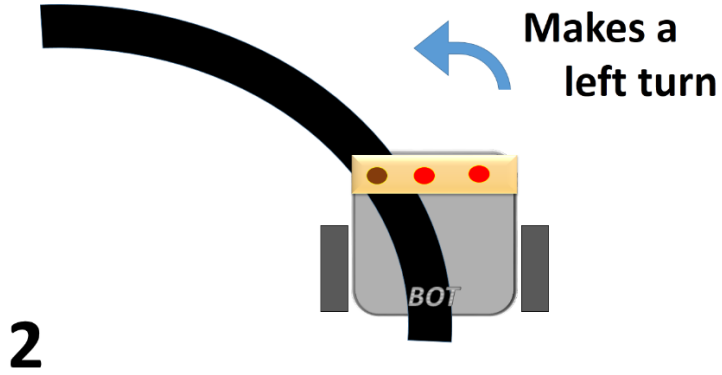
- It can follow a black line in white surface.
- It is capable of various degrees of turns.
- It can detect any obstacle in front of it and can avoid the obstacle.
- It can start or start on receiving a certain signal.
- This can be used in automation in industrial purpose.

Working Principle:

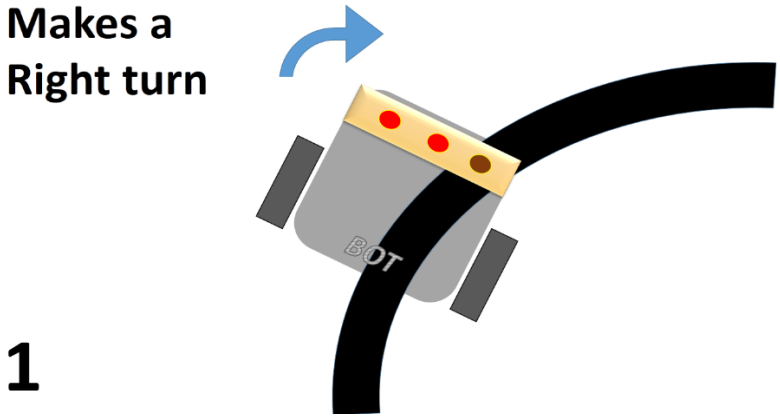
Working of line follower is very interesting. Line follower robot senses black line by using sensor and then sends the signal to arduino. Then arduino drives the motor according to sensors' output. For our project we have used 3 IR sensors namely left sensor, middle sensor and right sensor for following a line a sonar sensor for the obstacle avoidance.
When only middle sensor senses black and other two sensors senses white it moves forward.



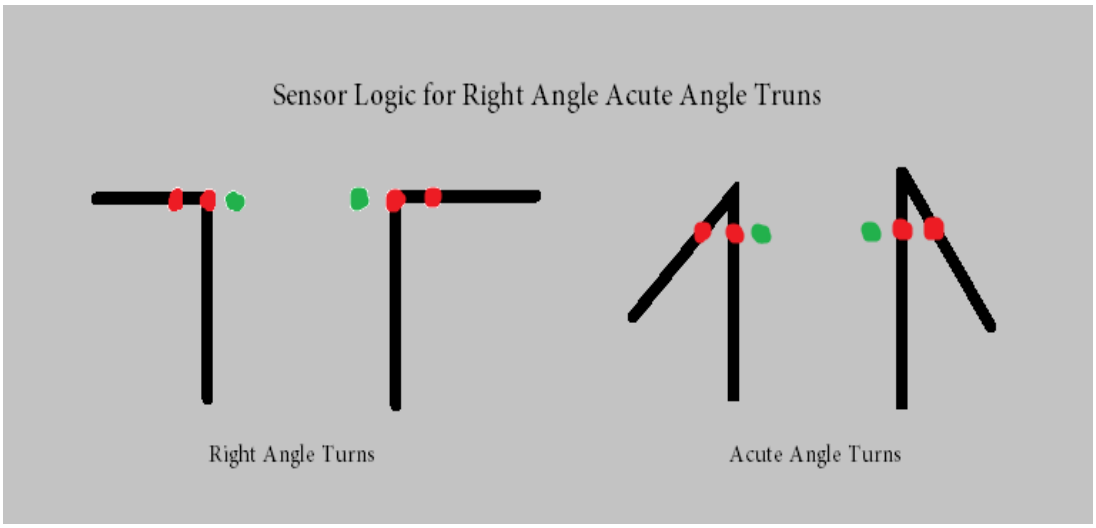
When only left sensor senses black and other two sensors senses white or when both left and middle sensor senses black and right sensor senses white it moves left.



