

A  
**Design Project Report on**  
**CONTROLLING HOME APPLIANCES THROUGH MOBILE**

Submitted in partial fulfilment of the  
Requirements for the award of the  
Degree of

BACHELOR OF ENGINEERINGIN  
COMPUTER SCIENCE & ENGINEERING

By

N.Naresh (1602-19-733-033)  
K.Bhikshapathi (1602-19-733-011)



Department of Computer Science &  
EngineeringVasavi College of Engineering  
(Autonomous) (Affiliated to Osmania  
University)  
Ibrahimbagh, Hyderabad-

312021



Edit with WPS Office

## **DECLARATION BY THE CANDIDATE**

We, P.Pranaya, Ch.Sahithi , bearing hall ticket number 1602-19-733-033,1602-19-733-011 hereby declare that the project report entitled "Controlling Home Appliances Using Mobile" Department of Computer Science & Engineering, VCE, Hyderabad, is submitted in partial fulfilment of the requirement for the award of the degree of Bachelor of Engineering in Computer Science & Engineering.

This is a record of bonafide work carried out by me and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

N.Naresh

1602-19-733-033.

K.Bhikshapathi

1602-19-733-011



Edit with WPS Office

**Vasavi College of Engineering  
(Autonomous)(Affiliated to Osmania  
University) Hyderabad-500 031**

**Department of Computer Science & Engineering**



**BONAFIDE CERTIFICATE**

This is to certify that the project entitled "Controlling Home Appliances Using Mobile" being submitted by N.Naresh, bearing 1602-19-733-033, in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in Computer Science & Engineering is a record of bonafide work carried out by him/her under my guidance.

Dr. T. Adilakshmi,  
Professor & HOD,  
Dept. of CSE



Edit with WPS Office

**Vasavi College of Engineering  
(Autonomous)(Affiliated to Osmania  
University) Hyderabad-500 031**

**Department of Computer Science & Engineering**



**BONAFIDE CERTIFICATE**

This is to certify that the project entitled "Controlling Home Appliances Using Mobile" being submitted by k.bhikshapathi, bearing 1602-19-733-011, in partial fulfilment of the requirements for the award of the degree of Bachelor of Engineering in Computer Science & Engineering is a record of bonafide work carried out by him/her under my guidance.

Dr. T. Adilakshmi,  
Professor & HOD,  
Dept. of CSE



Edit with WPS Office

## **ACKNOWLEDGEMENT**

We would like to express our special thanks of gratitude to our professor B.Symala as well as our Head of The Department Dr. T. ADILAKSHMI who gave us the golden opportunity to do this wonderful project. This helped us in doing a lot of Research and we came to know about so many new things for which we are thankful.

We are thankful and fortunate enough to get constant encouragement, support and guidance which helped us in successfully completing our project work.

We are pleased to acknowledge for the valuable guidance during this project.



Edit with WPS Office

## ABSTRACT

Home automation is one of the fundamental developing industries that may alternate the way humans stay. A few of these home automation systems target those seeking luxurious and sophisticated domestic automation structures; others goal those with special needs like the aged and the disabled. Internet of Things based home automation project aims to automate the functioning of household appliances and objects over the Internet. All the household objects that are connected over the Internet of Things network can be controlled and operated through your smartphone. This is not only convenient but also gives more power to the user to control and manage household appliances from any location in the world. The Home Automation increases convenience, safety, saves precious power, time and money.

An analog switch is mounted on the walls. Using them is a tedious task as they need to be physically fit is pressed every time something has to be turned on or off. This problem is being replaced by an intelligent strategy using mobile phone switches. Smart current changes found in the file the market is very expensive and requires additional devices as their operating hubs. This project tells us about managing hand-operated switches through a mobile app called blynk. Switch was terminated by NodeMCU with built-in Wi-Fi. It can use this to enable or disable it to change. User controls microcontroller via blynk system. The microcontroller then controls the switch according to the instructions received by the user and also updates the user with a change mode after administrative work was done.



