



**DESIGN AND CONSTRUCTION OF A
VOICE-ACTIVATED SWITCHING SYSTEM (VASS)**

ONYELA ONYEKACHUKWU

ENG1905372

NOSAKHARE JOSHUA

ENG1910173

FOLORUNSHO CHIOMA STEPHANIE

ENG1905327

COOKEYGAM CALEB JOSHUA

ENG1910171

DEPARTMENT OF ELECTRICAL/ELECTRONICS ENGINEERING

FACULTY OF ENGINEERING

UNIVERSITY OF BENIN

BENIN-CITY

FEBRUARY 2025

**A PROJECT WORK ON
DESIGN AND CONSTRUCTION OF A
VOICE-ACTIVATED SWITCHING SYSTEM (VASS)**

ONYELA ONYEKACHUKWU	ENG1905372
NOSAKHARE JOSHUA	ENG1910173
FOLORUNSHO CHIOMA STEPHANIE	ENG1905327
COOKEYGAM CALEB JOSHUA	ENG1910171

**SUBMITTED TO
THE DEPARTMENT OF ELECTRICAL/ELECTRONICS ENGINEERING,
FACULTY OF ENGINEERING,
UNIVERSITY OF BENIN, BENIN CITY, EDO STATE.**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF
THE BACHELOR OF ENGINEERING (B.ENG.) DEGREE IN
ELECTRICAL/ELECTRONICS ENGINEERING**

FEBRUARY, 2025

CERTIFICATION

This is to certify that this project “**Design and Construction of a Voice-Activated Switching System (VASS)**” was carried out by **Onyela Onyekachukwu** (ENG1905372), **Nosakhare Joshua** (ENG1910173), **Folorunsho Chioma Stephanie** (ENG1905327), **Cookeygam Caleb Joshua** (ENG1910171) and submitted to the Department of Electrical/Electronics Engineering, Faculty of Engineering, University of Benin, Benin City. Fulfilled the minimum requirements governing the award of a Bachelor of Engineering (B.Eng.) degree in Electrical/Electronics Engineering.

Engr. A. A. Muhammed

(Project Supervisor)

Signature

Date

Engr. Prof. K. O. Ogbeide

(Head of Department)

Signature

Date

DEDICATION

This project work is dedicated to God Almighty, our Father and Lord who is, has been and continues to be our Source, Sustainer and Sufficiency.

ACKNOWLEDGEMENTS

Foremost, we express profound gratitude to the Almighty God for His innumerable blessings, grace, and favour bestowed upon us throughout this project, enabling its successful completion.

We extend our sincere appreciation to our esteemed project supervisor, Engr. A. A. Muhammed, for his invaluable guidance, patience, and understanding during the entire duration of this project. Our heartfelt thanks also go to the Head of Department, Engr. Prof. K.O. Ogbeide and our Dean of Faculty, Engr. Prof. E. A. Ogujor for providing a conducive environment that facilitated the successful execution of this work.

Furthermore, we acknowledge and thank our parents and family members for their unwavering support and encouragement throughout this journey. Lastly, we express gratitude to our friends for their love, motivation, and backing, which played a crucial role in our accomplishment.

ABSTRACT

This project aims to develop an innovative voice-activated switching system that enhances home automation by enabling hands-free control of electrical appliances. Utilizing speech recognition technology and the Arduino Uno microcontroller, the system provides users with a seamless and accessible method of operating household devices through voice commands.

The project is built around an Arduino Uno interfaced with a Speech Recognition Module Easy VR3 plus, a relay module, and a 5V converter module, all housed within a protective enclosure. The Speech Recognition Module captures and processes voice commands, which are then interpreted by the Arduino Uno. Based on the received command, the Arduino activates the relay module, which switches the connected electrical appliances on or off. The system is powered by a 12V dc battery, regulated to 5V using an L298D motor driver to ensure stable operation.

The developed voice-activated switching system successfully demonstrated the capability to recognize and execute voice commands efficiently. The Speech Recognition Module accurately processed user input, and the Arduino Uno effectively translated the recognized commands into control signals for the relay module. The system exhibited high response accuracy in quiet environments and maintained reliable performance under various conditions. Ultimately, this project achieved its goal of creating an affordable, user-friendly, and accessible voice-controlled home automation solution.

TABLE OF CONTENTS

CERTIFICATION.....	iii
DEDICATION	iv
ACKNOWLEDGEMENTS	v
ABSTRACT	vi
TABLE OF CONTENTS.....	vii
LIST OF FIGURES.....	ix
LIST OF TABLES	xi
CHAPTER ONE	1
INTRODUCTION.....	1
1.1 Background of the Study	1
1.2 Problem Statement.....	1
1.3 Aim	2
1.4 Objectives	2
1.5 Methodologies	3
1.6 Scope of Work.....	3
1.7 Significance of Work.....	4
CHAPTER TWO	5
LITERATURE REVIEW.....	5
2.1 Introduction.....	5
2.2 Review of Previous Work.....	5
2.3 Challenges and Considerations	16
2.4 History of the Voice-Activated Switching System.....	20
2.5 Fundamentals of Basic Materials Used in the Development of a Voice-Activated Switching System	25
2.6 Principle of Operation.....	30
CHAPTER THREE.....	32
SYSTEM DESIGN AND ANALYSIS	32
3.1 Introduction	32
3.2 Project Design and Overview	32
3.3 Flowchart of the Voice-Activated Switching System	33
3.4 Description of Main Components	34
3.5 Hardware Integration and Operation	38
3.6 Complete descriptions of the functionality of the system	42

3.7 Comprehensive Overview	44
CHAPTER FOUR.....	46
CONSTRUCTION, TESTING AND RESULTS	46
4.1 Introduction	46
4.2 Implementation Process.....	46
4.3 Testing of Various Units	48
4.4 Results	52
4.5 Equipment Used	55
4.6 Comprehensive Overview	58
CHAPTER FIVE.....	59
CONCLUSION AND RECOMMENDATIONS.....	59
5.1 Conclusion	59
5.2 Recommendations	59
REFERENCES.....	61

LIST OF FIGURES

Fig 2.1: A Speech Recognition Module	26
Fig 2.2: An Arduino Uno module	27
Fig 2.3: A Relay Module	28
Fig 2.4: Wall Socket	28
Fig 2.5: Voltage Converter Module	29
Fig 2.6: Male to Female Connectors	29
Fig 2.7: A Plastic Enclosure	30
Fig 3.1: System Application Flowchart	34
Fig 3.2: The EasyVR 3 plus speech Recognition Module Schematics	34
Fig 3.3: Arduino Uno Schematics	35
Fig 3.4: 2-Channel Relay Module Schematics	36
Fig 3.5: 5V Converter Module (L298D) Schematics	36
Fig 3.6: Wall Socket Schematics	37
Fig 3.7: Energy-Saving Bulbs Schematics	37
Fig 3.8: Rechargeable battery	38
Fig 3.9: Arduino Uno with the Speech Recognition Module	38
Fig 3.10: Arduino Uno with the Speech Recognition Module and Relay Module	39
Fig 3.11: Arduino Uno with the Speech Recognition Module, Relay Modules, and AC Mains Connection	40
Fig 3.12: Arduino Uno with the Speech Recognition Module, Relay Modules, AC Mains, and L298D Motor Drive	41
Fig 3.13: Complete Schematic diagram of the system	42
Fig 4.1: The mounting of the Microprocessor on the board	46
Fig 4.2: The mounting of the Voice recognition module on the board	47

Fig 4.3: The mounting of the two channels relay module on the board	47
Fig 4.4: The mounting of the L298D converter module on the board	48
Fig 4.5: Testing the 12V battery	49
Fig 4.6: Testing the L298D motor drive	49
Fig 4.7: Testing the continuity of relay channel 1 in its OFF state	50
Fig 4.8: Voltage Measurement across Socket Load (Relay ON)	51
Fig 4.9: Testing the workability of the Arduino Uno	51
Fig 4.10: The completed system	

LIST OF TABLES

Table 4.1: Results Obtained from Power Supply Test	52
Table 4.2: Results Obtained from Relay Module Test	53
Table 4.3: Performance of the system in a noiseless environment	54
Table 4.4: Performance of the system in a noisy environment	54
Table 4.5: Bill of Engineering Measurement and Evaluation	58

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

With the continuous advancement in home automation technology, smart homes have transitioned from futuristic concepts to everyday realities. A significant area within this realm is the development of innovative control systems that enhances convenience as well as security. One of the technologies gaining prominence in this area is voice-activated switching. A voice-activated switching system is an intuitive and hands-free solution that allows users to control electrical devices through spoken commands, eliminating the need for physical interaction. . These systems are not only user-friendly but also cater to the needs of individuals with mobility issues or disabilities, offering them an accessible interface for operating electronics devices.

In this project, the voice-activated switching system is designed to control household appliances using speech recognition technology. By utilizing components such as the Speech Recognition Voice Module and the Arduino Uno, the system detects specific voice commands and responds by activating or deactivating connected appliances. This approach enhances convenience and accessibility, offering an efficient and seamless method for managing home electronics.

1.2 Problem Statement

Prevalent methods of controlling electrical appliances rely heavily on manual operation, presenting challenges for users with physical limitations or those seeking a more convenient and hands-free experience. While existing voice-controlled systems are available, many are expensive, complex to install, or require specialized technical knowledge. This project seeks

to address the need for an affordable, user-friendly, and easily installable voice-controlled switch system that caters to a wider range of users. The lack of readily available, low-cost solutions for voice-activated appliance control motivates the development of this project.

1.3 Aim

The primary aim of this project is to design and construct a functional voice-activated switching system using speech recognition technology, that allow users to control household appliances via voice commands. This switch is expected to provide a more convenient and accessible alternative to traditional manual switches.

1.4 Objectives

The objectives of this project are as follows:

1. To design a voice-activated switching system using Arduino Uno and a speech recognition module
2. To integrate the system with a relay module that is capable of controlling multiple electrical outlets.
3. To construct the system using readily available components such as a 5V converter module, plastic enclosure, and connectors.
4. To test the system's functionality in controlling electrical appliances without physical interaction.
5. To evaluate the system's efficiency and responsiveness to voice commands in a real-world environment.

1.5 Methodologies

The following methodologies will be used to achieve the project objectives:

1. **Component Selection and Schematic Design:** The project will begin by selecting the required components (Speech Recognition Module, Arduino Uno, Relay Module, etc.) and designing a schematic that shows how they are interconnected.
2. **Hardware Integration:** The selected components will be physically assembled using male-to-female connectors and will then be securely housed in a plastic enclosure.
3. **Software Development:** The Arduino microcontroller will be programmed to recognize specific voice commands with the aid of the speech recognition module and trigger the relay module to control the connected devices.
4. **Testing and validation:** After construction, the system will undergo rigorous testing to validate its performance, this includes its response to various voice commands and its ability to reliably switch on/off the various connected devices.
5. **Iteration:** Based on test results, any issues with command recognition, relay response, or system efficiency will be addressed through iterative improvements.

1.6 Scope of Work

This project focuses on the development of a voice-activated switch system capable of controlling multiple electrical devices in a typical home or office environment. The project encompasses the following key focus areas:

1. The project will focus on designing a system that can control up to four devices using the provided relay module.
2. The system will utilize speech recognition as the principal method of control.
3. The system will be designed to be portable, user-friendly, and capable of integration into existing electrical setups without significant modifications.

4. The system will be restricted to basic voice commands for turning devices on and off.

1.7 Significance of Work

The development of a voice-activated switching system holds significant potential in advancing home automation systems. By providing an accessible, hands-free method of controlling household appliances, this system can greatly benefit users, particularly those with physical disabilities. The integration of speech recognition technology into everyday home systems enhances user convenience and reduces reliance on physical interactions with devices. Moreover, this project showcases the practical application of microcontroller-based systems, such as Arduino, in creating innovative solutions for smart homes. Once fully developed, the voice-activated switching system could contribute to energy efficiency by allowing better management of household devices and could serve as a stepping stone for more advanced automation systems in the future.

This project not only provides a solution for enhanced accessibility and convenience but also highlights the increasing role of automation in improving everyday life. By focusing on voice control, it aligns with current trends in smart home technologies and offers a practical demonstration of how speech recognition can be harnessed to transform conventional home systems.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

The literature review examines existing research and developments in voice-activated switches, home automation, and IoT integration. It highlights key advancements, technological trends, and the impact of voice control systems on the usability and accessibility of home appliances. Additionally, the review identifies gaps in the current literature, providing a foundation for the design and construction of the voice-activated switch.

2.2 Review of Previous Work

Bayat (2007) proposed a voice-activated remote-control system designed specifically for television sets. The primary aim was to allow users to control their TV using voice commands instead of pressing buttons. The system was implemented using an HM2007 voice recognition chip, which processes voice commands, and an AT89C52 microcontroller to control the infrared remote functionality. While the project provided a practical solution for individuals with disabilities or mobility issues, a limitation was noted in the recognition accuracy, particularly when users with different accents or voice pitches operated the system. This made the system more suited for speaker-dependent use, reducing its versatility in multi-user environments.

Lee (2007) developed a multipurpose voice-activated controller aimed at controlling common household appliances such as lights and doorbells using human voice commands. The system was implemented using the HM2007 speech recognition IC, which can store up to 40 words. MATLAB was initially used for speech recognition, but due to its limitations in accuracy, the project shifted to the HM2007 IC for better performance. Despite the system's ability to

achieve up to 95% accuracy, challenges such as speaker dependency and the need for retraining after power loss were noted, particularly when the device was used by different individuals or in varying noise environments.

Schur et al. (2015) designed a voice-activated door opener system aimed at improving accessibility for individuals with disabilities. The primary goal was to create a wireless, voice-controlled device to help users open doors without physical interaction. The system used a wireless microphone to capture voice commands, which were transmitted to a door unit equipped with a microcontroller and an XBee transceiver. While the system provided enhanced accessibility, challenges were noted, including the potential for signal interference and limitations in voice recognition accuracy in noisy environments.

Rudžionis *et al.* (2008) developed a system for controlling computers and electric devices using voice commands, specifically aimed at providing accessibility for individuals with disabilities. The system utilized a combination of speech recognition software and additional USB-based hardware to allow users to control devices such as lamps, radios, and voltmeters through voice commands. The speech recognition engine, which was designed for the Lithuanian language, achieved an accuracy rate of 99.5% for Lithuanian commands and 93.8% for English commands. While the system showed high accuracy, it required training for the Lithuanian recognizer and was primarily speaker-dependent, presenting limitations in terms of user versatility.

