A

Project Report On

"HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL DEVICES"

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING

By

M. Ravi Teja Reddy	**19709A0442**
G. Jaya Chandra	**19709A0418**
P. Lakshmi Prasanna	**19709A0426**
V. Vidya Mayuri	**19709A0458**

Under the Esteemed Guidance of

Mr. R. MAHESH KUMAR M. Tech, (Ph. D)

Assistant professor

Submitted to

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES

(AUTONOMOUS)

Rajampet, Annamayya (Dist.), A.P- 516126.

(Approved by AICTE, New Delhi)

(Affiliated to J.N.T.U.A, Anantapuramu)

(Accredited By NAAC & NBA)

2022-2023

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES

(AUTONOMOUS)

Rajampet, Annamayya (Dist.), A.P- 516126.

(Approved by AICTE, New Delhi)

(Affiliated to J.N.T.U.A, Anantapuramu)

(Accredited By NAAC & NBA)

CERTIFICATE

This is to certify that the project entitled **"HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL DEVICES"** is a bonafied record submitted by

M. Ravi Teja Reddy	**19709A0442**
G. Jaya Chandra	**19709A0418**
P. Lakshmi Prasanna	**19709A0426**
V. Vidya Mayuri	**19709A0458**

in partial fulfillment for the award of **BACHELOR OF TECHNOLOGY** in DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING for the year 2022-2023. This record is a bonafied work carried out by them under my Guidance and Supervision.

SUPERVISOR PROJECT COORDINATOR HEAD OF THE DEPARTMENT

Project Viva-Voce held on ___________

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

An endeavour of a long period can be successful only with the advice of many well-wishers. We take this opportunity to express my deep gratitude and appreciation to all those who encouraged me for successful completion of the project.

Our heartful thanks to Guide, **Mr. R. MAHESH KUMAR,** *Assistant Professor,* Department of E.C.E, Annamacharya Institute of Technology and Sciences, Rajampet, for his valuable guidance and suggestions in analyzing and testing throughout the period.

Our special thanks to **Dr. C.H. NAGARAJU**, *Head, Department of Electronics & Communication Engineering,* Annamacharya Institute of Technology and Sciences, Rajampet, for his timely suggestions and help in spite of her busy schedule.

We wish to express our sincere gratitude to **Dr. S.M.V. NARAYANA**, *Principal*, Annamacharya Institute of Technology and Sciences, Rajampet, for their consistent help and encouragement to complete the project.

We are very much thankful to **Dr. C. GANGI REDDY,** *Hon' Secretary* of Annamacharya Educational Trust, Rajampet, for providing good facilities in our college.

We wish to convey our gratitude and express our sincere thanks to all Project Review Committee members for their support and co-operation rendered for successful completion of the project.

We also extend our gratitude to our parents, staff and friends for their moral support and their encouragement, which motivated towards successful completion of the project.

M. Ravi Teja Reddy	**(19709A0442)**
G. Jaya Chandra	**(19709A0418)**
P. Lakshmi Prasanna	**(19709A0426)**
V. Vidya Mayuri	**(19709A0458)**

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL DEVICES

ABSTRACT

Everyone is familiar with the usage of internet these days. It may raise the standard of living and make life more comfortable. Controlling every electrical appliance in our house with a single touch over the internet has become necessary. The internet not only grants us independence but also total relaxation with the ability to operate our household appliances from any location at any time. The internet might produce issues from time to time. A bad internet connection will cause the command time to be delayed and may even result in dangerous situations. Cyber-attacks may impact your internet-connected smart home, exposing all the safeguarded data and your private information. These are relay modules with a NodeMCU (ESP8266). NodeMCU is mostly used as a low-cost open source, which contains firmware to run on the Wi-Fi-based mode that is interfaced with JavaScript software that uses the HTTP request mechanism to operate the microcontroller. The major objective of employing relays is to operate the electrical switches remotely from the server, which would be interconnected with electrical devices to work in a simple manner. In this project, Zigbee or Z-wave protocols are used to operate all home gadgets without the need for constant internet access.

Keywords: NodeMCU (ESP8266), Wi-Fi, Relay Modules etc.

CHAPTER -1

INTRODUCTION

1.1 EMBEDDED SYSTEMS:

Embedded systems are customized PCs that are built into goods, equipment, or gadgets to carry out certain functions. They are not meant to be used as stand-alone systems, unlike general-purpose computers, and are created to accomplish a specific set of tasks. Cell phones, household appliances, cars, hospital instruments, and industrial automation systems are just a few examples of the many goods and gadgets that have embedded systems. They are made to work in real-time settings where prompt and precise answers are necessary. They are small, effective, and dependable.

The central processing unit (CPU) of embedded systems is often a microprocessor or microcontroller with constrained storage and input/output (I/O) features. The system's other parts, such as actuators and sensors, are under the CPU's supervision. It is also in charge of carrying out instructions. Additionally, real-time operating systems and other specialized software developed specifically for embedded systems' unique needs are frequently used in them. These operating systems are made to be small, effective, and offer real-time scheduling and synchronization capabilities that are crucial for many embedded applications. Finding a compromise between price, performance, and energy consumption while creating embedded systems is one of the biggest issues. Minimizing energy usage is crucial because embedded systems are frequently needed to function in areas with scarce resources and restricted power.

An embedded system consists of a combination of mechanical and electronic components as well as computer hardware and software or the system's other parts, such as actuators and sensors, are under the CPU's supervision. It is also in charge of carrying out instructions. Additionally, real-time operating systems and other specialized software developed specifically for embedded systems' unique needs are frequently used in them. These operating systems are made to be small, effective, and offer real-time scheduling and synchronization capabilities that are crucial for many embedded applications. Finding a compromise between price, performance, and energy consumption while creating embedded systems is one of the biggest issues. Minimizing energy usage is crucial because embedded systems are frequently needed to function in areas with scarce resources and restricted power. An embedded system consists of a combination of mechanical and electronic components as well as computer hardware and software. Eight 16-bit controllers, a basic operating system, and hardware designs specific to the job at hand are commonly used in lower-end embedded systems.

The following components are found in embedded systems, which are a type of electronics that combine both hardware and software:

- Processor architectures or embedded systems
- Graphics cards (GPUs): portable and non-storage
- Data multimedia messaging system (mms, ports, networks, and application program) Power sources
- Power sources

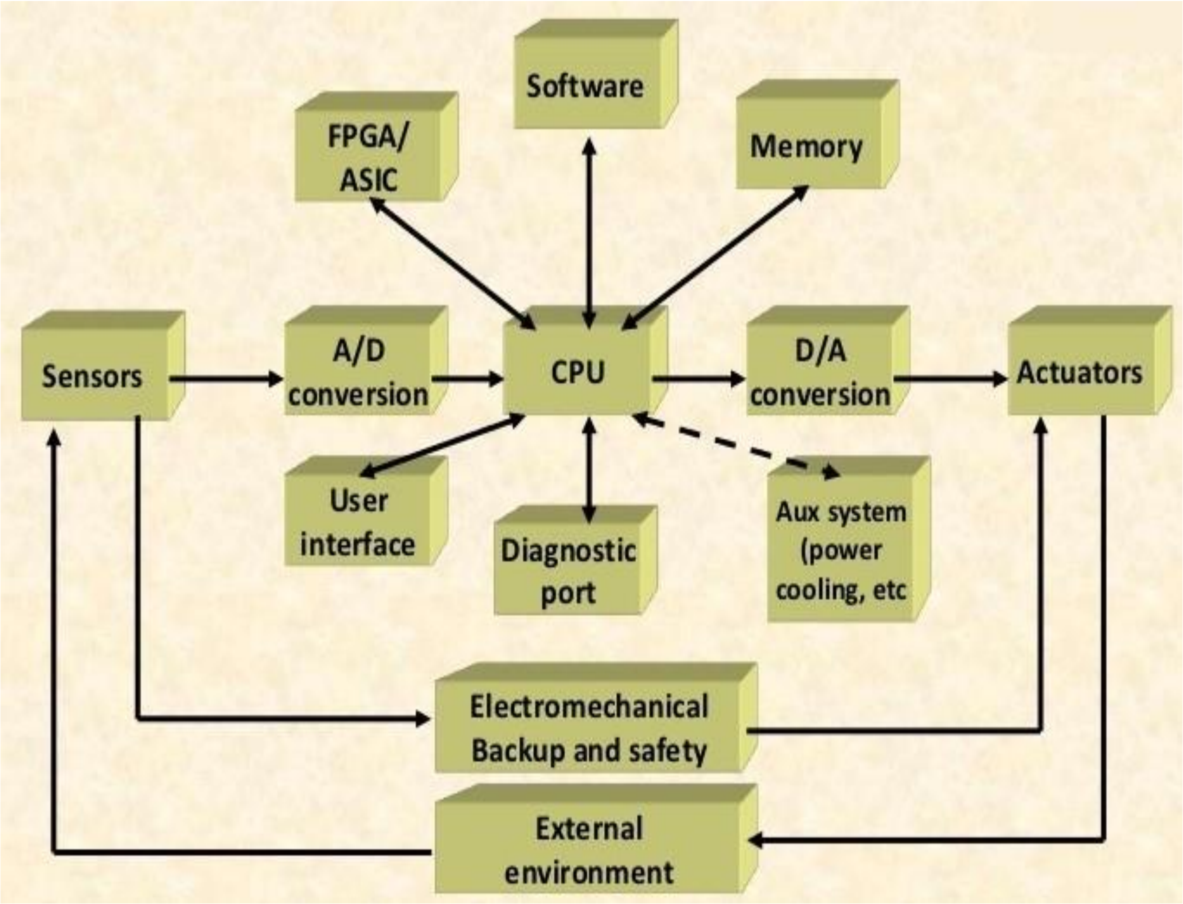
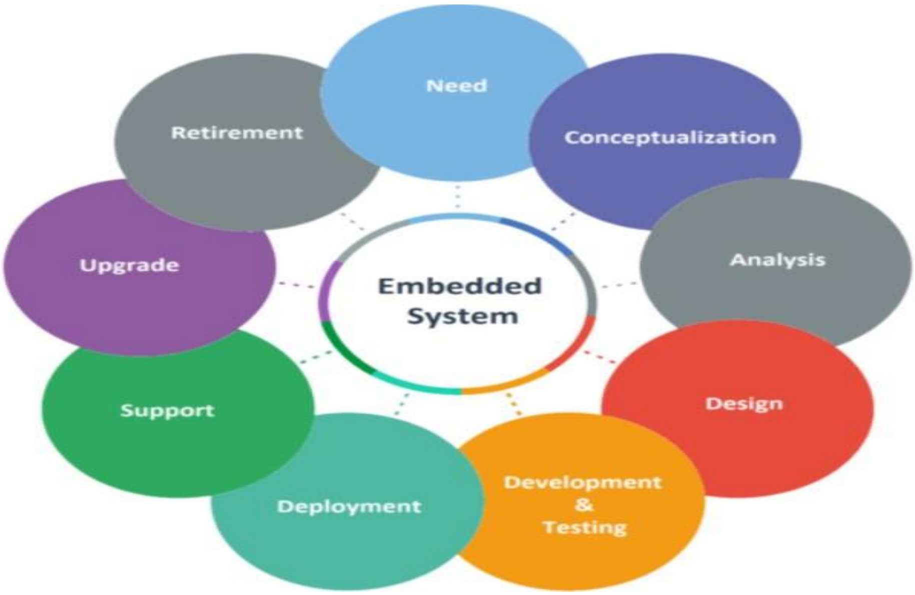


Figure 1.1: The Fundamental Embedded System

1.1.1 EMBEDDED SYSTEM DESIGN CYCLE:

A methodical procedure for developing embedded systems is the embedded device design cycle. The following actions are required:

- Requirements Analysis: Analysing the requirements the criteria for the embedded device must be determined at the initial stage of the design cycle. Defining the performance and functionality requirements as well as any limitations in terms of size, weight, power usage, and cost are all included in this. Due to breakthrough technological advances, the product is deemed outdated and removed from the market. A vibrant and cutthroat marketplace exists. To introduce a new version of a product, it is, therefore, necessary to make modifications to the design of the current one.
- System Design: The next stage is to create a high-level design of the system based on the requirements. This entails establishing the system architecture as well as the hardware and software modules that will be employed.
- Hardware Design: Hardware design is the process of choosing and putting into use a system's actual physical parts, such as microcontrollers, sensors, and actuators. It also covers the interconnections between the components and the design of the printed circuit boards (PCBs).
- Software Design: Software design is the process of putting software modules into practice and defining the data and communication interfaces between software and hardware parts. This covers the creation of the application software, real-time operating system, and device drivers.
- Prototype and Testing: After the software and hardware designs are finished, a prototype of the system is constructed for testing. This entails confirming that the system complies with the criteria and that it performs as planned.
- Integration and Verification: The integration of all the system's components and thorough testing to ensure that it functions properly constitute the design cycle's last stage. As part of this, the hardware, software, and data connections between them are all tested.
- Deployment: The embedded system is placed in the specific environment and put into operation following successful integration and testing. To make sure that the system performs as anticipated in a real-world setting, extra testing and verification may be required. It is normal to repeat any or all of the processes in the embedded system design cycle several times to increase the system's performance and fine-tune the design.



1.1.1: Embedded System Design Cycle

1.1.2 CHARACTERISTICS OF EMBEDDED SYSTEM:

An embedded system is any computer that is integrated into a non-computer product. Along with the problems we encounter while designing apps, we will encounter a variety of difficulties when creating embedded system software. Throughput: Our system may need to handle a lot of data quickly. Response: In certain situations, our system may need to react swiftly to occurrences. Testability: configuring tools to test embedded software can be difficult. Debuggability: It is hard to figure out what the software is doing improperly without a screen or a keyboard (other than not working). Reliability: Embedded systems must have the capacity to handle any condition without the assistance of a person.

For their design and operation, embedded systems properties are crucial and include the following:

- Real-time Operation: Many embedded systems must function in real-time settings when prompt and precise answers are necessary. This necessitates the system's capacity to process data and react to inputs in a predetermined amount of time.
- Limited Resources: Embedded systems are frequently made for locations with a limited amount of resources, where size, weight, power consumption, and cost are crucial considerations. This necessitates a small, effective system with constrained memory and processing resources.

- Specialized Functions Embedded systems are not meant to be general-purpose computers; instead, they are designed to carry out a specific set of specialized tasks. This necessitates that the system is created and tailored to its unique duties and needs.

- Interaction with the Physical World: Many embedded systems interface with the physical environment via sensors and actuators, and they must carry out real-time control and monitoring tasks. To accomplish this, the system must include physical environment interfaces as well as real-time change response capabilities.

- Reliability: Embedded systems are frequently employed in important applications, such as medical equipment, industrial control systems, and vehicles, where reliability is crucial. To achieve this, the system must be built to be extremely dependable and to run steadily and consistently.

- Security: As embedded systems grow more interconnected and are incorporated into the Internet of Things (IoT), security becomes a more crucial factor to consider. As a result, the system must be designed with security in mind to prevent unauthorized use.

- Ease of Use: A lot of embedded systems are used by non-technical users who may lack the knowledge or skills necessary to configure or program the system. This necessitates the system's design is user-friendly and simple to use, with clear documentation and intuitive interfaces.

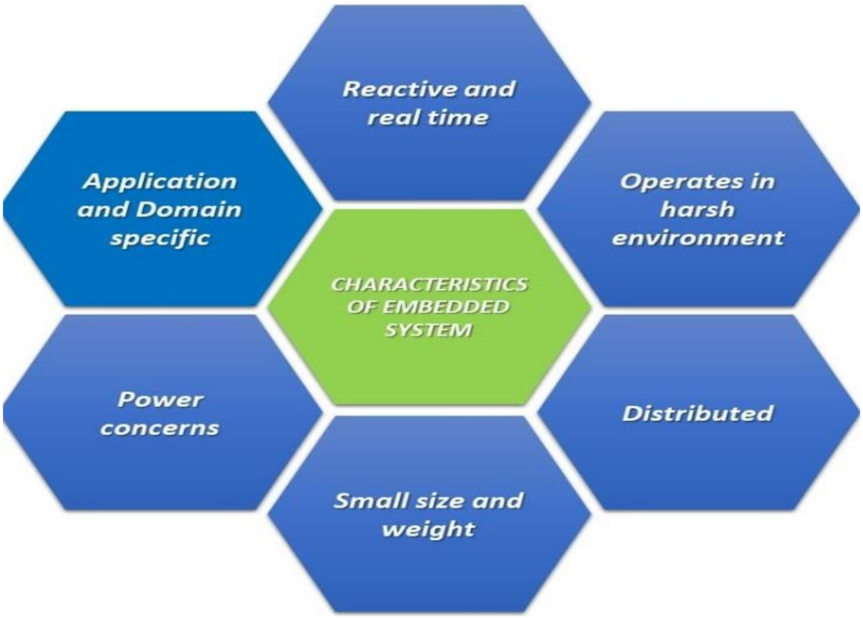

Reactive and real time
Application and Domain specific
Operates in harsh environment
CHARACTERISTICS OF EMBEDDED SYSTEM
Power concerns
Distributed
Small size and weight

Figure 1.1.2: Characteristics of Embedded Systems

1.1.3 APPLICATIONS:

Due to their capacity for specialized tasks to be carried out and real-time process control, embedded systems are widely employed in several applications. The following are some typical embedded system applications:

- Automotive systems: Automotive systems: Embedded systems are used in cars for features including engine management, traction control, anti-lock brakes, entertainment systems, and navigation systems.
- Consumer electronics: Products like cell phones, tablets, TVs, and gaming consoles all frequently employ embedded systems.
- Medical devices: Medical gadgets such as blood glucose meters, infusion pumps, and heart monitors all utilize embedded systems.
- Industrial automation: PLCs (programmable logic controllers) are embedded systems that are used in industrial automation systems to control and monitor industrial operations.
- Home appliances: Embedded systems are used in household appliances like air conditioners, refrigerators, and washing machines to monitor and regulate their operations.
- Security systems: Embedded systems are used in security systems such as burglar alarms, fire alarms, and surveillance cameras to monitor and react to events in real-time.
- Aerospace and defence: Applications for embedded systems include unmanned aerial vehicles, missile guidance systems, and aircraft control systems (UAVs).
- Telecommunications: Telecommunications systems such as mobile phones, routers, and switches to regulate and maintain communication networks.

Due to their capacity to carry out specified functions in real-time and interact with the actual environment, embedded systems have a wide range of applications.

1.1.4 CATEGORIES OF EMBEDDED DEVICES:

Three major categories may be used to classify embedded devices:

- Stand-alone Embedded Devices: These are standalone devices that carry out a certain task without the need for a link to other hardware or software. Digital cameras, digital calculators, and portable gaming consoles are a few examples of stand-alone embedded electronics.

- Networked Embedded Devices: Networked embedded devices interact with other systems or devices via communication in order to carry out their tasks. Smartphones, routers, and security cameras are a few examples of networked embedded devices.
- Complex Embedded Systems: Complex embedded systems are intricate systems made up of several interconnected embedded devices that cooperate to carry out a certain task. Automobiles, hospital equipment, and industrial control systems are a few examples of complicated embedded systems.

Each of these categories of embedded devices has its own unique set of requirements and design considerations, and the choice of category will depend on the specific requirements and application of the system. Additionally, many embedded devices may belong to multiple categories, as they may be both standalone and networked or may be part of a complex system.

1.1.5 UNIQUE EMBEDDED SYSTEMS:

Embedded systems that are considered "unique" are those that differ from other systems due to their functions and requirements. Autonomous automobiles are one instance of an embedded system that is special. Some examples of unique embedded systems include:

- Autonomous vehicles: Advanced embedded systems are used to guide and control the motions of autonomous vehicles such as self-driving automobiles, drones, and other types of vehicles.
- Wearable devices: To carry out their duties and interface with the user, wearable gadgets like smartwatches, fitness trackers, and augmented reality glasses need specific embedded systems.
- Smart home systems: Smart home systems employ embedded technologies to monitor and manage many aspects of the home, such as smart security and home automation systems.
- Robotics: To guide their motions and carry out their tasks, robots, including industrial and humanoid robots, require embedded systems.
- Aerospace and defence systems: Aerospace and defence systems, such as aircraft control systems, missile guidance systems, and satellite communication systems, use specialized embedded systems to perform their critical functions.

These unique embedded systems have specific requirements and design considerations, and often involve the integration of multiple subsystems and technologies to achieve their intended functions. The design and development of these systems requires a high level of expertise and specialized knowledge to ensure their reliability and performance.

1.1.6 REAL TIME SYSTEMS:

An embedded system called a "real-time system" is created to react to events and run under severe time limitations. The real-world events that the system must process and the outcomes that must be delivered within a certain time limit dictate how the system will behaves as Real-time systems are used in many applications where immediate response to events is critical, such as in control systems, medical systems, and defence systems. Examples of real-time systems include:

- Traffic control systems: Real-time systems are used in traffic control systems to regulate the flow of traffic and respond to incidents like accidents or congestion in real-time.
- Industrial control systems: Real-time systems are used in industrial control systems to monitor and manage production processes as well as to respond to events.
- Avionics systems: Real-time systems are used in avionics systems to control and monitor aircraft systems and respond to events in real-time.
- Medical systems: Real-time systems are used in medical systems such as heart monitors, patient monitors, and infusion pumps to respond to critical events in real-time.

The design of real-time systems requires a high level of expertise and specialized knowledge, as the systems must be designed to operate within strict time constraints and to respond to events in real-time. This often involves the use of specialized algorithms, data structures, and programming techniques to ensure that the system operates within its specified time constraints.

1.1.7 NETWORKED INFORMATION APPLIANCES:

Embedded systems that are connected to other devices or systems through a network are referred to as networked information appliances, sometimes known as networked embedded systems. The network connection is used by these devices to connect to the network, access information and services, and communicate with other devices and systems. Examples of networked information appliances include:

- Smartphones: Smartphones are a prime example of a networked information appliance since they are portable devices that connect to a network to access the internet and communicate with other hardware.
- Tablets: Tablets are mobile computer devices that connect to a network to access the internet and connect to other hardware.
- Smart home devices: Smart home appliances, such as smart lighting, and thermostats, link to a network to interact with other appliances and systems in the house.
- Internet of Things (IoT) devices: IoT devices, such as smart sensors and smart appliances, use a network connection to communicate with other devices and systems and to access information and services from the network.

The design and development of networked information appliances requires a high level of expertise in networking and embedded systems, as these systems must be designed to operate efficiently within a networked environment and to communicate with other devices and systems. Additionally, security and privacy considerations must be considered, as these systems may be vulnerable to hacking and other security threats.

1.1.8 MOBILE DEVICES:

The embedded systems used in mobile devices are built to be portable and mobile. The majority of these gadgets feature a small size, a prolonged battery life, and wireless network connectivity. The design and development of mobile devices necessitate a high degree of proficiency in both embedded systems and mobile computing since these systems must be built to function well in a mobile context and to offer a user-friendly experience. As mobile devices are frequently used for prolonged periods without a power source, power consumption, and battery life must be taken into consideration when designing mobile devices.

Examples of mobile devices include:

- Laptops: When it comes to computing power and functionality, laptops are small, portable computers that deliver desktop-level performance.

- Wearables: Wearables are small, wearable devices that offer a range of features and capabilities, such as fitness tracking, smartwatches, and augmented reality glasses. PDAs, but at the other hand, can now run general-purpose development tools like word processing programs and games.