Edit with WPS Office

## TABLE OF CONTENTS

1. Introduction.....	1
2. Introduction to NODE MCU.....	2
3. Introduction to Relay Module.....	4
4. Software Requirements.....	5
5. Hardware Requirements.....	8
6. System Design.....	12
7. Analysis and Implementation of code.....	13
8. Implementation of program.....	14
9. Results.....	15
10.....	Reference 15



Edit with WPS Office

## INTRODUCTION

### Overview

Traditional light control systems only support manual control (only when we are present at that particular place) while operators cannot operate them remotely in real time, which causes great inconvenience to management and maintenance.

### Problem Definition

Internet of Things based home automation project aims to automate the functioning of household appliances and objects over the Internet. All the household objects that are connected over the Internet of Things network can be controlled and operated with your mobile. This is not only convenient but also gives more power to the user to control and manage household appliances from any location in the world. The Home Automation increases convenience, safety, saves precious power, time and money.

Traditional light control system supports manual control while operators cannot operate them with the remote in real time, which causes great inconvenience for management and maintenance. By this technology we can control our home appliances through our mobile anywhere in the world with the help blynk application, relay module and ESP8266.



Edit with WPS Office

## INTRODUCTION TO NODE MCU

NodeMCU is an open-source platform, its hardware design is open for edit/modify/build. NodeMCU Dev Kit/board contains ESP8266 wifi enabled chip. The ESP8266 may be a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol.[12] More details are often found on ESP8266 Documentation. NodeMCU uses an on-module flash-based SPIFFS (Serial Peripheral Interface Flash File System) filing system.

NodeMCU is implemented in C and is layered on the Espressif NON-OS SDK. The firmware was initially developed as could be a companion project to the favored ESP8266-based NodeMCU development modules, but the project is now community-supported, and therefore the firmware can now be run on any ESP module. Generally, we can find NodeMCU Dev boards making Amica, DOIT, Login & D1 mini /Wemos, etc. in the market. Amica produces NodeMCU ESP8266 Development Boards v1.0(Version2) with designed hardware specifications

NodeMCU is an Arduino like device. Its main component is ESP8266. It has Programmable pins. It has built-in WiFi. It can get power through a micro-USB port. Its cost is low. It is often programmed through multiple programming environments. ESP8266 NodeMCU requires 2.5V to 3.6V Operating Voltage, Onboard 3.3V- 600mA regulator, 80mA Operating Current, 20 µA Current during Sleep Mode. Power to the ESP8266 NodeMCU is supplied via the onboard Micro USB connector. ESP8266 NodeMCU is equipped with 32 Kb RAM, 80 Kb DRAM, and 200 Kb Flash Memory. ESP8266 NodeMCU has Pin D0 to Pin D10 Digital Pins, 12 PWM Pins, A0 Analog Pin. It has 5 Ground Pins, 3 number of 3.3 V Pins, 1 Vin Pin for adding 1 external supply of +5V which is not connected to USB. The ESP8266 NodeMCU has a total of 17 GPIO pins. These pins are often assigned to all or any kinds of peripheral duties, including one 10-bit ADC channel, Two No. of UART interface which is used to load code serially, four PWM pins for dimming LEDs or controlling motors, SPI and I2C interface to hook up all sorts of sensors and peripherals, I2S interface for adding sound to project. ESP8266 features a pin multiplexing feature (Multiple peripherals multiplexed on one GPIO pin). Meaning a single GPIO pin can act as PWM/UART/SPI. NodeMCU has an RST button to Reset the ESP8266 chip, one FLASH button to Download new programs and one Blue LED that is user-programmable.

