Voice controlled Home Automation

By,

Rahul Reddy & **Rohit Prabhu**
Contact Us: embeddedenthusiasts@gmail.com

Contents

1 Description

In this project, voice commands are being used to control the electrical appliances.Both the Server and Client sides are entirely coded in python. A wireless connection is established by using TCP/IP Sockets. It makes use of the Google Speech Recognition API, in order to convert speech to text.

2 Prerequisites

2.1 Software

1. Linux system with latest version of Python3(Recommended Latest Version of Ubuntu)
2. RShell (Available only on Linux)
3. Libraries: PyAudio & SpeechRecognition
4. Windows System (Only for Firmware Changing)

2.2 Hardware

1. N-Channel Relay (N - No. of appliances as you need to connect)
2. ESP-8266 (NodeMCU) (Flashed with micropython)
3. Microphone (If your PC's microphone isn't that efficient)
4. Jumper Wires (Male to Male, Male to Female, Female to Female)

(Only For Testing):

(a) 2-pin Plug
(b) 2-pin External Socket
(c) 1 meter of connecting wires
(d) Electrical appliance like Table Fan, Hair-dryer, etc.

3 Installation

3.1 Linux System Installation (Ubuntu)

(If you have already installed Ubuntu as the Virtual Machine, you can skip this section.)

To install Ubuntu as the Virtual Machine, Please click here to watch this video and follow the steps given in the video.

3.2 Flashing NodeMCU(ESP8266) with MicroPython

To flash NodeMCU(ESP8266) with MicroPython, Please click here to watch this video and follow the steps given in the video.

3.3 PyAudio Library Setup

In order to install PyAudio the following commands are to be typed in the terminal in the same order as given below: (Recommended on Ubuntu)

1. `sudo apt-get install libasound-dev portaudio19-dev libportaudio2 libportaudiocpp0`
2. `sudo apt install python3`
3. `sudo apt install python3-pip`
4. `pip3 install pyaudio`

3.4 SpeechRecognition Library Setup

In order to install SpeechRecognition the following command is to be typed in the terminal as given below:

```
pip3 install speechrecognition
```

