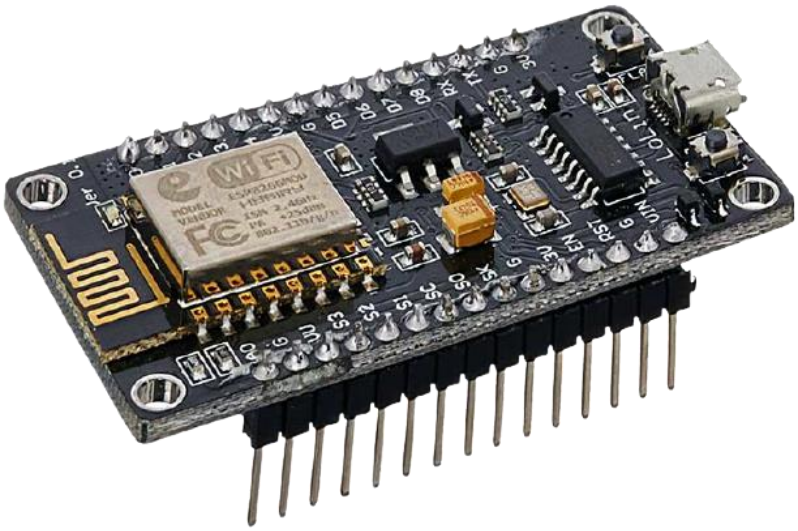


# Home Automation (Using Node MCU) Working on SDG Goal 4& 11


## Imran Khan Kerio

## Team Members

**Muhammad Fasih CSC-19F-089**

**Mehar Khan Niazi CSC-19F-050**

**Muhammad Azhar CSC-19F-069**

# Home Automation

(Using Nodemcu)Working on Sustainable Development Goal 4 & 11

## Objective

Technology is rapidly evolving with the passage of time. People and daily life routines rely heavily on the internet. While the cost of living continues to rise, there is a rising emphasis on using technology to reduce those costs. With this in mind, the Smart Home project enables the user to construct and manage a home that is smart enough to conserve energy while also delivering more automated applications. A smart house will make use of its environment and provide smooth management whether the user is there or not. With this benefit, we can be confident that our home is working optimally in terms of energy efficiency. The world's current challenge is to create smart homes and efficient energy systems (SDG goal 4 & 11)

**Home is where the comfort is. The real comfort lies in living smart & simple**

## Working

- The user has total control over all parts of house that can be controlled remotely.
- The automation system will be able to be operated from any location on the planet.
- The System will also detect objects and records it when motion found.

## Scope

IoT will add 15 trillion dollars to the global economy over the next 20 years

## Tools & Language

- Node mcu Board
- 5V 4-Channels Relay Module
- Jumper wires
- USB Cable
- 12V adaptor

Arduino IDE 1.8.16 & Programing Language is C++

## Operating System Involvement

The operating system involvement in this project is between the controlling device, Nodemcu board which we are using is a microcontroller which is a firmware ,works like RTOS that helps us rapidly switch between various running parts of our code. Run directly on the bios program which is uploaded using usb cable . The OS is also involved in transferring commands from the controller device to the receiver which give commands to pulse high or low.

The another part is of Smart security camera if the motion detection is found then the system start recording and create the mp4 file and save it.

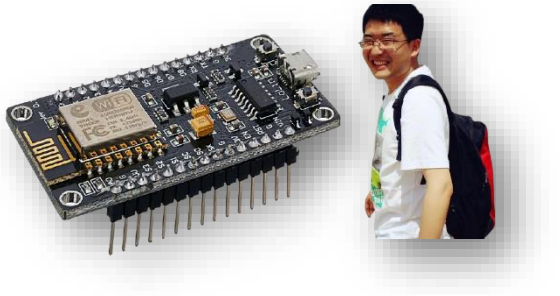
# IOT:

If we look around, we'll find at least one thing that has the ability to connect to the internet. It may be our phone.It may be a laptop, our TV, or even your fridge, internet of things generally refers to the collection of all those devices. But now just, you can argue that anything that has the ability to connect to the internet and collect and share data is a part of internet of things or IOT in short.

# Device :

## Node Mcu:

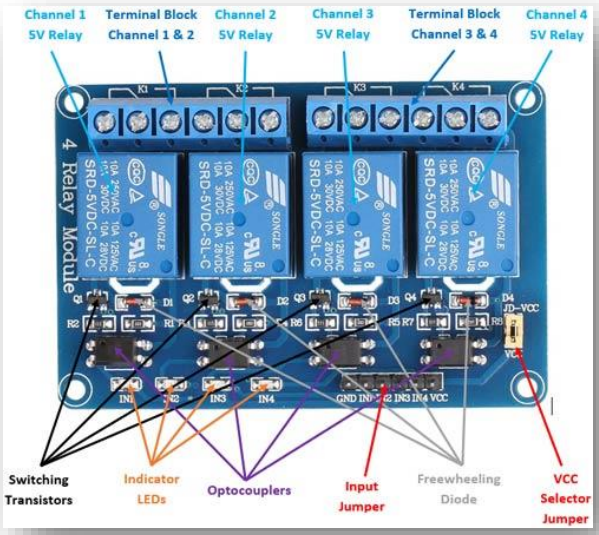
The NodeMCU (Node MicroController Unit) is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) . It initially included firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which was based on the ESP-12 module ,and built on the Espressif Non-OS SDK for ESP8266.Founded by Huang Rui.

