IoT BASED ESP8266 SWITCHING USING GOOGLE ASSISTANT, ALEXA & CADIO

A Project Report

Submitted by

ABHINAV BAGHEL

(Roll no:20EI01)

LOKESH

(Roll no:20EI23)

SHIVA PAL

(Roll no:20EI38)

In partial fulfillment for the award of the degree

of

Bachelor of Technology

in

Electronics & Instrumentation Engineering

चरैवेति चरैवेति

महात्मा ज्योतिबा फुले
रुहेलखण्ड विश्वविद्यालय, बरेली

Department of Electronics and Instrumentation Engineering

Faculty of Engineering and Technology

MJP ROHILKHAND UNIVERSITY, BAREILLY

DECEMBER, 2023

Extended Abstract

The advent of the Internet of Things (IoT) has revolutionized the way we interact with our living spaces, giving rise to the concept of IoT-based home automation. This extended abstract provides a comprehensive overview of the aims, conclusions, and implications of IoT-based home automation, exploring its potential to enhance convenience, efficiency, and security in modern households.

The primary aim of IoT-based home automation is to create intelligent, interconnected systems that enable seamless communication between various devices and appliances within a household. By integrating smart sensors, actuators, and communication protocols, the goal is to empower homeowners with centralized control over their domestic environments. This includes the ability to remotely monitor and manage lighting, heating, cooling, security systems, and other essential components, thereby optimizing energy consumption and improving overall quality of life.

IoT-based home automation has demonstrated significant advancements in achieving its stated aims. The integration of sensors and smart devices allows for real-time data collection and analysis, enabling automation systems to respond dynamically to changing conditions. For instance, smart thermostats can learn user preferences and adjust temperature settings accordingly, while intelligent lighting systems can adapt to natural light conditions. Moreover, the incorporation of machine learning algorithms enhances the system's ability to predict user behaviour, further optimizing energy usage and increasing efficiency. The integration of voice-activated assistants and mobile applications has facilitated user-friendly interfaces, making it more accessible for homeowners to interact with their automated systems. This user-centric approach enhances the overall user experience, contributing to the widespread adoption of IoT-based home automation.

The implications of IoT-based home automation extend beyond the realm of convenience. From a sustainability standpoint, the optimized energy usage facilitated by intelligent automation contributes to a reduction in overall energy consumption, thereby lowering the environmental impact of households. Additionally, the enhanced security features, such as smart surveillance systems and automated door locks, bolster home safety, providing homeowners with peace of mind.

Acknowledgement

We would like to express my sincere gratitude to all those who have contributed to the successful completion of this IoT-based home automation project.

First and foremost, we extend our deepest appreciation to our Project Guide Dr. Anil Kumar Singh, whose guidance and expertise played a crucial role in shaping the direction of this project. Your insightful feedback and continuous support have been invaluable throughout the entire journey.

We are immensely thankful to Department of Electronics & Instrumentation Engineering for providing the necessary resources, including hardware components and software tools, which were instrumental in the implementation of the home automation system

We extend our thanks to our teammates who worked tirelessly to bring this project to fruition. Your dedication, creativity, and collaborative spirit have been essential in overcoming challenges and achieving our goals.

Last but not least, we want to express our gratitude to the broader community and researchers in the field of IoT and home automation. The wealth of knowledge and open exchange of ideas in this community have been influential in shaping the conceptual framework and implementation strategies of this project.

This project would not have been possible without the collective effort and support from everyone mentioned above. Thank you for being an integral part of this endeavour.

Abhinav Baghel (20EI01)

Lokesh (20EI23)

Shiva Pal (20EI38)

TABLE OF CONTENT

Introduction

In recent years, the rapid evolution of technology has ushered in a new era of smart living, transforming traditional homes into intelligent and interconnected spaces. One of the key enablers of this revolution is the Internet of Things (IoT), a paradigm that connects everyday devices to the internet, allowing them to communicate, collect, and exchange data. Home automation, a subset of IoT, leverages this connectivity to enhance the control and monitoring of various household appliances and systems, making homes more efficient, convenient, and secure.[1] The 21st century witnessed an exponential growth in the field, with the proliferation of smartphones leading to the development of mobile app-controlled devices and systems. The introduction of voice assistants further revolutionized home automation, allowing users to control various aspects of their homes through voice commands.

Automation has entered a new era with the integration of IoT-based devices like the ESP8266 and voice command technology such as Amazon's Alexa. The ESP8266, a versatile microcontroller with Wi-Fi capabilities, forms the backbone of smart home systems, enabling seamless connectivity and control. The ESP8266 is a popular and versatile microcontroller with built-in Wi-Fi capabilities, making it a common choice for Internet of Things (IoT) applications. One of the common use cases for the ESP8266 is remote switching and control. By leveraging Alexa's voice recognition and natural language processing, users can effortlessly manage various smart devices within their homes.[2] From adjusting lights and thermostats to controlling entertainment systems, this synergy between IoT and voice command exemplifies the remarkable strides made in creating intuitive and interconnected living spaces.

Here we have made a simple IoT based esp8266 switching using Cadio. To make this project, we don’t have to write any code, we just need to upload the Cadio format on Esp8266, then configure it through their mobile App. After that we can control all our appliance through Cadio & voice command even if there is no internet connection. We have used Esp8266 because it is a versatile microcontroller renowned for its compact size and impressive. Its low-power modes optimize energy consumption, making it ideal for various applications. This microcontroller operates between 3.0V and 3.6V, simplifying integration with different power sources. It can be programmed using popular development environments like Arduino IDE and supports multiple operating modes.

1.1 Background

The concept of “Home Automation” has been in existence for several years. “Smart Home”, “Intelligent Home” are terms that followed and is been used to introduce the concept of networking appliance within the house. Home Automation Systems (HASs) includes centralized control and distance status monitoring of lighting, security system, and other appliances and systems within a house. HASs enables energy efficiency, improves the security systems, and certainly the comfort and ease of users. In the present emerging market, HASs is gaining popularity and has attracted the interests of many users. HASs comes with its own challenges. Mainly being, in the present day, end users especially elderly and disabled, even though hugely benefited, aren’t seen to accept the system due to the complexity and cost factors.

1.2 Project Objectives

Design of an independent HAS

To formulate the design of an interconnected network of home appliance to be integrated into the HAS. The objective to account for every appliance and its control to be automated and integrated into the network further formulated into the HAS.

Wireless control of home appliances (Switch and Voice mode)

To develop the application that would include features of switch and/or voice modes to control the applications.

Secure connection channels between application and Node MCU

Use of secure protocols over Wi-Fi so that other devices are prevented to achieve control over the HAS. Secure connections are obtained by SSL over TCP, SSH.

Controlled by any device capable of Wi-Fi (Android, iOS, PC)

To achieve flexibility in control of the home appliances, and device capable of Wi-Fi connectivity will be able to obtain a secure control on the HAS.

Extensible platform for future enhancement

With a strong existing possibility of adding and integrating more features and appliances to the system, the designed system needs to be highly extensible in nature.

1.3 Scope

The aim is to design a prototype that establishes wireless remote control over a network of home appliances. The application is designed to run on android device providing features like, switch mode control, voice command control and a provision to view the status of the devices on the application itself. Considering its wide range of application, following are the scope of this prototype. The system can be implemented in homes, small offices and malls as well, being in-charge of control of the electrical appliances. For remote access of appliances in internet or intranet. The appliances in the above-mentioned environment can be controlled in intra-network or can be accessed via internet. The development of technology friendly environment. The system incorporates the use of technology and making HAS. By the use of day-to-day gadgets, we can utilize them for a different perspective.

1.4 Overview & Benefits

The benefits of an established wireless remote switching system of home appliances include:

Reduced wiring issues

Considering the increase in price of copper, thus increases the possibility of the wire to be stolen. The use of a wireless remote system to control home appliances means no wire for thieves to steal.

Extended range

As the system establishes control over Wi-Fi, it was a generally considered descent range. That is 150 feet indoors. Outdoors it can be extended to 300 feet, but since the application is of a HAS, an indoor range is considered.

Security

As the connection of the control of the HAS is established over a secure network the system ensures security to the maximum extent.

Theory

3.1 IoT (Internet of Things)

IOT as a term has evolved long way as a result of convergence of multiple technologies, machine learning, embedded systems and commodity sensors. IOT is a system of interconnected devices assigned a UIDS, enabling data transfer and control of devices over a network. It reduced the necessity of actual interaction in order to control a device. IOT is an advanced automation and analytics system which exploits networking, sensing, big data, and artificial intelligence technology to deliver complete systems for a product or service. These systems allow greater transparency, control, and performance when applied to any industry or system.

3.1.1 Features of IOT

3.1.1.1 Intelligence

IOT comes with the combination of algorithms and computation, software & hardware that makes it smart. Ambient intelligence in IOT enhances its capabilities which facilitate the things to respond in an intelligent way to a particular situation and supports them in carrying out specific tasks. In spite of all the popularity of smart technologies, intelligence in IOT is only concerned as a means of interaction between devices, while user and device interaction are achieved by standard input methods and graphical user interface

3.1.1.2 Connectivity

Connectivity empowers the Internet of Things by bringing together everyday objects. Connectivity of these objects is pivotal because simple object level interactions contribute towards collective intelligence in the IOT network. It enables network accessibility and compatibility in the things. With this connectivity, new market opportunities for the Internet of things can be created by the networking of smart things and applications

3.1.1.3 Dynamic Nature

The primary activity of Internet of Things is to collect data from its environment, this is achieved with the dynamic changes that take place around the devices. The state of these devices change dynamically, example sleeping and waking up, connected and/or disconnected as well as the context of devices including temperature, location and speed. In addition to the state of the device, the number of devices also changes dynamically with a person, place and time

3.1.1.4 Enormous Scale

The number of devices that need to be managed and that communicate with each other will be much larger than the devices connected to the current Internet. The management of data generated from these devices and their interpretation for application purposes becomes more critical. Gartner (2015) confirms the enormous scale of IOT in the estimated report where it stated that 5.5 million new things will get connected every day and 6.4 billion connected things will be in use worldwide in 2016, which is up by 30 percent from 2015

3.1.1.5 Sensing

IOT wouldn't be possible without sensors that will detect or measure any changes in the environment to generate data that can report on their status or even interact with the environment. Sensing technologies provide the means to create capabilities that reflect a true awareness of the physical world and the people in it. The sensing information is simply the analog input from the physical world, but it can provide a rich understanding of our complex world

3.1.1.6 Heterogeneity

Heterogeneity in Internet of Things as one of the key characteristics. Devices in IOT are based on different hardware platforms and networks and can interact with other devices or service platforms through different networks. IOT architecture should support direct network connectivity between heterogeneous networks. The key design requirements for heterogeneous things and their environments in IOT are scalabilities, modularity, extensibility and interoperability.

