

DESCRIPTION DOCUMENT FOR WIFI TWO RELAY WITH POWER MONITORING MODULE

HARDWARE REVISION 0.1

| Department | Name | Signature | Date |
|------------|------|-----------|------|
| Author | | | |
| Reviewer | | | |
| Approver | | | |

Revision History

| Rev | Description of Change | Effective Date |
|-----|-----------------------|----------------|
| A | Initial Release | |

ABSTRACT:

This document is a detailed product description that describes the effective features of the product. It includes a functional hardware description of the product with its internal block diagram and product images.

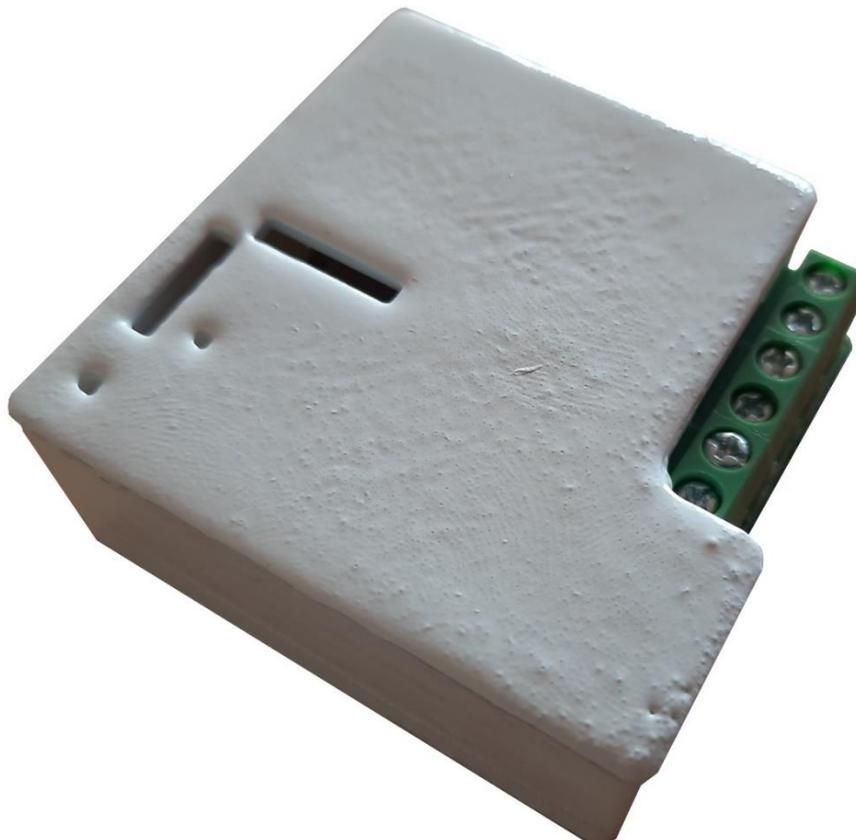


Table of Contents

| | |
|--|----|
| Revision History | 1 |
| 1. ABBREVIATIONS | 6 |
| 2. REFERENCES | 6 |
| 3. PURPOSE | 6 |
| 4. SCOPE | 6 |
| 5. SAFETY AND WARNING | 7 |
| 6. PRODUCT FEATURES | 7 |
| a. SPECIAL FEATURES | 7 |
| 7. PRODUCT DESCRIPTION | 8 |
| a. PHYSICAL DESCRIPTION | 8 |
| b. FUNCTIONAL DESCRIPTION | 8 |
| 8. SYSTEM OVERVIEW | 9 |
| 9. TECHNICAL SPECIFICATION | 9 |
| a. ELECTRICAL SPECIFICATION | 9 |
| i. Input Electrical Specification | 9 |
| ii. Output Electrical Specification | 9 |
| b. MECHANICAL SPECIFICATION | 10 |
| i. Box Dimension | 10 |
| 10. ELECTRICAL CONNECTIONS | 11 |
| a. HEADER AND SWITCH DETAILS | 11 |
| b. ESP USED GPIO PIN CONFIGURATION: | 11 |
| c. APPLICATION WIRING DIAGRAM | 12 |
| i. Example-1: Basic operation mode | 12 |
| ii. Example-2: Two-way switch operation mode. | 12 |
| d. HEADER PIN CONFIGURATION | 13 |
| i. Header J3: | 13 |
| ii. Header J2 | 13 |
| iii. Header J5 | 13 |
| 11. HOW TO USE THE PRODUCT | 14 |
| a. CONFIGURE THE DEVICE LOADED WITH TASMOTA FIRMWARE TO EXISTING NETWORK | 14 |
| i. CONFIGURING DEVICE FOR MQTT MODE: | 17 |
| b. CONFIGURE DEVICE LOADED WITH ARMtronix FIRMWARE TO EXISTING NETWORK | 20 |
| 12. HOW TO CUSTOMISE FIRMWARE | 23 |
| a. STEPS TO LOAD PROGRAM TO ESP8266: | 23 |
| 13. FREQUENTLY ASKED QUESTIONS ON HARDWARE (FAQ's) | 26 |
| 14. INTEGRATION WITH "OPENHAB" | 31 |

- a. THINGS REQUIRED TO INSTALL OPENHAB ON RASPBERRY PI: 31
 - i. Important things to be downloaded:..... 31
 - ii. Installing Raspberry Pi OS and Updating it: 31
 - iii. Installing/Updating required software/drivers:..... 32
- b. STEPS EXPLAINED ON, HOW TO USE OPENHAB IN WINDOWS..... 34
- 15. FAQ ON OPENHAB..... 39
- 16. INTEGRATION WITH “AMAZON ALEXA” 40
 - a. STEPS TO INSTALL ALEXA APP IN SMARTPHONE 40
- 17. INTEGRATION WITH “GOOGLE HOME” 44
 - a. FOLLOWING STEPS EXPLAIN, HOW TO INTEGRATE GOOGLE HOME WITH OPENHAB 44

Table of figures

| | |
|--|----|
| Figure 1: Block Diagram | 8 |
| Figure 2: Box Dimension | 10 |
| Figure 3: Header and Switch positions | 11 |
| Figure 4: Basic application connection diagram | 12 |
| Figure 5: Two-way switch connection diagram | 12 |
| Figure 6: Header J3 Pin Configuration | 13 |
| Figure 7: Device hosting Access point..... | 14 |
| Figure 8: Access point name | 14 |
| Figure 9: Hosting IP address..... | 14 |
| Figure 10: Web Server..... | 15 |
| Figure 11: Saving Configuration..... | 15 |
| Figure 12: Controlling a load | 16 |
| Figure 13: List of ESPs GPIOs available in dropdown list | 16 |
| Figure 14: Main Menu..... | 17 |
| Figure 15: Configuration Option | 17 |
| Figure 16: Option to configure to MQTT | 18 |
| Figure 17: Option to enter required MQTT details..... | 18 |
| Figure 18: Example test results | 19 |
| Figure 19: Device hosting Access point..... | 20 |
| Figure 20: Device configuration settings page..... | 21 |
| Figure 21: Homepage of the device | 22 |
| Figure 22: Control and monitor page of the device..... | 22 |
| Figure 23: BA015 to USB-UART converter connections..... | 23 |
| Figure 24: Board Selection | 24 |
| Figure 25: IDE Selection | 24 |
| Figure 26: COM port selection..... | 25 |
| Figure 27: Code execution | 25 |
| Figure 28: Accessing device manager | 26 |
| Figure 29: Device manager window..... | 26 |
| Figure 30: USB port selection..... | 27 |
| Figure 31: USB driver update path selection | 27 |
| Figure 32: Searching for USB drivers..... | 28 |
| Figure 33: Main menu page of device via http | 28 |
| Figure 34: Sub menu to select confirmation modes | 29 |
| Figure 35: Configuration details screen | 29 |
| Figure 36: MQTT parameter entry screen | 30 |
| Figure 37: Putty application for Raspberry Pi | 34 |
| Figure 38: Path for the file | 34 |

Figure 39: OpenHAB sub folders, Accessing item folder 35

Figure 40: Login page of OpenHab..... 36

Figure 41: Menu bard of OpenHab 36

Figure 42: Home page of device with entered IP..... 37

Figure 43: Devices are categorized as Groups under frame 37

Figure 44: OpenHAB Add-ons binding option selection 38

Figure 45: Binding option selected as OpenHAB 39

Figure 46: Device selection to configure with OpenHAB 39

Figure 47: Alexa app bootup screen 40

Figure 48: Alexa app home screen..... 41

Figure 49: Check # of devices connected..... 41

Figure 50: Devices devided as groups in Alexa app 42

Figure 51: Names are defined for loads..... 43

Figure 52: Google Home app’s Home screen..... 44

Figure 53: Google Home app’s Settings screen 45

Figure 54: Google Home app’s screen to Add IoT devices..... 46

Figure 55: Screen to Add cloud service provider 46

Figure 56: Login Screen..... 47

Figure 57: Apps’s permission athentication screen 47

Figure 58: List of added devices..... 48

Figure 59: Device control screen..... 49

1. ABBREVIATIONS

| Term | Description |
|-------|---|
| A | Ampere |
| AC | Alternating Current |
| ASIC | Application Specific Integrated Circuit |
| COM x | Communication Port (Where 'x' represents the port number) |
| COM | Common Pin of the Relay |
| DC | Direct Current |
| GPIO | General Purpose Input Output |
| HTTP | Hypertext Transfer Protocol |
| Hz | Hertz |
| I2C | Inter Integrated Circuit |
| IDE | Integrated Development Environment |
| IP | Internet Protocol |
| LED | Light Emitting Diode |
| MCU | Microcontroller Unit |
| MQTT | Message Queue Telemetry Transport |
| PCB | Printed Circuit Board |
| PWM | Pulse Width Modulation |
| SPI | Serial Peripheral Interface |
| SSID | Service Set Identifier |
| UART | Universal Asynchronous Receiver Transmitter |
| USB | Universal Serial Bus |
| V | Volts |
| ZCD | Zero Crossover Detection |

2. REFERENCES

| | |
|-----------------------|---|
| Company Weblink | https://www.armtronix.in |
| Youtube Weblink | https://www.youtube.com/channel/UCr3QN6s65jDSxKDX4QPc03oQ |
| Intractable's Weblink | https://www.instructables.com/member/Armtronix/ |
| Github's Weblink | https://github.com/armtronix |

3. PURPOSE

The purpose of this document is to outline the design description for the WiFi Two Relay with power monitoring Box. It provides a high-level summary of the product.

4. SCOPE

This document describes system architecture which includes Power supply, Relay, WiFi Module, Power monitoring, and other available GPIOs.

5. SAFETY AND WARNING

Note that, this board to be powered with AC 230V with required current. Work and handle carefully with AC power as it is harmful and danger for human beings. Touching live wire or board when it is ON is danger and not advisable, it may cause to death, please avoid it.

Even a 50 V AC supply is sufficient to kill you. Please Switch off the mains before you make or change connections, be very careful. If you are not sure of anything related to the AC supply lines, please call an electrician ask and him to help you with it. Do not attempt to interface to mains unless you have adequate training and access to appropriate safety equipment. Never work on high voltages by yourself when you are alone. Always ensure that you have a friend/partner who can see and hear you and who knows how to quickly turn off power in case of an accident. Use a 2A Fuse in series with the input to the board as a safety measure. Basic test codes are available on our GitHub web page. Please refer them.

Fire Hazard: Making wrong connections, drawing more than rated power, contact with water or other conducting material, and other types of misuse/overuse/malfunction can all cause overheating and risk starting a fire. Test your circuit and the environment in which it is deployed thoroughly before leaving it switched on and unsupervised. Always follow all fire safety precautions.

6. PRODUCT FEATURES

- Works directly with AC power 100 - 240 V AC 50-60 Hz.
- Product firmware can be updated/reloaded/changed as per user requirement.
- Small formfactor with 55 x 55 x 21 mm.
- Two relays with live AC powered output via NO pin of relay is accessible to user.
- Board outputs can handle loads up to 240W.
- WiFi with MQTT or HTTP protocol
- MQTT Authentication with Username and Password.
- Basic Firmware to enter SSID and password to connect to the router.
- Firmware has ability to control device through HTTP and MQTT mode.
- Push Button on board Provided for device Reset.

a. SPECIAL FEATURES

- AC to DC Power supply module on board.
- ESP8266-12 Wifi Module.
- AC Virtual Switch (physical switch can be rewired directly).
- Electro-Mechanical Relays to control loads.
- Output presence detection.
- Extra GPIOs for any external inputs like sensor if required.
- Protection Fuse for safety of on-board circuitry.
- I2C accessible for expansion.

7. PRODUCT DESCRIPTION

a. PHYSICAL DESCRIPTION

- AC to DC Power supply module
- Wifi Module
- AC Virtual Switch
- Electro-Mechanical Relay

b. FUNCTIONAL DESCRIPTION

Block Diagram

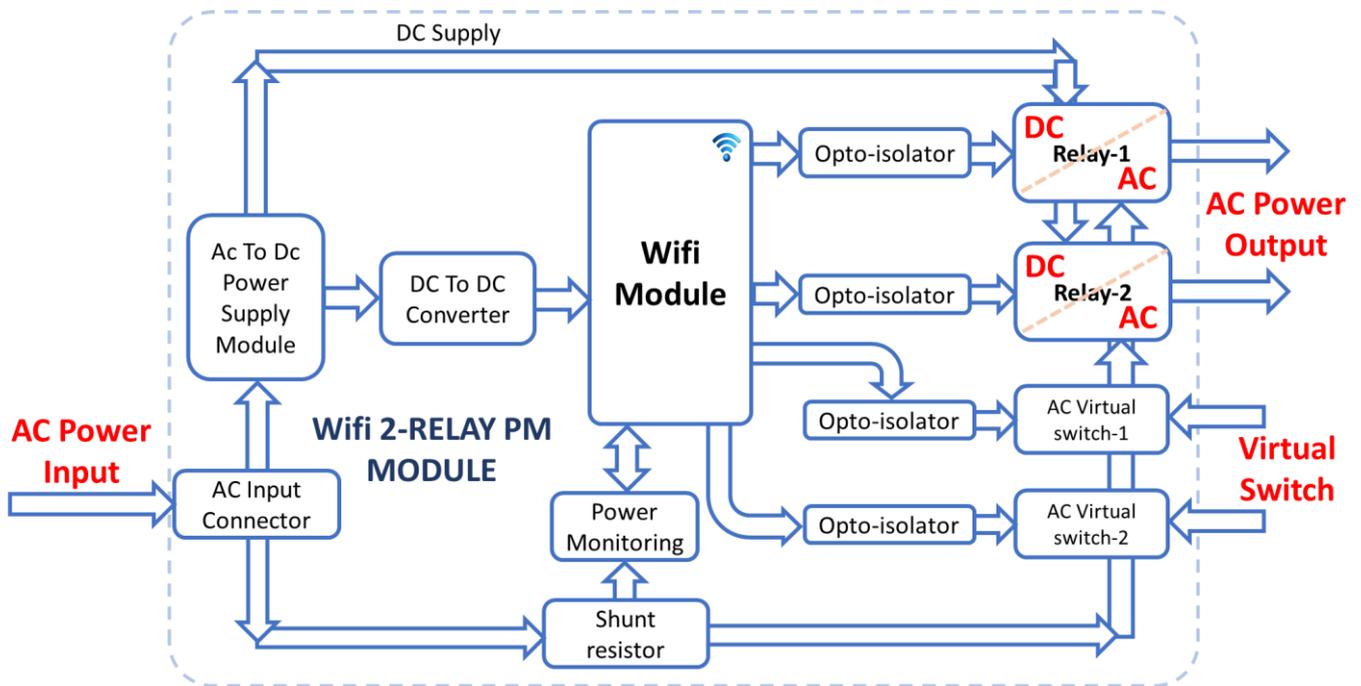


Figure 1: Block Diagram

WiFi Two Relay with Power Monitoring module is specially designed and brought to market for Wifi enabled Building automation application. This can handle a load at 240 V AC. There are two electro-mechanical relays mounted on board to control (ON/OFF) external electrical loads from a mobile application or webpages using MQTT/HTTP protocol. It also has features like, power monitoring and AC virtual switch. The board has programming header (TX, RX, DTRE, RTSE) in compatible to NodeMCU which can be used with Arduino IDE to help the user to program using an external USB-UART converter. It has on-board power supply module which takes standard AC voltage as input and provides required DC voltage as output. The DC voltage is used to power-up Wifi module used on board to establish Wifi communication and monitor & control loads.