1.1.9 CPU PLATFORM:

A CPU platform, commonly referred to as a "system-on-a-chip" (SoC), is an integrated circuit that integrates a central processing unit (CPU), memory, and other components onto a single chip. Many embedded systems employ a CPU platform as their primary processor, which gives the system its computational capability and control. There are many kinds of CPU platforms, from low-power microcontrollers for straightforward embedded systems to high-performance CPUs for more demanding applications. An embedded system's needs, such as processing power, memory capacity, power consumption, and price, determine the CPU platform that should be used.

Examples of CPU platforms used in embedded systems include:

- ARM processors: ARM processors are popular in embedded systems because of their low power consumption, strong performance, and broad support from chip suppliers.
- Intel Atom processors: These processors are used in embedded systems that need a lot of performance and computing power, such as networking gear and mobile devices.
- MIPS processors: MIPS processors are employed in embedded systems, including networking and consumer electronics, that need great performance while consuming little energy.

A high degree of skill in both hardware and software design is required for the design and development of a CPU platform for an embedded system since the platform must be created to satisfy the unique needs of the embedded system. Additionally, the choice of a CPU platform may also impact the choice of other components and subsystems in the system, as well as the development of the software for the system.

1.1.10 PERIPHERALS:

Peripherals are parts or things that are attached to a central processing unit (CPU) and provide the system with further features or abilities. It is common to practise in embedded systems to communicate with the outside world through peripherals, such as input devices like keyboards and touchscreens or output devices like displays and speakers.

Examples of peripherals used in embedded systems include:

- Input devices: Keyboards, touchscreens, buttons, and sensors are examples of input devices that give users a means to interact with the system.

- Output devices: Output devices give the system a method to give the user information or feedback and include screens, speakers, and LEDs.
- Storage devices: Solid-state drives (SSDs) and microSD cards are examples of storage devices that provide the system access to data.
- Communication devices: The system may connect with other systems and devices thanks to communication devices like Wi-Fi and Bluetooth modules.

Because the peripherals must be designed to integrate smoothly with the rest of the system and deliver reliable and effective performance, designing and developing them for an embedded system requires a high degree of knowledge in both hardware and software design. The selection of peripherals may also have an influence on the selection of the system's other parts and subsystems, as well as the creation of the system's software.

1.2 HOME AUTOMATION:

Home automation is a system that uses technology to provide homeowners with remote control over a variety of features in their house, including lighting, temperature, and appliances, from a central location or via a smartphone or other device. A home automation system consists of several interconnected components that may be managed by a single platform or device, including sensors, actuators, and controllers. In addition to increasing security and control, home automation aims to boost homeowners' comfort, convenience, and energy efficiency.

Examples of home automation systems include:

- Smart lighting systems: These systems, which can have functions like automated dimming and colour shifting, which let homeowners manage the lighting in their homes remotely or from a central location.
- Smart thermostats: Smart thermostats are devices that enable users to regulate the temperature in their homes remotely or from a central location. They may also include capabilities that automatically alter the temperature following occupancy or schedule.
- Smart appliances: Appliances like washing machines, refrigerators, and ovens are examples of smart appliances since they can be operated and seen remotely.

As these systems must be designed to run seamlessly, and securely, and give a user-friendly experience inside the home setting, the design and development of home automation systems demand a high degree of knowledge in both hardware and software design. Furthermore, scalable design is essential for home automation systems so that they may be readily expanded and upgraded as required. Automation of home features, activities, and appliances is referred to as "home automation."

To put it simply, it means you can easily control the appliances and amenities in your house over the Internet to make life easier and sense of security and even lower household bill spending. A network of hardware, communication, and electrical interfaces called home automation connects commonplace devices through the Internet.

Figure 1.2: Home Automation

1.3 HOME AUTOMATION WITHOUT INTERNET:

Systems for home automation can be made to function without an internet connection as well. These devices are frequently referred to as "local" or "stand-alone" home automation systems. All of the parts and gadgets in a stand-alone home automation system are linked to one another directly and don't need an internet connection to function. As a result of removing the possibility of hacking or other security breaches that may happen through a linked system, this can offer a more safe and more dependable solution.

Examples of standalone home automation systems include:

- Wireless lighting control systems: These systems let users manage their home's lighting without an internet connection by using a wireless remote or switch.
- Standalone thermostats: With these devices, house owners may manage the, without the need for an internet connection.
- Standalone home security systems: Systems that don't require an internet connection, such as standalone home security systems, offer security and surveillance features.

Because standalone home automation systems must be built to run smoothly, safely, and with the utmost user-friendliness within the home environment, they demand a high degree of hardware and software design knowledge to design and create. Furthermore, independent home automation systems must be created to be scalable and upgradeable, allowing for future expansion and modernization as necessary. When people routinely use your home's smart gadgets via the internet, they are not only in danger but also very much at risk of being hacked. Zigbee and Z-wave are examples of technologies and protocols that our science of robotics has created as a result of all your devices to control home automation devices without internet at all time.

Figure 1.3: Home Automation Without Internet

1.4 Z-WAVE A LIFESAVER IN AN INTERNET OUTAGE:

Systems for home automation frequently employ the wireless communication technology Z-Wave. Even in the case of a disruption in internet service, it is intended to offer dependable and secure communication between various devices and parts of a home automation system. These days, they are among the top wireless protocols. In terms of communication with more devices and at greater ranges, Zigbee has a sizable advantage over the Z-wave protocol. Indoor devices that are 150 feet away can still communicate and connect using the Zigbee protocol. Despite being long-distance connections, they function without the assistance of the internet. According to a preset, Zigbee operates between 2.4 GHz and 915 MHz. On the other hand, Z-wave technology can only operate at 908.42 MHz, so to choose a smart gadget for the home hub, ensure the compatibility of protocol whether Zigbee or Z-wave.

Figure 1.4: Zigbee / Z-wave a lifesaver in an Internet outage

CHAPTER -2

LITERATURE SURVEY

2.1 LITERATURE SURVEY:

Mandela et al. [1] developed a system in 2015 that may be used to regulate the data from sensors like light, gas, movement, and temperature and facilitate a process based on requirements, such as switching on the lights whenever it gets dark. You should also use Gmail to promptly save data. The fundamental concept of using computer technology to regulate household appliances was given by Pythonlike [2] in 2016. It conserves energy, offers security, and allows us to access home gadgets via websites from locations with Wi-Fi, even while we are miles from home.

The system designed by Singh et al. [3] in 2017 can carry out a variety of tasks that may be done at home. From everywhere on the globe, this enables connectivity. The project's goal was to use as little energy and labour as possible. The home automation system integrates a variety of technological elements, including cloud-based communication and wireless networking. The cloud-stored data was subsequently examined. Depending on their preferences, the user can utilise the internet to access various home gadgets. This method is inexpensive. This system has several household appliances controls. Kodaly et al. announced in 2016 [4] a project aimed at developing a wireless surveillance system with alarm notification sent to the house owner via the Internet whenever any action occurred and the alarm increasing optionally. Additionally, the sensors may be used with a smart home system. Additionally, the sensors may be used with a smart home system. The system's advantage over competing systems of the same kind is that it sends status updates and alerts through Wi-Fi and is microcontroller controlled. The owner may get these notifications on his mobile device from anywhere in the globe, regardless of whether his smart device is online or not. A smart home control system that can connect with an Android mobile device was developed in 2017 by Sharma et al. Wi-Fi enables mobile devices and systems to exchange data. Any device that is compatible with the proposed system can connect to it by downloading a mobile application. The instructions for turning electrical accessories like fans, lights, and air conditioners on and off. Setting a timer at home so that it may be sent quickly and simply from a mobile device using a graphical user interface that is easy to use and comfortable for most users The system for home automation reacts to user commands by performing the requested activities and reporting the outcome. Additionally, the user may view command results on an Android-based mobile application when they are within Wi-Fi range.

A real-time home automation system idea was designed and realized by Soliman et al. in 2017 [6] using the LabVIEW platform and the microcontroller "Arduino" board. The suggested smart home system includes two pieces of hardware: an Arduino microcontroller board and a laptop that serves as a local server and manages the LabVIEW platform. Attached to the Arduino PIC microcontroller are sensors and household appliances. Homeowner-given commands automatically operate, manage, and link all the equipment in the house. The suggested home automation system has advanced in terms of design and flowchart. A hardware implementation for three functional household appliances, including an assessment of the proposed system's effectiveness and dependability: Depending on an ultrasonic range-detecting sensor, security cameras, lighting, energy conservation, temperature control, and other features have been created. The system is suggested to be versatile, simple to use, and low-cost, making it appropriate for the future of smart homes. A cost-effective, dependable, and adaptable smart system for home automation with added security is presented by Sat apathy et al. in 2018 [7] using an Arduino microcontroller board. Local Wi-Fi with an established connection enables system users to utilize programmes on their smartphones to monitor, manage, and access home appliances with permission. The suggested method to monitor household appliances like industrial machines and consumer products depends on an independent server and the Internet of Things (IoT). an infrared remote module, a smart smartphone app, or an internet browser. In 2018, Hossain et al. [8] proposed a smart connected home system that facilitates the segregation of items connected via the use of motion sensors, fog computing, servers, and switches. The household appliances are managed by this system using a personal computer. using the motion sensor to detect and control home appliances. The electronics are instantly turned on or off when the sensor senses motion. The developed intelligent home automation system can monitor the users' household items and manage the following functions: fan ON/OFF, light ON/OFF, door ON/OFF, and webcam/CC-camera ON/OFF. On or off the window, the fire alarm, and the pump. In 2018, Kausalya et al. [9] presented a prototype called a "smart home automation system" that focused on controlling fans and lights, with extra security offered by taking a picture if necessary.

A wireless sensing node was considered by Singh et al. in 2018 to improve home automation. A home automation system combined a variety of electrical equipment and controlled it with little to no user involvement. The home automation system maintains track of all the environmental factors that are present and directs the home appliances to function in accordance with the system user's requirements. The system is used to automatically book the gas cylinder when the gas level falls below the threshold, as well as to automate home appliances, provide notifications about the cost of the homeowner's power bill at regular intervals, and automate the notification of household appliances. Hoque and Davidson [11] suggested a design for a smart door system in 2019, and it is a versatile and affordable system with a mobile Android application. The Raspberry Pi 2 board and the Eligo Mega 2560, an Arduino microcontroller board comparable, make up the system's architecture. Together, they link to a web application that implements a RESTful API. An effective smart system for home automation that can be used to access, operate, and monitor household appliances from any location on the globe was created in 2019 by Vishwakarma et al. [12]. Connectivity for the primary internet supply is provided through the intelligent home automation system. Wireless access relies on an IP static address. Using speech recognition and a browser application like Google Assistant, a multimedia application built on a home automation system may be controlled by the owner. Kulkarni et al. [13] planned to build a smart, automated home in 2017 that would employ an intelligent phone system to monitor house appliances. It depends on the Internet of Things (IoT) to operate this intelligent home automation system. When a new technology, such as the Internet of Things (IoT), is incorporated into this system, the field becomes extremely intriguing. A vast variety of devices are supported by the Raspberry Pi minicomputer, allowing for simultaneous control and monitoring of several household appliances. On the Raspberry Pi, a local server is built. Smart devices, such as a laptop or smartphones, are necessary for system users to control home appliances from anywhere in the globe thanks to the User Interface (UI) developed on websites.

A house automation system powered by Java was created by [14]. All the home automation components were physically connected via an integrated board, which also allowed for remote system access via connection with a PC-based web server. Java technology is used to create a safe solution since it has built-in network security capabilities. The system needs a high-end PC, which is both obtrusive and expensive to install. [15] unveiled a Bluetooth-based remote monitoring system that consists of a main controller and several Bluetooth sub-controllers. Every household appliance is physically connected to a nearby Bluetooth sub-controller. Wired connections are used between the home appliances and their respective sub-controllers. All connections from the sub-controller are transmitted through wireless means to the main controller. A specialized Bluetooth module should be included in every household appliance. One module, however, is shared by several devices due to the financial burden of Bluetooth technology. With wireless technology, this design decreases the amount of conventional cable necessary and, consequently, the installation's intrusiveness. Because certain wired communications are included, the architecture somewhat reduces the installation's inconvenience. In addition, there is the drawback of access latency when a single Bluetooth is shared by several devices.

2.2 IOT BASED HOME AUTOMATION:

A home's numerous systems and operations can be automated using IoT (Internet of Things)-based gadgets that are connected to the internet and sensors. Using a cell phone or other equipment with internet connectivity, these devices may be remotely monitored and controlled.

Devices for home automation based on the Internet of Things include:

- Remotely programmable smart temperature meters that allow you to change the home's thermostat.
- Remotely controllable smart lighting solutions that allow you to program schedules or switch the lights on and off remotely.
- Utilizing cameras and sensors, smart security systems allow for remote monitoring.
- Smart appliances, including ones that may be remotely started and stopped, include refrigerators, household appliances, and ovens.
- streaming music, movies, and other media through smart home entertainment devices that may be operated from a distance.

Generally, IoT-based home automation enables homeowners to wirelessly manage, monitor, and automate tasks, making their homes more convenient and effective.

2.3 OBJECTIVE OF THE PROJECT:

The recommended method overcomes the challenge of wiring in the case of wired automation. It is also possible to have a large power supply. In comparison to Bluetooth, the operating range is wider. The existing approach does not allow for remote control and monitoring of appliances. The recommended solution, on the other hand, makes use of a Wi-Fi-based home automation system that enables monitoring and administration of the appliances. In contrast to the 1990s' current home automation system, which relies on people operating electronic equipment manually, our proposed technology would allow us to remotely control any electronic appliance. IoT applications have proliferated in the twenty-first century as a result of the dominance of the internet, the development of smartphone technology, and the rising standard of communication devices.

The primary goal of house automation without the internet is to automate various systems and services in a home using wired or wireless technologies without relying on online access. As a result, even without an internet connection, homeowners may manage and keep an eye on several components of their house, including lighting, temperature, safety, and appliances.

Some of the benefits of home automation without internet include:

- Increased security: These systems may still offer security features like movement sensors and cameras even if the web connection fails.
- Greater control: With the use of wired or wireless technology, homeowners may remotely manage and keep an eye on several features of their house even when there is no internet connection.
- Energy efficiency: Automating temperature and lighting settings can help cut down on energy use and costs, even without internet connectivity.
- Convenience: Home automation devices that don't require an internet connection can simplify daily activities like shutting off appliances and lights when they aren't in use.
- Enhanced comfort: By controlling the house's lighting, temperature control, and other features, systems for home automation without the internet can offer improved comfort.
- Smart appliances include refrigerators, washers and dryers, and ovens that can be started and stopped remotely.
- This project's goal is to deliver a home appliance system that allows users to operate and manage electrical appliances, including lights, fans, cameras, and sensors, from their own homes. Zigbee or Z-wave protocols, which can operate all home gadgets without an internet connection at any time, are included in the system to provide these features.

- The technology enables the user to directly command devices, such as electrical gadgets, to be turned on or off. This project's goal is to create a product that can be easily controlled at home, with an emphasis on low cost and open software configurability.

The goal of remote monitoring without the internet is to give owners more control, security, comfort, and convenience in their homes, even when there is no internet connection available.

2.4 EXISTING METHOD:

A home automation device may be used by customers to operate a range of electrical appliances. A wired connection is used by several popular home automation systems. Unless such a system is created beforehand and implemented during the building of the structure, this will not be a problem. The Internet of Things (IoT) is a method that employs computers or portable devices to leverage the Internet to automate basic home features and operations from anywhere in the world. A common choice for home automation systems is communication that is based on the Internet, or IP protocol.

Interoperability is the capacity of a system or product to interact with other systems or goods in a uniform way. The current system's drawback is that it does not give the user a graphical interface (GUI), forcing them to memorize all the AT instructions to operate the linked devices. Additionally, the system uses Java-based functionalities. These days, few people are using those mobile gadgets. The user does not need to memorize any commands since, in the suggested scheme, all devices are managed by an Android smartphone and a web server. Freezers, fans, lights, and motors are a few examples of automated equipment.

A home automation system that makes use of IoT technology is now required to be operated via a cell phone and the internet to monitor and control electrical and electronic devices at the home from any location. Thanks to the concept of "smart houses," our living spaces are increasingly dynamic and user-responsive. Home automation makes it easier to operate all the electrical appliances in your house and offers a personalized way of living depending on your daily routine. The term "house automation" refers to a wide range of functions, including the simple management of your lamps, home security, and sliding doors. It could also provide you instant access to a cup of coffee anytime you need one and to practically anything, you come across during your daily activities.

There are several current home automation techniques, including:

- Wired systems: These systems link household appliances and control systems through wires. Although they might be more expensive and complex to install than wireless systems, wired systems are often more dependable and secure.
- Wireless systems: These integrated operational devices and systems regulate systems within the home using wireless technology like Wi-Fi, Bluetooth, or Zigbee. Although they might be less dependable and secure than wired systems, they are typically simpler to deploy.
- Smartphone or tablet control Many home automation systems can be controlled using a smartphone, enabling homeowners to manage and keep an eye on many parts of their house from a distance.
- Voice control: Many home automation devices can now be operated via voice commands because of the integration of chatbots like Amazon's Alexa, Android Auto, and Apple's Siri.
- Standalone devices, including remote controllers, touchscreens, and keypads, can also be used to operate several home automation systems.
- Infrared Control: Some home automation gadgets and devices can also be controlled by infrared remote controllers. Depending on the individual demands and tastes of the homeowner, each of these techniques has pros and cons of its own. It is always a good idea to consider your alternatives and speak with professionals before selecting the home automation system that is best for your residence.

Figure 2.4: Existing System Result

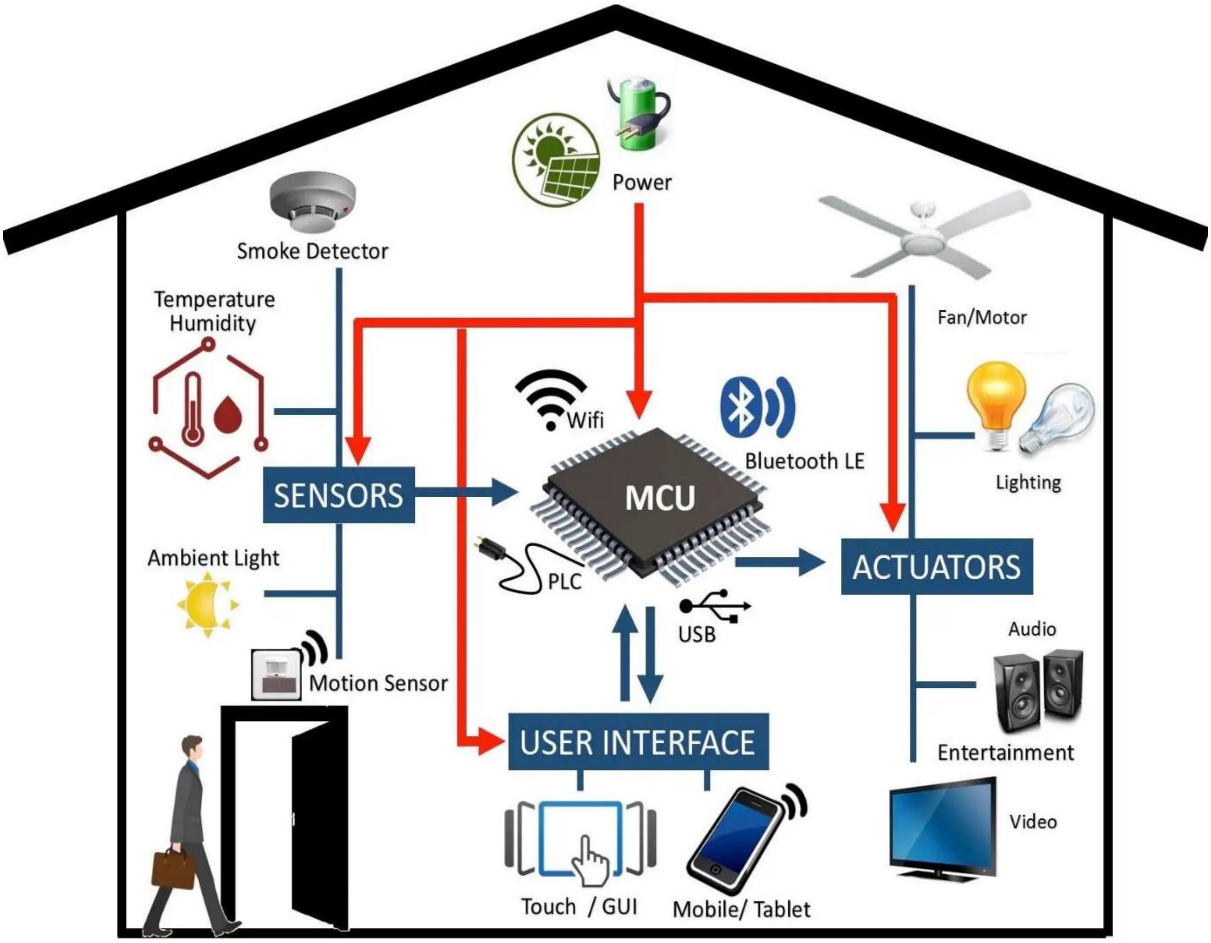


Figure 2.4: Home automation systems may be controlled remotely using a variety of sensor devices, microcontrollers, NodeMCU, and other technologies on an open platform with the help of Internet.

2.5 MOTIVATION:

A system like this was created to streamline and improve the complexity and busyness of our everyday lives. allowing for the management and monitoring of different components of the house even without an internet connection. This might be advantageous for several reasons, including:

- Reliability: Access to the internet can occasionally be sporadic, particularly in isolated or rural regions, during power outages, or in the event of a natural disaster.
- Security: Regardless of whether the internet is not working, home automation can still operate and monitor the house. Home automation devices dependent on the internet

might be subject to hackers, so some homeowners could be worried about their security. Since it does not rely on the internet, home automation without it can be a safer solution.

- Privacy: Some homeowners may be concerned about privacy when it comes to internet-based home automation systems, as they can collect and transmit data about the home and its inhabitants. Home automation without internet can provide a more private option, as it does not rely on internet connectivity.
- Cost: Because smart home devices do not require an internet connection and may use less expensive technology, they may be less expensive than systems that do.
- Simplicity: Due to its potential for simplicity in setup and operation, some homeowners may choose a home control system that is less complicated and does not require internet access.

The primary goal of home automation without the internet is to provide monitoring and control of the house even when there is not an online connection, while providing reliability, security, privacy, cost-effectiveness, and simplicity.

2.6 REVIEW OF FOREIGN STUDY:

Recent years have seen a rise in interest in the field of home automation via the Internet of Things. Numerous studies have been done on the user experience, security, energy efficiency, design, and implementation of IoT-based home automation.

