# Raspberry Pi and NodeMCU based Home Automation System

## Nikhesh Muralidharan

*UG Scholar, PSG College of Technology*

------------------------------------------------------------------------***------------------------------------------------------------------------

**ABSTRACT:-** *This project presents the overall design of Home Automation System (HAS) with low cost and wireless system. The system is designed to assist and provide support in order to fulfil the needs of elderly and disabled at home. The switch mode is used to control the home appliances. The design remains the existing electrical switches and provides more safety control on the switches with low voltage activating method. The switch statuses are synchronized and the user interface indicates the real times statuses of the switches. The system is intended to control electrical appliances and devices in the house with relatively low cost design, user-friendly interface and ease of installation.*

*Keywords: Home Automation, Low Cost Design, Wireless*

## 1. INTRODUCTION

Home automation has been on a rise since the start of 21st century. However, the first designed automation system dates back to 14th century where Heron of Alexandria where he installed an automatic door that works on the basis of steam accumulation. Automation refers to the transfer of the tasks performed by humans to a set of technological elements. Home Automation is the process of automating the home by controlling the home appliances over remote control or controlling the appliances over a set of pre-defined conditions which actuates the appliances. The inclusion of control in the automation system, allows to decide on the development of a process, manipulating certain variables to get these or other variables to act in the desired way. Today's home automation contributes to the increase in the quality of life, makes the distribution of house more versatile, changing environmental conditions by creating different predefined scenes and makes the home more functional by allowing the development of domestic, professional and leisure under one single roof.

It is estimated that by 2022 there will be over 50 billion interconnected devices [1]. The consequence of the increase in internet clients, and thus the increase in devices connected to the Internet is the adoption of IPv6 (Internet Protocol version 6), because each device needs the unique IP address to provide internet connectivity. The use of IPv6 provides, in theory, $3,403 \times 10^{38}$ unique addresses. Due to the extremely large amount of data (Big Data concept) the sensors collect in real time, they are in most cases stored and processed on a Cloud [2]. Their processing gives us a much more detailed insight into how things around us work, where problems arise, and where and how to react. The IoT already transforms our everyday life since many objects are equipped with miniature identification devices or machine-readable identifiers.

### 1.1 PROPOSAL

This project forwards the design of home automation system using the Raspberry Pi herein after called as RPi, a credit card sized computer and Node MCU. The RPi, acts as the server connecting to the internet provides the access to control the home anywhere from the world. A set of Node MCU's installed all over the house, one in each room with predefined static addresses. Whenever a command is issued, the command flows through the Rpi and reaches the particular addressed NodeMCU and actuation occurs. The real-time feedback is provided from the NodeMCU using the current sensor becoming a closed loop. The overall proposal is shown in Fig. 1.
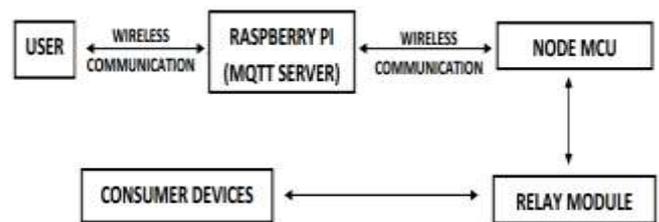


Fig. 1. Proposal – Flowchart.

### 1.2 NODE RED TOOL

Node-RED is an open source flow based development tool for the integration of IoT hardware devices and APIs [3]. The system contains nodes that are represented by appropriate icons. It can be operated by drag and drop or by importing JavaScript code. The node provides different functions, such as to monitor the flow as the debug out node, or to read and write with GPIO pins of Raspberry Pi as the Raspberry Pi node. Created flows are stored using JSON (JavaScript Object Notation). Node-RED enables developers to wire up input, output and processing nodes in order to create flows for data processing, controlling things, or sending alerts [4]. It works on the following principle: it enables the connection of web services or customizes nodes to one another or to things, in order to perform functions such as sending sensor data via e-mail or to services like Twitter, easily performing complex analysis, etc.