8. SYSTEM OVERVIEW

1. AC to DC Power supply module

AC to DC converter is a switch mode power supply. It rectifies and regulates voltage from 230 V AC to 5 V DC with output current capacity of 500 mA DC. The wattage of on-board power supply is at maximum of 2.5W.

The 5V supply is used to power on relay and USB-UATT converter. There is a DC-DC converter on board to regulate voltage from 5 V DC to 3.3 V DC to supply power to Wifi module.

2. Wifi Module

Wifi module used on the board is ESP32 with its minimum GPIOs are easily accessible in a header to user for their own application. Wifi module is powered on through 3.3 V DC. It works on both MQTT / HTTP protocol.

3. Electro Mechanical Relay

Electro Mechanical relay is powered by 5 V DC. The AC live powered terminal (NO) is given access to user in a terminal block to control loads. An opto-isolator based driver circuit is used to drive the relay, to create isolation between relay's AC and DC part. An opto-isolator based AC-DC isolation circuit is used to drive the relay.

4. AC Virtual Switch

AC Virtual switch circuit is connected to Wifi module through an opto isolator AC-DC isolation. It gives a ZCD output to Wifi Module to detect the change in status of switch.

5. Power Monitoring

Power monitoring block is implemented in the hardware to monitor the consolidated power consumption by the loads. The circuit is based on single phase energy metering IC HLW8012. Which has ability to monitor voltage, current and active power and give the output in the form of pulses.

User need to convert these pulse output to appropriate parameters. With these parameters, user can be able to calculate the energy consumption by the loads.

9. TECHNICAL SPECIFICATION

a. ELECTRICAL SPECIFICATION

i. Input Electrical Specification

| Input Specifications | | | | |
|----------------------|-----|-----|-----|-------|
| Description | Min | Typ | Max | Unit |
| Voltage AC | 100 | 220 | 240 | Volts |
| Current AC | - | 0.1 | - | Amps |
| Power AC | - | 2.5 | - | Watts |
| Frequency | 50 | - | 60 | Hz |

ii. Output Electrical Specification

| Relay Output Specifications (Maximum) | | | | |
|---------------------------------------|-----|-----|-----|-------|
| Description | Min | Typ | Max | Unit |
| Voltage AC | - | - | 500 | Volts |
| Power AC | - | - | 500 | Watts |

b. MECHANICAL SPECIFICATION

i. Box Dimension

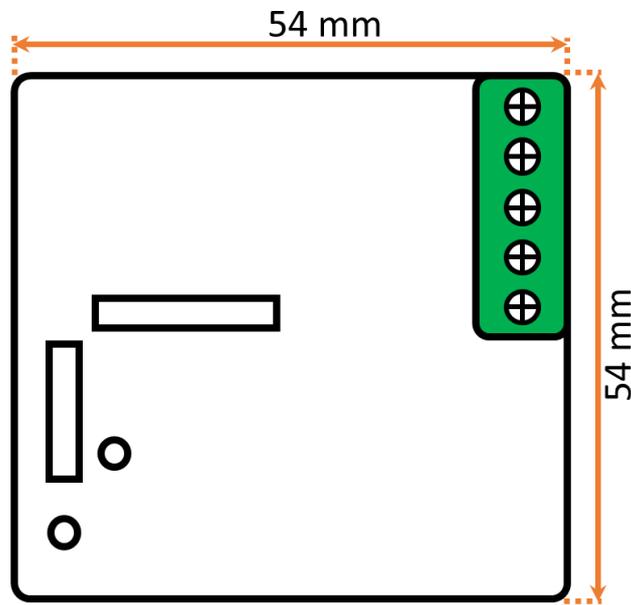


Figure 2: Box Dimension

10. ELECTRICAL CONNECTIONS

a. HEADER AND SWITCH DETAILS

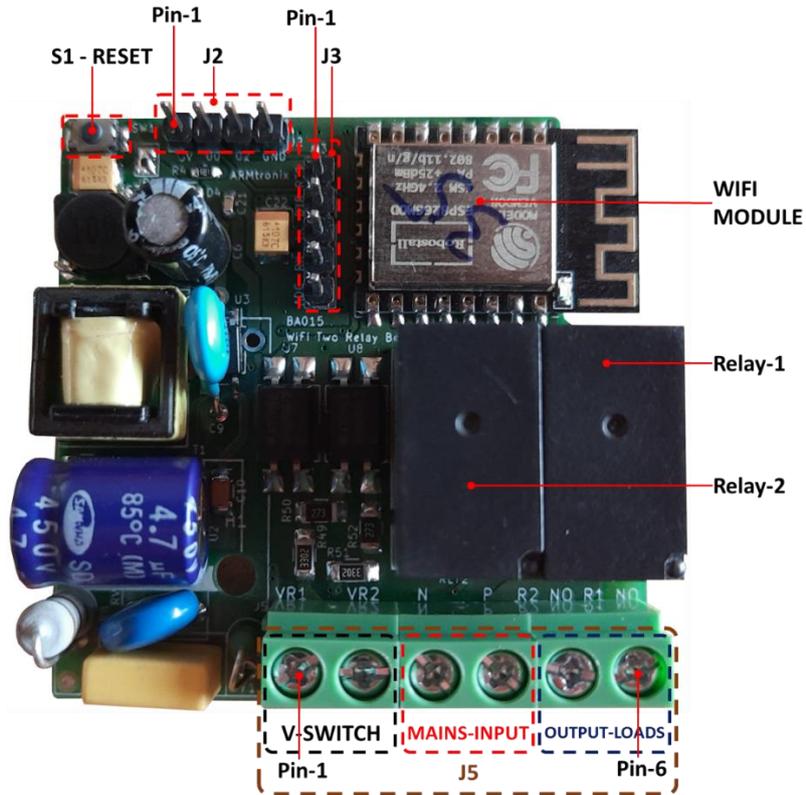


Figure 3: Header and Switch positions

Note:

- On-Board fuse is only for SMPS and not for loads.

b. ESP USED GPIO PIN CONFIGURATION:

| ESP8266-12 Pin no. | Arduino Pin # | Designator | Description |
|--------------------|---------------|------------|------------------|
| 20 | D1 | GPIO5 | Relay_1 |
| 19 | D2 | GPIO4 | Relay_2 |
| 6 | D6 | GPIO12 | Virtual_Switch_1 |
| 4 | D0 | GPIO16 | Virtual_Switch_2 |
| 16 | D8 | GPIO15 | EM_SEL |
| 7 | D7 | GPIO13 | EM_CF1 |
| 5 | D5 | GPIO14 | EM_CF |
| 17 | D4 | GPIO2 | I2C_SCL |
| 18 | D3 | GPIO0 | I2C_SDA |

Table 1:: ESP Pin Configuration

c. APPLICATION WIRING DIAGRAM

i. Example-1: Basic operation mode

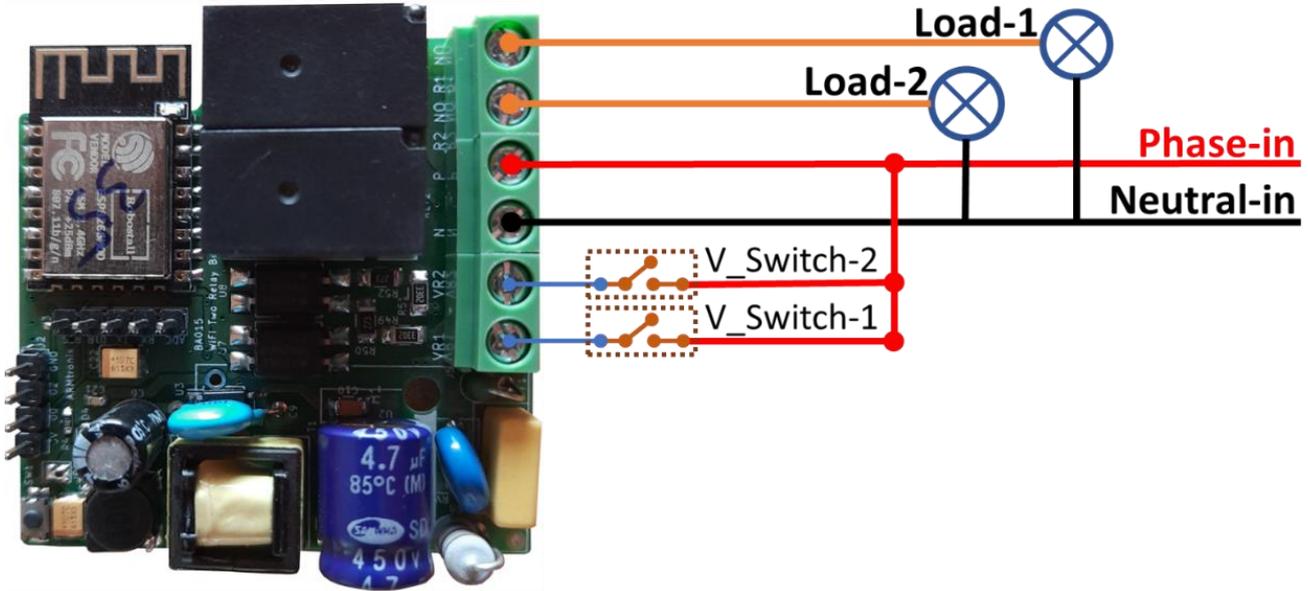


Figure 4: Basic application connection diagram

Figure 4 represents about connection between electrical load, AC virtual switches and board for basic operation.

ii. Example-2: Two-way switch operation mode.

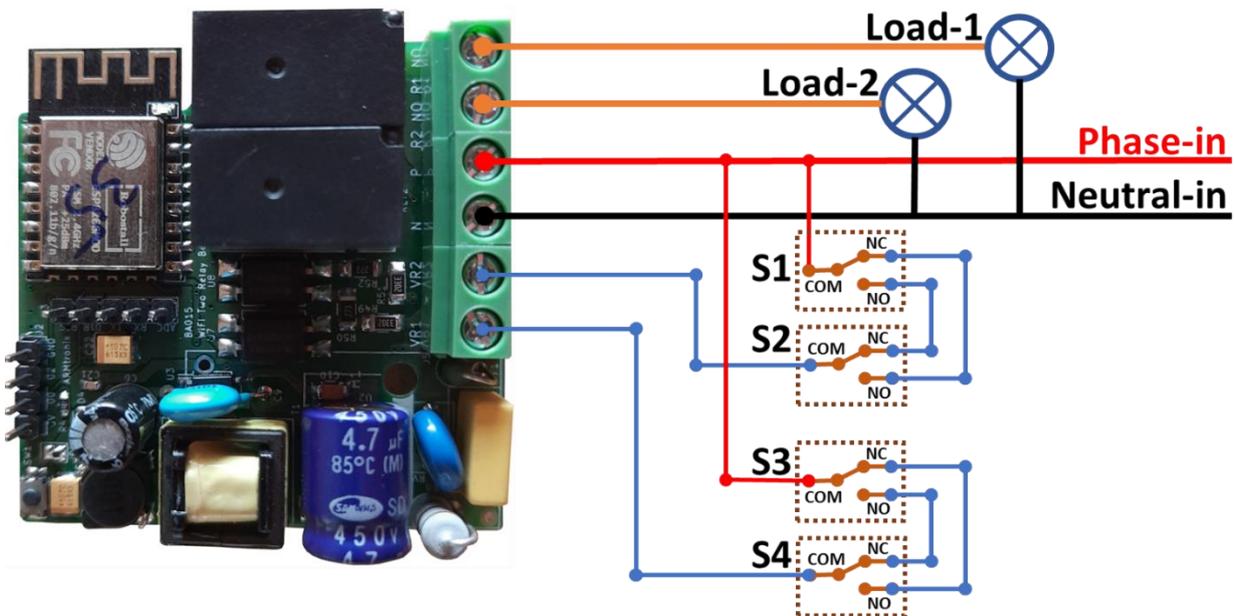


Figure 5: Two-way switch connection diagram

Figure 5 represents about connection between electrical load, AC virtual switches and board in two-way switch mode operation.

d. HEADER PIN CONFIGURATION

i. Header J3:

| Header Pin # | ESP8266-12 Pin # | Designator | Description |
|--------------|------------------|------------|-------------------------------------|
| 1 | - | RTSE | |
| 2 | - | DTRE | |
| 3 | 22 | TxDE | Transmission from board |
| 4 | 21 | RxDE | Reception by board |
| 5 | ADC | Ex_ADC | ESP ADC with External up to 3.3V DC |

Figure 6: Header J3 Pin Configuration

ii. Header J2

| Header Pin | ESP GPIO # | Designator | Description |
|------------|------------|------------|---------------|
| 1 | - | +5V | 5V DC |
| 2 | 0 | GPIO0 | ESP_GPIO0 |
| 3 | 2 | GPIO2 | ESP_GPIO2 |
| 4 | - | GNDD | Common Ground |

Table 2: Header J2 Pin Configuration

iii. Header J5

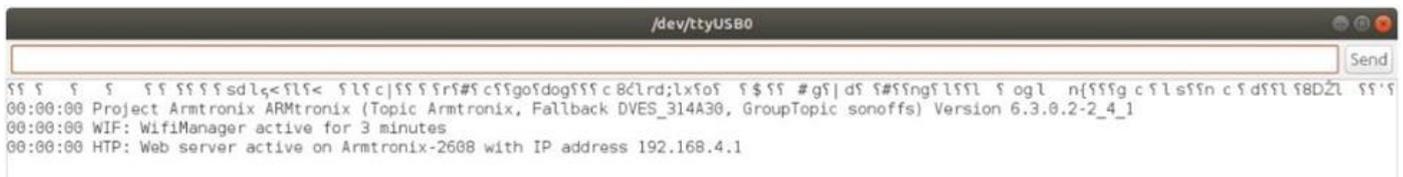
| Header Pin | Description | Type |
|------------|--------------------|-----------|
| 1 | AC_VirtualSwitch_1 | AC Input |
| 2 | AC_VirtualSwitch_2 | AC Input |
| 3 | Neutral | AC Input |
| 4 | Phase | AC Input |
| 5 | Load_1 | AC Output |
| 6 | Load_2 | AC Output |

