

**VOICE-CONTROLLED OFFLINE HOME AUTOMATION**

Department of Electrical and Electronics

Engineering

*Submitted by*

|  |  |
| --- | --- |
| **NAME** | **REGISTER NUMBER** |
| BALAJI V | 715522105009 |



Guided by

Mr. B. Nallathambi B.E

TRAINER

**TABLE OF CONTENTS:**

|  |
| --- |
| 1. ABSTRACT |
| 2. INTRODUCTION |
| 3. PROBLEM STATEMENT |
| 4. PROPOSED SYSTEM |
| 5. COMPONENTS REQUIRED |
| 6. SOFTWARE |
| 7. BLOCK DIAGRAM |
| 8. OUTPUT |
| 9. WORKING |
| 10. CONCLUSION |

# 1. ABSTRACT

This project presents a smart home automation system that can be controlled via voice commands without the need for an internet connection. It leverages the capabilities of the Raspberry Pi along with the VOSK speech recognition toolkit to interpret voice inputs and control household appliances using relays. The system is designed for convenience, especially for the elderly and physically challenged, while maintaining privacy and operational independence from cloud-based services. Its offline functionality ensures uninterrupted control even in the absence of internet access.

# 2. INTRODUCTION

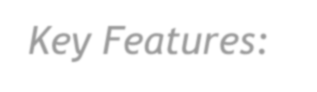
Home automation is becoming increasingly common, offering control over various appliances and systems through the internet. However, most voice activated systems today rely heavily on cloud computing, which raises concerns regarding privacy, reliability, and data usage. This project introduces an offline voice-controlled home automation system to address these issues. The system uses a Raspberry Pi, open-source speech recognition (VOSK), and Python programming to operate multiple household devices without needing internet connectivity. This allows users to control lights, fans, and other appliances using simple voice commands locally.

1. **PROBLEM STATEMENT:**

In many rural or remote areas, internet access is either limited or unstable, making conventional online voice assistants unreliable. Additionally, cloud-based voice assistants pose privacy risks as user data is sent to external servers. The physically disabled and elderly population often struggle with physical switches, further emphasizing the need for hands-free control. This project aims to solve these problems by designing an offline, voice-controlled system that can operate efficiently, securely, and locally.

1. **PROPOSED SYSTEM:**

The proposed system consists of a Raspberry Pi that continuously listens for specific voice commands through a microphone. Upon recognizing a valid command using the VOSK speech-to-text engine, the Raspberry Pi processes it and activates the corresponding GPIO pins. These pins are connected to relay modules that switch appliances ON or OFF. Python is used to handle speech recognition, logic control, and GPIO operations. The system can recognize commands like “light on”, “fan off”, “turn on TV”, etc., and map them to the respective relays.

*Key Features:*

* + Offline speech recognition using VOSK
  + Control of multiple home appliances
  + Low-cost and power-efficient setup
  + Privacy-respecting design
  + Expandable and customizable system

# 5. COMPONENTS REQUIRED 1. Raspberry Pi 3/4

* **Role**: Acts as the main processing unit.
* **Working**: Runs Python code and controls GPIO pins based on voice input. It processes audio, interprets commands using VOSK, and sends output signals to the relays.

## 2. Relay Module (4 or 8 Channel)

* **Role**: Acts as an electronic switch to control high-voltage appliances.
* **Working**: Receives digital signals (HIGH or LOW) from Raspberry Pi GPIO pins. When GPIO sends a HIGH signal, the relay switches ON the connected appliance (like light or fan), and OFF when LOW.

## 3. USB Microphone

* **Role**: Captures voice input from the user.
* **Working**: Converts spoken words into audio signals. The microphone continuously sends audio to the Raspberry Pi via USB or audio jack, which is then processed by PyAudio and VOSK.

## 4. Power Supply

* **Role**: Provides necessary power to the Raspberry Pi and relay module.
* **Working**: A 5V/3A adapter typically powers the Raspberry Pi. The relay module may also need an external 5V supply depending on the number of channels and load.

**6. S0FTWARE COMPONENT**

## 1. VOSK

* **Role**: Offline speech recognition engine.
* **Working**: Converts audio input into text using a pre-trained speech model. Works completely offline and supports multiple languages. For example, when a user says “fan on”, VOSK recognizes it and returns the text.

