Home Automation System

Submitted By

Afra Ibnat Tethye
ID: IT-18055
Session: 2017-2018

Supervised by

Dr. Md. Ahsan Habib
Professor
Department of ICT, MBSTU

In partial fulfillment of the requirement for the degree of Bachelor of Science (Engg.) in Information and Communication Technology under the Course Code of ICT-4000, Course Title: Thesis/Project, a report has been submitted.

Department of Information and Communication Technology
Mawlana Bhashani Science and Technology University
Santosh, Tangail-1902, Bangladesh
May 2023

DECLARATION

This is to certify that the project work entitled **"Home Automation System"** has been carried out by Afra Ibnat Tethye in the Department of Information and Communication Technology, Mawlana Bhashani Science and Technology University, Santosh, Tangail-1902, Bangladesh. The above research project work or any part of this work has not been submitted anywhere for the award of any degree or diploma.

Afra Ibnat Tethye
Department of ICT, MBSTU
Santosh,Tangail-1902, Dhaka, Bangladesh

.....
Signature of Candidate

Dr. Md. Ahsan Habib
Professor
Department of ICT, MBSTU
Santosh,Tangail-1902, Dhaka, Bangladesh

.....
Signature of Supervisor

APPROVAL

This is to certify that the project work submitted by Afra Ibnat Tethye(IT-18055) entitled **"Home Automation System"** has been approved by the examination committee for the partial fulfillment of the requirements for the degree of Bachelor of Science (Engineering) in the Department of Information and Communication Technology, Mawlana Bhashani Science and Technology University, Santosh, Tangail-1902, Bangladesh in May 23, 2023.

Examination Committee

Dr. Sajjad Waheed
Professor
Department of ICT, MBSTU
Santosh,Tangail-1902, Dhaka, Bangladesh

.....
Signature of Chairman

Dr. Monir Morshed
Professor
Department of ICT, MBSTU
Santosh,Tangail-1902, Dhaka, Bangladesh

.....
Signature of Chairman
Examination Committee

S.M. Shamim
Assistant Professor
Department of ICT, MBSTU
Santosh,Tangail-1902, Dhaka, Bangladesh

.....
Signature of Member
(Internal)

Dr. Md. Kamal Hossain
Professor
Department of ETE, RUET
Rajshahi, Bangladesh

.....
Signature of Member
(External)

Acknowledgements

First and foremost, praises and thanks to Allah, the Almighty, for His showers of blessings throughout our work to complete the project successfully.

We would like to express our deep and sincere gratitude to our honorable supervisor Professor Dr. Md. Ahsan Habib, Department of Information and Communication Technology, Mawlana Bhashani Science and Technology University (MBSTU), for his kindness as well as for his invaluable advice, continuous support for our project activities. Without his patience, encouragement, and careful review support, this project could not have been possible.

We feel pleasure to extend the heartiest respect and indebtedness to our honorable teachers of the same department and members of the examination committee for their valuable suggestions and inspiration throughout the study period.

We would like to thank all the staff of the Information and Communication Technology Department. Last but not least, we would like to express our sincere gratitude to our beloved parents, sisters, and brothers for their moral support and valued encouragement to reach the present stage.

Afra Ibnat Tethye

May 2023

Abstract

It's an era of the forth industrial technologies where the Internet of Things(IoT) are already started to control the world. A supervisory system replaces humans to control the home appliance and electronic devices. Home automation is an incredible invention of IoT. A smart home helps to control energy, automating things like opening and closing door locks, and turning on-off lights, fans, refrigerators, and other electronic devices through voice commands or mobile instructions by internet or Bluetooth or wifi connections. Radio Frequency Identification(RFID) is an extension of home automation that doesn't require any traditional key. It uses an RFID card which is replaceable. RFID door lock system works through an Electromagnetic wave. It is not compulsory to use an electric door lock and it optimizes the cost of home security. Electronic devices in a home or workplace can be controlled by Google Assistant or Amazon Alexa or Echo. [10] If there is no device like this, users can use mobile applications of these devices to control electronic devices and can give voice commands. The connected electronic devices which are connected to these assistants can be controlled from anywhere in the world through the Internet. Home automation system performs the basic functionality of a virtual assistant which reduces time, costs, energy, and human errors and creates a comfortable atmosphere. It is fastly emerging in technology making homes safer and better places to live. [13]

Contents

List of Figures

Chapter 1

Introduction

1.1 Basic Introduction of the System

Home automation is a supervisory system that involves computerized and automatic control of electronic devices. Tech Giants company as Google, Apple, and Amazon provide voice commands base assistants (like Google Assistant, Amazon Alexa, and Apple Siri etc) and their mobile apps. Electronic devices can also be controlled by those devices and their mobile applications. Radio Frequency can use to lock any system. By using radio frequency usually sends data. There is a sensor that checks the data. It is known as an RFID lock system. In RFID lock system users don't need any traditional key. [2]

Smart devices are the building blocks of home automation. They include smart thermostats, lighting systems, security cameras, door locks, smart plugs, sensors, and more. These devices are often connected to the internet and can communicate with each other. A central hub or controller serves as the brain of the home automation system. It enables communication between various smart devices and allows users to manage and control these devices from a single interface. [6]

With the help of mobile apps or web interfaces, homeowners can remotely monitor and control their smart devices from anywhere with an internet connection. This feature enhances convenience and provides peace of mind by allowing users to check on their home or adjust settings while away. Home automation enables the automation of routine tasks by setting up schedules or triggers. For instance, you can schedule lights to turn on or off at specific times, adjust the thermostat based on your preferences, or automate the locking and unlocking of doors.

1.2 Key Features

The main feature of the Home Automation System is to control electronic devices and lock systems through the internet by digital devices from anywhere in the world. RFID lock's secure access control and NodeMCU's IoT capabilities can create sophisticated and secure smart access systems that respond to voice commands from Alexa or Google Assistant, enabling convenient and integrated control over physical access points or electronic devices. RFID locks and NodeMCU-based electronic devices offer unique functionalities in different contexts. Here are the key features of each:

RFID Lock System:

- **Security:**

  RFID locks use radio frequency identification technology to authenticate access. Each RFID tag or card is unique, enhancing security by restricting access to authorized individuals.

- **Convenience:**

  Users can access a secured area or device by simply presenting an RFID tag or card, eliminating the need for traditional keys or codes.

- **Integration:**

  RFID locks can often be integrated with other systems, such as access control systems or smart home setups, enabling centralized management of multiple locks.

- **Audit Trails:**

  Some RFID systems offer the ability to track access history, providing insights into who accessed the area and when.

- **Customization:**

  Depending on the system, administrators can often manage user privileges and customize access rights for different RFID tags or cards.

Electronic Device Control with Alexa or Google Assistant:

- **IoT Connectivity:**

  NodeMCU, an open-source IoT platform based on the ESP8266 Wi-Fi module, enables the control of electronic devices via the internet.

- **Voice Control Integration:**

  By integrating NodeMCU with platforms like Amazon Alexa or Google Assistant, users can control connected devices using voice commands.

- **Smart Home Automation:**

  NodeMCU allows for the creation of smart home systems where users can remotely control various devices such as lights, switches, and sensors.

- **Customizability:**

  Developers can program NodeMCU to perform various tasks and respond to specific triggers, allowing for personalized automation and control.

- **Scalability:**

  NodeMCU-based systems can be expanded to control multiple devices, offering scalability in building smart environments.

1.3 Purpose and Objectives

Nowadays a common confusion is “Have I locked the door or not?”. Same things about electronic devices. By home automation, we can check it by mobile phone and also can control it. In a traditional lock system if we lose our key we need to change the lock. But in RFID lock system we can deactivate the old tag and activate a new tag. [4]

Here given some purposes and objectives of the Home Automation system:

- The main purpose and objective of the Home Automation system is energy savings and automated electronic devices and lock systems.
- To make home safer and ensure people’s better quality of life.
- To reduce time, cost, energy, and human errors and make a comfortable place to live.
- It ensures the security of people and their goods by prevention and detection.
- It connects workplaces and homes with large number of media and making it a more iterative place to live.
- Home automation guarantees universal accessibility in any environment.

1.4 Working Procedure

Home automation system involves a combination of smart devices like Amazon Alexa or Echo devices or Google Assistant or Smartphone. I tried to cover as many features to make the system more approachable. Here’s the general overview of working procedudre

- Setting up Amazon Alexa or Echo device or Google Assistant and connecting it with a home WIFI network. Then link it with an Amazon account.
- Create a Senric pro account and add the electronic devices (like bulbs, fans, refrigerators, motors, etc) which are going to be controlled.
- Add the Senric pro account with Amazon Alexa.

- Once smart devices are connected, use the Alexa app to discover and recognize them. The app will search for compatible devices on that network and add them to Alexa device's list of recognized devices.
- Then users can provide voice commands, there is also an on-off button on the Amazon Alexa account and Senric pro account.
- User can control electronic devices from anywhere in the world.
- Additionally, users can create routines that automate certain actions based on triggers or schedules. For instance, users can set up a routine to turn off all lights and lower the thermostat when the user says, "Alexa, goodnight." [3]
- In RFID lock system. Set up RFID lock and other hardware components. Scan RFID cards or tags.
- Install a RFID card activate and deactivate mobile application.
- User can also control the lock system over the internet or Bluetooth.

