ANDROID APPLICATION-BASED ROBOTIC CAR USING ARDUINO AND MIT APP INVENTOR

SAHANA.HC.
BHEEMAVVA.
PREETI.
MEGHANA.A.

INTRODUCTION:

- The Android Application-Based Robotic Car project aims to design and develop a robotic car controlled via an Android application built using MIT App Inventor.
- This project leverages the simplicity and versatility of the Arduino Uno microcontroller, the power of the L298N motor driver shield, and the convenience of Bluetooth communication to create an interactive and educational robotic system.

OBJECTIVES:

DESIGN AND BUILD THE ROBOTIC CAR CHASSIS:

Develop a sturdy and efficient mechanical structure for the car.

INTEGRATE MOTOR CONTROL:

Implement the L298N motor driver shield to control the DC motors.

DEVELOP AN ANDROID APPLICATION:

Use MIT App Inventor to create an intuitive and user-friendly app to control the robotic car.

IMPLEMENT BLUETOOTH COMMUNICATION:

Use the HC-05 Bluetooth module to enable communication between the Android app and the ArduinoUno.

ASSEMBLE AND CONNECT ALL COMPONENETS:

USE JUMPING WIRES TO CONNECT THE ARDUINO, MOTOR DRIVER, MOTORS, BLUETOOTH MODULE, AND POWER SUPPLY.

TOOLS REQUIRED :

HARDWARE:

- Arduino Uno
- L298N motor driver shield
- DC motors
- Jumping wires
- Wheels
- HC-05 Bluetooth module
- Battery
- Chassis and mechanical components

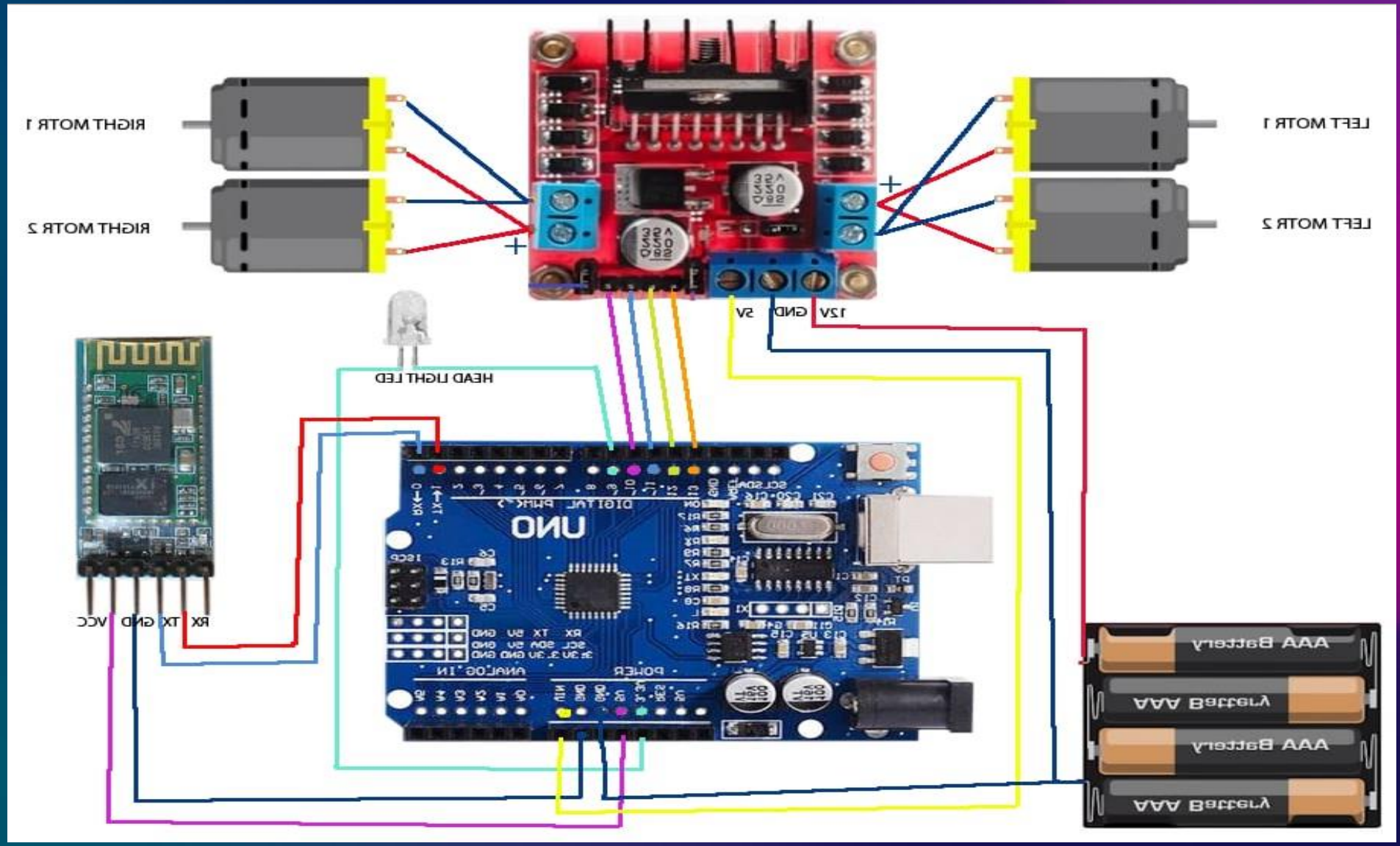
SOFTWARE:

- MIT App Inventor
- Arduino IDE

APPLICATIONS:

- **Practical applications:** Useful in areas such as surveillance, search and rescue operations, and entertainment.
- **Educational tool:** Ideal for learning about robotics, mobile computing, and Bluetooth communication.
- **Advanced robotics concepts:** A platform for experimenting with concepts like autonomous navigation and obstacle avoidance.
- **Search and Rescue Operations**:Assists rescue teams by providing real-time visuals and data transmission, improving coordination and efficiency.

CIRCUIT DIAGRAM:



METHODOLOGY:

1.Electronics and Control System:

- **Arduino Uno:** Acts as the brain of the robotic car, controlling the motors and processing commands from the Bluetooth module.
- **L298N motor driver shield:** Connect to the Arduino to enable precise control of the car's movement.
- **HC-05 Bluetooth module:** Allows wireless communication between the Android app and the Arduino, enabling remote control.

2.Android Application Development:

- **MIT App Inventor:** Use this platform to design a user-friendly interface that allows easy control of the robotic car.
- **Bluetooth communication:** Implement within the app to send commands to the Arduino, enabling real-time control.

3.Software Development:

- **Arduino code:** Write code to interpret commands from the Android app and control the car's movement.
- **Navigation and obstacle avoidance:** Develop basic algorithms if sensors are used, enabling the car to navigate and avoid obstacles autonomously.

4.Component Assembly:

- **Connect DC motors:** Attach the motors to the L298N motor driver shield.
- **Connect motor driver shield:** Link the shield to the Arduino Uno.
- **Connect HC-05 Bluetooth module:** Set up the Bluetooth module for wireless control.
- **Jumping wires:** Use wires to make all necessary connections between components.
- **Power the system:** Use a suitable battery to power the entire system.

CODE:

```
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
pinMode(9,OUTPUT);   //Led
Serial.begin(9600);

}

void loop() {
if(Serial.available()){
  t = Serial.read();
  Serial.println(t);
}

if(t == 'F'){           //move  forward(all motors rotate in forward direction)
  digitalWrite(13,HIGH);
  digitalWrite(11,HIGH);
}
else if(t == 'B'){      //move reverse (all  motors rotate in reverse direction)
  digitalWrite(12,HIGH);
  digitalWrite(10,HIGH);
}
```

