INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

Garden of Knowledge and Virtue

BUILDING A LIFT-UP FOLLOWER ROBOT

(LAB PROJECT REPORT)

MCTA 1101 SECTION 2

Group : 7

NAME	MATRIC NUMBER
MOHAMAD NASRI BIN MOHAMAD NAZRI	2219879
MUHAMMAD ARIEF ZULKARNAIN BIN ABDUL RAHMAN	2217275
MUHAMMAD AFIQ IKHWAN BIN NOR SHAHRIZAL	2215897

INTRODUCTION

An open-source hardware development board called an Arduino can be used to design and construct practical interactive devices. Although the term "Arduino" refers to a certain type of board, it comes in many different shapes and functionalities. Arduino boards that are offered commercially come pre-assembled or as DIY kits.

In this project, the Arduino Uno served as the robot's brain. Basically, the concept is that a vehicle robot will move in response to programming-controlled commands. The robot's goal is to move on its own or with a controller's assistance. Three main tasks are programmed for autonomous mode: it is intended to drive towards obstacles and should seamlessly avoid them if they are met. The car will finally follow any dark lines under it until it reaches its destination. When operating in semi-autonomous mode, a phone app serves as a remote controller, causing the robot to move in response to instructions (such as picking up the object required using the mechanism created). The user can select between the car robot's autonomous and semi-autonomous modes.

OBJECTIVE

- To be able to adapt the robot's speed on the circuit depending on the situation and preference by altering it in the our Arduino programme.
- To utilize the IR sensor to detect the obstacle on the circuit and the line-following sensor to follow the dark line on the circuit.
- To be able to control the servo motor and the robot's movement by using the bluetooth configuration.
- To be able to switch between semi-autonomous and autonomous mode by using the bluetooth configuration.

