Smart Framework for Automatic Control of Home Appliances Using IoT

Smart Framework for Automatic Control of Home Appliances Using IoT

Mallikarjuna Rao Gundavarapu [1], Srineeth Kumar Ineni [2], K. Sathvika [3], Gannavaram Shiva Keshava [4], Uppala Rishik Charan [5]

[1,2,3,4,5] Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, India.

gmallikarjuna628@grietcollege.com[1], srinithkumar35@gmail.com[2], sathvikarao09@gmail.com[3], chandrahaas.54321@gmail.com[4], rishikuppala@gmail.com[5]

Abstract: With the advent of the 21st century, comfort and efficiency started becoming a part of a common man's life. Virtual assistants like Amazon Alexa and mobile applications like Google Assistant have tremendously helped in making Home Automation Systems more effortless and orderly. Efforts are still being made for better intra-connection of our homes. Home automation has created a great impact in the field of the Internet Of Things. In this paper, we have used the pre-existing technology of Alexa Echo to make Home Automation more cost-effective by enabling non-smart devices to react to the virtual assistant. In this framework, we proposed a smart node (relay) that can remotely control the connected electrical gadget using text/voice commands. The status can be monitored and controlled by the Alexa app remotely without the need for physical voice commands to Alexa. Further, room ambience, as well as emotion-based lighting, can be provided.

Keywords: Home Automation, Virtual Assistants, Relay, Alexa Echo

1. Introduction

Home Automation makes it easier and more efficient to use home lighting, heating appliances, and other electronic appliances. It also provides users with increased levels of comfort and security. The resulting outcome is that of lesser energy consumption and a greener environment. [1] It might be as basic as controlling a few lights remotely or automatically, or it can be a comprehensive system that manages all key aspects of your home. Home automation is anything that turns your home into a smart environment by allowing you to control objects around the house automatically.

IoT stands for Internet of Things, which is defined as a network of physical objects that are embedded with circuits, software, network connectivity, electronics, and also sensors in order to connect and transfer data. Over an existing network infrastructure, the Internet of Things helps to sense the objects and operate them remotely, enabling more connections of the physical environment into virtual systems while also improving performance. [2]

To reduce "standby" power draws, simply enable the system to manage your energy consumption in the most effective way - switch lights off when rooms are vacant and lower the heating temperature when no one is home. Perhaps your key concerns are convenience and comfort, so your Smart Home can monitor air quality or operate your HVAC system to guarantee your home has a consistent flow of fresh, filtered air.

A wireless network for home automation is one example of a potential application. Consider a house fully equipped with sensors for controlling the lights with a remote control device as complicated as creating a network of items in your home like security system lighting which can be programmed with the help of a controller.[3]

2. Existing Approaches

2.1. Using E-mail

Interactive Home Automation System through E-mail using Raspberry pi, it helped to construct a home automation system on the Raspberry Pi by scanning the subject of an email and using Python, the Raspberry Pi's default programming language. LEDs represented the switching operations. Hence, this work limitation is known as tariff utilizing Dual Tone Multi-Frequency, and therefore it does not provide security for the system if the email is hacked, which means any unauthorized individual can access it.[4]

2.2. Using Raspberry pi 3

The component for providing the smart features to non-smart houses is a Raspberry Pi 3. It consists of Wi-Fi features, and the relay will activate and deactivate the appliances. [5]

2.3. Using Bluetooth Technology

In Home Automation using Bluetooth, instructions are delivered to the Arduino board's Bluetooth antenna with the help of an established Bluetooth connection from a mobile device; however, it is only practical for short distances. [6]

2.4. Using GSM

In Home Automation by using a Global System for Mobile Communication (GSM), GSM is used to send text messages to manage domestic equipment, such as lighting, climate control, and security systems.[7]

3. Proposed Architecture

3.1. Amazon Alexa

Alexa is an AI (Artificial intelligence) based virtual assistant invented by Amazon. Users can activate the device's skills through the wake word "Alexa" and these skills can be further deployed using specific voice commands. One can gather all the required information about weather, traffic, songs, and news. It can also help integrate and control several smart home devices by using itself as a hub.[8] Alexa can control such devices and these devices can consecutively access different content and services. When triggered, the recording of the voice command is sent to Amazon's cloud where the request is processed, and a response is given accordingly