Several ongoing initiatives make use of the IoT to offer affordable and real-time smart home automation solutions. An Internet of Things (IoT)-based actual home automation and surveillance system that relies on the Arduino Board, ESP8266 Wi-Fi chip, and the Arduino MQTT server was presented in 2017 by Kishore P et al. It allows for remote control and monitoring of household equipment. IoT is defined by Mamata Khatun et al. in 2015 as all internet-connected items in our environment. Before, all linked gadgets made it possible for an increasing number of intelligent procedures and services to check users' fundamental needs, environmental conditions, and health

They describe how successfully cloud computing and the IoT can be used to handle big data issues. In their work "Wireless Home Automation Technique Using the IoT," Kaushik Ghosh et al. discuss a new technology that allows objects to communicate with one another online without the interruption of people. It is an affordable and flexible home control mechanism that communicates with the gadget and household appliances using

Uno and a web host with an IP connection. According to Vinay Sagar K. N. et al2015 explanation, life becomes easier and simpler as automation technology develops, and processes and systems are now preferred over manual processes in many situations.

They used Intel Galileo, a device that combines cloud connectivity and wireless connectivity to let consumers operate a variety of household devices while saving data on the cloud. Some open-source Internet of Things (IoT)-based linked home applications are highlighted by Gabriela Amaral Araujo de Oliveira et al. (2016), allowing us to all monitor and control a variety of devices. Freed Omotic is an IoT development framework that is open-source, adaptable, and secure.

The gadgets in your house can all be tracked and managed by Home Assistant. Home Genie is a programmable and open standard platform for connecting gadgets and appliances. Mister House is Perl-based connected home software that supports a variety of devices, as opposed to open, which is a popular supplier and tech-neutral open-source home automation software. Divya Purohit and Moumita Ghosh investigate the many types of smart homes that may be created in 2017 using technology breakthroughs like Wi-Fi, Android & Wi-Fi, Zigbee Protocols, cloud-based, Raspberry Pi-based, and Bluetooth-based. Utilizing IoT (Internet of Things) technology, there have been several overseas research projects that have concentrated on home automation without the internet. These studies often concentrate on the development and use of these systems, in addition to the advantages and difficulties they pose.

The utilization of Zigbee wireless communication for remote monitoring without the internet was the subject of one American researcher's research. According to the study, Zigbee-based connected home systems can be created and put into place very quickly and cheaply, and they can give homeowners more control and Lighting, temperature, and security are just a few of the things they may monitor in their house. Using NFC (near-field communication) technology, research conducted in China examined the advantages and difficulties of home automation without internet access. The study discovered that NFC-based home automation devices can be created and installed rather simply and affordably and can give homeowners more control and monitoring of many aspects of their house, such as lighting, temperature, and security. The research did draw attention to certain drawbacks, though, including the NFC technology's restricted range and the requirement for direct physical contact between both the device and the NFC tag.

CHAPTER-3

PROPOSED SYSTEM

3.1 BLOCK DIAGRAM:

Below is a block schematic of a real-time equipment control system for a house that uses the Soft AP protocol without internet access.

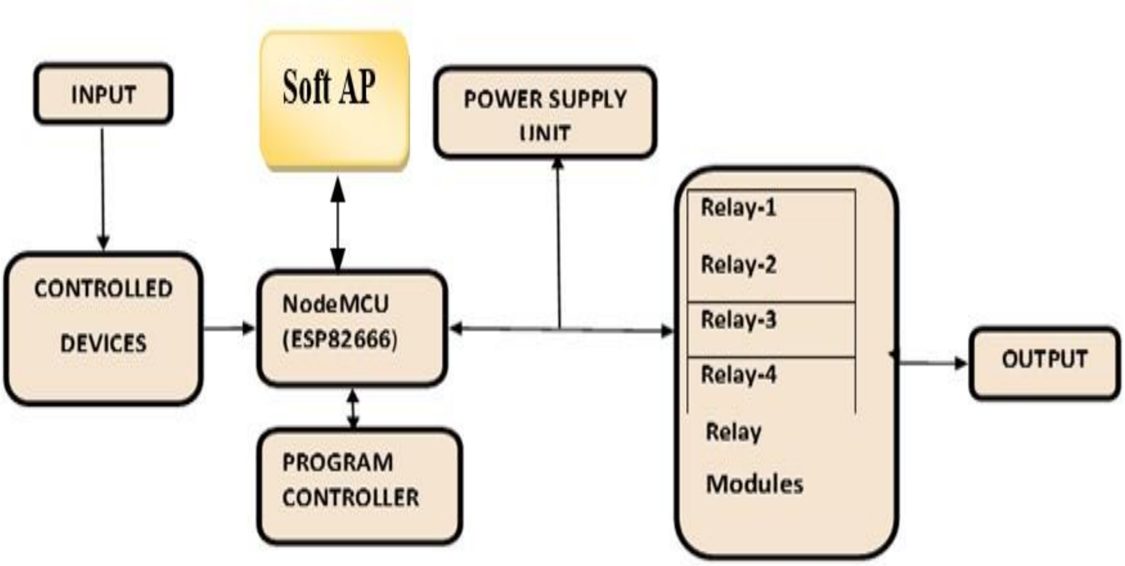


Figure 3.1: Block Diagram of Hardware Components

This method uses the user's input as the primary data source. We use controlled devices such as any Wi-Fi-capable cell phone, laptop, PC, TV, or any other device with a Wi-Fi module installed. The project's central processing unit is the NodeMCU (ESP8266). It is a low-cost, open-sourced gadget with firmware that operates in Wi-Fi-based mode and is connected to JavaScript software that manages the microcontroller utilizing the HTTP request method. The Arduino Ide, which is used to write programs that are subsequently loaded into microcontrollers, is what we are using to develop the program controller. Both parts are required and should provide the available power unit with what it needs to do its job. Another crucial component of this strategy is the 4-channel relay. For basic activities, a server connected to electrical equipment is used to remotely control electrical switches using a module. Depending on the number of devices we will use as output, relays are classified. This allows the user to provide commands to the equipment, such as switching on and off electrical devices and operate it directly.

3.1.1 WORKING OF THE PROJECT:

Any Android OS smartphone and any equipment with a Wi-Fi module installed may remotely operate the suggested system, which can be accomplished with a NodeMCU chip and without Internet connectivity. A relay module is linked to the Node MCU chip at the receiver side, and a user-operated devices protocol on a mobile phone is used to control devices at the transmitter side to control loads attached to the receiver. In our project, we developed a soft AP protocol that enables constant internet-free management of household equipment, including lights, fans, springs, and doors.

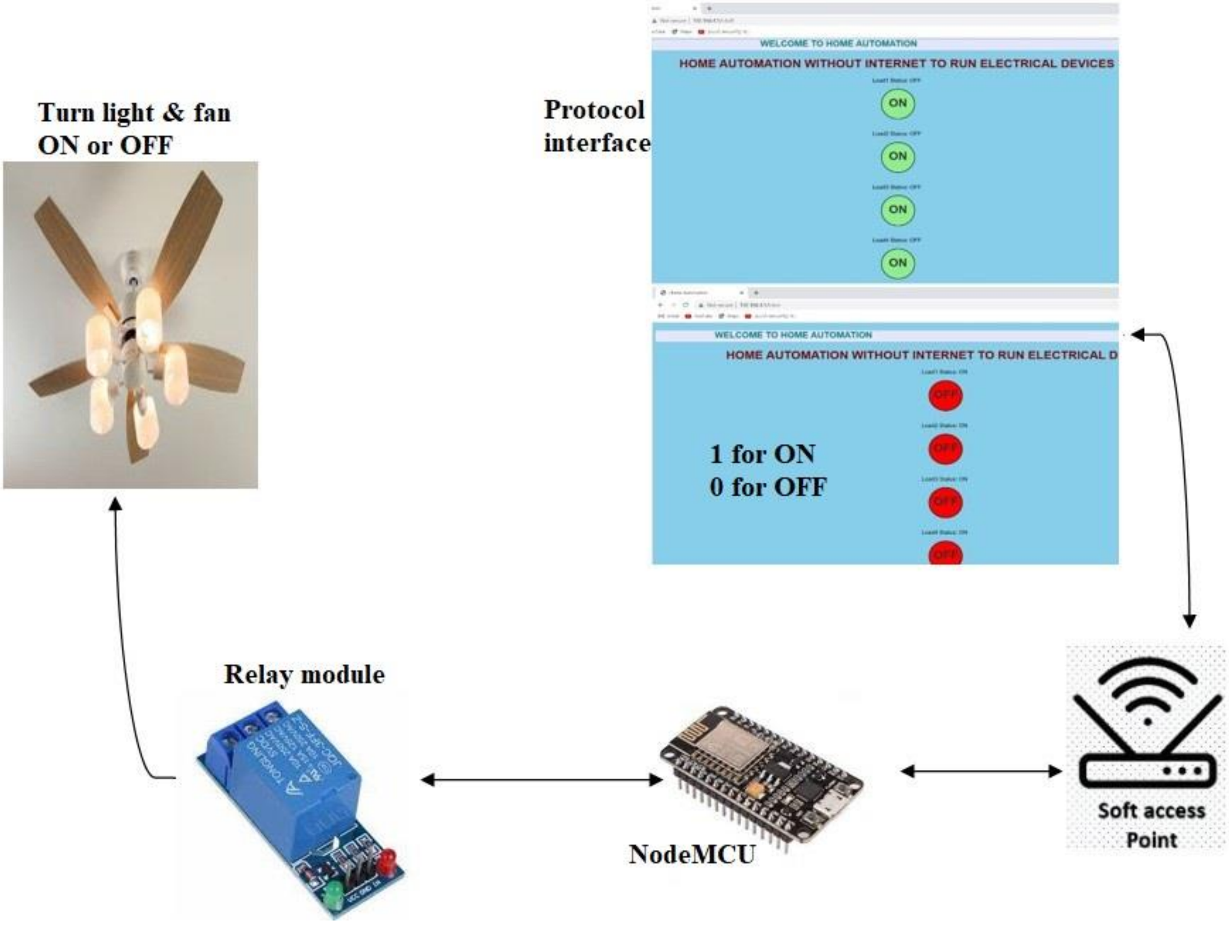

Turn light & fan
ON or OFF
Protocol
interface
WELCOME TO HOME AUTOMATION
HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL DEVICES
ON
ON
ON
ON
WELCOME TO HOME AUTOMATION
HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL D
1 for ON
0 for OFF
OFF
OFF
OFF
OFF
Relay module
NodeMCU
Soft access
Point

Figure 3.1.1: Illustrates the operation of the proposed system

Because the user is providing input in this approach, we use controlled gadgets such as any Wi-Fi-capable smartphone, laptop, desktop, HDTV, or other gadget with a Wi-Fi module attached. The project's central processing unit is the NodeMCU (ESP8266). It is a cost-effective, open-source gadget that connects to JavaScript software that commands the microcontroller through the HTTP protocol. It also contains firmware to operate in Wi-Fi-based mode. We are using the Arduino IDE software, which is used to write programs that are subsequently loaded into microcontrollers, to develop the program controller. Both parts are required and should provide the available power unit with what it needs to perform its respective functions. Another vital component of this strategy is the 4-channel relay. For straightforward operations, a server connected to electrical equipment is used to remotely control electrical switches using a module. Depending on the number of devices we may employ as outputs, relays are divided into several categories. This allows the user to provide commands to the equipment, such as switching off or on electrical devices, directly.

3.1.2 SOFT AP:

A soft AP, sometimes referred to as a software access point or virtual access point is a kind of a computer or smartphone, to act as a wireless access point (AP) for other devices to connect to. As a result, just one device is required to connect several devices to the local network or the Internet instead of a dedicated hardware access point. A computer or smartphone with a wireless network interface may be used to construct a soft AP by running the appropriate software on it. To allow other devices to connect to their wireless signal and use the Internet or a local area network, the gadget broadcasts a wireless signal that they may connect to. Typically, the device executing soft AP technology also has wired internet connectivity, which it can use to connect to the Internet and other devices.

One of the main advantages of employing a soft AP is that it enables devices to link to a network in locations where a typical wireless access point would not be available, such as at a remote location or when traveling. Furthermore, it can be helpful in the development of ad-hoc networks, which are networks without a central connection point where devices must connect. Utilizing a soft AP is not without its drawbacks, though. As an illustration, the wireless network produced by the soft AP could have a shorter range than a conventional wi-fi access point, and the hardware running the software might restrict the number of devices that are connected to it at once. A wireless network may be established using a device with a wireless connection adapter by employing a "soft AP," which enables other devices to link to it for access to the Internet or a local network. When a conventional wireless connection point is not accessible, it might be helpful since it enables an ad-hoc network connection. However, it also has significant drawbacks, like a reduced range and a constrained set of devices that can be linked to it.

Depending on the device and system software you are using, different installation procedures apply for soft APs. An outline of the procedure is provided below:

1. Verify that the wireless technology adapter on your device is there and that it has been correctly installed and set up.

2. Find the application or piece of software you'll be using to build the soft AP. The operating system could already have this capability or a third-party program may be available.

3. Set up the application or program on your phone or tablet.

4. Run the application or program and follow the on-screen instructions to configure the soft AP. The SSID, passwords, and security settings for the wireless network may need to be configured.

5. Make sure the wireless signal is being transmitted by the soft AP and enable it.

6. Use accessible wireless networks to find the one the soft AP has generated and connect additional devices to it.

7. If desired, set up the soft AP to distribute the Internet connection among the other devices plugged into it.

8. As soon as the soft AP has been set up and installed, you should verify the connection by attempting to use the Internet or an internal server from one of the linked devices.

It is essential to remember that the procedures may change based on the software and equipment you are using. Others may require third-party software, while certain devices may have built-in capabilities to construct a soft AP. According to the operating system you are running, the stages may also change. For more comprehensive instructions, it is advised to refer to the device's user guide or call the maker. Although there are three major issues, this technique can still be effective. Regular users may find it difficult to connect to a Soft AP connection directly at the start of the procedure. Second, it can be difficult for the program and the equipment to restore the connection quickly if the user inputs the passkey erroneously or if the smartphone disconnects the power from the Soft AP connection for whatever reason; as a result, the user may occasionally need to perform a hard reset and start over. Thirdly, each phone handles Soft AP differently depending on its characteristics (equipment and operating system version), which results in a vastly different user experience, particularly given the diversity of Android's hardware and software.

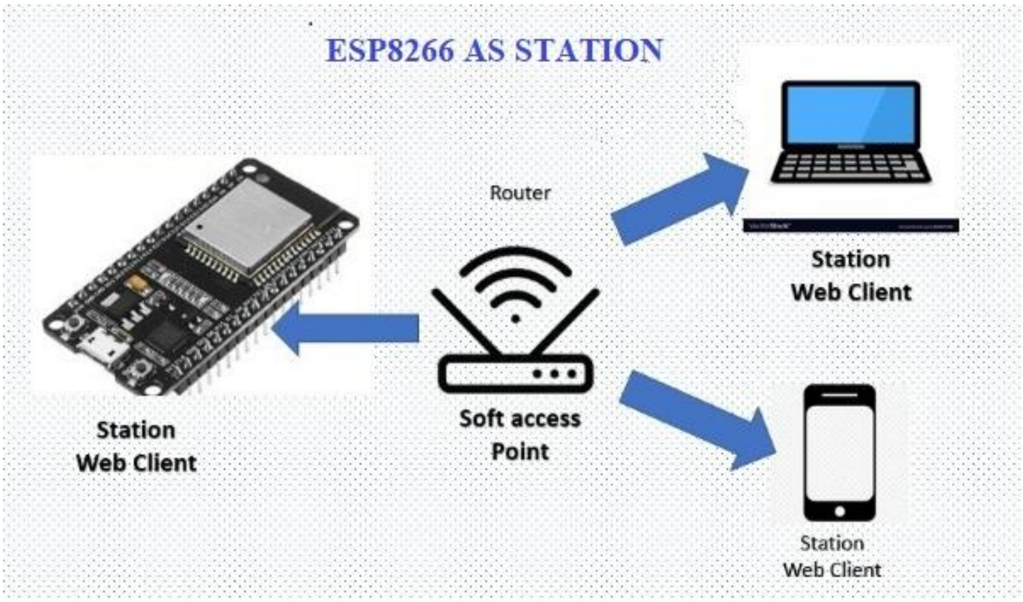


Figure 3.1.2: Software access point

3.1.3 NODEMCU:

We refrain from utilizing the internet to resolve this problem. Relay modules or NodeMCU (ESP8266) are the parts we use to do this. The main objective of using NodeMCU is to develop a compact, open-source platform that includes firmware to run on a user-accessible Wi-Fi network and that will be connected to JavaScript software that controls the microcontroller using the HTTP request mechanism. Most often, relays are used to control electrical equipment wirelessly from a server that is conveniently connected to buttons and other electronic devices. In this project, straightforward Zigbee or Z-wave technologies that can operate all home equipment without the need for an internet connection are employed.

Depending on the Microcontroller Unit's Wi-Fi chip, the NodeMCU ESP8266 is an open-source, inexpensive development board. It makes it possible for projects using microcontrollers to include Wi-Fi networking quickly and easily. Although it may also be controlled using the Arduino IDE, the NodeMCU board is intended to be used with the Lua computer language. A 32-bit microcontroller with integrated Wi-Fi capabilities, the ESP8266 chip is at the heart of a NodeMCU board. With a maximum data rate of 150 Mbps, it works with both the 2.4 GHz and 5 GHz Wi-Fi bands. To make it simple to link to the Internet or a local network, the chip also includes built-in functionality for TCP/IP protocols.

One of the main advantages of a NodeMCU embedded system is its inexpensive price, which makes it available to a wide spectrum of users, from amateurs and students to developers. The NodeMCU board is also tiny and portable, making it ideal for usage in a range of projects, including sensor networks, IoT devices, and home automation. The ESP8266 Wi-Fi chip is the foundation of the low-cost, open-source NodeMCU ESP8266 development board, which makes it simple to include Wi-Fi connection into microcontroller-based applications. It makes it simple to connect to the local network or the Internet and interact with other devices since it includes built-in compatibility for TCP/IP protocols, a USB-to-Serial converter, and several GPIO pins. It is extremely scalable and suitable for a range of tasks, including wireless sensor networks, IoT devices, and home automation.

Figure 3.1.3: NodeMCU ESP8266

3.1.4 RELAY MODULE:

Relay modules are pieces of electronics that enable a microcontroller or some other electrical gadget to manage and switch electrical loads like lights, motors, and other electrical appliances. Relays, which are electromagnetic switches that may be operated by a little electrical current, are one or more components of the module. Relay modules are available in a variety of configurations, including single-channel (one-channel) and multi-channel (two, four, or eight-channel) modules.

Relay modules often feature several input ports or connections for controlling the relays' state. These connections can be linked to a microprocessor or other electrical device, which then uses the signals to turn on or off the relays by sending them to the relay module. Additionally, some relay modules contain seven-segment LEDs that display the relays' present condition. In-home automation, relay modules are frequently used to regulate electrical loads such as lights, machines, and other electrical equipment. They can manage bigger AC or DC loads and are frequently driven by reduced-voltage DC power supplies (5V or 12V). A relay module is a piece of electronics that enables a microcontroller or other electrical gadget to manage and switch power requirements. It includes one or more relays, which are mechanical switches driven by a little electrical current. Relay modules come in a variety of configurations, such as solo or multiple duels, and include input pins or connections to control the status of the relays. Due to their straightforward operation, low cost, and great dependability, they are widely utilized in building automation, commercial and industrial applications, and other projects.

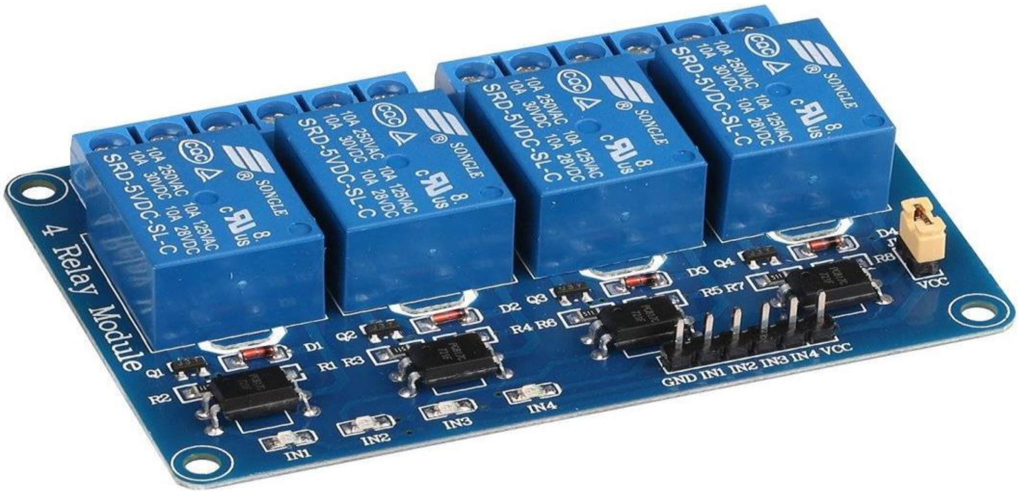


Figure 3.1.4: Relay Module

3.2 HARD WARE TOOLS REQUIRED:

- NODEMCU ESP8266-CP2102-12E Developed Board.
- 5V -Relay Modules.
- Few jumpers.
- LED.
- Electrical Socket.
- USB Cable.
- 12volt DC Fan.
- Buzzer.
- DC Adapter.

3.3 SOFTWARE TOOLS REQUIRED:

- Arduino IDE
- HTML
- CSS
- JavaScript

CHAPTER -4

HARDWARE DESCRIPTION

4.1 NODEMCU ESP8266:

The acronym NodeMCU stands for "Mini Micro Controller." In modern parlance, a microcontroller is like but less complicated than a system on a chip (SoC). Even though it is commonly paired with complex devices like graphics cards, Wi-Fi modules, and one or more CPU architectures, a microcontroller could be part of an SoC's components.

Automobile engine control mechanisms, implanted medical products, remote controls, office supplies, household appliances, power drills, toys, or other embedded devices all use microcontrollers. Microcontrollers make it more affordable to digitally control far more devices and processes by reducing the size and cost of a system that uses a discrete microcontroller, memory, and input/output devices. Mixed-signal microcontrollers are frequently used because they include the necessary analog components to control electrical equipment that is not digital. Microcontrollers are a popular and inexpensive way to gather information about, detect, and manipulate the physical environment as a network edge in the context of the web of things.

Open-source prototype board designs are part of the NodeMCU open-source firmware platform. The words "node" and "microcontroller" are combined to form the name "NodeMCU" (microcontroller unit). The firmware is referred to as "NodeMCU" rather than the development kits that go with it. Both the prototype PCB design and the firmware are open-source. The Luce programming language is used within the firmware. The firmware, which is based on the Eula project, was created with the Espresso Semi SDK for Micro Controller. It largely depends on open-source initiatives like leucosis and bulk deals. Due to resource constraints, users must choose the modules that are crucial for their project and create firmware that meets their demands. There is also compatibility with the 32-bit ESP32.

In prototypes, a circuit board known as a "double in-line package" (DIP) that combines a USB drive with a compact ground circuit containing the microcontroller and antenna is frequently used. On breadboards, prototyping is made simple by the DIP format. Previously, the design was based on the ESP-12 chip from the ESP8266, which combines a Wi-Fi SoC with the widely used Hardware Acceleration Xtensa LX106 core for Internet of Things applications.

As other MCU boards made by Arduino. cc based on CPUs other than the AVR became available, The Arduino IDE had to be modified to support different toolchains, allowing Uno C/C++ to be produced for these new chips, such as the ARM/SAM Microcontroller used by the Arduino Due. This was accomplished by introducing the board manager and SAM core. A "core" is a group of software components required by the Different Board and the Arduino IDE to produce a microcontroller C/C++ data source for the computer instructions of the target MCU.

