INTRODCUTION TO COMPUTER NETWORKING

II-B.Tech-CSE-AI (Semester 4)

End Semester Project

VOICE ENABLED HOME AUTOMATION SYSTEM

Submitted By:

Reddy Hema Radhika (CB.EN.U4AIE21050)

Sai Sri Darsini H (CB.EN.U4AIE21055)

Suda Hari Priya (CB.EN.U4AIE21067)

Sykam Sumanjali (CB.EN.U4AIE21068)

Under the supervision of

B.Ganga Gowri Mam

DEPT. OF COMPUTATIONAL ENGINEERING AND NETWORKING

AMRITA
VISHWA VIDYAPEETHAM

ACKNOWLEDGEMENT

We would like to express our special gratitude to our professor(Ganga Gowri Ma’am) who gave us this wonderful opportunity to do the hardware implementation with AURDINO. It helped us in doing a lot of research and we came to know about so many new things. We are thankful for the opportunity given. The implementation nurtured interest on the subject even more and pushed us forward to explore deep into the course.

TABLE OF CONTENTS

ABSTRACT

Mechanization involves mold in the 21st century and assumes an imperative part in our everyday lives. The fundamental fascination required in robotized frameworks are human work, exertion, time and decrease mistakes because of human carelessness.With the advancement of present day innovation,PDAs have turned into a need for each individual on this planet.In our venture clients in their home will control diverse home machines utilizing PDA. The inherent program accessible in the android market is utilized to control the gadgets inside the home.

This project presents the implementation of a voice-enabled home automation system utilizing Arduino, a Bluetooth module, and the ESP32-CAM module for security purposes. The system allows users to control home appliances through voice commands, providing convenience and ease of use. ESP32-CAM module enhances security by capturing images or videos upon detecting motion. This voice-enabled home automation system offers a seamless user experience, efficient control of devices, and advanced security features .

0.1 Hardware Requirements

1.ARDUINO UNO

2.BLUETOOTH HC-05 MODULE

3.RELAY MODULE

4.POWER SUPPLY

5.LED LIGHTS

6.DC MOTOR

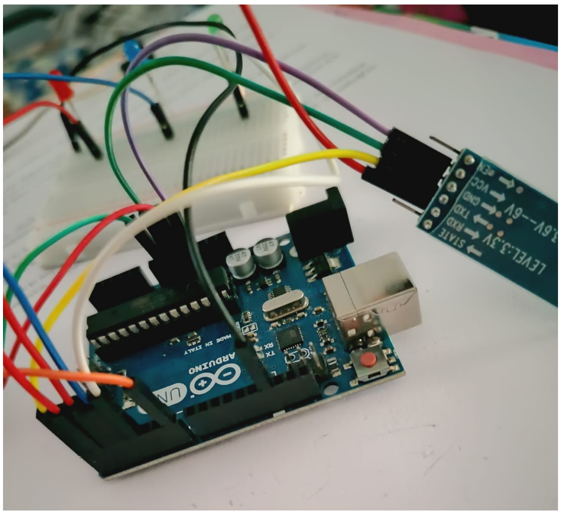
7.JUMPER WIRES

8.BREAD BOARD

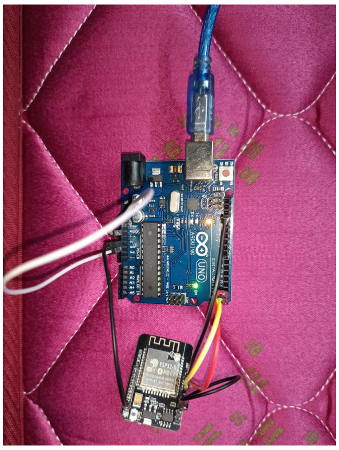
9.ESP 32 CAM MODULE

0.2 Hardware Setup

1.SET UP OF HOME AUTOMATION USING HOME APPLIANCES



2.ESP 32 CAM FOR SECURITY PURPOSE



0.3 AURDINO UNO

Arduino Uno is a popular microcontroller board based on the ATmega328P microcontroller. It is one of the main hardware components of the Arduino platform, designed for easy prototyping and development of electronic projects.