3.2. Node MCU (ESP 8266)

Node MCU which is called ESP 8266, is used to connect objects and helps to transfer data through Wi-Fi protocol. In this project, Node MCU and Alexa need to be connected to the same Wi-Fi to establish a connection between the two. Node MCU is a micro-controller + Wi-Fi platform designed specifically for the development and execution of Wi-Fi and IoT projects. It consists of a USB port to upload the code, a 3.3V regulator, and a logic level converter circuit so you can upload the program and connect the circuits. It includes the ESP-12E Wi-Fi Module. The Arduino IDE is a software used to create and upload codes to Arduino compatible boards and other boards with third-party core compatibility. The ESP8266 community upgraded the Arduino IDE by adding a few features, that allow users to program the ESP8266 with the Arduino IDE and its programming language. There are numerous additional platforms that can be used to program the ESP8266 NodeMCU. [9]

An IoT application inherently requires the sensors to be interfaced with WiFi and remote communication also needs to be established through WiFi. Node MCU contains a built-in ESP-12E Wi-Fi Module and hence it does not require any external WiFi component as needed for arduino. Since WiFi is built in, the power consumption speed of operation and programming becomes much simpler with a node MCU.

3.3. Relay Board

A Relay board is an electromagnetically gadget switch that controls voltage supply and allows you to turn on/turn off circuits using a control switch/voltage barrier. The relay module has connectors with three sockets each - Common, Normally Closed(NC) and Normally Open(NO). The relay coil is directly energised and triggers NC and NO. The COM connects to the main terminal that is required to be controlled. The NC pin is connected to the COM pin. The secondary circuit is normally complete so by default, the current is flowing. The circuit is opened when a signal is sent from the ESP 8266 and disconnects the load. The NO pin is not connected to the COM pin so the circuit is complete only when a signal is sent to close it.

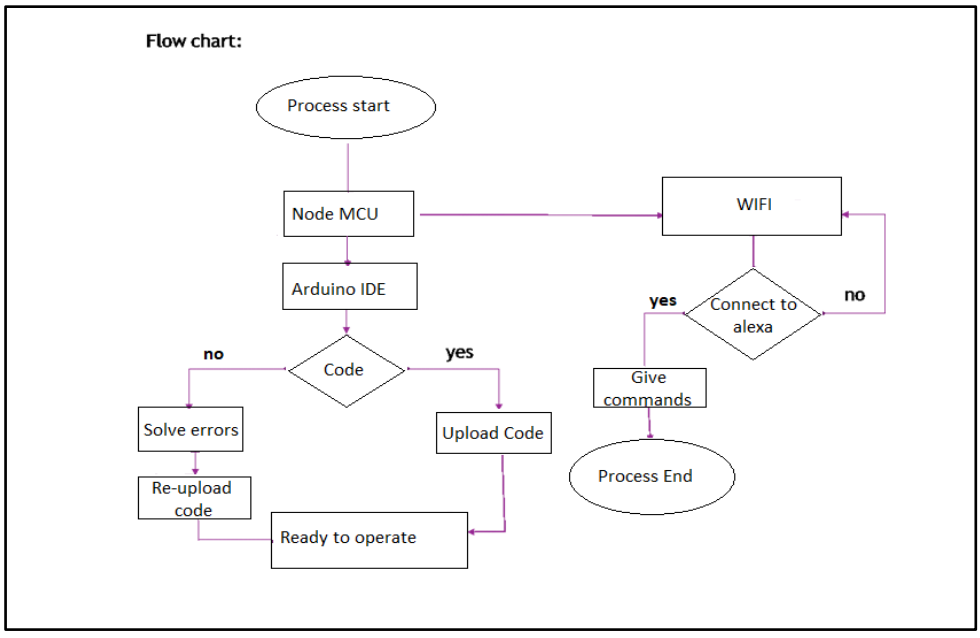

Fig 1. The architectural flow of the Framework

Electromagnetic relays are extremely responsive, minimizing needless delays. Relays are more efficient and depending on the number of devices to be controlled, multi-channel relays can be employed. [10]

4. Implementation

Before coding the program, it is necessary to install certain libraries like espalexa and select the required board from the board manager of IDE. The selected board for this project is NODE MCU 1.0(ESP-12E module). In the framework, all the devices used need to be registered with appropriate names and connected to NODE MCU digital pins (through which communication has to be established to the relay board). Alexa is connected to Node MCU with appropriate Wi-Fi credentials which need to be mentioned in the code. To trigger each individual device, a call-back function needs to be executed for each device. The call-back function code decides whether to switch on or switch off the device based on the input. To safeguard the devices appropriate relays, have to be used.