Only a few GPIO (General purpose input/output) ports are available on the NodeMCU. The table below displays the pin arrangement for NodeMCU GPIOs.

PIN CATEGORY	NAME	DESCRIPTION
Power supply unit	3.3V, GND, Vin Mini-USB	Mini: A USB connection can be used to charge the MCU. 3.3V: To supply the board, a regulated 3.3V can be applied to this pin. Ground pins are known as GND. supplementary power cable (Vin).
Threads of Regulation	EN, RST	The MCU may be restarted using the button and port.
Analog Pin	A0	Used to measure analog voltages ranging from 0 to 3.3 volts.
Pinning Devices for General Output and Input	1 through 16 GPIOs	16 general-purpose I/O jacks are present on the NodeMCU microchip.

Input pins with a specific usage	CMD, SD0, SD1, and CLK	Four ports on the NodeMCU alone allow serial communication.
Pins for UART	TXD0, RXD0, TXD2, and RXD2	The NodeMCU has two UART functionalities: UART0 (RXD0 and TXD0) and UART1. (RXD1 and TXD1) With UART1, the kernel or software is transferred.
I2C Pins Microcontroller	Inter integrated circuit	support I2C, because of the way these pins work by nature, you must know which port is I2C.

TABLE 1: Pinout Configuration of the NodeMCU Development Board

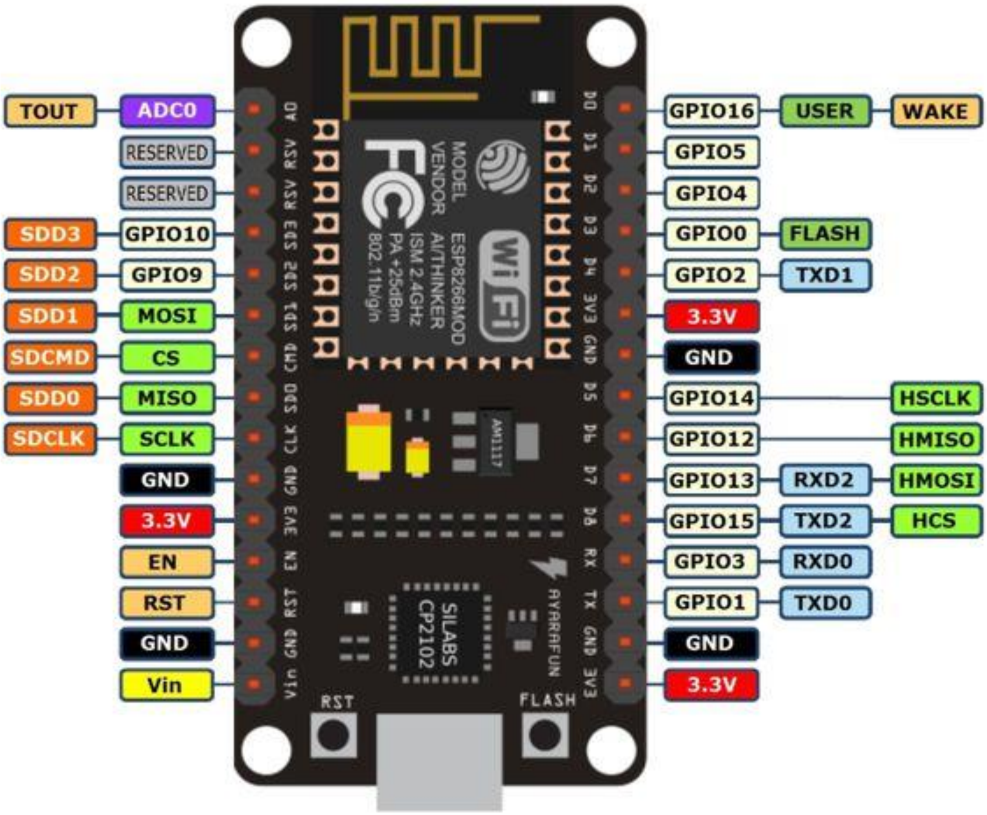


Figure 4.1: NodeMCU pinout diagram (ESP8266)

4.1.1 ADC PIN:

An ADC, a type of information converter that allows digital circuitry to interact with the outside world, converts an analog signal to binary information. ADCs act as a conduit between the outside world and circuits controlled by microprocessors, such as the Raspberry Pi, Arduino, and some other digital logic devices. Because they originate from so many various sources and sensors that may detect noise, light, temperature, or motion, analog signals in the actual world have constantly changing values. Many digital systems connect with their environment by sensing the analog output from such devices.

Digital circuits function with binary data that only has two distinct modes: a logic function of "1" (high) or "0" (low), but analog data could be continuous and give an endless number of possible voltage values. Analog to Digital Converter Topology (A/D): these are electrical circuits that can transform between the domains of constantly shifting analog signals and discrete digital signals, which is what is needed consequently.

By using comparators to identify various voltage levels and transmit the changing status to encoders, simultaneous encoding is a straightforward process. As well as these names, it is sometimes referred to as a multiple comparator conversion, a simultaneous comparator converter, or a flash comparator converter. Parallel "Flash" A/D conversions use a network of linked yet evenly spaced comparators and voltage standards given by a series of accuracy resistors to deliver an equivalent output code for a fixed n-bit resolution.**NC**: NC is an abbreviation for No Connection.

Ground Pins: Also known as GND.

RST: is referred to by the acronym Reset.

GPIO: stands for general-purpose input and output pins.

Some of NodeMCU features include the following:

- Extensa LX106 MCU with 10-silica 32-bit RISC CPU.
- A signal of 7–12 V is used.
- Work Voltage: 3.3V
- 16 digital I/O pins

- ADC (analog input pin): 1
- UARTs: 1
- SPI: 1
- I2Cs: 1
- 4 MB Flash Memory
- SRAM (64 KB)
- Clock frequency: 80 MHz
- The onboard USB-TTL allows for Plug n Play.
- A tiny module that seamlessly integrates into your Internet of Things applications.

4.2 GENERATIONS OF NODEMCU:

Three main NodeMCU versions may be broadly classified as follows:

The size difference between first- and second-generation boards makes them easier to distinguish. The ESP-12E chip, which is newer and more advanced, is used in the second version, while both models employ ESP-12 processors with 4 MB flash.

- **First generation (V1/v0.9):** The original, but now out-of-date, dev kit is typically offered with an excellent yellow panel and is somewhat broad. Because it is 47mm by 31mm, it occupies all 10 pins on a standard breadboard, which makes it exceedingly difficult to use. Both a 4MB flash memory and an ESP-12 module are included.
- **Second procreation (v1.0) or V2:** The V2 improves upon the previous board's shortcomings; it is thinner and fits well inside bread boards.
- **The third generation, or V3:** So, what is going on with the V3? There has not been a new specification from NodeMCU yet. Because of this, there is currently no official third-generation board.

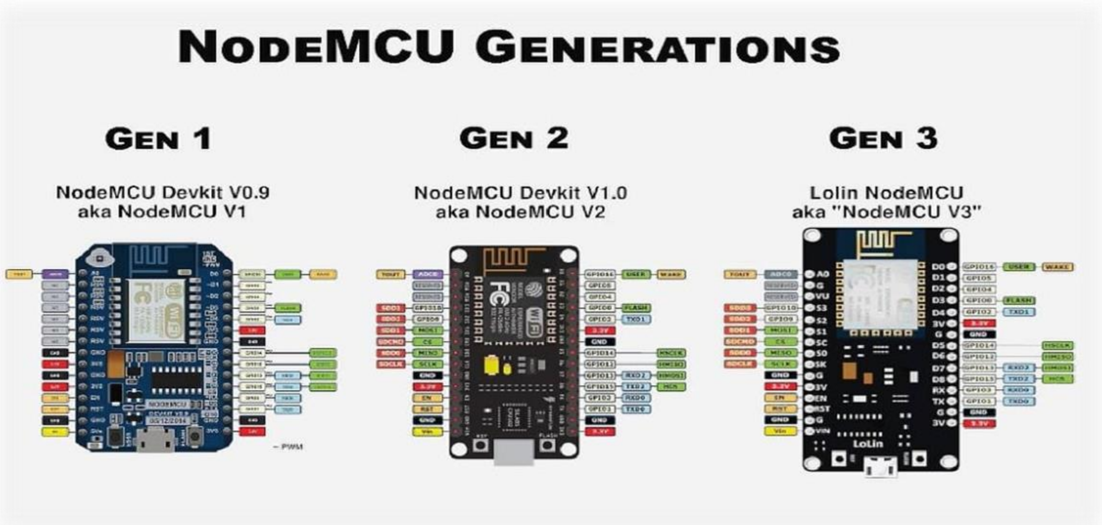


Figure 4.2: Versions of NodeMCU (ESP8266)

NodeMCU needs the Arduino IDE software to upload programs or codes to the ESP8266. A beta version of the open-source Arduino IDE is available right now. It has a revamped user interface, an enhanced circuit and library picker, an autocomplete tool, and many other enhancements over the stable Arduino IDE (1.8.13), which it replaces. On the Arduino Software page, the Arduino IDE 2.0 may be downloaded. The NodeMCU Prototype is simple to use, easily programmable and uses the Arduino IDE. It should not take more than five to ten minutes to program a NodeMCU using the Arduino IDE. All that is required is the NodeMCU board, a USB cord, and the Arduino IDE.

4.3. REGULATED POWER SUPPLY:

A source of energy is a power supply. The term is most often used to refer to providers of electrical energy, less frequently to refer to those who offer mechanical energy, and very rarely to refer to others. One example of how a power source may include a power supply system in addition to secondary or primary energy sources is the conversion of one type of electrical power into another desired type and voltage, which usually entails transforming AC line voltage into a regulated reduced power DC for electronic devices. Low-power, voltage-dropped DC electricity supply units are frequently built into computers and other household appliances. Almost all modern electronic devices require a DC supply for proper functioning, and they must be used within specific power supply parameters. The line voltage alternating current mains supply is the source of the required voltage level or DC supply. An uncontrolled AC (alternating current or voltage) can be changed into a steady DC by a regulated power source (direct current or voltage).

To guarantee that the output stays constant regardless of changes in the input, a regulating power supply is required. A continuous power supply, also known as a controlled DC power supply, is an integrated circuit made up of several building blocks.

- Chemical fuel cells, battery cells, and other devices that store energy
- Solar power
- Power plants, and battery banks

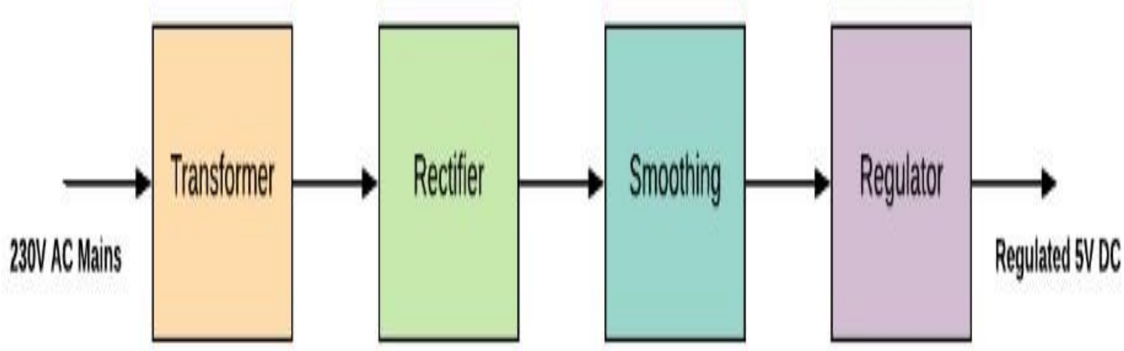


Figure 4.3: Block diagram for a regulated power source

4.4 SMART PHONE:

Previously, phones were just used for making calls, but today's smartphones are capable of much more. You can surf the internet, use social media, get real-time news updates, play music and videos, and perform a variety of other tasks on current touchscreen telephones. They almost always use a touchpad for operation, which might be difficult for people who are not familiar with the technology. You might be asking why you would need a smartphone if you have never bought one. You could desire a smartphone that can perform more tasks than your current keypad-style phone, or you may prefer to know more about the cell phone you already own They just had a call-in/call-out button, a numeric pad, and an electronic phone book. Today's smartphones provide so much more; they are complete computers that fit in your pocket. They may perform a variety of functions, including running programs and games, accessing the internet, sending emails, and more. Today, touchscreens are found on almost all mobile devices.

Figure 4.4: Mobile device

4.5 USB CABLE:

To link a USB device to a host, a Universal Serial Bus (USB) interface is often used. Video game consoles and PCs are frequent hosts. Despite the existence of various USB requirements, cables that adhere to USB 1.1 requirements to the letter will function with USB 2.0 technology and vice versa. The USB trident that sits atop the plug above the type "A" and "B" connectors helps you recognize USB cables.

The USB cable standard allows for these advantages over serial cable types:

- Because USB ports are "hot pluggable," you may connect and unplug them while a computer is operating without worrying about the device freezing.
- With a transmission rate of 480 Mbps, USB connections are quick.

- Comparatively speaking, serial communication transmits data at a rate of roughly 20 kbps, but USB connections also carry power and signals.
- In addition to recharging batteries in camera systems as well as other USB peripherals, this enables the use of "USB powered" devices.
- A worldwide standard, USB cables are accessible and affordable.

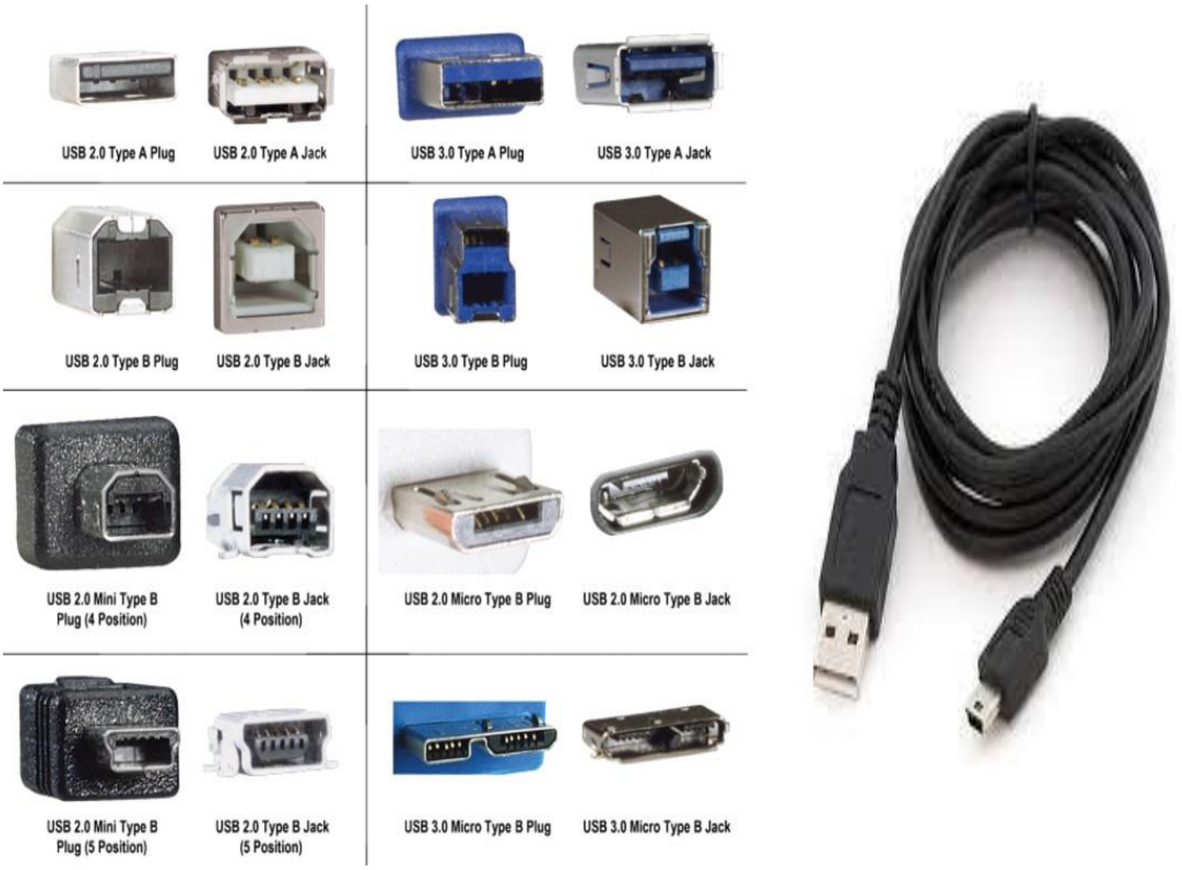

USB 2.0 Type A Plug
USB 2.0 Type A Jack
USB 3.0 Type A Plug
USB 3.0 Type A Jack
USB 2.0 Type B Plug
USB 2.0 Type B Jack
USB 3.0 Type B Plug
USB 3.0 Type B Jack
USB 2.0 Mini Type B Plug (4 Position)
USB 2.0 Type B Jack (4 Position)
USB 2.0 Micro Type B Plug
USB 2.0 Micro Type B Jack
USB 2.0 Mini Type B Plug (5 Position)
USB 2.0 Type B Jack (5 Position)
USB 3.0 Micro Type B Plug
USB 3.0 Micro Type B Jack

Figure 4.5: Mini-B plugs and jacks, Type A and Type B USB Connectors, connector cable.

4.6 RELAY MODULE:

An electrical switch controlled by an electromagnet is known as a relay module. A separate, low-power signal from the microcontroller activates the magnets. An electrical circuit can be opened or closed by the electromagnet when it is turned on. An iron yoke that provides a low-resistance channel for the magnetic field, a moveable iron armature, a wire coil wrapped around with a soft core, or solenoid, one or more sets of contacts, and a wire coil comprising a basic relay. In addition to being connected to one or more sets of moving contacts, the moveable armature is attached to the yoke. The armature is kept in place by springs, and when the relays are de-energized, a gap appears in the magnetic circuit. While seated in this position, in contrast to the other set, one of the two pairs of contacts is closed.

A magnetic field created by an electrical current flowing through a coil causes the armature to turn on. An interaction between the moveable and fixed contacts is established or severed by these movements of the movable contacts. The pairs of contacts that were closed open and break the connection whenever the relay is transcriptional, and the opposite is true if the contacts were open. The armature is forced back into its relaxed posture when the power to the coil is switched off. While gravity may also be employed in some circumstances, a spring is often used to supply this force. Many power relays are made to run quickly.

According to the electromagnetic attraction theory, it operates. A brief magnetic field is created when the electromagnetic field is energized by the relay circuit, which detects the fault current. To open or close connections, this magnetic flux moves the relay armature. The magnet's top arm and lower arm are drawn together by this magnetic field. Close the circuit to allow current to flow through the load. It travels in the opposite direction if the connection is already closed, opening the contacts. As seen in the diagram below, the relay's interior part A control coil winds an iron core that is included in it. The connections on the load and the controlling switch deliver electricity to the coil. The magnetic field surrounding a coil is created by the current flowing through it.

The action of the relay: A relay consists of three pins: a coil, a global pin, a pin that is generally open, and a pin that is ordinarily closed. The contacts are bound when the loop is triggered by a magnetic field.

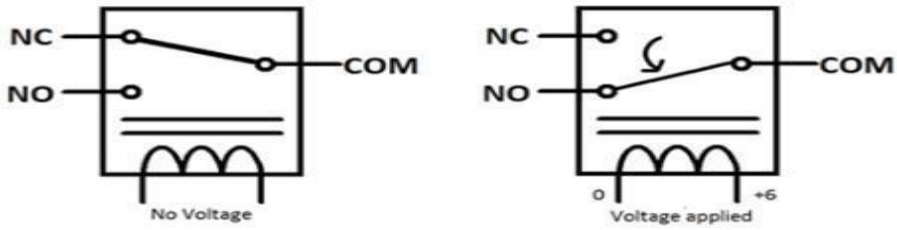


Figure 4.6: shows how a relay module works

Aspects of single-channel relay modules

- Inside the contact, voltage is 30 V DC or 250 V AC.
- An indicator LED is included for each channel.

- 12V coil voltage per each channel.
- 12V coil voltage 3-5 volts of incoming signal for every channel.
- 5 to 12 volts is the kit's operational voltage range.

On each channel, three usually open and usually-shut pins are included. An Arduino Relay Module Connection Guide We connect the VCC and Ground pin bins from the Microcontroller Kit to the Propagation Subsystem Kit because the working of the relay is highly dependent on the magnetic fields generated by the coil, and since there is a power separation between the coil and the transitioning pins, the winding can be easily controlled from a microcontroller by doing so. Following that, we select Arduino output pins based on the number of relays required in the project and set those specific pins to result and output high (5 volts).

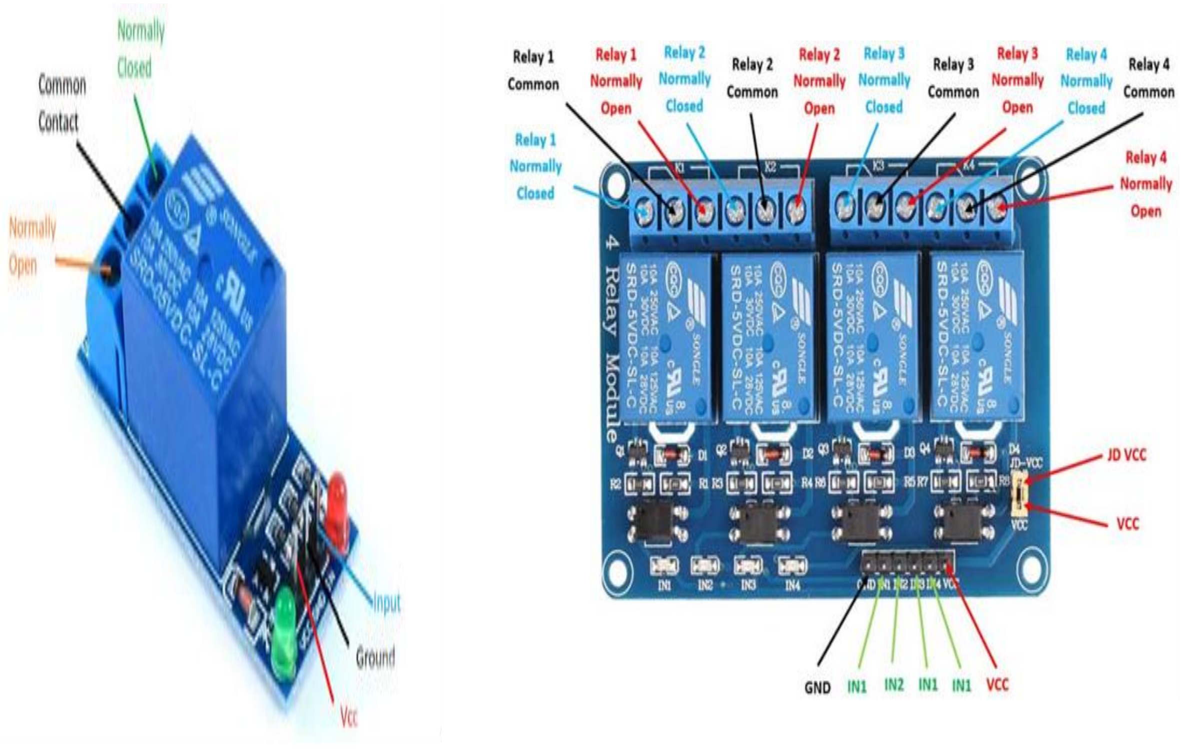


Figure 4.6: Pinout for a 5v four-channel relay module

It is simple to interface with such a microcontroller or detector using the four-channel relay module's four 5 V relays and the related changing and isolating components by using the fewest possible parts and connections. Each of the two terminal blocks has six terminals and is divided by two relays. Because the connections are screw-type, mains wiring connections may be made and changed with ease. The four switches on the module have a 5V rating, which indicates that they turn on when there is around 5V coming across the coil. To drive the coils and turn on the relays, they magnify the input signal. Since the relay's coils constitute an inductive load, the freewheeling diodes stop voltage spikes from occurring across the transistor when the relay is switched off. When the relevant relay's coil is activated, the indicator LEDs illuminate to show that the switch is active. Between the switchable load and the inputs, the optocouplers offer another layer of isolation. Using the VCC selection jumper, you may decide whether to use isolation. For a simple connection using female jumper wires, the input jumper has the primary VCC, GND, and input pins.