Node-RED has three basic components (Fig.2):

1. Node Panel,

2. Flow Panel,

3. Info and Debug Panel.
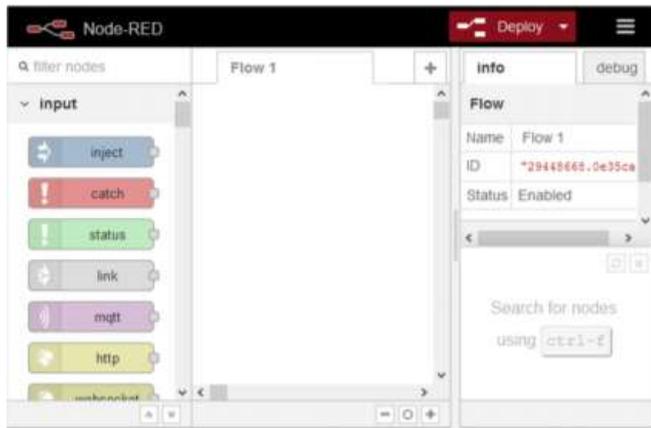


Fig. 2. Node-RED Editor.

## 1.3 RASPBERRY PI

The RPi is a series of small single-board computers developed in the United Kingdom by the RPi Foundation to promote the teaching of basic computer science in schools and in developing countries. The original model became far more popular than anticipated, selling outside its target market for uses such as robotics. It does not include peripherals (such as keyboards and mice) and cases. However, some accessories have been included in several official and unofficial bundles. The Raspberry Pi 3 shown in Fig. 3 is the third-generation Raspberry Pi released in February 2016. This model has following characteristics [5]:

1. Quad Core 1.2GHz Broadcom BCM2837 64bit CPU,

2. 1GB RAM, BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board,

3. 40-pin extended GPIO,

4. 4 USB 2 ports,

5. 4 Pole stereo output and composite video port,

6. Full size HDMI,

7. CSI camera port for connecting a Raspberry Pi camera,

8. Micro SD port for loading your operating system and storing data,

9. Upgraded switched Micro USB power source up to 2.5A.



Fig. 3. Raspberry Pi.

Initially, the RPi has to be loaded with any of the Raspbian operating system. This can be done by burning the image of the operating system over the SD Card and loading it over the Rpi. Later on, the following packages have to be installed over the Rpi.

1. Node-RED package for user front end interface,

2. Apache Server package – For Rpi to act as Server,

3. Mosquitto MQTT Server package – For the interaction between the NodeMCU and Rpi,

4. NGROK package – Since the Node-RED user interface is available only on the local network, for accessing it over the Internet, tunneling is required which is provided by a 3rd party application named NGROK.

## 1.4 HARDWARE IMPLEMENTATION

The hardware components used are

1. NodeMCU,

2. Relay,

3. Current Sensor and

4. RPi.

NodeMCU and the Rpi are connected wirelessly by feeding the IP address of the RPi to the NodeMCU by programming. NodeMCU and the relays are connected by wire. Relays of 3.3 V specification are used since the NodeMCU works on the voltage range of (0 – 3.3) V and hence the maximum voltage that can be outputted and fed to the relay is 3.3 V.
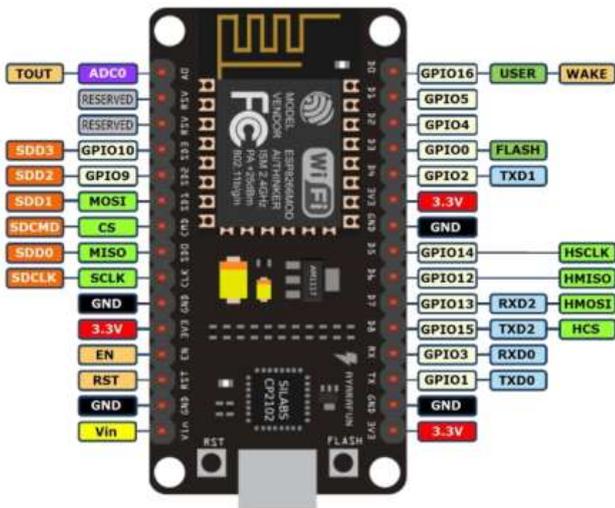
Fig. 4. Node MCU Pinout

The signal outputs are taken from the GPIO pins of the NodeMCU shown in Fig. 4 and connected to the "signal" pin of the Relay Module shown in Fig. 5. Since the relay has to switch the component ON when the signal is given through the Node MCU, the 220V component has to be connected across common (C) and Normally Open (NO) pins.