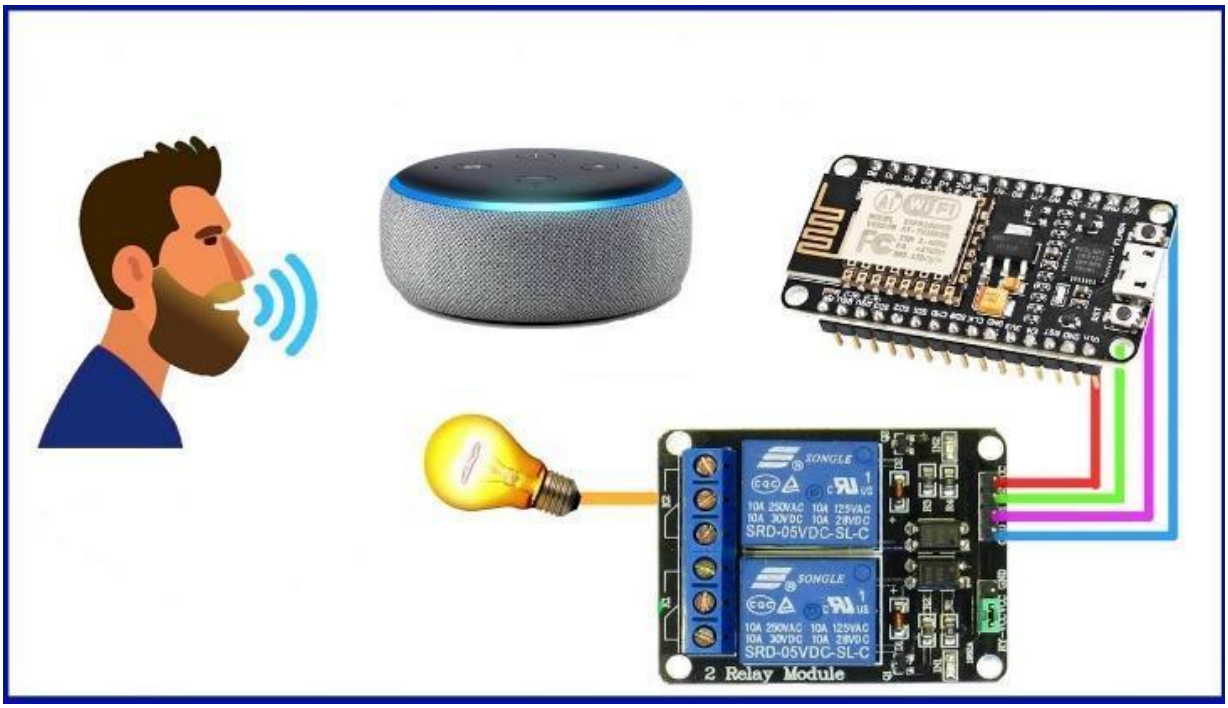

Fig. 2. Implementation

After deploying the code to Node MCU, the devices are required to be registered in the Alexa app to establish a connection between Alexa and Node MCU. In order to set up this connection, both the components need to be connected to the same network. After the connection is established, based on the commands, the Alexa will trigger the particular device through Node MCU with the help of a digital pin allotted to the particular device. Through digital pins, the input is sent to the relay board. According to the input, the relay board will switch the voltage barrier and turns on or off the device.

4.1. Technological stack

4.1.1. Physical design of IoT:

We have used the level-2 design of IoT where the single local node is responsible for sensing and controlling devices and the cloud is used to hold the database. We can monitor the power usage of connected appliances using the cloud with the remote app as shown in (Fig. 3).