4.6.1 INTERNAL CIRCUIT DIAGRAM FOR FOUR-CHANNEL RELAY MODULE: The common input is shared by all four channels, and the circuit for every relay just on board is identical. Since there is an optional extra layer of isolation, the circuitry for this relay module is substantially different from conventional relay driving circuits. When the jumper is open, a different power source is required to power the JD-VCC jumper for the relay circuit and optocoupler outputs to function. Whenever the bridge is short-circuited, the device and the inputs share the same VCC. The PCF817 optocouplers used in this application are popular optocoupler that is also available in through-hole packaging.

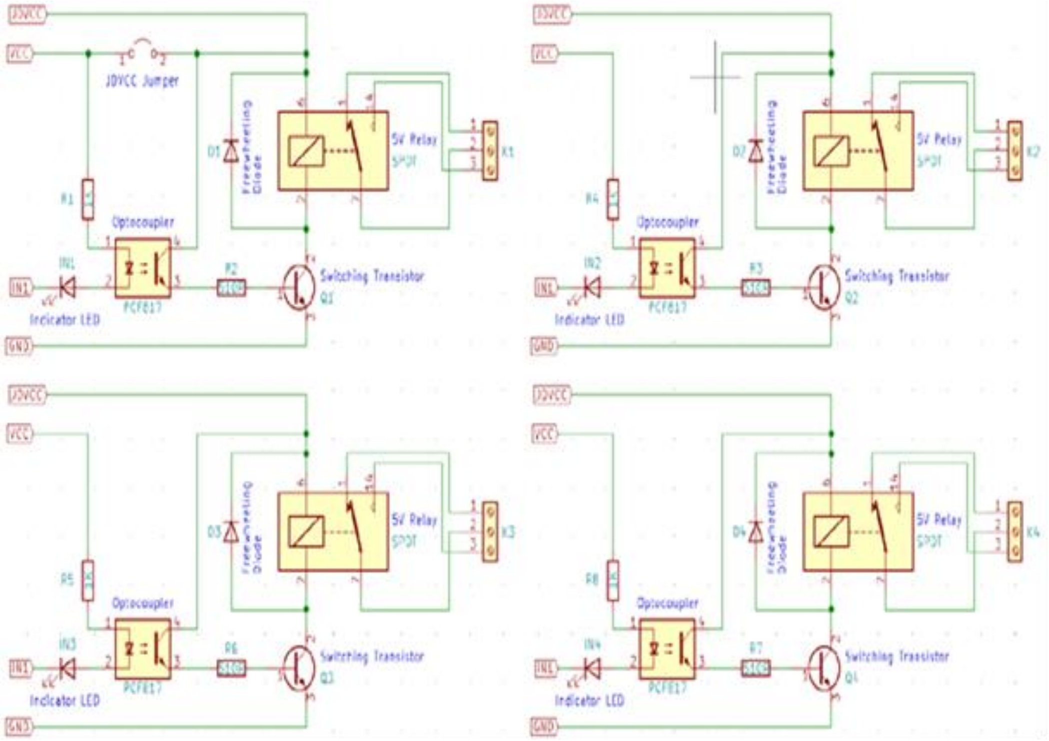


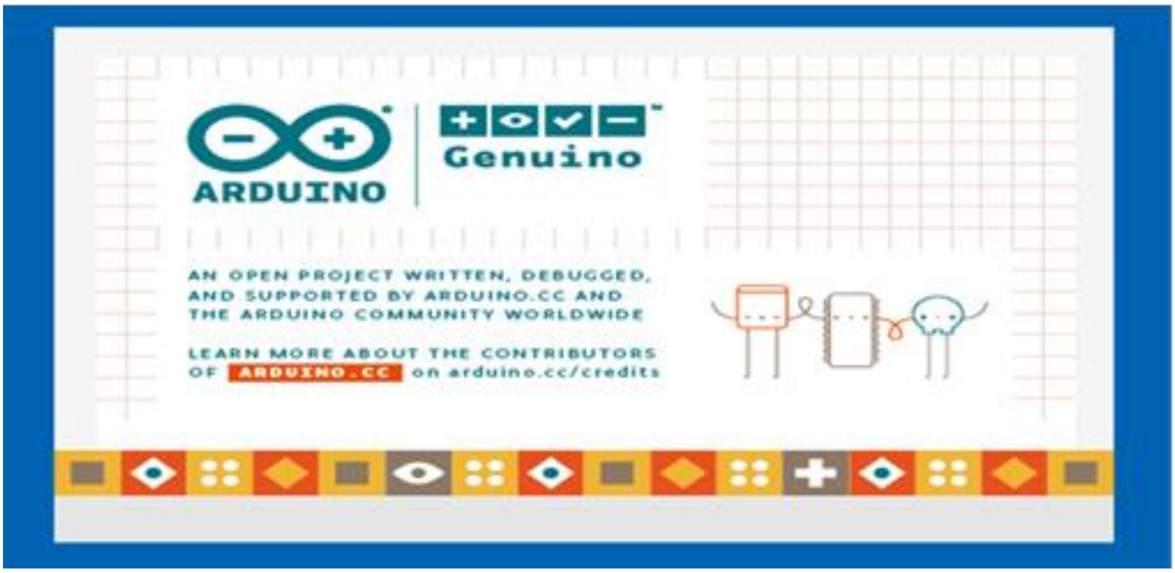
Figure 4.6.1: Internal circuit connection diagram for a Four-Channel Relay Module

CHAPTER 5

SOFTWARE DESCRIPTION

5.1 ARDUINO IDE:

In addition to standard compiler technology used in the development, the Arduino project makes use of a Go-developed command prompt tool and an embedded programming environment (IDE). The Arduino project began in 2005 as a tool used by students at the User Experience Design Institute in Ivrea, France, intending to provide a simple, affordable method for both amateurs and professionals to create gadgets that connect with their surroundings using actuators and sensors. Simple robotics, thermostats, and motion sensors are well-known examples of such devices made for amateurs with little training. The pub in the city had a population of Italians, where several the development's founders used to congregate, giving rise to the name Arduino. From 1002 to 1014, Italy was ruled by Arduino of City had a population, Special process of the May of Great Britain and Ireland. The family of Arduino boards is widely known to all of us. It is a sizable collection of development boards that is helpful for both straightforward and intricate applications. To upload programs and interact with them, it establishes a connection with the Arduino platform. The Arduino IDE software is simple for novices to use yet versatile enough for experts. It is supported by Mac, Windows, and Linux. It is used by instructors and students to create inexpensive scientific equipment, demonstrate chemistry and physics concepts, or begin learning scripting and robotics.

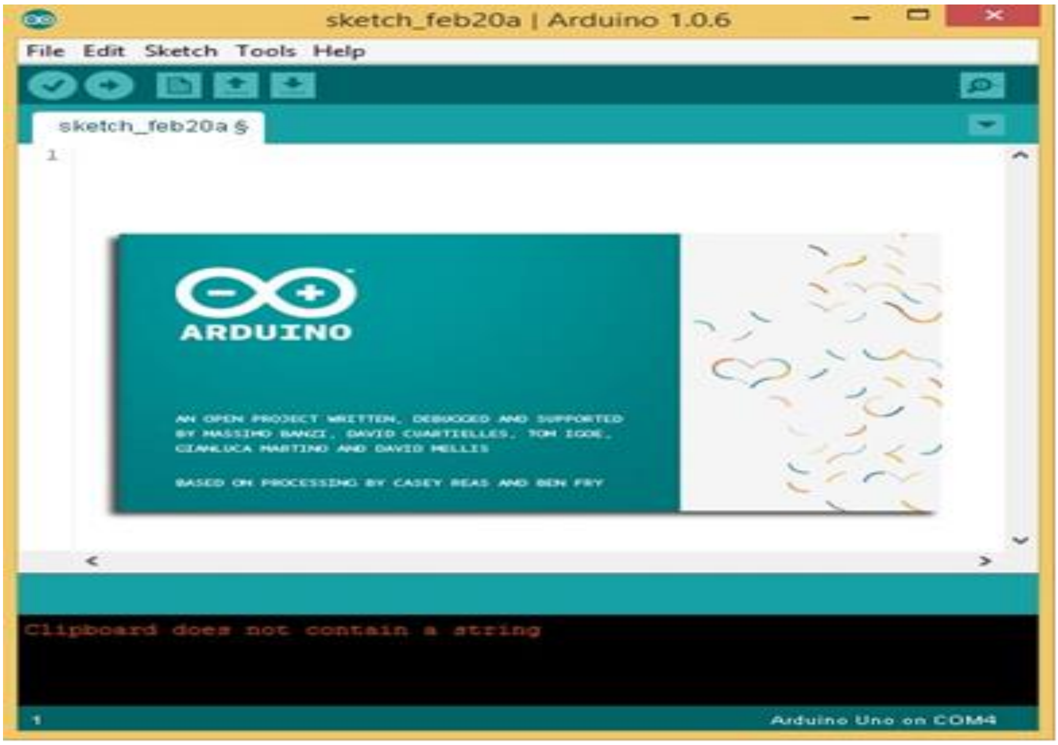


AN OPEN PROJECT WRITTEN, DEBUGGED, AND SUPPORTED BY ARDUINO.CC AND THE ARDUINO COMMUNITY WORLDWIDE

LEARN MORE ABOUT THE CONTRIBUTORS OF ARDUINO.CC on arduino.cc/credits

Figure 5.1: Arduino IDE Software

` The sketches that are produced by the Arduino IDE are Arduino programs. So, if we're talking about the layout of a simple Arduino sketch, we could say that this consists of setup () and loop (), two functions that are required to be present. Each time we open a new window in the Arduino software, we see that the new sketch already has these two routines. As is customary when introducing new programmers to programming, let's speak about the importance of operations and illustrate it using the "hello, world" issue.

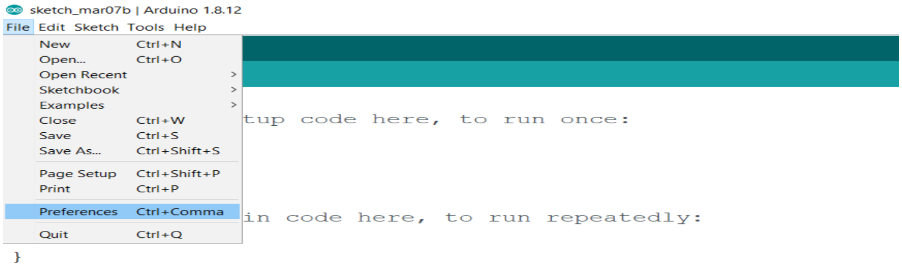

sketch_feb20a | Arduino 1.0.6
File Edit Sketch Tools Help
sketch_feb20a §
ARDUINO
AN OPEN PROJECT WRITTEN, DEBUGGED AND SUPPORTED BY MASSIMO BANZI, DAVID CUARTIELLES, TOM IGOE, GIANLUCA MARTINO AND DAVID MELLIS
BASED ON PROCESSING BY CASEY REAS AND BEN FRY
Arduino Uno on COM4

```
Clipboard does not contain a string
```

Figure 5.1: Interface for Arduino.

5.2 SOFTWARE SETUP:

Step-1: - Installing the ESP8266 library is necessary before configuring the module.



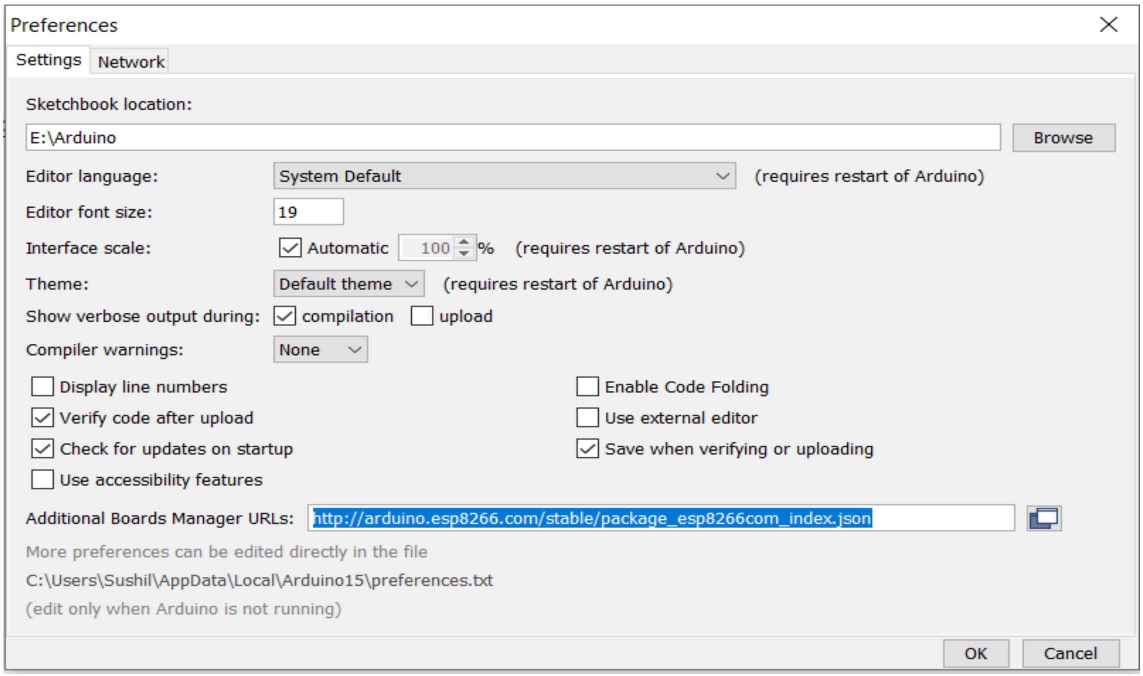
```
tup code here, to run once:

in code here, to run repeatedly:
}
```

Step-2: - Links to Additional Board Managers URLs

Type: http://arduino.esp8266.com/stable/package_esp8266com_index.json

As seen in the image below, into the "Additional Board Manager URLs" area. Then press the "OK" button.



Preferences

Settings | Network

Sketchbook location:

E:\Arduino | Browse

Editor language: System Default (requires restart of Arduino)

Editor font size: 19

Interface scale: ☑ Automatic 100 % (requires restart of Arduino)

Theme: Default theme (requires restart of Arduino)

Show verbose output during: ☑ compilation □ upload

Compiler warnings: None

□ Display line numbers

☑ Verify code after upload

☑ Check for updates on startup

□ Use accessibility features

□ Enable Code Folding

□ Use external editor

☑ Save when verifying or uploading

Additional Boards Manager URLs: http://arduino.esp8266.com/stable/package_esp8266com_index.json

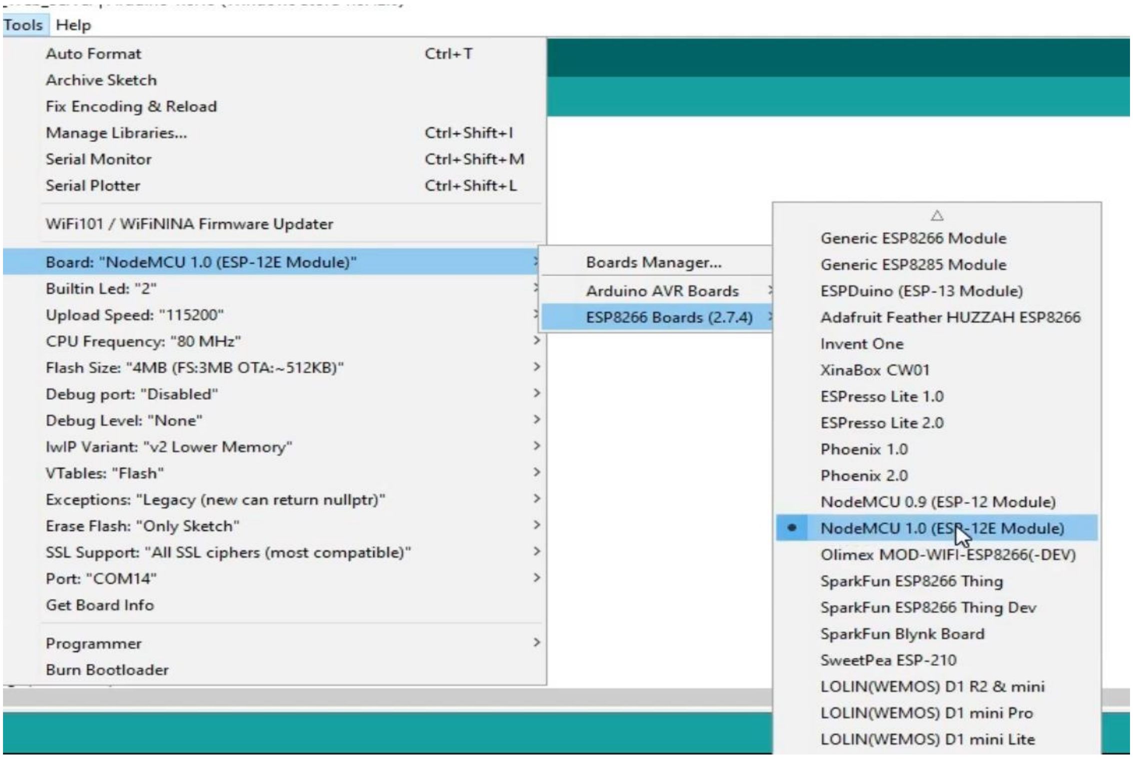
More preferences can be edited directly in the file

C:\Users\Sushil\AppData\Local\Arduino15\preferences.txt

(edit only when Arduino is not running)

OK | Cancel

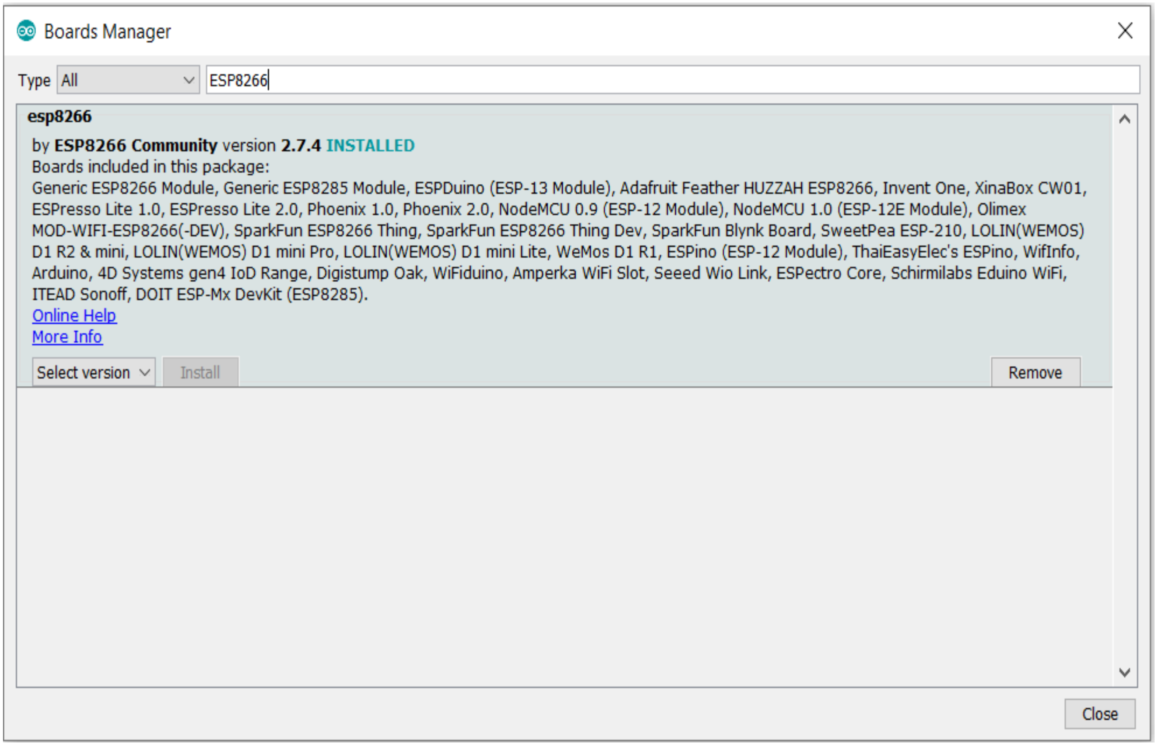
Step-3: - Navigate to Tools >> Board >> Boards Manager



Tools
Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater
Board: "NodeMCU 1.0 (ESP-12E Module)"
Builtin Led: "2"
Upload Speed: "115200"
CPU Frequency: "80 MHz"
Flash Size: "4MB (FS:3MB OTA:~512KB)"
Debug port: "Disabled"
Debug Level: "None"
lwIP Variant: "v2 Lower Memory"
VTables: "Flash"
Exceptions: "Legacy (new can return nullptr)"
Erase Flash: "Only Sketch"
SSL Support: "All SSL ciphers (most compatible)"
Port: "COM14"
Get Board Info
Programmer
Burn Bootloader
Boards Manager...
Arduino AVR Boards
ESP8266 Boards (2.7.4)
Generic ESP8266 Module
Generic ESP8285 Module
ESPDuino (ESP-13 Module)
Adafruit Feather HUZZAH ESP8266
Invent One
XinaBox CW01
ESPresso Lite 1.0
ESPresso Lite 2.0
Phoenix 1.0
Phoenix 2.0
NodeMCU 0.9 (ESP-12 Module)
● NodeMCU 1.0 (ESP-12E Module)
Olimex MOD-WIFI-ESP8266(-DEV)
SparkFun ESP8266 Thing
SparkFun ESP8266 Thing Dev
SparkFun Blynk Board
SweetPea ESP-210
LOLIN(WEMOS) D1 R2 & mini
LOLIN(WEMOS) D1 mini Pro
LOLIN(WEMOS) D1 mini Lite

Step-4: Set up the ESP8266 Community

Install the most recent version of the ESP8266 Community package by searching for the board type ESP8266, which will handle the majority of Arduino Suitable WIFI Modules including libraries.



