DOCUMENT #: A0011

DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI /BT QUAD RELAY BOARD.

DESCRIPTION DOCUMENT FOR WIFI/BT QUAD RELAY BOARD HARDWARE REVISION 0.1

Department	Name	Signature	Date
Author			
Reviewer			
Approver			

Revision History

Rev	Description of Change	Effective Date
А	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.



	DOCUMENT #: A0011	
DOCUMENT NAME: DESIGN DESCRIPTION, WIFI,	/BT QUAD RELAY BOARD.	
	Table of Contents	
Revision History		1
1. ABBREVIATIONS		
2. REFERENCES		
3. PURPOSE		
4. SCOPE		
5. SAFETY AND WARNING		
6. PRODUCT FEATURES		5
7. PRODUCT DESCRIPTION		5
a. PHYSICAL DESCRIPTION		5
b. FUNCTIONAL DESCRIPTION		5
8. SYSTEM OVERVIEW		6
9. TECHNICAL SPECIFICATION		6
a. ELECTRICAL SPECIFICATION		6
b. MECHANICAL SPECIFICATION		6
10. ELECTRICAL CONNECTIONS		7
a. HEADER PIN CONFIGURATION		9
i. HEADER J2		9
ii. HEADER J3		9
iii. HEADER J4		10
b. APPLICATION WIRING DIAGRAM		10
11. MQTT COMMANDS TO READ INPUTS		12
a. COMMANDS TO TRIGGER RELAY THROUGH	SUBSCRIPTION TOPIC	12
b. COMMAND TO RESET THE BOARD		13
<u>c.</u> REPONSE RECEIVED FROM THE BOARD THRO	OUGH PUBLISHING TOPIC	
12. HOW TO USE THE PRODUCT		
a. STEPS TO CONFIGURE THE DEVICE TO NETW	ORK HOSTED BY YOU:	
b. CONNECT VIA MQTT MODE		15
<u>c.</u> CONNECT VIA MQTT MODE		
d. STEPS TO CONNECT SMARTPHONE TO MQT	T BROKER / WIFI ROUTER / ACCESS POINT:	19
e. STEPS TO TEST THE DEVICE USING SMARTPH	IONE AND MQTT BROKER:	
<u>f.</u> CONTROL OUTPUTS VIA SMARTPHONE:		
g. RESET THE DEVICE USING MQTT COMMAND	VIA SMARTPHONE	
<u>h.</u> READ DIGITAL INPUTS VIA SMARTPHONE		25
13. Openhab Example		
a. Example of Openhab files in MQTT mode		
14. HOW TO CUSTOMISE FIRMWARE		
a. STEPS TO LOAD PROGRAM TO ESP32:		
IMPORTANT NOTICE		

>

DOCUMENT #: A0011

DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

Table of figures

Figure 1: Block Diagram	5
Figure 2: Header and Switch Details	7
Figure 3: Header Pin number references	8
Figure 4: AC Input connection	8
Figure 5: Relay Dry contact pin-out	8
Figure 6: Application wiring example	10
Figure 7: Application wiring example of DC connections	11
Figure 8: Application wiring example of AC connections	12
Figure 9: Available Wifi networks searched	14
Figure 10: Smartphone Connected to Wifi hosted by board	14
Figure 11: Default IP address entered in the Web browser	15
Figure 12: Accessed webpage of the device	15
Figure 13: Entered all the required details	16
Figure 14: Accessed webpage of the device	17
Figure 15: Entered all the required details	18
Figure 16: Smartphone searched for available Wifi networks	19
Figure 17: Trying to connect to pre-configured MQTT broker	19
Figure 18: Smartphone connected to MQTT broker	20
Figure 19: MyMQTT app menu page	21
Figure 20: MQTT broker IP address and port number entered	21
Figure 21: Saved the settings	22
Figure 22 Entered topic and message to control outputs	23
Figure 23: Message and topic published to control outputs	23
Figure 24: Taped on the default screen	24
Figure 25: Clicked on the publish option and entered the message to be displayed on LCD	24
Figure 26: Published the message by clicking on Publish button	25
Figure 27: Tapped on the home screen	25
Figure 28: Clicked on the Subscribe option	26
Figure 29: Entered the Subscription topic and clicked on the Add button	26
Figure 30: Dashboard window to monitor status of Digital Inputs	27
Figure 31: Openhab image of 4 relay board	28
Figure 32: Program Opened in IDE	30
Figure 33: Board Selection	31
Figure 34: Baudrate selection	32