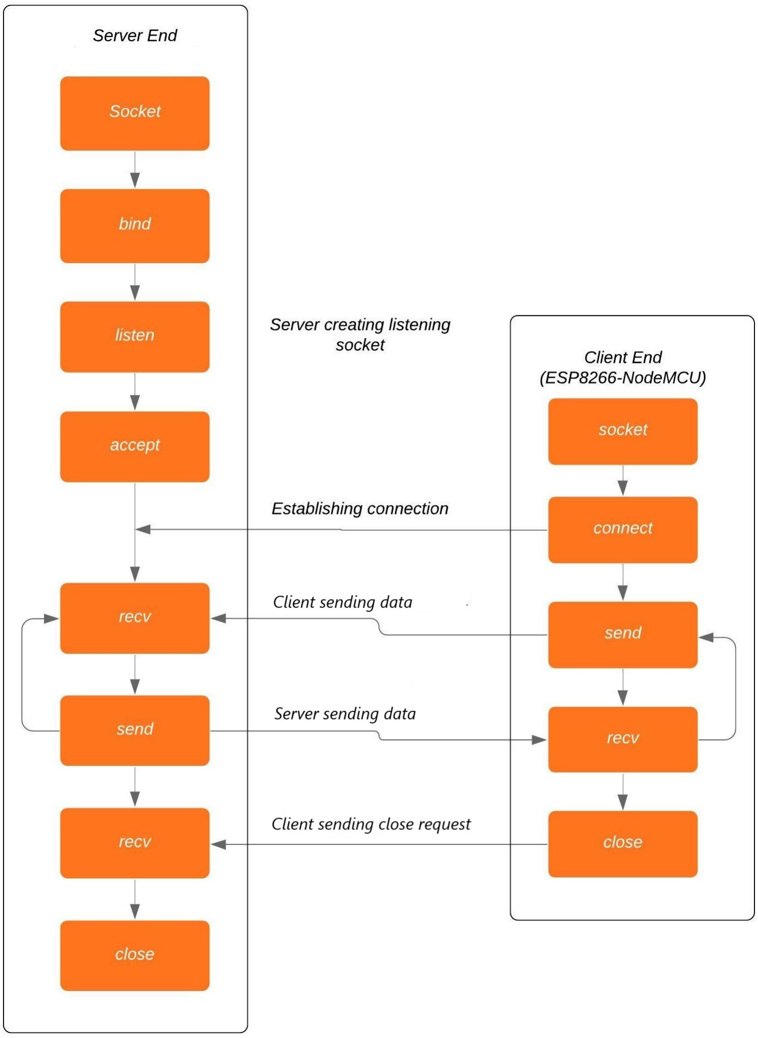
3.5 RShell Setup

In order to install RShell the following command is to be typed in the terminal as given below:

```
pip3 install rshell
```

To understand rshell and its commands click here.

4 Working Principle



The above diagram shows how a TCP Socket connection is established in between Server and Client. The data sent by the Server is the string established by converting speech to text. The data sent by the Client is the confirmation message used for debugging purposes.

5 Coding

5.1 Server Side

Steps:

1. Type the code in the text editor of your choice and save the file in this format <`filename.py`>
2. Run the above by entering this command on terminal:
   ```
   python3 filename.py
   ```

5.2 Client Side

Type the code in the text editor of your choice and the **file should be saved as** <`main.py`>

Copying the main.py into the file system of NodeMCU(ESP8266)

1. Connect the device to your PC and check if it has been connected to your Ubuntu Virtual Machine. If it is not connected go to the Menu of the VirtualBox and follow the steps:
   `Devices>USB>Silicon Labs CP2102 USB to UART Bridge Controller[0100]`

2. To attach the device type the command in the terminal: `dmesg|grep ttyUSB`

3. To enter into RShell type the following command:
   ```
   rshell --buffer-size=30 -p /dev/ttyUSBx
   ```
   Here, `x` = 0 or 1 or 2 or 3, depending upon which port the device gets attached to. This number can be noted after the execution of step 2.

4. To copy `main.py` into the file system of NodeMCU(ESP8266), enter the following command:
   ```
   cp main.py /pyboard/
   ```

Now the board can be powered by separate power source(like a 9V battery or any supply between 5-10V). If you want to reset the board, press the RESET button on the board.