Table 3: Header J5 Pin Configuration

11. HOW TO USE THE PRODUCT

a. CONFIGURE THE DEVICE LOADED WITH TASMOTA FIRMWARE TO EXISTING NETWORK

1. Make an input AC phase and Neutral connection as shown in Figure 4.
2. Use an electrical external fuse and MCB with rating 2A/250V, in series to input connections for the purpose of safety.
3. Check and make sure that there is no short circuit between phase and neutral.
4. Ensure that, required safety precautions are taken care.
5. Power ON the device by turning ON the main input supply.
6. Then observe the LED on the device is in ON condition to make sure device is powered up.
7. If the device has NOT powered ON, then turn OFF the main input supply and recheck for connections by following above steps.
8. Once the device Powered ON, it will host an access point as shown in Figure 7 below:

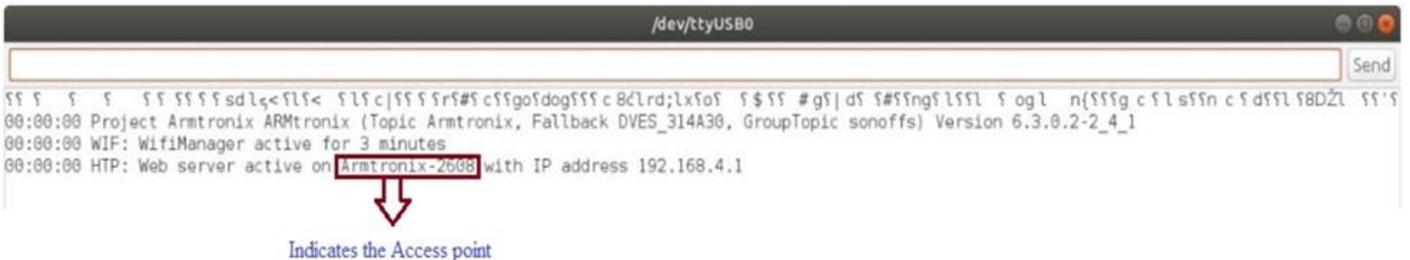


```

/dev/ttyUSB0
00:00:00 Project Armtronix ARMtronix (Topic Armtronix, Fallback DVES_314A30, GroupTopic sonoffs) Version 6.3.0.2-2_4_1
00:00:00 WIF: WifiManager active for 3 minutes
00:00:00 HTP: Web server active on Armtronix-2608 with IP address 192.168.4.1
  
```

Figure 7: Device hosting Access point

9. Connect the mobile to access point with Armtronix-(MAC ID) within 3 minutes as shown below.
EX: Armtronix-2608 as shown in Figure 8 below:



```

/dev/ttyUSB0
00:00:00 Project Armtronix ARMtronix (Topic Armtronix, Fallback DVES_314A30, GroupTopic sonoffs) Version 6.3.0.2-2_4_1
00:00:00 WIF: WifiManager active for 3 minutes
00:00:00 HTP: Web server active on Armtronix-2608 with IP address 192.168.4.1
  
```

↓
Indicates the Access point

Figure 8: Access point name

10. Once the MAC ID is obtained connect to the Wifi through your Smartphone and you will get an IP address as shown in the figure 9 below:

```

WiFi connected
mDNS responder started
192.168.1.34
Web server started
  
```

Figure 9: Hosting IP address

11. After connecting, open any browser and enter the IP address which is displayed on your Smartphone it opens the Web Server as shown in the Figure 10 below, then enter the SSID and Password and you can give the Host name of your choice (just for your reference) then click on SAVE button.

Armtronix SD Module

ARMtronix

Scan for wifi networks

Wifi parameters

AP1 SSId ()

AP1 Password

AP2 SSId ()

AP2 Password

Hostname (%s-%04d)

Save

Figure 10: Web Server

Once the **Configuration** is done it displays the message as shown in the figure 11 below,



Figure 11: Saving Configuration

Then user get access to control the load i.e he can Switch ON / OFF using TOGGLE button as shown in the figure 12 below.



Figure 12: Controlling a load

One advantage is that, this Web server provides Multiple Options, that is GPIO's are accessible to the user where a drop down is provided for different application as shown in the figure 13, below.



Figure 13: List of ESPs GPIOs available in dropdown list

12. One more advantage is that the user can easily **Configure** and **Upgrade** his **Firmware** through smartphone and an option also provided to choose file directly from the phone if it is available as shown in figure 14.

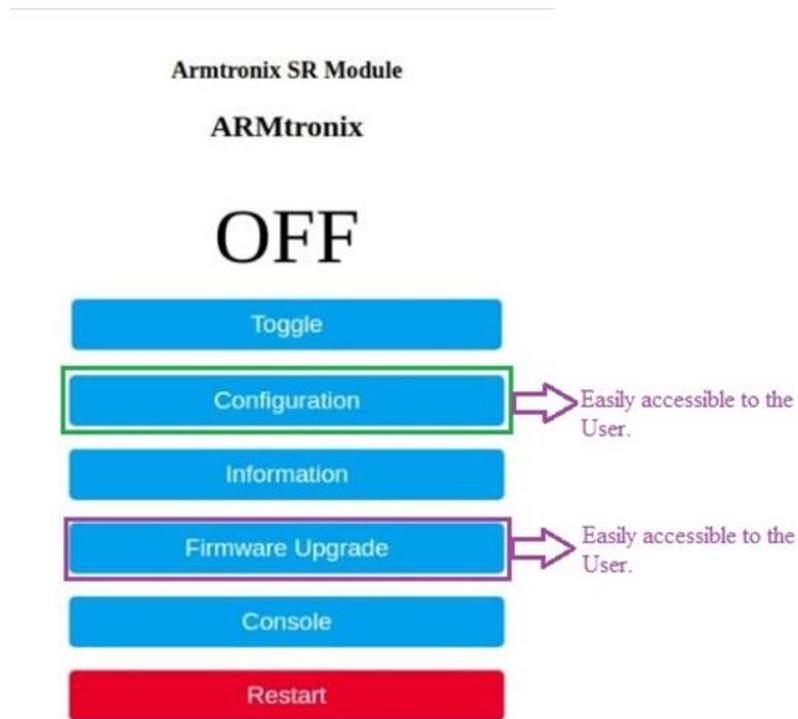


Figure 14: Main Menu

i. CONFIGURING DEVICE FOR MQTT MODE:

13. click on the Configuration option as shown in the figure 15,



Figure 15: Configuration Option

14. Click on Configure MQTT option provided as shown in the figure 16 below,



Figure 16: Option to configure to MQTT

15. User shall click on the SAVE button as shown in figure 17 below, once after entering required details to get access to MQTT protocol from the web server.

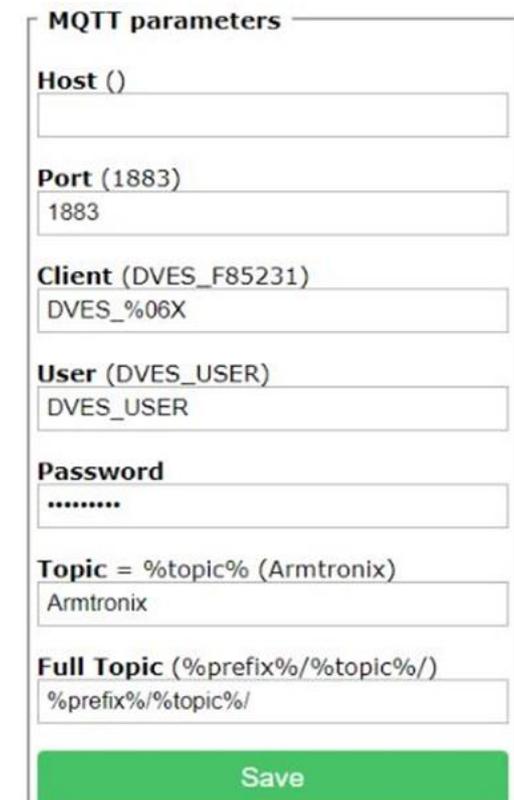


Figure 17: Option to enter required MQTT details

BA015 Module**Sonoff**

| | |
|------------------|-----------|
| Voltage | 231 V |
| Current | 2.387 A |
| Power | 68 W |
| Apparent Power | 551 VA |
| Reactive Power | 547 VAr |
| Power Factor | 0.12 |
| Energy Today | 0.007 kWh |
| Energy Yesterday | 0.000 kWh |
| Energy Total | 0.007 kWh |

ON ON

Sonoff-Tasmota 6.6.0 by Theo Arends

Figure 18: Example test results

b. CONFIGURE DEVICE LOADED WITH ARMtronix FIRMWARE TO EXISTING NETWORK

1. Follow steps 1 to 7 of section 11.a of this document.
2. Once the device Powered ON, it will host an access point as shown in Figure 19 below:

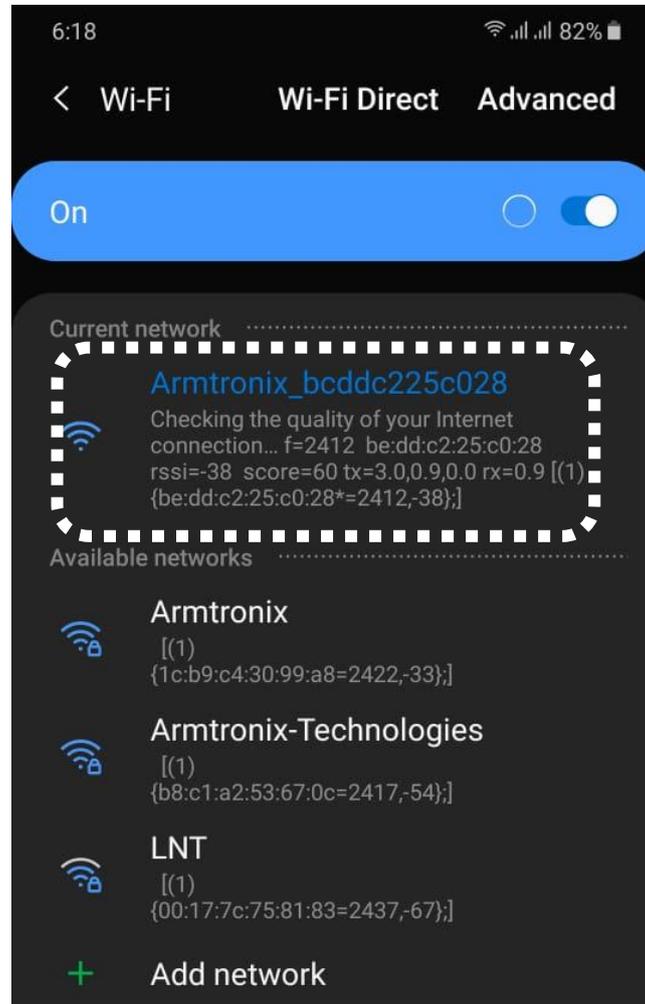
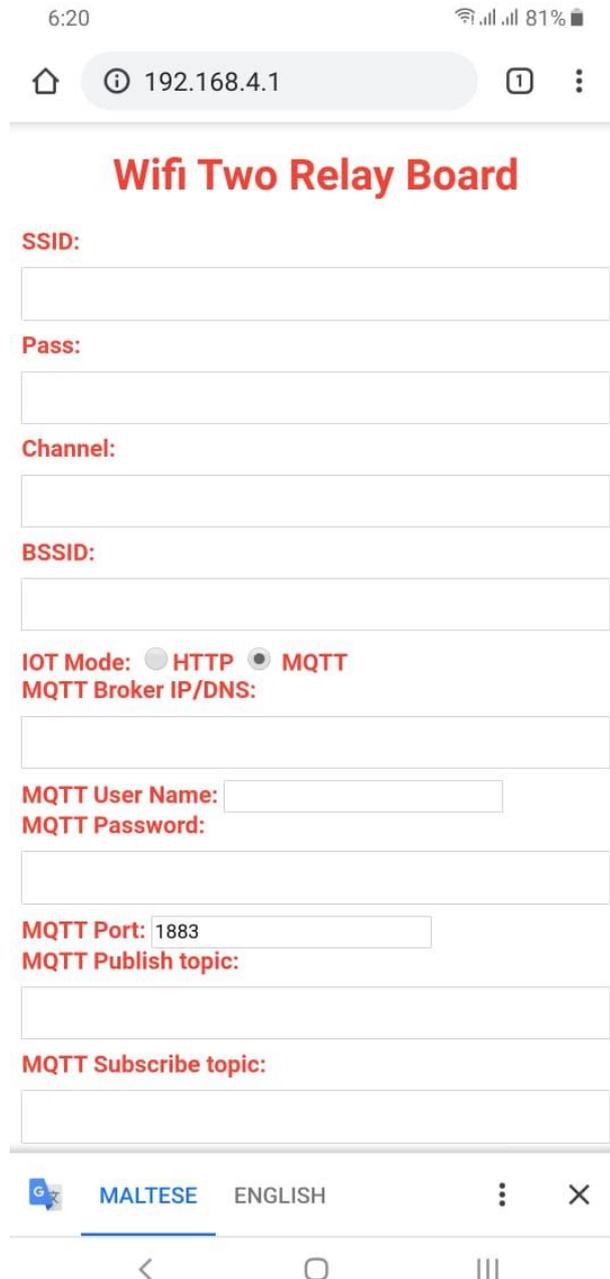


Figure 19: Device hosting Access point

5. Once you find the Access Point hosted by the device, connect to it through your Smartphone.

- Once connected, open any web browser in your connected smart phone and type 192.168.4.1 and you will access page shown in below Figure 20:



6:20 81%

192.168.4.1

Wifi Two Relay Board

SSID:

Pass:

Channel:

BSSID:

IOT Mode: HTTP MQTT

MQTT Broker IP/DNS:

MQTT User Name:

MQTT Password:

MQTT Port:

MQTT Publish topic:

MQTT Subscribe topic:

MALTESE ENGLISH

Figure 20: Device configuration settings page

- Enter all the required details like SSID and password of router or access point to which the device should get connect and MQTT broker IP, username, password (if any) and pub-sub topic to control and monitor the device.
- Click on save/submit button to save and restart the device. On restarting the device will be connected to above specified router or access point.

- After the device connected to your router or AP, kindly search for devices IP address from router or AP and type the same IP address in your web browser to get the control web page of the device as shown in the below Figure 21.