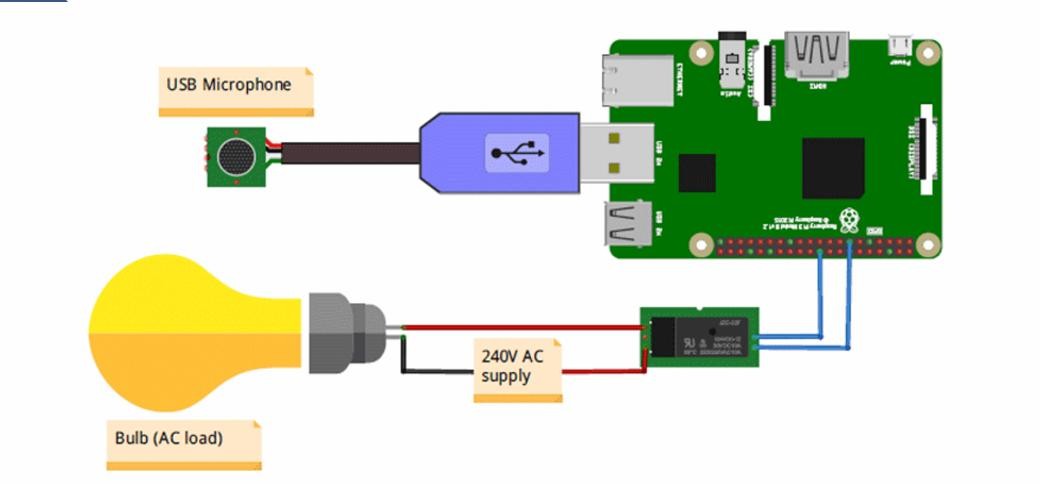
## 2. PyAudio

* **Role**: Audio input interface.
* **Working**: Captures real-time audio from the microphone. Feeds this audio to VOSK for speech-to-text conversion. **3. Raspbian OS (Now Raspberry Pi OS)**
* **Role**: Operating system for Raspberry Pi.
* **Working**: Manages all software and hardware processes. It provides the environment to install and run Python, VOSK, and other required libraries.

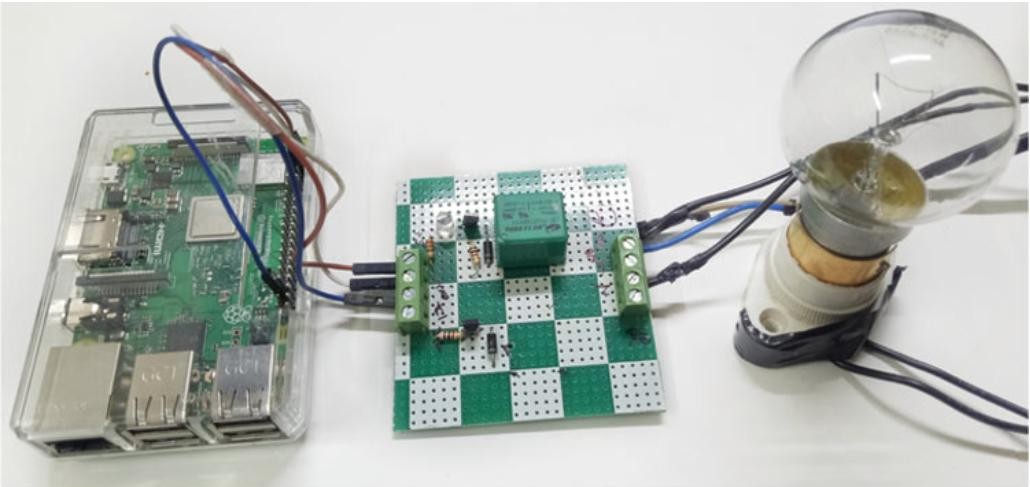
## 4. Thonny IDE / Terminal

* **Role**: Code editor or command line interface.
* **Working**: Used to write, edit, and execute Python scripts that control the automation process.

## 7. BLOCK DIAGRAM



**8. OUTPUT**



# 9. WORKING

The working of the Voice-Controlled Offline Home Automation system is centered around the integration of a Raspberry Pi, voice recognition, and relay-based electrical control. When the system is powered on, the Python program running on the Raspberry Pi initiates and begins listening for voice commands through a connected USB microphone. The voice input is processed using the VOSK speech recognition library, which works entirely offline, making the system independent of internet connectivity.

When a user says a command like “fan on” or “light off,” the VOSK engine converts the spoken words into text and checks if it matches any of the predefined control commands. If a match is found, the Python code identifies the specific GPIO pin associated with that appliance. The Raspberry Pi then sends an electrical signal (HIGH or LOW) to the corresponding GPIO pin, which is wired to a relay module. The relay acts as a switch, allowing the Raspberry Pi’s low-voltage signal to control the higher voltage required to operate household devices like lights, fans, or other appliances. For example, when the command “light on” is given, the GPIO pin linked to the light relay goes HIGH, closing the relay circuit and turning the light on. The same process applies in reverse for turning appliances off.

This setup allows seamless and real-time voice control of home appliances, without relying on cloud services. The system is efficient, fast, and suitable for low-cost smart home automation, particularly in areas with unreliable or no internet access. Each voice command directly maps to a physical action, making the user experience simple, intuitive, and responsive.

# 10. CONCLUSION

This project successfully demonstrates an efficient, cost-effective, and secure home automation system that functions entirely offline. By eliminating the dependency on internet services, it provides a more reliable and privacy-conscious solution for home automation. It also offers a valuable tool for improving accessibility for differently-abled individuals. The system is scalable and can be further enhanced by adding features like gesture control, remote monitoring (with local network access), and integration with sensors.