3.1.1.7 Security

IOT devices are naturally vulnerable to security threats. As we gain efficiencies, novel experiences, and other benefits from the IOT, it would be a mistake to forget about security concerns associated with it. There is a high level of transparency and privacy issues with IOT. It is important to secure the endpoints, the networks, and the data that is transferred across all of it means creating a security paradigm.

3.1.2 Advantages of IOT

3.1.2.1 Communication

IOT encourages the communication between devices, also famously known as Machine-to-Machine (M2M) communication. Because of this, the physical devices are able to stay connected and hence the total transparency is available with lesser inefficiencies and greater quality.

3.1.2.2 Automation and Control

Due to physical objects getting connected and controlled digitally and centrally with wireless infrastructure, there is a large amount of automation and control in the workings. Without human intervention, the machines are able to communicate with each other leading to faster and timely output.

3.1.2.3 Information

It is obvious that having more information helps making better decisions. Whether it is mundane decisions as needing to know what to buy at the grocery store or if your company has enough widgets and supplies, knowledge is power and more knowledge is better.

3.1.2.4 Monitor

The second most obvious advantage of IOT is monitoring. Knowing the exact quantity of supplies or the air quality in your home, can further provide more information that could not have previously been collected easily. For instance, knowing that you are low on milk or printer ink could save you another trip to the store in the near future. Furthermore, monitoring the expiration of products can and will improve safety.

3.1.2.5 Time

As hinted in the previous examples, the amount of time saved because of IOT could be quite large. And in today's modern life, we all could use more time.

3.1.2.6 Money

The biggest advantage of IOT is saving money. If the price of the tagging and monitoring equipment is less than the amount of money saved, then the Internet of Things will be very widely adopted. IOT fundamentally proves to be very helpful to people in their daily routines by making the appliances communicate to each other in an effective manner thereby saving and conserving energy and cost. Allowing the data to be communicated and shared between devices and then translating it into our required way, it makes our systems efficient.

3.1.2.7 Automation of daily tasks leads to better monitoring of devices

The IOT allows you to automate and control the tasks that are done on a daily basis, avoiding human intervention. Machine-to-machine communication helps to maintain transparency in the processes. It also leads to uniformity in the tasks. It can also maintain the quality of service. We can also take necessary action in case of emergencies.

3.1.2.8 Efficient and Saves Time

The machine-to-machine interaction provides better efficiency, hence; accurate results can be obtained fast. This results in saving valuable time. Instead of repeating the same tasks every day, it enables people to do other creative jobs.

3.1.2.9 Saves Money

Optimum utilization of energy and resources can be achieved by adopting this technology and keeping the devices under surveillance. We can be alerted in case of possible bottlenecks, breakdowns, and damages to the system. Hence, we can save money by using this technology.

3.1.2.10 Better Quality of Life

All the applications of this technology culminate in increased comfort, convenience, and better management, thereby improving the quality of life.

3.1.2.11 Industrial Automation

In industrial settings, the ESP8266 can control and monitor machines, motors, and other equipment remotely. This can help streamline operations, improve efficiency, and reduce downtime.

3.1.3 Disadvantages of IOT

3.1.3.1 Compatibility

Currently, there is no international standard of compatibility for the tagging and monitoring equipment. I believe this disadvantage is the most easy to overcome. The manufacturing companies of these equipment just need to agree to a standard, such as Bluetooth, USB, etc. This is nothing new or innovative needed.

3.1.3.2 Complexity

As with all complex systems, there are more opportunities of failure. With the Internet of Things, failures could sky rocket. For instance, let's say that both you and your spouse each get a message saying that your milk has expired, and both of you stop at a store on your way home, and you both purchase milk. As a result, you and your spouse have purchased twice the amount that you both need. Or maybe a bug in the software ends up automatically ordering a new ink cartridge for your printer each and every hour for a few days, or at least after each power failure, when you only need a single replacement.

3.1.3.3 Privacy/Security

With all of this IOT data being transmitted, the risk of losing privacy increases. For instance, how well encrypted will the data be kept and transmitted with? Do you want your neighbours or employers to know what medications that you are taking or your financial situation?

3.1.3.4 Safety

Imagine if a notorious hacker changes your prescription. Or if a store automatically ships you an equivalent product that you are allergic to, or a flavour that you do not like, or a product that is already expired. As a result, safety is ultimately in the hands of the consumer to verify any and all automation.

As all the household appliances, industrial machinery, public sector services like water supply and transport, and many other devices all are connected to the Internet, a lot of information is available on it. This information is prone to attack by hackers. It would be very disastrous if private and confidential information is accessed by unauthorized intruders.

3.1.3.5 Lesser Employment of Menial Staff

The unskilled workers and helpers may end up losing their jobs in the effect of automation of daily activities. This can lead to unemployment issues in the society. This is a problem with the advent of any technology and can be overcome with education. With daily activities getting automated, naturally, there will be fewer requirements of human resources, primarily, workers and less educated staff. This may create Unemployment issue in the society.

3.1.3.6 Technology Takes Control of Life

Our lives will be increasingly controlled by technology, and will be dependent on it. The younger generation is already addicted to technology for every little thing. We have to decide how much of our daily lives are we willing to mechanize and be controlled by technology.

3.1.4 Application Grounds of IOT

3.1.4.1 Wearables

Wearable technologies is a hallmark of IOT applications and is one of the earliest industries to have deployed IOT at its services. Fit Bits, heart rate monitors, smartwatches, glucose monitoring devices reflect the successful applications of IOT.

3.1.4.2 Smart homes

This area of application concerned to this particular project, so a detailed application is discussed further. *Jarvis*, an AI home automation employed by Mark Zuckerberg, is a remarkable example in this field of application.

3.1.4.3 Health care

IOT applications have turned reactive medical based system into proactive wellness-based system. IOT focuses on creating systems rather than equipment. IOT creates a future of medicine and healthcare which exploits a highly integrated network of sophisticated medical devices. The integration of all elements provides more accuracy, more attention to detail, faster reactions to events, and constant improvement while reducing the typical overhead of medical research and organizations

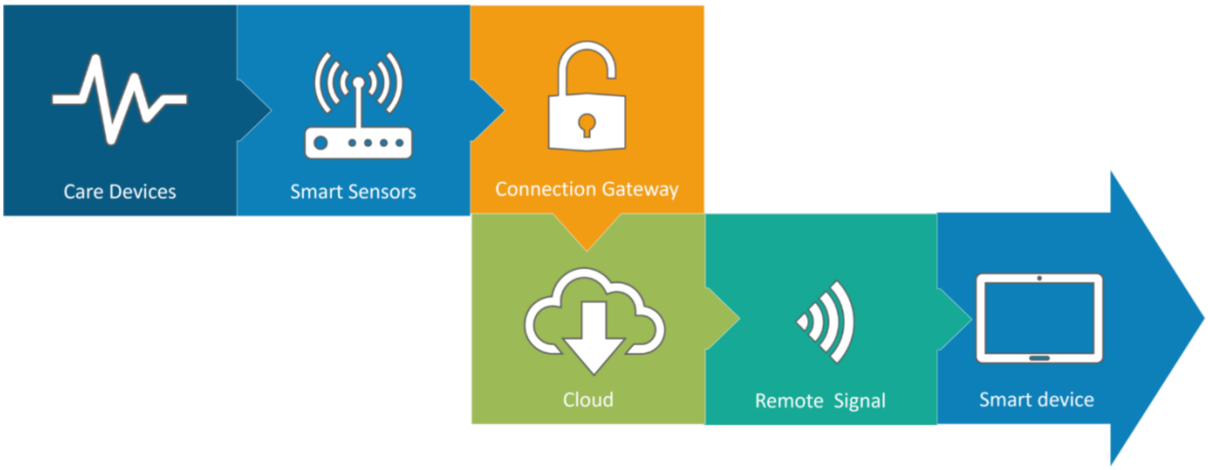


Figure 1. Working of IOT enables care devices.

3.1.4.4 Agriculture

A greenhouse farming technique enhances the yield of crops by controlling environmental parameters. However, manual handling results in production loss, energy loss, and labour cost, making the process less effective. A greenhouse with embedded devices not only makes it easier to be monitored but also, enables us to control the climate inside it. Sensors measure different parameters according to the plant requirement and send it to the cloud. It, then, processes the data and applies a control action.

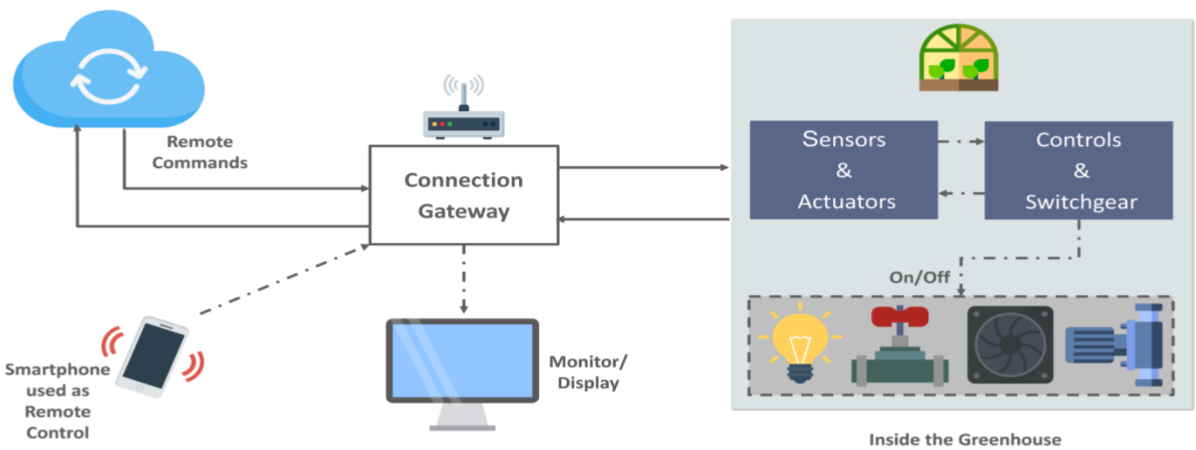


Figure 2. IOT controlled greenhouse environment.