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

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

else if(t ==  'W'){    //turn led on or off)
  digitalWrite(9,HIGH);
}
else if(t == 'w'){
  digitalWrite(9,LOW);
}

else if(t == 'S'){     //STOP (all motors stop)
  digitalWrite(13,LOW);
  digitalWrite(12,LOW);
  digitalWrite(11,LOW);
  digitalWrite(10,LOW);
}
delay(100);
}
```

ADVANTAGES:

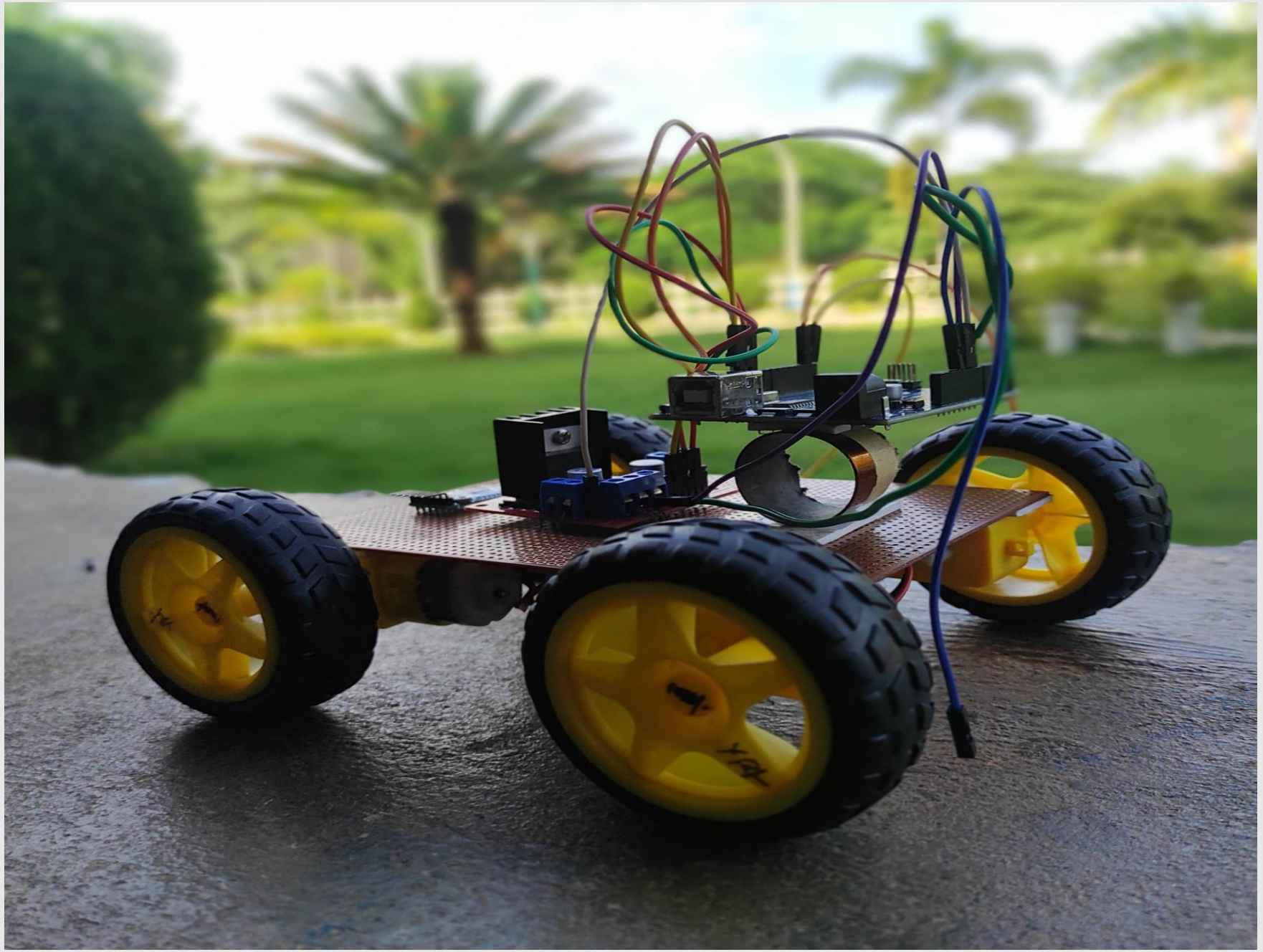
- Hands-on Learning Experience: Enhances understanding of theoretical concepts through real-world application.
- Cost-Effective Solution: Makes advanced robotics accessible without significant investment.
- Flexibility and Customizability: Encourages creativity and continuous project enhancement.
- Wireless Control and Mobility: Provides convenience and flexibility in operation
- Educational Value: Prepares students for future careers in STEM fields.
- Versatile Applications: Demonstrates wide-ranging potential in various industries.

DIS-ADVANTAGES:

- Limited Processing Power: May not handle complex tasks efficiently.
- Range Constraints: Limits operational distance to around 10 meters.
- Battery Dependency: Requires frequent recharging or battery replacement.
- Basic Functionality: May lack advanced features like autonomous navigation.
- Environmental Limitations: Performs poorly in harsh or uneven terrains.
- Technical Skill Requirement: Challenging for beginners without technical knowledge.

Expected Outcomes:

- **Functional robotic car:** A robotic car that can be controlled remotely via an Android application.
- **Real-time response:** Immediate reaction to control commands sent from the Android app through Bluetooth.
- **User-friendly app:** An intuitive interface developed with MIT App Inventor for controlling the robotic car.



