



GOOGLE ASSISTANT BASED HOME AUTOMATION SYSTEM

¹Ms. Sakina Md Aqueel Shaikh, ²Ms. Prajakta Shivanand Kambale, ³Ms. Potdar M. D

Student, Department of Electronics and Telecommunication Engineering, S.V.S.M.D's KKI polytechnic, Akkalkot, Solapur, Maharashtra, India^{1,2}, Lecturer, Department of Electronics and Telecommunication Engineering, S.V.S.M.D's KKI polytechnic, Akkalkot, Solapur, Maharashtra, India³

ABSTRACT

This paper presents a proposal for home automation using voice via Google Assistant. Home automation or domestics a term for home automation coined by Jim Hill has been evolving drastically. We saw many home automation technologies introduced over these years from ZigBee automation to Amazon Echo, Google Home and Home from Apple. It has become a craze these days. Google Home price is around 150\$ (USD) with an additional cost of the devices to be connected to, the total cost of the system reaches over 250\$ (USD). Apple Home Kit too is pretty more expensive, over 100\$ (USD) more than the Google Home just for a basic setup. Philips Hue, a smart light which is controlled by the Google Assistant, Amazon Echo and Siri, voice assistant by Apple is priced around 145\$ (USD). Similarly, Belikin's Wemo light is priced around 44\$ (USD) per unit and this can be controlled both by Siri and Google Assistant. So, overall we can see here that to make our home smart we need to invest quite a lot, let's say some 250\$ (USD) for a basic setup. What if we can automate our house within (cost of the Smartphone is not included as it is assumed to be owned by every individual these days) 10\$ (USD) and can control up to 8 appliances using Google Assistant? Well, this paper describes the implementation of such a system. The system is implemented using ordinary household appliances Natural language voice commands are given to the Google Assistant and with the help of IFTTT (If This Then That) application and the Blynk application the commands are decoded and then sent to the microcontroller, the microcontroller in turn controls the relays connected to it as required, turning the device connected to the respective relay On or OFF as per the users request to the Google Assistant. The microcontroller used is NodeMCU (ESP8266) and the communication between the microcontroller and the application is established via Wi-Fi (Internet).

Key Words: Home Automation, NodeMCU (ESP8266), IFTTT (If This Than That) Application, Blynk Application, Internet of Things (IoT), Google Assistant, Voice Control, Smartphone.

INTRODUCTION

Home, it is the place where one fancies or desires to be after a long tiring day. People come home exhausted after a long hard working day. Some are way too tired that they find it hard to move once they land on their couch, sofa or bed. So any small device/technology that would help them switch their lights on or off, or play their favorite music etc. on a go with their voice with the aid of their smart phones would make their home more comfortable.

Moreover, it would be better if everything such as warming bath water and adjusting the room temperature were already done before they reach their home just by giving a voice command. So, when people would arrive home, they would find the room temperature, the bath water adjusted to their suitable preferences, and they could relax right away and feel cozier and rather, feel more homely.

Human assistants like housekeepers were a way for millionaires to keep up their homes in the past. Even now when technology is handy enough only the well to do people of the society are blessed with these new smart home devices, as these devices costs are a bit high. However, not everyone is wealthy enough to be able to afford a human assistant, or some smart home kit. Hence, the need for finding an inexpensive and smart assistant for normal families keeps growing.

This paper proposes such inexpensive system. It uses the Google Assistant, the IFTTT [1] application, the Blynk [2] application and the NodeMCU [3] microcontroller as the major components along with a relay board comprising of 4/8 relays along with ULN 2803 IC. Natural language voice is used to give commands to the Google Assistant [4]. All of the components are connected over the internet using Wi- Fi which puts this system under the IoT [5].

SYSTEM DESIGN ANDIMPLEMENTATION

The system design is broken down into two main categories,

- i. The hardware- It has the capability to connect to the router. It would also be able to turn on/off specified devices, such as lights and fans. It is called the 'Control Unit'.And,
- ii. The Software- the Blynk app, the IFTTT app and the Google Assistant constitute the software of the design and these applications would be integrated in the Androiddevice.