1.Microcontroller: Arduino Uno is built around the ATmega328P microcontroller, which is a powerful 8-bit microcontroller with 32KB of flash memory, 2KB of SRAM, and 1KB of EEPROM.

2.Digital and Analog I/O: The board provides a set of digital input/output pins (14) and analog input pins (6), allowing you to connect and control various sensors, actuators, and other electronic components.

3.Programming: Arduino Uno can be programmed using the Arduino programming language, which is a simplified version of C/C++. The Arduino IDE (Integrated Development Environment) is used to write, compile, and upload code to the board.

4.Power Supply: The board can be powered using a USB cable connected to a computer or an external power supply.

It also has a built-in voltage regulator that allows it to be powered with a range of voltages (7-12V).

5.Communication: Arduino Uno supports serial communication via USB, allowing it to connect to a computer for programming and debugging. It also has a hardware UART (Universal Asynchronous Receiver-Transmitter) for serial communication with other devices.

0.4 BLUETOOTH HC-05 MODULE

- The Bluetooth HC-05 module plays a crucial role in a voice-enabled home automation system. It serves as the wireless communication interface between the voice control unit and the home automation devices.

- By establishing a Bluetooth connection, the HC-05 module enables seamless communication, allowing voice commands to be transmitted wirelessly to control various home automation functionalities.

- It acts as a bridge, enabling voice-controlled operation of lights, appliances, security systems, and other devices within the home. The HC-05 module's versatility, ease of use, and compatibility make it an essential component in integrating voice control capabilities into a home automation system.

0.5 RELAY MODULE

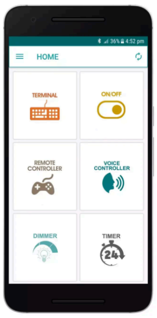
- The relay module serves as an interface between the microcontroller and the electrical devices being controlled.
- By using relays, the module can switch high-power appliances such as lights, fans, or motors based on voice commands. It provides electrical isolation and protection, ensuring the safe operation of the connected devices.
- The relay module allows the voice control unit to activate or deactivate various home automation functions, enabling seamless control over multiple devices. Its flexibility, reliability, and compatibility make it an essential part .

0.6 ESP-32 CAM MODULE

- The ESP-32 CAM module plays a significant role in a voice-enabled home automation system. With its integrated camera and Wi-Fi capabilities, it enables real-time video streaming and image capture based on voice commands.
- The module can be used for security purposes, allowing users to remotely monitor their homes through live video feeds. Additionally, the ESP-32 CAM module can be integrated with voice recognition algorithms to provide visual feedback or perform actions based on detected objects or gestures.
- Its compact size, versatility, and connectivity options make it an essential component for incorporating visual capabilities into a voice-enabled home automation system.

0.7 Working of Voice Assistant

1. Using the play store app(Arduino Bluetooth) we control the appliances using the voice control.
2. After installation pair the BT Module with our device.
3. According to the instructions given in the code we will operate the appliances using voice command.



0.8 Hardware Connection

1.Hardware Connections for Home Appliances.

- Connect the bluetooth module and Arduino using appropriate pins.
- Connect the BT module Pins to the corresponding Arduino pins.

VCC-5V

GND-GND

TXD-RX

RXD-TX

- Using Breadboard connect the LED and jumper wires according to the Arduino.
- Connect the relay module to the Arduino board. The pins of Relay are connected to Arduino as following.

IN(S) PIN-7

VCC (+) PIN-GND

GND(-) PIN -GND

- Connect the Relay module to the DC Motor .

2.Hardware Connection for ESP 32 Cam Module.

- Connect the ESP 32 PINS to the corresponding Arduino pins.