6 Hardware Connections

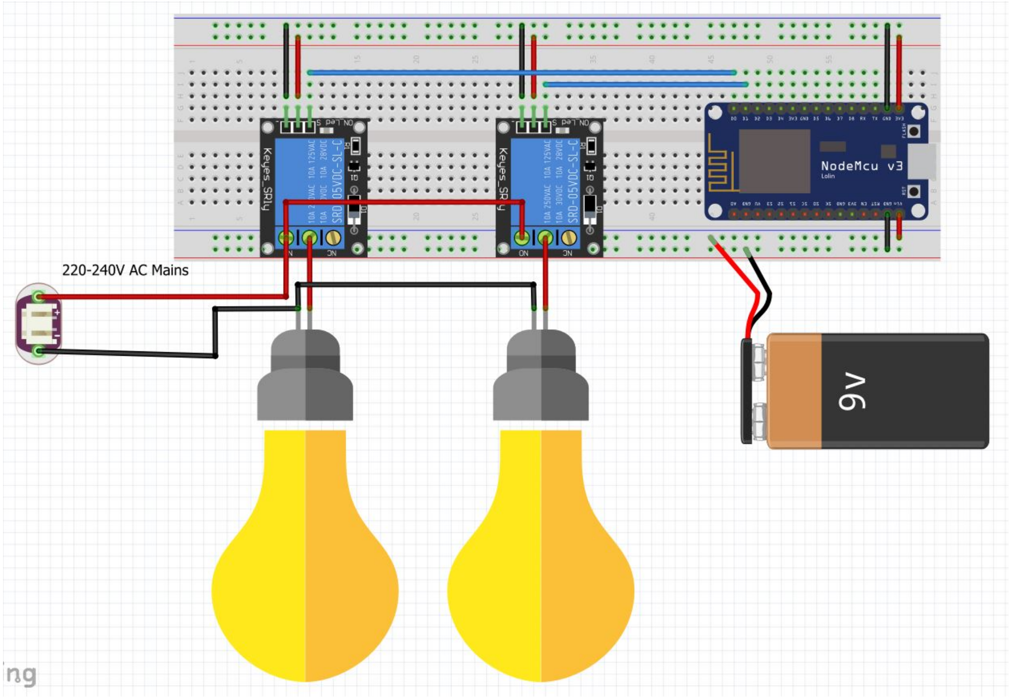


Fig1: Circuit Connections.

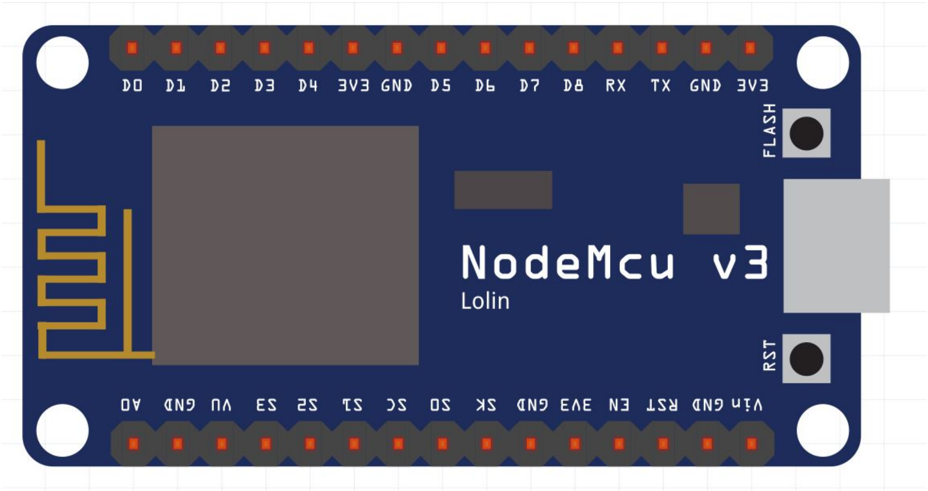

D0 D1 D2 D3 D4 3V3 GND D5 D6 D7 D8 RX TX GND 3V3
FLASH
NodeMcu v3
Lolin
RST
A0 GND VU S3 S2 S1 SC SO SK GND 3V3 EN RST GND Vin

Fig2: NodeMCU Pins.