1.5 Outline to use this Dissertation

Beginning with the evolution of technology, home automation has rapidly evolved, integrating various smart devices such as thermostats, lighting, security cameras, and more. These devices communicate through a central control hub, often accessed and managed remotely via mobile apps or web interfaces. Automation and scheduling functionalities allow users to customize and automate tasks, enhancing comfort and efficiency while reducing energy consumption.

Chapter 2

Related works

The advent of home automation systems has witnessed an evolutionary trajectory, evolving from rudimentary systems to sophisticated, interconnected networks of smart devices. Initially conceived as a means to streamline tasks and add convenience, these systems have now evolved into complex ecosystems encompassing a myriad of devices, sensors, and intelligent technologies. Understanding this evolution, including the pivotal role played by advancements in IoT connectivity, sensor technology, and machine learning, is fundamental in comprehending the current landscape of home automation.

2.1 Historical Perspective

The roots of IoT can be traced back to the early 20th century with the invention of the first electronic sensors and actuators. Early pioneers like Nikola Tesla envisioned a world where devices could communicate wirelessly.

The term "Internet of Things" was first coined by Kevin Ashton in 1999 while working at Procter Gamble. He used it to describe a system where objects could be uniquely identified and tracked using RFID tags. RFID (Radio-Frequency Identification) technology played a significant role in the early development of IoT. Companies like Walmart and the U.S. Department of Defense adopted RFID for supply chain management.

Organizations like the International Telecommunication Union (ITU) and the IEEE began developing standards for IoT. Protocols like MQTT (Message Queuing Telemetry Transport) and CoAP (Constrained Application Protocol) were developed for efficient IoT communication

2.2 Key Technologies

The Internet of Things (IoT) relies on a variety of key technologies to enable the interconnection of devices, sensors, and objects with the Internet and each other. Sensors are devices that can measure physical or environmental properties, such

as temperature, humidity, light, motion, and more. Actuators are devices that can initiate physical actions, such as turning on a light or locking a door. These components are at the heart of IoT, providing the data input and control mechanisms for connected devices.

IoT devices need a means to communicate with each other and with central systems or the cloud. Various connectivity protocols are used, including Wi-Fi, Bluetooth, Zigbee, Z-Wave, Cellular (3G/4G/5G), LoRaWAN, and Narrowband IoT (NB-IoT). The choice of protocol depends on factors like range, power consumption, and data requirements.

IoT devices often consist of embedded systems, which are specialized computing devices with limited resources (CPU, memory). These systems are designed for specific tasks and are optimized for power efficiency. They can run on microcontrollers or specialized IoT chipsets.

IoT generates vast amounts of data. Data analytics, including machine learning and artificial intelligence (AI), are used to extract meaningful insights from this data. Predictive analytics can be applied for forecasting and optimization in various IoT applications.

2.3 Research Trends

There are many platforms related to home automation. Each offers its own unique features and capabilities, allowing users to customize and automate their homes according to their preferences.

Radio-frequency identification (RFID) technology has gained significant attention in various industries due to its ability to track and identify objects wirelessly. RFID inventory management system is an RFID-based project which focuses on inventory tracking and management. There is also a project based on RFID detection prevention known as the attendance tracking system. It tracks attendance in schools, offices, or events.

Philips Hue is a popular smart lighting system that allows users to control and automate their home lighting. It starts by setting up Philips Hue Bridge, which serves as the central hub for users Hue system. Connect the Bridge to home network using an Ethernet cable and power it up. Then, download the Philips Hue mobile app on smartphone or tablet. August Smart Lock is a smart lock system that provides keyless entry to homes. Users can control and monitor access to their homes using a mobile app or voice commands.

2.4 Scope and Importance

Home automation systems represent a revolutionary convergence of technology and lifestyle, promising to transform the way we interact with our living spaces. The burgeoning field of home automation has garnered significant attention due to its potential to enhance convenience, efficiency, security, and sustainability within domestic environments.

Scope: This integration represents a fusion of multiple cutting-edge technologies, showcasing the potential for innovation in smart home or smart office environments. It opens doors for further advancements in IoT, voice technology, and access control systems.

The scope involves leveraging RFID (Radio-Frequency Identification) technology to enable secure and efficient access control and identification. NodeMCU, based on the ESP8266 Wi-Fi module, allows for IoT (Internet of Things) capabilities, enabling control and automation of electronic devices. This involves programming NodeMCU to connect devices to a Wi-Fi network, allowing them to communicate and be controlled remotely. Integrating RFID and NodeMCU-based systems with voice-controlled assistants like Amazon Alexa and Google Assistant expands the scope by enabling users to control RFID-enabled devices or NodeMCU-connected devices using voice commands. This integration aims to provide seamless and convenient control over electronic devices or access to secure areas through voice interactions.

Importance: The integration of RFID and NodeMCU-based systems with voice assistants enhances user convenience by enabling hands-free control over electronic devices or access to secured areas within a home or office environment. Voice-controlled access and device management streamline control mechanisms, simplifying the process of interacting with various electronic systems or managing access permissions through voice commands.

Integrating these technologies with voice assistants extends accessibility, allowing users with diverse abilities or situations (e.g., disabilities, multitasking scenarios) to control devices or access spaces more easily.

2.5 Challenges and Opportunities

Integrating NodeMCU-based electronic devices with an RFID lock system may pose technical challenges due to different communication protocols, compatibility issues, or varying hardware requirements. Combining NodeMCU-based electronic device control and RFID lock systems presents various challenges and opportunities in the realm of smart access control and home automation:

1. **Security Concerns:** Ensuring the security of RFID-based access control sys-

tems and NodeMCU-connected devices is crucial. Vulnerabilities in communication or authentication processes could lead to unauthorized access or cyber threats.

2. Interoperability: Achieving seamless interoperability between RFID locks and NodeMCU devices from different manufacturers might be challenging due to proprietary protocols or standards.

3. User Experience: Providing a user-friendly experience for managing and controlling devices using RFID and NodeMCU technology may require intuitive interfaces and straightforward user instructions

4. Enhanced Access Control: Combining NodeMCU and RFID technology offers comprehensive access control solutions. RFID provides secure physical access, while NodeMCU enables remote device control, fostering a holistic security environment.

5. Remote Management: NodeMCU's IoT capabilities enable remote monitoring and control of devices connected to the system. This offers convenience and flexibility for users to manage electronic devices from anywhere with internet access.

6. Customization and Automation: Integration allows for personalized automation scenarios, such as automatically unlocking doors upon RFID authentication or triggering NodeMCU-controlled devices based on access permissions.

7. Innovation and Smart Home Development: The integration presents opportunities for innovation in the smart home ecosystem, fostering advancements in IoT, voice-controlled assistants, and secure access technologies.

8. Data Insights and Analytics: NodeMCU's connectivity can gather data on device usage and access patterns. This information can offer insights for optimizing energy consumption or refining access control strategies.

2.6 Limitaions and Research Gaps

There are several limitations and research gaps that offer avenues for further exploration and improvement:

Limitations:

Implementing RFID lock systems and NodeMCU-based control can be costly, especially when considering the expenses associated with purchasing hardware, sensors, and compatible devices. RFID locks and NodeMCU devices may have specific power requirements. Battery-powered devices could pose challenges in terms of longevity and reliability. Both RFID and NodeMCU systems may have security vulnerabilities. Ensuring robust encryption, authentication, and protec-

tion against hacking or unauthorized access is critical.

Integration challenges may arise when attempting to combine devices from different manufacturers. Ensuring compatibility and seamless integration across systems can be complex. Expanding systems to accommodate more devices or access points might be limited by the scalability of the chosen RFID and NodeMCU technologies.

Research Gaps:
Research is needed to bolster security measures in RFID and NodeMCU systems. This includes exploring advanced encryption techniques, authentication protocols, and secure data transmission methods. Investigating standardized protocols or frameworks that facilitate better interoperability between RFID locks and various NodeMCU-based devices could streamline integration efforts.

Research into energy-efficient solutions for NodeMCU-based devices, especially those running on battery power, is essential to prolong their operational lifespan and reduce power consumption. Exploring intuitive user interfaces and user experience design to simplify control and management of RFID lock systems and NodeMCU-controlled devices could enhance adoption rates.

Investigating how these systems can integrate with emerging technologies like AI, machine learning, or blockchain for enhanced security and functionality is a promising area for research. Conducting comprehensive real-world studies to evaluate the practicality, efficiency, and user acceptance of combined RFID and NodeMCU systems in diverse environments can offer valuable insights.

Addressing these limitations and researching these gaps can pave the way for more efficient, secure, and user-friendly integration of RFID lock systems and NodeMCU-based electronic device control projects. This advancement can contribute significantly to the evolution of smart access control solutions for homes, offices, and various other applications.

2.7 Summary

The realm of IoT, a dynamic and rapidly evolving field, encompasses a vast array of interconnected devices and systems. Through an extensive survey of existing literature and research, the related work in IoT showcases the multifaceted landscape of smart technologies. Studies delve into the integration of diverse devices, such as sensors, actuators, and smart appliances, forming interconnected networks that collect, exchange, and process data. These works explore the advancements in communication protocols, emphasizing interoperability and seamless connectivity across platforms. Additionally, they highlight the transformative impact of IoT in various domains, from smart homes and cities to healthcare and industrial automation.