Haque *et al.* (2010) designed and manufactured a voice-controlled switching system for electrical devices. The system aimed to allow users to control electronic appliances through voice commands. The speech recognition process was facilitated using SAPI software, which converted voice commands into electrical signals. These signals were transmitted via infrared (IR) to control remote devices. The system successfully executed eight distinct voice commands, such as “fan-on” and “fan-off,” to operate various devices. A key limitation

observed was the low efficiency of the speech recognition software, which recognized voice commands with only 31% accuracy, highlighting the need for further improvements.

Fauzan *et al.* (2011) developed a voice recognition system to enhance home automation, focusing on creating a safer and more convenient way to control electrical devices. The system utilized the HM2007 voice recognition chip and the PIC18F8722 microcontroller to recognize voice commands and switch devices on or off. This small-scale design could control three different switches with eight distinct voice commands. While the system successfully executed tasks such as turning on or off individual switches, challenges remain in refining the system's accuracy and improving the voice capturing mechanism for better performance.

Spachos *et al.* (2022) presented a study on voice-activated Internet of Things (IoT) devices for healthcare applications, aiming to address design challenges and explore emerging applications. Their primary objective was to utilize advancements in low-power audio transducers and voice technology to enhance healthcare services, from telehealth to remote monitoring. The system integrates voice-activity detection (VAD) and natural language processing (NLP) into IoT frameworks, enabling hands-free operation in healthcare environments. While this approach demonstrated potential, significant challenges remain in terms of power consumption and the need for high accuracy in noisy environments.

Netinant *et al.* (2024) developed and assessed an IoT-driven smart home security and automation system with voice commands. The system aimed to enhance home security using passive infrared (PIR) sensors for intrusion detection and automate lighting control via voice commands and buttons. The proposed system utilized Google Assistant for voice recognition and the Blynk mobile application for manual control. The system demonstrated an 83% accuracy for Thai voice commands and 91.50% accuracy for English commands, with a 100% detection rate using PIR sensors. However, the system's responsiveness varied based

on the communication protocol, with 5G showing the best performance compared to 4G and Wi-Fi.

Woodside (2005) developed a voice-activated control system for tactical sonar displays, designed to simplify the tasks of sonar operators by replacing multiple keyboard and mouse inputs with simple voice commands. The system aimed to reduce operator fatigue, enhance efficiency, and eliminate the need for graphical pop-up menus that could obscure critical tactical data. The voice recognition software used Dragon NaturallySpeaking to interpret commands, streamlining the handling of sonar-related detection, classification, and localization tasks. However, challenges in voice command accuracy in noisy underwater environments were noted, requiring further refinement.

Vatansever *et al.* (2018) developed a voice-controlled home automation system aimed at making life easier for the elderly and physically disabled by allowing them to control household appliances via voice commands. The system utilized an Arduino Uno microcontroller paired with an EasyVR module for voice recognition. Commands such as “open the door” or “turn on the television” were transmitted using a ZigBee-based wireless communication module. Tests conducted in both normal and noisy environments showed a 95% success rate in command recognition. However, challenges were noted in the user discomfort of throat microphones, which could cause itching or sweating during prolonged use.

Ali *et al.* (2017) developed a voice recognition-based smart home control system designed for ease of use, particularly for the elderly and disabled. Their primary objective was to create a system capable of controlling home appliances (lights and other electrical devices) using voice commands within existing home environments, without requiring infrastructure modifications. The system utilized an Easy VR 2.0 voice recognition unit, an Arduino UNO microcontroller, and low-power RF modules for wireless communication. While the system

demonstrated functionality and achieved a high success rate in controlled testing (99% accuracy for seven out of ten commands), a limitation was observed in its susceptibility to ambient noise. Testing revealed a significant decrease in recognition accuracy when background noise (e.g., from a television or conversation) was present.

Pavan *et al.* (2017) developed a voice-controlled lighting system aimed at enhancing home convenience. The system employs a server-client architecture, using a central unit with a voice recognition module to manage multiple switch units. Communication occurs via radio frequency, reducing costs by centralizing voice processing. While effective in local environments, challenges include potential signal interference and the need for proximity due to RF limitations, which could restrict functionality in larger spaces.

Bhavani *et al.* (2016) presented a voice-controlled home appliance system (VoCA) designed for cost-effectiveness. Their primary objective was to create a simple and affordable system using an Arduino microcontroller and Bluetooth for controlling home appliances via voice commands from an Android application. The implementation successfully demonstrated basic voice-controlled switching of appliances. However, a significant limitation is the short operational range due to Bluetooth's inherent limitations. In scenarios requiring control from distances exceeding 10 meters, the system's functionality is compromised, restricting its usability and practicality in larger homes or for remote control applications.

Gideon (2019) designed and implemented a low-cost voice-controlled home automation system with integrated energy management and security features. The primary objective was to create a system controllable via voice commands issued through a smartphone, utilizing a microcontroller to execute commands. The system incorporated features such as automated lighting, ventilation, and an intrusion detection system, along with real-time environmental data display (temperature, time, light intensity). Communication between the smartphone application and the microcontroller was achieved via Bluetooth. While the system

successfully demonstrated voice control of home appliances and offered energy-saving and security functionalities, a limitation might be the reliance on Bluetooth for communication, restricting the system's range and potentially impacting reliability in scenarios with significant Bluetooth interference. Another limitation lies in the sensitivity of the android device microphone which propels the need for users to have their phones close to them.

Priya *et al.* (2020) presented a voice-activated home automation system utilizing a NodeMCU microcontroller, Google Assistant, and IFTTT. Their primary objective was to create a low-cost, flexible system enabling voice control of home appliances (lights and fans) through a smartphone. The system employed the NodeMCU's Wi-Fi capabilities (ESP8266) for communication, using Arduino IDE for programming and Adafruit IO as a cloud platform for data management. Gas (MQ2 sensor) and environmental (DHT11 sensor) data were also integrated and displayed. While the system successfully demonstrated voice control via Google Assistant and IFTTT, a potential limitation is the reliance on a cloud service (IFTTT and Adafruit IO) for command interpretation and data transmission, introducing potential points of failure and latency. The system's security also depends on the security of these cloud services.

Chintha *et al.* (2021) developed a voice-activated home appliance control system using Google Assistant, IFTTT, Adafruit IO, and a NodeMCU. Their primary objective was to create a system enabling remote control of appliances via voice commands, overcoming distance limitations of proximity-based systems. Implementation involved using Google Assistant for voice recognition, IFTTT to create conditional triggers based on voice commands, Adafruit IO as a cloud platform for data transmission, and a NodeMCU to control relays connected to the appliances. While their approach demonstrated effective remote control, a limitation lies in the reliance on multiple third-party services (Google Assistant,

IFTTT, Adafruit IO). The system's functionality depends on the continuous availability and stable performance of these services, creating potential points of failure and vulnerability.

Bhuvaneswari *et al.* (2023) presented a voice-controlled home automation system designed to enhance accessibility for individuals with disabilities and the elderly. Their primary objective was to enable wireless control of household appliances (lights, fans, etc.) using voice commands and a dedicated mobile application. The system utilized an Android smartphone, a microcontroller, and the Google Cloud Speech API for voice recognition, offering Bluetooth connectivity, voice control, and manual switching. While the system successfully demonstrated voice-controlled operation and incorporated security features like username/password authentication and a relay circuit for safe high-voltage appliance control, a limitation lies in its reliance on Bluetooth for communication. This restricts the operational range, potentially hindering usability in larger homes or situations requiring control from a distance beyond the Bluetooth's effective range.

Atoyebi (2023) designed and constructed a voice-activated home automation system for six appliances. The primary objective was to replace manual appliance control with a user-friendly, hands-free system. Implementation involved an Arduino Uno microcontroller as the central processing unit, receiving voice commands via a Bluetooth module from an Android application. A relay module facilitated the control of high-power appliances. While the system demonstrated a high success rate in voice command recognition and execution, a limitation was the reliance on a specific Android application and the potential for Bluetooth interference, impacting reliability and range.

Cheng *et al.* (2002) developed a voice-activated remote-control system (VoiceIR) for individuals with physical disabilities. Their primary objective was to enable control of infrared (IR)-based devices using voice commands. Implementation involved a wireless microphone, a central control unit (CCU) based on a Motorola HC11 microcontroller and a

Voice Direct voice recognition chip, and an IR module. The system allowed users to train voice commands linked to specific IR signals, enabling remote control of devices throughout a home. While the system successfully demonstrated voice-activated remote control, limitations arose from the chosen components. The Voice Direct chip lacked sufficient status pins for comprehensive error reporting, requiring workarounds in the HC11 programming. The initial wireless microphone design proved unreliable, necessitating a change to FRS radios. Furthermore, the complexity of IR signals and limitations of the RF communication between the CCU and IR module required significant modifications to the original design, including the addition of filtering and signal regeneration circuitry in the IR module.

Nwankwo *et al.* (2019) designed and implemented a computer-interfaced voice-activated switch for controlling electrical appliances. Their primary objective was to create a system controllable via voice commands or manual input, addressing limitations of manual switching in hazardous environments and for the physically challenged. Implementation involved a Visual Basic 6.0 application linked to an AT89C52 microcontroller via the computer's parallel port. The VB application handled speech recognition and the GUI, while the microcontroller controlled relays to switch the appliances. While the system successfully demonstrated voice and manual control, a key limitation is its reliance on a direct parallel port connection, restricting its range to the length of the cable connecting the computer to the hardware. This tethered design prevents remote operation beyond the physical cable's reach.

Bhatt (2016) designed an Intelligent Voice Activated Home Automation (IVA) system prioritizing accessibility for seniors and individuals with limited mobility. The system's core functionality revolved around processing natural language voice commands to control home appliances. Implementation leveraged a smartphone as the primary control interface, communicating with an Arduino-based control box via Wi-Fi. While the IVA system successfully demonstrated voice-controlled home automation using readily available and

inexpensive components, a limitation arose from the reliance on Wi-Fi for communication. The system's performance could be affected by Wi-Fi network instability and range limitations, potentially hindering seamless control in larger homes or areas with weak Wi-Fi signals.

Several research efforts have explored voice-activated control systems for various applications, highlighting both success and challenges. Prominent approaches involve the use of microcontrollers (such as the HM2007, AT89C52, Arduino Uno, and NodeMCU) coupled with voice recognition modules (like EasyVR and Speech Recognition ICs) to translate spoken commands into actionable instructions. While many projects achieved high accuracy in controlled environments, challenges persist. These include speaker dependency, susceptibility to ambient noise and reliance on specific software or cloud services by IoT based voice-activated control systems. Furthermore the need for retraining after power loss and the potential for signal interference in Bluetooth interfaced voice-activated control systems are recurring challenges.

Building upon these insights, the proposed "voice-activated switching system" aim to address these limitations by utilizing the Arduino Uno in conjunction with the speech recognition module V3. The speech recognition module V3 reduces speaker dependencies by allowing for multiple programmable voice commands, thereby making the system adaptable to different users. While the specific noise immunity and accuracy of the speech recognition module V3 require empirical testing, it's dedicated hardware processing is expected to provide improved performance compared to other voice recognition module used in previous projects. The Arduino Uno's flexibility and ease of programming allow for customization, enabling the integration of various components that enhance the overall functionality of the voice-activated switching system. Additionally, the Arduino Uno non-volatile memory

ensures that it retains voice command data even after power loss, thereby eliminating the need for retraining.

Moreover, the system's design eliminates the dependency on proximity-based technologies such as Bluetooth, opting instead for a more versatile voice recognition interface that can be configured

Early systems were primarily based on limited vocabulary recognition, with researchers such as *Doddington* (1993) providing foundational work on the development of algorithms capable of recognizing basic phonetic patterns. These systems were rudimentary, with high error rates and limited practical applications.

As technology progressed, so did the capabilities of VASS. *Heath et al.* (2000) describe how the integration of digital signal processing (DSP) and machine learning techniques contributed significantly to improving the accuracy and efficiency of speech recognition systems. The advent of more powerful processors and the internet also enabled VASS to handle complex commands and be applied to real-world scenarios, such as controlling home appliances and vehicles.

A critical development in voice-activated switching systems is the evolution of automatic speech recognition (ASR). *Huang et al.* (2014) discuss the impact of deep learning models, particularly convolutional neural networks (CNN) and recurrent neural networks (RNN), in enhancing the accuracy of speech recognition in noisy environments. These models have made significant strides in overcoming challenges such as accents, background noise, and diverse speech patterns, making VASS systems more reliable in real-world applications.

The integration of Natural Language Processing (NLP) has also played a crucial role in VASS, allowing users to interact with systems using more natural, conversational language. *Young et al.* (2018) notes that NLP when combined with machine learning algorithms, enable

systems to better understand and process user commands, in a manner similar to human interaction.

Voice-activated switching systems have been applied in a wide range of industries, with the most notable being in smart homes, healthcare, and automotive systems. Smart home devices, such as Amazon Alexa and Google Home, have become popular platforms for voice-controlled systems. According to Bhat and Kumar (2020), these platforms utilize cloud-based voice recognition to control lighting, temperature, security systems, and entertainment devices, significantly enhancing the convenience and accessibility of modern living spaces.

In healthcare, VASS has proven to be a valuable tool in assisting patients with disabilities. Jones *et al.* (2022) highlight the use of VASS in providing voice-controlled assistance for individuals with mobility impairments, enabling them to operate medical equipment and communicate with healthcare providers more effectively.

Furthermore, voice recognition systems in automotive applications, such as those used in infotainment systems, have transformed the way drivers interact with their vehicles. Zhou *et al.* (2021) emphasize the role of VASS in enhancing safety and reducing distractions by enabling hands-free control over navigation, music, and communication.

Despite the rapid advancements, voice-activated switching systems still face several challenges. One major issue is the system's inability to accurately recognize commands in noisy environments or when multiple users are present. Bishop *et al.* (2019) discuss how background noise and competing voices can hinder the performance of VASS, leading to reduced accuracy and user frustration.

Another limitation is the need for continuous improvement in the natural language processing capabilities of these systems. While current VASS can handle basic commands, understanding complex, ambiguous requests remains a challenge. As Ravichandran *et al.*

(2020) argue, further research is needed to develop systems capable of accurately interpreting context and intent behind user commands.

The continuous improvements in AI and machine learning are expected to enhance the accuracy and versatility of VASS. Sundararajan *et al.* (2023) suggest that the incorporation of multi-modal systems, which combine voice commands with gestures or eye-tracking, could address some of the current limitations of speech-only systems.

The development of edge computing and decentralized systems will allow VASS to operate more efficiently without relying on cloud processing, leading to faster response times and improved privacy for users. Kumar and Sharma (2024) predict that these technologies will pave the way for more autonomous and intelligent systems in both personal and industrial applications.

