CHITTAGONG UNIVERSITY OF ENGINEERING AND TECHNOLOGY
Department of Electrical and Electronic Engineering

Report for the final project

Line Follower Robot With Obstacle Detection.

By

Mehedi Hasan Akash 1702117

Bahar Bhuiyan 1702119

MD Hasnaine Ahmed 1702124

Supervised By

Dr. Joyprakash Chakraborty

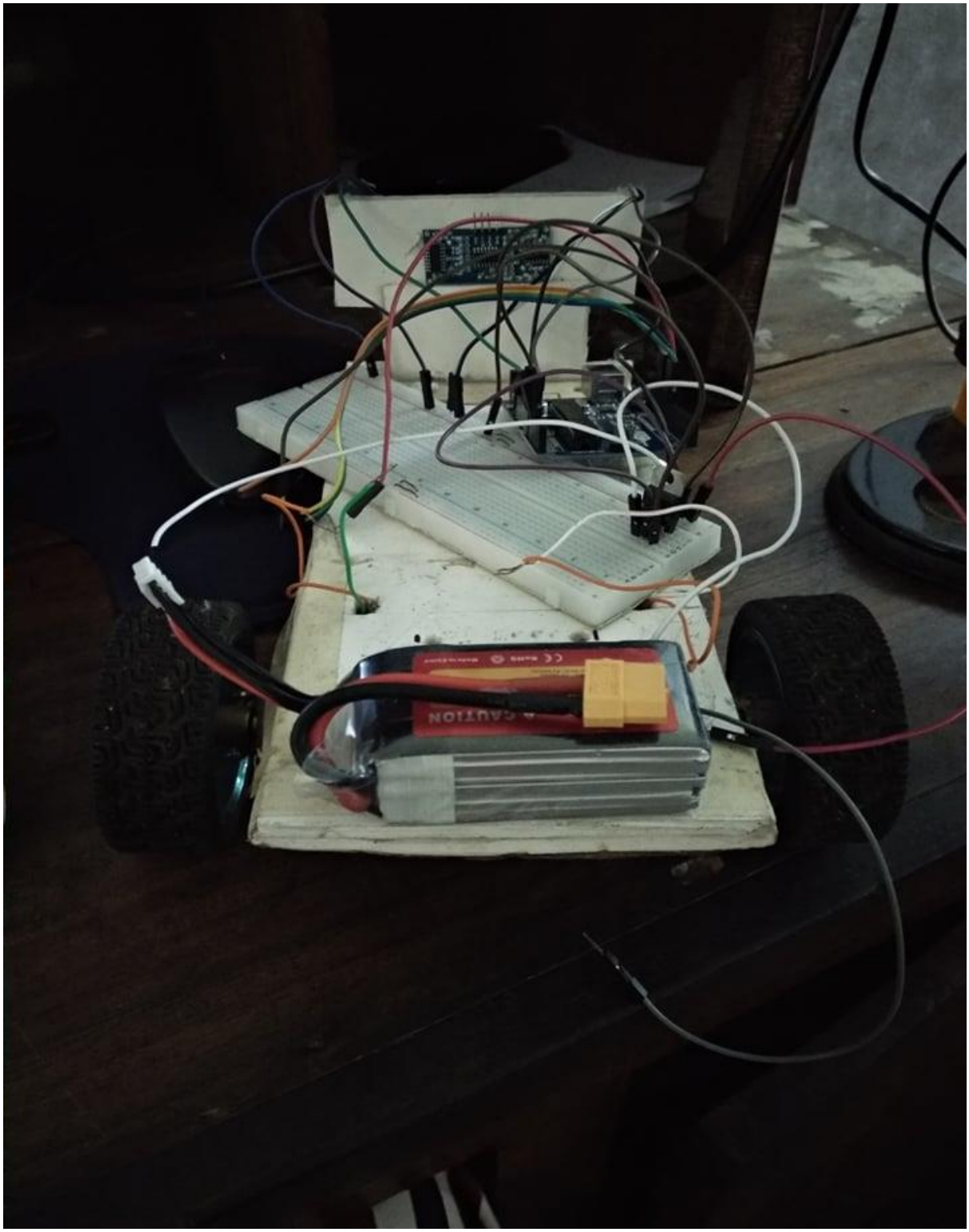
Assistant Professor, EEE,
CUET

Naqib Sad Pathan

Lecturer, EEE, CUET

Introduction:

Arduino sends commands to this motor driver and then it drives motors in any direction as we want. Working of line follower and obstacle avoidance robot is very interesting. Then Arduino drives the motor according to sensors' output.: When both left and right sensor senses white then robot move forward.



Theory:

It's basically a black line follower robot that is capable of stopping in front of an obstacle or reaching the end of the route (symbolized by when both line sensors detect a black line). The vehicle can listen to a command received by the serial to start or stop the route.

The full project consists of controlling this vehicle based on traffic light status that will be detected using an Android smartphone in a controlled testing environment.

These robots usually use an array of IR (infrared) sensors in order to calculate the reflectance of the surface beneath them. The basic criteria is that the black line has a lesser reflectance value (black absorbs light) than the lighter surface around it.