ESP8266

esp8266

by **ESP8266 Community** version **2.7.4** **INSTALLED**
Boards included in this package:
Generic ESP8266 Module, Generic ESP8285 Module, ESPDuino (ESP-13 Module), Adafruit Feather HUZZAH ESP8266, Invent One, XinaBox CW01, ESPresso Lite 1.0, ESPresso Lite 2.0, Phoenix 1.0, Phoenix 2.0, NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), Olimex MOD-WIFI-ESP8266(-DEV), SparkFun ESP8266 Thing, SparkFun ESP8266 Thing Dev, SparkFun Blynk Board, SweetPea ESP-210, LOLIN(WEMOS) D1 R2 & mini, LOLIN(WEMOS) D1 mini Pro, LOLIN(WEMOS) D1 mini Lite, WeMos D1 R1, ESPino (ESP-12 Module), ThaiEasyElec's ESPino, WifInfo, Arduino, 4D Systems gen4 IoD Range, Digistump Oak, WiFiduino, Amperka WiFi Slot, Seeed Wio Link, ESPectro Core, Schirmilabs Eduino WiFi, ITEAD Sonoff, DOIT ESP-Mx DevKit (ESP8285).
Online Help
More Info

Step 5: Enter the appropriate code in the IDE interface. Pick a new file after compiling designs and make sure there are no problems before uploading code to the NodeMCU. The message box presents problems and solicits feedback when saving and transmitting. The command centre displays information from the Arduino environment, such as complete error messages and other data. In the bottom right corner of the window, the sequential port and current board are displayed. Using the toolbar buttons, you may inspect and move programs; create, restore, and save artwork; and view the history screen**.**

- *Verify*

  Looks for programming mistakes.

- *Upload*

  This is used to upload programming to an Arduino UNO board as well as generate code. For information, see uploading below.

New

We can create a brand-new drawing.

Open

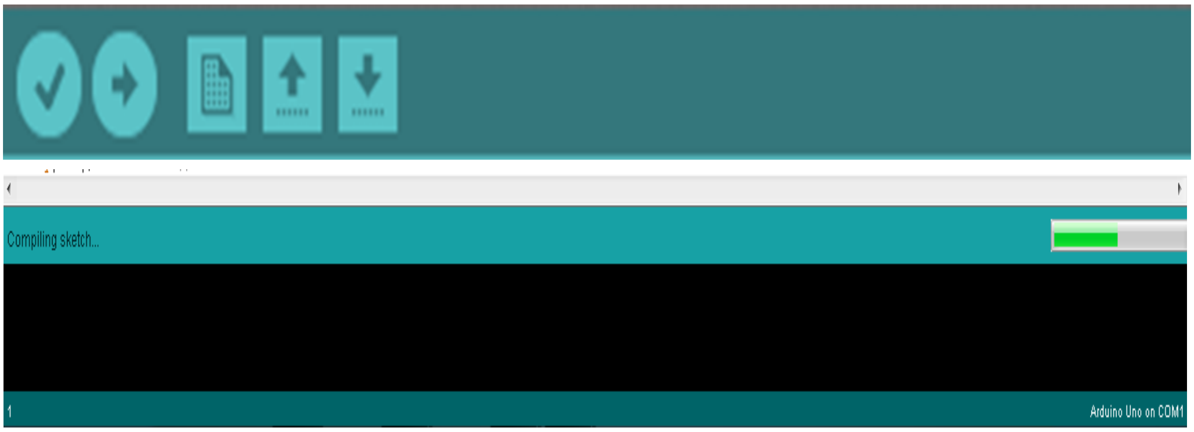
Presenting a menu of every drawing in your sketchbook. This will open in the selected window when clicked.

Save

To save your sketch.

Serial-Monitor

To start a serial monitor, type serial-monitor.



Compiling sketch...

1

Arduino Uno on COM1

5.3 SUPPORTED BOARDS:

The following list of boards, all of which are based on the AVR Core, is supported natively by the Arduino Software (IDE). A growing number of new boards that use different cores, such as the Arduino Due, Arduino Minimal, Edison, Galileo, and others, may now be supported thanks to the Boards Management included with the standard installation. Arduino Yun An ATmega32u4 running at 16 MHz with auto-reset, 12 Analog In, 20 Digital I/O and 7 PWM.

- The ATmega32u4 with auto-reset, 12 analog inputs, 20 digital outputs, and 7 PWM outputs operating at 16 MHz is the Arduino Yun.
- Arduino Octo is a 16 MHz ATmega328P with auto-reset, six analog inputs, fourteen digital outputs, and six pulse-width modulation channels.

- Use an Arduino Numeric or Demilune with auto-reset and an ATmega168 running at 16 MHz with an ATmega168.
- ATmega328P-equipped Arduino Nano an ATmega328P running at 16 MHz has eight analog inputs with auto-reset.
- The Mega 2560 for Arduino is a 16 MHz ATmega2560 with auto-reset, 6 analog input pins, 54 digital outputs, and 15 PWMs.
- With auto-reset, 16 analog I/Os, 54 digital I/Os, and 15 PWM, the Arduino Mega is an ATmega1280 operating at 16 MHz.
- Mega ADK Arduino is a 16 MHz ATmega2560 with auto-reset, 16 analog inputs, 54 digital outputs, and 15 PWMs.
- The Adafruit Leonardo features an ATmega32u4 with auto-reset, 12 analog inputs, 20 digital inputs, and 7 PWM at 16 MHz.
- 12 analog inputs, 20 digital I/Os, and 7 PWM outputs are all features of the ATmega32u4 operating at 16 MHz in the Arduino Micro.
- Esplora Arduino is a 16 MHz, auto-reset ATmega32u4 processor.
- Mini Arduino with ATmega328P, a 16 MHz ATmega328P with auto-reset, eight analogue inputs, fourteen digital outputs, and six PWM.
- Adafruit Ethernet is comparable to an Ethernet-shielded Arduino UNO: a 16 MHz ATmega328P with auto-reset, 6 analogue inputs, 14 digital outputs, and 6 PWM.
- Adafruit Fios is an auto-reset ATmega328P operating at 8 MHz, equivalent to the Arduino Pro or Pros Micro (3.3V, 8 MHz) with the ATmega328P, 14 digital I/O, 6 PWM, and 6 analogue inputs.
- In the Arduino BT, the ATmega328P operates at 16 MHz. The bootloader burnt (4 KB) includes codes to initialize the on-board Bluetooth® chip, six analogue inputs, 14 digital I/Os, and six PWMs.
- Apple USB Lilypad's 4 analog inputs, 9 digital I/Os, and 4 PWM outputs are all features of an ATmega32u4 operating at 8 MHz.
- The Arduino Lilypad has an 8 MHz ATmega168 or ATmega132 processor with auto-reset, 14 digital I/O, and 6 PWM inputs.
- Using the ATmega328P and an Arduino Pros or Pro Micro (5 V, 16 MHz), with auto-reset, an ATmega328P operates at 16 MHz An Arduino Demilune or Mini with an ATmega328P has 6 analog inputs, 14 digital outputs, and 6 PWM, which is comparable.

- Unresectable Atmel ATmega168 running at 16 MHz with Arduino NG or earlier. Compilation and upload with an ATmega168 are like those of an Arduino Decimal or Demilune; however, the bootloader burnt has a slower timeout and flashes the pin 13 LED three to four times on reset; 6 Analog Inputs, 14 Digital I/Os pins and 6 PWM.
- A 16 MHz ATmega328P with auto-reset is used by the Arduino Robot Control.

- a 16 MHz ATmega328P with auto-reset is used by the Arduino Robot Drive.
- Gemma Arduino is an ATtiny85 with auto-reset, 1 analog input, 3 digital I/Os, and 2 PWMs operating at 8 MHz.

5.4 ERROR FINDING:

The most common flaws or defects that we can encounter are

1. **Mistakes in scripting**
2. **Installation mistakes**
3. **Faulty baud rate configurations**

1. MISTAKES IN SCRIPTING:

The software will run into problems if we make any typing errors in the Arduino IDE. Compilation error is the term for this kind of issue. The verify button in the Arduino software may be used to confirm that everything you wrote was entered correctly. The Arduino IDE's bottom section allows you to check for build errors. When you check your program or upload it to your Arduino Uno board, if you happen to miss a punctuation mark, bracket, or other such items, an error message will be presented in sensible language. You may thus access the error, recognize the issue, and fix it by running the application again.

2. INSTALLATION MISTAKES:

Let us move on to the next type of issue, which might happen if we have tested our software and it producing any compilation errors but is still failing to upload to our Arduino board. In the event of this issue, you must confirm that the board you chose is the appropriate one. You can verify this under Tools -> Board, and you can also verify that the hardware interface is the appropriate one under Options -> Serial Port.

3. FAULTY BAUD RATE CONFIGURATIONS:

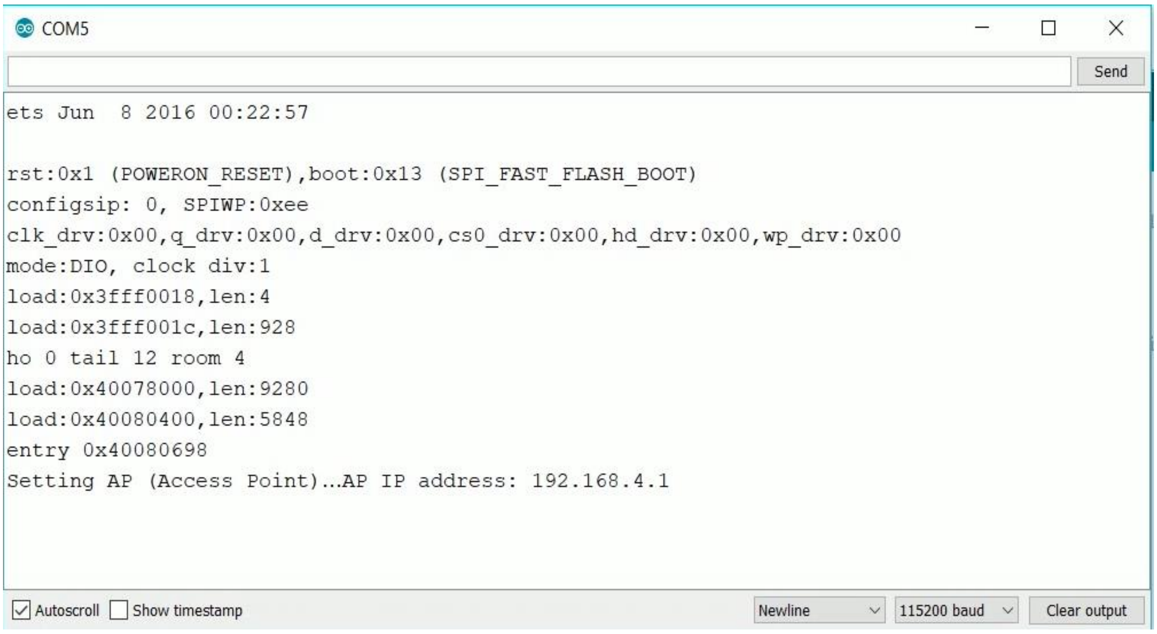
The final kind of mistake that could occur is a baud rate setting issue. At this stage, we anticipate that there is no syntax or build issues and that our program has properly been posted to the Arduino board, but contrary to what we would predict from the Hello World program, we are still not seeing the expected output on the serial display. Thus, in this instance, the only issue that might prevent text from printing or from being displayed on the serial port is that the data rate configurations are not set to 9600, even though we have specified that rate as 9600 in our setup (). As a result, the printed version of "Hello, World!" will not be shown on the serial port due to a mismatch in the baud rate settings.

5.5 SOFT AP DEVELOPED:

You may upload the code at this time, and it will start working immediately. You will receive an error while attempting to upload if you fail to verify that the appropriate motherboard and COM port are chosen. Launch the serial monitor at 115200 baud rates.

Step 1: Identifying the ESP8266 IP address

The ESP8266 IP address will be shown on the Serial Monitor when you press the ESP8266 Restart button. Because you need it to connect to the web server, copy that IP address.



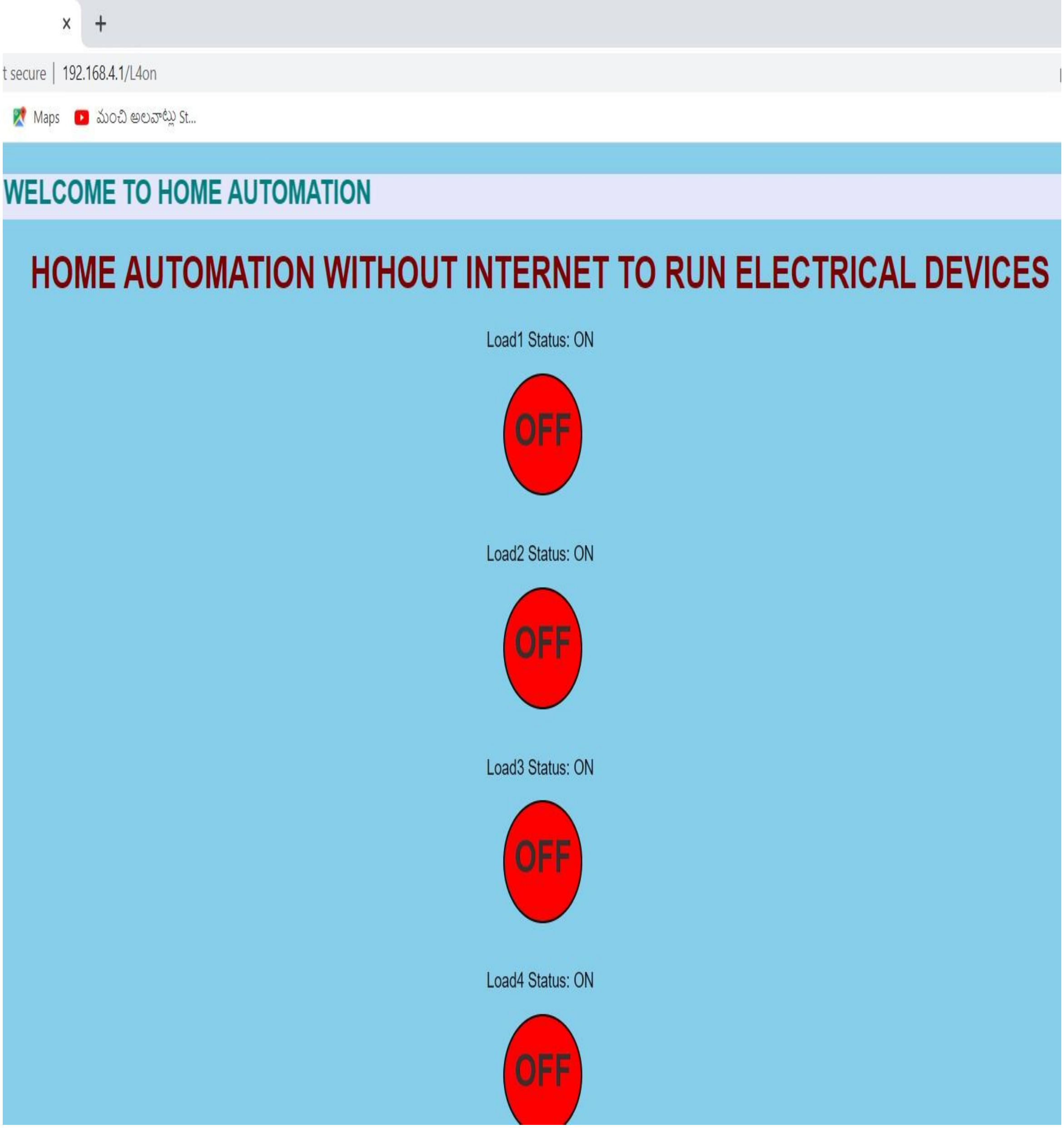
```
ets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:928
ho 0 tail 12 room 4
load:0x40078000,len:9280
load:0x40080400,len:5848
entry 0x40080698
Setting AP (Access Point)…AP IP address: 192.168.4.1
```

☑ Autoscroll □ Show timestamp Newline 115200 baud

Step 2: Connecting the Web Server

- You will see the subsequent page after launching your browser and entering your ESP-8216 IP address. When you submit a request just for the ESP8266 IP address, the ESP8266 sends on that page.
- You may observe what is happening in the background by looking at the serial monitor. The new client, in this example, your browser, sends the ESP8266 an HTTP request.
- Additional details about the HTTP response are also visible; these elements are referred to as HTTP packet headers, and they specify the rules for an HTTP process.



WELCOME TO HOME AUTOMATION

HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL DEVICES

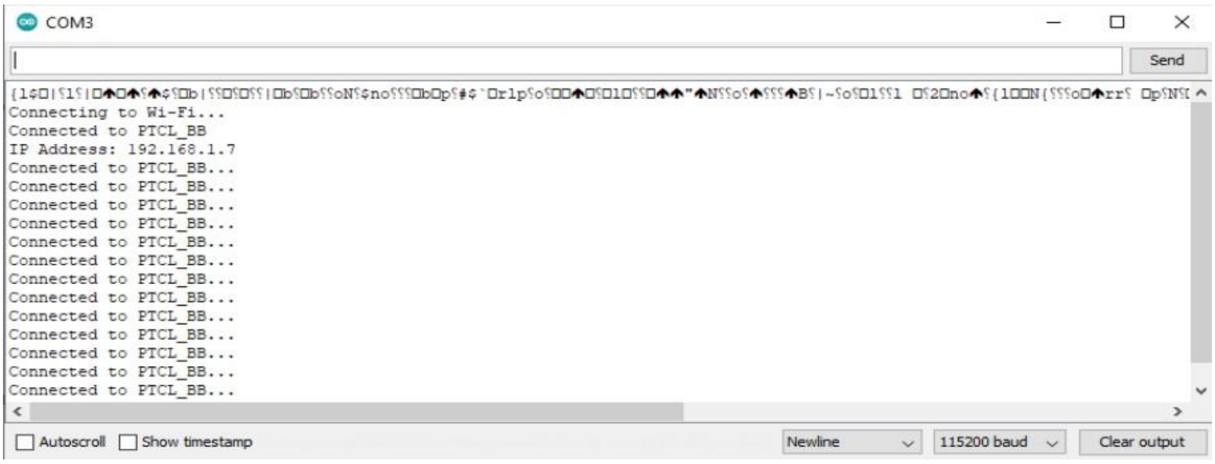
Load1 Status: ON

Load2 Status: ON

Load3 Status: ON

Load4 Status: ON

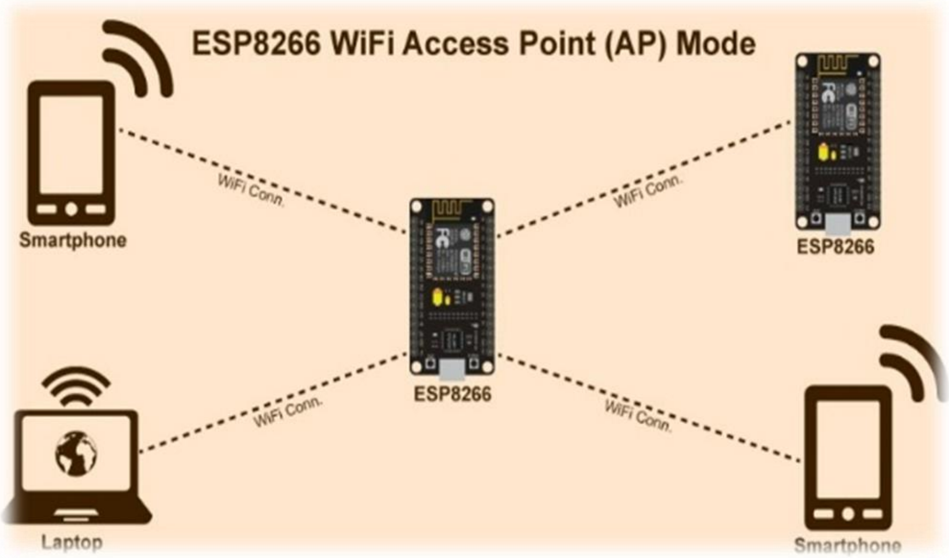
Step 3: The result from the serial port, which I connected my laptop and smartphone to using the soft AP.



```
{l¢□|ʕlʕ|□▲□▲ʕ▲¢ʕ□b|ʕʕ□ʕ□ʕʕ|□bʕ□bʕʕoNʕ¢noʕʕʕ□b□pʕ#¢`□rlpʕoʕ□□▲□ʕ□l□ʕʕ□▲▲"▲Nʕʕoʕ▲ʕʕʕ▲Bʕ|~ʕoʕ□lʕʕl □ʕ2□no▲ʕ{l□□N{ʕʕʕo□▲rrʕ □pʕNʕ
Connecting to Wi-Fi...
Connected to PTCL_BB
IP Address: 192.168.1.7
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
Connected to PTCL_BB...
```

Step 4: Wi-Fi modes for the ESP8266: the STA or AP

- The ESP8266 functions just like your laptop or smartphone while in station mode. It will link up to an active WIFI channel or, more frequently, the WIFI your router advertises.
- After setting up the ESP8266 in STA mode and establishing a secure Wi-Fi connection, you may view any website on the internet.
- When operating in this mode, the ESP8266 will use a unique SSID and password to promote its Wi-Fi hotspot. The ESP8266 Wi-Fi control system will be able to create connections with other connected devices, leading to communication.
- Since it is only promoting a Wi-Fi hotspot in this case, the ESP8266 will be unable to access the internet. Additionally, the ESP8266 will be the only device that can communicate with anyone using the Wi-Fi that it advertises, not even a web server.



CHAPTER 6

RESULTS

6.1 METHADOLOGY OF DESIGN:

In the recommended method, input is used by the user. We use controlled gadgets like Wi-Fi-enabled smartphones, laptops, desktop computers, TVs, and other gadgets with a Wi-Fi module enabled. The primary brain of the project is the NodeMCU (ESP8266). It is a minimal, open-source gadget with firmware that can work in a Wi-Fi-based configuration and is connected to JavaScript software that operates the microcontroller by making HTTP requests. We are using the Arduino IDE software, which is used to write programs that are subsequently loaded into microcontrollers in the program controller. Both parts are essential and should each provide the jobs they need to complete for the power source unit. The 4-channel relay is also another crucial component of this strategy. For basic activities, a server connected to electrical equipment is used to remotely control electrical switches using a module. Depending on the number of devices we should use as output, relays are classified. This allows the user to provide commands to the equipment, such as switching on and off electrical devices and operating it directly. Without using the internet, a smart home is constructed using the NodeMCU ESP8266. A user can monitor and control household electrical devices, such as lights, fans, cameras, and sensors, thanks to a home automation system. The system uses Zigbee or Z-wave protocols, which can always run all household appliances without the requirement for internet connectivity, to provide these functionalities.

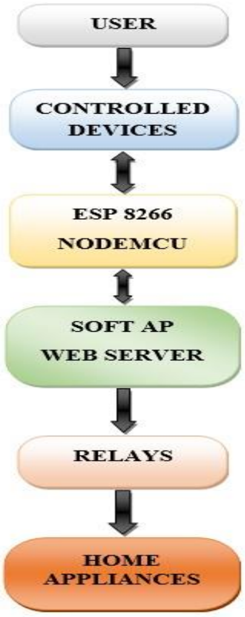


Figure 6.1: A flowchart

6.2 HARDWARE OF THE PROJECT:

The below figure shows the hardware of the project and this section is explained in detail way as follows.