3.1.4.5 Industrial Automation

For a higher return of investment this field requires both fast developments and quality of products. This vitality thus coined the term IIOT. This whole schematic is re-engineered by IOT applications. Following are the domains of IOT applications in industrial automation

- Factory Digitalization
- Product flow Monitoring
- Inventory Management
- Safety and Security
- Quality Control
- Packaging optimization
- Logistics and Supply Chain Optimization

3.1.4.6 Government and Safety

IOT applied to government and safety allows improved law enforcement, defence, city planning, and economic management. The technology fills in the current gaps, corrects many current flaws, and expands the reach of these efforts. For example, IOT can help city planners have a clearer view of the impact of their design, and governments have a better idea of the local economy.

3.1.5 IOT Technologies and Protocols

Several communication protocols and technologies cater to and meet the specific functional requirements of IOT system.

3.1.5.1 Bluetooth

Bluetooth is a short range IOT communication protocol/technology that is profound in many consumer product markets and computing. It is expected to be key for wearable products in particular, again connecting to the IOT albeit probably via a smartphone in many cases. The new Bluetooth Low-Energy (BLE) – or Bluetooth Smart, as it is now branded – is a significant protocol for IOT applications. Importantly, while it offers a similar range to Bluetooth it has been designed to offer significantly reduced power consumption.

3.1.5.2 Zigbee

ZigBee is similar to Bluetooth and is majorly used in industrial settings. It has some significant advantages in complex systems offering low-power operation, high security, robustness and high and is well positioned to take advantage of wireless control and sensor networks in IOT applications. The latest version of ZigBee is the recently launched 3.0, which is essentially the unification of the various ZigBee wireless standards into a single standard.

3.1.5.3 Z-Wave

Z-Wave is a low-power RF communications IOT technology that primarily design for home automation for products such as lamp controllers and sensors among many other devices. A ZWave uses a simpler protocol than some others, which can enable faster and simpler development, but the only maker of chips is Sigma Designs compared to multiple sources for other wireless technologies such as ZigBee and others.

3.1.5.4 Wi-Fi

Wi-Fi connectivity is one of the most popular IOT communication protocol, often an obvious choice for many developers, especially given the availability of Wi-Fi within the home environment within LANs. There is a wide existing infrastructure as well as offering fast data transfer and the ability to handle high quantities of data. Currently, the most common Wi-Fi standard used in homes and many businesses is 802.11n, which offers range of hundreds of megabits per second, which is fine for file transfers but may be too power-consuming for many IOT applications.

3.1.5.5 Cellular

Any IOT application that requires operation over longer distances can take advantage of GSM/3G/4G cellular communication capabilities. While cellular is clearly capable of sending high quantities of data, especially for 4G, the cost and also power consumption will be too high for many applications. But it can be ideal for sensor-based low-bandwidth-data projects that will send very low amounts of data over the Internet

3.1.5.6 LoRa WAN

LoRaWAN is one of popular IOT Technology, targets wide-area network (WAN) applications. The LoRa WAN design to provide low-power WANs with features specifically needed to support low-cost mobile secure communication in IOT, smart city, and industrial applications. Specifically meets requirements for low-power consumption and supports large networks with millions and millions of devices, data rates range from 0.3 kbps to 50 kbps.

3.1.6 IOT Software

IOT software addresses its key areas of networking and action through platforms, embedded systems, partner systems, and middleware. These individual and master applications are responsible for data collection, device integration, real-time analytics, and application and process extension within the IOT network. They exploit integration with critical business systems (e.g., ordering systems, robotics, scheduling, and more) in the execution of related tasks.

3.1.6.1 Data Collection

This software manages sensing, measurements, light data filtering, light data security, and aggregation of data. It uses certain protocols to aid sensors in connecting with real-time, machine to-machine networks. Then it collects data from multiple devices and distributes it in accordance with settings. It also works in reverse by distributing data over devices. The system eventually transmits all collected data to a central server.

3.1.6.2 Device Integration

Software supporting integration binds (dependent relationships) all system devices to create the body of the IOT system. It ensures the necessary cooperation and stable networking between devices. These applications are the defining software technology of the IOT network because without them, it is not an IOT system. They manage the various applications, protocols, and limitations of each device to allow communication.

3.1.6.3 Real-Time Analytics

These applications take data or input from various devices and convert it into feasible actions or clear patterns for human analysis. They analyse information based on various settings and designs in order to perform automation-related tasks or provide the data required by industry.

3.1.6.4 Application and Process Extension

These applications extend the reach of existing systems and software to allow a wider, more effective system. They integrate predefined devices for specific purposes such as allowing certain mobile devices or engineering instruments access. It supports improved productivity and more accurate data collection.

3.2 NODE MCU

NodeMCU (Node Microcontroller Unit) is a low-cost open source IOT platform. It initially included firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which was based on the ESP-12 module. Later, support for the ESP32 32-bit MCU was added.

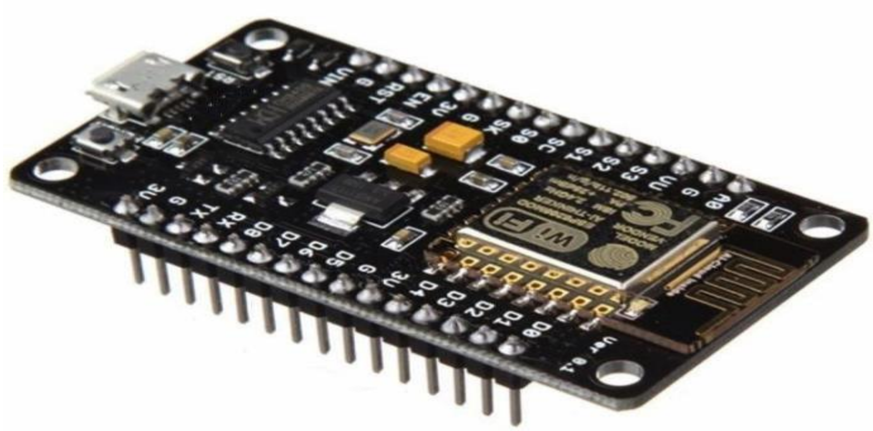


Figure 3. Node MCU Development Board.

NodeMCU is an open source firmware for which open source prototyping board designs are available. The name "NodeMCU" combines "node" and "MCU" (micro-controller unit). The term "NodeMCU" strictly speaking refers to the firmware rather than the associated development kits.

Both the firmware and prototyping board designs are open source.

The firmware uses the Lua scripting language. The firmware is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as luacjson and SPIFFS. Due to resource constraints, users need to select the modules relevant for their project and build a firmware tailored to their needs. Support for the 32-bit ESP32 has also been implemented.

The prototyping hardware typically used is a circuit board functioning as a dual in-line package (DIP) which integrates a USB controller with a smaller surface-mounted board containing the MCU and antenna. The choice of the DIP format allows for easy prototyping on breadboards. The design was initially was based on the ESP-12 module of the ESP8266, which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IOT applications.

3.2.1 Pin Configuration of Node MCU Development Board

This module provides an access to the GPIO subsystem. All the access is based on I/O index number of Node MCU kits, not the internal GPIO pins. For example, the D0 pin on the development kit is mapped to GPIO pin 16. Node MCU provides access to the GPIO pins and the following pin mapping table is a part of the API documentation.

PIN NAME ON NODE MCU DEVELOPMENT KIT	ESP8266 INTERNAL GPIO PIN NUMBER	PIN NAME ON NODE MCU DEVELOPMENT KIT	ESP8266 INTERNAL GPIO PIN NUMBER
0 [*]	GPIO16	**7**	GPIO13
1	GPIO5	**8**	GPIO15
2	GPIO4	**9**	GPIO3

3	GPIO0	**10**	GPIO1
4	GPIO2	**11**	GPIO9
5	GPIO14	**12**	GPIO10
6	GPIO12		

Table 1. Node MCU index ↔ GPIO mapping.

[*] D0 (GPIO16) can only be used for GPIO read/write. It does not support open drain/interrupt/PWM/I²C or 1-Wire.The ESP8266 Node MCU has total 30 pins that interface it to the outside world. The pins are grouped by their functionality as:

Power pins: There are four power pins viz. one VIN pin & three 3.3V pins. The VIN pin can be used to directly supply the ESP8266 and its peripherals, if you have a regulated 5V voltage source. The 3.3V pins are the output of an on-board voltage regulator. These pins can be used to supply power to external components.

GND: is a ground pin of ESP8266 Node MCU development board.

12 IC Pins: are used to hook up all sorts of I2C sensors and peripherals in your project. Both I2C Master and I2C Slave are supported. I2C interface functionality can be realized programmatically, and the clock frequency is 100 kHz at a maximum. It should be noted that I2C clock frequency should be higher than the slowest clock frequency of the slave device.

GPIO Pins: ESP8266 Node MCU has 17 GPIO pins which can be assigned to various functions such as I2C, I2S, UART, PWM, IR Remote Control, LED Light and Button programmatically. Each digital enabled GPIO can be configured to internal pull-up or pull-down, or set to high impedance. When configured as an input, it can also be set to edge-trigger or level-trigger to generate CPU interrupts.

ADC Channel: The Node MCU is embedded with a 10-bit precision SAR ADC. The two functions can be implemented using ADC viz. Testing power supply voltage of VDD3P3 pin and testing input voltage of TOUT pin. However, they cannot be implemented at the same time.