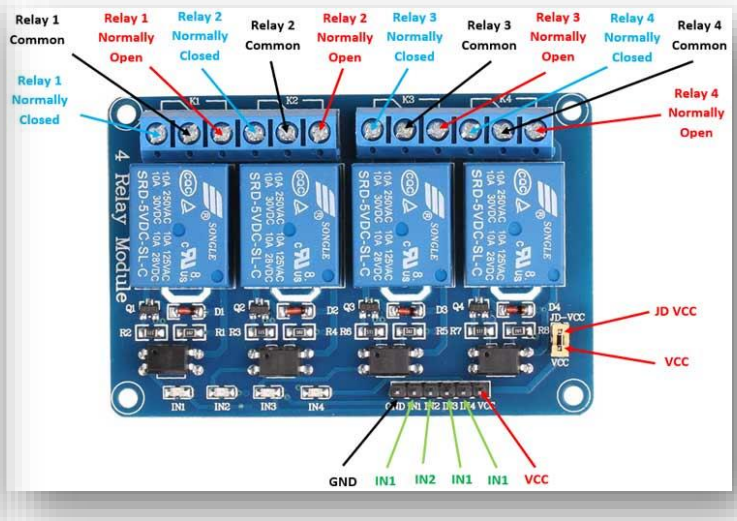


## Relay Module:

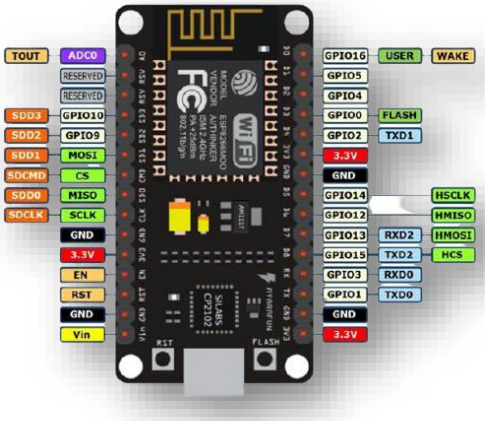
4 channel Relay Module

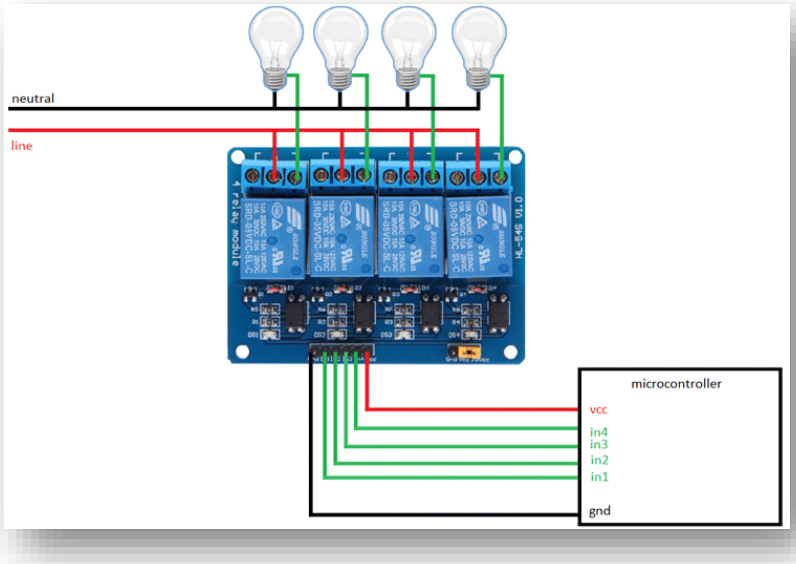
The four-channel relay module contains eight 5V relays and the associated switching and isolating components, which makes interfacing with a microcontroller or sensor easy with minimum components and connections. Each relay on the board has the same circuit, and the input ground is common to all eight channels.