PROBLEM STATEMENTS

- Program a robot that can move along the dark line prepared on the circuit and able to avoid obstacle and proceeding to go along the right track.
- Design a mechanism which will allow the robot to lift up the object prepared on the circuit and able to place it in a zone that has been set.
- Controlling and regulating the vehicled robot's speed and direction
- Assembling the robot's parts and connecting the connector wires to the right pins.
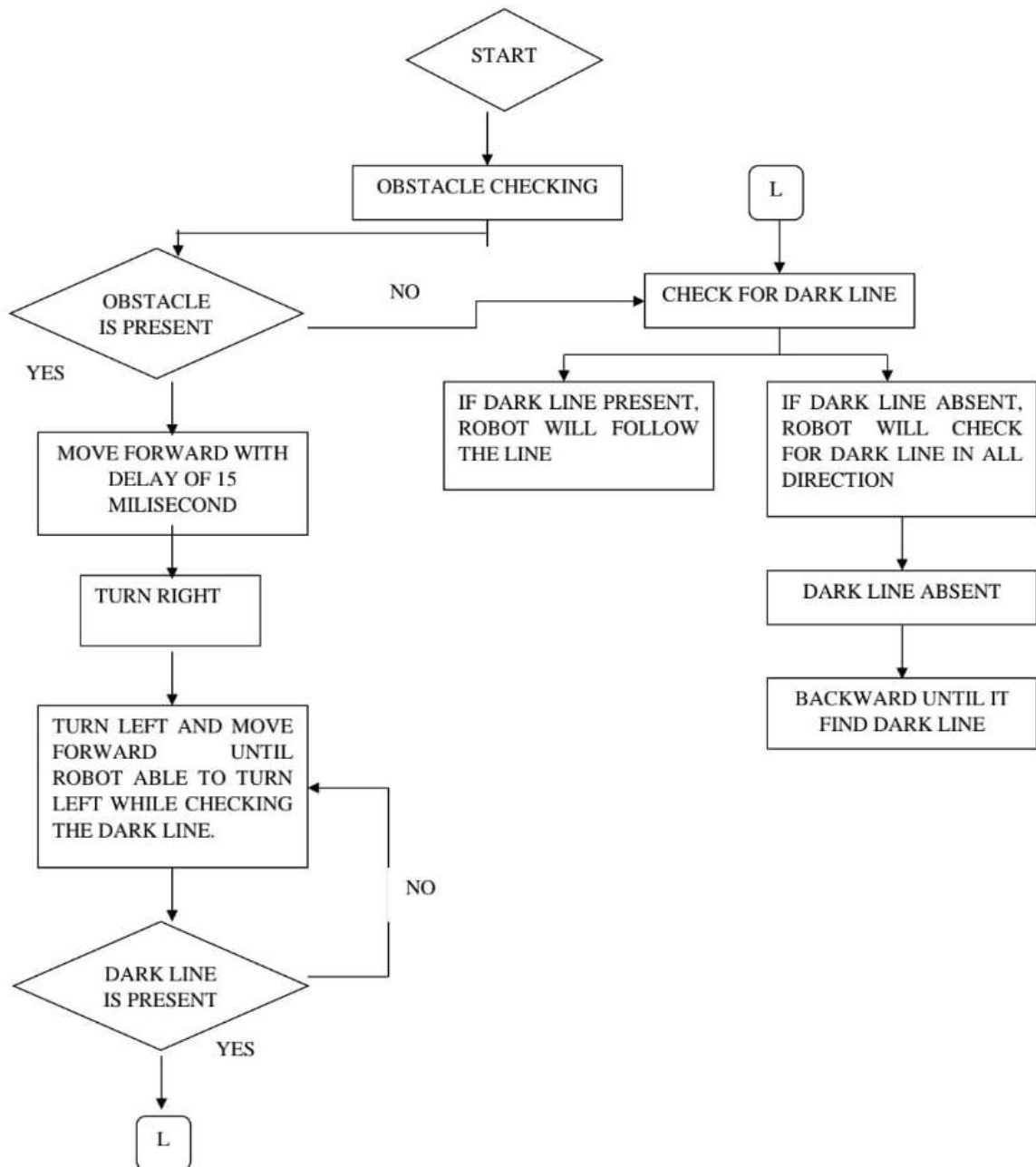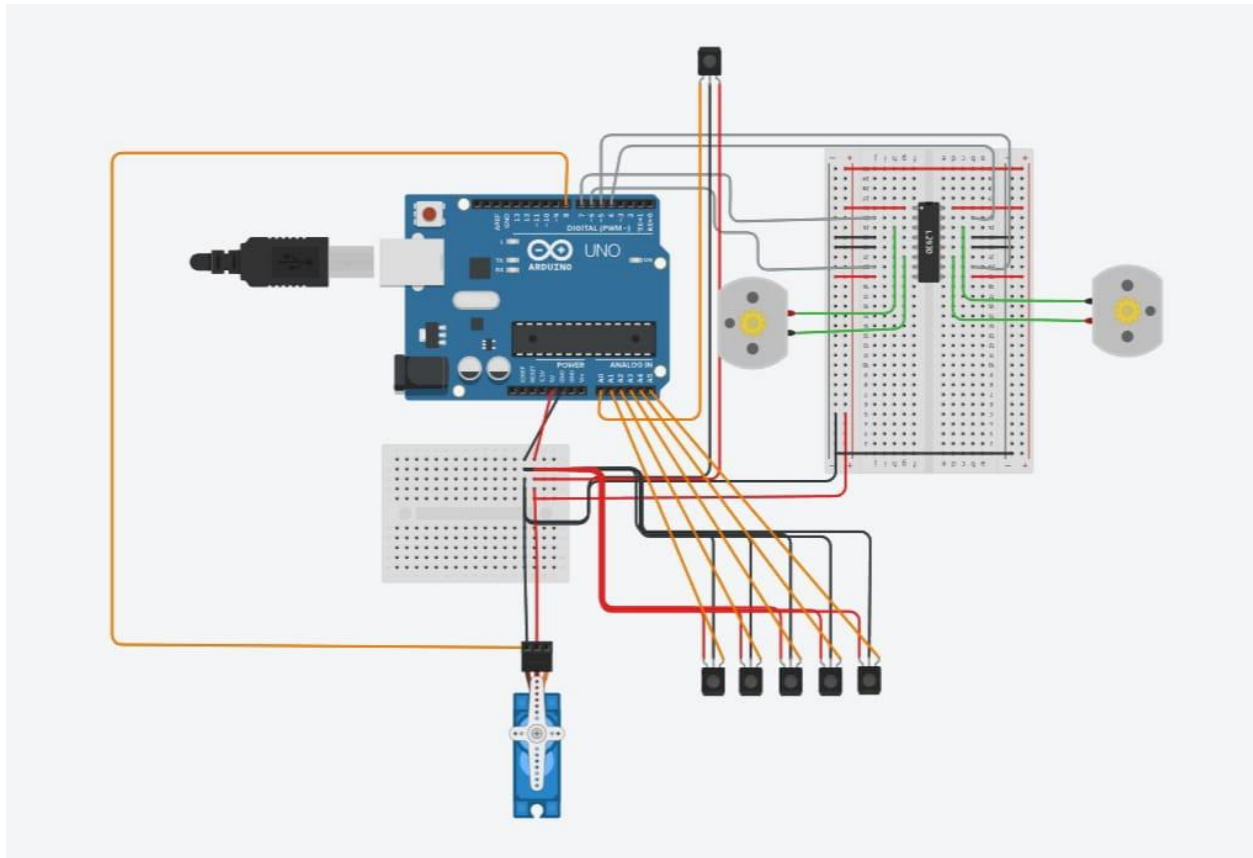
MECHANISM OF THE SYSTEM
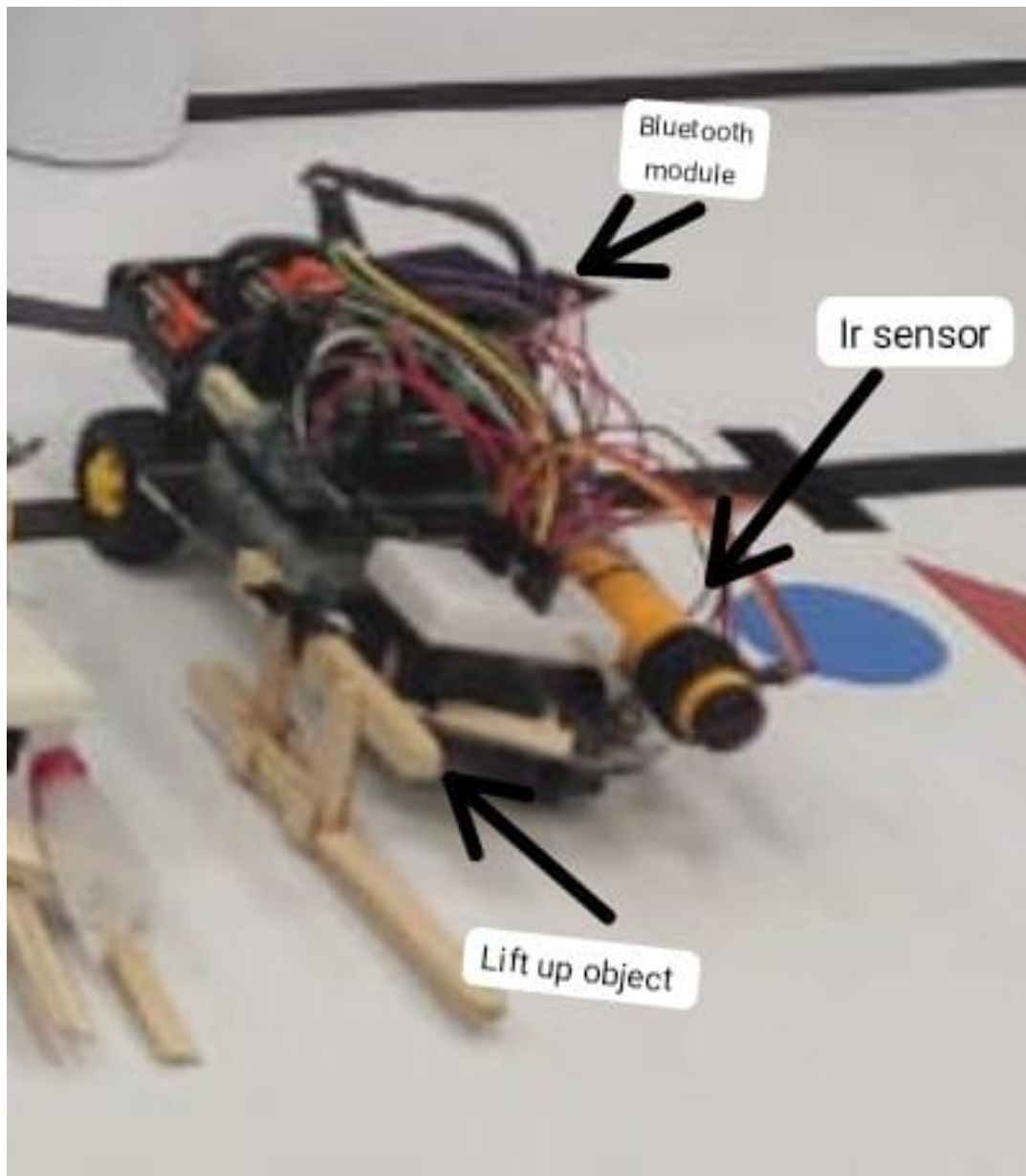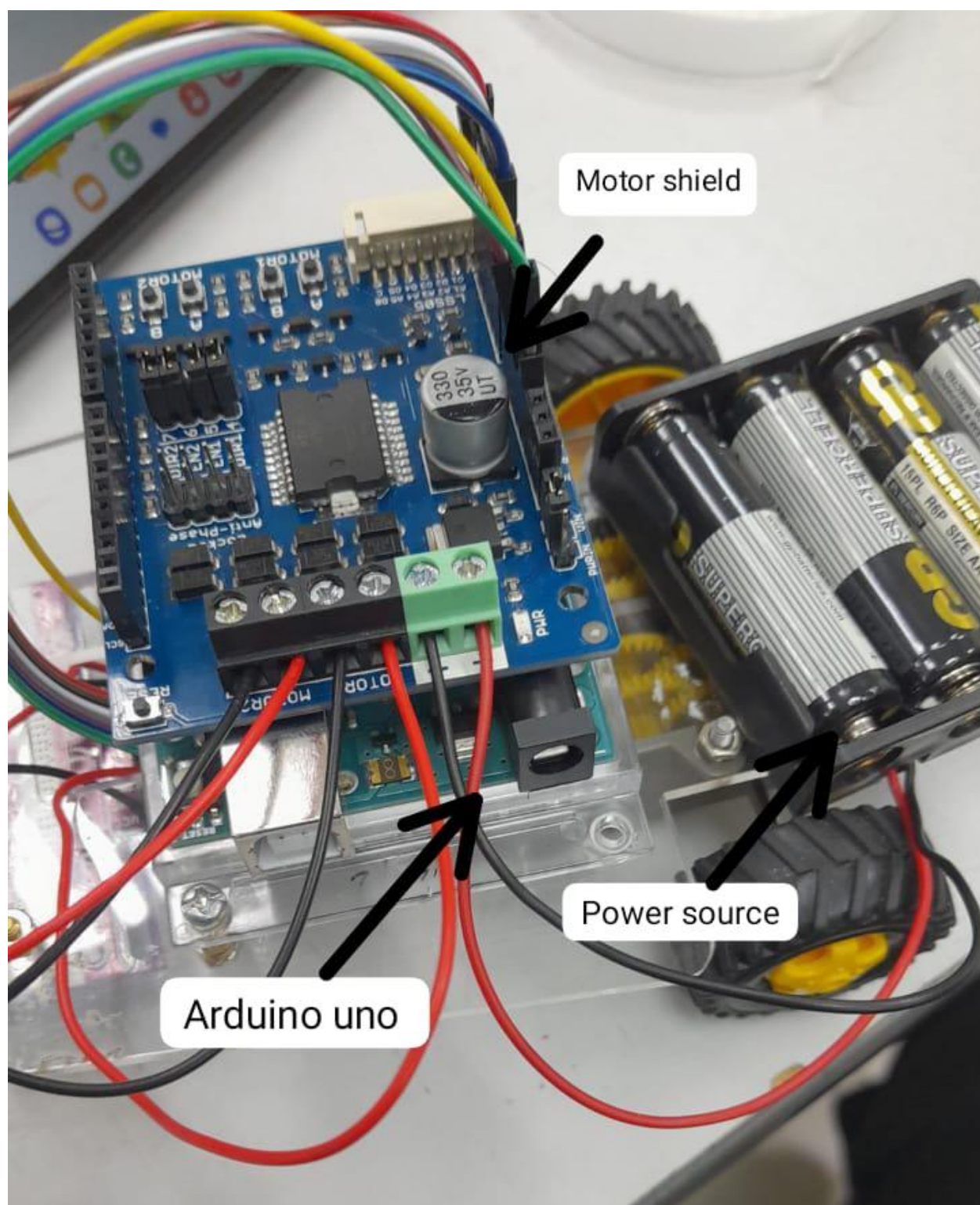
Motor shield