UART Pins: ESP8266 Node MCU has 2 UART interfaces, i.e. UART0 and UART1, which provide asynchronous communication (RS232 and RS485), and can communicate at up to 4.5 Mbps. UART0 (TXD0, RXD0, RST0 & CTS0 pins) can be used for communication. It supports fluid control. However, UART1 (TXD1 pin) features only data transmit signal so, it is usually used for printing log. **SPI Pins**: ESP8266 features two SPIs (SPI and HSPI) in slave and master modes. These SPIs also support the following general-purpose SPI features:

- 4 timing modes of the SPI format transfer
- Up to 80 MHz and the divided clocks of 80 MHz
- Up to 64-Byte FIFO
- **SDIO Pins**: ESP8266 features Secure Digital Input/output Interface (SDIO) which is used to directly interface SD cards. 4-bit 25 MHz SDIO v1.1 and 4-bit 50 MHz SDIO v2.0 are supported.

PWM Pins: The board has 4 channels of Pulse Width Modulation (PWM). The PWM output can be implemented programmatically and used for driving digital motors and LEDs. PWM frequency range is adjustable from 1000 μs to 10000 μs, i.e., between 100 Hz and 1 kHz.

Control Pins: are used to control ESP8266. These pins include Chip Enable pin (EN), Reset pin (RST) and WAKE pin.

- EN pin – The ESP8266 chip is enabled when EN pin is pulled HIGH. When pulled LOW the chip works at minimum power.
- RST pin – RST pin is used to reset the ESP8266 chip.
- WAKE pin – Wake pin is used to wake the chip from deep-sleep.

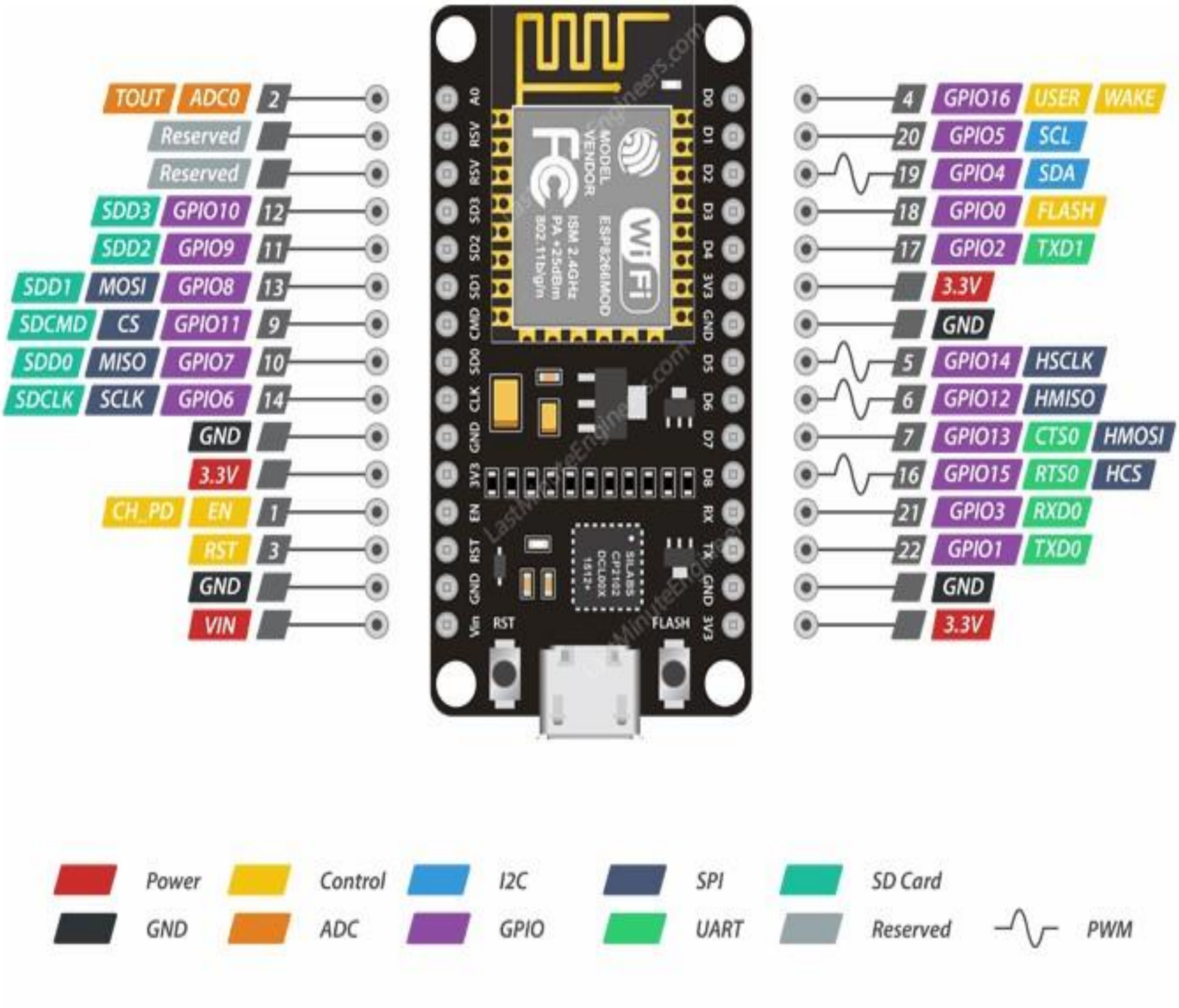


Figure 4. ESP8266 Node MCU pinout.

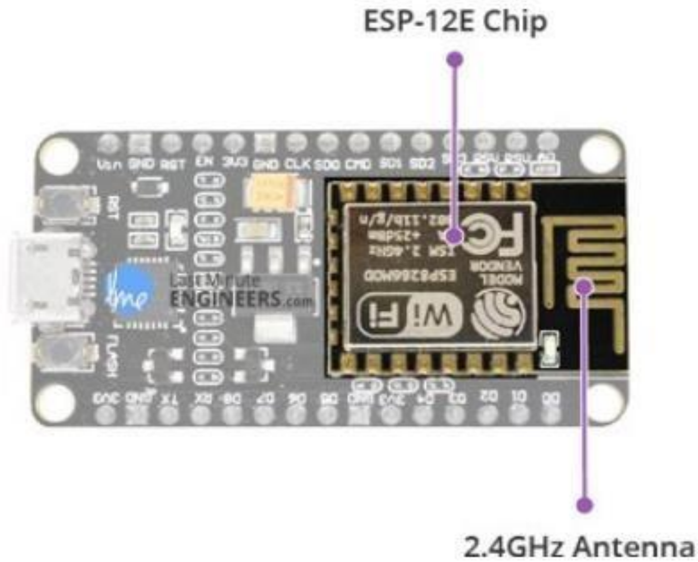
3.2.2 Parts of Node MCU Development Board

3.2.2.1 ESP 12-E Module

The development board equips the ESP-12E module containing ESP8266 chip having Tensilica Xtensa® 32-bit LX106 RISC microprocessor which operates at 80 to 160 MHz adjustable clock frequency and supports RTOS.

There's also 128 KB RAM and 4MB of Flash memory (for program and data storage) just enough to cope with the large strings that make up web pages, JSON/XML data, and everything we throw at IOT devices nowadays.

The ESP8266 Integrates 802.11b/g/n HT40 Wi-Fi transceiver, so it can not only connect to a Wi-Fi network and interact with the Internet, but it can also set up a network of its own, allowing other devices to connect directly to it. This makes the ESP8266 Node MCU even more versatile.



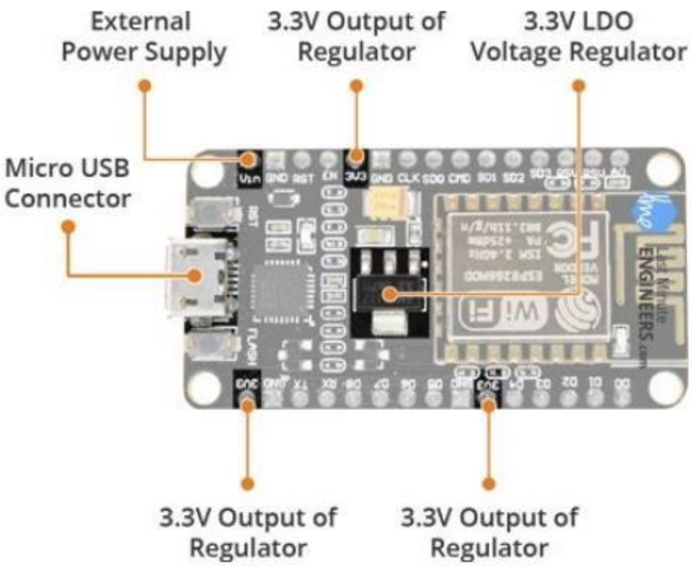
- Tensilica Xtensa® 32-bit LX106
- 80 to 160 MHz clock frequency
- 128 kb internal RAM
- 4 MB external flash
- 802.11b/g/n HT40 Wi-Fi transceiver

Figure 5. ESP 12E module in Node MCU Development board.

3.2.2.2 Power Requirements

As the operating voltage range of ESP8266 is 3V to 3.6V, the board comes with a LDO voltage regulator to keep the voltage steady at 3.3V. It can reliably supply up to 600mA, which should be more than enough when ESP8266 pulls as much as 80mA during RF transmissions. The output of the regulator is also broken out to one of the sides of the board and labelled as 3V3. This pin can be used to supply power to external components.

Power to the ESP8266 Node MCU is supplied via the on-board Micro B USB connector. Alternatively, if you have a regulated 5V voltage source, the VIN pin can be used to directly supply the ESP8266 and its peripherals.



- Operating voltage 2.5V to 3.6V
- On-board 3.6V 600mA regulator
- 80 mA operating current
- 20 µA during sleep mode

Figure 6. Power module on a Node MCU development board.

3.2.2.3 Peripheral I/O

The ESP8266 Node MCU has total 17 GPIO pins broken out to the pin headers on both sides of the development board. These pins can be assigned to all sorts of peripheral duties, including:

- ADC channel – A 10-bit ADC channel.
- UART interface – UART interface is used to load code serially.
- PWM outputs – PWM pins for dimming LEDs or controlling motors.
- SPI, I2C & I2S interface – SPI and I2C interface to hook up all sorts of sensors and peripherals.
- I2S interface – I2S interface if you want to add sound to your project.

As a result of the pin multiplexing feature (Multiple peripherals multiplexed on a single GPIO pin), a single GPIO pin can act as PWM/UART/SPI.