Chapter 3

Methodology and Requirement Analysis

3.1 Introduction

Understanding the specific requirements of the users and their living environment is crucial. This involves assessing the lifestyle, preferences, and routines of the occupants to tailor the automation system accordingly. Choosing the appropriate technologies forms the backbone of a home automation system.

This step involves installing and configuring the hardware components (smart devices, sensors, controllers) and programming the software (apps, interfaces, automation rules) to ensure proper functionality.

3.2 Procedure

The main purpose of the home automation system is to enhance convenience, comfort, efficiency, and security within a household by automating and controlling various aspects of the home environment. The procedure of the project can divide into two parts.

- Connecting the hardware components with the corresponding microcontroller properly.
- Upload the code to give instruction to the microcontroller to control the devices.

3.3 User Requirement

User requirements for home automation systems can vary based on individual preferences and needs. Here are some common user requirements to consider:

- **Convenience and Comfort:**

Ability to control various devices and systems with ease, such as lighting, heating/cooling, entertainment systems, and security features. Seamless integration with voice commands, mobile apps, or other user-friendly interfaces. [5] Automation of repetitive tasks and routines to enhance convenience and save time.

- **Energy Efficiency:**

  Monitoring and control of energy usage to optimize efficiency and reduce utility costs. Ability to schedule or automate devices to turn off when not in use or adjust settings based on occupancy or environmental factors.

- **Security and Safety:**

  Users may require secure access control mechanisms to ensure that only authorized individuals can interact with the RFID module or home automation system. This can involve features such as unique identification codes, password protection, biometric authentication, or encryption techniques. [12]

- **Customization and Personalization:**

  Flexibility to personalize settings and preferences for different users or scenarios. Ability to create customized automation routines and scenes to suit individual needs and preferences.

- **Expandability and Scalability:**

  Ability to add and integrate new devices and functionalities as needed without major disruptions or complexities. Support for future technology advancements and compatibility with emerging smart home standards.

3.4 System Requirement

System requirements for a home automation system encompass the technical specifications and capabilities needed to implement and operate the system effectively. Here are some typical system requirements for a home automation system:

- **Connectivity:**

  Reliable and stable network connectivity, such as Wi-Fi or Ethernet, to connect and control smart devices. Adequate signal coverage throughout the home to ensure seamless communication between devices.

- **Centralized Control:**

  A centralized hub or controller that serves as the main control unit for the home automation system. The ability to connect and manage multiple smart devices through a single interface or platform.

- **Scalability:**

  The ability to expand the system by adding new devices or functionalities easily. Support for a large number of devices to accommodate the needs of a growing smart home.

- **Flexibility:**

  Customization options to allow users to tailor settings and automation routines to their specific needs and preferences. Compatibility with various control interfaces, such as voice commands, mobile apps, or dedicated control panels.

- **Energy Efficiency:**

  Support for energy monitoring and management features to optimize energy usage and reduce wastage. Integration with energy-efficient devices and sensors to promote eco-friendly practices.

3.5 Used Platform/Tools

3.5.1 Hardware Tools

Home automation systems consist of various hardware components that work together to enable automation and control of smart devices. Here are some common hardware tools used in home automation systems:

- **Arduino Mega 2560:** The Arduino Mega 2560 is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega 2560 board is compatible with most shields designed for the Uno and the former boards Duemilanove or Diecimila.
  In this project Arduino mega is used to control the rfid module, servo motor and liquid crystal display to control the lock system of home automation.

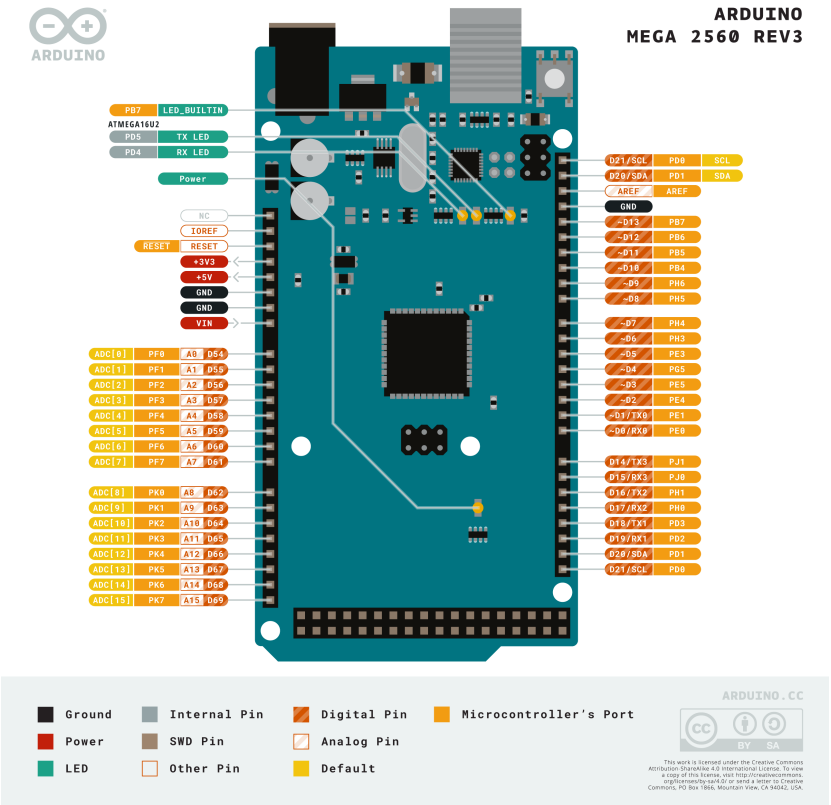

Figure 3.1: Arduino Mega 2560

- **NodeMCU (ESP8266):** The NodeMCU (Node MicroController Unit) is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266, designed and manufactured by Espressif Systems, contains the crucial elements of a computer: CPU, RAM, networking (WiFi), and even a modern operating system and SDK. That makes it an excellent choice for Internet of Things (IoT) projects of all kinds.However, as a chip, the ESP8266 is also hard to access and use. You must solder wires, with the appropriate analog voltage, to its pins for the simplest tasks such as powering it on or sending a keystroke to the "computer" on the chip. You also have to program it in low-level machine instructions that can be interpreted by the chip hardware. This level of integration is not a problem using the ESP8266 as an embedded controller chip in mass-produced electronics. It is a huge burden for hobbyists, hackers, or students who want to experiment with it in their own IoT projects.

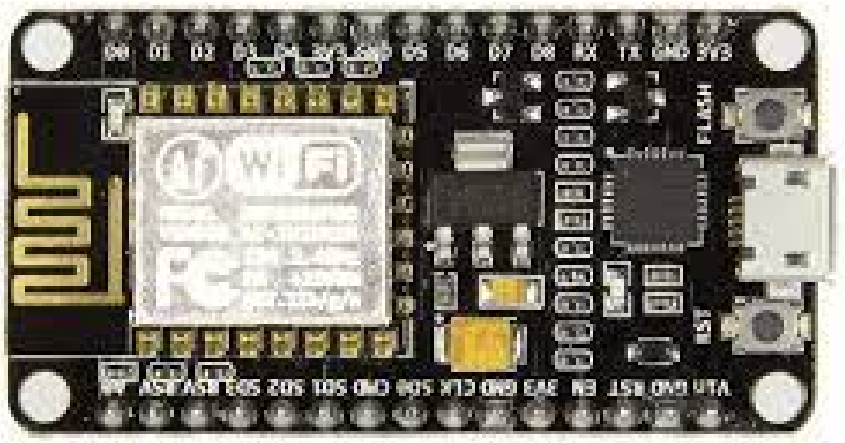

Figure 3.2: NodeMCU or ESP8266

- **RFID module & cards:** An RFID or radio frequency identification system consists of two main components, a tag attached to the object to be identified, and a reader that reads the tag. A reader consists of a radio frequency module and an antenna that generates a high-frequency electromagnetic field. Whereas the tag is usually a passive device (it does not have a battery). It consists of a microchip that stores and processes information, and an antenna for receiving and transmitting a signal. When the tag is brought close to the reader, the reader generates an electromagnetic field. This causes electrons to move through the tag's antenna and subsequently powers the chip.

Figure 3.3: RFID Module, Card and tag.

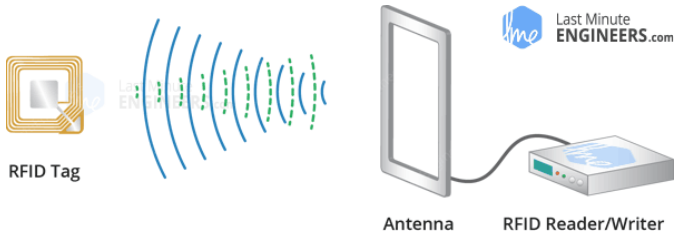

Figure 3.4: RFID working system.

