19Z009_INTERNET OF THINGS

HOME AUTOMATION WITH ESP8266

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

PSG COLLEGE OF TECHNOLOGY

(Autonomous Institution)

COIMBATORE – 641004

TEAM MEMBERS:

● Karan H	-20Z224
● Mohamed Fazil S F	-20Z232
● Senthuran Ravi	-20Z248
● Vishal R	-21Z436

ABSTRACT

Many home devices now have WiFi and can interact with other home devices, smartphone applications and home computers. An issue is that these devices cannot communicate with each other or require an additional device to do so and need an individual application on the smartphone to be controlled. A much better option is to unify these devices into one program/device that controls them. As an example, one can control the lights, microwave, oven, tv, air-conditioning and door locks through one application on the smartphone.

This project report presents a home automation system prototype that enables users to control a bulb remotely using an internet-based web server. The prototype consists of a NodeMCU device, single channel relay, and a bulb that is connected to the relay. NodeMCU is integrated with a Telegram server through Arduino IDE. With the help of a Telegram bot, users can control the bulb via the web server. The system provides users with the convenience of controlling a light source from anywhere as long as they have access to the internet. This project report explains the design, implementation, and testing of the home automation system prototype.

INTRODUCTION

Many home devices now have WiFi and can interact with other home devices, smartphone applications and home computers. An issue is that these devices cannot communicate with each other or require an additional device to do so and need an individual application on the smartphone to be controlled. A much better option is to unify these devices into one program/device that controls them. As an example, one can control the lights, microwave, oven, tv, air-conditioning and door locks through one application on the smartphone. This gives the consumer more control of their home, for example, it allows them to set up conditions for when the lights turn on, or if they are on their way home, to preheat the oven before they get home. Therefore, home automation can simplify many manual actions.

This project revolves around creating a home automation system prototype with the main focus being the ability to turn on/ off a bulb through the webserver (internet). The system consists of a single channel relay module , a web server and nodemcu esp8266.

How Does Home Automation Work?

Home automation works via a network of devices that are connected to the Internet through different communication protocols, i.e Wi-Fi, Bluetooth, ZigBee, and others. Through electronic interfaces, the devices can be managed remotely through controllers, either a voice assistant like Alexa or Google Assistant or an app. Many of these IoT devices have sensors that monitor changes in motion, temperature, and light so the user can gain information about the device's surroundings. To make physical changes to the device, the user triggers actuators, the physical mechanisms like smart light switches, motorized valves or motors that allow devices to be controlled remotely.

The proposed system consists of a single channel relay module , a web server and nodemcu esp8266.

NodeMCU is an open source platform based on ESP8266 which can connect objects and let data transfer using the Telegram application. Here , telegram bot is used to turn on or off the bulb . Single channel relay is connected to nodemcu and bulb. Nodemcu is integrated with a telegram server using Arduino IDE.

The proposed idea is to switch on / off the bulb using a single channel relay module, and nodemcu via web server. It provides security, safety and is accessible for all ages. It is also energy efficient.

Tools Used:

Software:

Arduino IDE:

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.

Telegram bot:

A web server uses HTTP Protocol (Hypertext Transfer Protocol) to respond to client requests made over the world wide web. Telegram Messenger is a cloud-based instant messaging and voice over IP service. You can easily install it in your smartphone (Android and iPhone) or computer (PC, Mac and Linux). It is free and without any ads. Telegram allows you to create bots that you can interact with.

The ESP32/ESP8266 will interact with the Telegram bot to receive and handle the messages, and send responses. In this tutorial you'll learn how to use Telegram to send messages to your bot to control the ESP outputs from anywhere (you just need Telegram and access to the internet).

Hardware:

NodeMCU ESP8266:

- NodeMCU is an open source Lua based firmware.
- It has one analog pin (A0) and nine digital pins (D0 – D9).
- It also has the support for serial communication protocols such as UART, SPI, I2C etc… Hence it can be used for interfacing with devices such as sensors, displays etc..
- The main feature of NodeMCU is its inbuilt WiFi functionality.