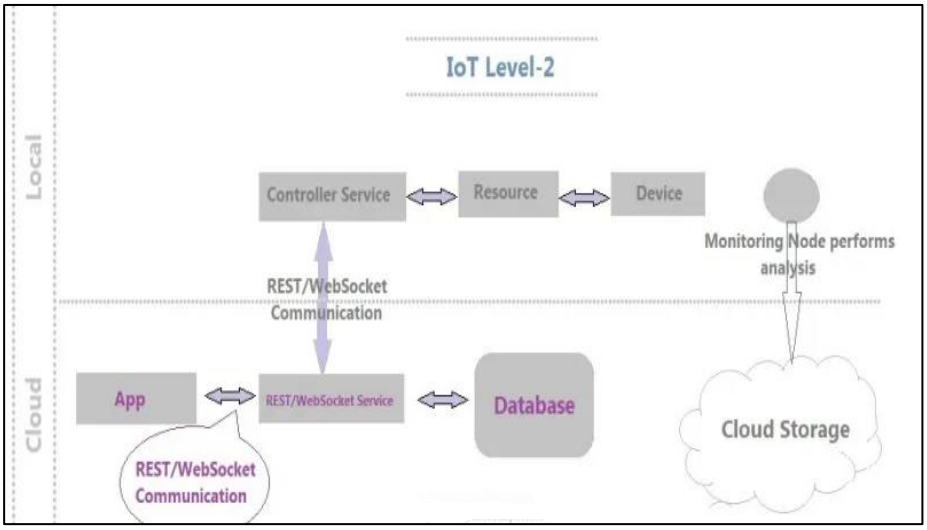

Fig 3. Level 2 Physical Design of IoT

4.1.2. Application Level

ASK stands for Alexa Skill kit, which is a simple Software Development Kit. ASK is used in this project to develop Amazon Echo skills. [11]

4.1.3. Programming Level

All the applications' source codes are programmed with C++, and the Arduino IDE was used to write and compile the code. It is a platform that is free to use. The first Arduino board was released in 2005, and there is now a range of options [12]. It is an Arduino-developed official program that is mostly used for editing, developing, and uploading code to IoT devices. The basic program in IDE is known as a sketch. It provides a HEX file as an output, and the output is uploaded into the board controller with the help of a USB.

5. Benefits

5.1. Energy Consumption

Home automation solutions reduce energy usage and thereby, reduce our carbon footprint too. It enables us to remotely control devices in order to minimise energy consumption by using the Alexa app on our phones. The system proposed by R S Hariharan et al [13] displays the energy

consumed and generates an expected monthly bill through a mobile app. This is beneficial in controlling energy consumption.

5.2. Security

Surveillance cameras can be installed in various spots which can record footage, detect any movements and accordingly alert the owner. In the event of an individual being inaccessible to the location or forgetting to lock the entryways, this process can also be automated to increase security levels. Eric Fry and Shawn Johnson [14] used buzzers and motion sensors to implement the same which can be the future scope for this project.

5.3. Cost-effective

As energy consumption is decreased, the overall cost to pay for electricity also decreases. The prototype can also be built by individuals and the price of the components decreases when bought in bulk.

6. Pseudocode

1. *Define WiFi modules and output pins.*
2. Declare the required functions.
3. Initialize the WiFi credentials SSID and password, and device names.
4. Set the flag as false to check for the initial WiFi connection.
5. Invoke the serial monitor.
6. Set the pin modes as output.
7. Invoke the connect WiFi function.
8. If WiFi is connected

- *Add the devices to the esp Alexa module, with the parameters as the device name and function call to change the default state of the device.*

9. If WiFi is not connected
- *Print the message as "Cannot connect to WiFi. Please check data and reset the ESP."*

Function firstLightChanged (argument 1)
{
If the value of the argument 1 is a non-zero number
- *If the value of argument 1 is equal to 255*
  - *Set the first output pin to HIGH and print the message as "Device1 ON".*

Else if the value of argument 1 is zero
- *Set the first output pin to LOW and print the message as "Device1 OFF".*

}
End function

Function connectWiFi ( )
{
1. Initialize a flag, state to true and a variable i to 0.