- **Servo Motor:** A servo motor is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision. If you want to rotate an object at some specific angles or distance, then you use a servo motor. It is just made up of a simple motor that runs through a servo mechanism. If the motor is powered by a DC power supply then it is called a DC servo motor, and if it is an AC-powered motor then it is called an AC servo motor. For this tutorial, we will be discussing only the DC servo motor working. Apart from these major classifications, there are many other types of servo motors based on the type of gear arrangement and operating characteristics. A servo motor usually comes with a gear arrangement that allows us to get a very high torque servo motor in small and lightweight packages. Due to these features, they are being used in many applications like toy cars, RC helicopters and planes, Robotics, etc.

Figure 3.5: Servo Motor

- **Liquid-crystal display:**The liquid crystal display (LCD) panel is designed to project on-screen information of a microcomputer onto a larger screen with the aid of a standard overhead projector, so that large audiences may view on-screen information without having to crowd around the TV monitor.

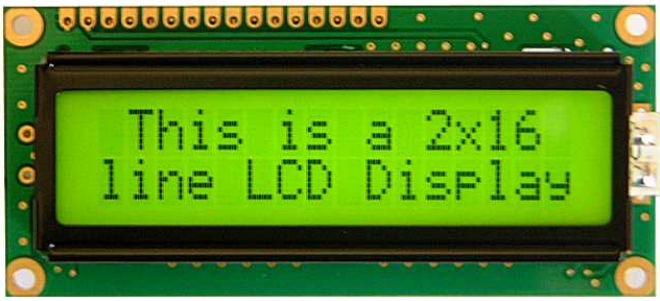

Figure 3.6: Liquid Crystal Display

- **I2C module:** This I2C 16x2 Arduino LCD Screen is using an I2C communication interface. It means it only needs 4 pins for the LCD display: VCC, GND, SDA, and SCL. It will save at least 4 digital/analog pins on Arduino. It can make the display easier. Using it can reduce the difficulty of making so that makers can focus on the core of the work.

Figure 3.7: I2C Module

- **Relay:** A relay module is a relay that's been mounted on a board with other components to provide isolation and protection. This makes them easier to use in a variety of applications. The use of relay module devices offers a simple and convenient way to control electrical equipment systems remotely.

Figure 3.8: 5V Relay Module

- **Switch:** A control is defined as an on-off switch when its function is to open or close an electrical circuit in a stable manner. If the closing or opening occurs in a non-stable or momentary manner, we are talking about a momentary on-off switch or push-button on-off switch, more briefly called on-off push-button.

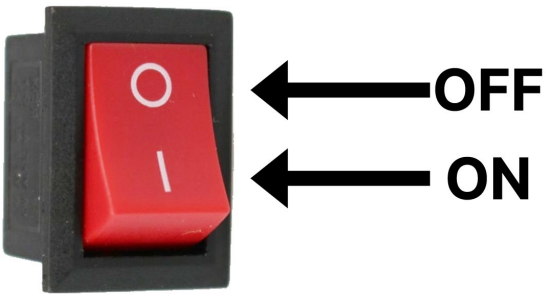

Figure 3.9: 0/1 switch

3.5.2 Software tools

Software components play a crucial role in IoT (Internet of Things) systems, including home automation. They provide the intelligence and functionality needed to control, manage, and connect devices, enabling seamless communication and automation. Here are the software components used in this project.

- **C++ (Libraries):** To give instructions on Arduino mega board and NodeMCU we used C++ language. We need some Libraries for peripheral devices.
  - **Servo.h:** In the door lock system we used a servo motor. The Servo motor is mainly used to define the lock position by changing it servo horn. In the code, we specified the position of the degree for moving.
  - **LiquidCrystal_I2C.h:** We use a liquid crystal display to see the door lock position. An I2C module to ease the use of LCD. With this Library function, one can set cursor on the display board and write on it.
  - **SPI.h:** SPI stands for the serial peripheral interface. SPI library simplifies the process of implementing SPI communication in Arduino sketches. It set up serial peripheral interface communication, sends and receives data, control the serial peripheral interface bus, and works with different data formats.
  - **MFRC522.h:** It is used to interface with the MFRC522 RFID module. By using the "MFRC522.h" library, you can interface with the MFRC522 module in a convenient and standardized manner. The key functionality of this library is the initialization of the RFID module, reading the Card's Hex number, Antenna control, cards or tags detection, etc.
  - **ESP8266WiFi.h:** It is the core library to code on NodeMCU or esp8266. As we know it is a wifi module and by using this library function we initialize wifi and set wifi name and password to connect the NodeMCU. It also helps to access point mode, network connectivity, web server functionality, DNS resolution, and client-server function etc.
  - **SinricPro.h:** It is a third-party library that is specially designed to integrate the Sinric pro service with Arduino or NodeMCU. It is used to create device management, integration with voice assistants, communication with the Sinric pro cloud and event handling and notification etc.
  - **SinricProSwitch.h:** It is created to facilitate the integration of Sinric Pro functionality with a specific type of device, specifically a switch.
- **Sinric pro website:** Sinric Pro is a cloud-based platform and service that allows developers to integrate voice control and smart home functionality into their products and applications. It provides a simplified way to connect and control IoT devices using voice commands through popular

voice assistants like Amazon Alexa and Google Assistant. Voice Control Integration, Amazon Alexa or Google Assitant Support, Device state synchronization and Device Emulation, etc are the main features of the Sinric Pro website. Sinric Pro website provides free services for three devices for more device control users need to buy a subscription yearly.

- **Amazon Alexa or Google Assistant:** Both Amazon Alexa and Google Assistant are part of larger ecosystems that allow users to control and interact with compatible smart home devices, access various services, and perform tasks through voice commands. They are designed to make everyday tasks easier and provide a hands-free experience. These devices also have a mobile application that can also do the same tasks.

- **Blink app:** The Blink app is a mobile application developed by Blink Home Security, a company owned by Amazon. The Blink app provides a user-friendly interface that allows easy access to camera controls and settings. It supports both iOS and Android devices, enabling users to monitor their homes or any other locations where they have installed Blink cameras.

3.6 Summary

Identifying the functional needs of users is paramount. Understanding the specific tasks, objectives, or problems users seek to address through IoT devices lays the groundwork for designing relevant functionalities and features.Users expect IoT devices to seamlessly interact and integrate with other devices and platforms. Compatibility with different protocols and ecosystems enhances the versatility and utility of these devices. Users should have high expectations regarding the reliability and performance of IoT devices.

Chapter 4

Implementation

4.1 Introduction

Developing a comprehensive system design involves planning the layout, identifying automation points (lighting, HVAC, security, etc.), and creating a network topology for devices to communicate effectively. Implementing an RFID lock system coupled with NodeMCU-based electronic device control presents a transformative approach to access control and smart device management.

4.2 Project Design

There can be many individual sectors in the home automation system [1]. The main sectors of home automation systems are ensuring security and electronic devices control and monitoring.

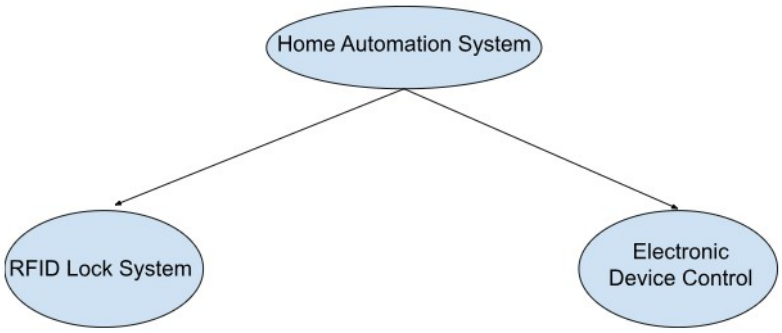

Figure 4.1: Home Automation System

RFID Lock System Design

For creating a complete and comfortable lock system I used LCD display, I2C module and a servo motor. We know every lock system has a key or password. In RFID it is known as cards or tags which contain a hexa decimal number. It is transferred by radio frequency [9].
The procedure of RFID lock system is shown in the diagram.