DOCUMENT #: A0011 DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

1. ABBREVIATIONS

Term	Description	
AC	Alternating Current	
AP	Access Point	
BT	Bluetooth	
COM	Common	
DC	Direct Current	
HTTP	Hypertext Transfer Protocol	
Hz	Hertz	
MQTT	Message Queue Telemetry Transport	
NC	Normally Closed	
NO	Normally Open	
UART	Universal Asynchronous Receiver Transmitter	
USB	Universal Serial Bus	

2. REFERENCES

Company Website link	https://www.armtronix.in
Github Weblink	https://github.com/armtronix/Wifi_Bt_Esp32_Quad_Relay
Youtube Weblink	https://www.youtube.com/watch?v=wqkkvoWiCZI

3. PURPOSE

The purpose of this document is to outline the design description for the Wifi/BT Four Relay Board. It provides a high level summary of the product.

4. SCOPE

This document describes system architecture which includes Power supply, relay, WiFi/BT ESP32 Module and UART to USB converter.

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 Wiring diagram is available on our instructables page and github. Please refer it.

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.

AR

DOCUMENT #: A0011

DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

6. PRODUCT FEATURES

- Works directly with AC power 100 240 V AC 50-60 Hz.
- Device firmware can be updated/reloaded/changed as per user requirement.
- Four Dry contact relay output with COM, NO and NC accessible to user.
- Board can handle up-to 4 Amps of current at relay output.
- WiFi with MQTT or HTTP protocol
- On board USB UART converter to program WiFi Module
- 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 configuration and Reset function.
- Board is compatible and configurable to Amazon Alexa.

7. PRODUCT DESCRIPTION

a. PHYSICAL DESCRIPTION

- AC to DC Power supply module
- Mechanical Relay 4 numbers
- > Wifi Module
- USB-UART converter

b. FUNCTIONAL DESCRIPTION Block Diagram



Figure 1: Block Diagram

Wifi/BT relay board is ESP32 based relay board, has on-board power supply module which takes standard AC power as input and provides required DC power as output. The DC power is used to power-up Wifi/BT module used on board to establish Wifi communication with mobile phones or wifi routers or access points. There are four relays mounted on board to control (ON/OFF) four external electrical loads independently from a mobile application using MQTT/HTTP protocol.

DOCUMENT #: A0011 DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

8. SYSTEM OVERVIEW

1. AC to DC Power supply module

AC to DC converter is power supply module manufactured from Hi-Link part number HLK-PM01. This power supply module rectifies and regulates voltage from 230 V AC to 5 V DC with output current capacity of 0.6A DC. The power of HLK-PM01 is at maximum of 3W.

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/BT Module

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

3. Mechanical Relay – 4 Numbers

All relays are powered by 5 V DC. The three load terminals (COM, NO and NC) of all relays are given accessible to user to control loads independently. A driver circuit with an opto-isolator is used to drive the relay.

4. USB-UART converter

USB-UART converter is an integrated chip used to convert serial UART data to high speed USB to program the Wifi module using Arduino IDE. This is much user friendly to customize the code and reload it. A micro USB connector given on board to make hassle free connection between computer and Wifi Four relay board for programming purpose.

9. TECHNICAL SPECIFICATION

a. **ELECTRICAL SPECIFICATION**

Input Specifications				
Description	Min	Тур	Max	Unit
Voltage AC	100	220	230	Volts
Current AC	-	0.1	-	Amps
Power AC	-	3	-	Watts
Frequency	50	-	60	Hz

Relays Output Specifications (Maximum)				
Description	Min	Тур	Max	Unit
Voltage AC	-	-	240	Volts
Current AC	-	-	3	Amps
Power AC	-	-	980	Watts
Voltage DC	-	-	24	Volts
Current DC	-	-	3	Amps

b. MECHANICAL SPECIFICATION

- Mechanical Dimensions of PCB are 140 x 60 x 20 mm (Length x Width x Height)
- Mounting Holes are compatible with M3 screws pan head dia maximum of 5.65mm.