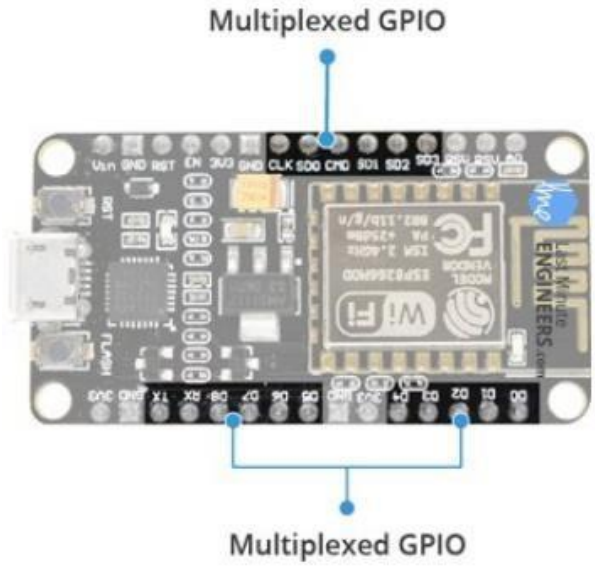
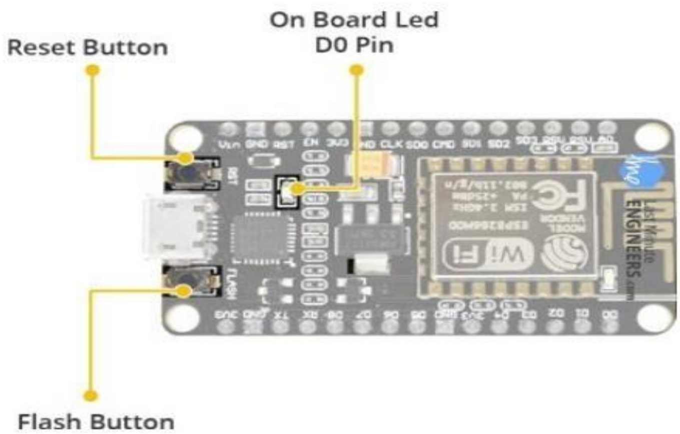


Figure 7. GPIO pins on Node MCU development board.

3.2.2.4 On Board Switches and LED Indicators

The ESP8266 Node MCU features two buttons. One marked as RST located on the top left corner is the Reset button, used of course to reset the ESP8266 chip. The other FLASH button on the bottom left corner is the download button used while upgrading firmware. The board also has a LED indicator which is user programmable and is connected to the D0 pin of the board.



<u>Switches and indicators</u>

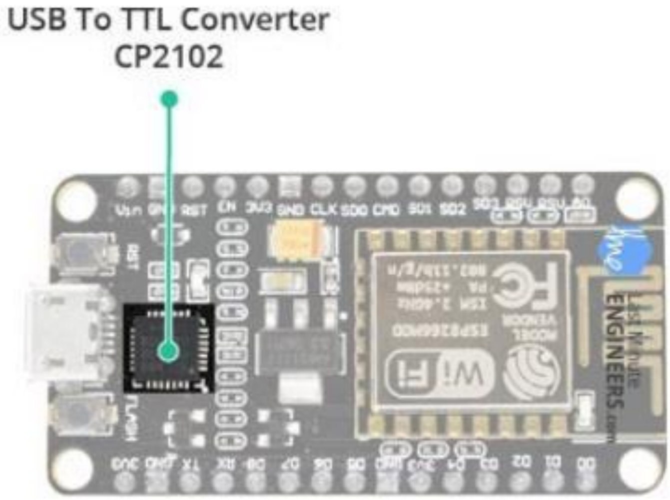
- RST: Reset the ESP8266 chip
- FLASH: Download new programs

Blue LED: User programmable

Figure 8. ON board switches and LED indicators on Node MCU development board.

3.2.2.5 Serial Communication

The board includes CP2102 USB-to-UART Bridge Controller from Silicon Labs, which converts USB signal to serial and allows your computer to program and communicate with the ESP8266 chip.



- CP2120 USB-to-UART converter
- 4.5 Mbps communication speed
- Flow control support

Figure 9. CP2120 on Node MCU development board.

3.2.3 Installation of Node MCU

Mostly these days devices download and install drivers on their own, automatically. Windows doesn’t know how to talk to the USB driver on the Node MCU so it can’t figure out that the board is a Node MCU and proceed normally. Node MCU Amica is an ESP8266 Wi-Fi module-based development board. It has got Micro USB slot that can directly be connected to the computer or other USB host devices. Ti has got 15X2 header pins and a Micro USB slot, the headers can be mounted on a breadboard and Micro USB slot is to establish connection to USB host device. It has CP2120 USB to serial converter.

In order to install CP2120 (USB to serial converter), user is needed to download the driver for the same. Once user downloads drivers as per its respective operating system, the system establishes connection to Node MCU. The user needs to node down the COM post allotted to newly connected USB device (Node MCU) from device manager of the system. This com port number will be required while using Node MCU Amica. As the CP2120 driver is been installed, the Node MCU can be programmed using Arduino IDE software by coding in embedded C. this requires ESP8266 board installation in Arduino IDE from board manager, and assigning communication port.

3.3 Block Diagram

3.3.1 Block diagram of the Proposed System

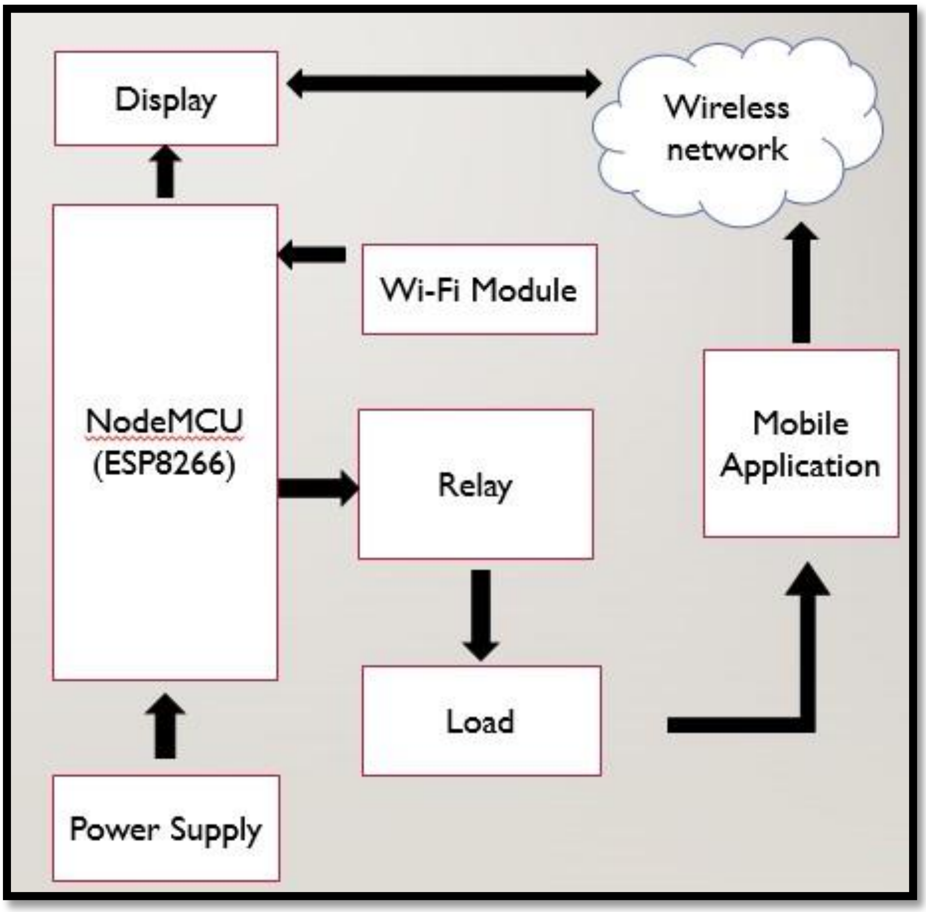


Figure 10. Block diagram of proposed system.

The block diagram gives the functionality of the overall project. The Node MCU unit is the microcontroller or the main controlling unit of the system. The user uses the mobile application in setting commands for functioning of the appliances. The mobile application interprets the command form in user in voice or switch mode and sends signal to the Node MCU unit, over a wireless network established by Wi-Fi communication. Hence the Wi-Fi module (actually inbuilt into Node MCU), helps the microcontroller establish Wi-Fi communication with a device and take commands from an application over wireless network. The Node MCU on further receiving the signal then turns on/off the appliance with the help of relay. The Node MCU, relay and the final appliances are physically connected. There is a power supply unit that powers the microcontroller, the relay as well as the final appliances. There is also a display unit that displays the status of the application.

3.3.2 Proposed system

The android OS provides the flexibility of using the open source. The inbuilt sensors can be accessed easily. The application used to control the system has the following features. Android Phone acts as a client and data are sent via sockets programming. The application takes command from user in two different modes.

- **Switch mode:** Switch mode uses the radio buttons that are used to control the home appliances. The radio button sends the status of the switch.

- **Voice mode:** Voice Mode is used to control the home appliances using voice command. Using the inbuilt microphone of Smartphone, the application creates an intent that fetches the speech data to the Google server which responds with a string data. The string data are further analysed and then processed.

More detailed discussion about the modes of control and how they actually control the system is discussed if coming chapters.

3.4 Overview of the Project

The following describes the process of creating an account in Cadio application and generating unique ID against a particular device. This ID acts as an identifier for the particular device on the Cadio server.