GND -GND

TX0-GPIO 1

RX0-GPIO 3

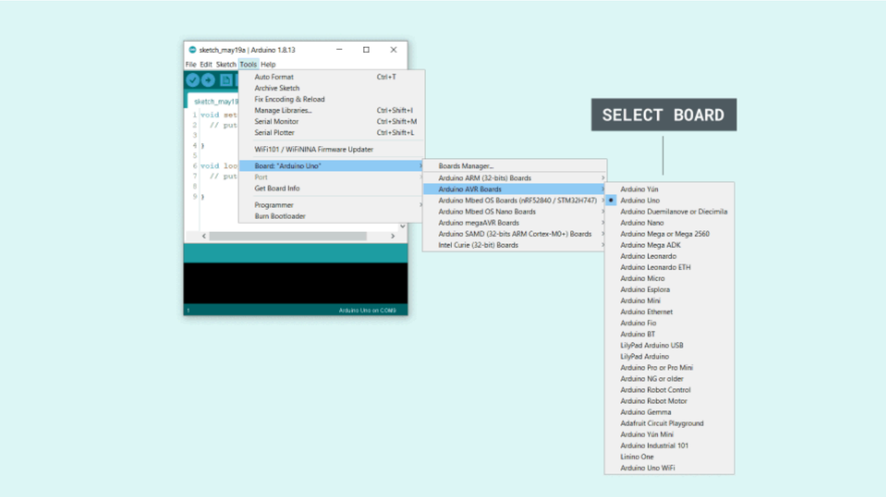
5V -5V

GPIO 0 -GND

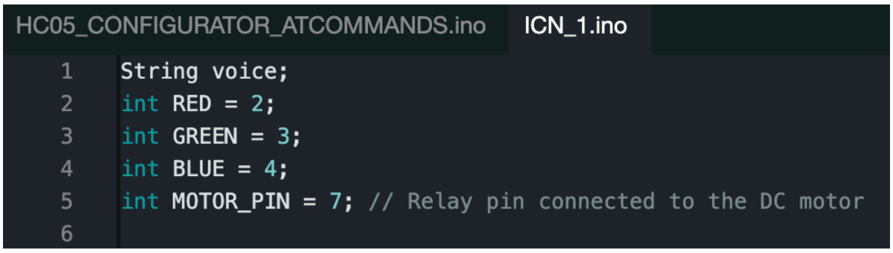
0.9 IMPLEMENTATION

- Before the Arduino connection to the device, remove the RX and TX pins of the Arduino.

- Open Arduino IDE: Go to Tools and Select the board(Arduino Uno) Next select the port. Now Upload the code and compile it.

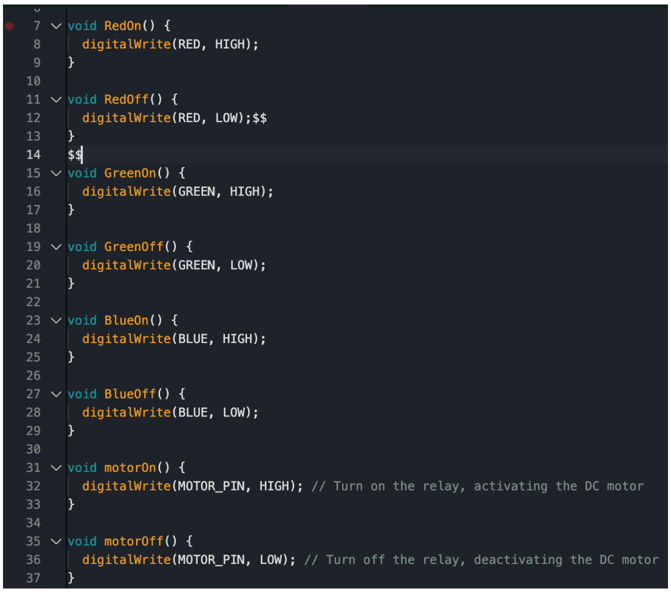


- This sets up variables for storing voice input and defines pin assignments for an RGB LED (using RED, GREEN, and BLUE variables) and a DC motor (using MOTOR PIN variable).



```
String voice;
int RED = 2;
int GREEN = 3;
int BLUE = 4;
int MOTOR_PIN = 7; // Relay pin connected to the DC motor