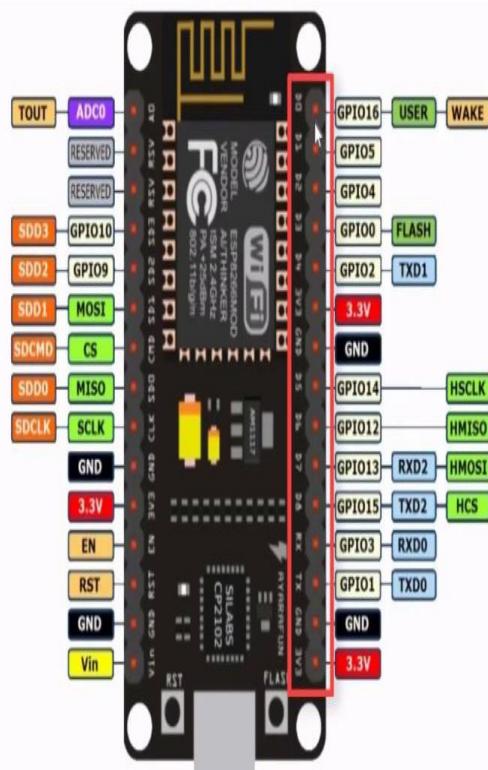


Edit with WPS Office

## ESP8266



ESP 8266 PIN DIAGRAM



Edit with WPS Office

## INTRODUCTION TO RELAY MODULE

A relay is an electrically operated switch that can be turned on or off, allowing the current to pass or not, and can be controlled by low voltages, such as the 5V supplied by Arduino pins.

This transmission module has one channel. There are other models with two channels, four and eight. This module should be powered by 5V, suitable for use with a small controller. There are other transmission modules powered by 3.3V, suitable for ESP32, ESP8266, and other microcontroller. we have 5V 2 channel transmission module,

5V 1-channel relay module, 5V 8-channel relay module, 3.3V 1-channel relay module.

The six pins on the left side of the transmission module connect the mains, and the right-hand anchors connect the part that requires less electricity - the Arduino pins.

The high-powered side has two connectors, each with three sockets: standard (COM), normally closed (NC), and always open (NO).

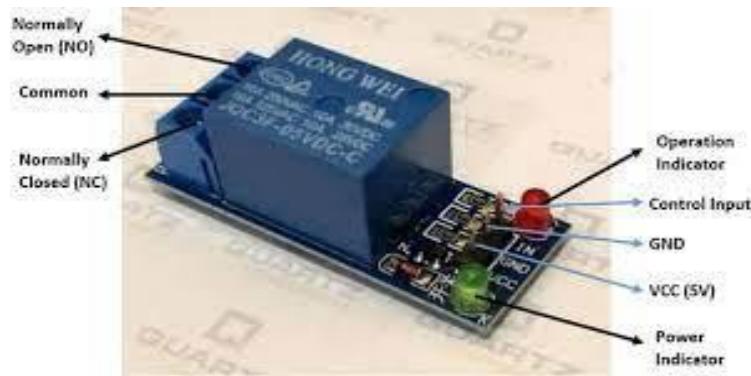
The low-power side has a set of three anchors set to the right consisting of a VCC, a GND power supply module, and anchor pins.

The connection between the transfer module and Arduino is really

easy: GND: goes down

VCC: up to 5V

INT: input pin in esp8266



## **SOFTWARE REQUIREMENT**

### **BLYNK APPLICATION**

With Blynk, you can create smartphone applications that allow you to easily interact with microcontrollers or even full computers such as the Raspberry Pi.

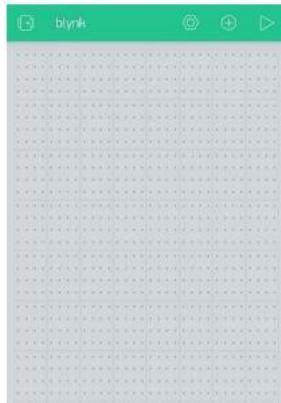
The main focus of the Blynk platform is to make it super-easy to develop the mobile phone application. As you will see in this course, developing a mobile app that can talk to your Arduino is as easy as dragging a widget and configuring a pin.

