VOICE CONTROLLED BLUETOOTH CAR

A REPORT

Submitted in partial fulfilment of the requirement for the award of degree

Of

BACHELOR OF ENG.

IN

ELECTRONICS & COMMUNICATION ENGINEERING

SUBMITTED BY

HARSH & PRATEEM

(Roll Number: 2150-39 & 70)

UNDER THE GUIDANCE OF

Dr. CHARU MADHU

(Assistant Professor, ECE, UIET)

UNIVERSITY INSTITUTE OF ENGINEERING AND TECHNOLOGY
PANJAB UNIVERSITY CHANDIGARH 160014

ACKNOWLEDGEMENT

We would like to extend our sincere gratitude to all those who have supported and guided us throughout the development of our voice-controlled humanoid robot.

First and foremost, we express our deepest appreciation to **Dr. Charu Madhu**, whose expertise, guidance, and encouragement have been invaluable to us. Her insightful suggestions and constant support helped us navigate through various challenges and enriched our learning experience.

We also want to thank our families and friends for their unwavering support and encouragement, which has been a constant source of motivation.

Finally, we acknowledge each other, **Jeevan** and **Harsh**, for our dedication, teamwork, and relentless pursuit of excellence in bringing this project to fruition. It has been a rewarding journey of innovation and collaboration.

Thank you all for making this project a success.

Table of Contents

INTRODUCTION

In recent years, advancements in artificial intelligence and robotics have paved the way for innovative developments that aim to enhance human life. Among these, humanoid robots have garnered significant

attention due to their potential to perform tasks in a human-like manner, offering immense benefits across various domains such as healthcare, education, and personal assistance.

Our project focuses on the development of a **voice-controlled humanoid robot**. This robot is designed to understand and respond to voice commands, enabling intuitive and seamless interaction between humans and machines. By leveraging advanced speech recognition and processing algorithms, our robot can perform a variety of tasks ranging from simple movements to more complex actions, making it a versatile tool for numerous applications.

1.1 Project Description

The combination of major fields such as mechanical, Electrical, and electronics provide automation systems which are known as Robots. The growths in these industries are a major reason for the efficiencies in every sector by reducing human effort and interaction. Doings such, as this promises us a safer environment on Dangerous and insightful grounds. Due to its precision and Absolute accuracy, it has made a major presence in all the Essential fields whether it is education, bio-medicals, Engineering.

The primary objective of this project is to create a robot that not only mimics human actions but also understands human language, thereby facilitating a more natural and efficient means of communication. Our humanoid robot is equipped with a sophisticated auditory system that can discern commands in different environments and respond appropriately, demonstrating the potential of voice control in robotic systems.

Throughout the development process, we have focused on integrating cutting-edge technologies to ensure the robot's functionality, reliability, and user-friendliness. This project stands as a testament to the collaborative efforts of our team and the guidance of our mentor, reflecting our commitment to pushing the boundaries of what's possible in the realm of robotics.