```

- The Off() function turns color of the RGB LED by setting the pin assigned to a LOW state using the digitalWrite() function. This will turn off the red color.The On() function turns on color of the RGB LED by setting the pin assigned to the to a HIGH state using the digitalWrite() function. This will illuminate the color.The motorOn() function turns on a DC motor by activating a relay connected to the Arduino's MOTOR PIN. It sets the MOTOR PIN to a HIGH state using the digitalWrite() function, which turns on the relay and activates the motor.

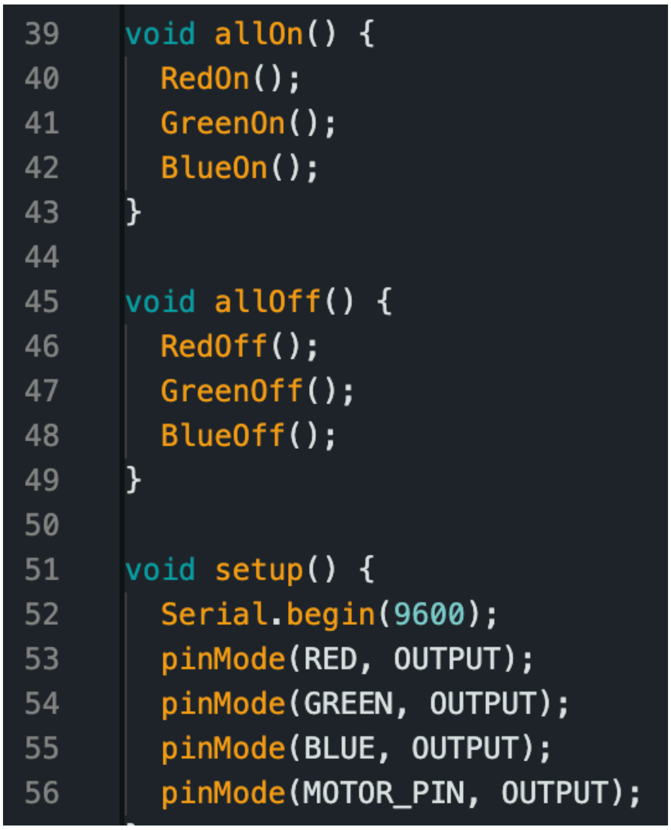


```
void RedOn() {
  digitalWrite(RED, HIGH);
}

void RedOff() {
  digitalWrite(RED, LOW);$$
}
$$
void GreenOn() {
  digitalWrite(GREEN, HIGH);
}

void GreenOff() {
  digitalWrite(GREEN, LOW);
}

void BlueOn() {
  digitalWrite(BLUE, HIGH);
}

void BlueOff() {
  digitalWrite(BLUE, LOW);
}

void motorOn() {
  digitalWrite(MOTOR_PIN, HIGH); // Turn on the relay, activating the DC motor
}

void motorOff() {
  digitalWrite(MOTOR_PIN, LOW); // Turn off the relay, deactivating the DC motor
}
```

- The allOn() function turns on all the colors of the RGB LED simultaneously.The allOff() function turns off all the colors of the RGB LED simultaneously.

- The setup() function is a required function in an Arduino sketch. It is called once at the beginning of the program and is used to initialize settings and configurations.

- Serial.begin(9600): Initializes the serial communication with a baud rate of 9600. This allows you to communicate with the Arduino board via a serial connection.