- It can be easily programmed with the help of Arduino IDE (Integrated Development Environment).
- The other way of programming NodeMCU is using ESPlorer IDE with the help of Lua programming language. Luas is an open source, light weight, embedded scripting language built on top of C programming language.
- It is mainly used for IOT based projects.

Single channel relay module:

- A relay is an electrically operated switch. Usually, like any other switch, it can be turned on or off, with or without current. It can be controlled through low voltages such as 3.3V provided by ESP8266 GPIOs and helps to control high voltages.
- Relay is an electro-mechanical device that uses an electric current to open or close the contacts of a switch.
- The relay module has two connectors, each of them has three sockets: common (COM), Normally Closed (NC), and Normally Open (NO).

- COM: main voltage connection (current that you want to control).
- Normally Closed: In normally closed configuration the relay is closed by default. The NC and COM pins are connected, which means the current is flowing until you send a signal from the ESP8266 to the relay module. To open the circuit and stop the flow of current.

Working:

As per the circuit connections,

- One end of the power supply is connected to the NO (Normally Open) pin of the relay module and the other end of the power supply is connected to the 12V bulb.
- The COM (common) pin is connected to the other end of the bulb.
- By default, the relay acts as an open switch in this circuit (No connection between NO pin and COM pin).

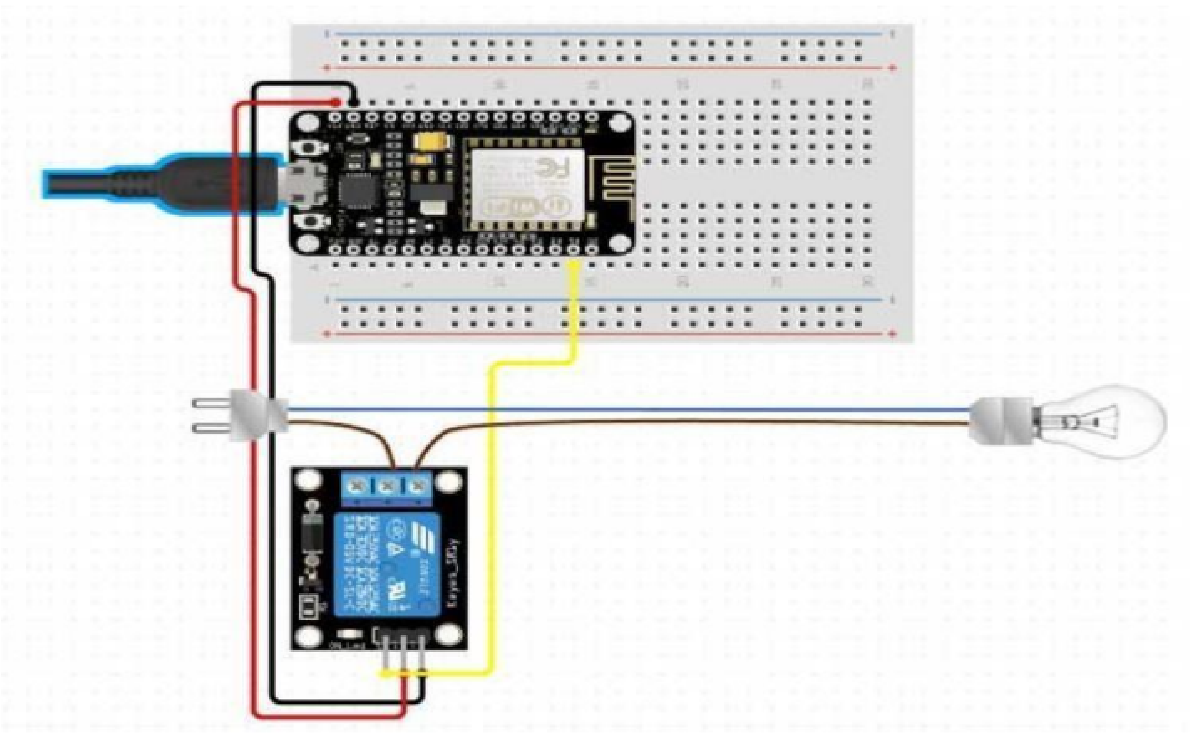
- VCC and GND pins of the relay module are connected to 3.3V and GND pins of NodeMCU.
- The digital pin D0 of NodeMCU is connected to the INPUT pin of the relay module.