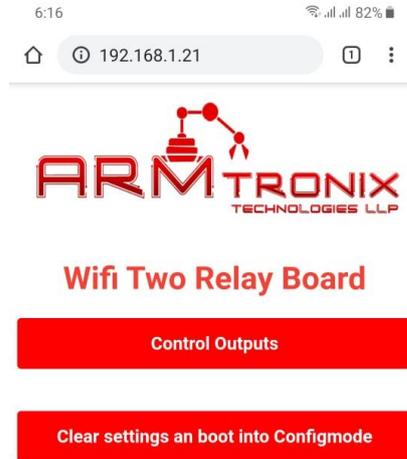


Figure 21: Homepage of the device

- Click on the “Control Outputs” tab button to access the Control and Monitor page as shown in below Figure 22.



Figure 22: Control and monitor page of the device

- Click on “Clear settings and boot into Configmode” button to re-configure the device to different network, on clicking the button you get the page shown in Figure 20.

12. HOW TO CUSTOMISE FIRMWARE

You can program this board using Arduino IDE. Please follow the below steps to program the board by yourself with easy steps as mentioned below:

a. STEPS TO LOAD PROGRAM TO ESP8266:

1. Use external USB-UART converter between computer and this board.
2. Connect external USB-UART converter with DTR and RTS pins shown in Figure 23 below:

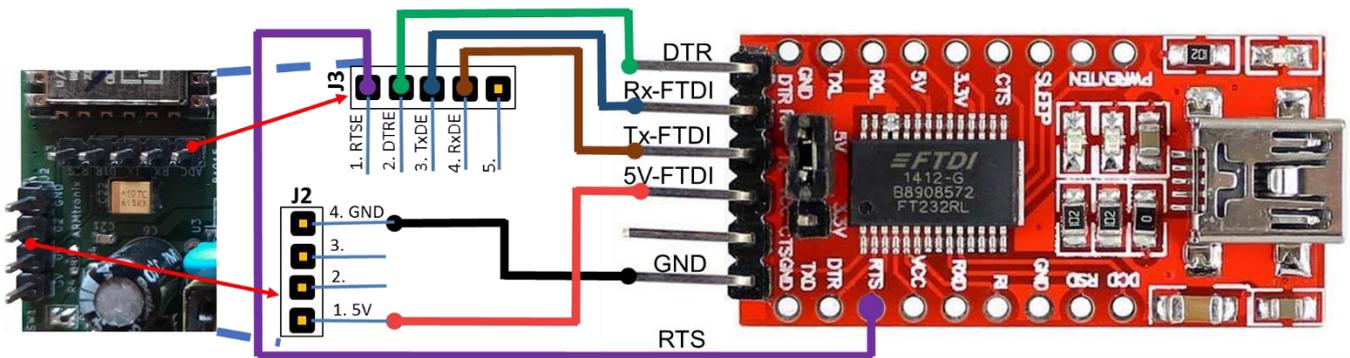


Figure 23: BA015 to USB-UART converter connections

| J3 Header | | USB-UART Converter | |
|------------|-----------------|--------------------|--|
| J3 Pin No. | Pin Description | Pin Description | |
| 1 | RTSE | RTS | |
| 2 | DTRE | DTR | |
| 3 | TxDE | RXD | |
| 4 | RxDE | TXD | |
| J2 Pin No. | Pin Description | | |
| 1 | VCC_5V | 5V | |
| 4 | DGND | GND | |

Table 4: USB-UART to board pin configuration

3. Open your code in Arduino IDE.

- Click on Tools Tab, move mouse pointer on “Board: xxxxxxxxxxx” and click on “NodeMCU0.9 (ESP-12 Module)” as shown in figure 24.

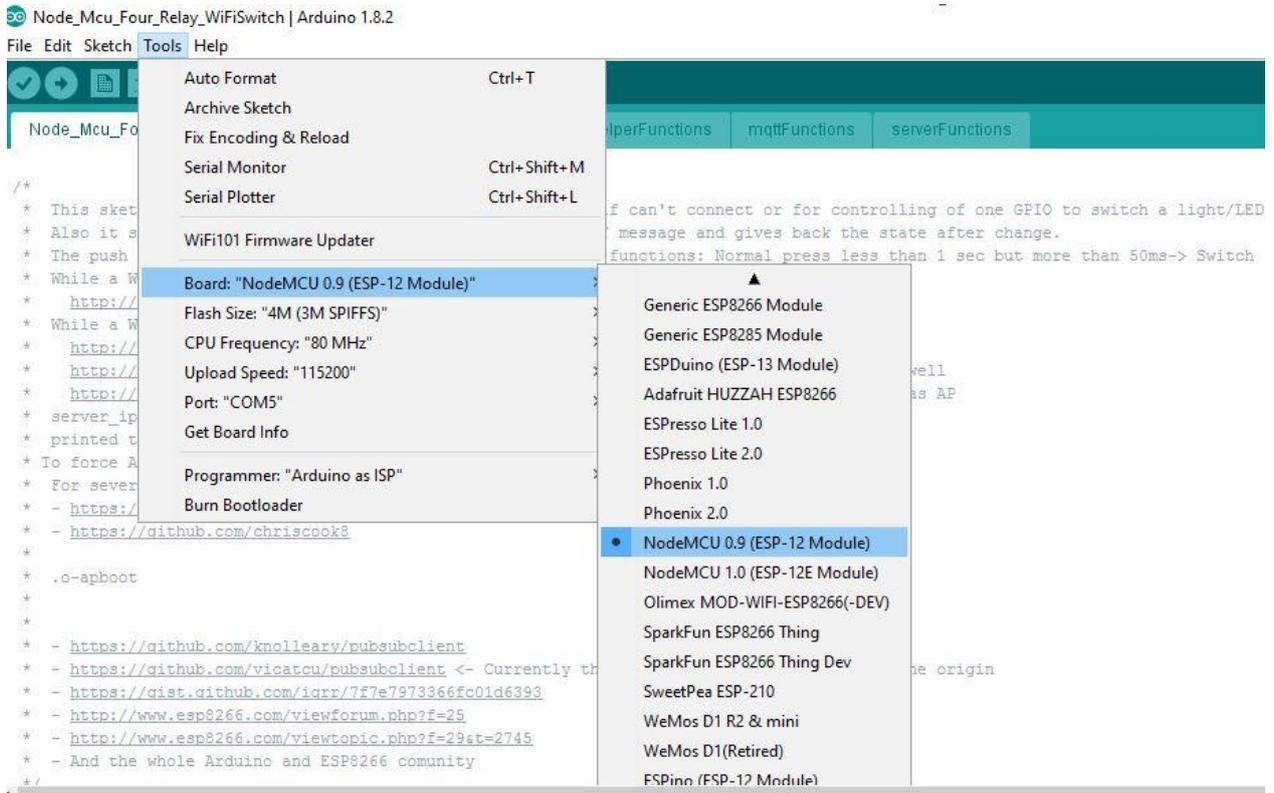


Figure 24: Board Selection

- Click on tools tab, move mouse pointer to “Programmer: “Arduino as ISP””, under this click on “Arduino as ISP” to select. Refer to figure 25

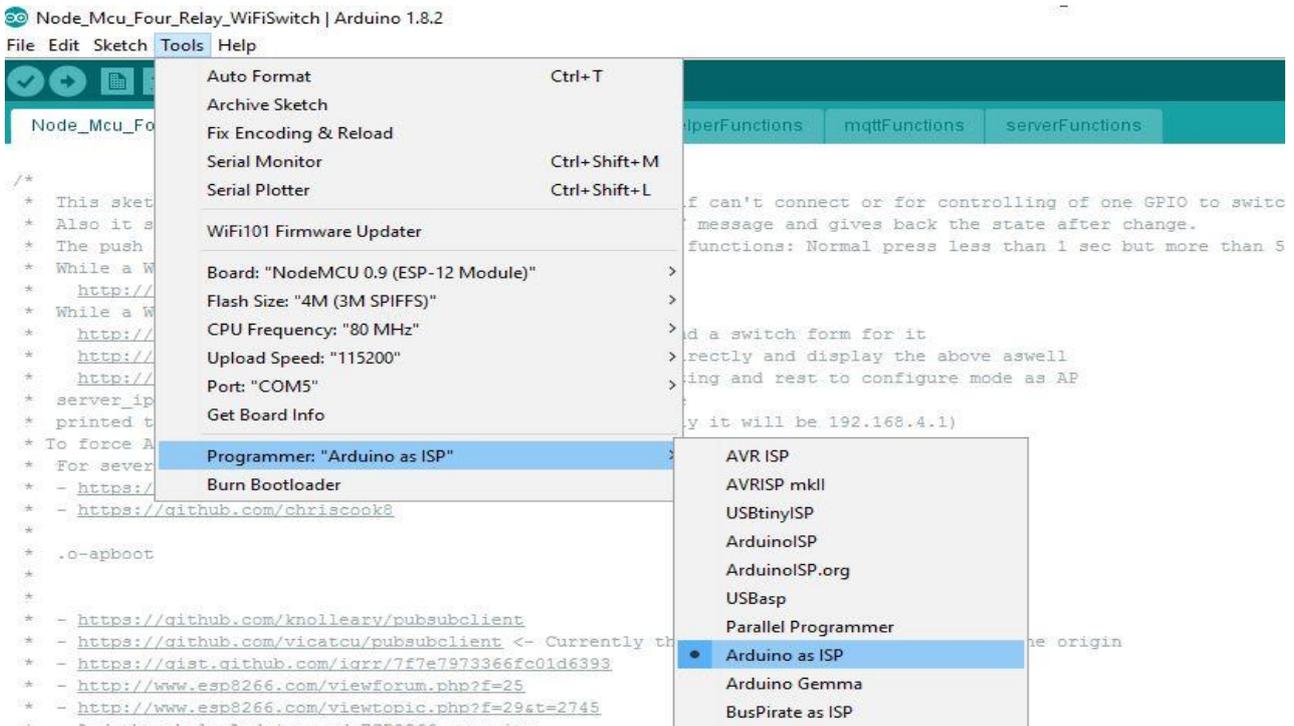


Figure 25: IDE Selection

- Click on tools tab, move mouse pointer to “Port: “COMx”, under this click on “COMx” to select. (“x” refers to port number available in your computer) Refer to figure 61.

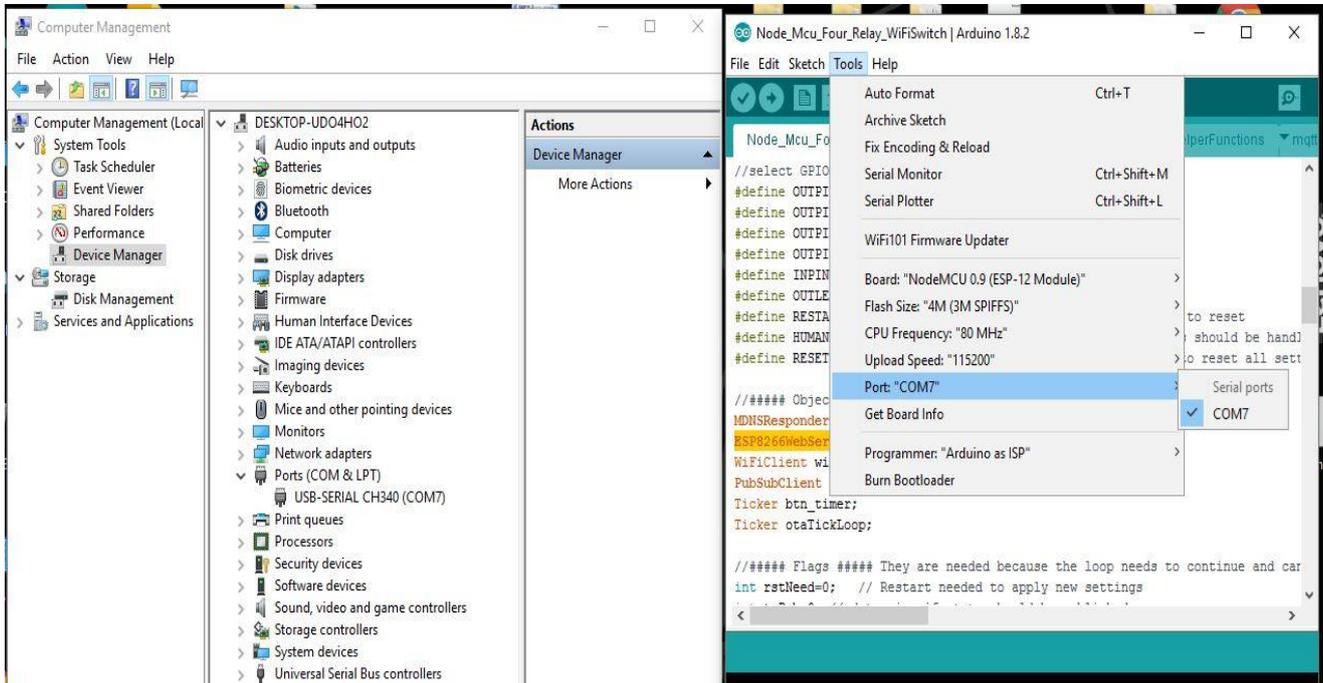


Figure 26: COM port selection.

- Run the program. Refer to Figure 27.

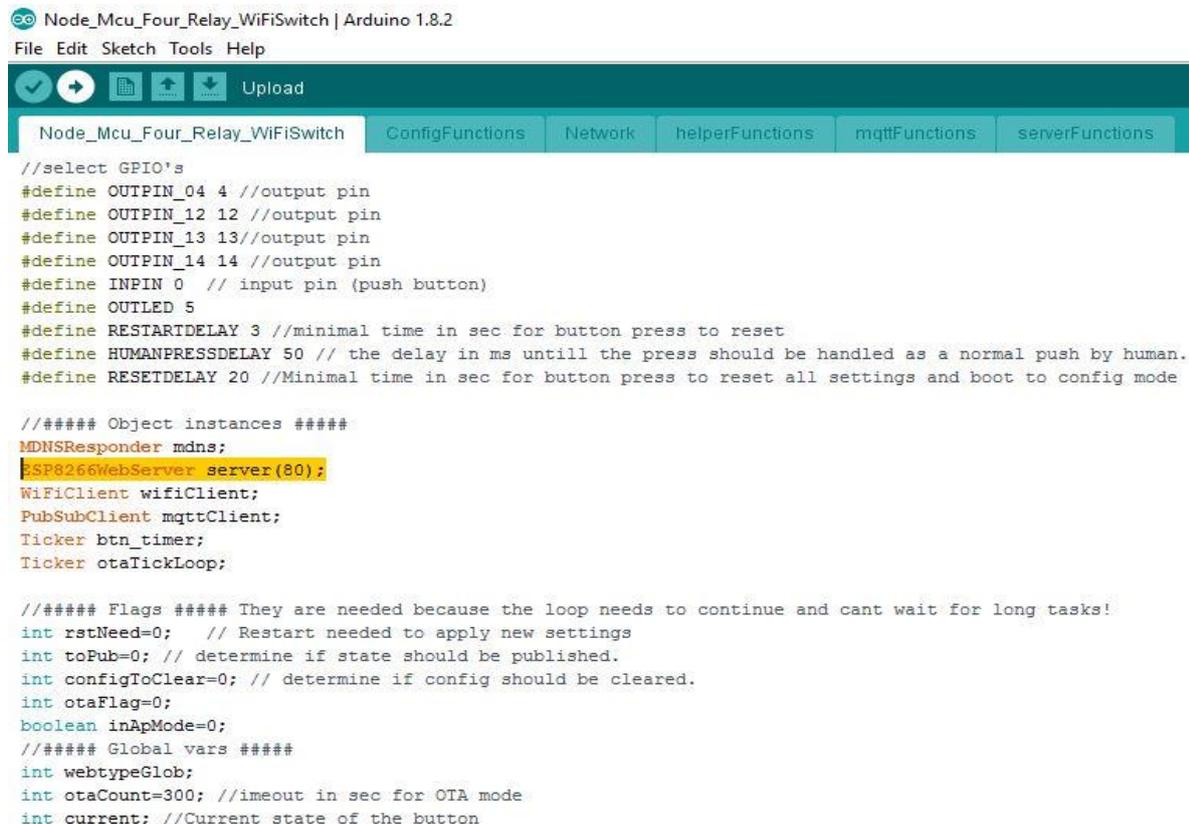


Figure 27: Code execution

13. FREQUENTLY ASKED QUESTIONS ON HARDWARE (FAQ's)

- i. **Although you are connecting USB to your computer but not able to run or upload the program?**

Necessary and required **drivers** has to be installed on your computer so that you can easily connect, run, and upload the program.