Power source

Arduino uno

FLOWCHART

```
                          ┌──────────────┐
                          │    START     │
                          └──────┬───────┘
                                 │
                          ┌──────▼───────────┐         ┌─────┐
                          │ OBSTACLE CHECKING│         │  L  │
                          └──────┬───────────┘         └──┬──┘
                                 │                        │
          ┌──────────────────────▼──┐   NO    ┌───────────▼──────────┐
          │   OBSTACLE IS PRESENT    │───────▶│ CHECK FOR DARK LINE  │
          └──────────────────────────┘        └──────────┬───────────┘
   YES                                      ┌─────────────┴──────────────┐
    │                                       ▼                            ▼
    ▼                          ┌──────────────────────┐  ┌────────────────────────────┐
┌─────────────────────┐        │ IF DARK LINE PRESENT,│  │ IF DARK LINE ABSENT,       │
│ MOVE FORWARD WITH    │        │ ROBOT WILL FOLLOW    │  │ ROBOT WILL CHECK FOR       │
│ DELAY OF 15          │        │ THE LINE             │  │ DARK LINE IN ALL DIRECTION │
│ MILISECOND           │        └──────────────────────┘  └─────────────┬──────────────┘
└─────────┬───────────┘                                                 ▼
          ▼                                                 ┌────────────────────────┐
┌─────────────────────┐                                     │   DARK LINE ABSENT     │
│    TURN RIGHT        │                                     └───────────┬────────────┘
└─────────┬───────────┘                                                 ▼
          ▼                                                 ┌────────────────────────┐
┌─────────────────────┐                                     │ BACKWARD UNTIL IT      │
│ TURN LEFT AND MOVE   │                                     │ FIND DARK LINE         │
│ FORWARD  UNTIL ROBOT │                                     └────────────────────────┘
│ ABLE TO TURN LEFT    │
│ WHILE CHECKING THE   │
│ DARK LINE.           │
└─────────┬───────────┘
          ▼                          NO
┌─────────────────────┐─────────────▶
│ DARK LINE IS PRESENT │
└─────────┬───────────┘
    YES    │
          ▼
       ┌─────┐
       │  L  │
       └─────┘
```