Fig. 5. Relay Pinout

Fig. 6. shows the pinout of the current sensor. The Load has to be connected on the left two pins, +5 V and the Ground is provided and the output is taken via the middle pin on the right side.
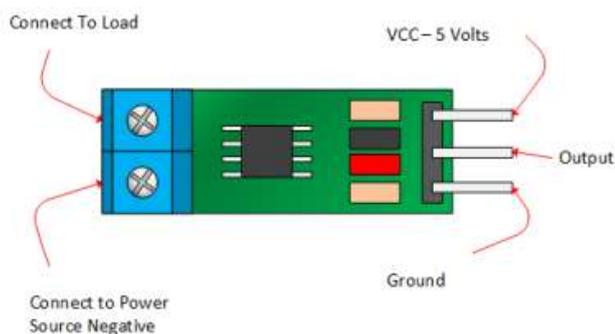


Fig. 6. Current Sensor Pinout

## 1.5 SOFTWARE IMPLEMENTATION

Initially the Apache setup has to be installed on the RPi and it has to be enabled. The status of the Apache2 Server is shown in Fig. 7. The Mosquitto MQTT Broker is installed over the Apache2 Server shown in Fig. 8.
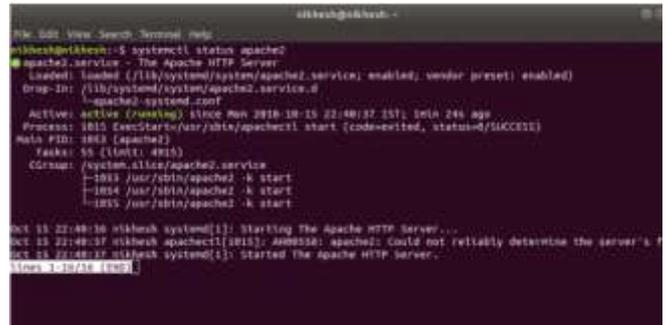


Fig. 7. Apache Server Status
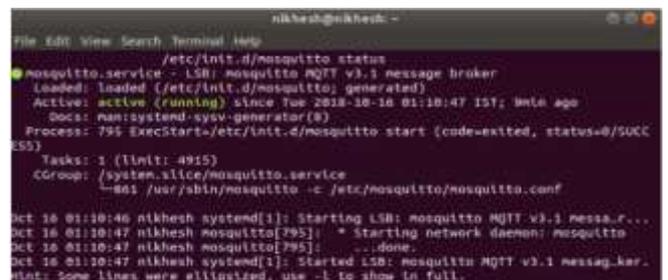


Fig. 8. Mosquitto MQTT Broker Status

Once the installation of the Apache Web server and the broker has been completed, Node-RED interface has to be setup according to our need. The starting up of Node Red is shown in Fig. 9.
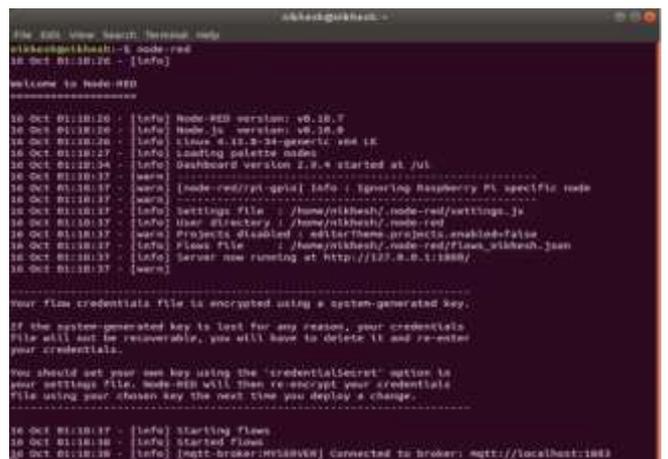


Fig. 9. Node-RED Status

After the Node-RED setup has been completed, the back end programming for the user interface can be done using the Node Red application by opening it in the browser under the 1880 port on the local host. The back end programming in the Node Red is shown in Fig. 10.