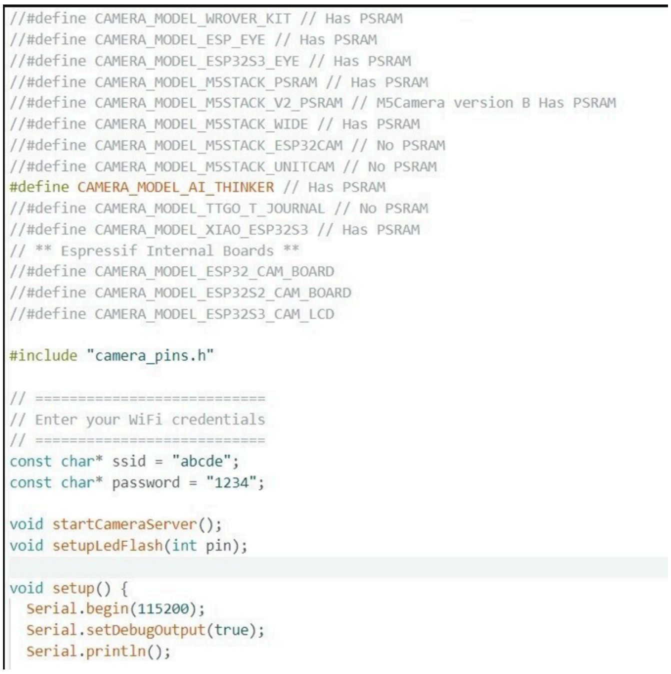


```
void allOn() {
  RedOn();
  GreenOn();
  BlueOn();
}

void allOff() {
  RedOff();
  GreenOff();
  BlueOff();
}

void setup() {
  Serial.begin(9600);
  pinMode(RED, OUTPUT);
  pinMode(GREEN, OUTPUT);
  pinMode(BLUE, OUTPUT);
  pinMode(MOTOR_PIN, OUTPUT);
```

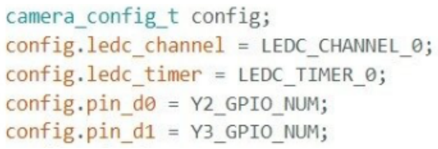
- The loop() function continuously checks for available characters in the serial input buffer. If there are characters available, it reads them one by one, storing them in the voice variable until the termination character ” is encountered. Once the termination character is received, the loop is exited, and the complete input sequence is stored in the voice variable.

- After executing the actions based on the voice input, the line voice = assigns a new value to the voice variable. This line effectively clears the voice variable, preparing it to store the next voice input.

- Install the required libraries for ESP camera control in the Arduino IDE .

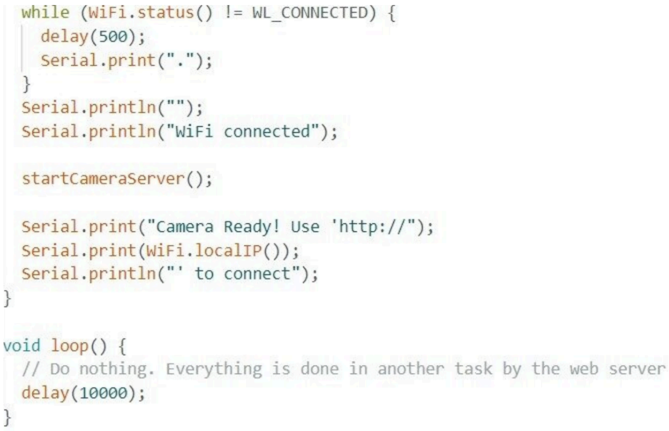


```
//#define CAMERA_MODEL_WROVER_KIT // Has PSRAM
//#define CAMERA_MODEL_ESP_EYE // Has PSRAM
//#define CAMERA_MODEL_ESP32S3_EYE // Has PSRAM
//#define CAMERA_MODEL_M5STACK_PSRAM // Has PSRAM
//#define CAMERA_MODEL_M5STACK_V2_PSRAM // M5Camera version B Has PSRAM
//#define CAMERA_MODEL_M5STACK_WIDE // Has PSRAM
//#define CAMERA_MODEL_M5STACK_ESP32CAM // No PSRAM
//#define CAMERA_MODEL_M5STACK_UNITCAM // No PSRAM
#define CAMERA_MODEL_AI_THINKER // Has PSRAM
//#define CAMERA_MODEL_TTGO_T_JOURNAL // No PSRAM
//#define CAMERA_MODEL_XIAO_ESP32S3 // Has PSRAM
// ** Espressif Internal Boards **
//#define CAMERA_MODEL_ESP32_CAM_BOARD
//#define CAMERA_MODEL_ESP32S2_CAM_BOARD
//#define CAMERA_MODEL_ESP32S3_CAM_LCD

#include "camera_pins.h"

// ===========================
// Enter your WiFi credentials
// ===========================
const char* ssid = "abcde";
const char* password = "1234";

void startCameraServer();
void setupLedFlash(int pin);

void setup() {
  Serial.begin(115200);
  Serial.setDebugOutput(true);
  Serial.println();
```



```
camera_config_t config;
config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_0;
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
```