ELECTRICAL DIAGRAM

ARDUINO PROGRAMMING

```cpp
#include <Servo.h>
#include <SoftwareSerial.h>
Servo myservo;

SoftwareSerial BTSerial(2, 3);

// Pin servo = 9
// Motor pin assignments
const int L_A = 6;
const int L_B = 7;      // Left motor
const int R_A = 5;
const int R_B = 4;       // Right motor

// Line sensor pin assignments
const int p1 = A0;
const int p2 = A1;
const int p3 = A2;
const int p4 = A3;
const int p5 = A4;

// IR motion sensor
const int sensorPin = 12;  // Sensor pin

char command = 0; //command for Bluetooth

// Function for moving forward
void forward()
{
  // Left motor
  analogWrite(L_A, 255);    // Speed control
  digitalWrite(L_B, HIGH);       // Direction

  // Right motor
  analogWrite(R_A, 255);   // Speed control
  digitalWrite(R_B, HIGH);       // Direction
}

// Function for moving backward
void backward()
{
  // Left motor
  analogWrite(L_A, 200);   // Speed control
  digitalWrite(L_B, LOW);        // Direction

  // Right motor
  analogWrite(R_A, 200);    // Speed control
  digitalWrite(R_B, LOW);        // Direction
}
```

```cpp
// Function for turning right
void right()
{
  // Left motor
  analogWrite(L_A, 200);    // Speed control
  digitalWrite(L_B, HIGH);       // Direction

  // Right motor
  analogWrite(R_A, 150);    // Speed control
  digitalWrite(R_B, HIGH);        // Direction
}

void cornerRight()
{
  // Left motor
  analogWrite(L_A, 220);    // Speed control
  digitalWrite(L_B, HIGH);       // Direction

  // Right motor
  analogWrite(R_A, 0);     // Speed control
  digitalWrite(R_B, HIGH);        // Direction
}

// Function for turning left
void left()
{
  // Left motor
  analogWrite(L_A, 150);    // Speed control
  digitalWrite(L_B, HIGH);       // Direction

  // Right motor
  analogWrite(R_A, 200);    // Speed control
  digitalWrite(R_B, HIGH);        // Direction
}

void cornerleft()
{
  // Left motor
  analogWrite(L_A, 0);    // Speed control
  digitalWrite(L_B, HIGH);       // Direction

  // Right motor
  analogWrite(R_A, 220);    // Speed control
  digitalWrite(R_B, HIGH);        // Direction
}

// Function to stop the motors
```

```cpp
void stop()
{
  // Left motor
  digitalWrite(L_A, 0);          // Speed control
  digitalWrite(L_B, HIGH);       // Direction

  // Right motor
  digitalWrite(R_A, 0);          // Speed control
  digitalWrite(R_B, HIGH);       // Direction
}


void bluetooth(char)
{
  if (BTSerial.available() > 0) {  // condition when you receive data:
    command = BTSerial.read();
    Serial.println(command);
    Serial.print("\n");
  }
    //Manual Robot Control
    switch(command)
      {
        case 'F':
        forward();
        break;

        case 'B':
        backward();
        break;

        case 'S':
        stop();
        break;

        case 'L':
        left();
        break;

        case 'R':
        right();
        break;

        case 'W':
        myservo.write(180);
         break;

        case 'w':
        myservo.write(80);
        break;
```

```arduino
            case 'i':
              myservo.write(30);
              break;


              default : stop();
              }

        if (Serial.available())
        {
            char sendChar = Serial.read();
            Serial.print("Sending: ");
            Serial.println(sendChar);
            // hantar kat phone
            BTSerial.write(sendChar);
        }
}

void setup()
{
  Serial.begin(9600);
  BTSerial.begin(9600);
  Serial.println("Ready");
  myservo.attach(9);


  pinMode(p1, INPUT);
  pinMode(p2, INPUT);
  pinMode(p3, INPUT);
  pinMode(p4, INPUT);
  pinMode(p5, INPUT);
  pinMode(sensorPin, INPUT );
  pinMode(L_A, OUTPUT);
  pinMode(L_B, OUTPUT);
  pinMode(R_A, OUTPUT);
  pinMode(R_B, OUTPUT);


}

void loop()
{
 // Read sensor inputs
  int sensor1 = digitalRead(p1);
  int sensor2 = digitalRead(p2);
  int sensor3 = digitalRead(p3);
  int sensor4 = digitalRead(p4);
```

```cpp
  int sensor5 = digitalRead(p5);
  int objectdetect = digitalRead(sensorPin); // Read sensor input


  if(command == 'A')
{
    // Object not detected
    if (objectdetect == HIGH)
    {
      // Turn corner left
      if (sensor1 == HIGH && sensor2 == LOW && sensor3 == LOW && sensor4 ==
LOW && sensor5 == LOW)
      {
        cornerleft();
      }
      // Turn left
      else if (sensor1 == HIGH && sensor2 == HIGH && sensor3 == LOW && sensor4
== LOW && sensor5 == LOW)
      {
        left();
      }
      // Go forward
      else if (sensor1 == LOW && sensor2 == HIGH && sensor3 == HIGH && sensor4
== LOW && sensor5 == LOW)
      {
        forward();
      }
      // Go forward
      else if (sensor1 == LOW && sensor2 == HIGH && sensor3 == HIGH && sensor4
== HIGH && sensor5 == LOW)
      {
        forward();
      }

      // Go forward
      else if (sensor1 == LOW && sensor2 == LOW && sensor3 == HIGH && sensor4
== HIGH && sensor5 == LOW)
      {
        forward();
      }


  // Turn right
      else if (sensor1 == LOW && sensor2 == LOW && sensor3 == LOW && sensor4
== HIGH && sensor5 == HIGH)
      {
        right();
      }
```

```cpp
      // Right corner
      else if (sensor1 == LOW && sensor2 == LOW && sensor3 == LOW && sensor4
== LOW && sensor5 == HIGH)
      {
        cornerRight();
      }
      else if (sensor1 == LOW && sensor2 == LOW && sensor3 == LOW && sensor4
== LOW && sensor5 == LOW)
      {
        backward();
      }
    }
    // If an object is detected in front
    else
    {
      forward();
      delay(15);
      right();
      delay (100);
      left();
      delay(50);
      forward();
      delay(100);


    }
  }
  else // Bluetooth control mode
  {
    if(BTSerial.available())
    {
      command = BTSerial.read();
      Serial.println(command);
      Serial.print("\n");
      bluetooth(command);
    }
  }
}
```