The Control Unit comprises of the microcontroller- NodeMCU and the 4/8 Channel Relay board. Relay board uses ULN 2803 IC to control the relays. The Blynk app on an Android device communicates with the microcontroller and sends the desired signal via the internet. Figure 1 below shows the basic system design architecture.



Fig -1: Basic System Architecture

The hardware also called the Control Unit comprises of the NodeMCU microcontroller and the Relay board. NodeMCU digital output pins are connected to the Relay pins of the Relay board. Finally, each Relay is connected to an appliance. In the fig- 1 above the second relay is connected to a bulb.

Node MCU(ESP8266)

The NodeMCU [3] (Node Microcontroller Unit) is an open source software and hardware development environment that is built around a very inexpensive System-on-a-Chip (SoC) called the ESP8266. The ESP8266 is designed and manufactured by Express, contains all crucial elements of the modern computer: CPU, RAM, networking (Wi-Fi), and even a modern operating system and SDK. When purchased at bulk, the ESP8266 chip costs only \$2 USD a piece. That makes it an excellent choice for this system design.

The NodeMCU aims to simplify ESP8266 development. It has two key components.

- i. An open source ESP8266 firmware that is built on top of the chip manufacturer's proprietary SDK. The firmware provides a simple programming environment based on eLua (embedded Lua), which is a very simple and fast scripting language with an established developer community. For new comers, the Lua scripting language is easy to learn. And to add on NodeMCU can be programmed with the Android IDEtoo.
- ii. A development kit board that incorporates the ESP8266 chips on a standard circuit board. The board has a built-in USB port that is already wired up with the chip; hardware reset button, Wi-Fi antenna, LED lights, and standard-sized GPIO (General Purpose Input Output) pins that can plug into a bread board. Figure 2 below shows the NodeMCU developmentboard.

RELAYBOARD

A relay is an electromagnetic switch. It is activated when a small current of some microampere is applied to it Normally a relay is used in a circuit as a type of switch, an automatic switch. There are different types of relays and they operate at different voltages. When a circuit is built the voltage that will trigger it has to be considered. In this system the relay circuit is used to turn the appliances ON/OFF. The high/low signal is supplied from the NodeMCU microcontroller. When a low voltage is given to the relay of an appliance it is turned off and when a high voltage is given it is turned on. The relay circuit to drive four appliances in the Home automation system is shown below in figure 3. The number of appliances can be modified according to the user's requirements.



Fig -2: NodeMCU (ESP8266) Development Board

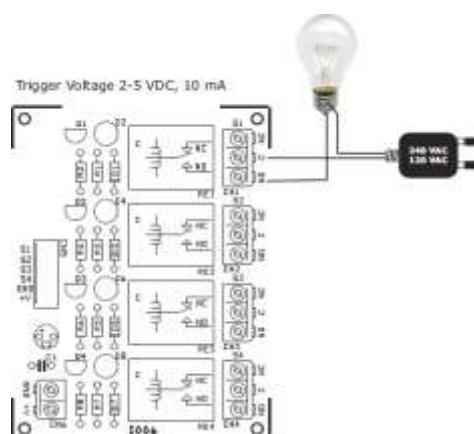


Fig -3: Relay Board Wiring Diagram

ULN 2803IC

ULN 2803 IC [6] is used as a relay driver. It is a High voltage, high current Transistor Array IC used especially with Microcontrollers where we need to drive high power loads. This IC consists of eight NPN

Darlington connected transistors with common Clamp diodes for switching the loads connected to the output. This IC is widely used to drive high loads such Lamps, relays, motors etc.

Most of the Chips operates with low level signals such as TTL, CMOS, PMOS, and NMOS which operates at the range of (0-5) V and are incapable to drive high power inductive loads. However this chip takes low level input signals (TTL) and uses that to switch/turn off the higher voltage loads that are connected to the output side.

The ULN2803 IC consists of eight NPN Darlington pair which provides the proper current amplification required by the loads. A Darlington pair has two transistors that act

as a single transistor providing high current gain. In this pair the current amplified by the first transistor is further amplified by the next transistor providing high current to the output terminal.