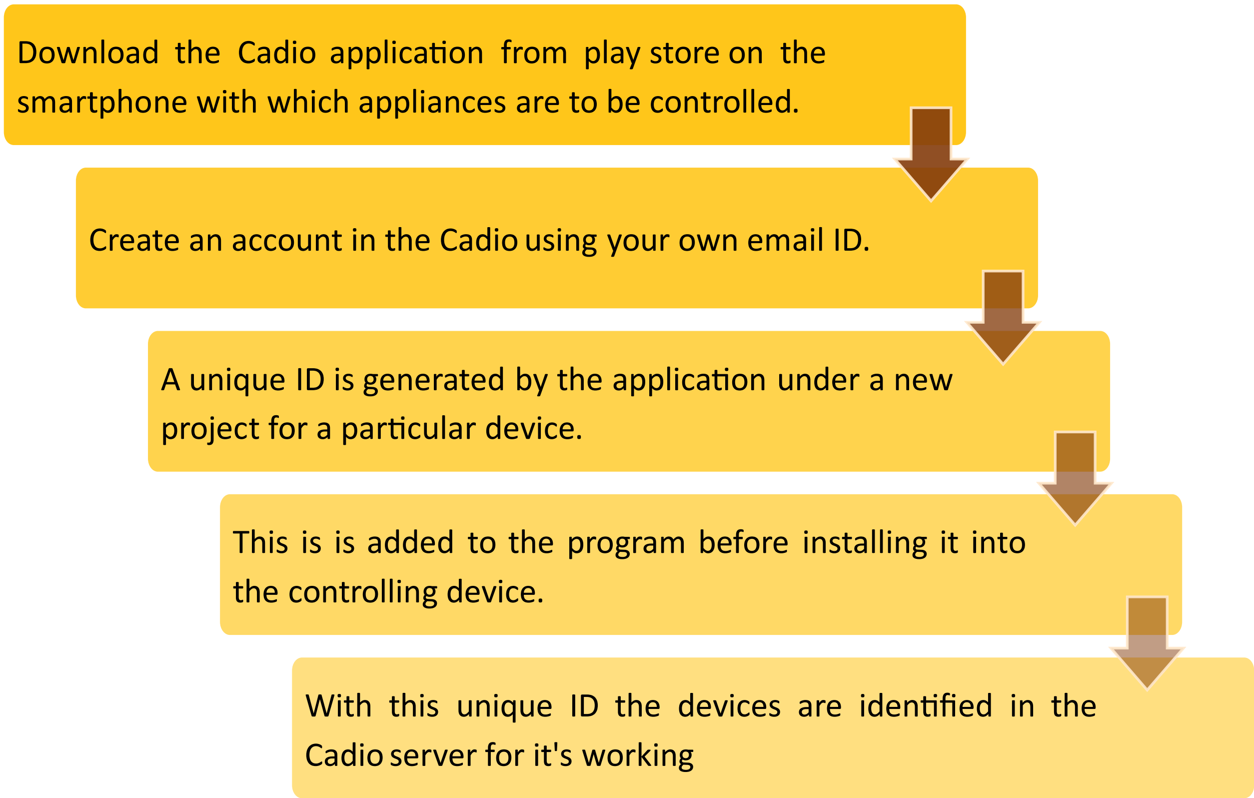


Figure 11. Creating an account and generating unique ID in Server.

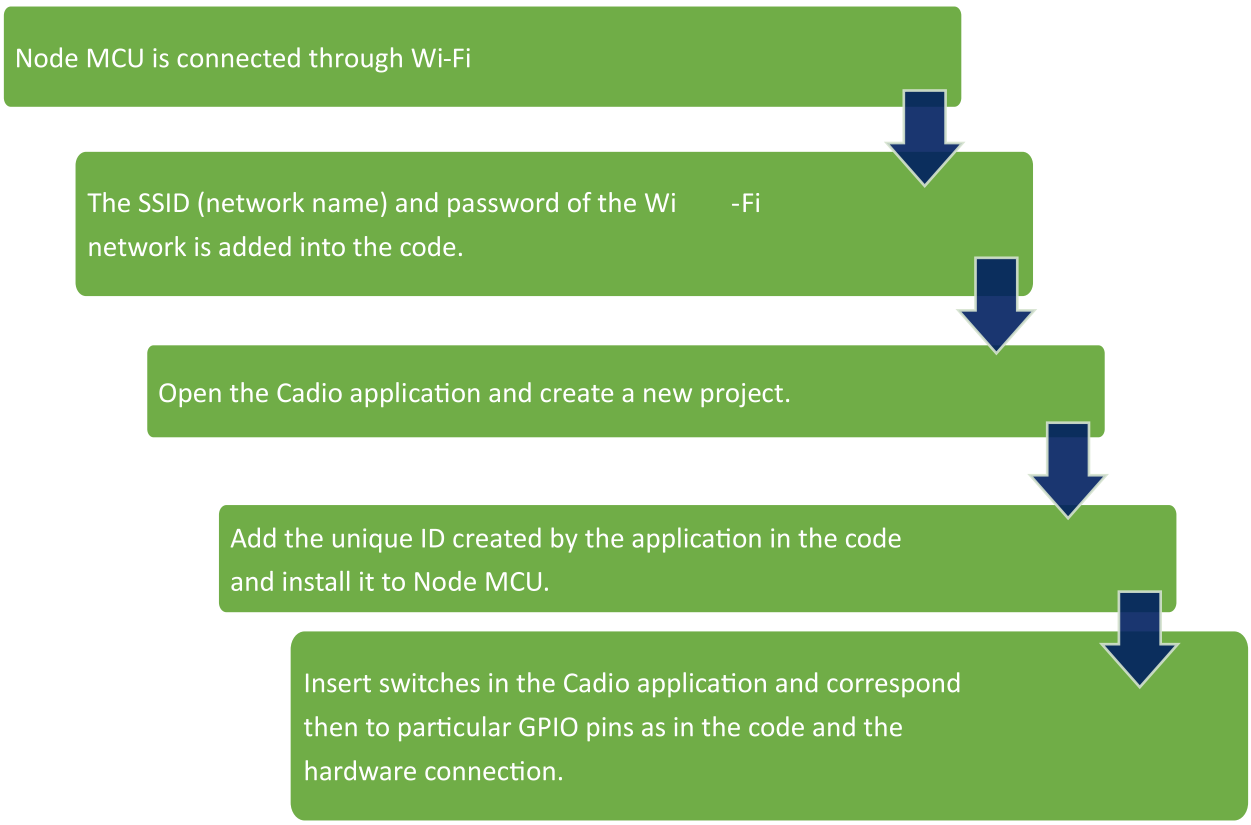


Figure 12. Setup to control Node MCU from Cadio application

3.5 Circuit Diagram

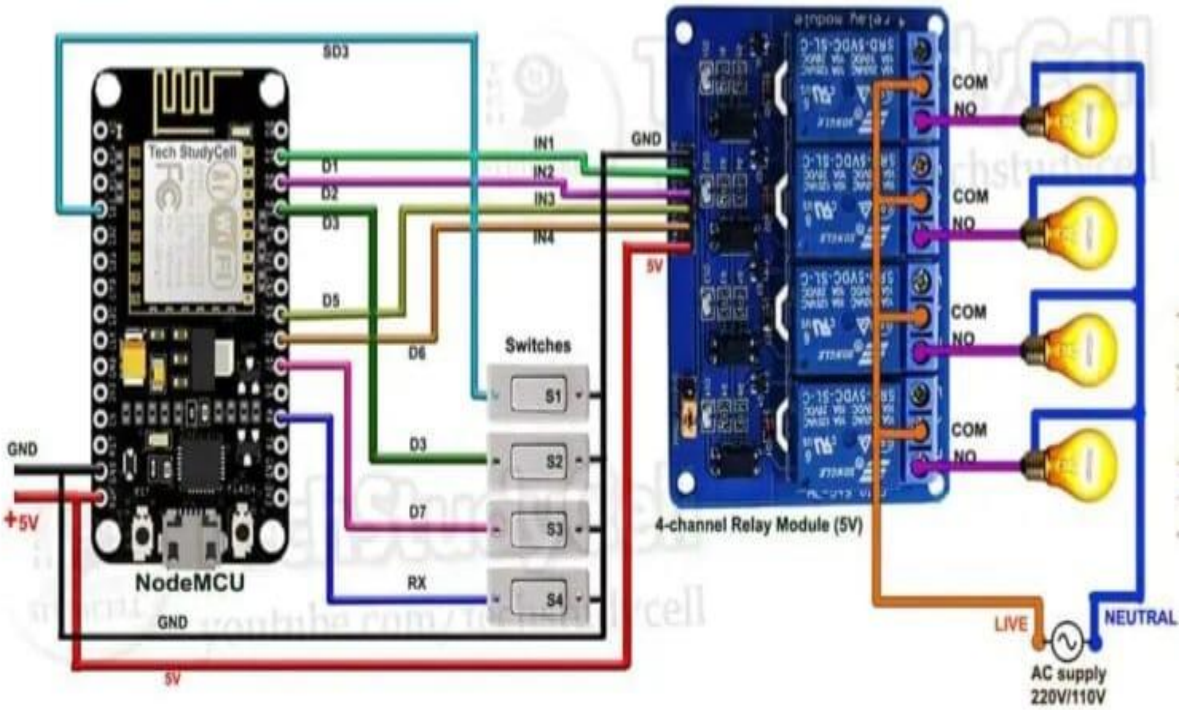


Figure 13. Connection diagram of Node MCU controlling 4 channel relay modules

This is the complete circuit diagram for this IoT based Esp8266 switching. We have used the GPIO pins D1 (GPIO-5), D2 (GPIO-4), D5 (GPIO-14), and D6 (GPIO-12) to control the 4 relays. And the GPIO pins SD3 (GPIO-10), D3 (GPIO-0), D7 (GPIO-13**),** and RX (GPIO-3) are connected with pushbuttons to control the 4 relays manually. The INPUT_PULLUP function in Arduino IDE instead of using the pull-up resistors and a 5V mobile charger to supply the smart relay module.

Hardware Modelling and Setup

4.1 Main Features of the Prototype

The features of the developed prototype are:

- The prototype establishes a wireless remote switching system of home appliances.
- The prototype uses Wi-Fi to establish wireless control, which gives an indoor range to about 150 feet.
- The command to switch on and off an appliance can be given from radio buttons on the application from one's smartphone.
- There is also a provision developed to use voice commands on smartphone to remotely switch home appliances
- Any device capable of Wi-Fi connectivity can be used to control the prototype.
- Displays the status of each appliance on the application in smartphone &Cost effective.