2. Set the WiFi mode to WiFi_STA.
3. Initialize the WiFi by passing the SSID and password.
4. Print "Connecting the WiFi" on the serial monitor
5. While status is not equal to WL_CONNECTED
- With a delay of 500 milliseconds, print "." to the serial monitor and increment the value of i.
- If the value of i is greater than 20
  - Set the state flag to false and break out of the loop.

6. If the state is true
- Print the SSID and IP address to the serial monitor

7. If the state is false
- Print "Connection failed" to the serial monitor.

8. Return state.
}
End function

The header file 'WiFi.h' is used to define the WiFi modules and create the necessary environment for WiFi interaction. WiFi.h also contains the function which enables the developer to authenticate ssid and password. On the occasion that user details match the aforementioned attributes, it will allow the user to access the cloud environment using WiFi.After connecting to the WiFi, we give voice commands to alexa, to switch ON or OFF any given device. From which the brightness is set to value based on the command (ON:255, OFF:0) and then call back function of the corresponding device is invoked by "Espalexa" module.

7. Screenshots of Experimentation Results

7.1. Command: ***Alexa, turn on light one.***

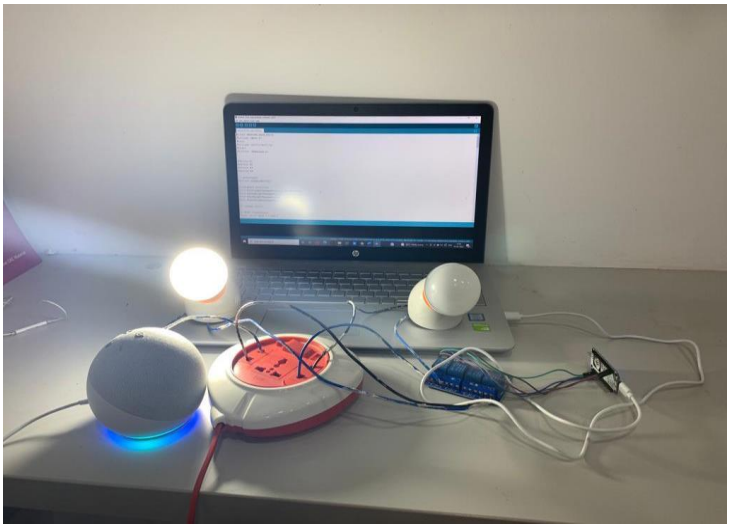

Fig4. Hardware Integration and controlling Left Light using Alexa

When Alexa Echo receives the command “Switch on the bulb 1”, that particular digital pin in node MCU will send a high signal to the relay board. This turns on the bulb.
Similarly, a low signal is sent to turn off the bulb when the command “Switch off bulb 1” is given.

*7.2. Command: **Alexa, turn on light two**.*

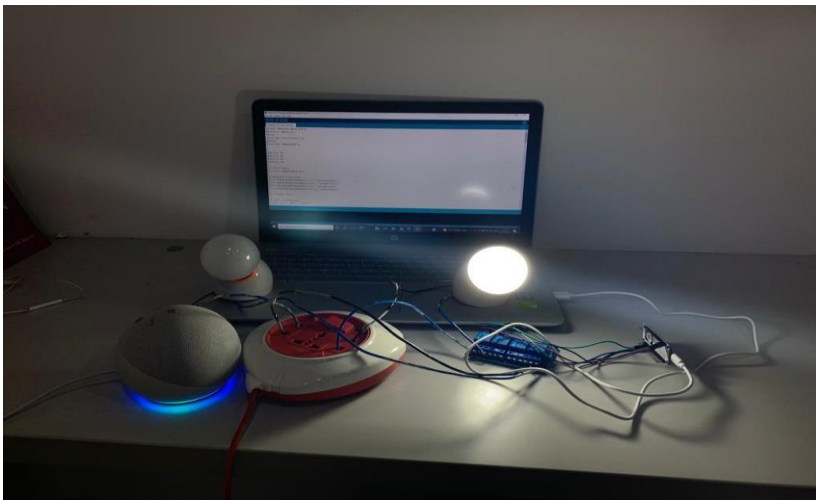

Fig5. Hard ware Integration and controlling Right Light using Alexa

When Alexa Echo receives the command “Switch on the bulb 2”, that particular digital pin in node MCU will send a high signal to the relay board. This turns on the bulb.
Similarly, a low signal is sent to turn off the bulb when the command “Switch off bulb 2” is given.