Steps are as follows to install USB drivers:

Step 1: Right click on the **This PC** or **PC** icon

Step 2: Click on **Manage** option.

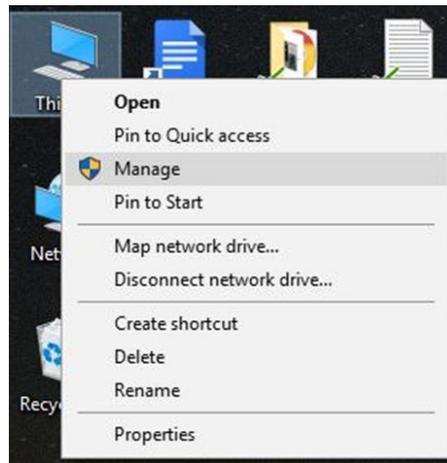


Figure 28: Accessing device manager

Step 3: To understand whether USB is connected or not ensure that your USB LED is blinking when connected to the respective port

Step 4: Then click on the **Device manager** option on the left corner and select Ports option

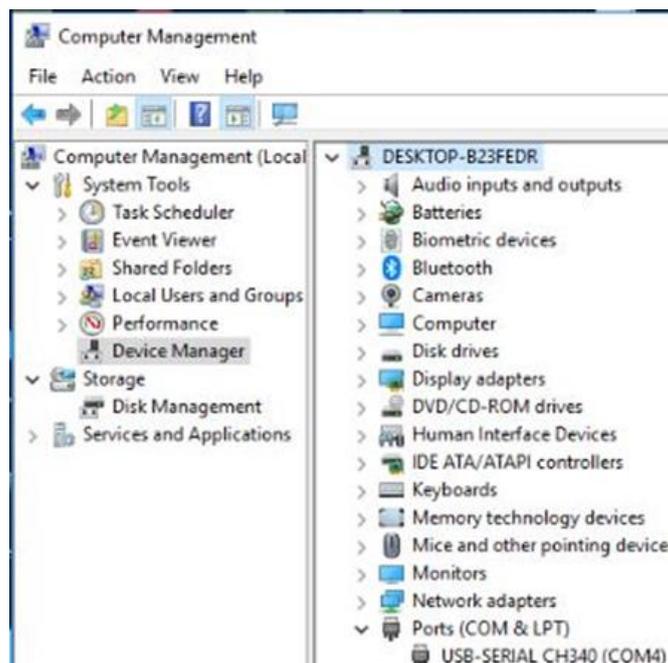


Figure 29: Device manager window

Step 5: Select USB-SERIAL xxxxx

Step 6: Right click on USB-SERIAL xxxxx and click on update driver

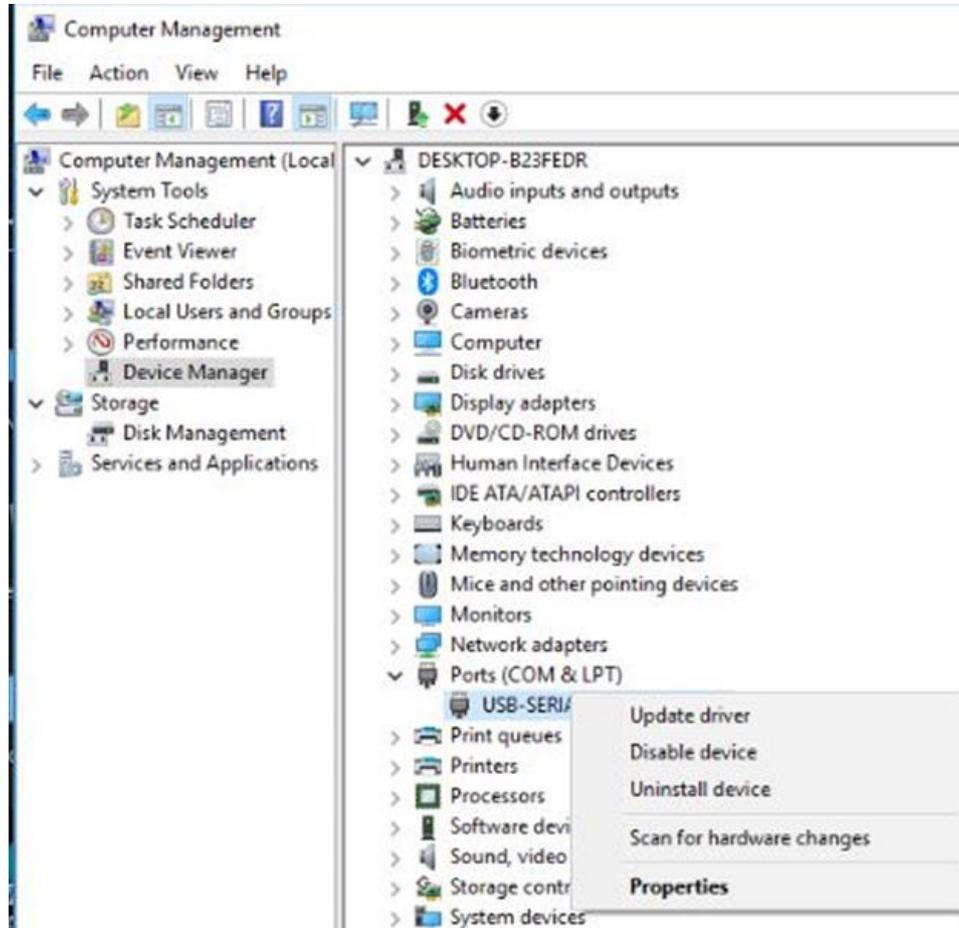


Figure 30: USB port selection

Step 7: Then click on Search automatically as shown below

Update Drivers - USB-SERIAL CH340 (COM4)

How do you want to search for drivers?

→ Search automatically for updated driver software
Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.

→ Browse my computer for driver software
Locate and install driver software manually.

Figure 31: USB driver update path selection

Step 8: Then check whether required drivers are installed properly.

←  Update Drivers - USB-SERIAL CH340 (COM4)

Searching online for drivers...

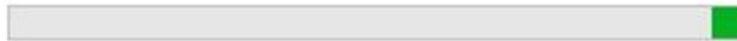


Figure 32: Searching for USB drivers

Step 9: Then proceed with the further procedure run, upload the program.

ii. How to reconfigure your Wifi Board?

Steps is as follows:

Step 1: enter the IP address (198.168.xxx.xxx) URL and will be directly connected to the Web server as shown in the figure below and click on **Configuration** option.



Figure 33: Main menu page of device via http

Step 2: Select the **Configuration Wifi** option from the list.

Armtronix SD Module

ARMtronix

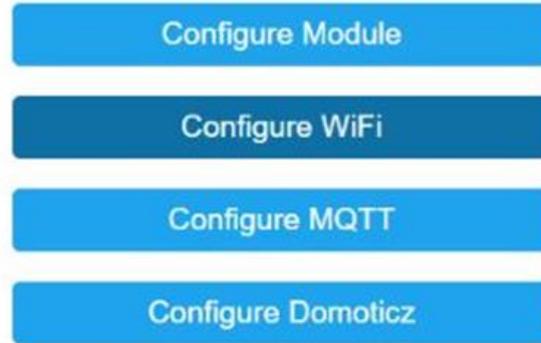


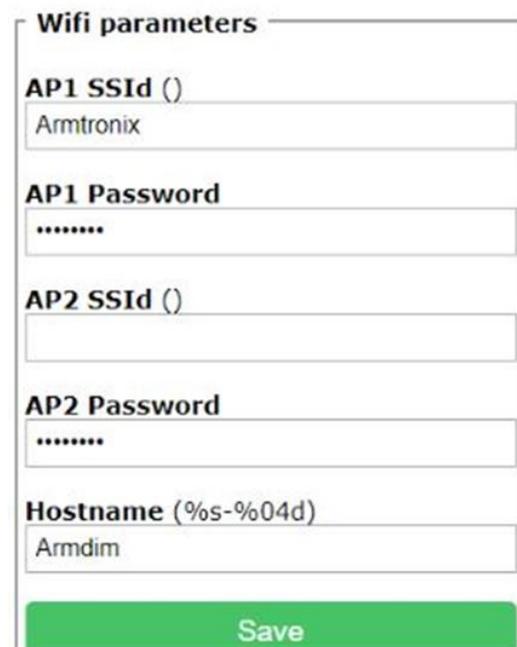
Figure 34: Sub menu to select confiration modes

Step 3: Modify the SSID and Password as per your choice, also one more **SSID** and **Password** option is provided as backup. If the first SSID and Password is not connected to the Wifi then the second SSID and Password (if provided) will automatically get connected. You can provide Hostname (as per your choice).

Armtronix SD Module

ARMtronix

Scan for wifi networks



The image shows a configuration form titled 'Wifi parameters'. It contains the following fields and labels:

- AP1 SSId ()**: Text input field containing 'Armtronix'.
- AP1 Password**: Password input field with masked characters '*****'.
- AP2 SSId ()**: Empty text input field.
- AP2 Password**: Password input field with masked characters '*****'.
- Hostname (%s-%04d)**: Text input field containing 'Armdim'.
- Save**: A green button at the bottom of the form.

Figure 35: Configuration details screen

Step 4: Go to Step 2 and click on “Configure MQTT”

MQTT parameters

Host ()

Port (1883)

Client (DVES_F85231)

User (DVES_USER)

Password

Topic = %topic% (Armtronix)

Full Topic (%prefix%/ %topic%/)

Save

Figure 36: MQTT parameter entry screen

Step 5: Enter the MQTT parameters and click on **SAVE** button.

14. INTEGRATION WITH “OPENHAB”

a. THINGS REQUIRED TO INSTALL OPENHAB ON RASPBERRY PI:

- Raspberry Pi 3 X 1
- SD Card 16 Gb X1
- 2A Power Supply 5V with USB data cable (micro)/USB Charger X1
- Raspberry Pi Case (Optional)
- SD Card Reader
- Monitor
- Keyboard
- Mouse
- HDMI to VGA Cable (optional in case Monitor is of VGA type)
- Amazon echo/ Amazon echo dot

i. Important things to be downloaded:

- NOOBS from Raspberry Pi Website
- (<https://www.raspberrypi.org/downloads/noobs/>)
- SD Card Formatter
(https://www.sdcard.org/downloads/formatter_4/eula_windows/index.html)
- You should have active internet connection to update the OS and install Java updated version
- Putty for SSH in case you are using windows (<http://www.putty.org/>)
- FileZilla (optional) in case you want to transfer some demo files directly to pi
(<https://filezilla-project.org/download.php>)

ii. Installing Raspberry Pi OS and Updating it:

- First Step is to setup your Raspberry Pi. To do that, NOOBS should be extracted into a folder, example let's say folder “NOOBS”.
- In this folder you will find around three folders (defaults, OS, overlays) and around fourteen files (Not mentioned here).
- Plug in your SD card reader with the 16GB SD card inserted to your computer USB. Format it using the SD card formatter (FAT32 format)
- Dump/Copy the contents of NOOBS folder into the formatted SD card (3 Folders + around 14 files)
- Eject the SD card reader, plug out the SD card and place it on your Raspberry Pi SD card holder
- Connect Power Supply/Charger, monitor via HDMI /VGA (HDMI to VGA Cable required in case monitor is of VGA type), Keyboard, Mouse to the raspberry pi.
- Power on the Charger/Power supply
- Raspberry Pi takes you through the setup process and it is quite simple and intuitive.
- A few points to be considered while installing are, select Debian/Raspbian OS while installing, select Us keyboard /Language for most of the countries except UK and give access to the internet, you have to set up your SSID and Password.
- Once the installation is done please cross check that you are connected to the internet.
- If you are not connected to the internet then on the right-hand corner of the computer you

will see an icon/Wifi icon (something like a computer icon) click it, it will show you various Wifi connections available.

- Pick the appropriate SSID. It will ask you to enter its Key (password).
- Once that is done open the terminal (greyish black color monitor image) found on left hand side mid corner.
- Type “**sudo apt-get update**”
- After the source list is updated on the same terminal type “**sudo apt-get upgrade**”
- This will take some time but will update your raspberry pi OS
- This will setup your Raspberry Pi
- Now we have few things to install and will be discussed in the next step.

iii. Installing/Updating required software/drivers:

a) To update Java please follow the instructions mentioned below:

- Check Java version by typing “java -version” in the terminal
- If your java version is lesser than "1.8.0_101" please follow the below commands
- First remove openjdk
- sudo apt-get purge openjdk*
- add packet source
- sudo nano /etc/apt/sources.list
- add the following lines at the end of the sources.list file
- deb http://ppa.launchpad.net/webupd8team/java/ubuntu trusty main
- deb-src http://ppa.launchpad.net/webupd8team/java/ubuntu trusty main
-

b) Install Java 8

- sudo apt-get update sudo apt-get install oracle-java8-installer sudo apt-get install oracle-java8-set-default
- remove old Java
- sudo apt-get purge openjdk*
- sudo apt-get purge java7*
- sudo apt-get autoremove
- check success of upgrade by typing
- java -version
- Check if it is above "1.8.0_101"
-

c) Install Mosquitto Mqtt Broker by using the following command

- sudo apt-get install mosquitto
- To install mqtt client use
- sudo apt-get install mosquitto_client
-

d) Activating SSH if it is not Open in your Raspberry pi's terminal Type

- sudo raspi-config
- Go to Interfacing Options and press enter
- There you will find P2 SSH Enable/Disable SSH
- Select that by using up/down arrow key and press enter

- We are basically enabling the ssh option for further use.

e) Installing OpenHab

We can now continue to install OpenHab on raspberry pi. To do this, open terminal again and type the following commands

- `sudo apt-get update`
- `sudo apt-get upgrade`
- `sudo apt-get install screen mc vim git htop`
- First, add the openHAB 2 Bintray repository key to your package manager and allow Apt to use the HTTPS Protocol:
- `Wget -qO - 'https://bintray.com/user/downloadSubjectPublicKey?username=openhab' | sudo apt-key add -`
- `sudo apt-get install apt-transport-https`
- `echo 'deb https://dl.bintray.com/openhab/apt-repo2 stable main' |`
- `sudo tee/etc/apt/sources.list.d/openhab2.list`
- `sudo apt-get update`

f) Now install OpenHAB with the following commands

- `sudo apt-get install openhab2`
- When you choose to install an add-on, openHAB will download it from the internet on request. If you plan to disconnect your machine from the internet, then you have to install the add-ons package.
- `sudo apt-get install openhab2-addons`
- If everything goes well, you can start openHAB and register, it will be automatically executed at system startup.
- `sudo systemctl start openhab2.service` `sudo systemctl status openhab2.service`
- `sudo systemctl daemon-reload`
- `sudo systemctl enable openhab2.service`
- The first start may take up to 15 minutes, you should be able to reach the openHAB 2 Dashboard at `http://your_raspberry_pi_ip:8080` at this point from any computer in the same network, any browser.
- Once you open it, click on the Paper UI and then go to addons.
- Inside addons click on bindings.
- Here go the Http Binding and install it (here instead of mqtt we are configuring armtronix boards under Http mode).