4.2 Project Layout

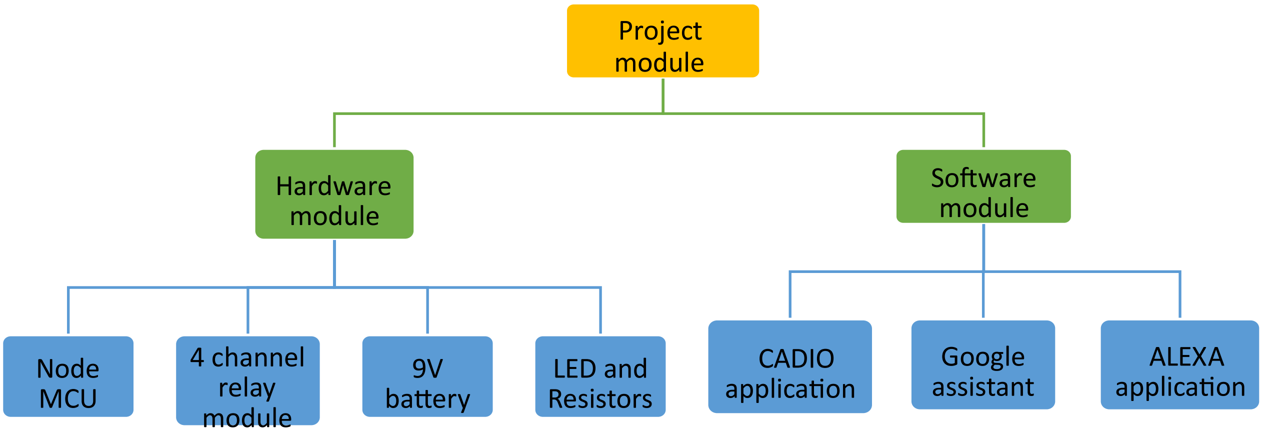


Figure 14. Layout of project module

Node MCU is the microcontroller unit in the prototype. It has an in-built Wi-Fi module (ESP8266) that establishes wireless remote switching of home appliances.

Four channel relay module consists 4 individual relays physically connected between Node MCU and the home appliances. It takes signals form GPIO pins of Node MCU and accordingly connects or disconnects home appliances from the supply. They act as the switching device.

LED and resistors are used in this prototype to replace real appliances. They indicate power being turned on and off to the appliances. In real time operation they would be replaced by actual home appliances.

Cadio is complete home automation platform allows you to build and control smart home devices, With many new features developed to give the best smart home experience.

Google assistant is a system software present on the android phone. It interprets the voice commands by the user to turn on or off an appliance.

Alexa home automation devices work by utilizing your home wi-fi and Bluetooth to let you control smart devices with Alexa voice commands and Alexa routines.

4.3 Components Required

SL. NO	Component	Quantity
1.	Node MCU	1
2.	4 channel relay board	1
3.	9V battery	1
4.	LED	4
5.	2.2K Ω Resistor	4
6.	Blank PCB (KS100)	1
7.	Male pin header	1

8.	Female pin header	1
9.	Jumper wires	8
10.	USB Cable	1

Table 2. Component listing.

4.4 Setting Up System

4.4.1 Downloading and installing Cadio Firmware on ESP8266

To make this project, first, you must install the Cadio Firmware on ESP8266.

- The CADIO dynamic firmware for ESP8266 NodeMCU is available to download from the Web'.
- SPI flash mode is **DIO &** SPI flash speed is **40MHz.**
- Make sure that the **DoNotChgBin** checkbox is activated.
- You should **erase** the current flash before start flashing the firmware to avoid any issues.
- Now click on **Start** to install the Cadio Firmware.

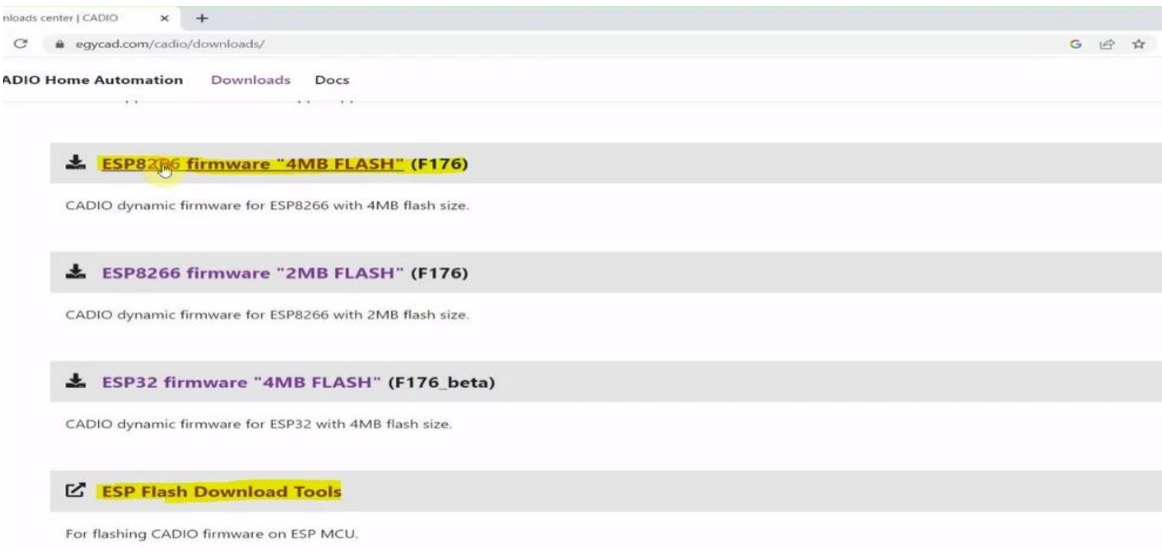


Figure 16. Set up Cadio application

4.4.2 Configure the NodeMCU using Cadio App

1. Turn off mobile data and connect the "CADIO" hotspot created by the ESP8266.
2. Open the Cadio app, and click on the 3-dash icon.
3. Select **Configuration.**
4. Select Switches Mode: "**PRESS**" for the push button. For latched switch select "NORMAL"
5. Select Relay Mode: "**ACTIVE LOW**". For active HIGH relay select "**ACTIVE HIGH**".

Figure 23. Configuration of NodeMCU

4.4.3 Enter GPIO for Relays and Wi-Fi Details

1. Select the Switches wiring "**PULLUP**".
2. Now enter the GPIO pins for **Config button, Indicator LEDs** and **Devices** as shown in the picture. Then click on the tick icon (in the top right corner).
3. Click on **YES** to export the info file.
4. Select the **Wi-Fi Name** and enter the **Wi-Fi Password**. Also, enter a unique **unit name** and tap on the tick icon.

5. Wait for some time the dashboard will appear automatically.

Now we have to connect the ESP8266 NodeMCU to circuit.

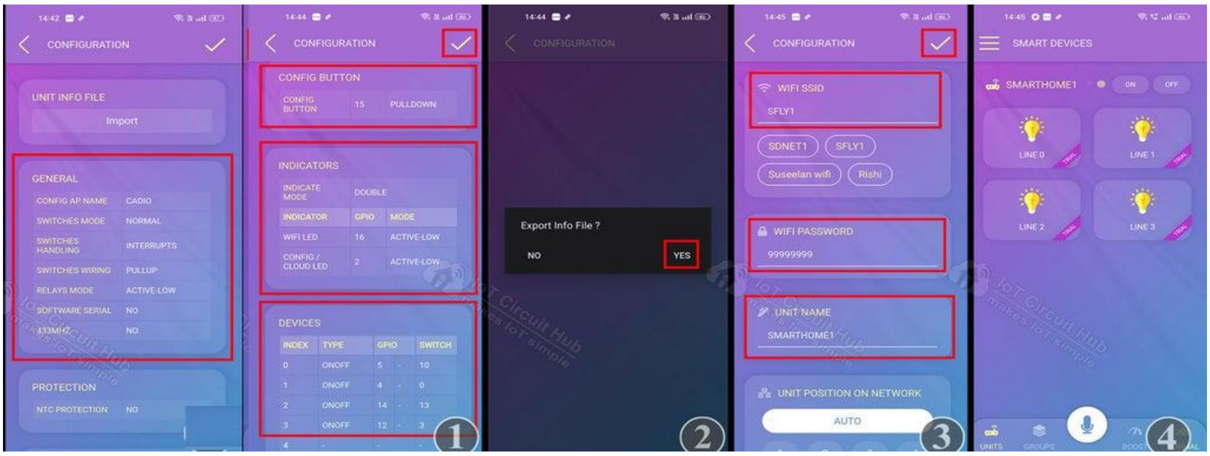


Figure 24. Configuration of NodeMCU using Cadio

4.4.4 Control Relays with Cadio App

If the NodeMCU ESP8266 is connected to Wi-Fi, then you can control the home appliances from Cadio App.

Figure 25. Controlling Relays with Cadio App

4.4.5 Connect Cadio with Google Home

Open the Google Home app and create a Home. Then click on "+" and select "Set up device", the " works with Google”. Now search for CADIO, and log in to CADIO account.

All the devices from CADIO will be added to Google Home. Now you can also control the appliances with voice commands using Google Assistant.

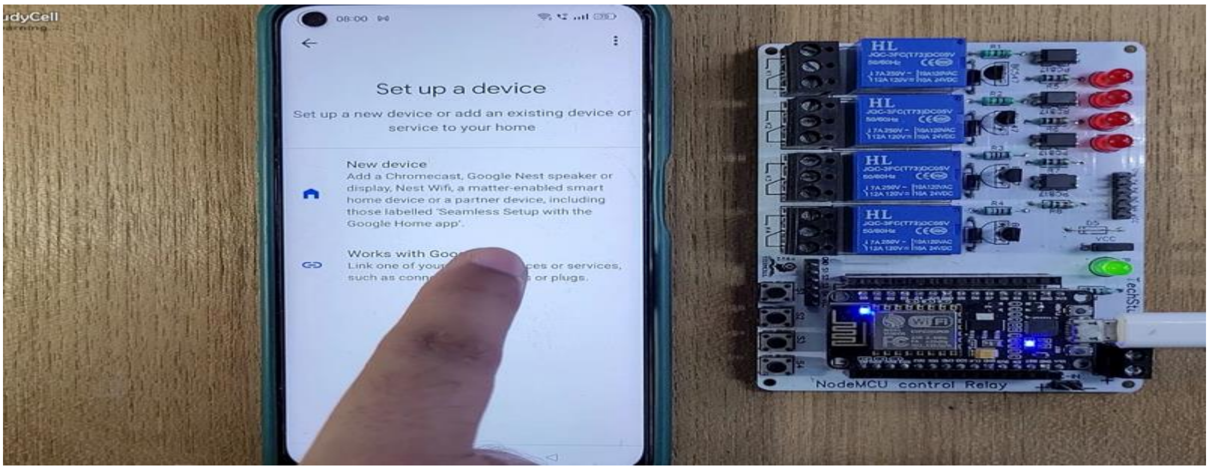


Figure 26. Connecting Cadio with Google Home

4.4.6 Connect Cadio with Amazon Alexa

1. Open the Amazon Alexa app then click on **YOUR SMART HOME SKILLS.**
2. Click on **ENABLE SMART HOME SKILLS.**
3. Type CADIO in the Search bar, then click on CADIO.
4. Click on **ENABLE TO USE** to enable CADIO skill.
5. Now Log in using your CADIO account.
6. Your CADIO account is now linked with Amazon Alexa. You can now control your smart devices from the Amazon Alexa app and supported devices.