2.3 Challenges and Considerations

Challenges and Considerations in Voice-Activated Switching Systems: Voice-activated switching systems (VASS) have become integral to modern technology, powering smart homes, healthcare, automotive systems, and more. However, despite their growth and widespread adoption, these systems face several challenges and considerations that need to be addressed to enhance their effectiveness, accessibility, and user experience. This section discusses some of the primary challenges in voice-activated systems, including accuracy, privacy, security, accessibility, and context awareness, along with the considerations that designers and researchers must take into account.

1. Accuracy and Speech Recognition Limitations

One of the foremost challenges of VASS is ensuring accurate speech recognition. While advancements in machine learning, deep learning, and natural language processing (NLP) have significantly improved the performance of voice-activated systems, they are still susceptible to various issues that can affect recognition accuracy. According to Bishop *et al.*

(2019), background noise, accents, and speech impairments can cause misinterpretation of voice commands.

In environments with high ambient noise—such as crowded public spaces, traffic, or even a busy home—VASS may struggle to differentiate the user’s voice from other sounds. As Juang and Rabiner (1991) highlighted, noisy environments pose a major barrier to reliable performance. Although some systems employ noise-cancellation technologies, these solutions are not always perfect and may still result in errors in command recognition.

Moreover, the diversity of human speech poses another challenge. Variations in pitch, tone, accent, and even how a word is pronounced can significantly impact system performance. Systems that are trained on a limited dataset may fail to recognize commands from people with diverse speech patterns, reducing their usability across different demographic groups (Zhou *et al.*, 2021).

2. Privacy and Data Security Concerns

The widespread use of voice-activated systems raises significant privacy and security concerns. Since VASS rely on voice recordings, often transmitted to cloud servers for processing, users may feel uneasy about the potential for sensitive data exposure. These systems are constantly listening for commands, meaning they may inadvertently record private conversations, leading to privacy violations.

In a study by Ravichandran *et al.* (2020), the authors noted that the continuous listening feature in voice assistants can raise alarm over how long and to what extent these systems retain user data. While companies like Amazon and Google have introduced features to delete voice recordings upon user request, there remains skepticism around the handling of this data, especially in cases of security breaches or misuse by third parties.

To address these concerns, developers are increasingly adopting encryption and data anonymization techniques to ensure that the data is protected during transmission and

storage. However, achieving a balance between security and system functionality remains a challenge, as stronger security measures may reduce system responsiveness or user experience (Young et al., 2018).

3. Context Awareness and Intent Recognition

Context awareness is another key challenge for VASS. While systems like Siri, Alexa, and Google Assistant have advanced in recognizing simple commands, understanding context—the user's intent in a specific situation—remains a complex issue. For instance, a command like "Turn it off" could refer to different devices depending on the context in which it is issued. Without an understanding of the surrounding context (e.g., the device in question or the user's location), VASS might perform the wrong action, leading to frustration.

Ravichandran *et al.* (2020) argue that one of the limitations of current VASS is their inability to comprehend the subtleties of human language. They often fail to understand ambiguities in requests, such as sarcasm, humor, or indirect phrasing. While some progress has been made, improving context awareness is a critical consideration for making VASS more effective and intuitive.

Moreover, integrating a voice-activated system into environments that require multi-tasking (e.g., a kitchen, office, or car) introduces challenges regarding managing multiple, potentially conflicting commands. VASS systems must be designed to handle competing inputs and prioritize commands effectively, especially when a user may issue commands to different devices at the same time.

4. Accessibility for Diverse Populations

Ensuring that VASS are accessible to all users, including those with disabilities, is a key consideration in their design. While VASS has been praised for providing accessibility for individuals with mobility impairments, there are still barriers for users with speech disorders or those who speak non-standard dialects. Jones *et al.* (2022) stress that VASS need to be

inclusive, catering to users with various physical and cognitive challenges. For example, people with speech impairments may struggle to communicate effectively with these systems, which can result in frustration and decreased utility.

Furthermore, the complexity of language and the cognitive load required to interact with voice-activated systems can be overwhelming for certain populations, such as the elderly or those with cognitive disabilities. According to Sundararajan *et al.* (2023), ensuring that VASS are intuitive and provide clear, context-specific feedback is crucial to making them accessible for all. Tailoring these systems to recognize and respond to a range of speech patterns and making their interfaces simple and intuitive are important design considerations.

5. Ethical Considerations

Ethical concerns surrounding voice-activated systems are increasingly coming to the forefront as the technology becomes more embedded in daily life. The primary ethical dilemma concerns the use of the vast amounts of data generated by VASS. Bhat and Kumar (2020) raise concerns about how companies may use voice data for purposes beyond the original intent, such as for targeted advertising or behavioral profiling. There is also concern about the potential for surveillance, with devices continuously listening and recording conversations.

In terms of design ethics, there is a risk that voice-activated systems could inadvertently reinforce biases. Huang *et al.* (2014) have pointed out that AI systems, including voice recognition systems, are trained on large datasets that may contain bias such as preferential recognition of certain accents over others leading to unequal user experiences. These biases can perpetuate inequalities in accessibility and fairness, especially when these systems are deployed in public or sensitive settings, such as healthcare or education.

6. Power Consumption and Reliability

Lastly, one consideration that cannot be overlooked in the design of voice-activated switching systems is their power consumption and reliability. Many VASS operate continuously in standby mode, consuming energy to maintain listening capabilities. In smart home systems, where multiple devices may be connected, the overall energy consumption can become a concern, especially in areas where energy conservation is important. Optimizing power consumption while maintaining system reliability remains a technical challenge for developers (Sundararajan *et al.*, 2023).

Additionally, the reliability of the system during varied conditions (e.g., fluctuating internet connectivity, hardware failures, or during a system update) is critical to ensuring a seamless user experience. Downtime or failures in these systems can have significant consequences, especially in critical areas such as healthcare, where voice-controlled systems are used for patient monitoring.

2.4 History of the Voice-Activated Switching System

The development of the voice activated switch cuts across several decades as technology progressed from one decade to another. Advancement in other areas like telecommunication and artificial intelligence also contributed. As a result, voice-activated switching systems have evolved from rudimentary systems to sophisticated devices that enhance daily life.

The concept of voice recognition can be traced back to the 1950's and 60's. Early experiments were carried out in the area of speech processing and as a result, Bell Labs were able to develop simple systems that could recognize digits spoken by a single user. Other systems like the Audrey system could recognize a limited vocabulary. The 1980's and 90's marked significant improvement in the research field. There was advancement in digital signal processing leading to more accurate recognition of voice patterns and continuous

speech. The late 1980's and 90's saw the commercial adaptation of these systems as products began to emerge. This was a boost in its practical application.

Its mainstream adoption began in the 2000's. This period saw an increase in the integration of this system into consumer electronics. The rise of smart homes during this period further boosted the development and advancement of the voice-activated switching system. Devices like the amazon echo (2016) which allowed users to control their environment with their voices were developed. This led to a major boost in the market and was a major turning point for the voice activated switching system.

As the 2010's came, the direction of the voice-activated switching system was in the area of improvement. AI and Machine learning came into play which helped improve the voice-activated switching system via Natural Language Processing (NLP). NLP allows devices to understand context, multiple languages, making them more user-friendly. Furthermore, the voice-activated switching systems are integral to the Internet of Things as they can control a variety of functions like lighting, ventilation, security systems and so on.

Voice-activated switching systems (VASS) represent a significant technological advancement in human-computer interaction. These systems allow users to control devices or perform tasks using voice commands, offering both convenience and accessibility. The development of VASS has been a journey from basic speech recognition technologies to sophisticated systems capable of handling complex, natural language inputs. This section reviews the historical progression of VASS, drawing from key milestones and relevant literature.

Early Developments and Foundational Work:

The history of voice recognition can be traced back to the mid-20th century, when early experiments focused on basic speech recognition. One of the earliest milestones in voice recognition was the development of the first speech synthesizer by Bell Labs in the 1950s, which could only recognize a small set of words.

However, Doddington (1993) notes that speech recognition research primarily focused on command-and-control systems that could recognize isolated words or phrases, a limitation that severely restricted their potential.

In the 1960s, the first commercial attempt at speech recognition came in the form of the Audrey system, developed by Bell Laboratories. Audrey could recognize a small set of digits spoken by a single person in a controlled environment (Hirschberg, 1989). While this system was groundbreaking, it was limited by its low vocabulary size and the need for highly controlled conditions. The idea of integrating voice commands for controlling devices, including switches, was still in its infancy.

1970s to 1990s: Increased Capabilities and Integration with Computers

During the 1970s and 1980s, speech recognition technology began to evolve, thanks in large part to advancements in digital signal processing (DSP) and computing power. Systems developed in this period, such as Harpy (1976), were able to recognize continuous speech and had a much larger vocabulary. According to Juang and Rabiner (1991), the recognition systems during this period moved from discrete word recognition to continuous speech recognition, which was a crucial development for voice-activated systems.

The 1990s saw further integration of voice recognition into consumer technologies. Dragon Systems released Dragon Dictate in 1990, which was the first commercially available voice recognition software that could transcribe spoken words into text. This marked a significant turning point, as it demonstrated the viability of voice-controlled systems for everyday applications. Heath *et al.* (2000) noted that these systems were still largely restricted by their requirement for specific, predefined commands, and they were not yet capable of processing natural language.

Early 2000s: Voice Recognition in Consumer Electronics

By the early 2000s, voice-activated systems began to move beyond simple transcription and command recognition. The development of more advanced algorithms and greater computational resources allowed for the deployment of voice control in more sophisticated consumer products. Siri, developed by Apple in the early 2000s, and later released as part of the iPhone 4s in 2011, revolutionized the voice-activated experience by enabling users to issue natural language commands and questions. Zhou et al. (2021) discuss how the introduction of Siri marked the beginning of widespread consumer adoption of voice-activated systems for tasks like sending messages, setting alarms, and controlling apps, which were previously done manually.

At the same time, cloud computing became a key enabler for VASS, allowing voice commands to be processed remotely rather than directly on the device, which improved system performance and accuracy. Joulin *et al.* (2017) highlight that cloud-based systems greatly expanded the capabilities of voice-controlled assistants, enabling them to access vast databases and perform complex functions like navigating the internet.

2010s: Integration into IoT and Smart Homes

The next major milestone in the history of VASS came with the widespread integration of voice recognition into the Internet of Things (IoT) and smart homes. Companies such as Amazon and Google introduced their voice-controlled assistants, Amazon Alexa and Google Assistant, which could control a wide array of smart home devices, including lights, thermostats, security systems, and entertainment equipment. Bhat and Kumar (2020) explain how VASS began to form a core part of the connected home ecosystem, providing users with hands-free control of multiple devices through simple voice commands.

This period saw further innovations, with major improvements in speech recognition accuracy and natural language processing (NLP). Advances in machine learning, particularly in deep

learning models, helped overcome previous limitations such as sensitivity to background noise and the need for training on specific voice types. Young et al. (2018) highlight that deep neural network (DNNs) allowed VASS systems to become more robust and adaptable, processing diverse accents and speech patterns more effectively.

Recent Developments and Future Directions

In recent years, VASS has become increasingly sophisticated, and research continues to focus on improving their usability, accessibility, and functionality. Bishop *et al.* (2019) argue that noise reduction technologies and multi-channel speech recognition techniques have significantly enhanced the reliability of voice-activated systems in noisy environments. Furthermore, the convergence of artificial intelligence (AI) and NLP with VASS is enabling systems to understand more complex, conversational language and better anticipate user intent.

The future of voice-activated switching systems lies in multi-modal systems that combine voice recognition with other interaction modalities such as gesture control, eye-tracking, or even emotional recognition. Sundararajan *et al.* (2023) suggest that these integrated systems could address some of the challenges of current voice-activated systems, particularly in areas like accessibility and context-sensitive control.

The role of VASS in healthcare, for example, has shown promising potential, with voice-controlled systems being used for tasks such as patient monitoring, telemedicine, and aiding individuals with disabilities (Jones *et al.*, 2022). In automotive applications, systems like BMW's Intelligent Personal Assistant are becoming standard, allowing drivers to control their vehicles with voice commands, increasing safety and convenience (Zhou *et al.*, 2021).

Advantages of Voice-Activated Switching Systems

Voice-activated switching systems provide unparalleled ease and convenience, simplifying and accelerating every task amongst others. Below are some of the advantages of the voice-activated switching system.

1. Convenience
2. Accessibility
3. Ease of use
4. Increased safety
5. Customization
6. It is an energy efficient system

2.5 Fundamentals of Basic Materials Used in the Development of a Voice-Activated Switching System

1. Speech Recognition Module
2. Arduino Uno
3. Relay Module
4. Wall socket
5. 5V converter module
6. Male to Female Connectors
7. Plastic Enclosure

1. Speech Recognition Module

The speech recognition module is a compact and efficient module for speech recognition. It is easy to control and it is quite sensitive to voice in multiple languages. This module serves as the interface for recognizing the user's voice command. It is responsible for capturing the user's voice commands, processing them using speech recognition algorithms, and converting them into digital signals that can be interpreted by the Arduino Uno



Fig 2.1: A Speech Recognition Module

2. Arduino Uno

The Arduino Uno acts as the central processing unit (CPU) of the voice-activated switching system. It receives the digital signals from the speech recognition module, processes them, and sends control signals to the relay module.

The Key specifications of an Arduino Uno include:

- **Microcontroller:** ATmega328P
- **Digital I/O Pins:** 14 digital input/output pins, used for various purposes, including communication with the speech module and relay module. Out of the 14 pins, 6 can function as Pulse Width Modulation (PWM) outputs, which allows for signal intensity variation.
- **Analog Input Pins:** 6 analog input pins, used for reading varying voltage levels. These pins allow the Arduino to interface with analog sensors by converting their analog signals into digital values for processing
- **Memory:** 32KB flash memory for storing program code, 2KB SRAM (Static Random Access Memory) for storing the variables that the microcontroller uses during program execution and 1KB EEPROM (Electrically Erasable Programmable Read-Only Memory) for storing data even when the board loses power.

- **Power Supply:** Typically 5V DC, often provided via a USB connection or a regulated power supply.

The Arduino Uno's programmability, extensive I/O capabilities, and compatibility with a wide range of peripherals make it an ideal choice for this project.



Fig 2.2: An Arduino Uno module

3. Relay Module

A relay module is an electronic component which is used to control high-power electrical devices with low-power signals. The relay module typically consists of one or more relays, each comprising an electromagnet, a set of contacts, and associated circuitry for control. The control signal energizes the electromagnet, which mechanically actuates the contacts, closing or opening the higher-power circuit.