With Blynk, we can control an LED or a motor from your mobile phone with literally zero programming.

#### **SETTING UP BLYNK APPLICATION:**

##### **Step 1:**

Create New Project and click on Home Screen



##### **Step 2:**

From Widget Box select Button

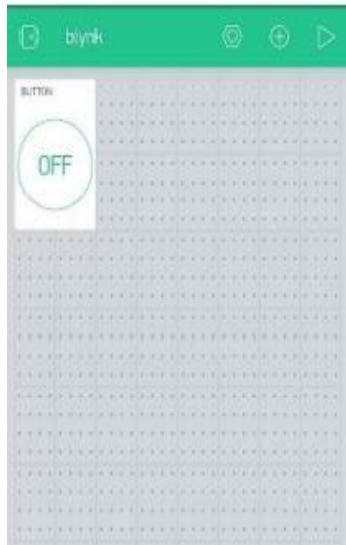


Edit with WPS Office



### Step 3:

Now click on the button to change the settings



### Step 4:

You can select the pin you need to control, your button name here. Once done click on the backbutton so that it will go back to the home page.



Edit with WPS Office



### Step 5:

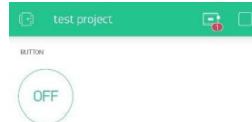
Click on execute button on the right top.



Edit with WPS Office

## Step 6:

The Project is ready to operate. The error symbol will be cleared when you upload code to yournodemcu and once it is connected to blynk application.

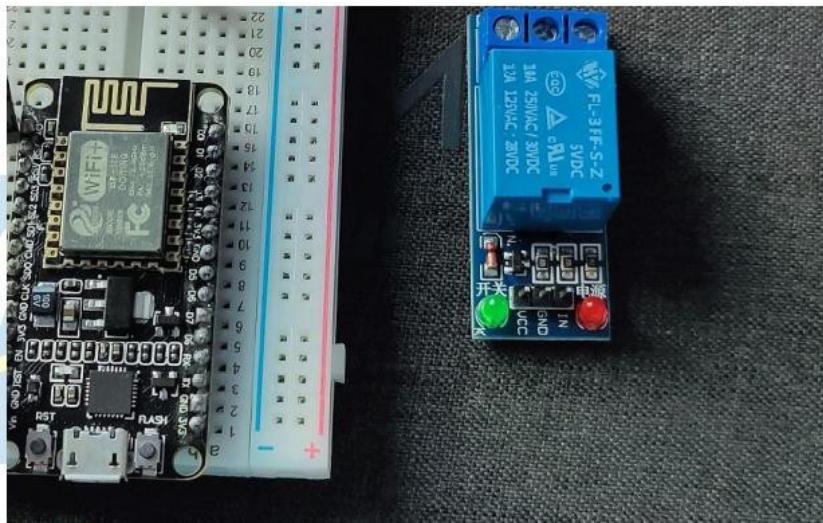


## HARDWARE REQUIREMENT

### Interfacing Relay and Node

#### MCU Step 1:

Relay has 3 pins Voltage (Vcc), Ground (gnd), Input pin (IN)