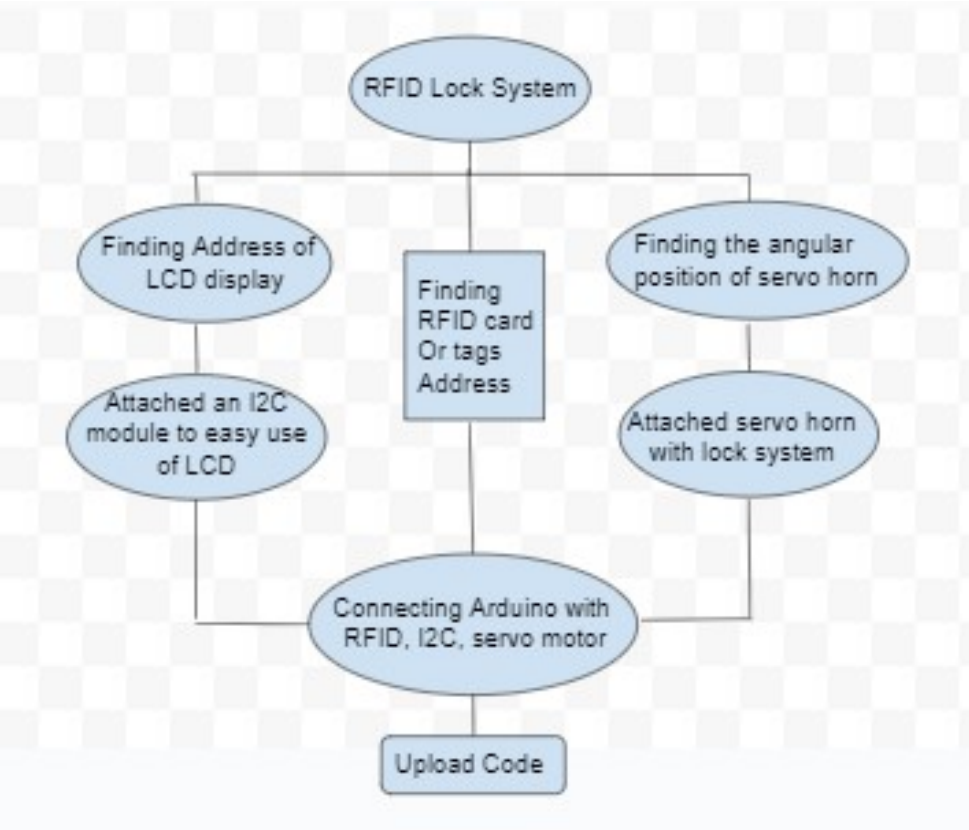

Figure 4.2: RFID System Implementation

Electronic Device Control Design

The implementation procedure of electronic devices includes voice commands on Google Assistant or Amazon Alexa [8]. The processed commands are then translated into device control instructions, which are sent to the respective electronic devices for desired actions. It can also control or monitor through web server of the Sinric Pro or corresponding sites [11]. The procedure diagram is shown below:

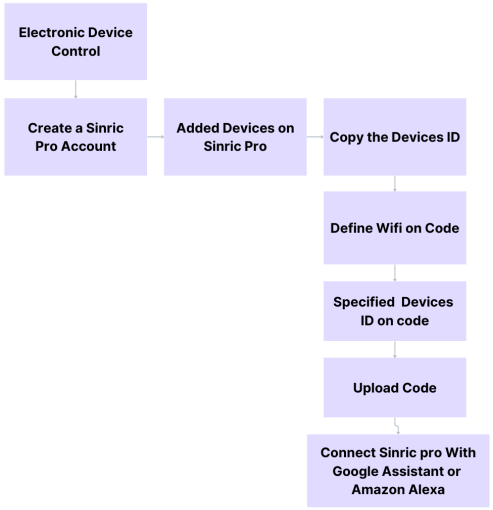

Figure 4.3: Electronic Device Control Implementation

4.3 Circuit Diagram

Circuit diagrams serve as visual blueprints that convey the structure and connections of an electrical circuit. circuit diagrams are essentially visual representations of electrical circuits, enabling comprehension, design, analysis, and troubleshooting of complex electrical systems.

4.3.1 RFID Lock System Diagram

In this circuit diagram, We attached Arduino Mega 2560 with RFID module, Servo Motor and I2C module. Here we use I2C module to connect LCD display and it makes LCD display connection easier. We can also use Arduino Uno or Arduino Nano, but digital pin numbers will be different for connecting the RFID module. Here Arduino MEGA 2560 is the main processor. If we want to get the feasibility of Bluetooth and Wifi then we can use ESP32 or ESP8266 as main processor. We can also use extra Bluetooth or Wifi module with Arduino Mega.

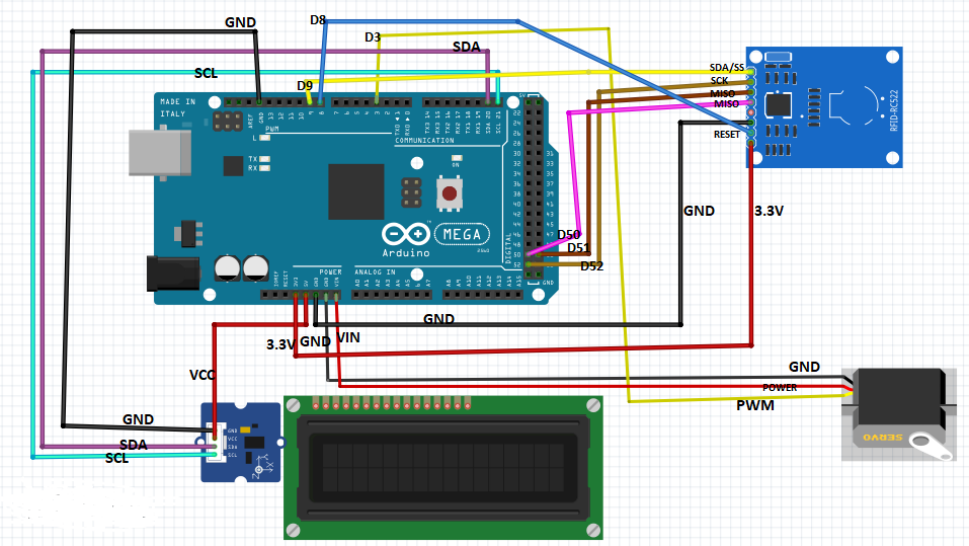

Figure 4.4: RFID Circuit Diagram

4.3.2 Electronic Device Control Diagram

We use NodeMCU (ESP8266) as the main processor. It is a microprocessor with Wifi module which connects with Wifi. In this circuit diagram, we connect NodeMCU with three relay modules and three switches. Digital pins D5, D6, and D7 of NodeMCU connect relay module IN pins. The reserve(RSV) pin is connected to the positive pin of the relay module. Relay Module is connected with electronic devices. Electronic devices is connected with relay module and AC current.

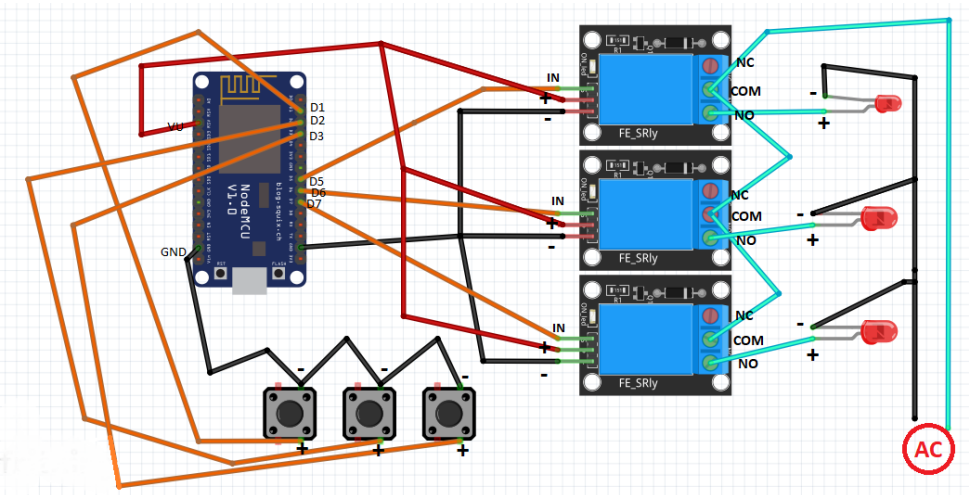

Figure 4.5: NodeMCU control Circuit Diagram

4.4 Code Implementation

Code plays a pivotal role in IoT (Internet of Things) projects and is essential for several reasons. Code implements the logic and decision-making processes that enable devices to respond to various conditions, events, or inputs. Code defines how IoT devices function and operate. It determines how devices interact with sensors, actuators, and other components to perform specific tasks or functions.

4.4.1 Test LCD

Testing LCDs helps identify any manufacturing defects or anomalies in the display, such as dead pixels, color irregularities, or backlight issues. Ensuring a defect-free display is critical for delivering a high-quality product.

Code:

```
#include <Wire.h>

#include <LiquidCrystal_I2C.h>

#include Wire.h

//initialize the liquid crystal library
//the first parameter is the I2C address
//the second parameter is how many rows are on your
//screen the third parameter is how many columns are
//on your screen LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup() {

  //initialize lcd screen
  lcd.init();
  // turn on the backlight
  lcd.backlight();
}
void loop() {
  //wait for a second
  delay(1000);
  // tell the screen to write on the top row
  lcd.setCursor(0,0);
  // tell the screen to write  hello , from on
  // the top row
  lcd.print('Hello, From');
  // tell the screen to write on the bottom row
  lcd.setCursor(0,1);
  // tell the screen to write     Arduino_uno_guy
```

```
  // on the bottom row you can change whats in the
  //quotes to be what you want it to be!
  lcd.print('Arduino_uno_gu');

}
```

4.4.2 I2C Address Finder

I2C devices communicate over a shared bus, and each device is assigned a unique address. In complex systems with multiple I2C devices, it's essential to identify which devices are connected to the bus and their respective addresses. This information is crucial for proper initialization and communication with each device.

