AMRITA VISHWA VIDYAPEETHAM

DEEMED TO BE UNIVERSITY UNDER SECTION 3 OF THE UGC ACT, 1956

AMRITA SCHOOL OF ENGINEERING, BENGALURU

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

B. TECH. II SEMESTER(ECE)

AY 2023-2024

23ECE184

INTRODUCTION TO INTERNET OF THINGS

LAB SYSTEM DESIGN REPORT

MAY 2024

Submitted by:

BL.EN. U4ECE23054 Harshavardhan reddy

BL.EN. U4ECE23055 Yakkanti V S Siddhartha Reddy

BL.EN. U4ECE23057 Chaitanya

BATCH NUMBER: 07

Faculty Examiner Signature with Date:

Comments (If any):

TITLE: VOICE CONTROL ROBOT

OBJECTIVE:

Our project aims to improve the mobility features of a car by enhancing it with obstacle avoiding feature and voice control set up with an assigned app for the respective.

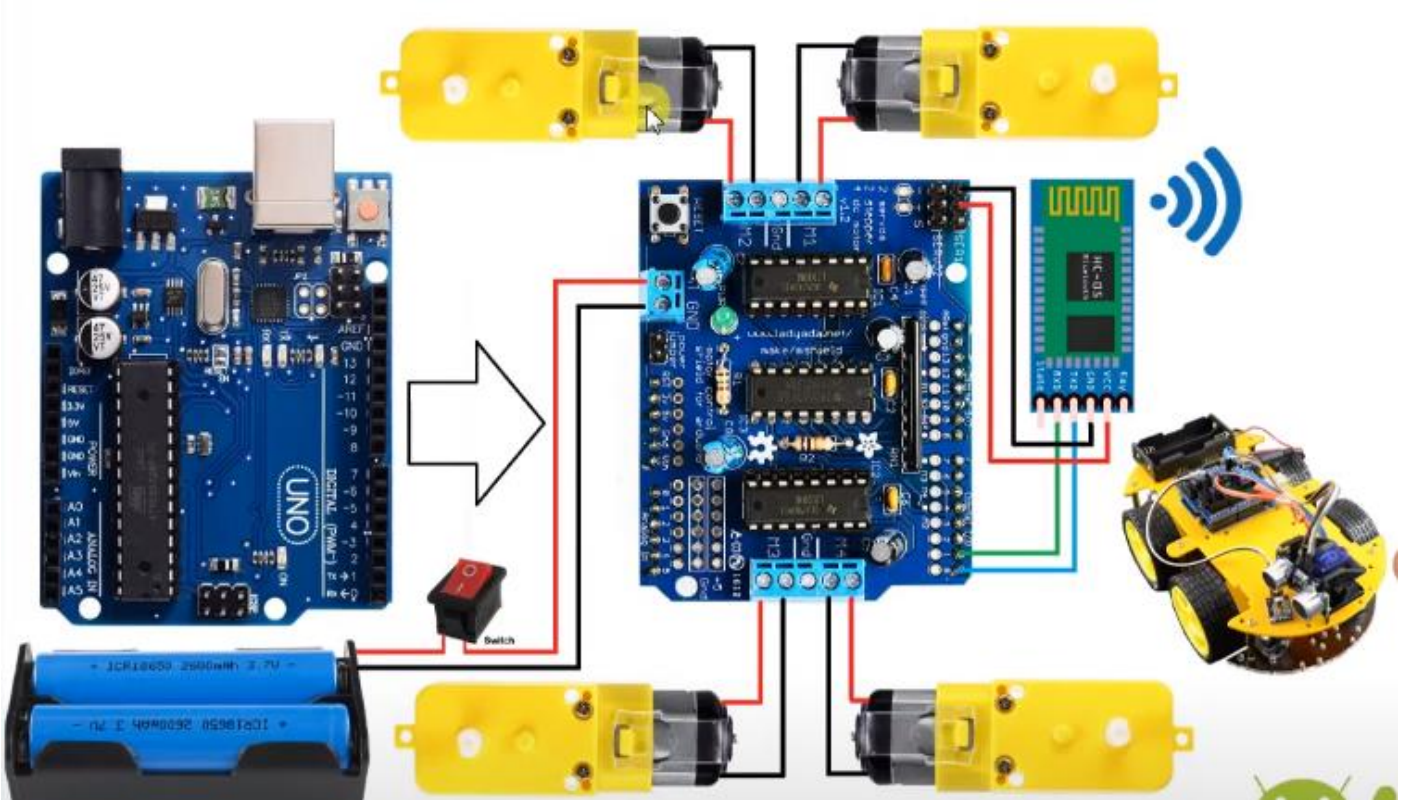
COMPONENTS REQUIRED ALONG WITH THEIR SPECIFICATIONS:

SI.NO	Name of the Component	Specification	Quantity
1	Arduino UNO	R3 SMD board	**1**
2	Gear motor		4
3	Robot wheel		4
4	Motor control shield	L293D	1
5	Bluetooth module	HC -06	1
6	Battery Cell	18650 3.7V	2
8	Jumper wires (*Female to female)*		50

MOTIVATION/SOCIATAL/INDUSTRIAL IMPACT:

As students, we chose this project for its potential to address urban mobility challenges and foster sustainability. The use of voice control in cars holds immense potential to transform both society and industry. When cars autonomously navigate around obstacles it significantly reduces accidents and fatalities, particularly for vulnerable populations. This technology could also empower individuals with physical limitations to drive independently, fostering greater mobility and freedom. Furthermore, autonomous delivery vehicles could revolutionize logistics, optimizing traffic flow and potentially reducing. In essence, this project has the power to create a safer, more accessible, and efficient future across various societal and industrial domains.

BLOCK DIAGRAM/SYSTEM DESIGN:



CODE

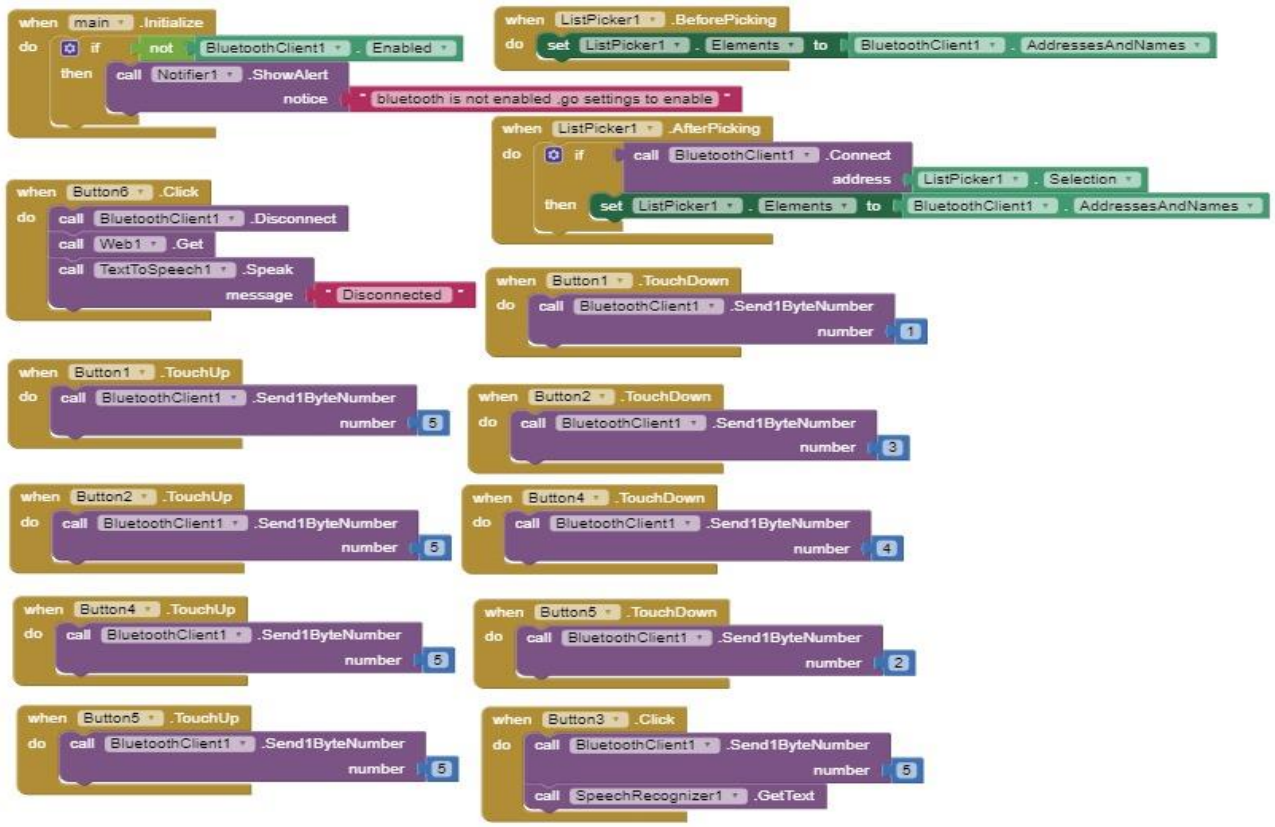
Arduino code:

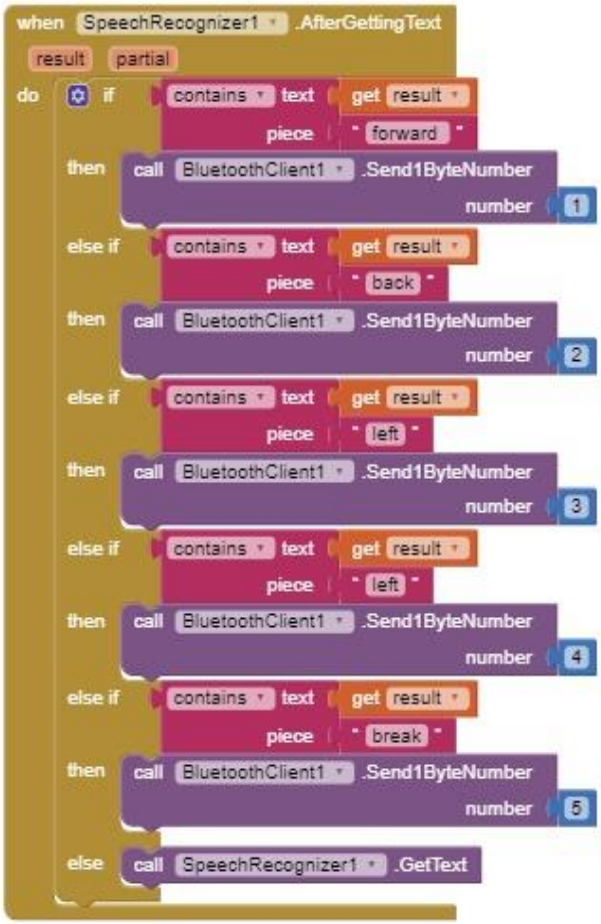
```
int t;


void setup() {
pinMode(13, OUTPUT);   //left motors  forward
pinMode(12, OUTPUT);   //left motors reverse
pinMode(11, OUTPUT);   //right  motors forward
pinMode(10, OUTPUT);   //right motors reverse
Serial.begin(9600);


}


void loop () {
if (Serial.available()){
```

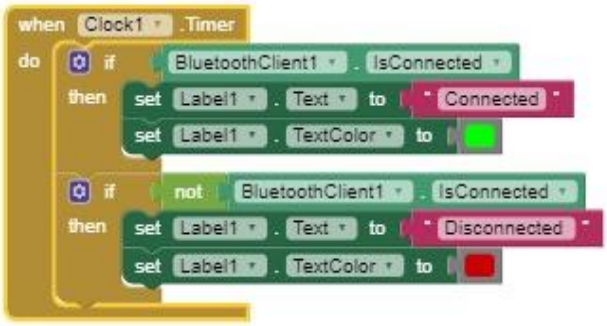
```
  t = Serial.read();
  Serial.println(t);
}

if(t == 1){          //move  forward(all motors rotate in forward direction)
  digitalWrite(13,HIGH);
  digitalWrite(12,LOW);
  digitalWrite(11,HIGH);
  digitalWrite(10,LOW);
}

else if(t == 2){     //move reverse (all  motors rotate in reverse direction)
  digitalWrite(13,LOW);
  digitalWrite(12,HIGH);
  digitalWrite(11,LOW);
  digitalWrite(10,HIGH);
}

else if(t == 3){
  digitalWrite(13,HIGH);
  digitalWrite(12,LOW);
  digitalWrite(11,LOW);
  digitalWrite(10,LOW);    //turn right (left side motors rotate in forward direction,  right side motors doesn't rotate)
}

else  if(t == 4){     //turn left (right side motors rotate in forward direction, left  side motors doesn't rotate)
  digitalWrite(11,HIGH);
  digitalWrite(13,LOW);
```

```
  digitalWrite(12,LOW);
  digitalWrite(10,LOW);
}

else if(t == 5){      //STOP (all motors stop)
  digitalWrite(13,LOW);
  digitalWrite(12,LOW);
  digitalWrite(11,LOW);
  digitalWrite(10,LOW);
}
}
```