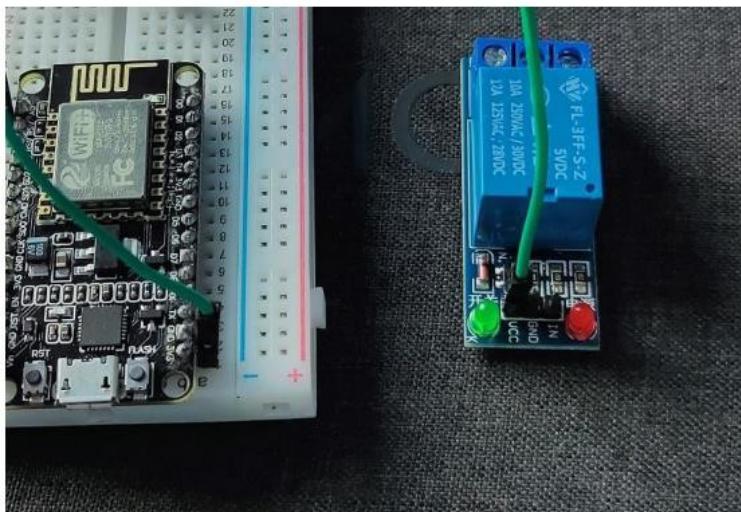


#### Step 2:

Connect Vcc of Relay Module to 3V3 of Node MCU

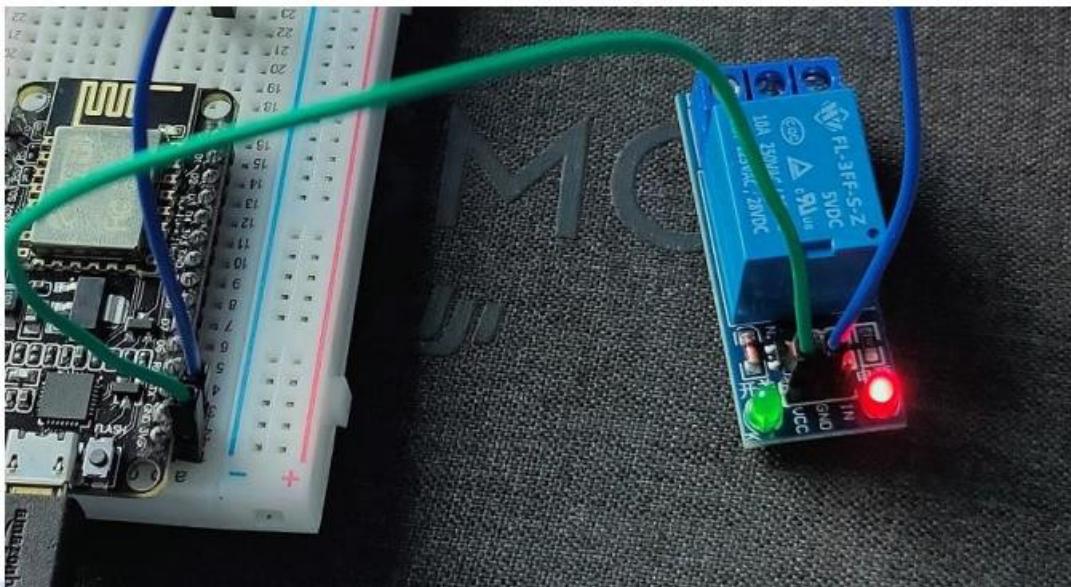


Edit with WPS Office



### Step 3:

Connect Gnd of Relay Module to Gnd of Node MCU (Once Voltage and ground are connected then it will be powered up and red Led will glow)