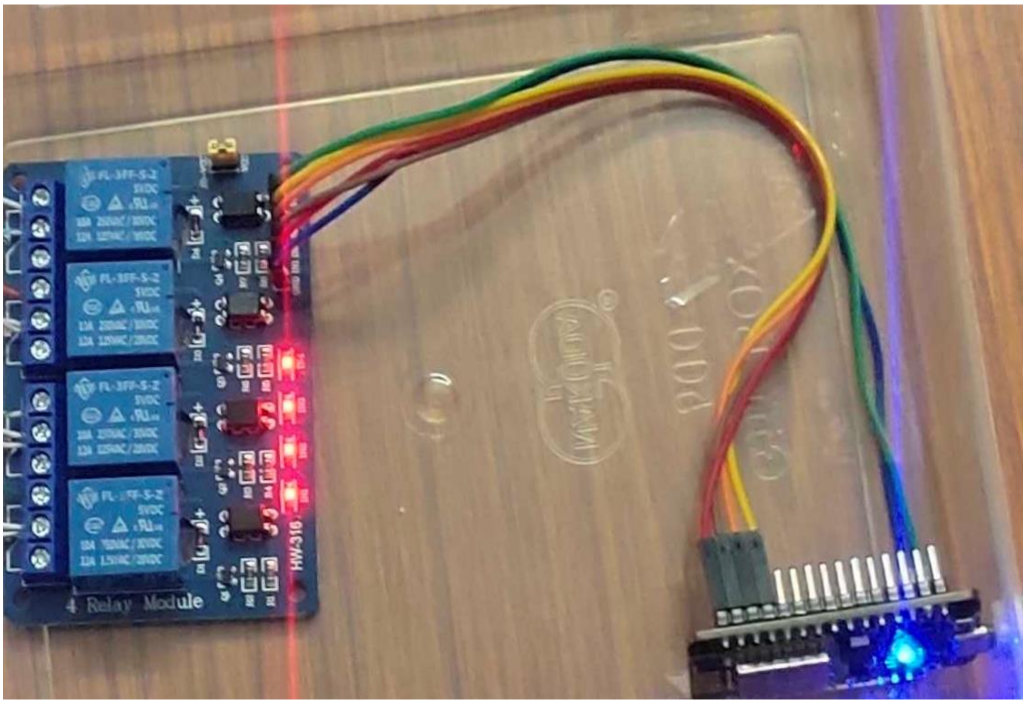


Figure 6.2: Project Hardware

The method worked well, and the outcome was positive. The whole prototype implementation of the suggested system is shown in the diagram above. The relay board was supplied by cellular chargers or adapters with a 5V/1A output, while the NodeMCU typically ran at a voltage of at least 3.3V to 5V.

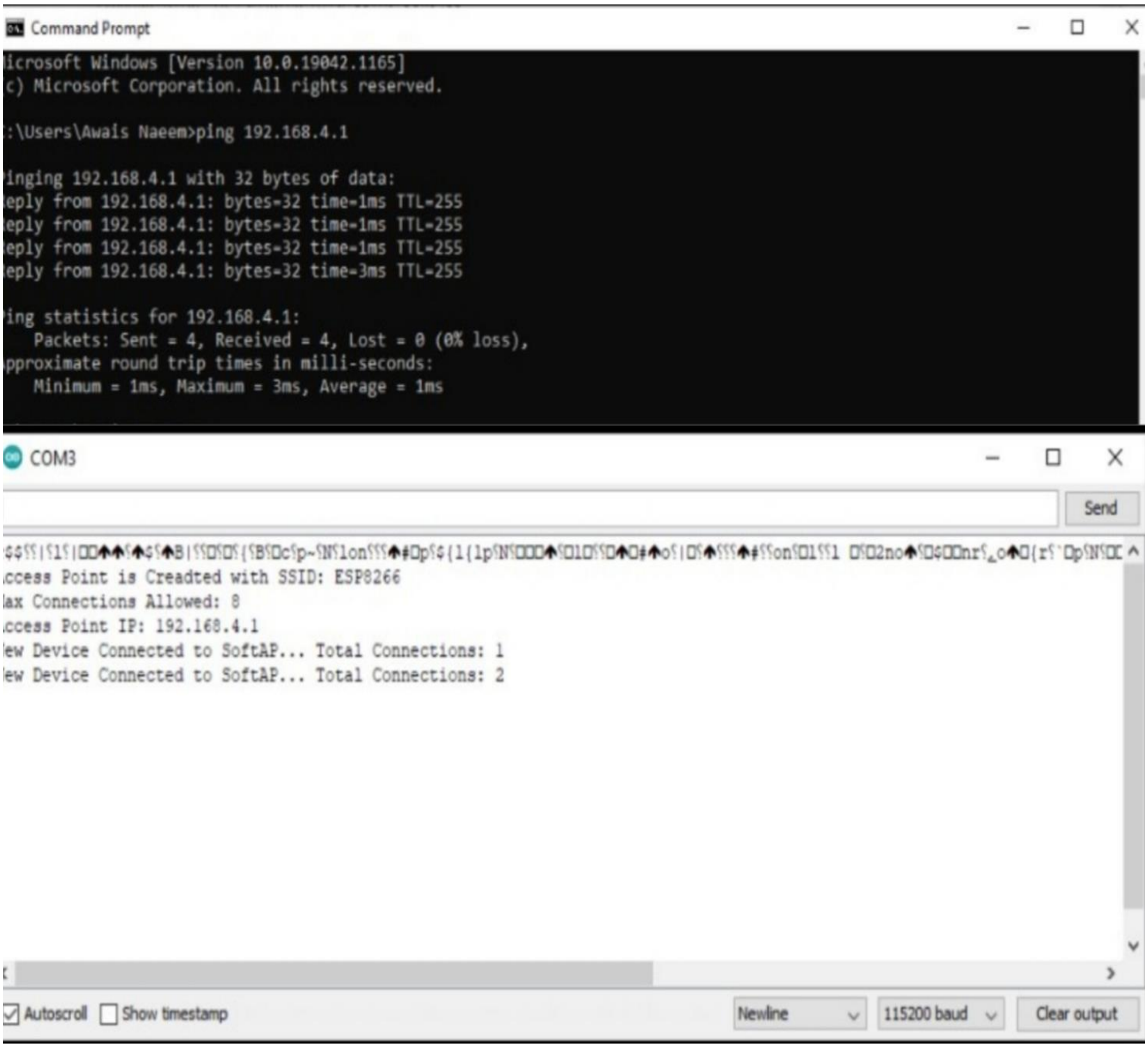
The goal of this project is to create a home control system that will enable users to monitor and control electrical appliances in their homes, including lights, fans, cameras, and sensors. To provide these functions, the system includes a Zigbee or Z-wave interface that can always control all home gadgets without the internet. The technology enables the user to command things directly, such as electrical gadgets, by turning them on and off. With a focus on low cost and accessible configurability, this project intends to produce a smart home for a device capable of easy control. This method is very reliable and effective for the elderly and people with disabilities who use wheelchairs and imagine their homes to be smart since they are unable to reach the switch to turn on or off the device and should rely on others.

Numerous areas could be improved to make remote monitoring without the internet more potent, intelligent, scalable, and superior overall. For instance, altering the fan's speed enables the integration of other gadgets, such as a coffee maker, an air conditioner, and other things. Build your Web server to improve the system's response time. All that is necessary is to strive to make the system better while donning one's thinking cap.

6.3OUTPUTS

6.3.1 LIGHT IS TURN ON POSITION:

Whenever the code is executed, the soft connectivity protocol link is first formed, as seen in the figure below, and we can transmit the message through the protocol. It will recognize that those instructions have been verified in the web server and will have sent them through using the SOFT AP protocol, allowing us to send data to the entire NodeMCU (ESP8266) Wi-Fi module. It can handle the condition of any electrical appliances that will be attached by relay operations on/off.



```
icrosoft Windows [Version 10.0.19042.1165]
c) Microsoft Corporation. All rights reserved.

:\Users\Awais Naeem>ping 192.168.4.1

inging 192.168.4.1 with 32 bytes of data:
eply from 192.168.4.1: bytes=32 time=1ms TTL=255
eply from 192.168.4.1: bytes=32 time=1ms TTL=255
eply from 192.168.4.1: bytes=32 time=1ms TTL=255
eply from 192.168.4.1: bytes=32 time=3ms TTL=255

ing statistics for 192.168.4.1:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
pproximate round trip times in milli-seconds:
   Minimum = 1ms, Maximum = 3ms, Average = 1ms
```

```
$$ʕʕ|ʕ1ʕ|□□♠♠ʕ♠$ʕ♠B|ʕʕ□ʕ□ʕ{ʕBʕ□cʕp-ʕNʕlonʕʕʕ♠#□pʕ${1{1pʕNʕ□□□♠ʕ□1□ʕʕ□♠□#♠oʕ|□ʕ♠ʕʕʕ♠#ʕʕonʕ□1ʕʕ1 □ʕ□2no♠ʕ□$□□nrʕ¸o♠□{rʕ`□pʕNʕ□□
ccess Point is Creadted with SSID: ESP8266
ax Connections Allowed: 8
ccess Point IP: 192.168.4.1
ew Device Connected to SoftAP... Total Connections: 1
ew Device Connected to SoftAP... Total Connections: 2
```

The following are the instructions that we may send once they have been stored inside the web server or soft AP:

- Switch on the light.
- So Light is on.
- Turn on the light etc.

After submitting single button instructions through a web server, those instructions are transmitted to the access protocol, which analyses this server for those orders and, if they exist, transmits them to another web server, where the wi-fi module receives them and uses the soft AP protocol to execute them. As seen in the illustration below, the light is then switched on.

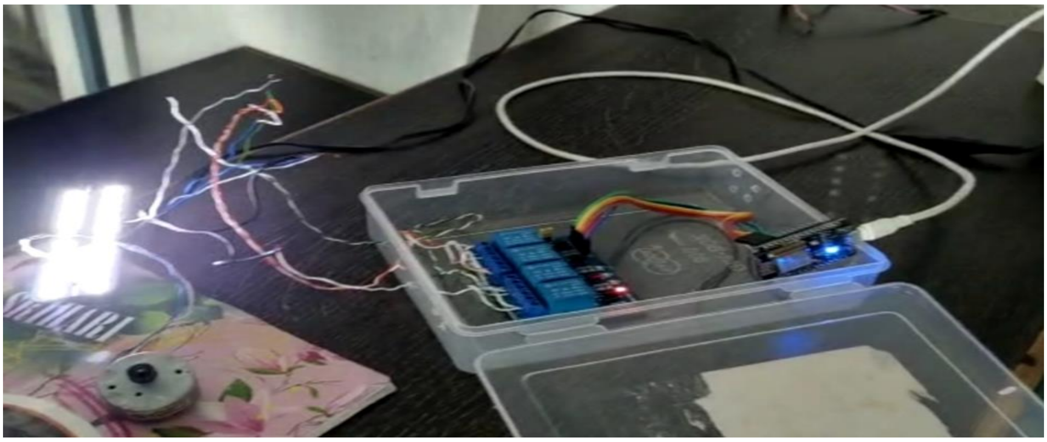
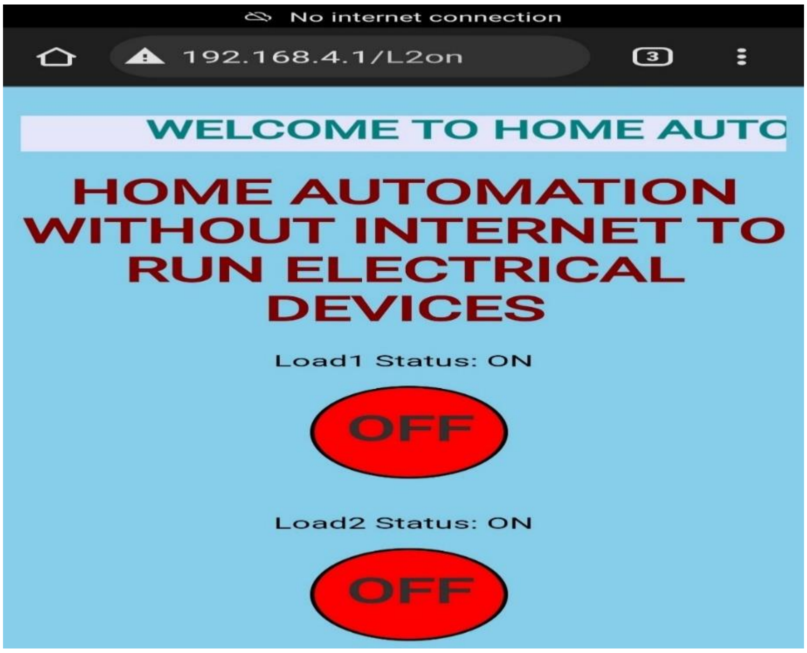


Figure 6.4: Light is turn on position



No internet connection

192.168.4.1/L2on

WELCOME TO HOME AUTO

HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL DEVICES

Load1 Status: ON

OFF

Load2 Status: ON

OFF

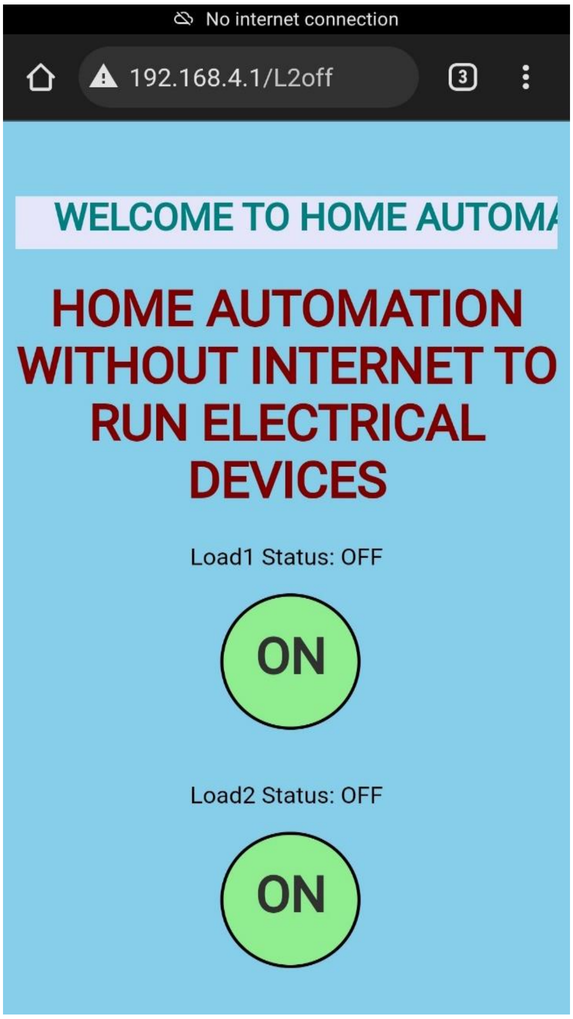
Figure 6.5: Soft AP based light turn on

6.3.2 LIGHT IS TURN OFF POSITION:

The following are the instructions that we may send once they have been stored inside the web server or soft AP:

- Switch the light off
- Light is turn off
- Turn the light off etc.

Once a single touch instruction has been transmitted through a web server, it is checked to see whether it already exists there before being transferred to the access protocol, which then transmits it to another web server, where the wi-fi module receives it and uses the soft AP protocol to carry it out. In the end, the light is switched off, as seen in the following figure.


No internet connection

192.168.4.1/L2off

WELCOME TO HOME AUTOMA

HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL DEVICES

Load1 Status: OFF

Load2 Status: OFF

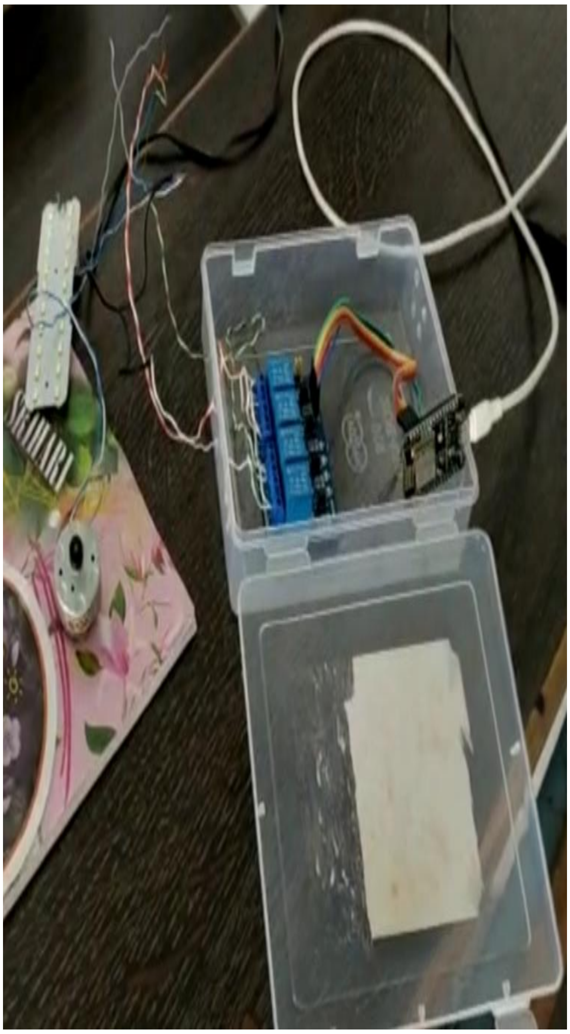


Figure 6.6: Soft AP based light turn off

Figure 6.7: Light is turn off position

CHAPTER 7
CONCLUSION

7.1 CONCLUSION:

In this project, we created and implemented a home automation system that can control power electrical appliances without the need for a network connection using the NodeMCU ESP8266. The user can transmit commands to the devices directly using this system, including switching on or off electrical appliances, giving the user a great deal of power and convenience.

This project's objective is to develop an open-source, easily configurable system for home automation that is both cost-effective and easy to use. Without an internet connection, the suggested solution offers a more flexible method of controlling household equipment. All age groups may easily utilize the suggested method, which is highly reliable and secure. It is also extremely beneficial for people who are physically active and think clearly, as staying at home becomes a smart challenge.

CHAPTER 8

FUTURE SCOPE

8.1 FUTURE SCOPE:

A key requirement for future homes that are sophisticated enough to provide inhabitants with the maximum degree of comfort this duty is referenced in the term "smart home” is becoming increasingly widespread. Virtual worlds, the incorporation of smart home technology, the development of more intelligent home equipment, and improvements in control, personalization, and performance are just a few of the possible advantages this technology may provide. a must-have features, Future housing will include services such as security, electricity, communication, healthcare, and entertainment. As we move closer to the next generation, more and more technology will be able to communicate with one another.

A bright future lies ahead for this system. Connected home systems have the potential to make houses much smarter in the future since the Internet of Everything is just getting started. Before turning on appliances and switching off the lights when not in use, make sure the house is occupied to conserve even more power. To provide consumers with additional privacy and security, the system may be seamlessly integrated with home security options.

The development of this technology for the automation of large-scale environments, including offices and factories, would be the next step. A typical benchmark for device suitability is provided by home automation. Smart houses may link to other networks and regulate lights, appliances, the environment, energy use, and security thanks to standardization. In the end, no system is ever perfect. It is continually under construction. Put on your thinking cap and try to improve the system gets, appliances, the environment, energy use, and security thanks to standardization. In the end, no system is ever perfect. It is continually under construction. Put on your thinking cap and try to improve the system that is all that is required.

8.2 ADVANTAGES:

Additionally, the use of wireless communication in the home and building automation systems offers several advantages that a cable connection cannot, including:

- **L**ess power consumption
- To reduced time

- Less cost
- Ease to use
- Long life
- To maximize home security

8.3 APPLICATIONS:

- Used In smart household appliances
- Useful for industrial applications
- It is incorporated into security systems.
- Used for including other, Offices, Schools, Hotels, Colleges, Banks

8.3.1 USED IN HOME AUTOMATION CIRCUIT DIAGRAM:

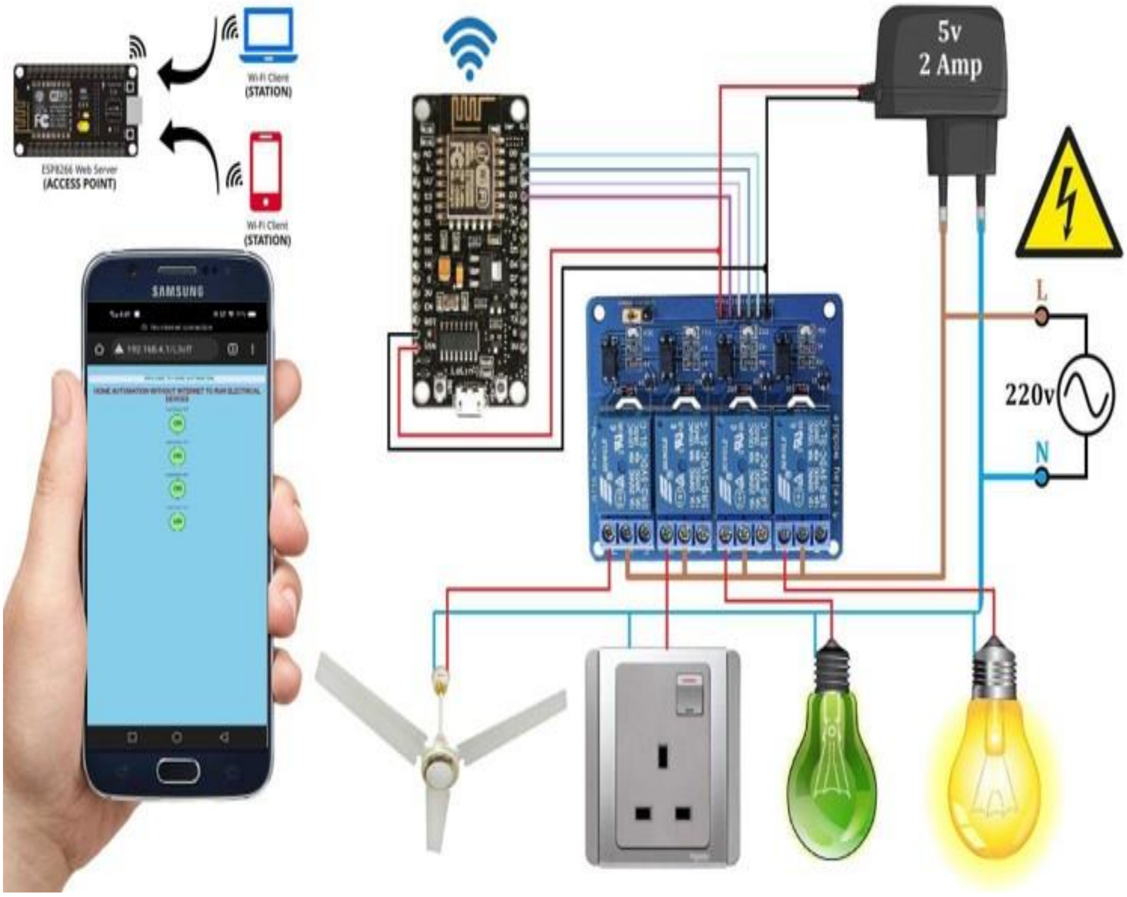


Figure 8.3.1: Automating the Home Using NodeMCU

Using a protocol, a smart device may manage a system for home automation. Lighting, fans, air conditioning systems, and security and surveillance locks are just a few of the household items that it can operate. Bluetooth or Wi-Fi technologies would be employed for remote system control. Technology has taken up a sizable part of our lives and has been cited by many as a source of concerning systems, and security and surveillance locks are just a few of the household items that it can operate. Most definitely! We live in a current age when smart and intelligent technologies are necessary to be present everywhere we are to make our daily lives simpler and considerably better, such as the capacity to accomplish many tasks faster, better, and more correctly.