Fig. 10. Node-RED Backend Programming

The NGROK application can be used for the application of tunneling the specific port on our local host to the internet. In this case port 1880 of our local host is port forwarded. Fig. 11. shows the web address that can be used to access the port 1880 of the local host over the internet. It is provided under the "Forwarding" option that specifies the web address as http://9041a202.ngrok.io.



Fig. 11. NGROK Application

The Node MCU is programmed using the Arduino IDE such that it has a wireless communication with the RPi. This can be established by feeding the IP address of the RPi into the NodeMCU through the Arduino Ide while programming it. The IP Address of the RPi can be determined by using the command "ifconfig" under the name "inet" as shown in Fig. 12.
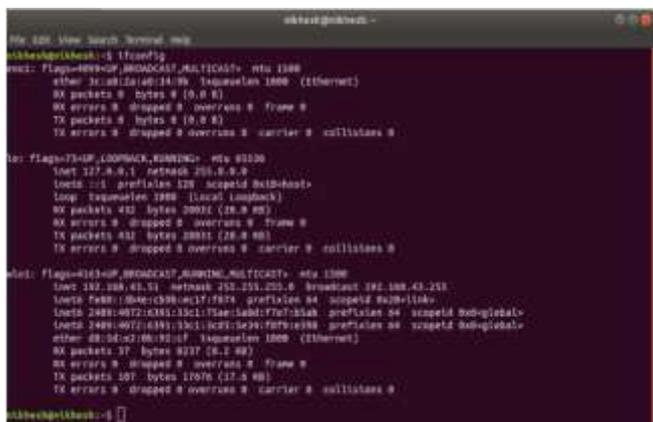


Fig. 12. IP Address Determination

## 1.6 RESULTS

The implementation of this project was done with two NodeMCUs, one each for each room with 2 appliances. However, it can be extended over many rooms. The Fig. 13. and Fig. 14. shows the generated user interfaces on the web browser. The URL link can be obtained from the NGROK Application.
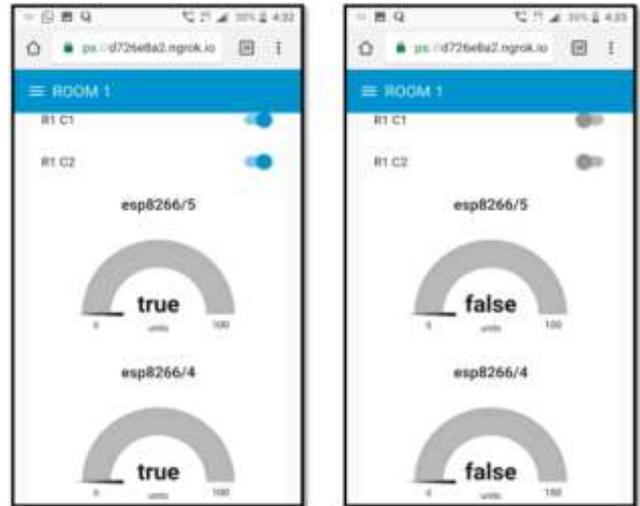