DISCUSSION

To be fully honest, our group has significant internal issues, including poor communication between group members, which prevents the robot from functioning properly (in bluetooth mode). Our group received some helpful advise from our lecturers as a result of this issue. This is a great opportunity for us in terms of how we might develop as a group.

Returning to the lab report, we are happy with how the project turned out, although it may have been better if our group had been effectively cooperating.We broke each task into smaller pieces so that it could be finished, yet each includes flaws that affect the outcome. It required a great deal of perseverance and trial and error.

The recommendation is that it will be very helpful and a lot more enjoyable and interesting if the project is released earlier to the students to increase the quality of our project and giving us some chance to create a very competitive environment in the class.

From the islamic point of view, we can see that this project had what it takes to teach us the importance of patience in developing out robot. Building the robot parts and programme is a very time-consuming process and we do encountered a lot of problem during our journey in this project but finally we completed the project successfully as we kept being patient dealing with it. We can see that in a hadith, our beloved Prophet Muhammad (S.A.W) said that "Whoever persists in being patient, Allah will make him patient.". This hadith showed that Allah will reward those who keep being patient even at their hardest.

Finally, even if what happened has already happened, we are nevertheless glad to state that we are happy with what we have done. In Sha Allah, we shall be able to do even better in the future.

CONCLUSION

 In conclusion, Arduino is a software programme, hardware, and hardware platform that may be used to carry out a variety of activities. For our project, made a car robot that is capable of fully and partially autonomous driving. People who aren't involved in engineering field of study should know about this awesome project because it is so much fun in general. This is because it can inspire us to think of fresh concepts and boost our creativity, particularly in our daily lives. Everything else is really helpful and has greatly increased our knowledge of the field of mechatronics and robotics. Finally, we would like to express our gratitude to all of the instructors involved in helping us excel in this topic.