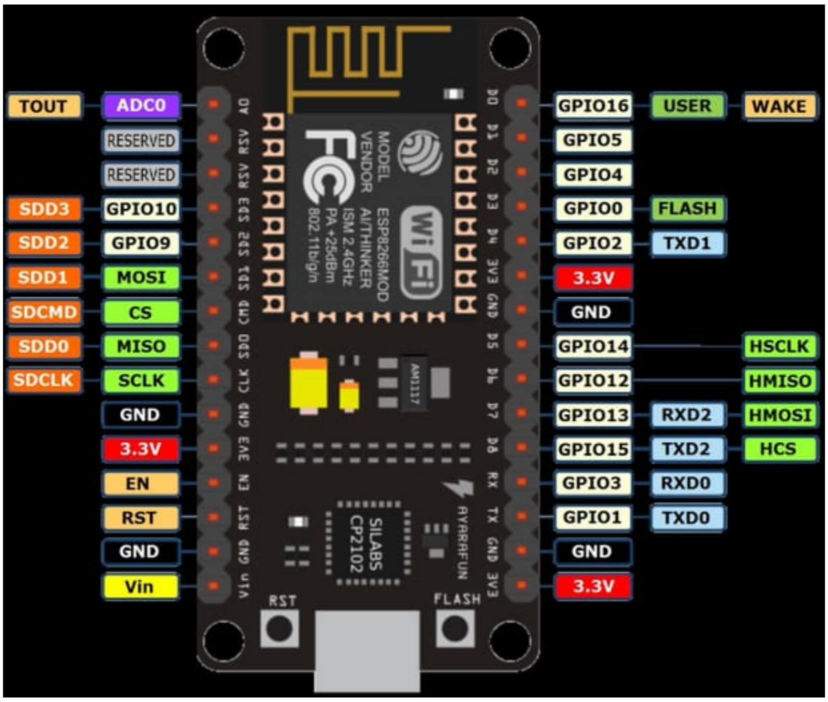
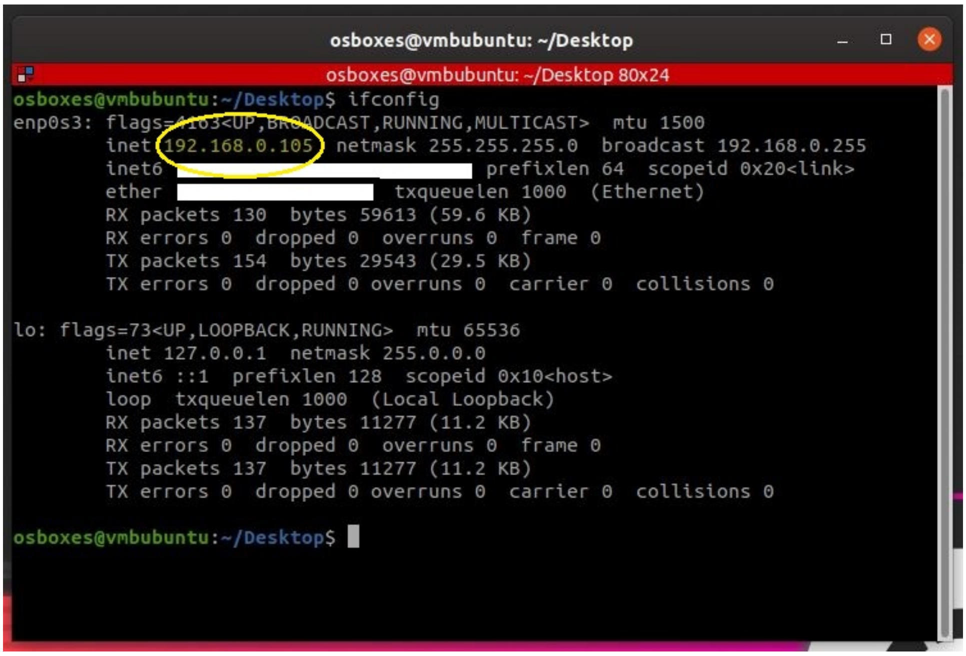


Fig3: MicroPython Pinout for NodeMCU

7 Important Steps/Points to be noted before executing or copying files into file system

1. In order to get the server IP(On any Linux), type the following command on the terminal: `ifconfig` Once the the IP address is found replace the right hand side of this `host="Put_Server_IP_address_here"` with your actual IP address within double qoutes in both Server and Client(main.py) codes.


osboxes@vmbubuntu: ~/Desktop
osboxes@vmbubuntu: ~/Desktop 80x24

```
osboxes@vmbubuntu:~/Desktop$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.105  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 [redacted]  prefixlen 64  scopeid 0x20<link>
        ether [redacted]  txqueuelen 1000  (Ethernet)
        RX packets 130  bytes 59613 (59.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 154  bytes 29543 (29.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 137  bytes 11277 (11.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 137  bytes 11277 (11.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

osboxes@vmbubuntu:~/Desktop$ 
```

Fig3: To find Computer Local IP address

2. To connect NodeMCU to WiFi, in the main.py code in the function def do_connect(), please replace `sta_if.connect('Your_WiFi_Name', 'Your_WiFi_Password')` with your actual WiFi name and password.

THANK YOU!