Fig. 13. Room 1 Control Interface

The user interface of the room 1 is shown in Fig. 13. R1 C1 refers to Room 1 Component 1 and R1 C2 refers to Room 1 Component 2. When the any of the component is switched on, the corresponding gauge indicates the status of the component indicating if the component is working or not.
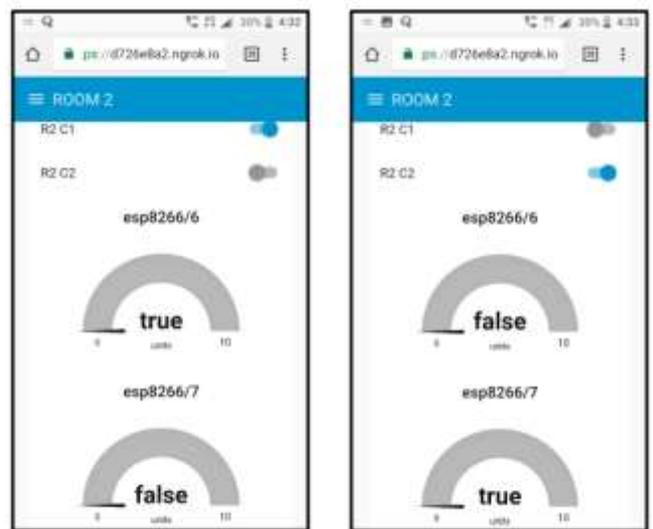


Fig. 14. Room 2 Control Interface

The user interface of the room 2 is shown in Fig. 14. R2 C1 refers to Room 2 Component 1 and R2 C2 refers to Room 2 Component 2. When the any of the component is switched on, the corresponding gauge indicates the

status of the component indicating if the component is working or not. The room selection can be done using the strips at the left top button indicated in the below Fig. 15.
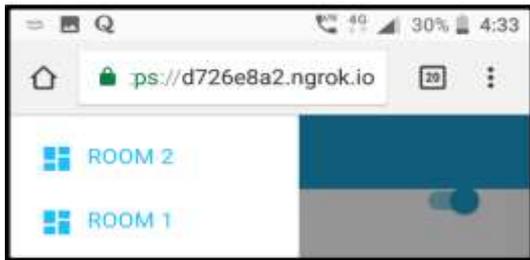


Fig. 15. Room Selection

The first image on the Fig. 16. Shows that the LEDs are switched off when the switches are in off position. When the R1 C1 is switched on shown in second image of Fig. 16. the corresponding LED is switched on. The signal given to the LED, when given to the relay(s) controls the component(s).
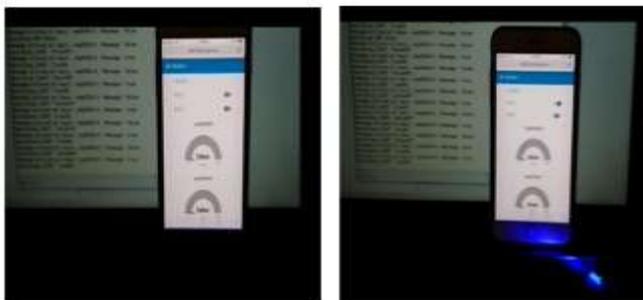


Fig. 16. Real Time Interface

## 2. FUTURE SCOPE

The future scope of this project will be the inclusion of the security aspects and provide make sure authenticated user only can control the devices. The reliability and the cost have to be further decreased. This can be done only by customizing the Raspberry Pi board according to our requirements which can be easily done by using the schematics of the Raspberry Pi and the Node MCU and editing it via software like the Eagle Professional PCB software. The fabrication cost can be further cut down during the mass production of the boards. Thus, Home automation systems with Security enabled customized boards can have a great impact on the market and will prove to be of great assistance.

## 3. REFERENCES

[1]  E. Park, Y. Cho, J. Han, and S. J. Kwon, "Comprehensive Approaches to User Acceptance of Internet of Things in a Smart Home Environment," IEEE Internet of Things Journal, pp. 2342-50, 2017.

[2]  S. H. Shah and I. Yaqoob, "A survey: Internet of Things (IOT) technologies, applications and challenges," IEEE SEGE, pp. 381-5, 2016.

[3]  Node-RED guide, Web site: http://noderedguide.com/

[4]  G. Gardasevic, H. Fotouhi, I. Tomasic, M. Vahabi, M. Bjorkman, M. Linden, "A Heterogeneous IoT-based Architecture for Remote Monitoring of Physiological and Environmental Parameters," 4th EAI International Conference on IoT Technologies for HealthCare, Angers, France, Oct. 24-25, 2017.

[5]  Raspberry Pi, Offici Web site: https://www.raspberrypi.org/