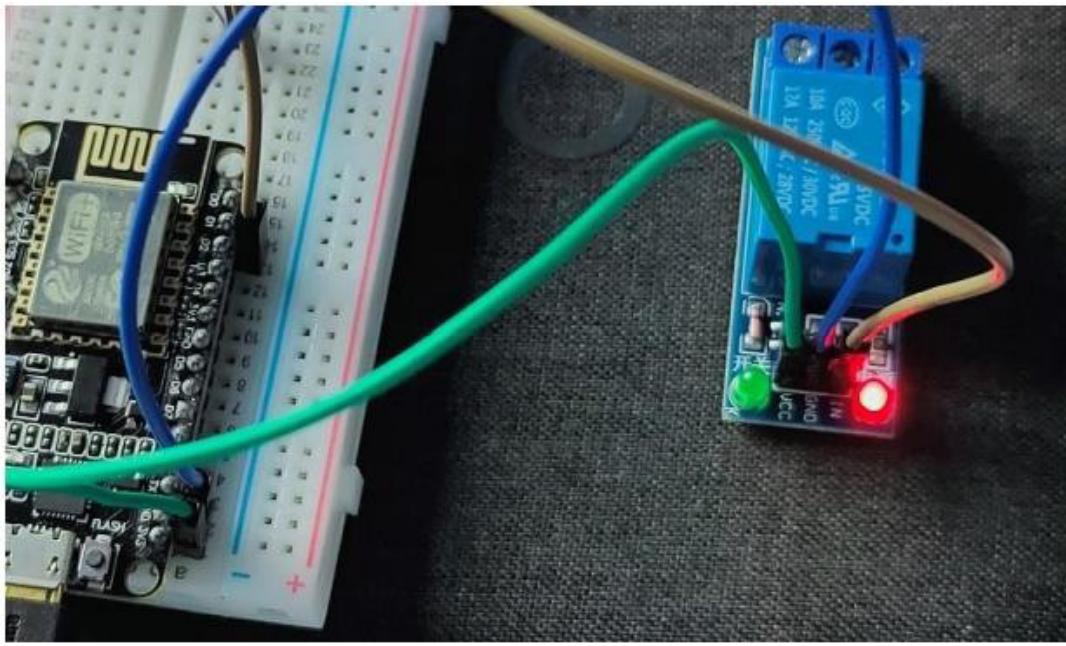


### Step 4:

Connect IN of Relay Module to Digital pin of Node MCU (Through this pin relay module will receive signal from Node MCU)

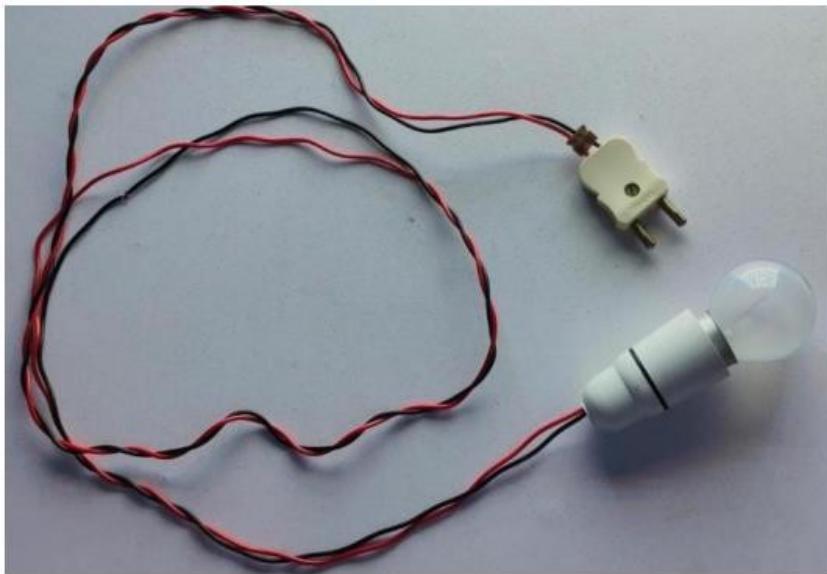


Edit with WPS Office



### Step 5:

Connect 2-pin plug and holder on either sides of the electrical wire.