ROBOTIC CAR

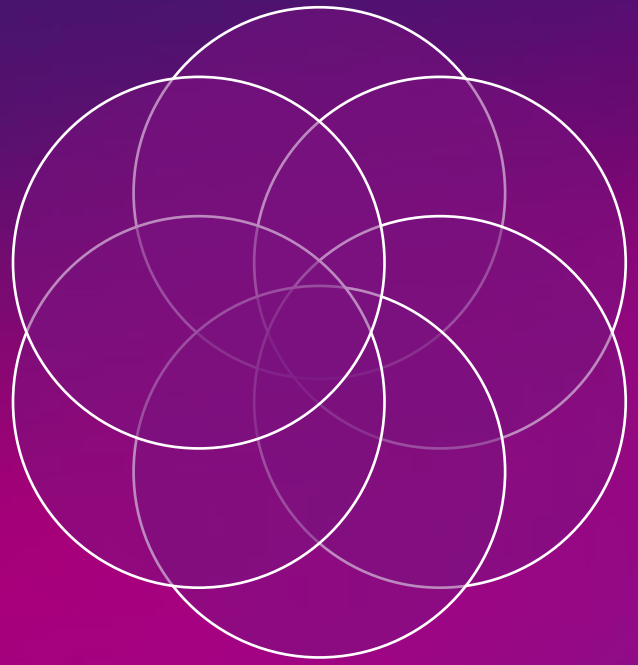
REFERENCES:

- Arduino. "Arduino Uno." www.arduino.cc/en/Main/ArduinoBoardUno.
- MIT App Inventor. "Tutorials." appinventor.mit.edu/explore/tutorials.
- Arduino. "Motor Driver Shield L298N." www.arduino.cc/en/Main/ArduinoMotorShieldR3.
- Random Nerd Tutorials. "Complete Guide for HC-05 Bluetooth Module with Arduino." randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04.
- Simon Monk. "Programming Arduino: Getting Started with Sketches." McGraw-Hill Education, 2011.
- Adafruit Learning System. "Battery Power for your Projects." learn.adafruit.com/battery-power.

FUTURE SCOPE:

Implement advanced algorithms for autonomous navigation, allowing the robotic car to navigate and avoid obstacles independently.

Adapt the robotic car for specific applications like warehouse automation, agriculture monitoring, and delivery services.



Add sensors such as ultrasonic, infrared, or LIDAR to enhance environmental detection and response capabilities.

Explore efficient power management and renewable energy sources to extend the robotic car's operational time.

Integrate IoT connectivity for remote monitoring and control over the internet, expanding operational range and applications.

Develop a more sophisticated Android app with features like voice control and gesture recognition for enhanced user interaction.

CONCLUSION:

- **Integration of mobile and robotic technologies:** Demonstrates the potential for remote control and automation.
- **Foundation for future developments:** Provides a basis for future advancements in robotics, showcasing seamless interaction between hardware and software.


Any
Question?

THANKS A LOT!