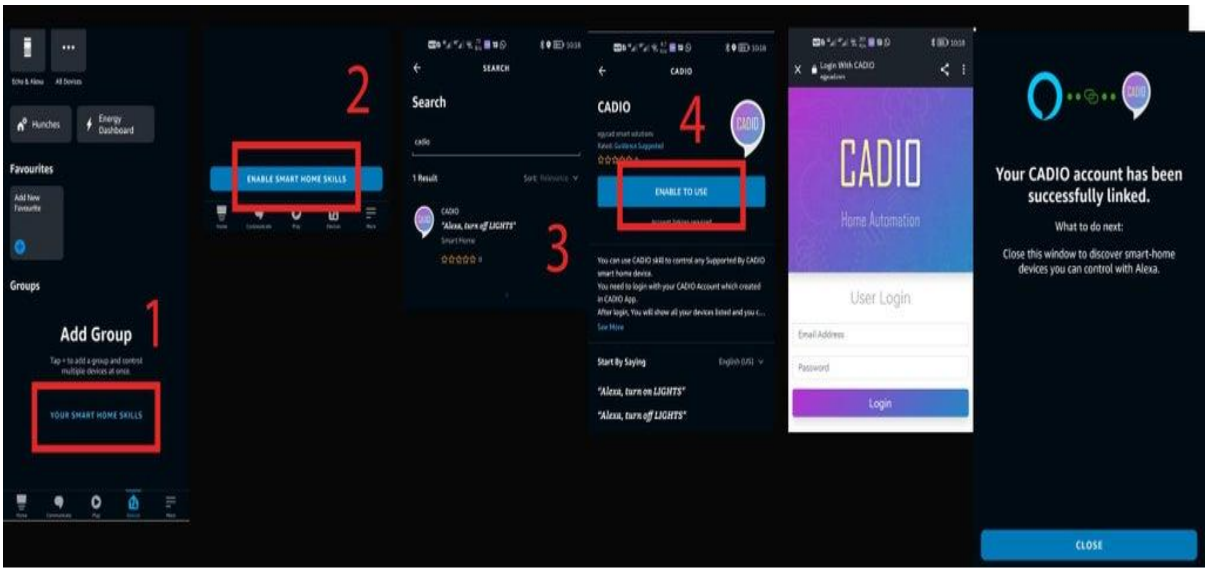


Figure 27. Connecting Cadio with Alexa

4.5 Hardware Assembly

Hardware assembly mainly includes connecting specific digital pins of NodeMCU to the 4 relays on the relay module, including the connection of supply and ground pins. The main functional assemble in this prototype is simple. The further 4 relays are fit to be connected to any appliance desired to be controlled.

The vital part in hardware assembly is taking into account the digital pin that corresponds to which relay. This connection is done as per the setup of Blynk application. The radio buttons on Blynk application are set up to switch a particular digital pin in Node MCU. It is made sure that the relay connection are physically made according to this set up. For example, we have assigned the radio button on Blynk application corresponding to relay 1 to work with D3. Then physical connection of relay 1 is made with D3 of Node MCU.

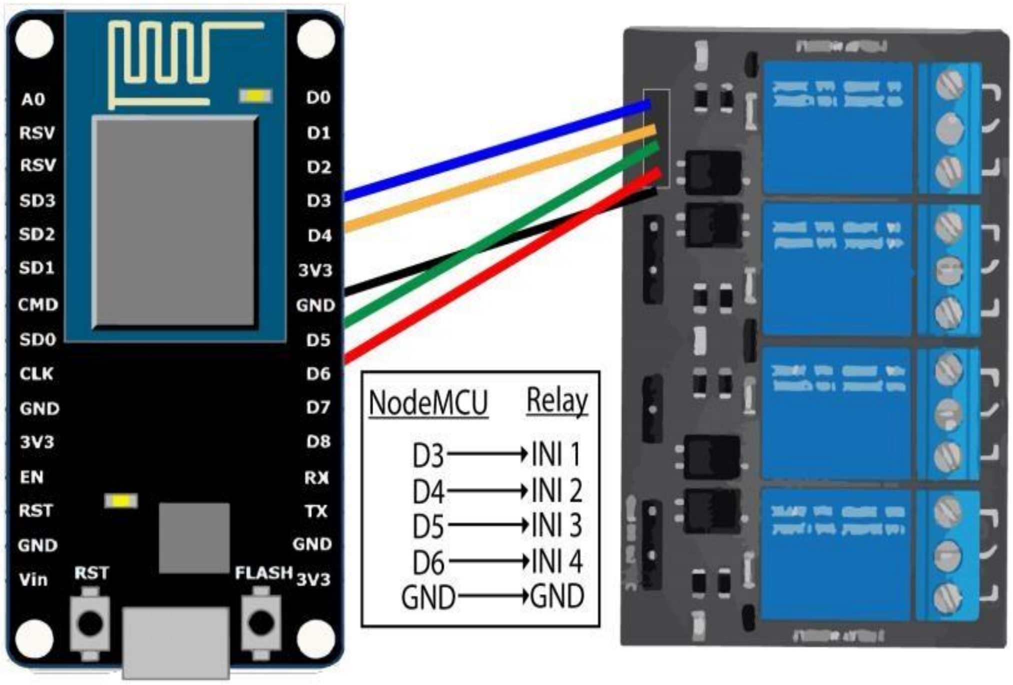


Figure 28. Node MCU & 4 channel relay connection.

In this prototype instead of real home appliances, we connect the relays to LEDs, (according to circuit diagram) to just ensure the functionality of the prototype. The prototype is given a supply from a 9V battery.

Conclusion And Future Scope

We successfully IoT based Esp8266 switching using Cadio. Home automation using IoT-based ESP8266 modules presents several challenges. Security vulnerabilities can arise due to increased connectivity, requiring robust encryption and authentication measures to safeguard against unauthorized access. Reliability issues might emerge due to network instability or device malfunctions, necessitating reliable fail-safe mechanisms to prevent disruptions in automated processes. Lastly, managing power consumption efficiently in a network of connected devices is essential to maintain sustainability and avoid energy wastage. IoT-based home automation revolutionizes the way we interact with our living spaces. It involves integrating a variety of smart devices, sensors, and appliances into a network that can be controlled, monitored, and automated remotely through the internet. This technology enhances comfort, convenience, energy efficiency, and security within homes.

6.1 Further Enhancement & Future Scope

Looking at the current situation we can build cross platform system that can be deployed on various platforms like iOS, Windows. Limitation to control only several devices can be removed by extending automation of all other home appliances. The prototype can include sensors to implement automatic control of the home appliances like; an LDR that can sense daylight and switch lamp accordingly, a PIR to detect motion and be used for security purposes making an alarm buzz, or a DHT11 sensor that's senses ambient temperature and humidity of atmosphere and switch fan/air conditioner accordingly. Scope of this project can be expanded to many areas by not restricting to only home, but to small offices'

It is evident from this project work that an individual control home automation system can be cheaply made from low-cost locally available components and can be used to control multifarious home appliances ranging from the security lamps, the television to the air conditioning system and even the entire house lighting system. And better still, the components required are so small and few that they can be packaged into a small inconspicuous container. The designed home automation system was tested a number of times and certified to control different home appliances used in the lighting system, air conditioning system, home entertainment system and many more . Hence, this system is scalable and flexible.

REFFERENCES

[1] "Smart Energy Efficient Home Automation System using IOT", by Satyendra K. Vishwakarma, Prashant Upadhyaya, Babita Kumari, Arun Kumar Mishra.

[2] "IOT Based Smart Security and Home Automation", by Shardha Somani, Parikshit Solunke, Shaunak Oke, Parth Medhi, Prof. P. P. Laturkar.

[3] "A Dynamic Distributed Energy Management Algorithm of Home Sensor Network for Home Automation System", by Tui-Yi Yang, Chu-Sing Yang, Tien-Wen Sung; in 2016 Third International Conference on Computing Measurement Control and Sensor Network.

[4] "Enhance Smart Home Automation System based on Internet of Things", by Tushar Churasia and Prashant Kumar Jain; in Proceedings of the Third International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2019) IEEE Xplore.

[5] "Visual Machine Intelligence for Home Automation", by Suraj, Ish Kool, Dharmendra Kumar, Shovan Barman.

[6] "A Low-Cost Home Automation System Using Wi-Fi based Wireless Sensor Network Incorporating internet of Things", by Vikram.N, Harish.K.S, Nihaal.M.S, Raksha Umesh, Shetty Aashik Ashok Kumar; in 2017 IEEE 7th International Advance Computing Conference.

[7] Theory of IOT from :https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT

[8] About Node MCU from: https://lastminuteengineers.com/esp8266-nodemcu-arduino-tutorial/

[9] Techstudy Cell, "NodeMCU Home Automation Project using Cadio" , Autodesk Instructables, 2023, Available at: NodeMCU Home Automation Project With Cadio : 14 Steps - Instructables

[10] Krishna Kumar, Shwetav Sharad, "IoT based Home Automation using Smart Switch"(January, 2020)at;https://www.researchgate.net/publication/338680972_IOT_Based_Home_Automation_Using_Smart_Switch