8.3.2 USED FOR SECURITY SYSTEMS:

The fundamental idea behind all home surveillance systems is to protect access points, such as windows and doors, in addition to interior spaces that house valuables, such as artwork, computers, firearms, and coin collections. The only significant difference is the number of security components installed throughout the house and controlled by the control panel, regardless of the dimensions of your home, the number of windows and doors, or the number of interior rooms a homeowner chooses to secure.

The following are common components of a home security system:

- A control system that serves as the main controller of a home security system
- Sensors for the doors and windows
- Motion detectors inside and outside
- Wired and wireless surveillance cameras

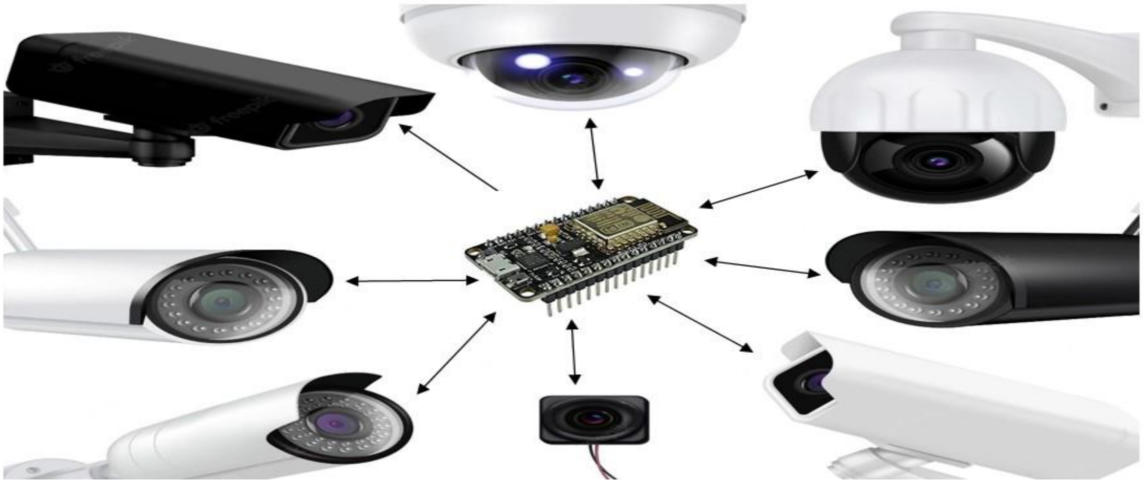


Figure 8.3.2: Utilizing NodeMCU for security (ESP8266)

BIBILOGRAPHY

[1] K. Mandala, R. Parupalli, C. A. Murty, E. Magesh, and R. Lunagariya, "Mobile based home automation using Internet of Things (IoT)," in 2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2015, pp. 340–343.

[2] "(PDF) Smart Home Automation Using IoT," ResearchGate. https://www.researchgate.net/publication/309855780_Smart_Home_Automation_Using_IoT (accessed Jun. 04, 2020).

[3] S. Singh, P. K. Sharma, S. Y. Moon, and J. H. Park, "i-SHSS: An IoT Based Smart Home Security System," in Advanced Multimedia and Ubiquitous Engineering, Springer, 2017, pp. 303–306.

[4] R. K. Kodali, V. Jain, S. Bose, and L. Boppana, "IoT based smart security and home automation system," in 2016 international conference on computing, communication and automation (ICCCA), 2016, pp. 1286–1289.

[5] M. L. Sharma, K. Sachin, and M. Nipuri, "Smart Home System Using IOT," Int. Res. J. Eng. Technol., vol. 4, no. 11, pp. 1108–1112, 2017.

[6] M. S. Soliman, A. A. Alahmadi, A. A. Maash, and M. O. Elhabib, "Design and Implementation of a Real-Time Smart Home Automation System Based on Arduino Microcontroller Kit and LabVIEW Platform," Int. J. Appl. Eng. Res., vol. 12, no. 18, pp. 7259–7264, 2017.

[7] L. M. Satapathy, S. K. Bastia, and N. Mohanty, "Arduino based home automation using Internet of things (IoT)," Int J Pure Appl Math, vol. 118, pp. 769–778, 2018.

[8] N. Hossain, M. A. Hossain, R. Sultana, and F. A. Lima, "A Security Framework for IOT based Smart Home Automation System," Glob. J. Comput. Sci. Technol., 2018.

[9] S. Kousalya, G. Reddi Priya, R. Vasanthi, and B. Venkatesh, "IOT based smart security and smart home automation," Int J Eng Res TechnolIJERT, vol. 7, no. 04, pp. 2278–0181, 2018.

[10] R. M. Zeebaree, B. W. Salim, R. R. Zebari, H. M. Shukur, A. S. Abdulraheem, A. I. Abdulla, S. M. Mohammed, "Enterprise Resource Planning Systems and Challenges," Technol. Rep. Kansai Univ., vol. 62, no. 4, pp. 1885–1894, Apr. 2020.

[11] H. Singh, V. Pallagani, V. Khandelwal, and U. Venkanna, “IoT based smart home automation system using sensor node,” in 2018 4th International Conference on Recent Advances in Information Technology (RAIT), 2018, pp. 1–5.

[12] S. K. Vishwakarma, P. Upadhyaya, B. Kumari, and A. K. Mishra, “Smart energy efficient home automation system using iot,” in 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), 2019, pp. 1–4.

[13] B. P. Kulkarni, A. V. Joshi, V. V. Jadhav, and A. T. Dhamange, “IoT based home automation using Raspberry PI,” Int J Innov Stud Sci Eng Technol, vol. 3, no. 4, pp. 13–16, 2017

[14] A. R. Al-Ali and M. Al-Rousan, "Java-based home automation system", IEEE Transactions on Consumer Electronics, vol. 50, no. 2, pp. 498- 504, 2004.

[15] N. Sriskanthan, F. Tan and A. Karande, "Bluetooth based home automation system"

1. https://www.irjet.net/archives/internet-of-things-iot-based-home-automation-system-has-implementation-of-real-time-experiment-IJERTCONV9IS11048.pdf.pdf
2. https://www.irjet.net/archives/V4/i10/IRJET-V4I1061.pdf
3. https://hmautomate.com/the-practicality-of-home automation-without-internet/
4. https://www.researchgate.net/publication/230883124_Design_and_Implementation_of_a_WiFi_Based_Home_Automation_System
5. A ZigBee-based home automation system | IEEE Journals & Magazine | IEEE Xplore
6. https://projects.digilentinc.com/embeddedlab786/iot-based-home-automation-project-9ba396
7. https://iotcircuithub.com/esp32-home-automation-using-alexa/

APPENDICES

APPENDIX A- PROJECT CODE

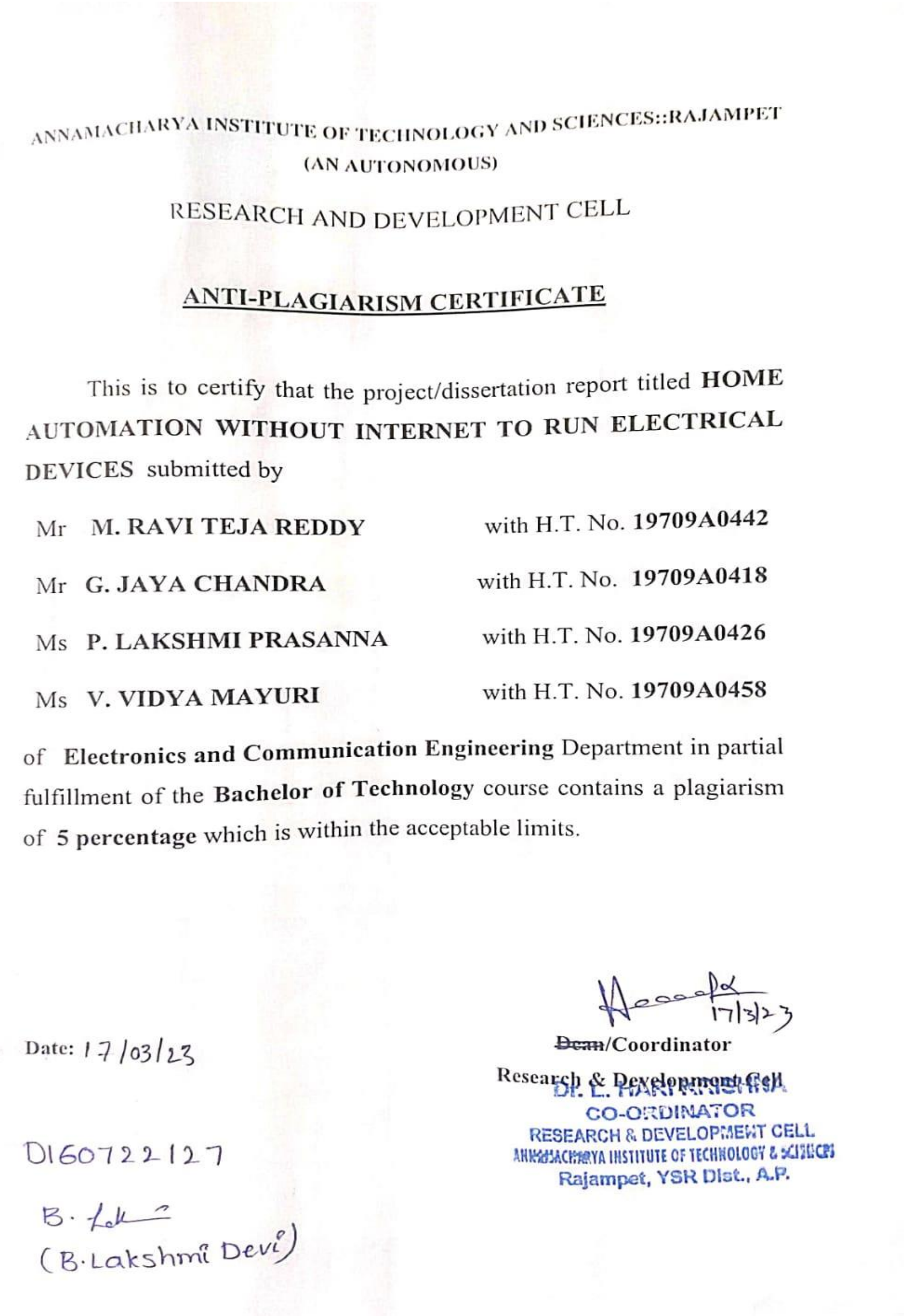
```
#include <WiFi.h>
#include <ESP8266WiFi.h>
#include <EEPROM.h>
#include <ESP8266WebServer.h>
const char* userid = "PROJECT E3”; // Enter SSID here
const char* password = "E3PROJECT”; //Enter Password here
ESP8266WebServer server (80);
Uint16_t R1 = D0;
Uint16_t R2 = D1;
Uint16_t R3 = D2;
Uint16_t R4 = D3;
int l1, l2, l3, l4;
void setup () {
  Serial.begin(115200);
  Serial.println();
  pinMode (R1, OUTPUT);
  pinMode (R2, OUTPUT);
  pinMode (R3, OUTPUT);
  pinMode (R4, OUTPUT);
EEPROM.begin(512);
  l1 = EEPROM.read(1);
  l2 = EEPROM.read(2);
  l3 = EEPROM.read(3);
  l4 = EEPROM.read(4);
  Serial.print(l1); Serial.print(" ");
  Serial.print(l2); Serial.print(" ");
  Serial.print(l3); Serial.print(" ");
  Serial.println(l4);
  digitalWrite (R1, l1);
  digitalWrite (R2, l2);
  digitalWrite (R3, l3);
  digitalWrite (R4, l4);
  WiFi.softAP(userid, password);
  IPAddress protocol = WiFi.softAPIP();
  Serial.print("visit: \n");
  Serial.println(protocol);
  delay (100);
  server. on ("/", OnTag);
  server. on ("/L1on”, L1on);
  server. on ("/L1off", L1off);
  server. on ("/L2on”, L2on);
  server. on ("/L2off", L2off);
  server. on ("/L3on”, L3on);
  server. on ("/L3off", L3off);
  server. on ("/L4on”, L4on);
  server. on ("/L4off", L4off);
  server. onNotFound (Notset);
  server. begin ();
  Serial.println("Protocol begin");
}
void loop () {
  server. handleClient ();
  digitalWrite (R1, l1);
  digitalWrite (R2, l2);
  digitalWrite (R3, l3);
  digitalWrite (R4, l4);
}
void OnTag () {
  server. send (200, "text/html", SendHTML (l1, l2, l3, l4));
}
void L1on () {l1=0; data ();}
void L1off () {l1=1; data ();}
void L2on () {l2=0; data ();}
void L2off () {l2=1; data ();}
void L3on () {l3=0; data ();}
void L3off () {l3=1; data ();}
void L4on () {l4=0; data ();}
void L4off () {l4=1; data ();}
void data () {
EEPROM.write(1, l1);
EEPROM.write(2, l2);
EEPROM.write(3, l3);
EEPROM.write(4, l4);
EEPROM.commit();
server. send (200, "text/html", SendHTML (l1, l2, l3, l4));
}
void Notset () {
  server. send (404, "text/plain", "Error");
}
```

```
String SendHTML (uint16_t l1r1, uint16_t l2r2, uint16_t l3r3, uint16_t l4r4){
  String ptr = "<!DOCTYPE html> <html>\n";
  ptr +="<head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1.0, user-scalable=no\">\n";
  ptr +="<title>Home Automation</title>\n";
  ptr +="<style>html {background-color: skyblue; font-family: Helvetica; display: inline-block; margin: 0px auto; text-align: center;} \n";
  ptr +="body {margin-top: 50px;} \n";
  ptr += “. circle {width: 85px; height: 85px; border-radius: 50px;background: Red;}\n";
  ptr += “. circle1 {width: 85px; height: 85px;border-radius: 50px;background: lightgreen;}\n";
  ptr +="p {font-size:15px; text-align: center;} \n";
  ptr +="div {display: inline-block;margin-left:5px;height:100px;width:100px;border-radius:100px;border:2px solid black;}\n";
  ptr +="a {font-size:32px; color: rgb(50,50,50);text-decoration: none; text-align:center;}\n";
  ptr +="</style>\n";
  ptr +="</head>\n";
  ptr +="<body>\n";
  ptr +="<h2 style=\"background-color: Lavender\"><marquee style=\"color:teal\" font-family:\"Times New Roman\">WELCOME TO HOME AUTOMATION</marquee></h2>\n";
  ptr +="<center style=\"color: maroon\"><h1 font-family: \"Times New Roman\">HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL DEVICES</center></center>\n";
   if(l1r1==0)
  {ptr +="<p>Load1 Status: ON</p><div class=\"circle\"><h1><a class=\"circle\" href=\"/L1off\">OFF</a></h1></div><br><br>\n";}
  else
```

```
 {ptr +="<p>Load1 Status: OFF</p><div class=\"circle1\"><h1><a class=\"circle1\"
href=\"/L1on\">ON</a></h1></div><br><br>\n";}
 if(l2r2==0)
 {ptr +="<p>Load2 Status: ON</p><div class=\"circle\"><h1><a class=\"circle\"
href=\"/L2off\">OFF</a></h1></div><br><br>\n";}
 else
 {ptr +="<p>Load2 Status: OFF</p><div class=\"circle1\"><h1><a class=\"circle1\"
href=\"/L2on\">ON</a></h1></div><br><br>\n";}
 if(l3r3==0)
 {ptr +="<p>Load3 Status: ON</p><div class=\"circle\"><h1><a class=\"circle\"
href=\"/L3off\">OFF</a></h1></div><br><br>\n";}
 else
 {ptr +="<p>Load3 Status: OFF</p><div class=\"circle1\"><h1><a class=\"circle1\"
href=\"/L3on\">ON</a></h1></div><br><br>\n";}
 if(l4r4==0)
 {ptr +="<p>Load4 Status: ON</p><div class=\"circle\"><h1><a class=\"circle\"
href=\"/L4off\">OFF</a></h1></div><br><br>\n";}
 else
 {ptr +="<p>Load4 Status: OFF</p><div class=\"circle1\"><h1><a class=\"circle1\"
href=\"/L4on\">ON</a></h1></div><br><br>\n";}
 ptr +="</body>\n";
 ptr +="</html>\n";
 return ptr;
}
```

APPENDIX-B

ANTI PLAGIARISM CERTIFICATE



ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES::RAJAMPET

(AN AUTONOMOUS)

RESEARCH AND DEVELOPMENT CELL

ANTI-PLAGIARISM CERTIFICATE

This is to certify that the project/dissertation report titled **HOME AUTOMATION WITHOUT INTERNET TO RUN ELECTRICAL DEVICES** submitted by

Mr	**M. RAVI TEJA REDDY**	with H.T. No. **19709A0442**
Mr	**G. JAYA CHANDRA**	with H.T. No. **19709A0418**
Ms	**P. LAKSHMI PRASANNA**	with H.T. No. **19709A0426**
Ms	**V. VIDYA MAYURI**	with H.T. No. **19709A0458**

of **Electronics and Communication Engineering** Department in partial fulfillment of the **Bachelor of Technology** course contains a plagiarism of **5 percentage** which is within the acceptable limits.

17/3/23

~~Dean~~/Coordinator

Research & Development Cell

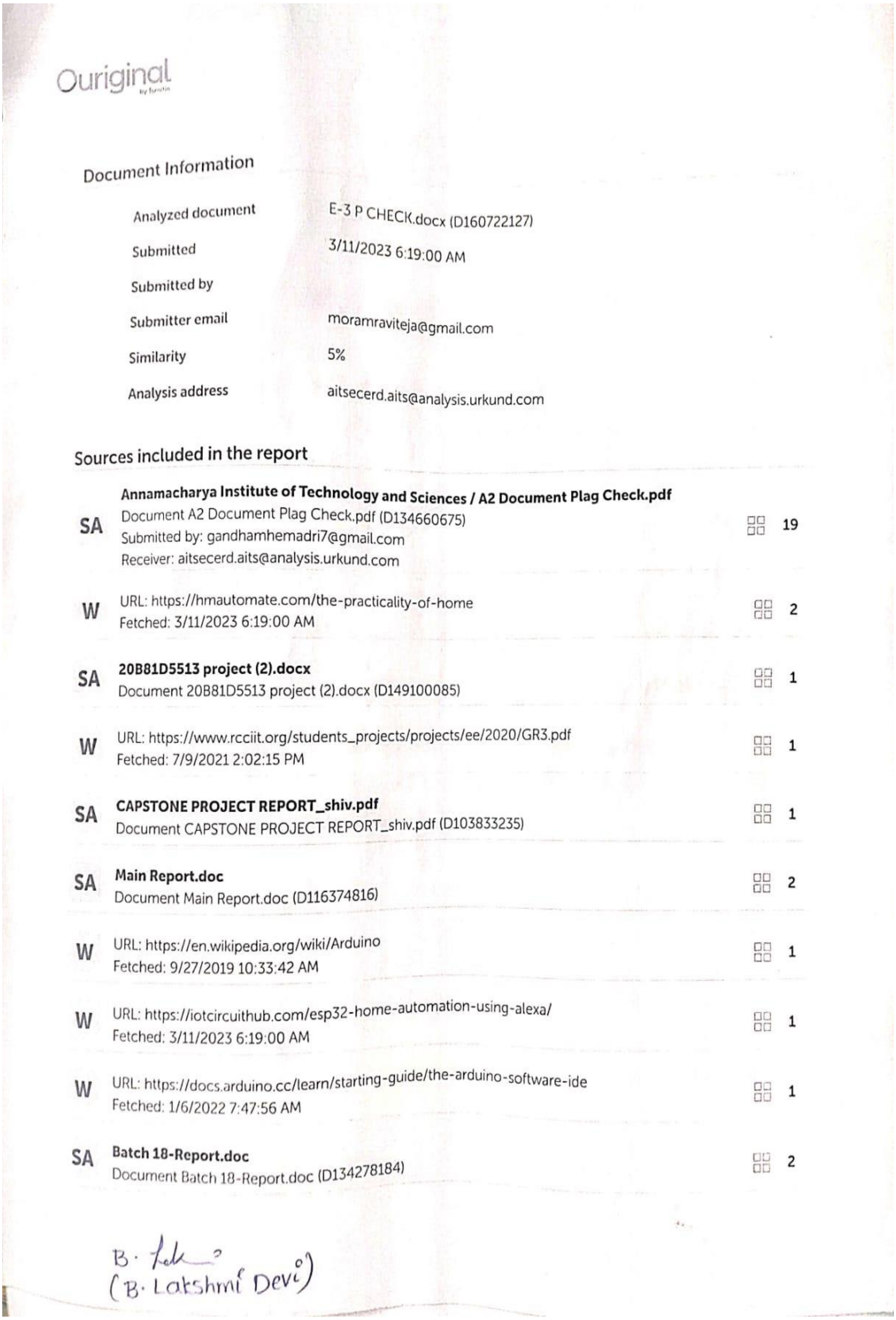
Dr. L. HARI KRISHNA
CO-ORDINATOR
RESEARCH & DEVELOPMENT CELL
ANNAMACHARYA INSTITUTE OF TECHNOLOGY & SCIENCES
Rajampet, YSR Dist., A.P.

Date: 17/03/23

D160722127

B. Lak

(B. Lakshmi Devi)



Ouriginal

Document Information

Analyzed document	E-3 P CHECK.docx (D160722127)
Submitted	3/11/2023 6:19:00 AM
Submitted by	
Submitter email	moramraviteja@gmail.com
Similarity	5%
Analysis address	aitsecerd.aits@analysis.urkund.com

Sources included in the report

SA	**Annamacharya Institute of Technology and Sciences / A2 Document Plag Check.pdf** Document A2 Document Plag Check.pdf (D134660675) Submitted by: gandhamhemadri7@gmail.com Receiver: aitsecerd.aits@analysis.urkund.com	19
W	URL: https://hmautomate.com/the-practicality-of-home Fetched: 3/11/2023 6:19:00 AM	2
SA	**20B81D5513 project (2).docx** Document 20B81D5513 project (2).docx (D149100085)	1
W	URL: https://www.rcciit.org/students_projects/projects/ee/2020/GR3.pdf Fetched: 7/9/2021 2:02:15 PM	1
SA	**CAPSTONE PROJECT REPORT_shiv.pdf** Document CAPSTONE PROJECT REPORT_shiv.pdf (D103833235)	1
SA	**Main Report.doc** Document Main Report.doc (D116374816)	2
W	URL: https://en.wikipedia.org/wiki/Arduino Fetched: 9/27/2019 10:33:42 AM	1
W	URL: https://iotcircuithub.com/esp32-home-automation-using-alexa/ Fetched: 3/11/2023 6:19:00 AM	1
W	URL: https://docs.arduino.cc/learn/starting-guide/the-arduino-software-ide Fetched: 1/6/2022 7:47:56 AM	1
SA	**Batch 18-Report.doc** Document Batch 18-Report.doc (D134278184)	2

B. Lak
(B. Lakshmi Devi)