Figure 2: Header and Switch Details

Description of Header and Switches shown in Figure 1:

- 1. S1 Button to GPIO_0.
- 2. S2 Button to reset the ESP.
- 3. Power Input AC input terminal block.
- 4. J4 Relay Selection Header.
- 5. J2 and J3 Headers are compatible to standard ESP32 extra GPIO headers.
- 6. U5 Micro USB for programming.





Table 2: Header J3 Pin configuration

Table 1 and 2, shows the header J2 and J3 which are in compatible with Node MCU headers. Freely available GPIOs are also shown in connector, can be used for user application.

9 | P a g e

DOCUMENT #: A0011

DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

iii. HEADER J4			
Header Pin Number	Pin Name	Header Pin Number	Pin Name
1	GPIO15	2	Relay 01
3	GPIO4	4	Relay 01
5	GPIO12	6	Relay_02
7	GPIO13	8	Relay 03
9	GPIO14	10	Relay 04
11	GPIO2	12	Test_LED (D3)

Table 3: Header J4 Pin Configuration

GPIOs mentioned in the Table 3 are used to control relays. By default the GPIOs will be shorted (using removable jumpers) with respective relay pins as mentioned in above table. If you want to use those GPIOs for your own application instead of relays, then you have to disconnect by opening the jumper and make use of them.

b. APPLICATION WIRING DIAGRAM



terminal of the relay.



Figure 7: Application wiring example of DC connections

Note: Virtual switch GPIOs are 3.3 V DC connections, please do not connect AC lines to it. Connecting AC lines to it, may damage the board and may lead fire and cause hazardous effects.

Physical switch are connected to 4 GPIOs (refer to Table 4).

Relay No.	Relay GPIOs	Header Pin Number
Relay 01	GPIO 04	GPIO 33
Relay 01	GPIO 12	GPIO 32
Relay_02	GPIO_13	GPIO_35
Relay 03	GPIO 14	GPIO 34

Table 4: Virtual switch GPIO details

Figure 8: Application wiring example of AC connections

Outputs of relay 1-4 are connected to Socket 1-4 through NO and COM pin of relay. The socket will get power when the relay is triggered by virtual switch or mobile application. The advantage of this configuration is, it will act as two-way switch and

you can Turn-On the load through physical switch and Turn it OFF through relay or vice-versa.

The device can also be connected to Amazon Alexa by configuring using Alexa mobile App and OpenHab by configuring it with openhab sever.

11. MQTT COMMANDS TO READ INPUTS

a. COMMANDS TO TRIGGER RELAY THROUGH SUBSCRIPTION TOPIC

- R4_ON ; Will turn-on the Relay_1
- R4_OFF ; Will turn-off the Relay_1
- R12_ON ; Will turn-on the Relay_2
- R12_OFF ; Will turn-off the Relay_2
- R13_ON ; Will turn-on the Relay_3
- R13_OFF ; Will turn-off the Relay_3

R14_ON ; Will turn-on the Relay_4

R14_OFF ; Will turn-off the Relay_4

Note: "Subscription Topic" will be the name entered while configuring the board.

DOCUMENT #: A0011 DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

b. COMMAND TO RESET THE BOARD

Reset ; will reset the board and board will start hosting an AP.

Note: "Subscription Topic" will be the name entered while configuring the board.

c. REPONSE RECEIVED FROM THE BOARD THROUGH PUBLISHING TOPIC

On change in status of Relay_1 R04isON R04isOFF

On change in status of Relay_2 R12isON R12isOFF

On change in status of Relay_3 R13isON R13isOFF

On change in status of Relay_4 R14isON R14isOFF

Note: "Publishing Topic" will be the name entered while configuring the board.

DOCUMENT #: A0011 DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

12. HOW TO USE THE PRODUCT

a. STEPS TO CONFIGURE THE DEVICE TO NETWORK HOSTED BY YOU:

- i. Switch ON the device.
- ii. Make sure that Power presence indication Green LED is glowing.
- iii. Take any Smartphone.
- iv. Switch ON Wifi in it. (make sure that, its Mobile Data connection is turned OFF).
- v. Search for available Wifi networks in the range

Figure 9: Available Wifi networks searched

vi. You will observe one of the available Wifi network as "Armtronix-xx-xx-xx". Where xx: is last 6 digits of MAC address of the particular device. Click on that particular available network connect your smart phone to it. So in this scenario, the device is 'Wifi Host' and Smartphone is 'Wifi Client'.