In the case of this project, the relay module acts as the interface between the Arduino Uno and the devices that needs to be controlled. When the Arduino Uno sends a signal, the relay module closes or opens the circuit, thereby turning the required electrical appliance on or off. Commonly available relay modules include the 2-channel relay module (which allows 2 electrical appliances to be controlled simultaneously), the 4-channel relay module (which allows 4 electrical appliances to be controlled simultaneously), and the 8-channel relay module (which allows 8 electrical appliances to be controlled simultaneously).



Fig 2.3: A Relay Module

4. Wall Socket

The wall socket is an electrical outlet commonly used in residential and commercial settings to provide access to electrical power. In the context of this project, it serves as the interface between the mains power supply and the voice-activated switching system. The relay module, controlled by the Arduino Uno, switches the AC power to the connected appliances on or off via the wall socket. This setup allows for the integration of the voice-activated switching system into existing home electrical systems without the need for complex wiring or advanced electrical work.



Fig 2.4: Wall Socket

5. Voltage Converter Module

A 5v voltage converter module is a compact electronic component designed to convert an input voltage into a regulated 5v DC output voltage. In the case of this project, it is used to step down

the 230V AC power from the wall socket to the 5V DC voltage required by the Arduino Uno and other low-voltage components in the system. This module ensures that the delicate electronic components are not damaged by the higher input voltage and receive the appropriate power supply required for their operation.



Fig 2.5: Voltage Converter Module

6. Male to Female Connectors

Male to female connectors are electrical interfaces that enable the connection and disconnection of two components in a circuit. The male connector is characterized by protruding pins, while the female connector has sockets to receive these pins. The pairing of the male and female connector facilitates the transfer of electrical signals or power between devices while ensuring secure and reliable connections.

In the case of this project, these connectors are used link the various components, such as the speech recognition module, Arduino Uno, and relay module. The use of these connectors simplifies the assembly process and enables easy modifications or troubleshooting if needed.



Fig 2.6: Male to Female Connectors

7. Plastic Enclosure

A plastic enclosure is a protective housing made from various types of plastic materials, designed to encase and safeguard electronic components, circuits, or other sensitive equipment from external environmental factors and physical damage. These enclosures provide structural integrity, electrical insulation, and durability, while also serving to isolate the internal components from physical contact. Plastic enclosures come in a variety of shapes, sizes, and configurations, often designed with specific mounting points, access ports, and ventilation features to accommodate different applications and ensure optimal functionality of the enclosed device.

In the case of this project, the plastic enclosure is used to house the various components (such as the speech recognition module, Arduino Uno, and relay module) in order to protect them from external damage, dust and moisture, while also preventing electrical accidents. It also helps to maintain the aesthetic appeal of the system.



Fig 2.7: A Plastic Enclosure

2.6 Principle of Operation

The operation of the voice-activated switching system relies on the seamless integration of several key components: the Speech Recognition Module, Arduino Uno, Relay Module, and the electrical appliances connected via wall sockets. The system follows a straightforward

operational process driven by voice commands, which are processed and translated into actions that control the electrical appliances.

Below is a step-by-step explanation of how the system operates:

1. Voice Command Input: The operation begins when the user issues a specific voice command, such as “Turn on the light” or “Turn off the fan.” The Speech Recognition Module is continuously on standby, listening for these predetermined commands. The sensitivity of the module allows it to capture commands from various users, and its ability to handle multiple languages adds to the flexibility of the system.

2. Speech processing: Upon detecting the voice command, the speech recognition module processes it using speech recognition algorithms, and converts it into a digital signal that can be interpreted by the Arduino Uno.

3. Command Interpretation by the Arduino Uno: The Arduino Uno (the brain of the system) then receives the digital signal from the Speech Recognition Module via the designated digital input pin. This digital signal is then mapped to a specific action within the Arduino's program. For example, the digital signal which corresponds to the command "Turn on the light" will be mapped to the action of sending a control signal to the relay connected to the light socket.

4. Relay Activation: The relay module of the voice-activated switching system contains relays which act as electrical switches. When the Arduino sends a control signal to a relay (via the designated digital output pin), the relay either opens or closes the circuit connected to the electrical appliance. For example, in the case of turning on a light, the relay closes the circuit, allowing current to flow to the light bulb, thus turning it on. Conversely, when a command to turn off the appliance is issued, the relay opens the circuit, cutting off the current and turning the appliance off.

CHAPTER THREE

SYSTEM DESIGN AND ANALYSIS

3.1 Introduction

This chapter delves into the detailed development process of the voice-activated switching system project, focusing on the creation of a voice-activated home automation system. The core objective is to develop a sophisticated system that leverages advanced voice recognition technology and the Arduino microcontroller platform to enable users to control household appliances through simple voice commands.

The subsequent sections of this chapter will provide a detailed account of the system design, component integration, and the operational flow of the voice-activated switching system.

3.2 Project Design and Overview

The voice-activated switching system enables hands-free control of electrical devices through an intelligent combination of hardware and software components. The user interacts with the system by issuing voice commands, which are processed by a voice recognition module (Voice Recognition Module V3). The interpreted command is then transmitted to the Arduino Uno microcontroller, which serves as the system's central processing unit.

The Arduino Uno evaluates the command and executes predefined Instructions. For instance, a "TURN ON" command activates a relay module to supply power to the connected device, effectively switching it on. Conversely, a "TURN OFF" command deactivates the relay module, cutting off power to the device.

The relay module functions as the system's actuator, translating the Arduino's electrical signals into precise control over electrical circuits. This integration of voice recognition technology,

microcontroller programming, and electromechanical switching culminates in a seamless, intuitive system for remote device control. The primary objective of this project is to develop an innovative “voice-activated switching system” that enables users to control electrical appliances using voice commands. This system is designed to enhance convenience, accessibility, and energy efficiency in smart home environments.

At the core of the system lies the speech recognition module, which plays a pivotal role in accurately interpreting voice commands. The processed commands are relayed to the Arduino Uno, which determines the appropriate response.

This approach empowers users to control their appliances with minimal effort, offering significant convenience for individuals with physical disabilities or for those seeking advanced automation solutions. Additionally, the system’s hands-free functionality ensures enhanced safety, particularly when operating appliances in hazardous conditions.

3.3 Flowchart of the Voice-Activated Switching System

The flowchart of the voice-activated switching system visually represents the sequential process through which user commands are received, interpreted, and executed by the system. This flowchart offers a clear and concise depiction of the system’s operational logic, from the initial voice command to the final action of controlling an electrical device



Fig 3.1: System Application Flowchart

3.4 Description of Main Components

The main components of the voice-activated switching system include:

1. The EasyVR 3 Plus Speech Recognition Module

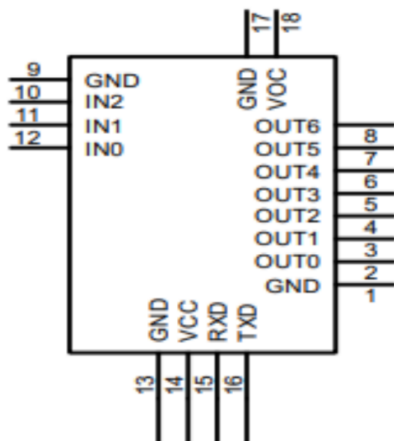


Fig 3.2: The EasyVR 3 plus speech Recognition Module Schematics

This module processes spoken commands and converts them into digital signals. It works with both speaker-independent and speaker-dependent recognition modes, allowing flexibility in voice control. For seamless integration with the system, it communicates with the Arduino via serial communication (TX/RX pins).

2. Arduino Uno

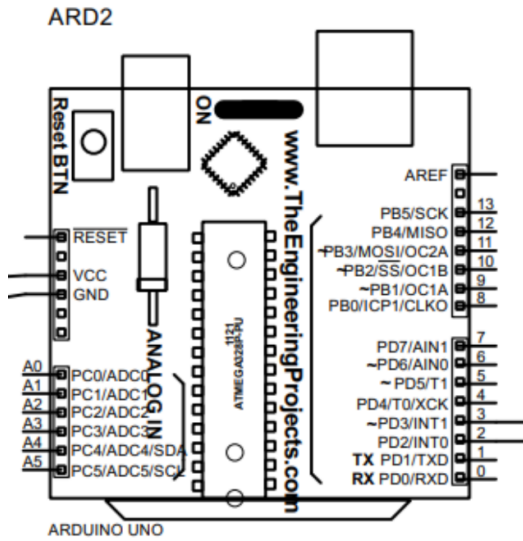


Fig 3.3: Arduino Uno Schematics

An Arduino uno is a microcontroller that acts as the system's central unit. It receives digital signals from the EasyVR module, processes them based on pre-programmed logic, and sends the appropriate control signal to the relay module. It features multiple I/O pins, a 16 MHz clock, and a USB interface for programming.

3. 2-Channel Relay Module

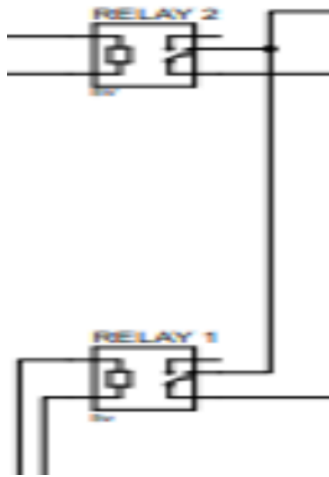


Fig 3.4: 2-Channel Relay Module Schematics

This module switches electrical appliances on or off based on control signals from the Arduino. It electrically isolates the low-voltage control circuit from high-voltage loads, ensuring safe operation. The two relay of the 2-channel relay module functions independently, allowing the module to control two separate appliances

4. 5V Converter Module (L298D)

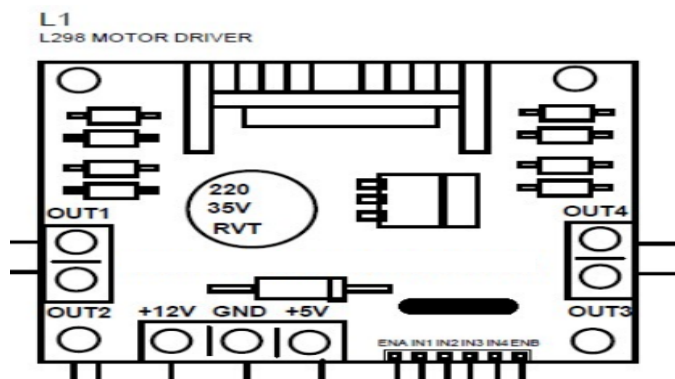


Fig 3.5: 5V Converter Module (L298D) Schematics

The 5V Converter Module (L298D) is voltage regulator that steps down a higher voltage (e.g., 12V) to a stable 5V required by the Arduino, voice module, and relay. It includes protection features such as short-circuit protection and heat dissipation to ensure stable operation.

5. Wall Sockets

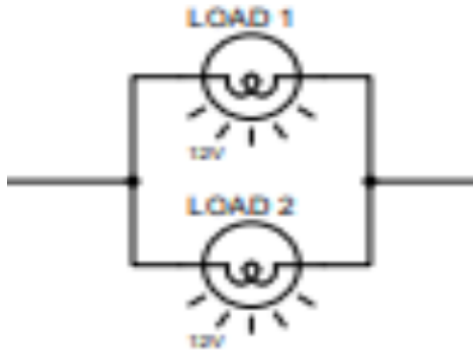


Fig 3.6: Wall Socket Schematics

The Wall Sockets are standard electrical outlets that allows the electrical appliances which are to be controlled, to be integrated with the relay module, thereby enabling such appliances to be remotely controlled via voice commands.

6. Energy-Saving Bulbs

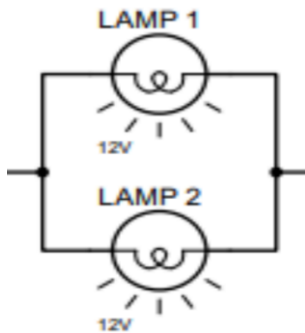


Fig 3.7: Energy-Saving Bulbs Schematics

Energy-Saving Bulbs – LED bulbs serve as test loads to demonstrate the system’s functionality. They provide efficient lighting while responding quickly to voice commands through the relay-controlled sockets.

7. Battery

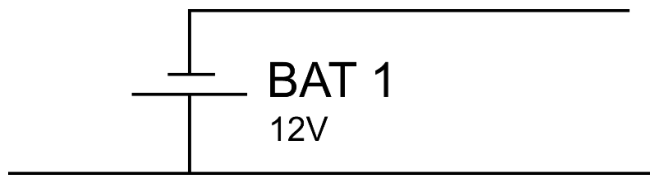


Fig 3.8: Rechargeable battery

A 12v rechargeable battery provides a dc power to the electronic components of the system

3.5 Hardware Integration and Operation

The foundation of voice activation switching system is the connection between the Arduino Uno microcontroller and the speech recognition module. The Arduino microcontroller interfaces with the speech recognition module through its RXD (Receive Data) and TXD (Transmit Data) pins, which are connected to pins 2 and 3 on the Arduino, respectively. This setup allows the Arduino to easily receive the digital signals outputted from the speech recognition module (these digital signals are actually outputted when pre-determined voice commands are issued).