Code:

```
#include <Wire.h> //include Wire.h library

void setup()
{
  Wire.begin(); // Wire communication begin
  Serial.begin(9600); // The baudrate of Serial
  // monitor is set in 9600
  while (!Serial); // Waiting for Serial Monitor
  Serial.println("\nI2C␣Scanner");
}

void loop()
{
  byte error, address; //variable for error and
  //I2C address
  int nDevices;

  Serial.println("Scanning...");

  nDevices = 0;
  for (address = 1; address < 127; address++ )
  {
    // The i2c_scanner uses the return value of
    // the Write.endTransmisstion to see if
    // a device did acknowledge to the address.
    Wire.beginTransmission(address);
    error = Wire.endTransmission();

    if (error == 0)
    {
```

```
      Serial.print("I2C␣device␣found␣at␣address␣0x");
      if (address < 16)
        Serial.print("0");
      Serial.print(address, HEX);
      Serial.println("␣␣!");
      nDevices++;
    }
    else if (error == 4)
    {
      Serial.print("Unknown␣error␣at␣address␣0x");
      if (address < 16)
        Serial.print("0");
      Serial.println(address, HEX);
    }
  }
  if (nDevices == 0)
    Serial.println("No␣I2C␣devices␣found\n");
  else
    Serial.println("done\n");

  delay(5000); // wait 5 seconds for the next I2C scan
}
```

4.4.3 Servo Motor set

This Arduino code demonstrates a simple access control system using an RFID (Radio-Frequency Identification) module, a servo motor, and an I2C LCD display. The system allows access based on the RFID card's unique identifier (UID). The LCD provides feedback on the system status. The system locks the door on the first authorized card scan and unlocks it on subsequent scans of the same card.

Code:

```
#include <Servo.h>
#include <LiquidCrystal_I2C.h>
#include <SPI.h>
#include <MFRC522.h>

#define SS_PIN 9
#define RST_PIN 8
String UID = "F0␣F7␣F8␣25";
byte lock = 0;

Servo servo;
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
MFRC522 rfid(SS_PIN, RST_PIN);


void setup() {
  Serial.begin(9600);
  servo.write(70);
  lcd.init();
  lcd.backlight();
  servo.attach(3);
  SPI.begin();
  rfid.PCD_Init();
}

void loop() {
  lcd.setCursor(4, 0);
  lcd.print("Welcome!");
  lcd.setCursor(1, 1);
  lcd.print("Put␣your␣card");

  if ( ! rfid.PICC_IsNewCardPresent())
    return;
  if ( ! rfid.PICC_ReadCardSerial())
    return;

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Scanning");
  Serial.print("NUID␣tag␣is␣:");
  String ID = "";
  for (byte i = 0; i < rfid.uid.size; i++) {
    lcd.print(".");
    ID.concat(String(rfid.uid.uidByte[i]
                     < 0x10 ? "␣0" : "␣"));
    ID.concat(String(rfid.uid.uidByte[i], HEX));
    delay(300);
  }
  ID.toUpperCase();

  if (ID.substring(1) == UID && lock == 0 ) {
    servo.write(70);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Door␣is␣locked");
    delay(1500);
```

```
    lcd.clear();
    lock = 1;
  } else if (ID.substring(1) == UID && lock == 1 ) {
    servo.write(160);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Door␣is␣open");
    delay(1500);
    lcd.clear();
    lock = 0;
  } else {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Wrong␣card!");
    delay(1500);
    lcd.clear();
  }
}
```

4.4.4 RFID Lock

This code defines the SS pin, RST pin, an array (byte readCard[4]) to store the UID (Unique Identifier) of the RFID card and a variable(byte a) to control the LCD cursor position. Then it initializes the object of LiquidCrystal$_I2C, MFRC522, and setup() function for initializing LCD, SPI and MFRC522 module.$

Code:

```
#include <LiquidCrystal_I2C.h>
#include <SPI.h>
#include <MFRC522.h>

#define RST_PIN 8
#define SS_PIN  9

byte readCard[4];
byte a = 0;

LiquidCrystal_I2C lcd(0x27, 16, 2);
MFRC522 mfrc522(SS_PIN, RST_PIN);

void setup() {
  Serial.begin(9600);
  lcd.init();
```

```
  lcd.backlight();
  while (!Serial);
  SPI.begin();
  mfrc522.PCD_Init();
  delay(4);
  mfrc522.PCD_DumpVersionToSerial();
  lcd.setCursor(2, 0);
  lcd.print("Put␣your␣card");
}

void loop() {
  if ( ! mfrc522.PICC_IsNewCardPresent()) {
    return 0;
  }
  if ( ! mfrc522.PICC_ReadCardSerial()) {
    return 0;
  }

  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Scanned␣UID");
  a = 0;
  Serial.println(F("Scanned␣PICC's␣UID:"));
  for ( uint8_t i = 0; i < 4; i++) {  //
    readCard[i] = mfrc522.uid.uidByte[i];
    Serial.print(readCard[i], HEX);
    Serial.print("␣");
    lcd.setCursor(a, 1);
    lcd.print(readCard[i], HEX);
    lcd.print("␣");
    delay(500);
    a += 3;
  }
  Serial.println("");
  mfrc522.PICC_HaltA();
  return 1;
}
```

4.4.5 Electronic Device Control

Electronic device control code implementation refers to the development of software or firmware that allows a microcontroller or processor to interact with and control electronic devices. Code must abstract the underlying hardware details to provide a higher-level interface for controlling electronic devices. This involves configuring pins, communication protocols, and other hardware-specific

settings.
This Arduino C++ code appears to be part of a home automation project that involves controlling relays and switches using the SinricPro library. The code includes debounce logic for flip switches to prevent rapid state changes due to noise. It utilizes the SinricPro framework to integrate devices with Amazon Alexa, Google Home, and other IoT platforms.

Code:

```
//#define ENABLE_DEBUG

#ifdef ENABLE_DEBUG
       #define DEBUG_ESP_PORT Serial
       #define NODEBUG_WEBSOCKETS
       #define NDEBUG
#endif

#include <Arduino.h>
#include <ESP8266WiFi.h>
#include "SinricPro.h"
#include "SinricProSwitch.h"

#include <map>

#define WIFI_SSID         "mce"
#define WIFI_PASS "Murad@mce"

#define APP_KEY
            "8b82090d-1cc4-4c9b-ab3d-8d6d8acdf401"
// Should look like
// "de0bxxxx-1x3x-4x3x-ax2x-5dabxxxxxxxx"
#define APP_SECRET
"4536f2f6-8363-46dd-8d5d-04f32a157618-b6ef5d03
-1a72-454c-9dfc-4686016b50f2"
 //Should look like "5f36xxxx-x3x7-4x3x-
 xexe-e86724a9xxxx-4c4axxxx-3x3x-x5xe-x9x3-333d65
 xxxxxx"

//Enter the device IDs here
#define device_ID_1   "645c71ec929949c1da63ef91"
#define device_ID_2   "645c7196929949c1da63eeb0"
#define device_ID_3   "645c71c6743f91207015f66d"

// define the GPIO connected with Relays and switches
```

```
#define RelayPin1 14  //D5
#define RelayPin2 12  //D6
#define RelayPin3 13 //D7

#define SwitchPin1 5  //D1
#define SwitchPin2 4   //D2
#define SwitchPin3 0  //D3

#define wifiLed   16   //D0

//comment the following line if you use a toggle
//switches instead of tactile buttons
//#define TACTILE_BUTTON 1

#define BAUD_RATE   9600

#define DEBOUNCE_TIME 250

typedef struct {      // struct for the std::map below
  int relayPIN;
  int flipSwitchPIN;
} deviceConfig_t;


// this is the main configuration
// please put in your deviceId, the
// PIN for Relay and PIN for flipSwitch
// this can be up to N devices...
//depending on how much pin's available on your device
// right now we have 4 devicesIds going to 4 relays
//and 4 flip switches to switch the relay manually
std::map<String, deviceConfig_t> devices = {
    //{deviceId, {relayPIN,  flipSwitchPIN}}
    {device_ID_1, {  RelayPin1, SwitchPin1 }},
    {device_ID_2, {  RelayPin2, SwitchPin2 }},
    {device_ID_3, {  RelayPin3, SwitchPin3 }}

};

typedef struct {
// struct for the std::map below
  String deviceId;
  bool lastFlipSwitchState;
  unsigned long lastFlipSwitchChange;
} flipSwitchConfig_t;
```

```
std::map<int, flipSwitchConfig_t> flipSwitches;
// this map is used to map flipSwitch PINs to deviceId
//and handling debounce and last flipSwitch state check
//it will be setup in "setupFlipSwitches" function,
//using informations from devices map

void setupRelays() {
  for (auto &device : devices) {
// for each device (relay, flipSwitch combination)
    int relayPIN = device.second.relayPIN;
// get the relay pin
    pinMode(relayPIN, OUTPUT);
// set relay pin to OUTPUT
    digitalWrite(relayPIN, HIGH);
  }
}

void setupFlipSwitches() {
  for (auto &device : devices)  {
// for each device (relay / flipSwitch combination)
    flipSwitchConfig_t flipSwitchConfig;
// create a new flipSwitch configuration
    flipSwitchConfig.deviceId = device.first;
// set the deviceId
    flipSwitchConfig.lastFlipSwitchChange = 0;
// set debounce time
    flipSwitchConfig.lastFlipSwitchState = true;
// set lastFlipSwitchState to false (LOW)--
    int flipSwitchPIN = device.second.flipSwitchPIN;
// get the flipSwitchPIN
    flipSwitches[flipSwitchPIN] = flipSwitchConfig;
// save the flipSwitch config to flipSwitches map
    pinMode(flipSwitchPIN, INPUT_PULLUP);
// set the flipSwitch pin to INPUT
  }
}

bool onPowerState(String deviceId, bool &state)
{
  Serial.printf("%s: %s\r\n", deviceId.c_str(),
  state ? "on" : "off");
  int relayPIN = devices[deviceId].relayPIN;
  //get the relay pin for corresponding device
  digitalWrite(relayPIN, !state);
```

```
  //set the new relay state
  return true;
}

void handleFlipSwitches() {
  unsigned long actualMillis = millis();
  // get actual millis
  for (auto &flipSwitch : flipSwitches) {
  // for each flipSwitch in flipSwitches map
    unsigned long lastFlipSwitchChange =
    flipSwitch.second.lastFlipSwitchChange;
// get the timestamp when flipSwitch was pressed
//last time (used to debounce / limit events)

    if (actualMillis - lastFlipSwitchChange>
    DEBOUNCE_TIME) {
    // if time is > debounce time...
      int flipSwitchPIN = flipSwitch.first;
      // get the flipSwitch pin from configuration
      bool lastFlipSwitchState =
      flipSwitch.second.lastFlipSwitchState;
      // get the lastFlipSwitchState
bool flipSwitchState = digitalRead(flipSwitchPIN);
// read the current flipSwitch state
      if (flipSwitchState != lastFlipSwitchState){
//if the flipSwitchState has changed...
#ifdef TACTILE_BUTTON
        if(flipSwitchState) {
// if the tactile button is pressed
#endif
          flipSwitch.second.lastFlipSwitchChange
          = actualMillis;
// update lastFlipSwitchChange time
        String deviceId = flipSwitch.second.deviceId;
// get the deviceId from config
        int relayPIN = devices[deviceId].relayPIN;
// get the relayPIN from config
        bool newRelayState = !digitalRead(relayPIN);
// set the new relay State
        digitalWrite(relayPIN, newRelayState);
//set the trelay to the new state
    SinricProSwitch &mySwitch = SinricPro[deviceId];
// get Switch device from SinricPro
        mySwitch.sendPowerStateEvent(!newRelayState);
// send the event
```

```
#ifdef TACTILE_BUTTON
        }
#endif
        flipSwitch.second.lastFlipSwitchState
        = flipSwitchState;
        // update lastFlipSwitchState
      }
    }
  }
}

void setupWiFi()
{
  Serial.printf("\r\n[Wifi]: Connecting");
  WiFi.begin(WIFI_SSID, WIFI_PASS);

  while (WiFi.status() != WL_CONNECTED)
  {
    Serial.printf(".");
    delay(250);
  }
  digitalWrite(wifiLed, LOW);
  Serial.printf("connected!\r\n[WiFi]: IP-Address is
  %s\r\n", WiFi.localIP().toString().c_str());
}

void setupSinricPro()
{
  for (auto &device : devices)
  {
    const char *deviceId = device.first.c_str();
  SinricProSwitch &mySwitch = SinricPro[deviceId];
    mySwitch.onPowerState(onPowerState);
  }

  SinricPro.begin(APP_KEY, APP_SECRET);
  SinricPro.restoreDeviceStates(true);
}

void setup()
{
  Serial.begin(BAUD_RATE);

  pinMode(wifiLed, OUTPUT);
  digitalWrite(wifiLed, HIGH);
```

```
  setupRelays();
  setupFlipSwitches();
  setupWiFi();
  setupSinricPro();
}

void loop()
{
  SinricPro.handle();
  handleFlipSwitches();
}
```