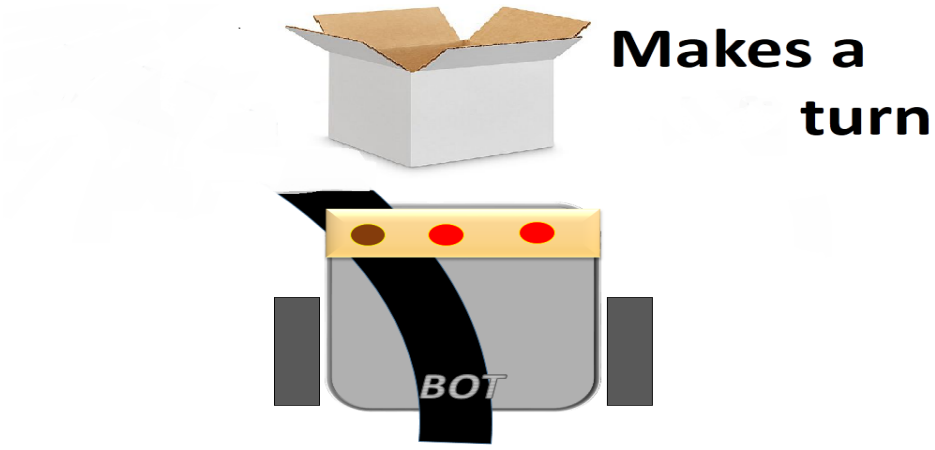
When only right sensor senses black and other two sensors senses white or when both right and middle sensor senses black and left sensor senses white it moves right.



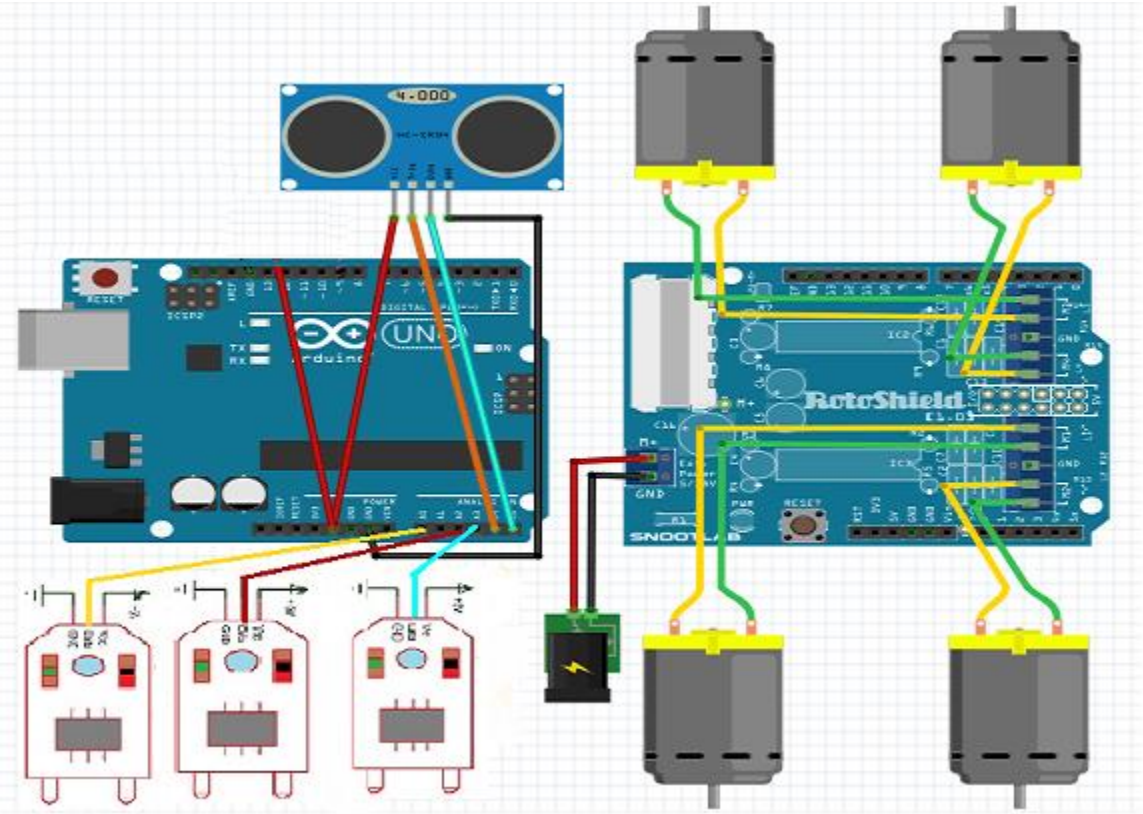
It can also be able to take right angle turn and acute angle turns by following manner.