After installing the http binding go to the tab MISC and install openHAB Cloud Connector. This is required for internet access and to interface it with Alexa.

You can also go to the USER interface tab and install the Basic/Classic UI to control your appliances after integration.

b. STEPS EXPLAINED ON, HOW TO USE OPENHAB IN WINDOWS

Step 1: Open Putty Configuration application and enter the IP address 193.168.1.33 as shown in the figure below and make sure to select the connection type as SSH and click on OPEN. Putty for SSH in case you are using windows (<http://www.putty.org/>)

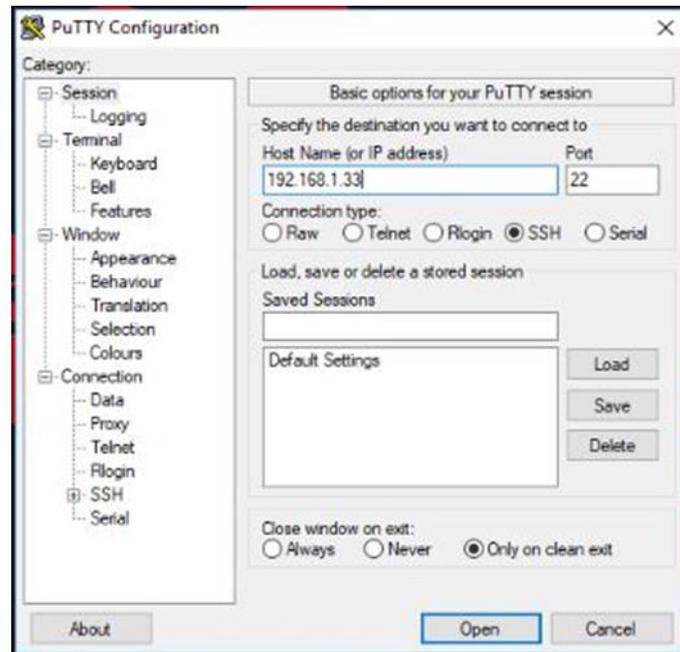


Figure 37: Putty application for Raspberry Pi

Step 2: Login as Pi and enter the password as raspberry.

Login as : pi

root@192.168.1.33's

Password : raspberry

Step 3: Type the command as etc/openhab2 on the Terminal (greyish black color monitor) to enter into Openhab2 file as shown in the figure below.

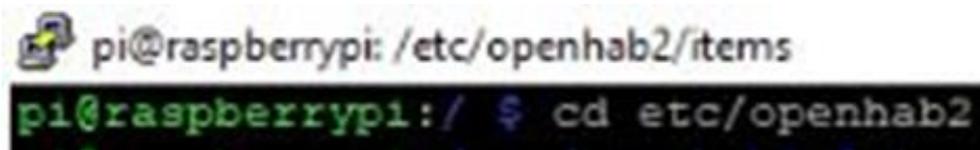


Figure 38: Path for the file

Step 4: cd /etc/openhab2/sitemaps/ You will be in the site map folder you need to create a file with extension as .sitemap, To do that

Type nano

Amtronix_Office.sitemap Example: "Armtronix_Office.sitemap" is the name of that file . Once you press enter, it will give you a blank file for editing. Site map is basically a layout.

Group item=w10 label="Single relay" icon="group"

```
{
    Switch item=w101 label="Relay"
}
```

Step 7: Next thing is to create the file with **.item** as an extension. To do this change to your item directory by typing **cd /etc/openhab2/items**. For http mode, the URL **http://(IP address of the board)/ay?0=1**, (IP address is different for different boards) for toggling (**ON / OFF**).

Example: **http://192.168.1.22/ay?0=1**

```
pi@raspberrypi:/etc/openhab2 $ ls
html icons items persistence rules scripts services sitemaps sounds things transform
pi@raspberrypi:/etc/openhab2 $ cd items
pi@raspberrypi:/etc/openhab2/items $
```

Figure 39: OpenHAB sub folders, Accessing item folder

Group w101

Switch w101 "Relay"(w10,Lights)

```
{http=">[ON:POST:http://192.168.1.22/ay?o=1] >[OFF:POST:http://192.168.1.22/ay?o=1]"}
```

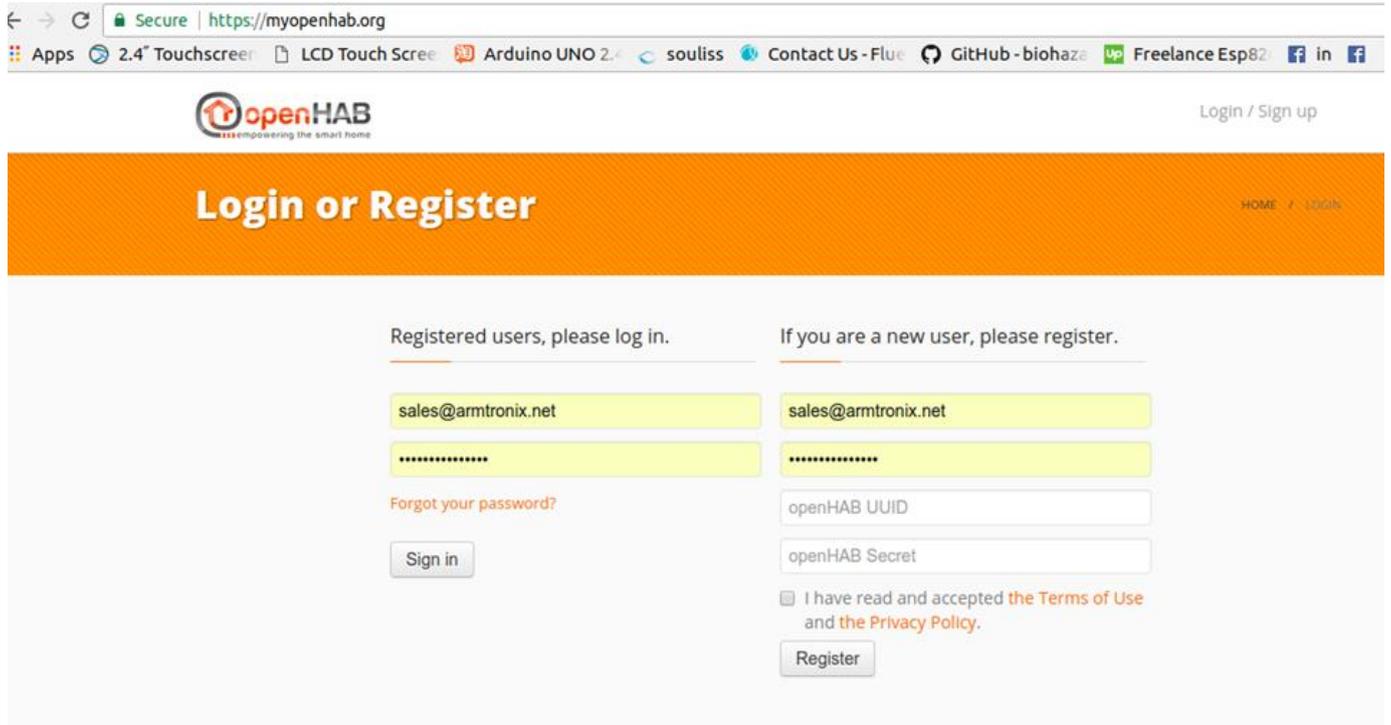
```
Switch w101 "Relay" (w10,Lights) {http=">[ON:POST: http://192.168.1.22/ay?o=1] >[OFF:POST: http://192.168.1.22/ay?o=1] "}
```

Step 8: Code for Item file in MQTT mode, where 'sub' indicates the subscribe topic which is mentioned during the configuration.

Switch sr "Relay"(w10)

```
{mqtt">[broker:test_sub/test/:command:ON:D2_ON],>[broker:test_sub/test/:command:OFF:D2_OFF],>[broker:/w10:state:MAP(w10d2.map)]",autoupdate="false"}
```

Step 9: You can Register yourself in OpenHAB by typing the URL **https://myopenhab.org**, if you have already registered then you can enter the E-mail address and password and then click on **Sign in**, as shown in the figure below.



The screenshot shows a web browser at <https://myopenhab.org>. The page features the OpenHab logo and navigation links. A large orange banner reads "Login or Register". Below this, there are two columns of forms. The left column is for "Registered users, please log in." and contains fields for email (sales@armtronix.net) and password, a "Forgot your password?" link, and a "Sign in" button. The right column is for "If you are a new user, please register." and contains fields for email (sales@armtronix.net), password, openHAB UUID, and openHAB Secret, along with a checkbox for "I have read and accepted the Terms of Use and the Privacy Policy." and a "Register" button.

Figure 40: Login page of OpenHab

Step 10: Once you register yourself with the OpenHab you will be taken to the following page as shown in the figure below.



Figure 41: Menu bard of OpenHab

Step 11: Enter the IP address for which your Board is configured, then you will be taken to the following page as shown in the figure below.

Example: 192.168.1.11

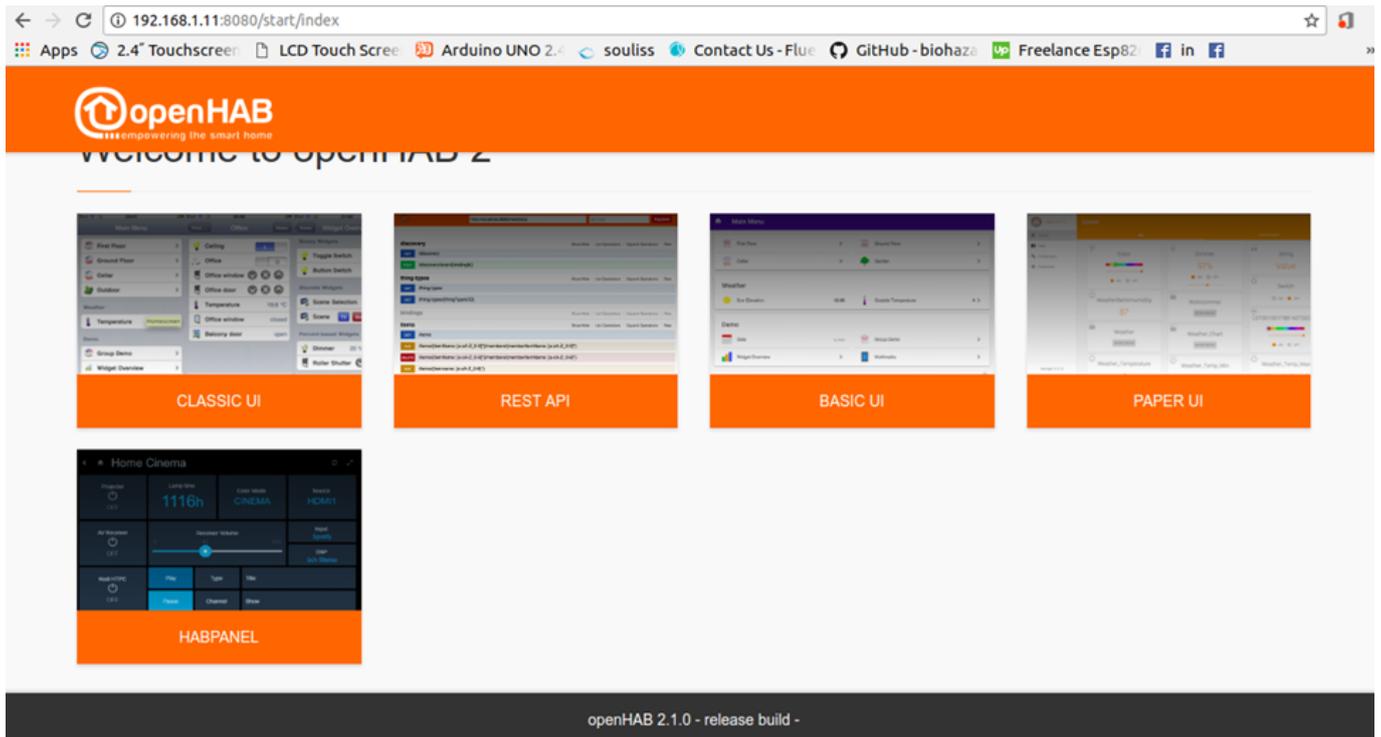


Figure 42: Home page of device with entered IP

Step 12: After creating the Sitemap file in **.sitemap** extension and Item file in **.item** extension as mentioned in the previous steps, the external appearance in OpenHab application is as shown in the figure below.

Example: ARMtronix Office.