*7.3. Command: **Alexa, turn on all lights.***

Fig6. Hardware Integration and controlling both lights using Alexa

When Alexa Echo receives the command “Switch on all bulbs”, that particular digital pin in node MCU will send a high signal to the relay board. This turns on the bulb.
Similarly, a low signal is sent to turn off the bulb when the command “Switch off all bulbs” is given.

8. Conclusion and Future Scope Development

The 21st-century world seeks comfort and security at a more effective and personal level, hence the scope and market for home automation systems are greater than ever. But it is still a privilege enjoyed by few due to the presence of various products with exorbitant valuation. With our product, we aimed at bringing service into every home at a reasonable and affordable price that is compatible with any other appliance.

Presently, we have given a cost-effective solution by converting non-smart bulbs into functioning smart bulbs while incorporating a simple command given to a virtual assistant. We believe that this can be further developed. On a small scale, we can delve into regulating the speed of fans, dim the lights based on the external environment, and regulate the temperature of Air Conditioners.

The explosive boom of the production and deployment of smarter systems has raised concerns for privacy of the users. James L [15] presented a study to evaluate users’ behavior when privacy is embedded into the home automation system. We wish to implement a middleware layer to provide protection and privacy for our users.

References

[1] Hamdan, Yasir Babiker. "Smart Home Environment Future Challenges and Issues-A Survey." *Journal of Electronics* 3, no. 01 (2021): 239-246 **3** 1

[2] Bhat O, Bhat S and Gokhale P 2018 Introduction to IoT *International Advanced Research Journal in Science, Engineering and Technology* **5** 41

[3] Shimi S. L, and Kumar M. 2015 Voice Recognition Based Home Automation System for Paralyzed People System *IOSR Journal of Computer Engineering* **4** 1

[4] Goyal L, Vaibhav A and Jain S 2014 Raspberry Pi based Interactive Home Automation System through E-mail *International Conference on Reliability, Optimization, and Information Technology* **1** 278-280

[5] T.S K and K M 2018 Voice Based Home Automation using Amazon Dot *Technical Research Organisation India* **4** 50

[6] Tazil M and Piyare R 2011 Bluetooth based home automation system using cell phone *IEEE 15th International Symposium on Consumer Electronics* **5** 28

[7] Ahmed S, Chan K, Teymourzadeh R and Hoong M 2013 Smart GSM Based Home Automation System *IEEE Conference on Systems, Process & Control* **1** 306-308

[8] Kaundinya A, Atreyas N, Srinivas S, Kehav V and Kumar N 2017 Voice Enabled Home Automation Using Amazon Echo *International Research Journal of Engineering and Technology (IRJET)* **4** 682-683

[9] Parihar Y 2019 Internet of Things and NodeMCU *Journal of Emerging Technologies and Innovative Research* ***6*** *1087*

[10] Kumar A, Senthilnathan N, Rishikeshanan R, Muthusamy S, Somesh S, and Bala, V. 2020 Real-time Implementation of Home appliance control using ALEXA *IOP Conference Series: Materials Science and Engineering* **937** 5

[11] Matriaya S and Purwar R 2018 Voice Controlled Home Automation Using Amazon Echo Dot *IOSR Journal of Computer Engineering* **20** 17

[12] Melin P, Amestica O, Lagos G and Duran-Faundez C 2019 An Experimental Comparison of Arduino IDE Compatible Platforms for Digital Control and Data Acquisition Applications *IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies* **10** 7575

[13] R S Hariharan, Reema Agarwal, Madhurya Kandamuru and H Abdul Gaffar 2021 *IOP Conf. Ser.: Mater. Sci. Eng.* 1085 012026 9

[14] Fry E and Johnson S Embedded Home Security System *Computer Science, United States: College of Natural Sciences Colorado State University-Fort Collins* 2-3

[15] James L 2020 Smart Privacy for IoT: Privacy Embedded Design for Home Automation Systems *College of Computing and Engineering Nova Southeastern University* 1