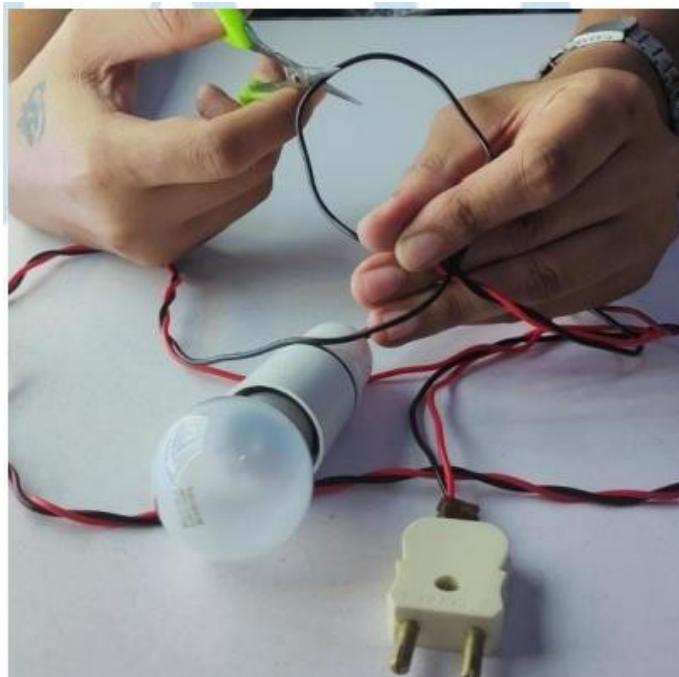


### Step 6:



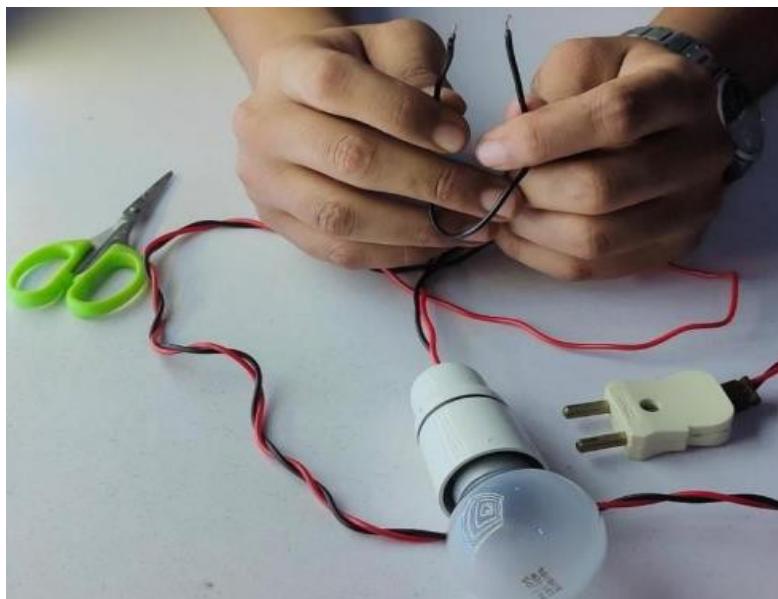
Edit with WPS Office

Disconnect one of the electrical wire in the circuit in-between



### Step 7:

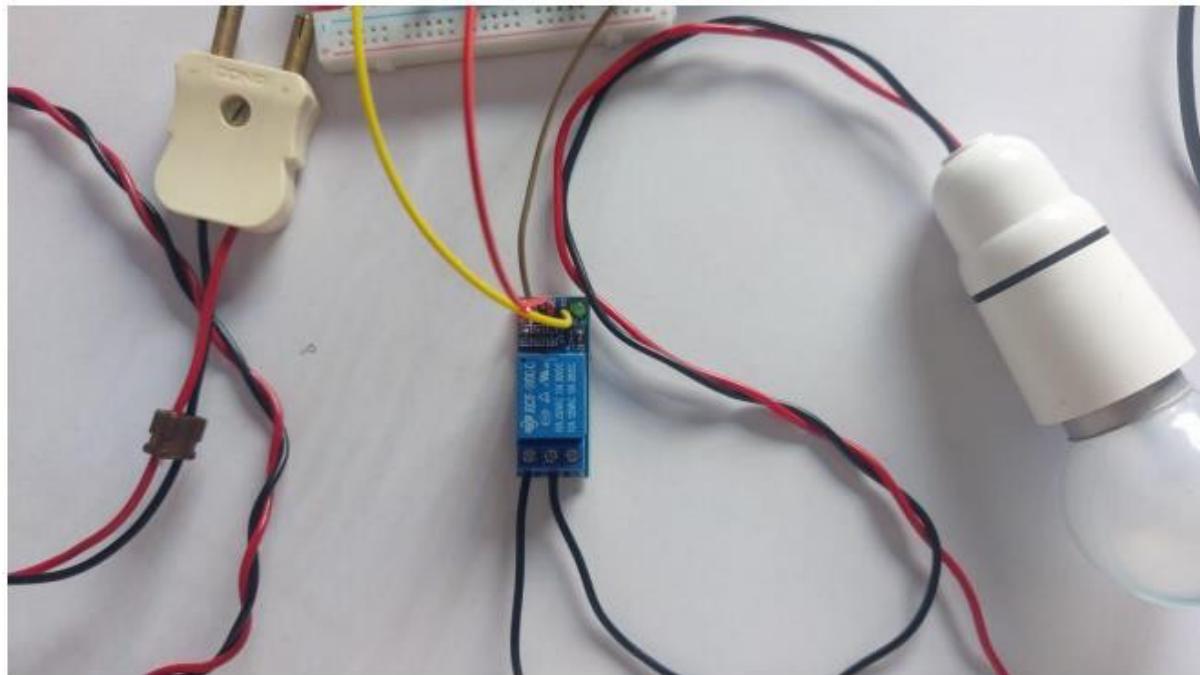
These two ends of the high voltage end of relay.



Edit with WPS Office

## Step 8:

Connect one in the middle and other on either left/right.



## SYSTEM DESIGN

### (i) Input: Mobile Application

- We will build a mobile application which has a button to control our appliance.
- Application sends signal as input to the micro controller

### (ii) Processing: Node MCU

- Microcontroller
- Receive input from mobile and send output to relay module

### (iii) Output: Relay Module

- Acts as a switch to control supply to the light
- Receive instructions from the microcontroller.



Edit with WPS Office

## ANALYSIS AND UNDERSTANDING OF IMPLEMENTATION CODE:

Starting with the complete overview of the project directories with a given code which is been used in the compiling and execution of the project:

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

char auth[] = "Auth_Code";
char ssid[] = "Wifi_ID";
char pass[] = "Wifi_pwd";

void setup()
{
    Serial.begin(9600);
    Blynk.begin(auth, ssid, pass);