1.2 Main Features:

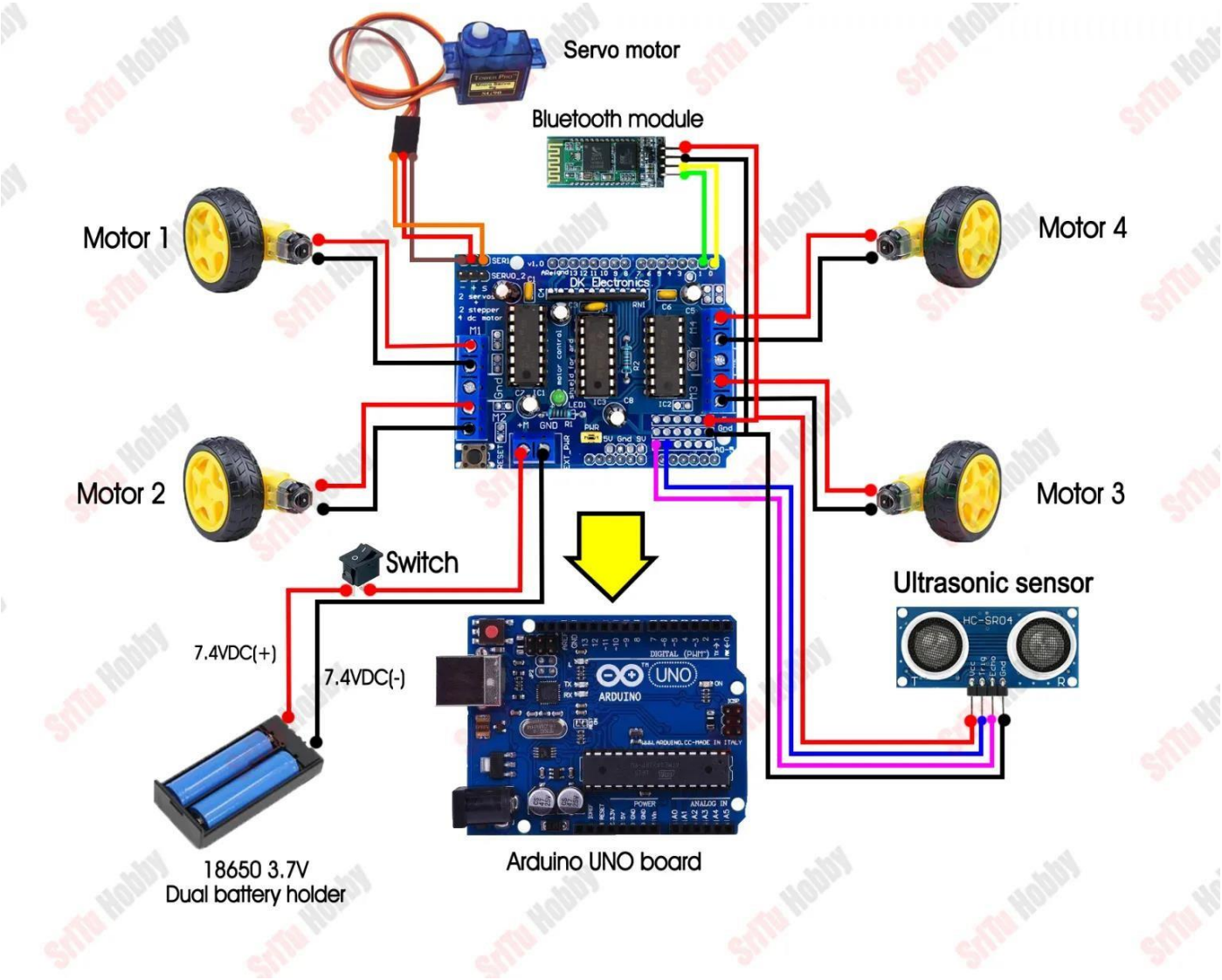
- **Obstacle avoiding**

To achieve obstacle avoidance in the system we are using an Ultrasonic distance sensor which is connected to the L298D motor s shield followed by Arduino UNO. This sensor is used to detect any object at some certain distance using so nearby. This non-contact ultrasound sonar is used to measure the distance between the object and the sensor. It consists of two Transmitters, a control circuit, and a receiver for emitting and receiving pulse data respectively. A high ultrasonic sound is emitted by transmitters which will get reflected by any nearby object and the sensor will see toward to get any return echo. The distance for object detection can be subsequently changed by our means in the coding algorithm. The distance will get calculated for the transmitted signal and receive an echo in the control system.

- **Speech recognition**

The ability of the machine to receive and interpret the human voice or to understand and carry out spoken commands can be concluded as speech recognition. It works based on algorithms codes that match the sound of the detected speech or voice with word Sequences and interpret it as a command in Arduino IDE which is a coding platform for Arduino UNO. With the help of these, we can command our system as per the desired needs.