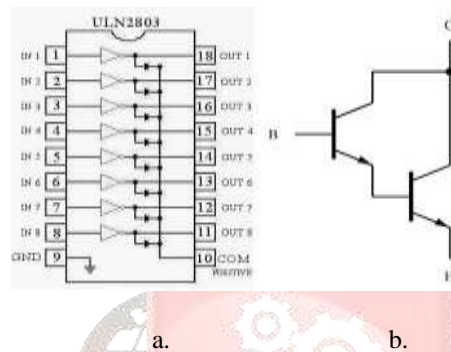


Fig -4: a. ULN 2803; b. Darlington Pair

SOFTWARE

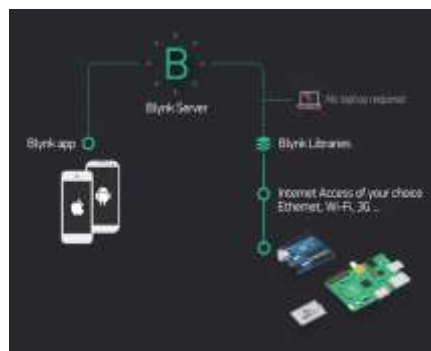
The software of the system proposed consists of mainly the Blynk Application and the IFTTT application.

BLYNKAPPLICATION

Blynk [2] is a Platform with iOS and Android apps to control Arduino, Raspberry Pi, Node MCU and several other boards over the Internet. Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other cool things.

Blynk App setup is required; we set it up as per the requirement. We begin by creating a project and then selecting the microcontroller we are using. After which we create the toggle buttons for each relay associated with the digital pins of the microcontroller. Once this is done, Blynk sends an authentication token to the registered email id for this particular project. This token should be noted and saved for its use while programming the Node MCU and setting up the IFTTT application.

Fig -5: Functioning of the Blynk Application



IFTTT APPLICATION

IFTTT [1] derives its name from the programming conditional statement “if this, then that.” IFTTT is both a website and a mobile app that launched in 2010 and has the slogan "Put the Internet to work for you". The idea is that you use IFTTT to automate everything from your favorite apps and websites to app-enabled accessories and smart devices. What the company provides is a software platform that connects apps, devices and services from different developers in order to trigger one or more automations involving those apps, devices and services. Here, IFTTT application is used to bridge the gap between the Google Assistant commands and the Blynk app.

Setting up the IFTTT application first requires logging in after which we need to create an applet and then “This”,

I.e. the trigger, here we select Google Assistant and then we will type in the commands to which the Google Assistant should respond and to this command it should control the appliance/relay associated with it. The response command from the Goggle Assistant can also be typed in asdesired.

After configuring the trigger, i.e. “This” of the application we need to configure the “That”. What should be done once the Google Assistant hears the command which we just configured? This is decided by setting “That” of the app. We click “That” and then select web hooks [7] and click connects. Web hooks will allow us to send commands to the Blynk Server. Now, in the URL we type the IP address of the Blynk server followed by the Authentation token sent by the Blynk and then the pin number of the microcontroller to which the device to be controlled is connected. The URL should be in the following format:

http://188.166.206.43/AuthToken/pin/CorrespondingDigitalPinNo

Then in the method we select ‘PUT’ and the content type is ‘Application/JSON’ [8] and in the body we write [“1”] to turn ON and [“0”] to turn OFF. This creates the action for the trigger i.e. the Google Assistant command. The action taken by it is simply sending a message to the Blynk app to either turn ON or OFF the concerned connected device.

Finally, the microcontroller is programmed with the actions it needs to do once it receives the signal from the Blynk application. Before that, the Blynk and the microcontroller should communicate and the communication is done via the internet and since the microcontroller, NodeMCU comes with inbuilt Wi-Fi module, it is programmed to connect to the desired network once plugged in. ‘C’ language is used to program the microcontroller and is programmed in the Arduino IDE[9].