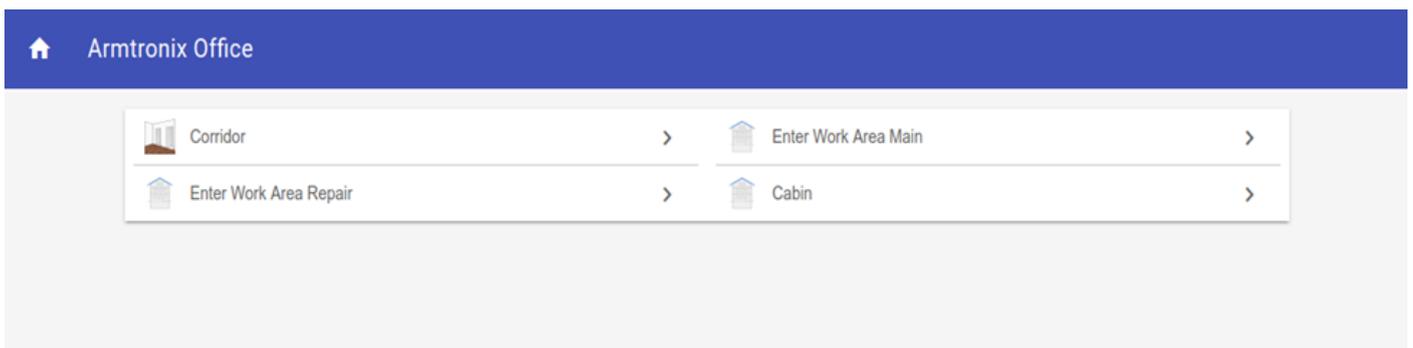
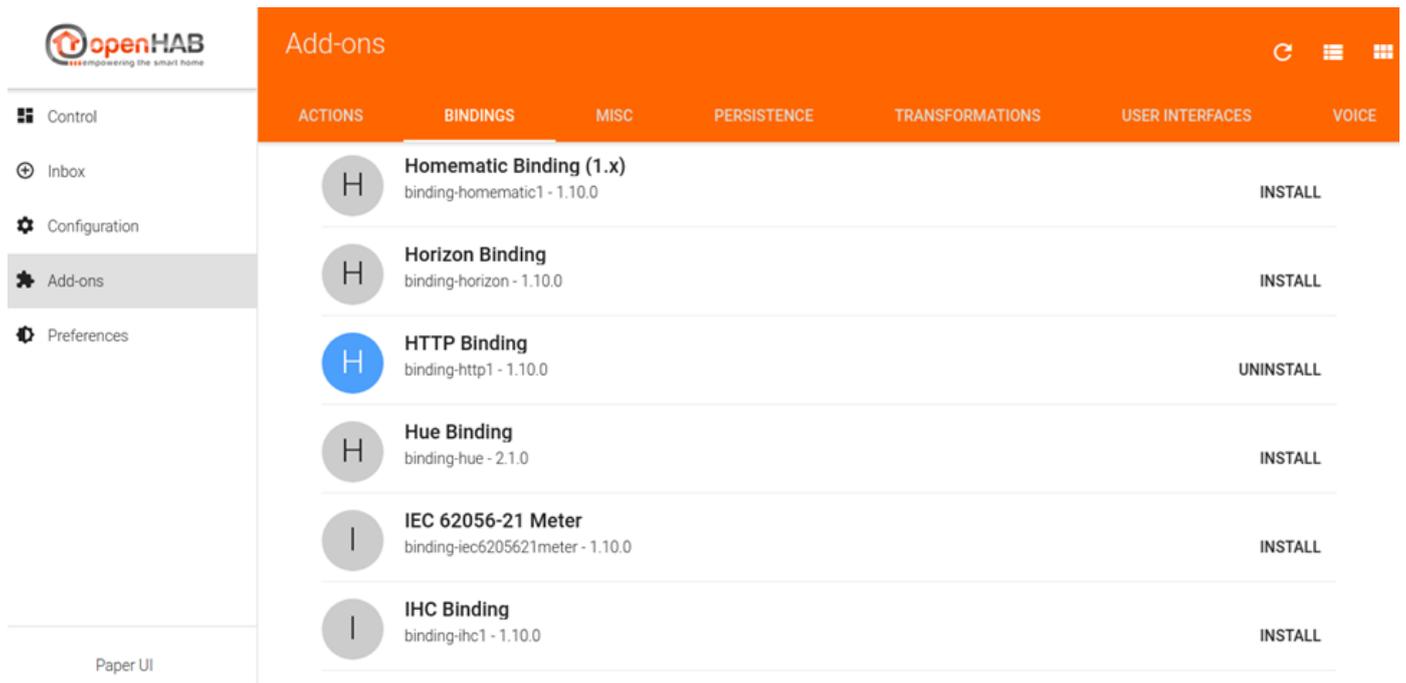


Figure 43: Devices are categorized as Groups under frame

Step 13: In OpenHab application you will be provided with the options such as Control, Inbox, Configuration, Add-on and Preferences. If you click on Add-on option then you will be provided with many options as shown in the figure below.

Example: Add-on option in that Binding option and you can select any binding of your choice and install it (basically ARMtronix boards are configured for HTTP mode), which is suitable for your application.



The screenshot shows the OpenHAB Add-ons interface. On the left is a navigation menu with options: Control, Inbox, Configuration, Add-ons (selected), and Preferences. Below the menu is a 'Paper UI' button. The main area is titled 'Add-ons' and has a top navigation bar with categories: ACTIONS, BINDINGS (selected), MISC, PERSISTENCE, TRANSFORMATIONS, USER INTERFACES, and VOICE. A list of bindings is displayed with their names, IDs, and actions:

| Icon | Binding Name | Binding ID | Action |
|------|-------------------------|----------------------------------|-----------|
| H | Homematic Binding (1.x) | binding-homematic1 - 1.10.0 | INSTALL |
| H | Horizon Binding | binding-horizon - 1.10.0 | INSTALL |
| H | HTTP Binding | binding-http1 - 1.10.0 | UNINSTALL |
| H | Hue Binding | binding-hue - 2.1.0 | INSTALL |
| I | IEC 62056-21 Meter | binding-iec6205621meter - 1.10.0 | INSTALL |
| I | IHC Binding | binding-ihc1 - 1.10.0 | INSTALL |

Figure 44: OpenHAB Add-ons binding option selection

15. FAQ ON OPENHAB

i. How To Configure OpenHab Application?

Consider the following steps to configure your OpenHab application.

Step 1: Make sure that your Smartphone is connected to the OpenHab cloud connector, if not then click on Add-on option in that you have to select the MISC option under that check whether OpenHab cloud connector is installed first, if not install it (It is required basically to interface with Alexa).

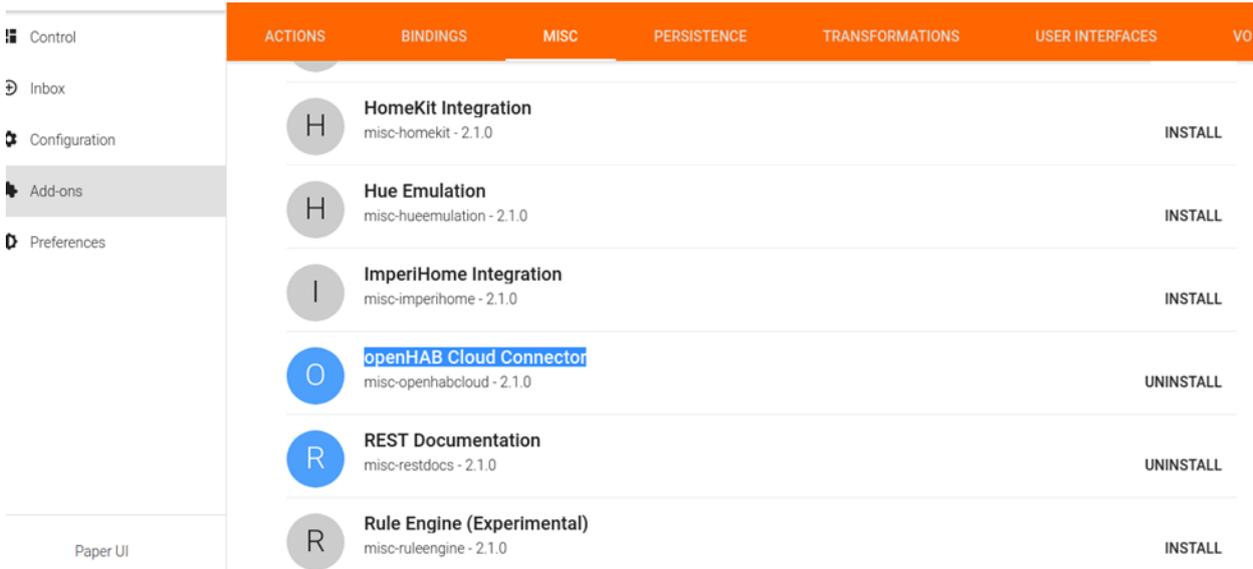


Figure 45: Binding option selected as OpenHAB

Step 2: In OpenHab application there is Configuration option under that select the option Services then a dialog box appears where you are able to see Configure OpenHab cloud and you can check the Mode, for that you will be provided with the dropdown where you can select the suitable Mode for your application, Base URL for OpenHab cloud server and items to expose to apps and you can select by clicking on the checkboxes provided in front of the options and you can save it by clicking on the SAVE option, as shown in the figure below.

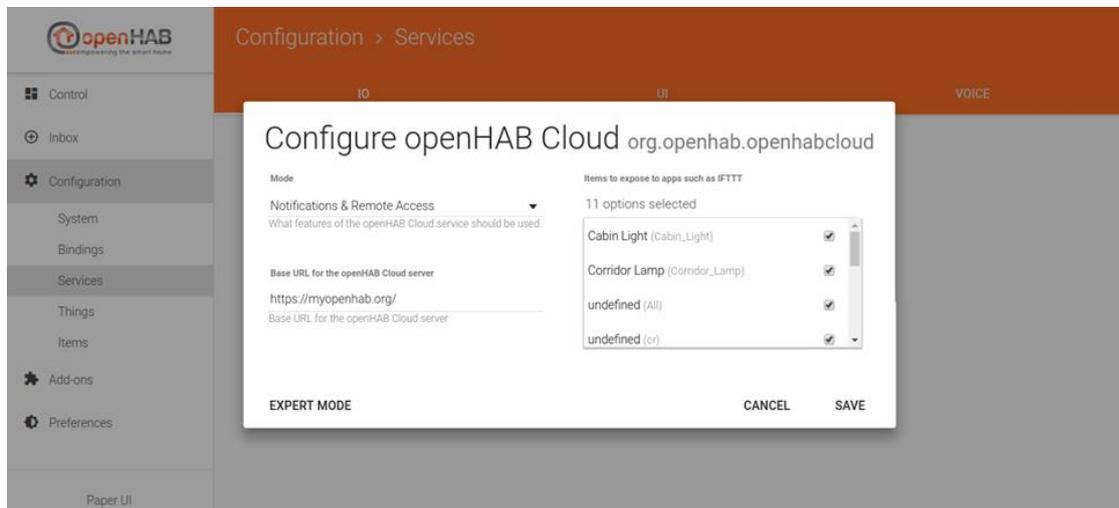


Figure 46: Device selection to configure with OpenHAB

16. INTEGRATION WITH “AMAZON ALEXA”

a. STEPS TO INSTALL ALEXA APP IN SMARTPHONE

- Download the Amazon Alexa app from the Google Play Store.
- Open it and sign in to your Amazon account.
- Open Settings on your Android phone
- Open Apps
- Tap "Default Apps"
- Select "Assist & voice input"
- Choose Alexa instead of Google Assistant.

After you've followed the steps above, you'll see a prompt to talk to Alexa. It'll ask for permissions, so tap "Allow" to let Alexa hear your requests and control devices in your smart home.

Now you'll be able to hold the home button to pull up Alexa. You can use Alexa to turn on smart lights at home ("Alexa, turn on the living room lights," for example)

Following steps explain how to use Amazon Alexa through your Smartphone

Step 1: Click on Amazon Alexa app installed in your Smartphone.



Figure 47: Alexa app bootup screen

Step 2: Click on  the icon which is present on the right-bottom corner of your Smartphone as shown in the figure below.



Figure 48: Alexa app home screen

Step 3: Click on **All Devices** option to check which devices are connected to the Amazon Alexa as shown in the figure below.

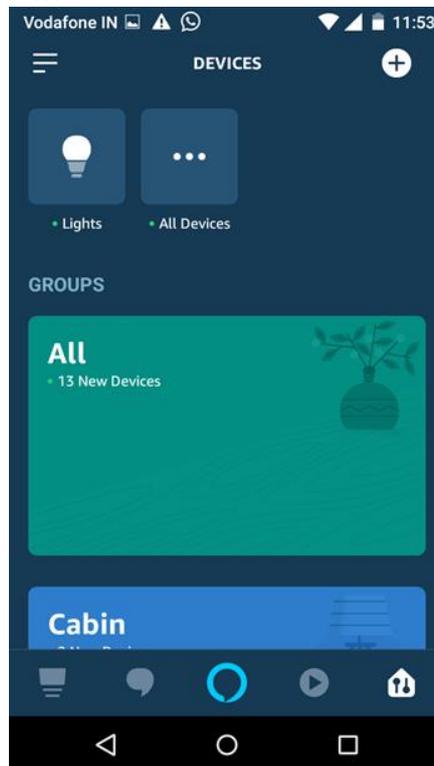


Figure 49: Check # of devices connected

Step 4: You can make groups of your choice and it shows how many devices are connected to that particular group as shown in the figure below,

Example: Cabin, Lab, Entrance, Office etc.

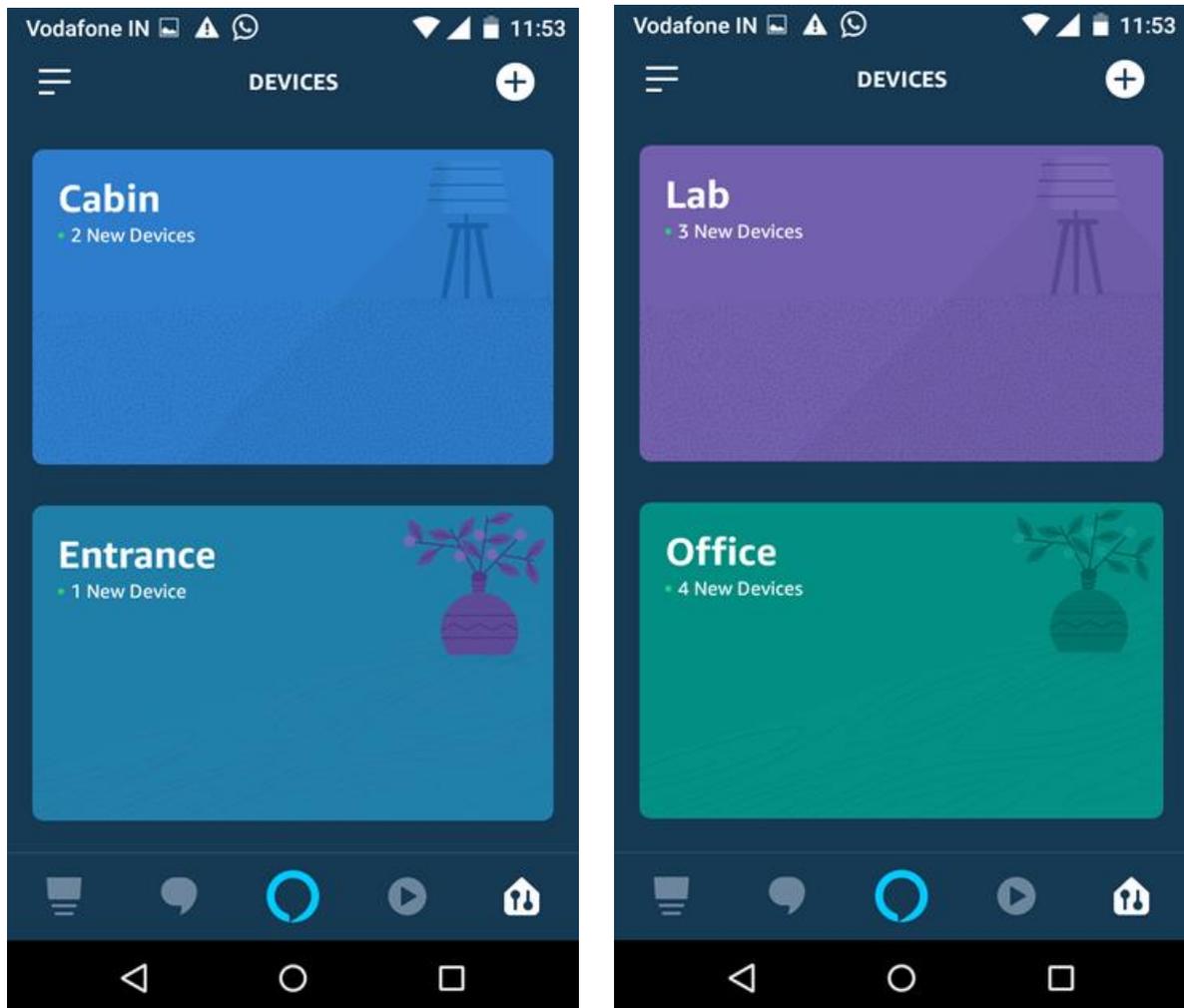


Figure 50: Devices divided as groups in Alexa app

Step 5: Upon voice recognition by Alexa you can control Loads (Fan, Lights) by giving commands.