Principle:

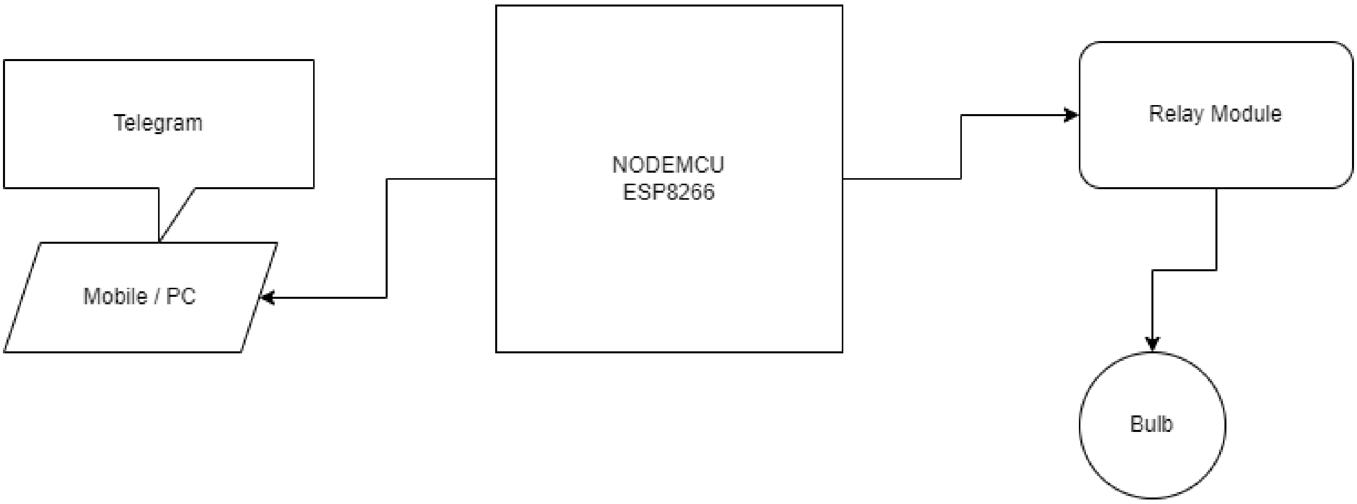
We're using a NO configuration. Hence, we need to send a High signal to stop the current flow, and a Low signal to let the current flow.

- When D0 (Control Signal) is HIGH, the Relay module acts as a closed switch. Hence the bulb glows.
- When D0 (Control Signal) is LOW, the Relay module acts as an open switch. Hence, the bulb does not glow.

Diagrams:



Block diagram :



Results:

In the telegram application we start the Automation bot using the start command.Then the bot responds with a messages which contains a set of applications that it can perform, here we have turned on the 1st bulb, i.e the 1st Relay, once we select the command in the telegram bot we can see that the 1st relay has been turned on.Performing the same operation for the 3rd relay we can turn on the 3rd relay.

We can also identify whether the bulb is turned on or off using the state command in the telegram bot. Here we are finding out whether the 1st and 3 rd bulbs are turned on or off using the state commands of the respective bulbs. We receive a message saying that only Bulb 1 is on and bulb 3 is turned off.

Conclusion and Future scope:

Thus the result has been obtained by switching on/ off the bulb through the web server. This Home automated system is energy efficient and can be implemented as it provides safety, security and can be accessed by all ages. This field of home automation is fastly emerging in technology making homes safer and better places to live.

The same can be extended to controlling other appliances such as microwave, oven, tv, air-conditioning and so on, so that these features help users to virtually monitor and control home attributes like lights, entertainment systems, security, climate control, etc.

Reference:

Project idea:
https://www.electroniclinic.com/esp32-web-server-based-home-automation-circuit-diagram-and-programming/

https://www.hackster.io/ashshaks/esp8266-web-server-python-home-automation-f0d8f2

https://www.youtube.com/watch?v=jDaRPsvvcz4

About ESP8266:

https://randomnerdtutorials.com/telegram-control-esp32-esp8266-nodemcu-outputs/

About Iot products:

https://www.security.org/home-automation/