1.3 Circuit Diagram/Image:



COMPONENTS DESCRIPTION

2.1 Components Used:

Hardware Components: ○

Arduino Uno R3 ○ Motor

Driver Shield ○ Wheels (DC)

Motors ○ Servo Motors ○

Ultrasonic Sensor ○ 7.4 Volt

Lithium-Ion Battery ○ Battery

Holder ○ Jumper and Normal

Wires ○ HC-05 Bluetooth

Module

- Double Sided Tape

Software Components:

- Arduino IDE
- Arduino Bluetooth Controller App

2.2 Components Explanation:

Arduino Uno R3: The Arduino Uno R3 is a widely used microcontroller board that is highly regarded for its ease of use and versatility, making it suitable for both beginners and experienced users in electronics projects. At its core, it features the ATmega328P microcontroller, which operates at 5V and offers 32 KB of flash memory for storing code, 2 KB of SRAM, and 1 KB of EEPROM. The board provides 14 digital I/O pins, with 6 capable of PWM output, and 6 analog input pins for reading signals from sensors. It includes a USB connection for programming and power, an external power jack, a reset button, and various LED indicators for power and data transmission. The 16 MHz crystal oscillator ensures accurate timing, while the voltage regulator maintains a stable 5V supply. The Arduino Uno R3 is programmed using the Arduino IDE, which employs a simplified version of C++, and supports standard communication protocols like SPI, I2C, and UART. It can be expanded with shields and modules to add functionality such as motor control and wireless communication, making it a flexible tool for a wide range of electronic applications.

HC-05 Bluetooth Module: The HC-05 Bluetooth module is a widely used transceiver module that facilitates wireless serial communication between microcontrollers, such as Arduino, and other Bluetooth-enabled devices. Operating on Bluetooth 2.0 with Enhanced Data Rate (EDR), it supports reliable communication up to 10 meters. The module uses UART (Universal Asynchronous Receiver/Transmitter) protocol for serial communication, typically at a default baud rate of 9600 bps, which can be configured using AT commands. It can function as either a master or slave, allowing it to initiate or accept connections. The HC-05 operates at 3.3V, though it is often powered by 5V in projects, and features pins for power (VCC), ground

(GND), data transmission (TXD), data reception (RXD), and configuration (Key). This module is ideal for applications such as wireless communication, remote control systems, and data logging, making it a popular choice for DIY electronics and educational projects due to its simplicity and versatility**.**

HS-04 Ultrasonic Sensor: The HC-SR05 ultrasonic sensor is a reliable device used for measuring distances from 2 cm to 450 cm with an accuracy of ±3 mm. Operating at 5V, it emits 40 kHz ultrasonic waves and calculates the time taken for the echo to return after hitting an object. It has four pins: VCC (power), GND (ground), Trig (trigger), and Echo (echo). To measure distance, a microcontroller like an Arduino sends a pulse to the Trig pin, triggering the sensor to emit sound waves, and the Echo pin receives the reflected signal. The time interval between sending and receiving the pulse is used to calculate the distance, making the HC-SR05 ideal for applications in robotics, automation, and obstacle detection.

Motor Driver Shield: A motor driver shield is an add-on board for microcontrollers, such as Arduino, designed to control DC motors, stepper motors, and servos. It simplifies the process of motor control by providing built-in motor drivers, usually based on chips like the L298N or L293D, which can handle higher currents and voltages required by motors. The shield typically includes terminals for motor connections, power inputs, and control pins that interface with the microcontroller. It allows for easy control of motor direction and speed through simple code, making it ideal for robotics and other motor-driven projects.