4.5 Output of the System

RFID lock:
In this system, when the user wants to open the lock. The LCD will display "Scanning..." on the screen. If the RFID tag is right then the door will open and display "Door is Open...". Otherwise, it will display "Wrong Card!". In the same way, when the user wants to close the door LCD will display "Scanning on the screen. If the RFID tag is right then the door will close and display "Door is Closed...". Otherwise, it will display "Wrong Card!.

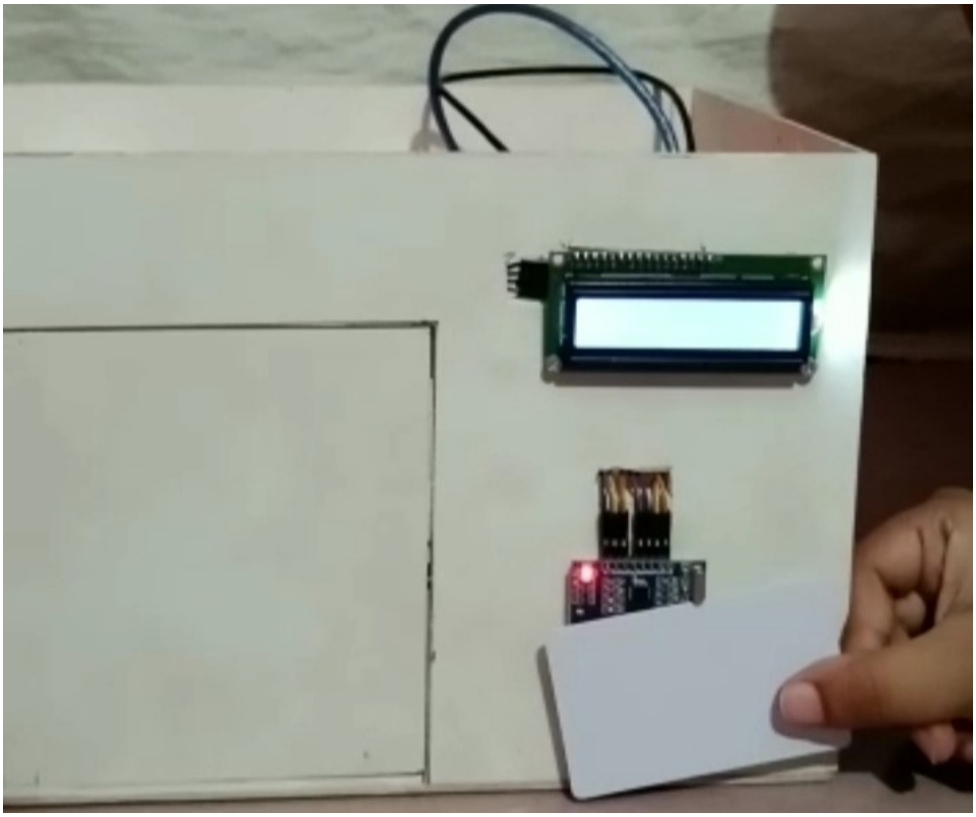

Figure 4.6: RFID Lock System

Electronic Device Control: In the NodeMCU-based electronic device control system, First, we have to create a Sinric pro account and add the devices. Used the device IDs on the code. Then upload the code on NodeMCU. Attach the Sinris Pro with Google Assistant or Amazon alexa. If we use Google Assistant, we need to pay. But Amazon Alexa is free to use.

In Sinric Pro, We need to give every device a unique name. When we command Google Assistant or Amazon Alexa saying the name of the device. The device starts working based on the command. If user commands "Turn on study light", Study light will turn on.

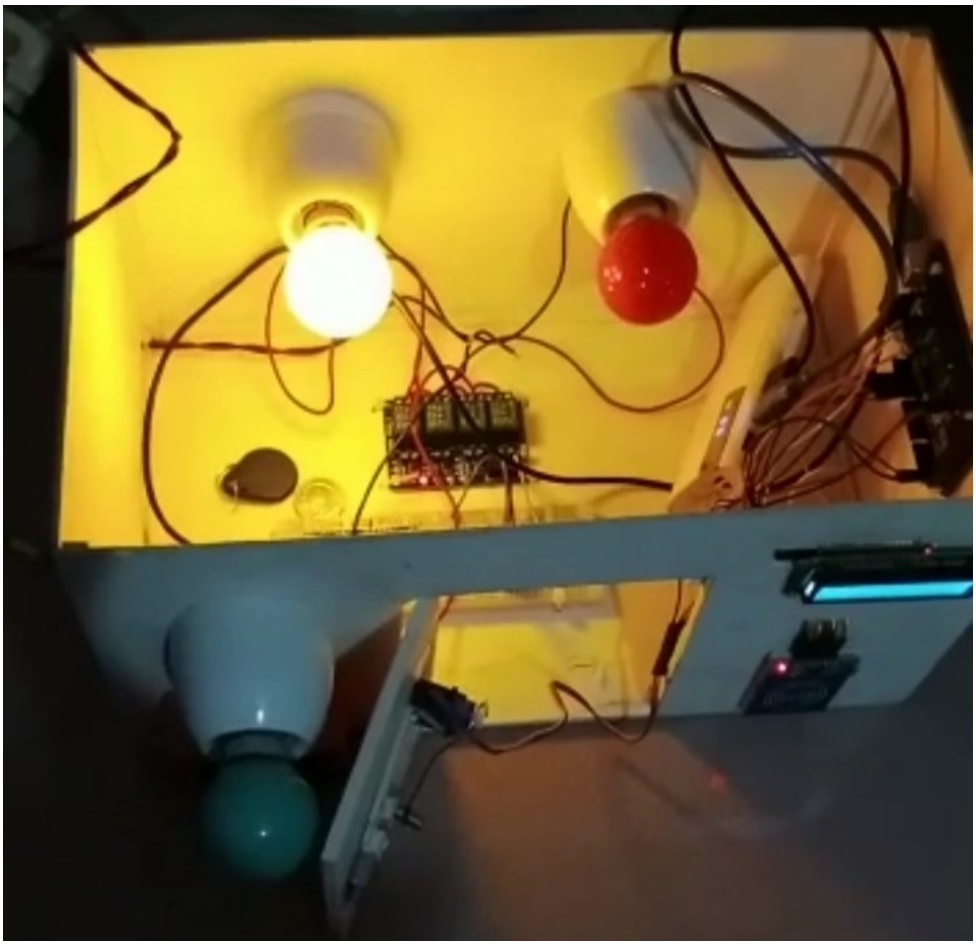

Figure 4.7: Study Light Turn On

We can Turn on lights by giving voice commands, Sinric Pro switch, Google Assistant or Amazon Alexa switch, and for offline use the electronic on-off switch. We can also turn on all lights by Giving voice commands saying "Turn On All Lights".