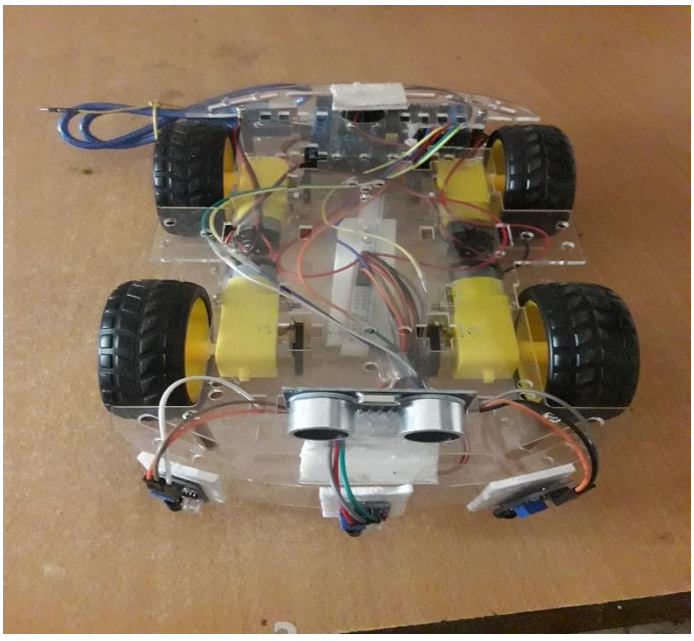
If any obstacle comes within 15 cm in front of it makes a turn to avoid the obstacle.



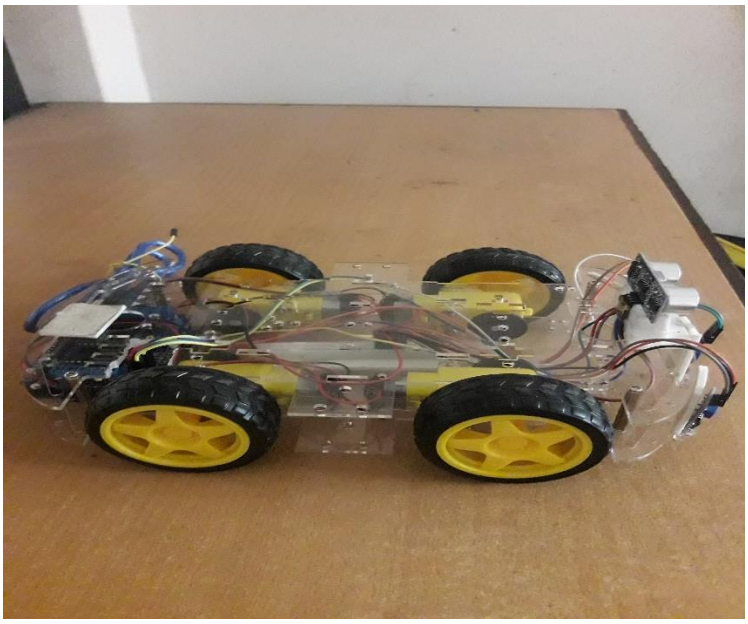
Circuit Diagram:



Figures of The Project:



Upper view



Side View

Constrains:

- Calibration is difficult and it is not easy to set a perfect value.
- Sunlight or some sort of lighting may affect the robot from detecting the line.
- The width of the path can not be too large or too small.
- The path should be plane surface.

Do and Don’ts:

Functionality	Does	Doesn’t
Follow black line on white surface	✓	
Follow white line on black surface		✓
Right angle and acute angle turn	✓	
Line gap detection	✓	
Stop on end point	✓	
Carry heavy loads		✓
Auto calibration		✓
Obstacle avoidance	✓	

Conclusion:

The line following robot is automobile system that has ability to recognize its path, move and change robot’s position toward the line in the best to remain on the track. We have made a 3 IR sensor based small line follower robot which can follow a black line on white surface and with

sonar sensor it can be able to avoid any obstacle in front of it. This project has helped us to cooperate, communicate and make us to understand basic electronics, mechanical systems and their integration with programming. This project may later be implemented in broader scale in automation of industries or automatic vehicle.

Appendix:

Program Code:

```
#include<AFMotor.h>
#include<NewPing.h>

#define ultrasonicTrigger A3
#define ultrasonicEcho A4

AF_DCMotor motor_fl(3);   //front left
AF_DCMotor motor_fr(4);   //front right
AF_DCMotor motor_br(1);   //back right
AF_DCMotor motor_bl(2);   //back left

int r_ir = A0;
int c_ir = A1;
int l_ir = A2;
int black = 0;
int white = 0;

const int maxDistance = 400;
NewPing ultrasonic(ultrasonicTrigger, ultrasonicEcho, maxDistance);

const int MIN_DISTANCE_FROM_OBJECT = 12;
// Error of the distance read by the ultrasonic sensor
const float DISTANCE_OFFSET = -0.31;

int maxv = 450;
int maxspeed = 255;
long duration;
int distance;

void setup() {
```

```
  motor_fl.setSpeed(maxspeed);
  motor_fr.setSpeed(maxspeed);
  motor_br.setSpeed(maxspeed);
  motor_bl.setSpeed(maxspeed);

  pinMode(r_ir,INPUT);
  pinMode(c_ir,INPUT);
  pinMode(l_ir,INPUT);
  Serial.begin(9600);

  pinMode(ultrasonicTrigger, OUTPUT);
  pinMode(ultrasonicEcho, INPUT);
}


void loop() {

  if (hasObstacle()){
    black = 0;
    white = 0;
    motor_br.setSpeed(maxspeed);
    motor_fr.setSpeed(maxspeed);
    motor_bl.setSpeed(0);
    motor_fl.setSpeed(0);

    //motor_fl.run(RELEASE);
    motor_fr.run(FORWARD);
    //motor_bl.run(RELEASE);
    motor_br.run(FORWARD);
  }
  while (hasObstacle()) delay(100);

  if ((digitalRead(l_ir) == 0 ) && (digitalRead(c_ir) == 1) && (digitalRead(r_ir) == 0)){
    //forward();
    //delay(100);
    black = 0;
    white = 0;
    motor_fl.setSpeed(maxspeed);
    motor_fr.setSpeed(maxspeed);
    motor_br.setSpeed(maxspeed);
```

```
  motor_bl.setSpeed(maxspeed);

  motor_fl.run(FORWARD);
  motor_fr.run(FORWARD);
  motor_bl.run(FORWARD);
  motor_br.run(FORWARD);
}

if((digitalRead(l_ir) == 1) && (digitalRead(c_ir) == 1) && (digitalRead(r_ir)  == 0)){
  //left();
  black = 0;
  white = 0;
  motor_br.setSpeed(maxspeed);
  motor_fr.setSpeed(maxspeed);
  motor_bl.setSpeed(0);
  motor_fl.setSpeed(0);

  //motor_fl.run(RELEASE);
  motor_fr.run(FORWARD);
  //motor_bl.run(RELEASE);
  motor_br.run(FORWARD);

  }

if((digitalRead(l_ir) == 1) && (digitalRead(c_ir) == 0) && (digitalRead(r_ir) == 0)){
  //left();
  black = 0;
  white = 0;
  motor_br.setSpeed(maxspeed);
  motor_fr.setSpeed(maxspeed);
  motor_bl.setSpeed(0);
  motor_fl.setSpeed(0);

  //motor_fl.run(RELEASE);
  motor_fr.run(FORWARD);
  //motor_bl.run(RELEASE);
  motor_br.run(FORWARD);
  }

if((digitalRead(l_ir) == 0) && (digitalRead(c_ir) == 1) && (digitalRead(r_ir) == 1)){
```

```
  //rigth();
  black = 0;
  white = 0;
  motor_bl.setSpeed(maxspeed);
  motor_fl.setSpeed(maxspeed);
  motor_br.setSpeed(0);
  motor_fr.setSpeed(0);
  motor_fl.run(FORWARD);
  //motor_fr.run(RELEASE);
  motor_bl.run(FORWARD);
  //motor_br.run(RELEASE);
  }

if((digitalRead(l_ir) == 0) && (digitalRead(c_ir) == 0) && (digitalRead(r_ir) == 1)){
  //rigth();
  black = 0;
  white = 0;
  motor_bl.setSpeed(maxspeed);
  motor_fl.setSpeed(maxspeed);
  motor_br.setSpeed(0);
  motor_fr.setSpeed(0);
  motor_fl.run(FORWARD);
  //motor_fr.run(RELEASE);
  motor_bl.run(FORWARD);
  //motor_br.run(RELEASE);
  }

if((digitalRead(l_ir) == 1) && (digitalRead(c_ir) == 1) && (digitalRead(r_ir) == 1)){
  //Stop();
  black++;
  white = 0;
  if(black>1)
  {
    motor_fl.run(RELEASE);
    motor_fr.run(RELEASE);
    motor_bl.run(RELEASE);
    motor_br.run(RELEASE);
  }
  else
  {
```

```
      motor_fl.setSpeed(maxspeed);
      motor_fr.setSpeed(maxspeed);
      motor_br.setSpeed(maxspeed);
      motor_bl.setSpeed(maxspeed);

      motor_fl.run(FORWARD);
      motor_fr.run(FORWARD);
      motor_bl.run(FORWARD);
      motor_br.run(FORWARD);
    }
  }
}

boolean hasObstacle() {
  int distance = ultrasonic.ping_cm();
  return distance > 0 && distance <= MIN_DISTANCE_FROM_OBJECT;
}
```