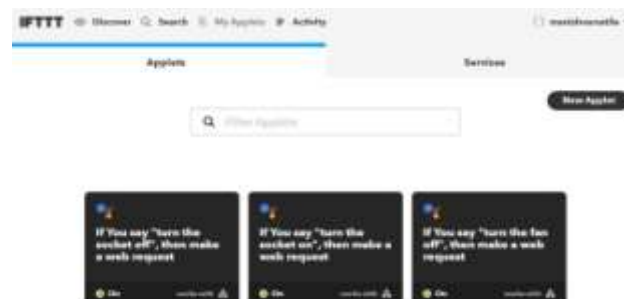
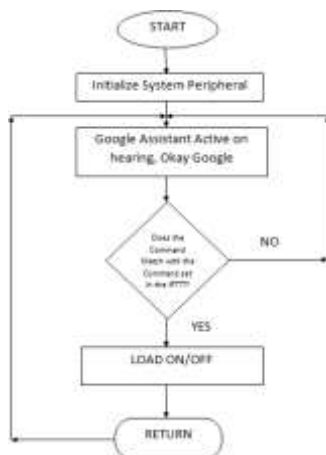


Fig -6: Screenshot of the IFTTT Application after Creating Several Applets

FLOWCHART**RESULT**

The result was positive and the system responded well. The diagram below shows the complete prototype implementation of the proposed system.

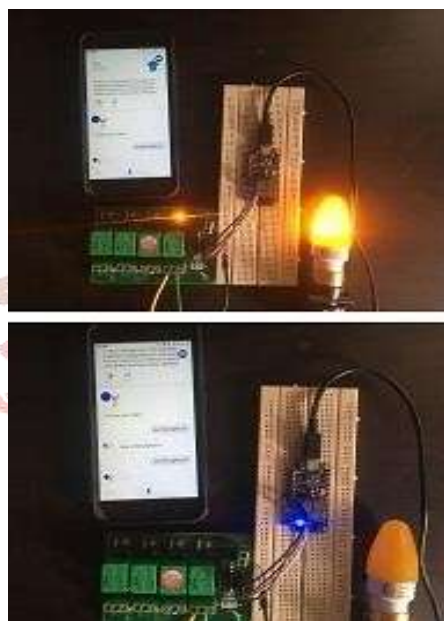


Fig -7: Light Turned ON and then OFF

NOTE: 5V/1A Output, Mobile Chargers were used to power the NodeMCU and the Relay Board

CONCLUSION AND FUTUREWORK

The aim of this paper was to propose a cost effective voice controlled (Google Assistant) home automation controlling general appliances found in one's home. The approach discussed in the paper was successful as GACHA's (Google Assistant Controlled Home Automation) design was successfully implemented. This system is highly reliable and efficient for the aged people and differently abled person on a wheel chair who cannot reach the switch for the switching ON/OFF the device and are dependent on others.

The future scope for GACHA can be huge. There are many factors to improve on to make GACHA more powerful, intelligent, scalable, and to become better overall for home automation. For example, controlling the

speed of the fan, more number of devices can be integrated, like a coffee machine, air conditioner etc. To make the system respond more faster own private Blynk server can be made. Well, no system is ever perfect. It always has a scope for improvement. One just needs to put on a thinking cap and try and make the system better.

REFERENCES

1. IFTTT: <https://ifttt.com/discoverhttps://www.pocketlint.com/SmartHome/SmartHomenews>
2. Blynk : <https://www.blynk.cc/> <https://docs.blynk.cc/>
3. NodeMCU: <https://nodemcu.readthedocs.io/en/masterhttps://iotbytes.wordpress.com/nodemcupinout/>
4. Google Assistant:
https://assistant.google.com/intl/en_inhttps://www.pocketlint.com/Apps/Appsnews/Googleappnews
5. IoT: <https://internetofthingsagenda.techtarget.com/definition/IoT-device>
6. ULN2803 IC working: <http://www.gadgetronicx.com/working-of-uln2803-ic/>
7. Webhooks: <https://webhooks.pbworks.com/w/page/13385124/FrontPage>
8. Application/JSON: <https://www.json.org/>
9. Arduino IDE: <https://www.arduino.cc/en/Guide/Environment>
10. Wikipedia:<https://www.wikipedia.org/>