Vodafone IN – Vod	afone IN	79% 📋 10:28
		-©-
	Vodafona IN 🔻	Vorlafone IN
44-18	Volutione IIV	VOUBIONE IN
A	0	8
Vodafone IN	Battery Saver	Do not disturb 👻
	••	
්රි Sat 7:30 AM	8 /	¢ ^
8		
		crosoft Play Store
Vodafo	one IN - Vodafor	

Figure 10: Smartphone Connected to Wifi hosted by board

1 1 1 N	DOCUMENT #: A0011
	DOCUMENT REV: A
DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.	

vii. Open any web browser, enter default IP address 192.168.4.1 of the device when it is hosting its own Wifi network and click enter.

S == L C T:23
192.168.4.1
192.168.4.1 192.168.4.1
192.168.4.1
Search or type web address 🌵
I A I T IRCTC Next Welcome to IRCTC Next Buy and Sel
abc अ ⊽
1 2 3 4 5 6 7 8 9 0
qwertyuiop
as dfghjkl
🛧 z x c v b n m 🛚
?123 / 🌐 📃 . 🔿

Figure 11: Default IP address entered in the Web browser

b. CONNECT VIA MQTT MODE

i. Clinking on Enter button after entering default IP address, you will be able to access its webpage as shown in Figure 9.

2	•
	•
4	

1	DOCUMENT #: A0011	
	DOCUMENT REV: A	
DOCUMENT NAME: DESIGN DESCRIPTION. WIFI/BT OUAD RELAY BOARD.		
ii. In the accessed webpage, fill-in	all the required details like:	
> SSID	: SSID of Access Point	
➢ Pass	: Password of Access point	
IOT Mode	: MQTT	
MQTT Broker IP/DNS	: xxx.xxx.xxx.xxx (Ex. 192.168.0.1)	
Publish to Topic 1 (IN)	: /I/xxx (Ex. /I/008)	
Subscribe to topic 1 (OP)	:/O/xxx(Ex./O/008)	
Device Name	: Alexa command name; Which we ask Alexa to trigger	
Ex. Alexa turn ON light	; "Light" is the name of the device	
Ex. Alexa turn ON Fan	; "Fan" is the name of the device	
Ex. Alexa turn ON Charger	r ; "Charger" is the name of the device	
Ex. Alexa turn ON Tube light ; "Tube Light" is the name of the device		
2 🛋	(i) Tr 10:29	
· 192.168.4.1	2:	
Configuration of Armtronix-c:44:18 at 192.168.4.1		

Control GPIO		
 1: Armtronix-Technologie 2: Armtronix (-53)* 3: LNT (-85)* 	s (-45)	*
SSID: Armtronix	Pass:	
The following is not ready yet!		
IOT mode: 🔍 HTTP 🖲 MQTT		
MQTT Broker IP/DNS: 192.168.1.33		
MQTT Publish topic: /test		
MQTT Subscribe topic: /test		
Device First Name: One		
Device Second Name: Two		
Device Third Name: Three		
Device Fourth Name: Four		
Submit		
	_	

iii. After entering all the required details, click on Submit button. It will save the parameters you entered and reboot the device and acknowledge the user in the webpage.
 Do not turn OFF the device, it will automatically reboot.

c. CONNECT VIA MQTT MODE

i. Clinking on Enter button after entering default IP address, you will be able to access its webpage as shown in Figure 9.

?	Õ 🔭 🖌 🗖	10:28
(i) 192.168.4.1	2	•••
Configuration of Armtronix-c:44:18 at 192.168.4.1		
Control GPIO I : Armtronix-Technologies (-45)* 2 : Armtronix (-53)* 3 : LNT (-85)* SSID: Pass: The following is not ready yet! IOT mode: HTTP MQTT MQTT Broker IPDNS: MQTT Fublish topic: Device First Name: Device Second Name: Device Second Name: Device Third Name: Submit		

Figure 14: Accessed webpage of the device

In the accessed webpage, fill-in all the required details like: ii.