Arduino IDE: The Arduino IDE (Integrated Development Environment) is a software tool used for writing, compiling, and uploading code to Arduino microcontroller boards. It provides a user-friendly interface for programming, with features like syntax highlighting, auto-completion, and a serial monitor for debugging. The IDE supports the Arduino programming language, which is based on C/C++, making it accessible for beginners while also offering advanced features for experienced users. With the Arduino IDE, users can quickly develop and deploy code for a wide range of electronics projects.

Arduino Bluetooth Controller APP: An Arduino Bluetooth controller app is a mobile application designed to communicate with an Arduino board wirelessly via Bluetooth. Typically, such apps allow users to send commands or data from their mobile device to the Arduino board, enabling them to control various functions or devices connected to the Arduino. This app is used to give different voice command to the Robot perform different operations such as to move in right, left, forward and backward directions.

Servo Motor: A servo motor is a rotary actuator that enables precise control of angular position. It consists of a motor coupled with a feedback mechanism, typically a potentiometer. This feedback mechanism allows the servo motor to accurately maintain its position. Servo motors are commonly used in applications where precise control of movement is required, such as robotics, RC vehicles, and industrial automation. They are controlled using a PWM (Pulse Width Modulation) signal, with the width of the pulse determining the position of the motor shaft. Servo motors are known for their compact size, ease of control, and ability to provide variable torque depending on the design. They are widely used due to their reliability and accuracy in positioning tasks.

7.4 Volt Lithium Ion Battery: A lithium-ion battery (Li-ion) is a rechargeable type of battery that uses lithium ions as the primary component of its electrochemistry. It's widely used in portable electronics, electric vehicles, and renewable energy storage systems due to its high energy density, long cycle life, and relatively low self-discharge rate. It is used as to give power supply to the 4 Wheel DC motors.

Jumper Wire: Jumper wires are electrical wires used to create connections between various electronic components on a breadboard or between different points on a circuit board. They typically consist of a flexible wire with metal pins or connectors at each end. Jumper wires are available in various lengths and colors, making it easy to organize and distinguish connections in a circuit.

Battery Holder: A battery holder is a device designed to securely hold batteries in place and provide electrical connections for powering electronic devices. Battery holders come in various shapes and sizes, depending on the type and number of batteries they are intended to hold. They are commonly used in electronic projects, toys, remote controls, and other batteryoperated devices.

Double Sided Tape: Double-sided tape is an adhesive tape coated with adhesive on both sides, allowing it to bond two surfaces together. It is commonly used for mounting or attaching objects to surfaces where traditional fasteners like screws, nails, or glue may not be suitable or desired.

ARDUINO SKETCH/CODE

Arduino Sketch/Code for the Project:

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

#define Echo A0
#define Trig A1
#define motor 10
#define Speed 175
#define spoint 103
  char value;
int distance;
int Left;
int Right;
int L = 0;
int R = 0;
Servo servo;
Servo servo1;
AF_DCMotor M1(1);
AF_DCMotor M2(2);
AF_DCMotor M3(3);
AF_DCMotor M4(4);
```

```
void setup() {
Serial.begin(9600);
pinMode(Trig, OUTPUT);
pinMode(Echo, INPUT);
pinMode(8, OUTPUT);
servo.attach(motor);
servo1.attach(9);
M1.setSpeed(Speed);
  M2.setSpeed(Speed);
  M3.setSpeed(Speed);
  M4.setSpeed(Speed);
}    void loop()
{    //
Obstacle();
voicecontrol();
}
 void Obstacle() {
distance = ultrasonic();
if (distance <= 12) {
Stop();      backward();
delay(100);      Stop();
L = leftsee();
servo.write(spoint);
delay(800);      R =
rightsee();
servo.write(spoint);
if (L < R) {
right();
delay(500);        Stop();
delay(200);      } else if
(L > R) {        left();
delay(500);        Stop();
delay(200);      }    }
else {      forward();
  }
}    void voicecontrol() {
if (Serial.available() > 0) {
value = Serial.read();
Serial.println(value);      if
(value == '^') {
forward();      } else if
(value == '-') {
backward();      } else if
(value == '<') {        L =
leftsee();
servo.write(spoint);        if
(L >= 10 ) {          left();
        delay(500);
```