That is proximity sensor and IR sensor. The proximity sensor used for path detection and IR sensor used for obstacle detection. These sensors mounted at front end of the robot. The microcontroller is an intelligent device the whole circuit is controlled by the microcontroller.

The VEX Line Tracking Sensor allows the robot to tell objects or surfaces apart based on how dark or light they are. It shines a beam of infrared light out onto the object, and measures how much light is reflected back. The Line Tracking Sensor is an analog sensor, and it returns values in the range of 0 to 4095.

Arduino refers to an open-source electronics platform or board and the software used to program it. Arduino is designed to make electronics more accessible to artists, designers, hobbyists and ayone interested in creating interactive objects or environments.

nfrared proximity sensor made by Sharp. Part # GP2Y0A02YK0F has an analog output that varies from 2.8V at 15cm to 0.4V at 150cm with a supply voltage between 4.5 and 5.5VDC. ... This sensor is great for sensing objects up to 5 feet away!

Components list:

1. Arduino uno
2. Ultrasonic Sensors
3. Geared dc motor
4. Li-po battery
5. Motor driver
6. Resistors
7. Voltage regulator 7805, 7812
8. Connecting wires
9. **Cost Estimation:**
10.

Parts Name	**Amount (pcs)**	**Expenditure(Tk)**
Arduino Uno (set)	1	300
IR Sensor	2	90
Geared dc Motor	2	320
Motor driver	1	250
Voltage regulator(7805)	1	20
Li-po battery	1	500
Hardboard	1	90
360 rotary wheel	1	20
Breadboard	1	80
Jumper Wires	20-25	40
Others		200 (approx)
	Total	2500TK

Code:

```
int vSpeed = 110;         // MAX 255
  int turn_speed = 230;     // MAX 255
  int turn_delay = 10;

//L293 Connection
  const int motorA1       = 8;
  const int motorA2       = 10;
  const int motorAspeed   = 9;
  const int motorB1       = 12;
  const int motorB2       = 13;
  const int motorBspeed   = 11;

//Sensor Connection
  const int left_sensor_pin =A0;
  const int right_sensor_pin =A1;



  int left_sensor_state;
  int right_sensor_state;

void setup() {
  pinMode(motorA1, OUTPUT);
  pinMode(motorA2, OUTPUT);
  pinMode(motorB1, OUTPUT);
  pinMode(motorB2, OUTPUT);

  Serial.begin(9600);

  delay(3000);

}

void loop() {




left_sensor_state = analogRead(left_sensor_pin);
right_sensor_state = analogRead(right_sensor_pin);
```

```
if(right_sensor_state > 500 && left_sensor_state < 500)
{
  Serial.println("turning right");

  digitalWrite (motorA1,LOW);
  digitalWrite(motorA2,HIGH);
  digitalWrite (motorB1,LOW);
  digitalWrite(motorB2,HIGH);

  analogWrite (motorAspeed, vSpeed);
  analogWrite (motorBspeed, turn_speed);

  }
if(right_sensor_state < 500 && left_sensor_state > 500)
{
  Serial.println("turning left");

  digitalWrite (motorA1,HIGH);
  digitalWrite(motorA2,LOW);
  digitalWrite (motorB1,HIGH);
  digitalWrite(motorB2,LOW);

  analogWrite (motorAspeed, turn_speed);
  analogWrite (motorBspeed, vSpeed);

  delay(turn_delay);
  }

if(right_sensor_state > 500 && left_sensor_state > 500)
{
  Serial.println("going forward");

  digitalWrite (motorA2,LOW);
  digitalWrite(motorA1,HIGH);
  digitalWrite (motorB2,HIGH);
  digitalWrite(motorB1,LOW);

  analogWrite (motorAspeed, vSpeed);
  analogWrite (motorBspeed, vSpeed);

  delay(turn_delay);

  }

if(right_sensor_state < 500 && left_sensor_state < 500)
{
```

```
  Serial.println("stop");

  analogWrite (motorAspeed, 0);
  analogWrite (motorBspeed, 0);

  }


}
```

Application:

Line Following is one of the most important aspects of robotics. A Line Following Robot is an autonomous robot which is able to follow either a black line that is drawn on the surface consisting of a contrasting color. It is designed to move automatically and follow the line. The robot uses arrays of optical sensors to identify the line, thus assisting the robot to stay on the track. The array of four sensor makes its movement precise and flexible. The robot is driven by DC gear motors to control the movement of the wheels. The Arduino Uno interface is used to perform and implement algorithms to control the speed of the motors, steering the robot to travel along the line smoothly.This project aims to implement the algorithm and control the movement of the robot by proper tuning of the control parameters and thus achieve better performance.In addition the LCD interface is added in order to display the distance travelled by the robot. It can be used industrial automated equipment carriers, small household applications, tour guides in museums and other similar applications, etc.

Discussion:

The backbone of our design is the differential steering system which is familiar from ordinary life because it is the arrangement used in a wheelchair. Two wheels mounted on a single axis are independently powered and controlled, thus providing both drive and steering. Additional passive wheels (usually casters) are provided for support. Most of us have an intuitive grasp of the basic behavior of a differential steering system. If both drive wheels turn in tandem, the robot moves in a straig