Example: **Alexa turn ON Cabin Lights**
 Alexa turn OFF Cabin Lights

Alexa turn ON Office Lights
Alexa turn OFF Office Lights

Note: Ensure that to interface with Alexa you are connected to the "OpenHAB Cloud Connector".

Step 6: Ensure that when you click on All option under groups, it displays the device name which you have mentioned during the configuration time as shown in the figure below.

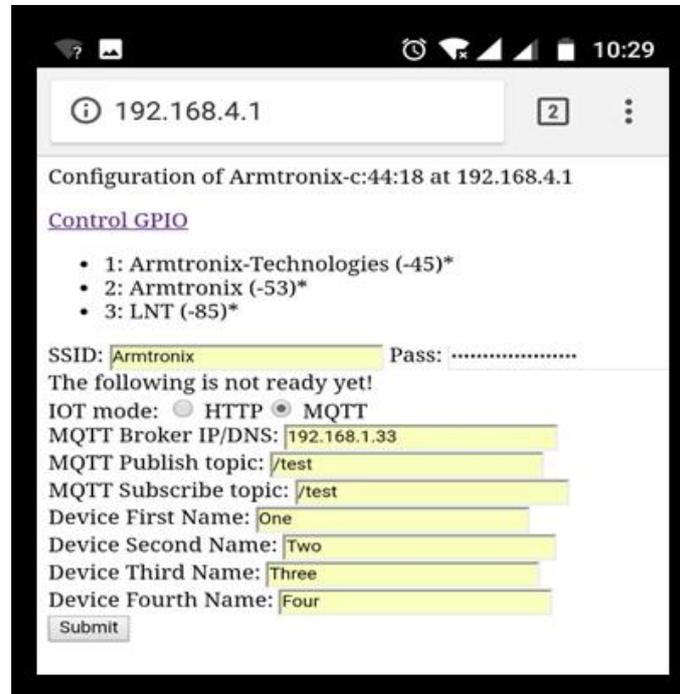


Figure 51: Names are defined for loads

17. INTEGRATION WITH “GOOGLE HOME”

a. FOLLOWING STEPS EXPLAIN, HOW TO INTEGRATE GOOGLE HOME WITH OPENHAB

To use the OpenHab integration for Google Assistant on your smartphone, you will need the Google Assistant or Google Home app (iOS or Android)

Before you start integration, make sure that the OpenHAB is installed and configured with our devices.

Step 1: Make sure Google Play Services is up to date

- Visit "Google Assistant" app entry in Google Play Store on Android
- Set up the voice-activated speaker, Pixel, or Android phone (version 6+) with the same test account
- Make sure you're the correct user
- Start the updated Google Assistant app on your phone

Step 2: Open Google Assistant app in your Smartphone as shown in the figure below

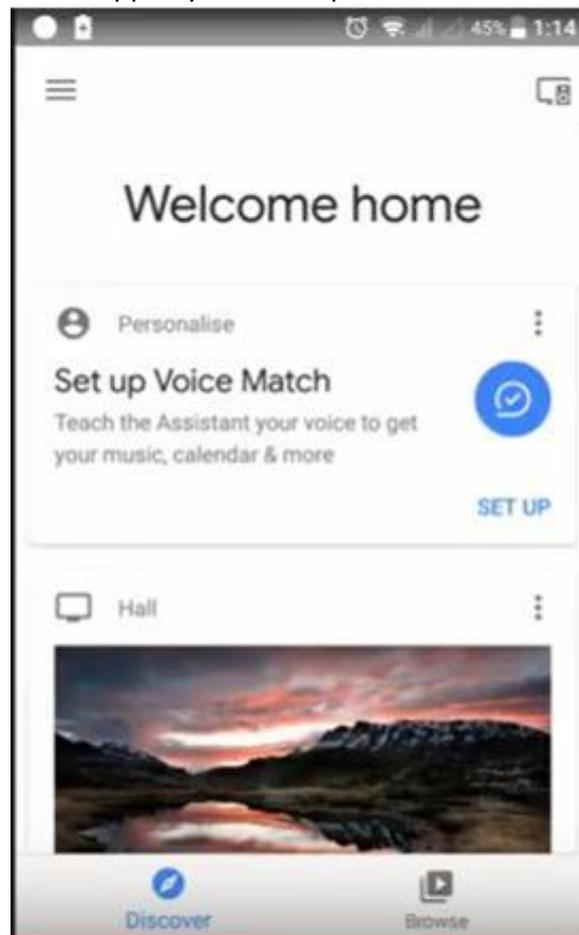


Figure 52: Google Home app’s Home screen

Step 3: Go to Settings, under that go to services there you will be able to see the option Home control, click it.

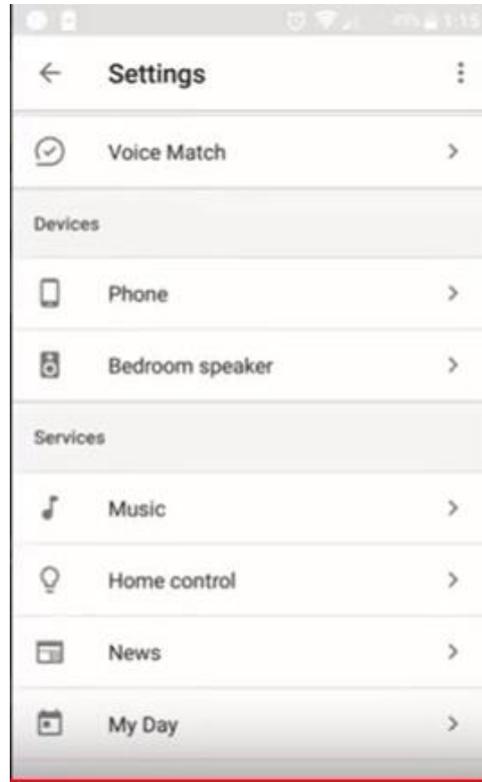


Figure 53: Google Home app's Settings screen

Step 4: Click on the “+” sign under Home control which will take you to the Add devices

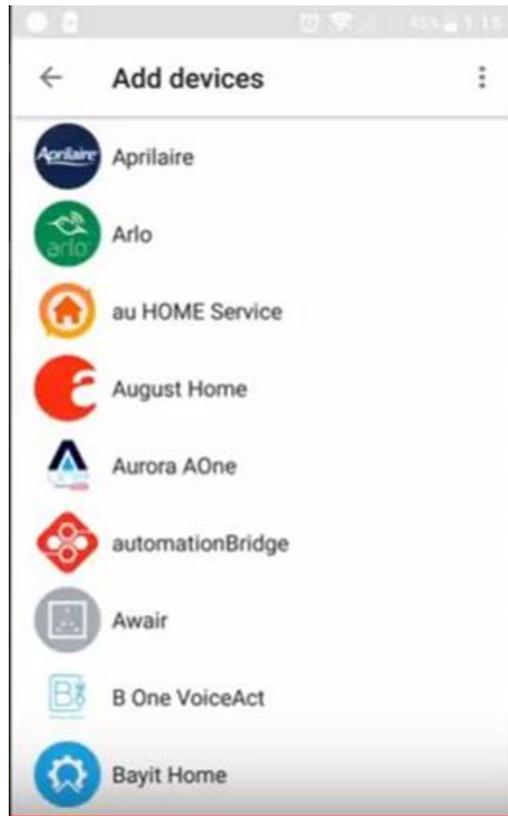


Figure 54: Google Home app's screen to Add IoT devices

Step 5: Under Add devices select the option OpenHab as shown in the figure below

Note: Ensure that your Smartphone is connected to the OpenHab cloud connector

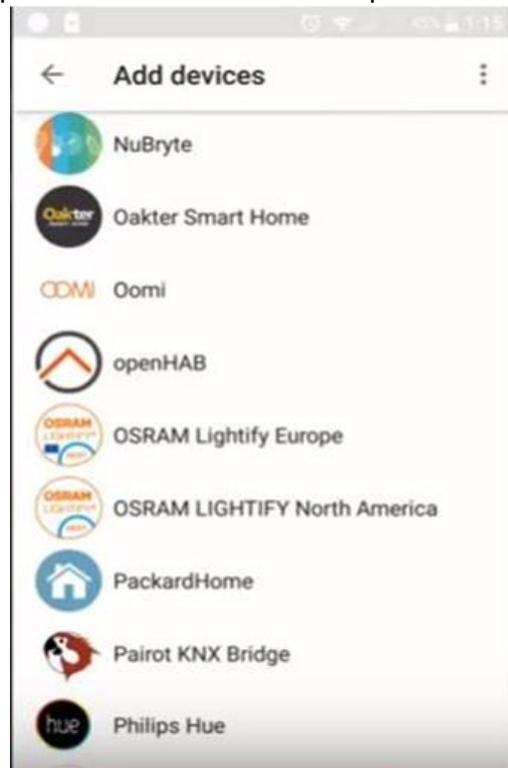


Figure 55: Screen to Add cloud service provider

Step 6: Click on OpenHab and enter the E-mail ID and Password.



Figure 56: Login Screen

Step 7: Click on Allow button as it is asking permission for Authentication with Google home.

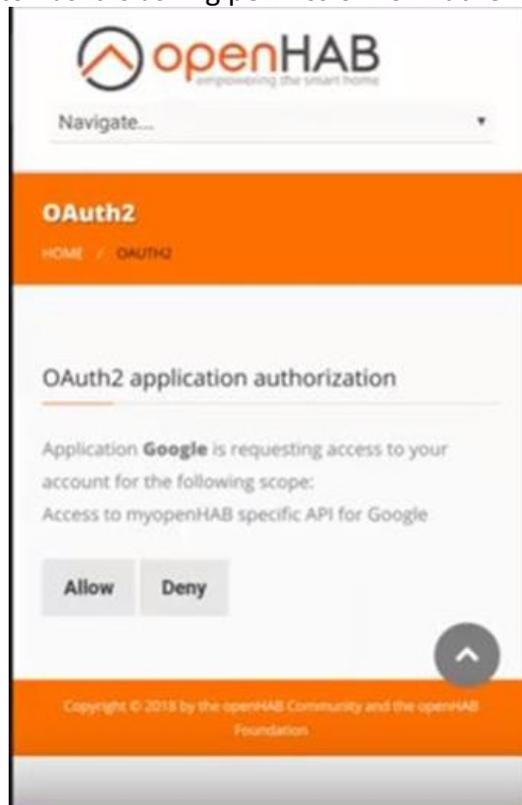


Figure 57: Apps's permission authentication screen

Step 8: It will display the devices connected with the OpenHab as shown in the figure below.

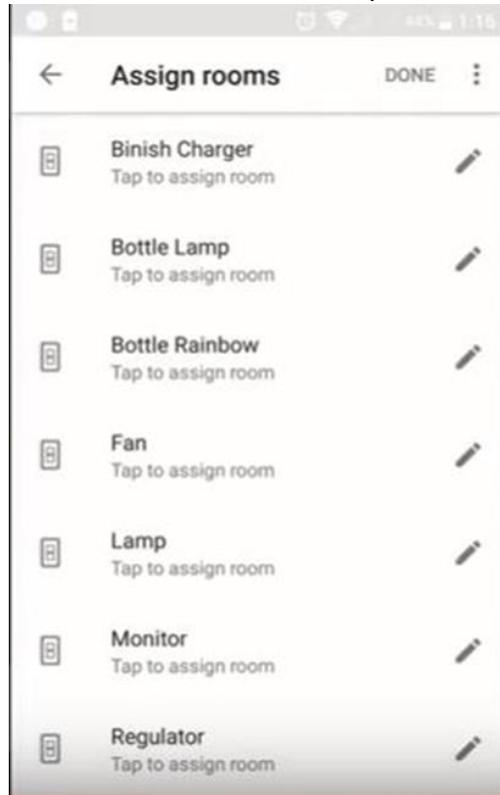


Figure 58: List of added devices

Step 9: Now it is ready to give commands and upon voice recognition Google home respond to the commands given.

Example: **Google turn on fan**

Google turn off fan

Google turn on Lights

Google turn off Lights

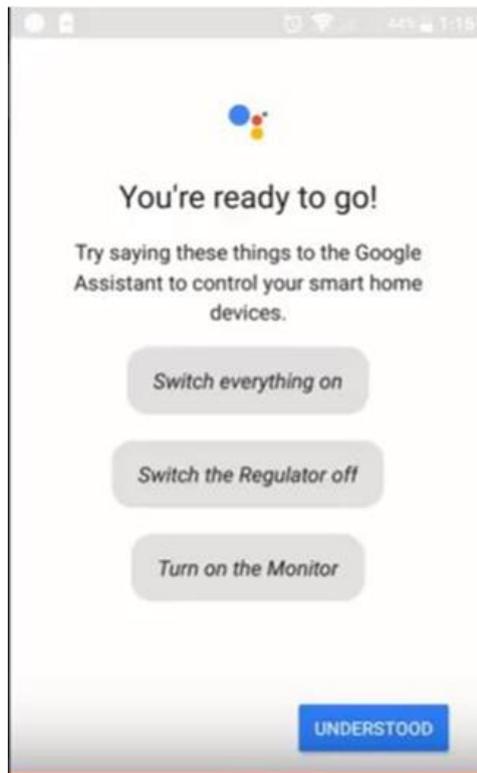


Figure 59: Device control screen



DOCUMENT #: BA015

DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, IoT HEAVY DUTY RELAY WITH POWER MONITORING MODULE.

IMPORTANT NOTICE

ARMtronix Technologies LLP and its subsidiaries reserve the right to make corrections, enhancements, improvements and other changes to its products, services and documents and to discontinue any product or service. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to ARMtronix Technologies LLPs terms and conditions of sale supplied at the time of order acknowledgment.

The information in this document is subject to update without notice. The contents of this document thereof must not be used for any unauthorized purpose.

-----END OF DOCUMENT-----