```
       Stop();
     } else if (L < 10) {
        Stop();
      }
    } else if (value == '>') {
R = rightsee();
servo.write(spoint);        if
(R >= 10 ) {          right();
delay(500);          Stop();
      } else if (R < 10) {
        Stop();
      }
    } else if (value == '*') {
      Stop();    }if (value ==
'1') {        digitalWrite(8,
HIGH);      } else if (value
== '0') {
digitalWrite(8, LOW);    }if
(value == 'l') {
servo1.write(0);
delay(2000);
servo1.write(180);
delay(2000);
servo1.write(180);
    }
  }
}

// Ultrasonic sensor distance reading function
int ultrasonic() {    digitalWrite(Trig, LOW);
delayMicroseconds(4);    digitalWrite(Trig,
HIGH);    delayMicroseconds(10);
digitalWrite(Trig, LOW);    long t =
pulseIn(Echo, HIGH);    long cm = t / 29 / 2;
//time convert distance    return cm;
}    void
forward() {
M1.run(FORWARD);
M2.run(FORWARD);
 M3.run(FORWARD);
M4.run(FORWARD);
}    void
backward() {
M1.run(BACKWARD);
  M2.run(BACKWARD);
  M3.run(BACKWARD);
  M4.run(BACKWARD);
```

```
}    void right() {
M1.run(BACKWARD);
  M2.run(BACKWARD);
  M3.run(FORWARD);
  M4.run(FORWARD);
}  void left() {
M1.run(FORWARD);
M2.run(FORWARD);
M3.run(BACKWARD);
M4.run(BACKWARD);
}  void Stop() {
M1.run(RELEASE);
M2.run(RELEASE);
M3.run(RELEASE);
M4.run(RELEASE);  }
int rightsee() {
servo.write(20);
delay(800);  Left =
ultrasonic();  return
Left;  }  int
leftsee() {
servo.write(180);
delay(800);  Right =
ultrasonic();  return
Right; }
```

METHODOLOGY

1. **Requirement Analysis**:
    - Define the objectives and functionalities of the voice-controlled humanoid robot.
    - Identify the target audience and potential use cases for the robot.
    - Gather user requirements through surveys, interviews, and feedback sessions.
2. **Hardware Selection**:
    - Choose appropriate hardware components such as microcontrollers, actuators, sensors, and communication modules.
    - Consider factors like power consumption, processing capability, and compatibility with the chosen software platform.
3. **Software Platform Selection**:
    - Evaluate different software platforms for speech recognition and processing, such as Google Speech API or custom-built solutions.

- Choose a platform based on accuracy, real-time performance, and compatibility with the hardware.

4. **Speech Recognition Implementation**:
   - Develop or integrate speech recognition algorithms to convert audio input into text commands.
   - Implement noise reduction and signal processing techniques to enhance the accuracy of speech recognition, especially in noisy environments.
5. **Natural Language Understanding (NLU)**:
   - Develop NLU algorithms to parse and interpret user commands. o Use techniques such as natural language processing (NLP) and machine learning to understand the context and intent behind user utterances.
6. **Integration with Robot Control System**:
   - Develop interfaces to integrate the speech recognition and NLU modules with the robot's control system. o Implement protocols for communication between the software modules and hardware components of the robot.
7. **Voice Command Mapping**: o Define a set of voice commands and their corresponding robot actions or behaviours. o Create a mapping between recognized voice commands and specific functions or routines to be executed by the robot.
8. **Robot Behaviour Design**:
   - Design behaviours and motion sequences for the robot to execute in response to user commands.
   - Implement algorithms for motion planning, trajectory generation, and obstacle avoidance to ensure smooth and safe movements.
9. **User Interface Design**:
   - Develop a user-friendly interface for initiating voice commands and interacting with the robot.
   - Incorporate visual or auditory feedback to inform users of the robot's status and responses.
10. **Testing and Evaluation**:
    - Conduct extensive testing to evaluate the performance of the voice-controlled humanoid robot.
    - Test the accuracy and robustness of speech recognition in various environments and with different speakers.

- Evaluate the responsiveness and reliability of the robot's actions in executing user commands.
- Gather feedback from users to identify areas for improvement and refinement.

11. **Iterative Development**:
    - Iterate on the design and implementation based on testing results and user feedback. o Continuously refine algorithms, optimize performance, and add new features to enhance the capabilities of the robot.
12. **Documentation and Deployment**:
    - Document the design, implementation, and testing processes for future reference and replication.
    - Prepare user manuals and guides for operating the voice-controlled humanoid robot.
    - Deploy the robot in real-world settings or demonstrations, showcasing its capabilities and potential applications.

APPLICATIONS

Voice-controlled humanoid robots have a wide range of applications across various industries and domains. Here are some notable applications:

1. **Assistive Technology**: Voice-controlled humanoid robots can assist individuals with disabilities or mobility impairments by performing tasks such as fetching objects, opening doors, or providing companionship. They can enhance the independence and quality of life for people with limited mobility.
2. **Healthcare**: In healthcare settings, voice-controlled humanoid robots can serve as companions for patients, reminding them to take medication, assisting with rehabilitation exercises, or providing entertainment and emotional support. They can also help healthcare professionals by carrying out simple tasks such as delivering supplies or assisting in patient care.
3. **Education**: Voice-controlled humanoid robots can be used in educational settings to engage students and facilitate interactive learning experiences. They can serve as tutors, providing personalized instruction and feedback, or as educational companions, guiding students through activities and experiments.
4. **Customer Service**: Voice-controlled humanoid robots can be deployed in customer service roles, such as receptionists or information assistants, in various settings including hotels, airports, and

shopping malls. They can greet customers, provide information about products or services, and assist with inquiries or reservations.

5. **Home Automation**: Voice-controlled humanoid robots can function as central hubs for home automation systems, allowing users to control smart devices such as lights, thermostats, and appliances using voice commands. They can also perform tasks such as monitoring home security or managing schedules and reminders.
6. **Entertainment and Hospitality**: Voice-controlled humanoid robots can entertain guests in entertainment venues, theme parks, or resorts by interacting with them, performing shows or demonstrations, and providing information about attractions or events. They can enhance the overall guest experience and create memorable interactions.
7. **Research and Development**: Voice-controlled humanoid robots are valuable research platforms for studying human-robot interaction, artificial intelligence, and cognitive science. Researchers can use them to investigate topics such as natural language processing, emotion recognition, and social behaviour in robots.
8. **Manufacturing and Industry**: Voice-controlled humanoid robots can be employed in manufacturing and industrial environments for tasks such as assembly, inspection, and inventory management. They can work alongside human workers, assisting them with repetitive or physically demanding tasks, and improving productivity and efficiency.
9. **Public Service and Assistance**: Voice-controlled humanoid robots can be deployed in public spaces such as airports, train stations, and museums to provide information, guidance, and assistance to visitors. They can help people navigate complex environments, answer questions, and provide emergency support if needed.
10. **Personal Assistance**: Voice-controlled humanoid robots can serve as personal assistants in homes or workplaces, managing tasks such as scheduling appointments, organizing emails, or conducting internet searches. They can provide personalized assistance tailored to individual preferences and needs.

These applications demonstrate the versatility and potential impact of voice-controlled humanoid robots in various fields, contributing to improved efficiency, accessibility, and convenience in everyday life.

REFERENCES

- Bluetooth Module Working with Arduino, www.circuitdigest.com
- Arduino.cc Projects
- Tinkercad
- Github Repositories
- Arduino Forum

Roll No. – UE215039 & 70

****