- > SSID
- > IOT Mode

> Pass

- Device Name
- : SSID of Access Point
- : Password of Access point
- : HTTP
- : Alexa command name; Which we ask Alexa to trigger
- Ex. Alexa turn ON light ; "Light" is the name of the device
- Ex. Alexa turn ON Fan
- ; "Fan" is the name of the device
- **Ex.** Alexa turn ON **Charger** ; "Charger" is the name of the device
- **Ex.** Alexa turn ON **Tube light** ; "Tube Light" is the name of the device

Figure 15: Entered all the required details

iii. After entering all the required details, click on Submit button. It will save the parameters you entered and reboot the device and acknowledge the user in the webpage.
 Do not turn OFF the device, it will automatically reboot.

DOCUMENT #: A0011

DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

d. STEPS TO CONNECT SMARTPHONE TO MQTT BROKER / WIFI ROUTER / ACCESS POINT:

- i. Disconnect Smartphone from any other Wifi network if connected.
- ii. Search for available Wifi network where the MQTT broker / WIFI ROUTER / ACCESSVPOINT is running.

In our case it is "Armtronix-Home" is the wifi network where our MQTT broker is running.

Figure 16: Smartphone searched for available Wifi networks

iii. Click on that particular available network to connect your smart phone to it.

	DOCUMENT #: A0011
	DOCUMENT REV: A
DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD	RELAY BOARD.
``` 😪 ⊑ 🏯	🔞 🗣 🔟 📕 17:31
Wi-Fi	<b>\$</b>
On	
Armtronix-Hor Connected, no Ir	ne Iternet
Armtronix	

Armtronix-Technologies

Figure 18: Smartphone connected to MQTT broker

shubham

Add network

DLink

1°.	DOCUMENT #: A0011
	DOCUMENT REV: A
DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD	RELAY BOARD.

### e. STEPS TO TEST THE DEVICE USING SMARTPHONE AND MQTT BROKER:

- i. Install 'MyMQTT' Android app in to a Smartphone you would use for testing.
- ii. Open an app 'MyMQTT' app Smartphone.

![](_page_20_Picture_4.jpeg)

Figure 19: MyMQTT app menu page

- iii. Click on settings option.
- iv. Enter MQTT broker IP address and default Port number as 1883 (if not changed) Our MQTT broker IP address is 192.168.0.1

II	©▼⊿⊿ ■	17:30
Con Settings		¥
100.168.1.1		
1883		
Username (optional)		
Password (optional	)	
Save		

Figure 20: MQTT broker IP address and port number entered

⊥ <u></u> *	DOCUMENT #: A0011
	DOCUMENT REV: A
<b>DOCUMENT NAME:</b> DESIGN DESCRIPTION, WIFI/BT QUAD	RELAY BOARD.

v. On the completion of your IP address and port number entry, Save the settings by clicking on **Save** button. Popup will indicate once the settings saved.

🎫 🕄 🖪 🏯	© ▼⊿ ⊿	17:30
Certings		¥
100.168.1.1		
1883		
Username (optional)		
Password (optional	l)	
Save		
Settings s	aved	

Figure 21: Saved the settings

![](_page_22_Picture_0.jpeg)

Dublish	0 71	17:34 🛋 🛋 ا
/test		
		1
Count		
	Publish	
	Messane nublished	
	message published	

Figure 23: Message and topic published to control outputs

viii. On publishing the topic, popup will arrive as '*Message Published*' the device will take action on the outputs.

![](_page_23_Picture_0.jpeg)

## **DOCUMENT #:** A0011

DOCUMENT REV: A

**DOCUMENT NAME:** DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

## g. RESET THE DEVICE USING MQTT COMMAND VIA SMARTPHONE.

- i. Connect Smartphone to network hosted having MQTT Broker as said in section 13.b.
- ii. Open MyMQTT app in Smartphone.
- iii. Tap on the screen, it will open

![](_page_23_Picture_8.jpeg)

Figure 24: Taped on the default screen

- iv. Click on the Publish option.
- v. Enter topic "as you entered while configuring"
- vi. Enter Message as "Reset" (Device will get RESET)

11112 <b></b>				Ö	•	IG 📕		10:26
	ublis	h						
/test								
<u>Reset</u>								
Count								
			P	ublis	h			
<b>G</b> F	Rese	et	Re	setti	ng	Re	sets	Ŷ
q ¹ w	² e	³ r	4 t	⁵ ک	/ ⁶ ι	J ⁷	i [®] c	p p
а	s	d	f	g	h	j	k	I
순	z	x	с	۷	b	n	m	$\propto$
?123	© ,	۲					·	e

Figure 25: Clicked on the publish option and entered the message to be displayed on LCD

vii. Click on Publish button on the screen to publish the topic.

1	DOCUMENT #: A0011			
	DOCUMENT REV: A			
<b>DOCUMENT NAME:</b> DESIGN DESCRIPTION,	WIFI/BT QUAD	RELAY BOARD.		
<b>.</b>	-	51 B 4 4 8 1744	1	
	Publish	P		
	Annet			
	Alest			
	Count	1		
	Pub	lish		
	Message	published		

Figure 26: Published the message by clicking on Publish button

viii. On publishing on the topic, popup will arrive as '*Message Published*' the device will take action on the outputs.

#### h. READ DIGITAL INPUTS VIA SMARTPHONE.

- i. Connect Smartphone to network hosted having MQTT Broker as said in section 13.b.
- ii. Open MyMQTT app in Smartphone.
- iii. Tap on the screen, it will open menu window.

![](_page_24_Picture_7.jpeg)

Figure 27: Tapped on the home screen

iv. Click on the Subscribe option.

<b>1</b>	DOCUMENT #: A0011
ARMTRONIX	DOCUMENT REV: A
DOCUMENT NAME: DESIGN DESCRIPTION	ON, WIFI/BT OUAD RELAY BOARD.
DOCUMENT NAME: DESIGN DESCRIPTION	ON, WIFI/BT QUAD RELAY BOARD.

Figure 28: Clicked on the Subscribe option

i. Enter subscription topic "as you entered while configuring"

	🔞 💎 ⁴ G🖌	8:08
Control Subscribe		
Торіс		Add
/test May 25, 2018 8:08:06 PM		×

Figure 29: Entered the Subscription topic and clicked on the Add button

v. Click on back button located at left-top-corner of the screen.

![](_page_26_Figure_0.jpeg)

**DOCUMENT #:** A0011 ARMTRONIX **DOCUMENT REV:** A DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD. 13. Openhab Example Sat 11:07 🧿 Google Chrome Wifi BT Board → C ③ 192.168.1.33:8080/basicui/app?w=0005&sitemap=Ar ← 0 Wifi_BT_Board C Relay 1 C Relay 2 ( Relay 3 ( Relay 4 ? a >____ 0  $\odot$ 0000 Figure 31: Openhab image of 4 relay board a. Example of Openhab files in MQTT mode In our case,

- MQTT broker's name is "broker"
- Topic for publishing and subscription is "/test"
- Sitemap file name is Armtronix_Office.sitemap
- Item file name is Armtronix_Office.items
- Map file names are r4.map, r12.map, r13.map and r14.map

Path of sitemap file: /etc/openhab2/sitempas/Armtronix_Office.sitemap