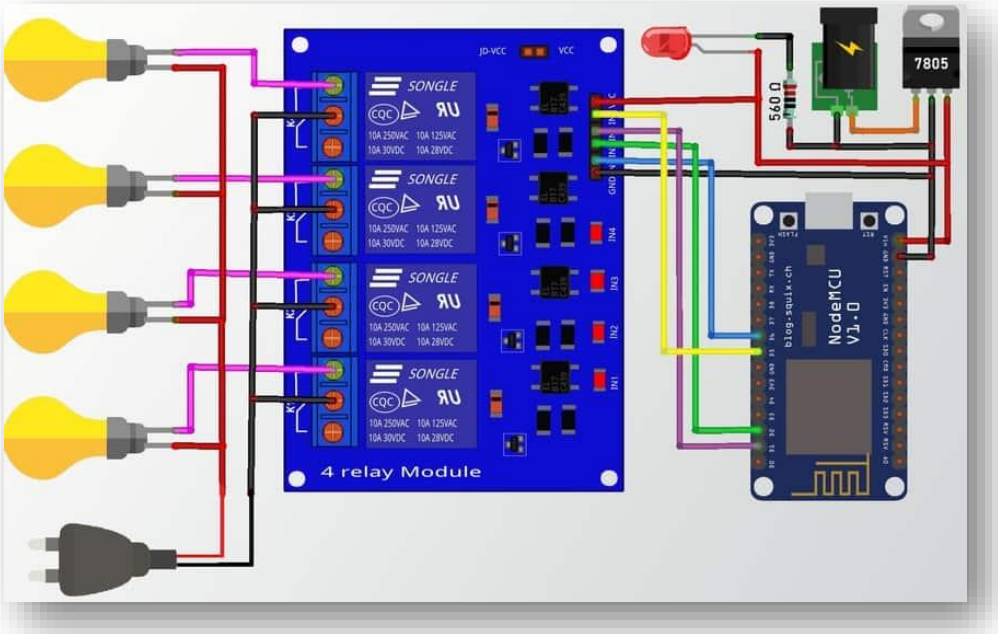


# Circuit Diagram:



# Code: (Node MCU Boarad)

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

char auth[] = "FOGcF7HORvCsF_eXy_ZrCimYN7hsBMyN";
char ssid[] = "Niazi House_EXT";
char pass[] = "Niazi7979";

 int relay1 = 12;  // D6 pin
 int relay2 = 14;  // D2 pin
 int relay3 = 4;   // D1 pin
 int relay4 = 5;   // D5 pin

void setup()
{
 Serial.begin(115200);
 pinMode(LED_BUILTIN, OUTPUT);
 pinMode(relay1,OUTPUT);
 pinMode(relay2,OUTPUT);
 pinMode(relay3,OUTPUT);
 pinMode(relay4,OUTPUT);

 digitalWrite(relay1, HIGH);
 digitalWrite(relay2, HIGH);
 digitalWrite(relay3, HIGH);
 digitalWrite(relay4, HIGH);
 Blynk.begin(auth, ssid, pass);
}
void loop()
{
 digitalWrite(LED_BUILTIN, HIGH);  // turn the LED on (HIGH is the voltage level)
 delay(1000);
 digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the voltage LOW
 delay(1000);
 Blynk.run();
}
```

# Code: (Object Detection recording )

# Opencv

```
import cv2
import time
import datetime

cameraseletion = cv2.VideoCapture(0)

face_cascade = cv2.CascadeClassifier(
    cv2.data.haarcascades + "haarcascade_frontalface_default.xml")
body_cascade = cv2.CascadeClassifier(
    cv2.data.haarcascades + "haarcascade_fullbody.xml")

detection = False
detection_stopped_time = None
timer_started = False
SECONDS_TO_RECORD_AFTER_DETECTION = 5

frame_size = (int(cameraseletion.get(3)), int(cameraseletion.get(4)))
fourcc = cv2.VideoWriter_fourcc(*"mp4v")

while True:
    _, frame = cameraseletion.read()

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, 1.3, 5)
    bodies = face_cascade.detectMultiScale(gray, 1.3, 5)
```

```
    if len(faces) + len(bodies) > 0:
        if detection:
            timer_started = False
        else:
            detection = True
            current_time = datetime.datetime.now().strftime("%d-%m-%Y-%H-%M-%S")
            out = cv2.VideoWriter(
                f"{current_time}.mp4", fourcc, 20, frame_size)
            print("Started Recording!")
    elif detection:
        if timer_started:
            if time.time() - detection_stopped_time >= SECONDS_TO_RECORD_AFTER_DETECTION:
                detection = False
                timer_started = False
                out.release()
                print('Stop Recording!')
        else:
            timer_started = True
            detection_stopped_time = time.time()

    if detection:
        out.write(frame)
    cv2.imshow("Camera", frame)
    if cv2.waitKey(1) == ord('q'):
        break
out.release()
cameraseletion.release()
cv2.destroyAllWindows()
```

# Code and Research Papers:

Code and research paper available on : https://techshoor.com/homeautomationos

Presentation Available on : https://techshoor.com/osproject

# References:

https://youtu.be/KeaeuUcw02Q (Iot Architecture)

https://docs.google.com/presentation/d/14OcW4HfS2i1Db1uKOU6SrckFEFjhSLMgfnHYB3XlEZo/edit#slide=id.g58bcf95047_0_56 (15 trillion Doller)

https://www.xiaomistore.pk/mi-smart-home-min0-max0-attr0-1-sort_order-ASC/ (Price prediction)

https://www.geeksforgeeks.org/architecture-of-internet-of-things-iot/ (IOT layers)

https://create.arduino.cc/projecthub/electropeak/getting-started-w-nodemcu-esp8266-on-arduino-ide-28184f (Nodemcu detail)

https://components101.com/switches/5v-eight-channel-relay-module-pinout-features-applications-working-datasheet (relay module)

https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet ( Node MCU)

https://opencv.org/ (Open CV)