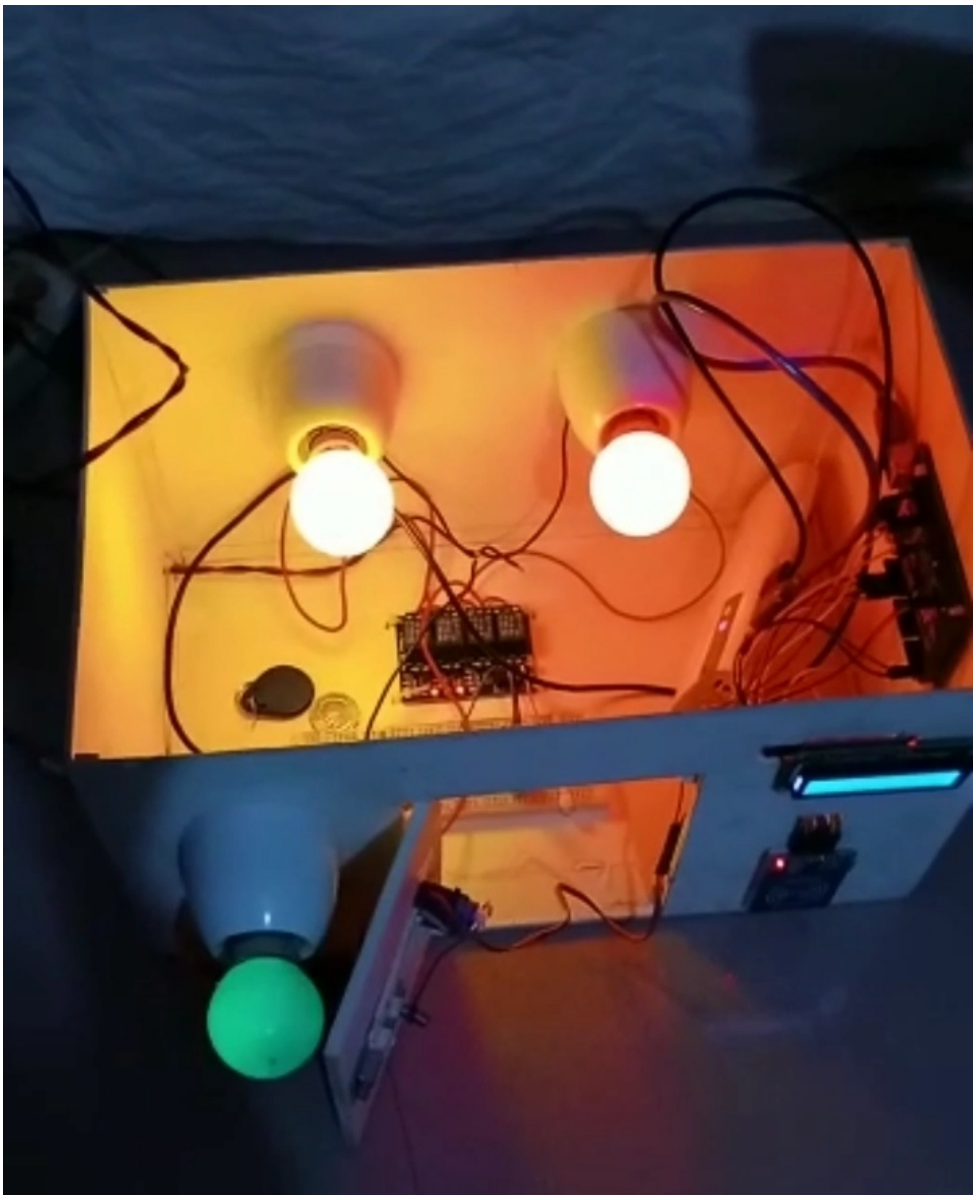

Figure 4.8: RFID Lock System

4.6 summary

In summary, the implementation of an RFID lock system integrated with NodeMCU-based electronic device control offers a comprehensive and sophisticated approach to access control and device management, paving the way for secure, connected, and automated environments. Addressing integration challenges while maximizing the benefits leads to efficient and user-friendly implementations

Chapter 5

Conclusion

5.1 Conclusion

The home automation project has successfully demonstrated the capabilities and benefits of integrating smart technologies into residential environments. By employing a combination of sensors and actuators, the home automation project has enabled the monitoring and control of lighting, temperature, security, and other home systems. This has provided homeowners with the ability to customize their living spaces, create personalized environments, and optimize energy consumption.

Through the implementation of a microcontroller-based system, the RFID lock project has effectively processed and validated the RFID data, enabling seamless control of the lock mechanism. The RFID lock project has proven to be user-friendly, eliminating the need for traditional keys or passwords. Users can simply present their RFID tags/cards to the reader for authentication, offering a convenient and efficient access control method.

The successful implementation of the RFID lock project opens up possibilities for broader applications in various domains, such as home security, office access control, and asset management [7]. Moreover, the home automation project has emphasized the importance of energy efficiency and sustainability.

The project has also highlighted the significance of security and safety in home automation. By integrating security systems, such as cameras, door locks, and intrusion detection, it has provided homeowners with enhanced monitoring and control over their home's security. The ability to receive alerts and remotely manage security devices adds an extra layer of protection and peace of mind.

5.2 Scope For Future Development

The future scope of home automation is vast and holds significant potential for advancements in technology and its integration into our daily lives. The main objective of developing the home automation system is to provide an integrated and computerized platform for homes and workplaces. This system is a basic platform that can be further improved to meet the user's demand.

- Future developments for the Home automation system could include a complete mobile application or website to control or monitor the lock systems and deactivate and activate any cards or tags by the owner.
- Home security could be more powerful by using fingerprint sensors, cameras for face detection, or using voice recognition methods.
- The circuit implementation could be optimized. We could use only Nodemcu or ESP8266 wifi module for both security and electronic device control. By using the wifi module user can control or monitor the lock from anywhere of the world.
- Continued research and development efforts can focus on reducing the overall cost of RFID components and systems.
- The integration of AI and ML technologies with home automation systems can enable more intelligent and adaptive automation. Smart homes can learn user preferences, anticipate their needs, and optimize energy usage based on historical data and patterns.

In conclusion, the home automation project has demonstrated the potential for transforming traditional homes into intelligent and connected spaces. The integration of smart technologies has not only provided convenience and comfort but also offered energy savings, security enhancements, and improved overall quality of life for homeowners. As the field of home automation continues to evolve, there are opportunities for further advancements and integration with emerging technologies, paving the way for even smarter and more efficient homes in the future.

Bibliography

[1] Kumar A. Alshammari, R. A survey of smart home automation systems and technologies. *IEEE Access*, 6:20153–20168, 2018.

[2] A. N. Author. Top 10 iot platforms for developing iot projects.

[3] A. N. Author. Wireless communication networks. 2007.

[4] Candid Wueest Barcena, Mario Ballano. Insecurity in the internet of things. Accessed on March 12, 2015.

[5] Mirko Hohmann Benner, Thorsten. The encryption debate we need. Accessed on May 19, 2016.

[6] Wim Elfrink Chambers, John. The future of cities," foreign affairs. Accessed on October 31, 2014.

[7] K. Finkenzeller. Rfid technology and applications: A review. 98:1576–1597, 2010.

[8] et al. Ghose, T. Voice interaction with home automation systems: A comparative study of amazon echo, google home, and apple homepod. *Proceedings of the 2019 CHI Conference on Human Factors in Computing systems*, pages 1–12, 2019.

[9] Bill Glover. *RFID Essentials*. O'Reilly Media, 2006.

[10] et al Hodo, Elike. *"Threat Analysis of IoT Networks Using Artificial Neural Network Intrusion Detection System.* International Symposium on Networks, Computers and Communications (ISNCC)„ 2016.

[11] Totzauer L. Kühnel, M. Voice assistant integration in smart homes: An empirical investigation of user preferences. *International Journal of Human-Computer Studies*, (145):102523, 2021.

[12] A. Rayes M. Dabbagh. Internet of things security and privacy. *Internet of Things From Hype to Reality*, page 195–223, 2016.

[13] Sanjay Sarma. The internet of things: Roadmap to a connected world. Accessed on March 11, 2016.