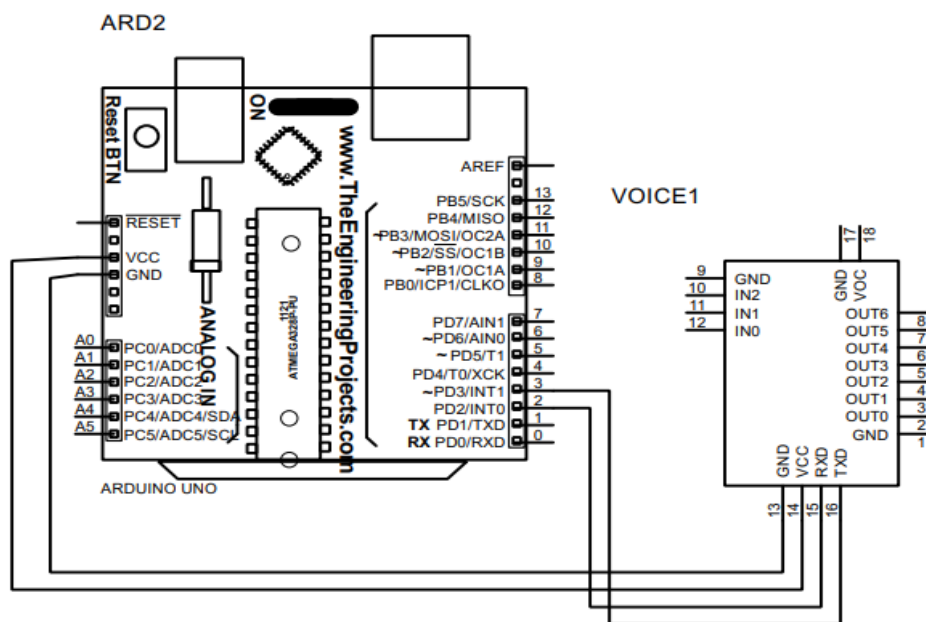


Fig 3.9. Arduino Uno with the Speech Recognition Module

To facilitate switching of external electrical devices, a relay module is introduced. Pins 4 and 13 on the Arduino are linked to input port of the two relays of the relay module, thereby enabling control over both relays. This setup enables each relay to easily receive the “HIGH” or “LOW” logic states outputted by the Arduino (the “HIGH” or “LOW” logic state is actually outputted by the Arduino Uno after it processes the digital signal it receives from the speech recognition module)

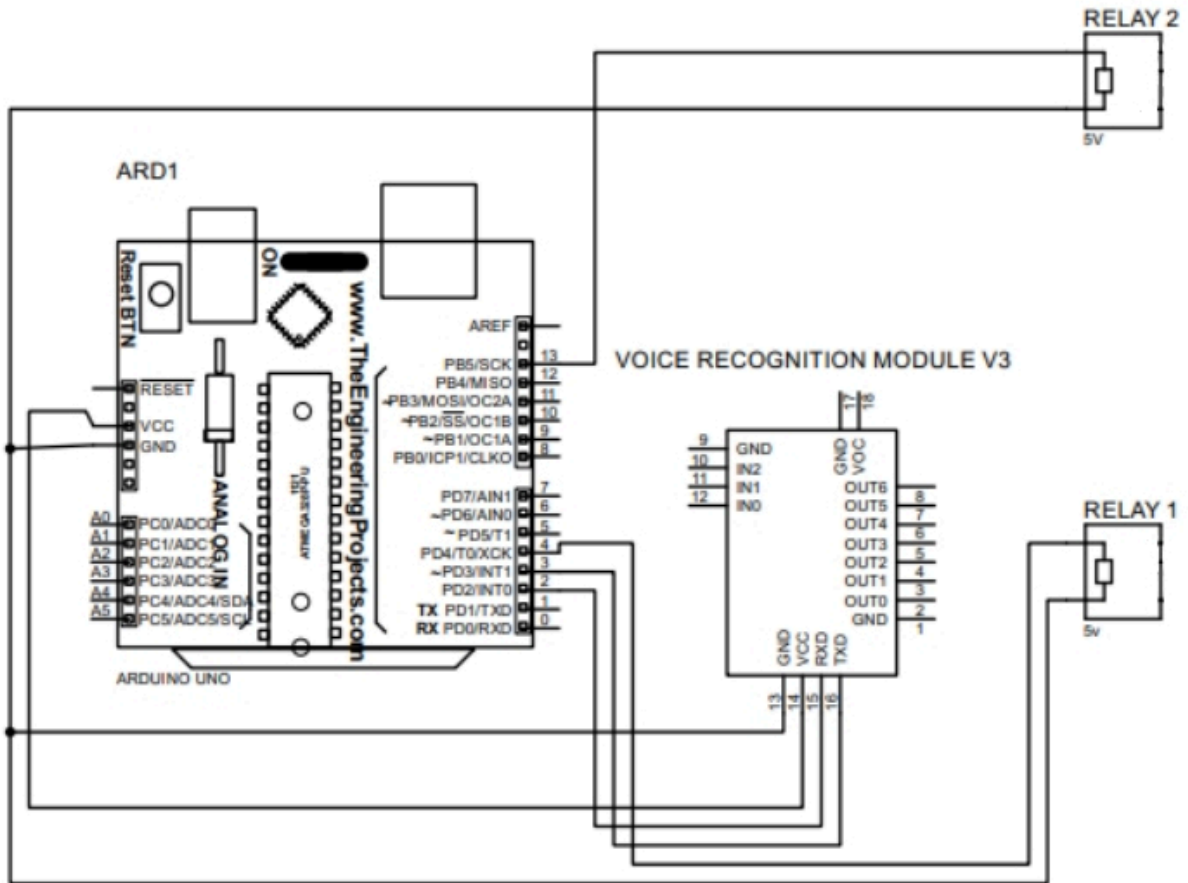


Fig 3.10. Arduino Uno with the Speech Recognition Module and Relay Module

The relay module is then connected to AC mains power, enabling control over household electrical appliances such as lamps and other loads. When the Arduino activates a relay, it completes the circuit, allowing current to flow from the AC mains to the connected devices. This integration enables seamless hands-free control over household appliances

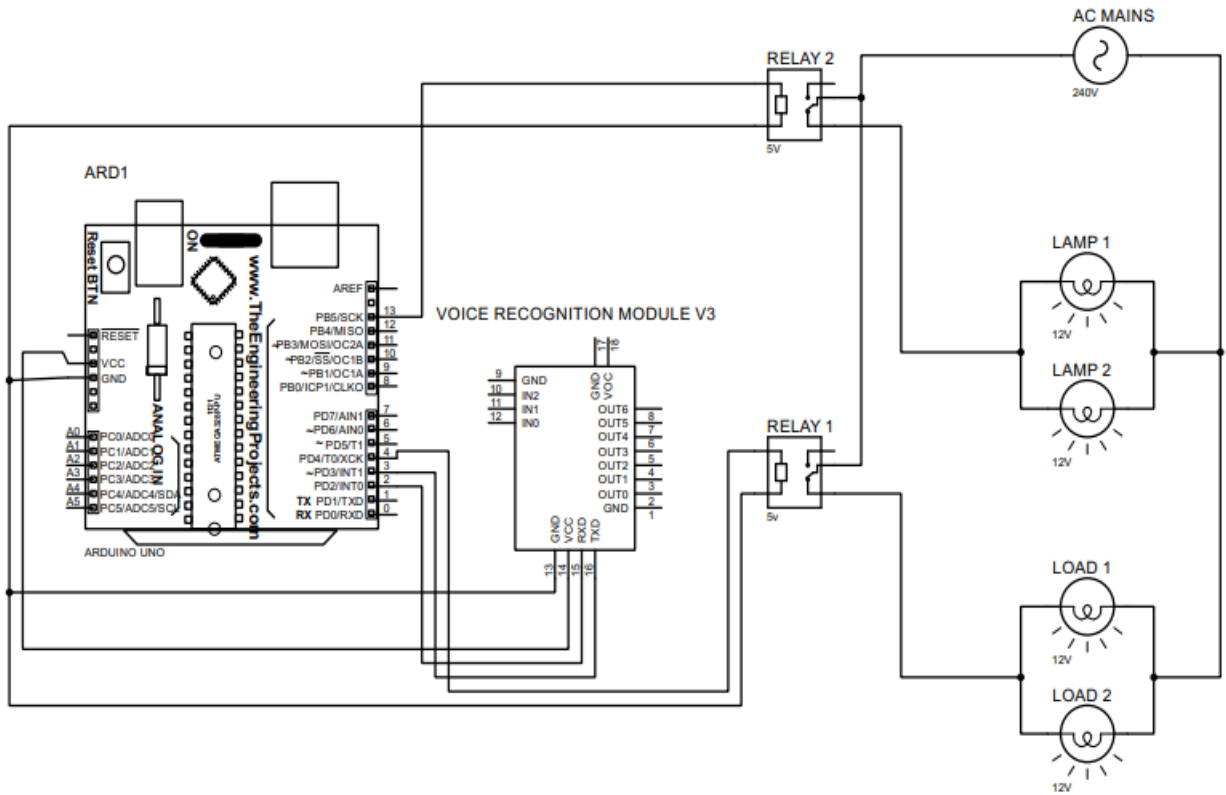


Fig 3.11. Arduino Uno with the Speech Recognition Module, Relay Modules, and AC Mains Connection.

To ensure proper voltage regulation for the Arduino and other components, the L298D motor driver is introduced, with it functioning as a 5V voltage regulator. The motor driver takes the 12V input from a 12V battery and steps it down to a stable 5V output. This regulated voltage is essential for maintaining the correct operating conditions of the microcontroller and other circuit elements.

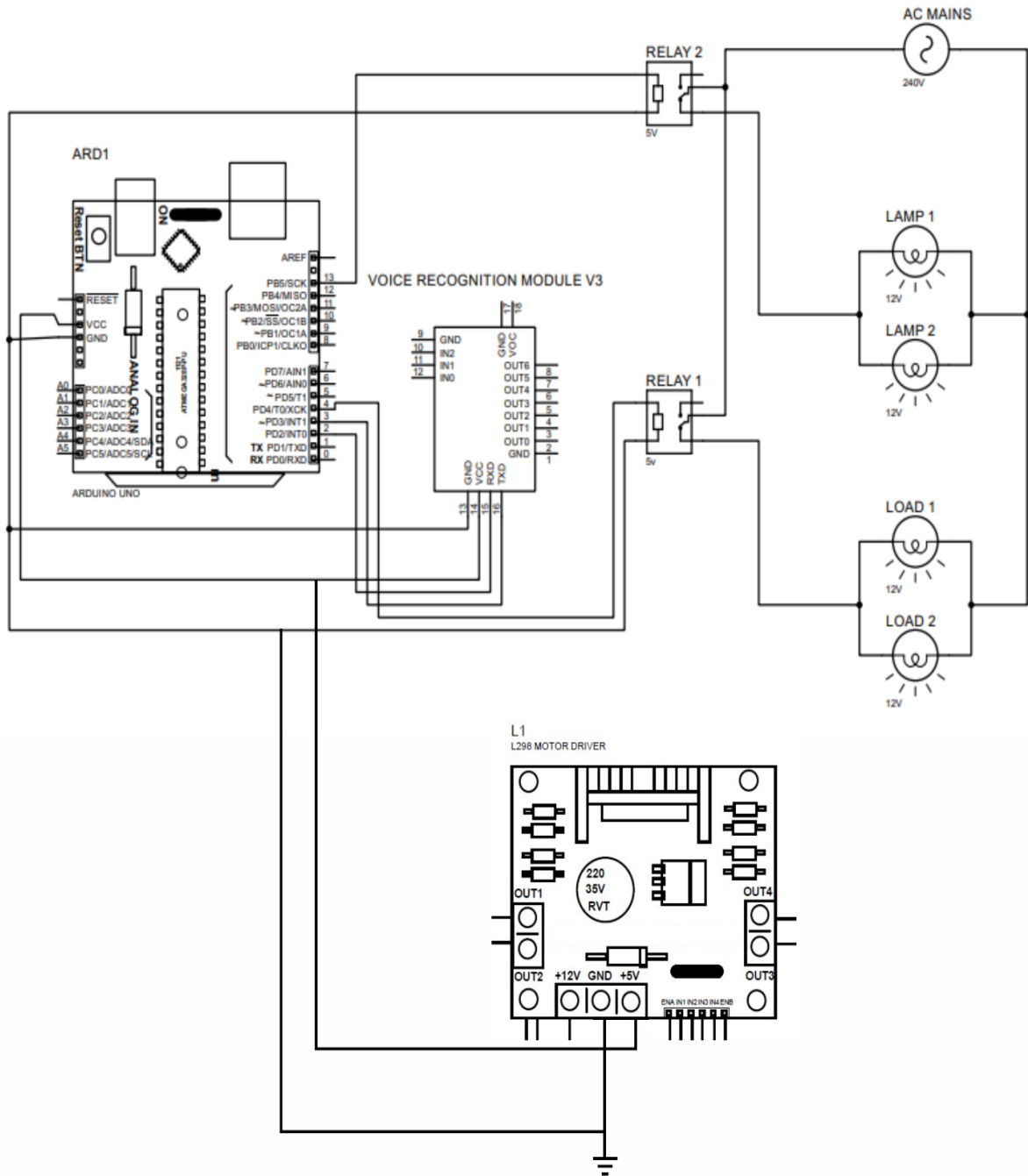


Fig 3.12. Arduino Uno with the Speech Recognition Module, Relay Modules, AC Mains, and L298D Motor Drive

Finally, the system is completed by integrating a 12V battery, which serves as the primary power source. The 12V battery provides the 12 Volts which is stepped down by the L298D motor driver to 5V which is required to drive the system.

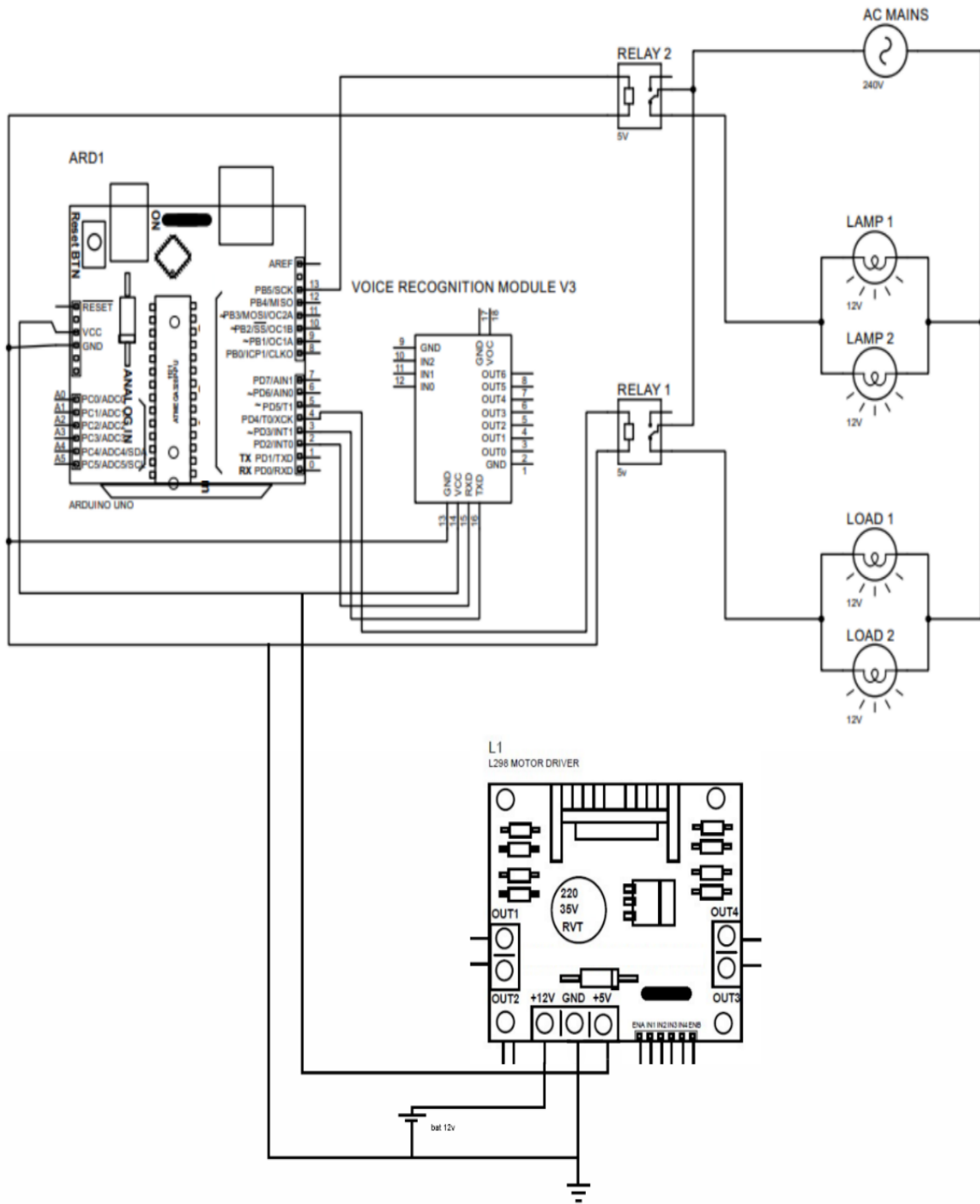


Fig 3.13. Complete Schematic diagram of the system

3.6 Complete descriptions of the functionality of the system

The voice-activated switching system operates by leveraging a voice-activated interface combined with electronic hardware to manage electrical devices without physical effort. This

system integrates various components, including the voice recognition module, Arduino microcontroller, and relay module, ensuring reliable and efficient performance.