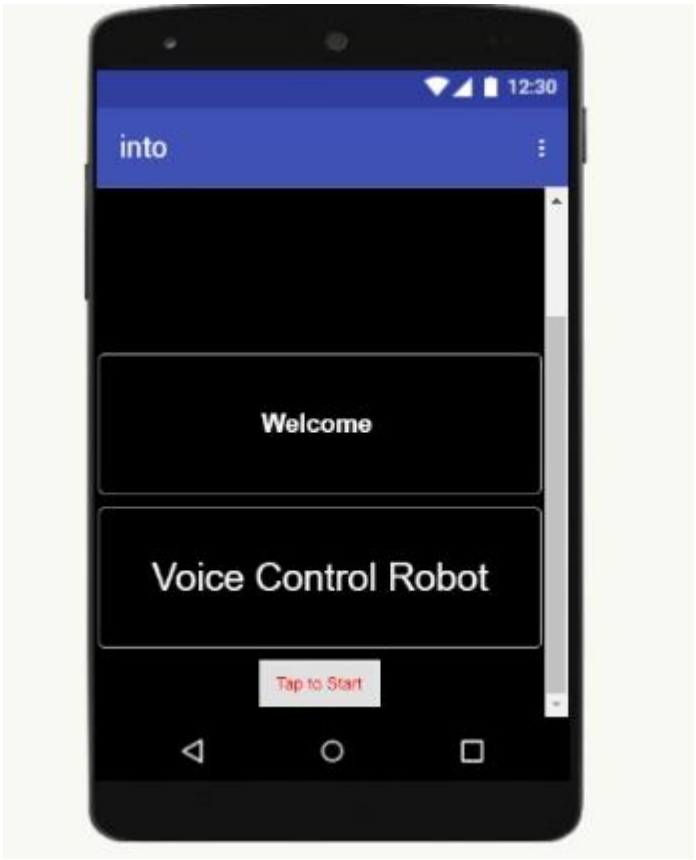
MIT APP code:



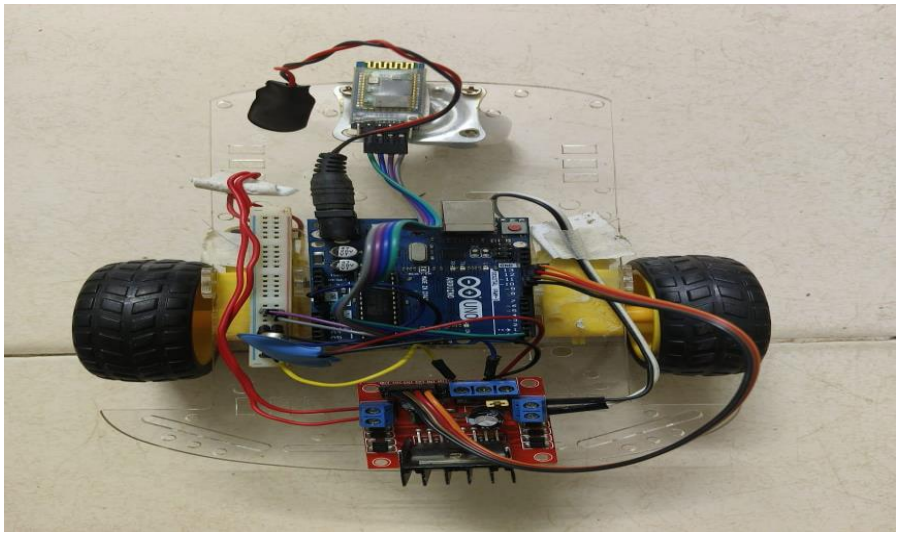

when SpeechRecognizer1 .AfterGettingText
result partial
do if contains text get result
piece " forward "
then call BluetoothClient1 .Send1ByteNumber
number 1
else if contains text get result
piece " back "
then call BluetoothClient1 .Send1ByteNumber
number 2
else if contains text get result
piece " left "
then call BluetoothClient1 .Send1ByteNumber
number 3
else if contains text get result
piece " left "
then call BluetoothClient1 .Send1ByteNumber
number 4
else if contains text get result
piece " break "
then call BluetoothClient1 .Send1ByteNumber
number 5
else call SpeechRecognizer1 .GetText

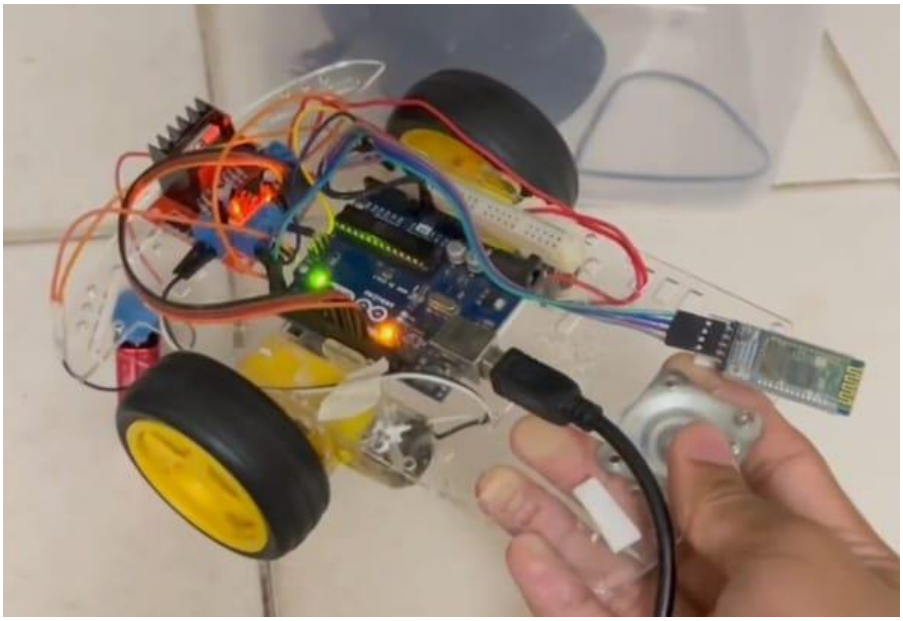

when Clock1 .Timer
do if BluetoothClient1 . IsConnected
then set Label1 . Text to " Connected "
set Label1 . TextColor to
if not BluetoothClient1 . IsConnected
then set Label1 . Text to " Disconnected "
set Label1 . TextColor to


when Button1 .Click
do open another screen screenName main



SCREENSHOT OF ALL THE WORKING CIRCUITS WITH OUTPUT:





RESULT ANALYSIS:

After analysing the data from the voice control robot implemented with Arduino and Bluetooth technology, involves a fascinating interplay of sensors and voice recognition. Ultrasonic sensors constantly scan for obstacles, while voice commands like "forward," "turn," or "stop" are received and interpreted. Motors adjust direction and speed based on detected obstacles and voice instructions.

CONCLUSION:

Our project has effectively implemented building a voice-controlled car with Arduino allows you to steer it using spoken commands like "forward," "backward," "left," and "right." The Arduino board receives and interprets these commands, then controls the car's motors accordingly. This creates a fun and interactive project, showcasing the power of Arduino in combining voice recognition and basic robotics.

PROBLEMS FACED DURING THE IMPLEMENTATION (IF ANY):

- The MIT app code was only recognizing canvas for the first condition code-block due to use of one time only.
- Coding the logic in MIT app inventor was complicated.

FUTURE SCOPE:

The future of voice-controlled Arduino robots is bright, with potential for significant advancements: integrating Natural Language Processing could enable more natural and complex commands like "follow me" or "park the car." Additionally, combining voice control with obstacle detection sensors would create a fully autonomous mini-vehicle, navigating its surroundings independently.