}

void loop()
{
    Blynk.run();
}
```

Including blynk libraries

Blynk app authentication code

WiFi/Hotspot user ID

WiFi/Hotspot pwd

Setup serial monitor

Node MCU uses wifi user ID and PWD to get connected to Internet and uses auth code to connect to our application

Continuously fetches information from the blink app



Edit with WPS Office

## IMPLEMENTATION OF PROGRAM

```
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>

#include
<BlynkSimpleEsp8266.h>

char auth[] =
"srwSmVNhKMdP6mTW3QLr80Clzqu5CrOV";char ssid[]
= "DIR-615-208B";

char pass[] =
"12334455667@";void setup()

{
    Serial.begin(9600);

    Blynk.begin(auth, ssid,
pass);

}

void loop()
{
    Blynk.run();

}
```



Edit with WPS Office

## RESULTS:

Off State



On State



## References:

[https://www.researchgate.net/publication/337656615\\_Internet\\_of\\_Things\\_and\\_Nodemcu\\_A\\_review\\_of\\_use\\_of\\_Nodemcu\\_ESP8266\\_in\\_IoT\\_products](https://www.researchgate.net/publication/337656615_Internet_of_Things_and_Nodemcu_A_review_of_use_of_Nodemcu_ESP8266_in_IoT_products)

<https://randomnerdtutorials.com/guide-for-relay-module-with-arduino/>

<https://techexplorations.com/guides/blynk/1-what-is-blynk/>



Edit with WPS Office



Edit with WPS Office