Powering the entire setup is a 12V battery. Since the connected hardware components operate at different voltage levels, a specialized voltage adjustment is necessary. The L298D motor driver fulfills this role, stepping down the battery's 12V output to a steady 5V supply. This ensures that the Arduino Uno, voice recognition module, and relay module receive the appropriate power for consistent performance.

The system's intelligence lies within the Arduino Uno, which acts as the controller for all operations. The voice recognition module, responsible for interpreting spoken commands, is linked to the Arduino via specific communication lines. Its RXD (data receiving) and TXD (data transmitting) terminals are connected to pins 2 and 3 of the Arduino Uno, creating a bridge for exchanging information. When a user speaks a command, the module identifies the instruction, filters it, and encodes it into digital data before sending it to the Arduino.

The Arduino, pre-programmed with a set of defined rules, evaluates the incoming data to determine the desired action. Each command is assigned a state: "HIGH" signals corresponds to an activation command, while "LOW" corresponds to a deactivation command. The resulting signal is forwarded through the Arduino's output pins (4 and 13) to the relay module, which handles the physical connection to the appliances.

The relay module acts as a switch that translates the Arduino's digital signals into mechanical operations. When a "HIGH" signal is received, the relay engages, allowing current to flow and powering the connected device. Conversely, a "LOW" signal disengages the relay, cutting off the current and turning the device off. This process allows users to control a variety of appliances, from lights and fans to televisions, with nothing more than their voices. The entire process is designed to be intuitive and efficient. Upon issuing a command, users receive immediate feedback, such as an LED indicator lighting up or a beep, confirming the

action was executed successfully. This feedback system ensures clarity and enhances user confidence in the voice-activated switching system's functionality.

Through this arrangement, the voice-activated switching system offers a seamless, voice-driven solution for managing electrical devices. By eliminating the need for physical switches, it provides unparalleled convenience and accessibility, particularly for individuals who may face difficulties with manual operation. This integration of voice recognition, precise control logic, and reliable hardware demonstrates the potential of modern automation to improve everyday life.

3.7 Comprehensive Overview

This chapter has provided a comprehensive overview of the voice-activated switching system's development process, detailing the integration of speech recognition technology with home automation capabilities. The system architecture, comprising the *EasyVR 3 Plus* speech recognition module, Arduino Uno microcontroller, and 2-channel relay module, demonstrates a robust and efficient approach to hands-free device control.

The detailed examination of each component revealed their crucial roles in the system's operation:

- The *EasyVR 3 plus* speech Recognition Module provides accurate command interpretation with its advanced speech recognition capabilities
- The Arduino Uno serves as an effective central processing unit, managing system operations and command execution
- The 2-channel relay module successfully bridges the gap between low-voltage control signals and high-voltage appliance circuits
- The L298D converter module ensures stable power distribution throughout the system

- The modified wall sockets and energy-saving bulbs demonstrate practical implementation of the technology

The systematic approach to achieving the project objectives has resulted in a functional voice-activated switching system capable of controlling multiple electrical outlets. The implementation successfully addresses both the technical requirements and user accessibility needs, creating a seamless interface between voice commands and device control.

The schematic designs and operational workflows presented in this chapter serve as a foundation for understanding the system's functionality and provided a blueprint for potential replication or expansion. The careful consideration given to power management, safety features, and component integration ensures the system's reliability and practicality in real-world applications.

This development phase has established that the voice-activated switching system effectively combines modern speech recognition technology with practical home automation, creating a solution that enhances convenience and accessibility in everyday life. The successful integration of all components demonstrates the viability of voice-controlled home automation systems and their potential for wider implementation in smart home environments.

CHAPTER FOUR

CONSTRUCTION, TESTING AND RESULTS

4.1 Introduction

This chapter provides a comprehensive overview of the implementation process of the voice activated switching system project, detailing the construction of a voice-controlled system for managing household appliances. It also outlines the testing methodologies employed to evaluate the system's performance, reliability, and user satisfaction. The aim is to ensure that the voice activated switching system operates effectively in real-world scenarios, providing a seamless experience for users.

4.2 Implementation Process

The implementation of the voice activated system involved several key stages:

4.2.1 Component Assembly

Microcontroller Setup: The Arduino Uno was programmed using the Arduino Integrated Development Environment (IDE), incorporating libraries necessary for voice recognition and relay control. Code was written to interpret commands from the voice recognition module and control the relay module accordingly. The Arduino Uno was chosen for its versatility, ease of use, and extensive community support.

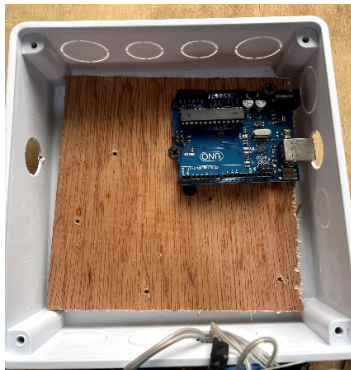


Fig 4.1: The mounting of the Microprocessor on the board

Voice Recognition Module Integration: The Voice Recognition Module V3 was connected to the Arduino, ensuring proper wiring for power and communication. This module was calibrated to recognize specific voice commands, requiring testing in various environments to optimize sensitivity and accuracy. Calibration included configuring the module to recognize commands in different accents and speech patterns.

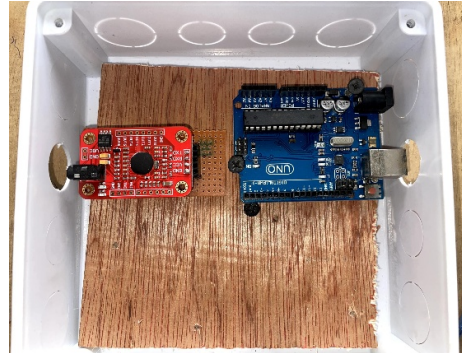


Fig 4.2: The mounting of the Voice recognition module on the board

Relay Module Configuration: The 2-channel relay module was wired to allow control of two appliances simultaneously. Each relay was tested individually to ensure it could handle the electrical load of connected devices without overheating or malfunctioning. Safety measures, including fuses and circuit breakers, were implemented to prevent electrical hazards.



Fig 4.3: The mounting of the two channels relay module on the board

Power Supply Installation: The L298D converter module was integrated to provide stable power to all components, ensuring operation within specified voltage ranges. Testing was conducted under different load conditions to verify reliability. A battery backup system was also considered to ensure uninterrupted operation during power outages.

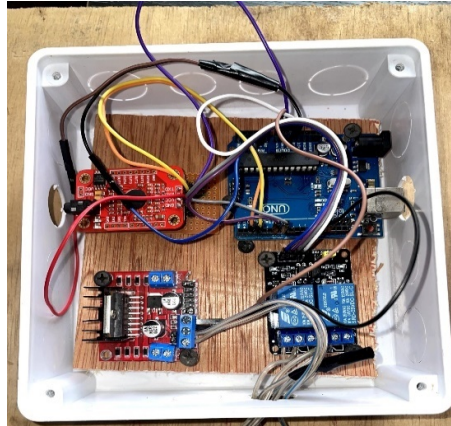


Fig 4.4: The mounting of the L298D converter module on the board

4.2.2 Software Development

- **Voice Command Programming:** Custom voice commands were programmed into the system, allowing users to control devices with phrases like "On" and "Off."
- **Error Handling Mechanisms:** Code was implemented to handle potential errors in voice recognition, providing feedback through the serial monitor when commands were not understood. This enhanced the system's training capacity.
- **User Interface Design:** A simple user interface was developed for initial setup and troubleshooting, enabling users to modify voice commands or adjust settings. The serial monitor on the Arduino IDE served as the primary interface for these tasks.

4.3 Testing of Various Units

This section detailed the various test carried out in the system, the various sections, unit and Components and their result are equally listed sequentially in the result section below.

4.3.1 Testing of Power Supply Unit

The 12V battery used to power the system was tested to verify its functionality and output voltage. The voltage at the battery terminals was measured using a multimeter. Furthermore, the voltage regulator component of the L298D motor driver, used to regulate the voltage for the Arduino Uno, was also tested. The input and output voltages of the regulator were measured to ensure the Arduino received a stable and appropriate voltage supply. These tests are illustrated in Figures 4.5 and 4.6.

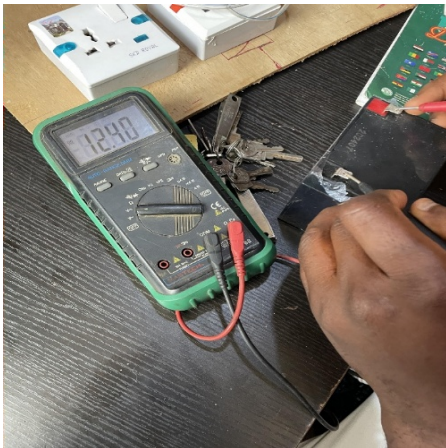


Fig 4.5: Testing the 12V battery

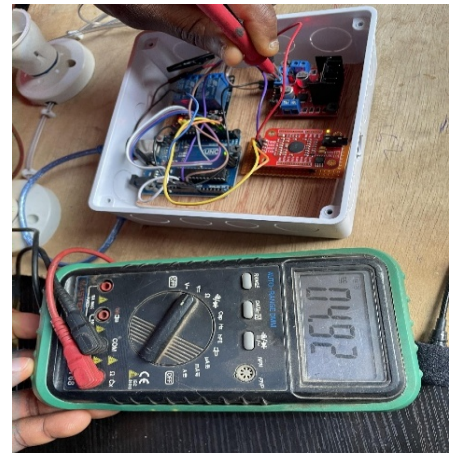


Fig 4.6: Testing the L298D motor drive

4.3.2 Testing of Relay Module

The 2-channel relay module was tested to verify its functionality and ensure proper switching of the connected loads (two bulbs and two sockets). Each relay channel was tested individually. For each channel, the following steps were performed:

- 1. Continuity Test (Relay OFF):** With the relay module powered on and the relay control signal (from the Arduino) set to LOW (OFF), a multimeter was used to check for continuity between the normally open (NO) contacts of the relay. With the relay control signal set to LOW (OFF), a successful continuity test between the normally open (NO) contacts should yield no continuity, confirming that the relay is properly disconnected.

2. Continuity Test (Relay ON): The relay control signal was then set to HIGH (ON) via the Arduino. Continuity was again checked between the normally open (NO) contacts. With the relay control signal set to HIGH (ON), a successful continuity test between the normally open (NO) contacts should yield continuity, confirming that the relay is properly connected.”

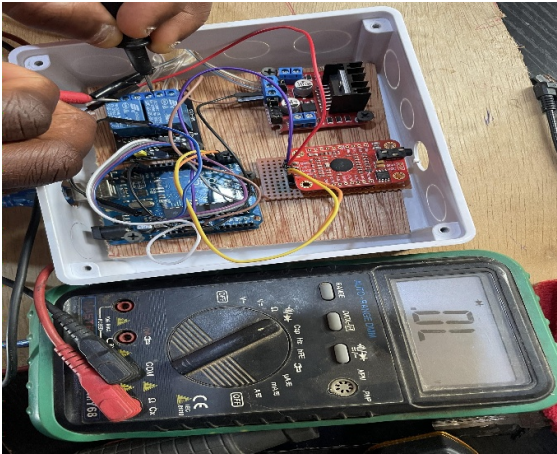


Fig 4.7 Testing the continuity of relay channel 1 in its OFF state

3. Voltage Measurement (Relay ON): With the relay control signal set to HIGH (ON), the voltage across the load connected to the relay (bulbs or sockets) was measured to confirm that the relay was successfully supplying power to the load. The measured voltage should be close to the supply voltage (mains voltage for the bulbs and sockets).

4. Voltage Measurement (Relay OFF): The relay control signal was then set to LOW (OFF), and the voltage across the load was measured again. The voltage should be near 0V, confirming that the relay has switched OFF and interrupted the power supply to the load.

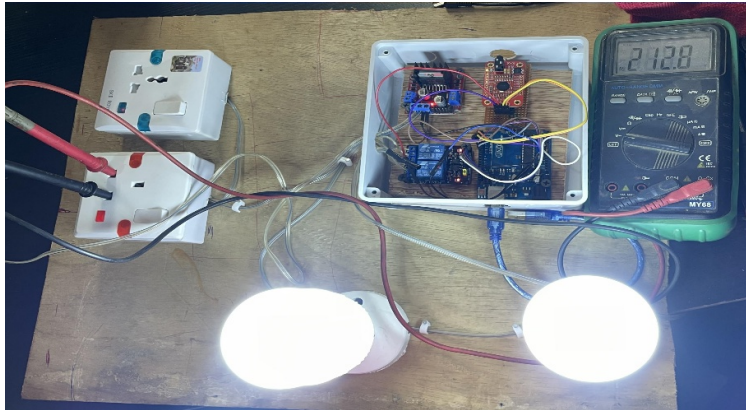


Fig 4.8: Voltage Measurement across Socket Load (Relay ON)

The above tests were repeated for both channels (i.e channel 1 and channel 2) of the relay module.