- This loop waits for the WiFi connection to be established by continuously checking the WiFi status. Once the connection is established, it prints a message indicating successful WiFi connection. After that, it starts the camera server and prints the local IP address to connect to the camera server. The loop() function then enters a delay of 10,000 milliseconds (10 seconds) before repeating the process. This delay ensures that the code inside the loop is executed at a specific interval.



```
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");

  startCameraServer();

  Serial.print("Camera Ready! Use 'http://");
  Serial.print(WiFi.localIP());
  Serial.println("' to connect");
}

void loop() {
  // Do nothing. Everything is done in another task by the web server
  delay(10000);
}
```

0.10 Involvement of Networking

- Communication and Connectivity: communication and connectivity are essential for receiving voice commands from the user and transmitting control signals to the connected devices. Wireless technologies like Wi-Fi or Bluetooth enable seamless communication, allowing users to control their home devices remotely and receive real-time feedback on their status.

- Internet of Things (IoT) Integration: Incorporating Internet of Things (IoT) integration in a voice-enabled home automation system enables the devices to connect to the internet and communicate with each other. This integration enables remote control, real-time monitoring, and access to the system from anywhere, enhancing convenience and providing advanced automation capabilities to users.

- Cloud Services and Voice Processing: These play integral roles in a voice-enabled home automation system. Cloud services provide the infrastructure for storing and processing voice data, allowing for advanced voice recognition and natural language processing capabilities. Voice processing algorithms analyze and interpret the voice commands received from users, transforming them into meaningful instructions for controlling connected devices. The cloud also enables seamless integration with other smart home platforms and remote access to the system. This combination of cloud services and voice processing enhances the overall functionality, flexibility, and convenience of the home automation system.

- Network Security: It is essential in a voice-enabled home automation system to safeguard against unauthorized access and protect user privacy. Implementing robust security measures, such as encryption protocols and secure authentication mechanisms, ensures that communication between devices and the central control unit is secure. Regular security updates and monitoring help mitigate potential vulnerabilities, maintaining the integrity and confidentiality of the system.

- Remote Access and Control: Remote access and control are key features of a voice-enabled home automation system, allowing users to manage their connected devices from anywhere. Through a secure internet connection, users can remotely issue voice commands, monitor device status, and adjust settings using their smartphones or other internet-connected devices. This flexibility enables convenience, energy efficiency, and peace of mind, as users can control and monitor their home even when they are away.

0.11 ADVANTAGES

- 1.ENERGY EFFICIENCY: It is a significant advantage of a voice-enabled home automation system. By intelligently controlling and automating devices, such as lights, thermostats, and appliances, based on voice commands, the system optimizes energy usage. This leads to reduced energy consumption, lower utility bills, and a more sustainable and environmentally friendly home environment.
- 2.ENVIRONMENTAL IMPACT : This can have a positive environmental impact. By optimizing energy usage and reducing wasteful practices through automation, it promotes energy efficiency and conservation. This can result in lower carbon emissions, reduced resource consumption, and a more eco-friendly lifestyle, contributing to a greener and more sustainable environment.