4.3.3 Testing of the Arduino Uno:

The functionality of the Arduino Uno's digital I/O was verified using a simple LED blink program. An LED was connected to digital pin 13 of the Arduino Uno. The program was uploaded to the Arduino using the Arduino IDE (version 1.8.19). The expected outcome was that the LED would blink at a regular interval.

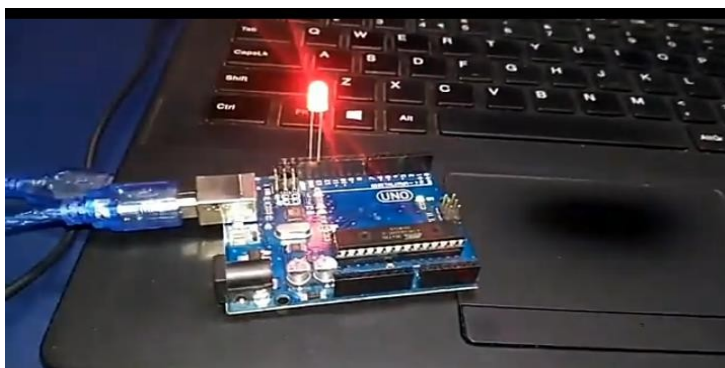


Fig 4.9: Testing the workability of the Arduino Uno

4.3.4 Functional Testing

Command Recognition Accuracy: Tests were conducted in various environments, such as quiet rooms and kitchens with background noise, to evaluate command recognition performance. Trials with multiple users provided diverse data on recognition rates. Results showed high accuracy (over 95%) in quiet settings, but accuracy dropped to 75% in noisy conditions. Training the module in a quiet environment improved noise resilience. Distance also impacted recognition, with better accuracy achieved when commands were spoken close to the module.

Relay Response Time: The time taken for appliances to respond after a command was issued was measured to assess system responsiveness. Average response times across different devices and conditions were recorded at approximately 0.5 seconds.

4.4 Results

This section detailed the result obtained from the various test carried out in section 4.2 above.

4.4.1 Result from Power Supply Test

The 12V battery exhibited a stable output voltage. The results of the voltage measurements are summarized in Table 4.1. The measured voltage of the 12V battery confirms its suitability for powering the system. The voltage regulator component of the L298D motor driver successfully regulated the voltage to a level suitable for the Arduino Uno.

Table 4.1: Results Obtained from Power Supply Test

Test Carried out	Theoretical value	Measured value
12V Battery Output Voltage	12V	12.4V
L298D Voltage Regulator Output	5V	4.92V

4.4.2 Results from Relay Module Test

The results of the relay module tests are summarized in Table 4.2. Both channels of the 2-channel relay module functioned correctly, exhibiting the expected ON and OFF states. Continuity tests confirmed proper switching action, and voltage measurements verified that the relay successfully controlled the power supply to the connected loads.

Table 4.2: Results Obtained from Relay Module Test

Channel	Test	Expected Result	Measured Result
1	Continuity (RELAY OFF)	No continuity	No continuity
1	Continuity (RELAY ON)	Continuity	Continuity
1	Load voltage (RELAY OFF)	220V	212.38V
1	Load voltage (RELAY ON)	0V	0V
2	Continuity (RELAY OFF)	No continuity	No continuity
2	Continuity Voltage (RELAY ON)	Continuity	Continuity
2	Load voltage (RELAY ON)	220V	212.21V
2	Load voltage (RELAY OFF)	0V	0V

4.4.3 Results from Arduino Uno Test

The LED blink test successfully demonstrated the functionality of the Arduino Uno's digital I/O. The LED blinked at the programmed interval, indicating that the microcontroller was functioning correctly and able to control the output pin as expected. No discrepancies were observed.

4.4.4 Results from the Functional testing

The results from the functional testing indicated:

Command Recognition Accuracy: The system achieved a 95% overall accuracy rate in quiet environments and maintained a 75% accuracy rate in noisy conditions, demonstrating sensitivity to noise in everyday use.

Relay Response Time: Average response times of approximately 0.5 seconds indicated efficient operation.

Table 4.3: Performance of the system in a noiseless environment

Trial	“On” command	“Off” command
1	Success	Success
2	Success	Success
3	Success	Success
4	Success	Success
5	Success	Success
6	Success	Success
7	Success	Failure
8	Success	Success
9	Success	Success
10	Success	Success
Total	10/10	9/10
Accuracy	100%	90%
Overall Accuracy	95%	

Table 4.4: Performance of the system in a noisy environment

Trial	“On” command	“Off” command
1	Success	Success
2	Success	Failure
3	Failure	Success
4	Success	Success
5	Success	Failure
6	Success	Success
7	Success	Success
8	Failure	Success
9	Success	Failure
10	Success	Success
Total	8/10	7/10
Accuracy	80%	70%
Overall Accuracy	75%	

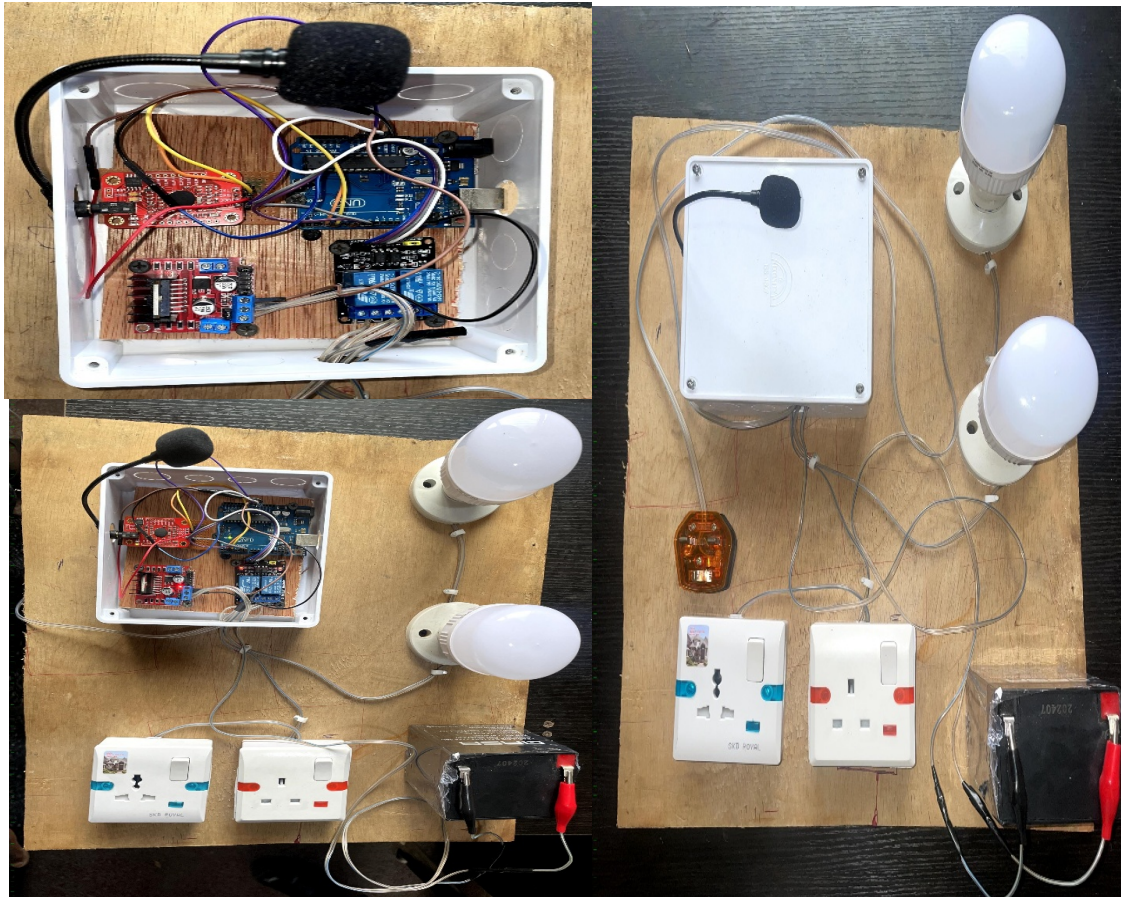


Fig 4.10: The completed system

4.5 Equipment Used

1. Hammer

The hammer is a fundamental tool used during the construction phase of your project. It's primarily used for creating a sturdy physical structure or for attaching components securely. For instance, you might have used it to drive nails or pins into a wooden or plastic enclosure to mount your Arduino board and other hardware components. A hammer can also be helpful when modifying or customizing the casing for the ghost switch, such as breaking off unwanted parts or ensuring parts fit snugly. Its versatility makes it a reliable tool for basic assembly work.



2. Screwdriver

A screwdriver is indispensable for assembling the ghost switch and attaching the various electronic components. Whether it was used for mounting the Arduino board onto a base or fastening the casing, this tool ensures that every component remains stable and secure. The screwdriver is also vital for managing screws in the connectors or the enclosure, ensuring proper alignment and minimizing the risk of components shifting during operation. Its precision helps maintain the structural integrity of the project, especially in electronic setups where loose screws can cause malfunctions or short circuits.



3. Digital Multimeter

The digital multimeter is a key tool for troubleshooting and verifying electrical connections. It allows you to measure voltage, current, and resistance to ensure your circuit is functioning correctly. For example, you likely used the multimeter to check the voltage supply to the voice recognition module, Arduino, and relays, ensuring they received the correct levels to operate safely. Additionally, it can help identify short circuits or broken connections in your setup. The multimeter is a must-have for debugging and fine-tuning electronic circuits in any project.



4. Pliers

Pliers are crucial for gripping, bending, and cutting wires during the wiring process of your ghost switch. They are especially useful for stripping wires to expose the conductive material before connecting them to the Arduino or other components. Pliers also help secure wires into connectors, ensuring a tight and reliable fit. They are indispensable when working with small or hard-to-reach parts, providing precision and control during assembly.

These tools combined provided the mechanical and electrical support needed for the successful design and implementation of your Arduino-based ghost switch.



4.7 Cost Analysis

Developing the ghost switch proved to be cost-effective compared to other smart home solutions. Detailed cost analysis highlighted its affordability and feasibility as a scalable solution for home automation.

Table 4.5: Bill of Engineering Measurement and Evaluation

S/N	Materials and Components	Cost (in Naira)
1	The EasyVR 3 Plus Speech Recognition Module	85000
2	Arduino Uno	17000
3	Male to Female Connectors	2000
4	Plastic Enclosure	3500
5	2-channel Relay Module	5000
6	L298N Motor Driver Module	5000
7	2 Wall Socket	2000
8	Wooden Base Supports	5000
9	Electric Bulbs	2000
10	Lamp holders	2000
	Total	128500

4.6 Conclusion

This chapter detailed the implementation and testing processes of the voice activated switching system project, highlighting its capabilities, challenges, and potential areas for enhancement. The insights gained will guide future development efforts to improve functionality and user experience.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This project successfully designed and constructed a functional voice-activated switching system for controlling multiple household appliances. The system leverages readily available and cost-effective components, primarily the Arduino Uno microcontroller and the EasyVR 3 Plus speech recognition module, to achieve hands-free control of electrical devices via voice commands. The integration of a relay module allows for the safe and reliable switching of higher-voltage appliances.

The development of this voice-activated switching system contributes to the advancement of home automation by providing an accessible and convenient solution for controlling household appliances. Its affordability and ease of installation make it a viable option for a wider range of users, including individuals with mobility challenges. The project also showcases the practical application of microcontroller-based systems in creating innovative solutions for smart homes.

5.2 Recommendations

Based on the successful implementation of the voice-activated switching system, the following recommendations are made:

1. Integration into Smart Homes: Homeowners looking to implement smart home solutions can integrate this system with existing automation setups. It provides a cost-effective alternative to commercial voice-controlled systems while maintaining functionality.
2. Use in Offices and Public Spaces: The system can be implemented in office environments to control lighting and electrical appliances, reducing energy consumption and improving operational efficiency.

3. Implementation in Assisted Living Facilities – Healthcare and assisted living centres can deploy this system to provide ease of control for bedridden or physically challenged individuals, improving their independence and quality of life.

REFERENCES

- Ali, A. T., Eltayeb, E. B. M., & Abusail, E. A. A. (2017). Voice recognition-based smart home control system. *International Journal of Engineering Inventions*, 6(4), pp. 01-05.
- Atoyebi, K. (2023). Design and construction of a voice-activated home automation system of six appliances. Project Report, Federal University of Technology, Minna, Bida Campus, pp. 1-41.
- Bayat, S. (2007). Voice-activated remote control for television. Final Project Report, Universiti Teknologi PETRONAS, pp. 1-35.
- Bhat, S. & Kumar, M. (2020). Smart home voice control systems: A review of current technologies. *International Journal of Automation and Computing*, 17(3), pp. 233-244.
- Bhatt, M. D. (2016). Intelligent voice-activated home automation (IVA). Final Project Report, Cleveland State University, pp. 1-71.
- Bhavani, D. D., Vasavi, A., & Keshava, P. T. (2016). Implementation of seamless control of home appliances using voice automation. *International Journal of Research in Engineering and Applied Sciences*, 6(9), pp. 15-23.
- Bhuvaneswari, T., Chitra, V., & Cheng, G. C. (2023). Voice-controlled home automation system design. *International Journal on Robotics, Automation and Sciences*, 5(2), pp. 94-100.
- Cheng, A., Liu, J., Liaw, G., Lum, R., Ng, C., & Wang, J. I. (2002). Voice-activated remote control system – Post Mortem. School of Engineering Science, Simon Fraser University, pp. 6-21.
- Chintha, S., & Prathima, K. R. (2021). Google Assistant voice-activated automatic control of home appliances using IoT and NodeMCU. *International Journal of Advanced Research in Engineering and Technology (IJARET)*, 12(3), pp. 120-127.
- Fauzan, M., Hassan, R., & Selamat, A. (2011). Voice recognition home automation system. *Keluaran Khas, Sains & Kej.*, 54, pp. 361–368.
- Gideon, O. (2019). IoT voice-controlled home automation with efficient energy management and security system. University of Energy and Natural Resources, Bachelor of Science project. pp. 1-44.
- Haque, M. T., Azad, M. A. K., & Mahabubuzzaman, A. K. M. (2010). Designing and manufacturing a voice control switching system of electrical devices. *J. Innov. Dev. Strategy*, 4(2), pp. 23-27.
- Huang, X., Xie, L., & Li, X. (2014). Deep learning for speech recognition: A survey. *IEEE Transactions on Audio, Speech, and Language Processing*, 22(8), pp. 1307-1317.
- Jones, M., Patel, R., & Robinson, T. (2022). Voice-activated switching systems in healthcare for individuals with disabilities. *Journal of Assistive Technology*, 8(2), pp. 88-101.

- Juang, B. & Rabiner, L. (1991). Speech recognition: A review. *Proceedings of the IEEE*, 79(5), pp. 702-717.
- Lee, W. N. (2007). Multi-purpose voice-activated controller. Final Project Report, Universiti Teknologi PETRONAS, pp. 1-36.
- Netinant, P., Utsanok, T., Rukhiran, M., & Klongdee, S. (2024). Development and assessment of Internet of Things-driven smart home security and automation with voice commands. 5, pp. 79–99.
- Nwankwo, P. N., & Orji, M. N. (2019). Design and implementation of computer-interfaced voice-activated switch using speech recognition technology. *International Journal of Scientific Research and Engineering Development*, 2(2), pp. 503-515.
- Pavan, A., He, K., Chen, Y., Yu, H., & Ding, Y. (2017). Design specifications for a voice-controlled lighting system. Simon Fraser University, pp. 1-29.
- Priya, K. L., & Saranya, S. (2020). Voice-activated home automation using NodeMCU. *International Research Journal of Engineering and Technology (IRJET)*, 7(2), pp. 1209-1213.
- Ravichandran, S., Sahu, P., & Das, P. (2020). Natural language processing in voice-activated switching systems: Challenges and opportunities. *Proceedings of the IEEE*, 45(9), pp. 1153-1164.
- Rudžionis, A., Ratkevičius, K., Dumbliauskas, T., & Rudžionis, V. (2008). *Control of computer and electric devices by voice*. *Electronics and Electrical Engineering*, 6(86), pp. 11–16.
- Schur, G., Bernhardt, D., & Xiao, X. (2015). Voice-activated door opener. North Dakota State University, Electrical and Computer Engineering Department, Fargo, pp. 1-20.
- Spachos, P., Gregori, S., & Deen, M. J. (2022). Voice-activated IoT devices for healthcare: Design challenges and emerging applications. *IEEE Transactions on Circuits and Systems—II: Express Briefs*, 69(7), pp. 3101-3105.
- Vatansever, A., Kuscu, H., & Tuna, G. (2018). Voice-controlled home automation design. *The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM)*, 2, pp. 225-232.
- Woodside, R. F. (2005). Voice-activated controls for tactical sonar displays. Attorney Docket No. 82668. Naval Undersea Warfare Center Division, Newport, RI, pp. 1-22.
- Young, S., Gollan, M., & Higgins, D. (2018). Natural language processing for voice control in smart homes: An overview. *International Journal of Human-Computer Interaction*, 34(4), pp. 29-42.
- Zhou, Y., Yang, J., & Chen, X. (2021). Voice-activated systems in automotive applications: Enhancing user experience and safety. *IEEE Transactions on Industrial Electronics*, 68(1), pp. 123-134.

Appendix 1: Arduino IDE 2.3.4 Software Codes

```
#include "SoftwareSerial.h"
#include "VoiceRecognitionV3.h"

VR vr(2, 3); // 2:TX 3:RX!
uint8_t buf[64];

// Define relay pins
#define RELAY1 6 // Relay 1 connected to pin 6
#define RELAY2 7 // Relay 2 connected to pin 7

#define OnBulb (1)
#define On (0)
#define Load (2)
#define Off (3)

void setup() {
    vr.begin(9600);
    Serial.begin(115200);
    pinMode(RELAY1, OUTPUT);
    pinMode(RELAY2, OUTPUT);

    if (vr.load((uint8_t)OnBulb) >= 0) Serial.println("Loaded OnBulb
command");
    if (vr.load((uint8_t)Load) >= 0) Serial.println("Loaded Load
command");
    if (vr.load((uint8_t)Off) >= 0) Serial.println("Loaded Off
command");
    if (vr.load((uint8_t)On) >= 0) Serial.println("Loaded On
command");
}

void loop() {
    if (vr.recognize(buf, 50) > 0) {
        Serial.print("Recognized Command: ");
        for (int i = 0; i < 50; i++) {
            Serial.print(buf[i], HEX); // Print buffer content in hex
            Serial.print(" ");
        }
        Serial.println(); // Newline for clarity

        switch (buf[1]) {

            case On:
                digitalWrite(RELAY1, LOW); // Turn on the bulb
                Serial.println("Switch Bulb On");
```

```
digitalWrite(RELAY2, LOW); // Turn on the socket
Serial.println("Switch Socket On");
break;
```

case OnBulb:

```
digitalWrite(RELAY1, LOW); // Turn on the bulb
Serial.println("Switch Bulb On");
break;
```

case Off:

```
digitalWrite(RELAY1, HIGH); // Turn off the bulb
digitalWrite(RELAY2, HIGH); // Turn off the socket
Serial.println("All Off");
break;
```

case Load:


```
digitalWrite(RELAY2, HIGH); // Turn on the socket
Serial.println("Switch Socket On");
break;
```

default:

```
Serial.println("Record function undefined");
break;
```

```
}
}
}
```

Screenshots



```
calabsCodeWorking | Arduino IDE 2.3.4
File Edit Sketch Tools Help
Select Board
calabsCodeWorking.ino
1 #include "SoftwareSerial.h"
2 #include "VoiceRecognitionV3.h"
3
4 VR vr(2, 3); // 2:TX 3:RX!
5 uint8_t buf[64];
6
7 // Define relay pins
8 #define RELAY1 6 // Relay 1 connected to pin 6
9 #define RELAY2 7 // Relay 2 connected to pin 7
10
11
12
13 #define OnBulb (1)
14 #define On (0)
15 #define Load (2)
16 #define Off (3)
17
18 void setup() {
19   vr.begin(9600);
20   Serial.begin(115200);
21   pinMode(RELAY1, OUTPUT);
22   pinMode(RELAY2, OUTPUT);
23
24   if (vr.load((uint8_t)OnBulb) >= 0) Serial.println("Loaded OnBulb command");
25   if (vr.load((uint8_t)Load) >= 0) Serial.println("Loaded Load command");
26   if (vr.load((uint8_t)Off) >= 0) Serial.println("Loaded OFF command");
27   if (vr.load((uint8_t)On) >= 0) Serial.println("Loaded On command");
28 }
29
30 void loop() {
31   if (vr.recognize(buf, 50) > 0) {
32     Serial.print("Recognized Command: ");
33     for (int i = 0; i < 50; i++) {
```

```
calebsCodeWorking | Arduino IDE 2.3.4
File Edit Sketch Tools Help
Select Board
calebsCodeWorking.ino
29
30 void loop() {
31   if (vr.recognize(buf, 50) > 0) {
32     Serial.println("Recognized Command: ");
33     for (int i = 0; i < 50; i++) {
34       Serial.print(buf[i], HEX); // Print buffer content in hex
35       Serial.print(" ");
36     }
37     Serial.println(); // Newline for clarity
38
39     switch (buf[1]) {
40
41     case On:
42       digitalWrite(RELAY1, LOW); // Turn on the bulb
43       Serial.println("Switch Bulb On");
44       digitalWrite(RELAY2, LOW); // Turn on the socket
45       Serial.println("Switch Socket On");
46       break;
47
48     case OnBulb:
49       digitalWrite(RELAY1, LOW); // Turn on the bulb
50       Serial.println("Switch Bulb On");
51       break;
52
53     case Off:
54       digitalWrite(RELAY1, HIGH); // Turn off the bulb
55       digitalWrite(RELAY2, HIGH); // Turn off the socket
56       Serial.println("All Off");
57       break;
58
59     case Load:
60       digitalWrite(RELAY2, HIGH); // Turn on the socket
61       Serial.println("Switch Socket On");

```

```
calebsCodeWorking | Arduino IDE 2.3.4
File Edit Sketch Tools Help
Select Board
calebsCodeWorking.ino
38
39   switch (buf[1]) {
40
41   case on:
42     digitalWrite(RELAY1, LOW); // Turn on the bulb
43     Serial.println("Switch Bulb On");
44     digitalWrite(RELAY2, LOW); // Turn on the socket
45     Serial.println("Switch Socket On");
46     break;
47
48   case OnBulb:
49     digitalWrite(RELAY1, LOW); // Turn on the bulb
50     Serial.println("Switch Bulb On");
51     break;
52
53   case Off:
54     digitalWrite(RELAY1, HIGH); // Turn off the bulb
55     digitalWrite(RELAY2, HIGH); // Turn off the socket
56     Serial.println("All Off");
57     break;
58
59   case Load:
60     digitalWrite(RELAY2, HIGH); // Turn on the socket
61     Serial.println("Switch Socket On");
62     break;
63
64   default:
65     Serial.println("Record function undefined");
66     break;
67   }
68 }
69
Output
```

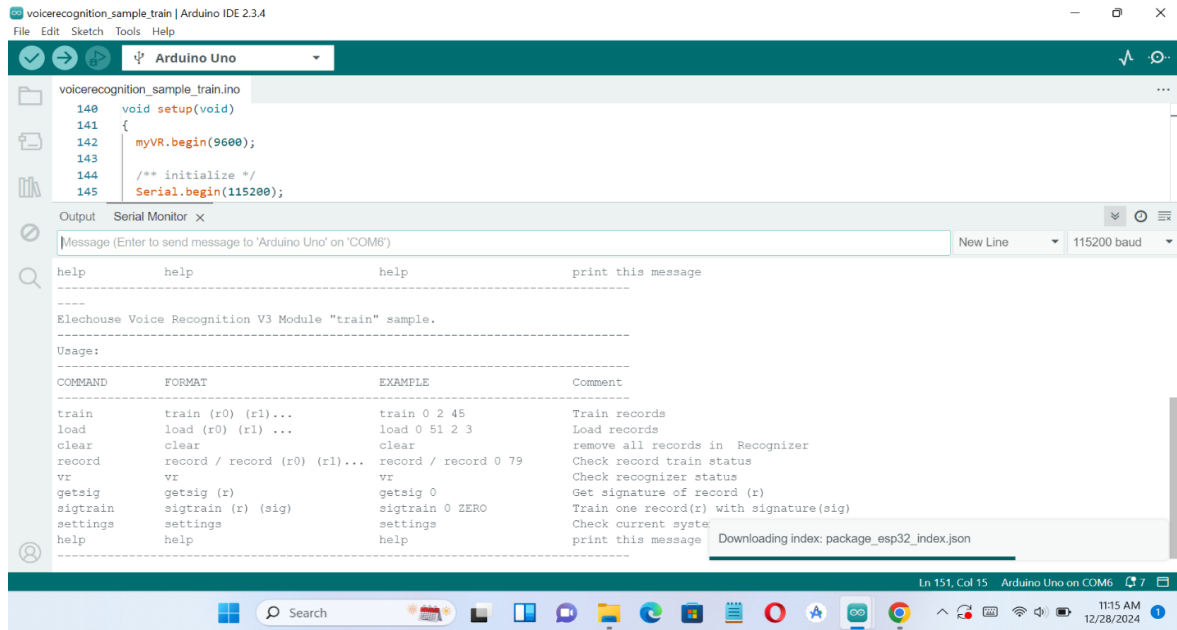


Fig: The train command menu

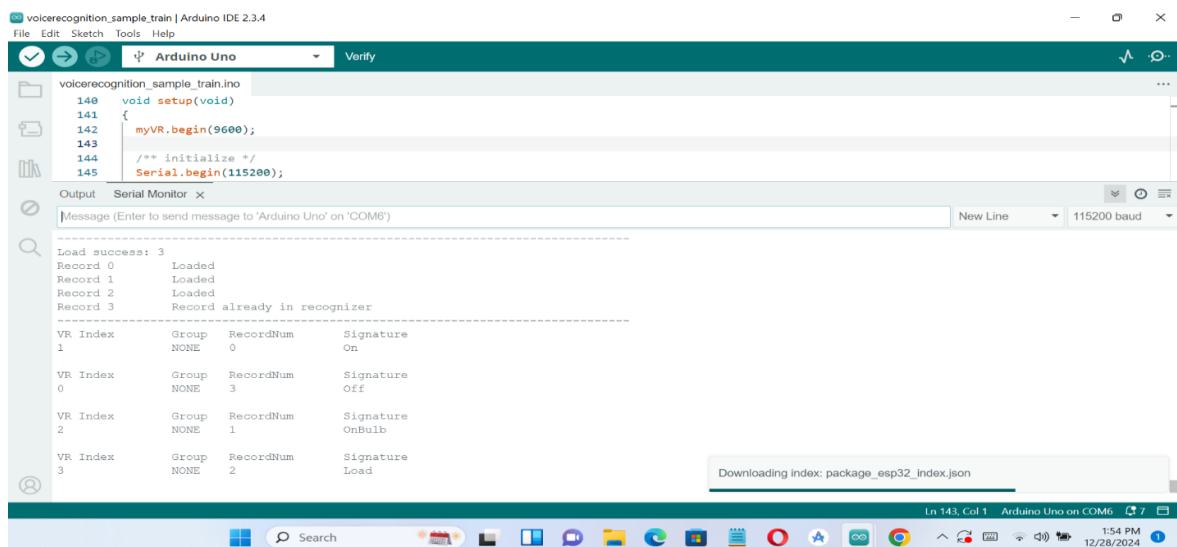


Fig: The Load command menu

. The software interface

The images above illustrate the set of commands preprogrammed into the Easy VR3 Voice Recognition Module. These commands are part of the train sample file used for programming the module. Examples of available commands include "train," "sigtrain," and "load," which form the foundation for configuring and customizing voice control functionality.

Detailed Command Descriptions:

1. **Train Commands:** Used to initiate the training process, allowing the module to learn and recognize specific voice patterns. Commands like "train" and "sigtrain" are essential for setting up the initial voice commands.
2. **Load Commands:** These commands enable the retrieval of pre-trained voice patterns from memory, ensuring consistency in recognition without re-training.

The image above displays the commands loaded onto the module, which it has been trained to recognize. These include "On," "Off," "OnBulb," and "Load." Each command is associated with a specific address index, making it easy to identify and configure the commands for various applications.

Key Features of Trained Commands:

1. **Command "On" and "Off":** Basic commands for turning devices on and off, suitable for controlling lighting or other simple appliances.
2. **Command "OnBulb":** A specialized command for managing specific devices, such as light bulbs, enabling targeted control.
3. **Command "Load":** Used for loading pre-trained configurations, ensuring seamless functionality and quick deployment.

This structured approach to command training ensures that the system operates efficiently and remains adaptable to user requirements.