```
Code of sitemap file:

sitemap demo label="Armtronix Office"

{

Frame

{

Group item=tsu label="Wifi_BT_Board" icon="group"

{

Switch item=GPIO4 label="Relay 1"

Switch item=GPIO12 label="Relay 2"

Switch item=GPIO13 label="Relay 3"

Switch item=GPIO14 label="Relay 4"

}

Path of item file: /etc/openhab2/items/Armtronix_Office.items
```

NIX

#### DOCUMENT #: A0011 DOCUMENT REV: A

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

Code of item file: Group All Group tsu (All)

Switch GPIO4 "Relay 1" (tsu,Lights) {mqtt=">[broker:/test:command:ON:R4_ON],>[broker:/test:command:OFF:R4_OFF],[broker:/test:state: MAP(r4.map)]",autoupdate="false"}

Switch GPIO12 "Relay 2" (tsu,Lights) {mqtt=">[broker:/test:command:ON:R12_ON],>[broker:/test:command:OFF:R12_OFF],<[broker:/test:st ate:MAP(r12.map)]",autoupdate="false"}

Switch GPIO12 "Relay 2" (tsu,Lights) {mqtt=">[broker:/test:command:ON:R12_ON],>[broker:/test:command:OFF:R12_OFF],<[broker:/test:st ate:MAP(r12.map)]",autoupdate="false"}

Switch GPIO14 "Relay 4" (tsu,Lights) {mqtt=">[broker:/test:command:ON:R14_ON],>[broker:/test:command:OFF:R14_OFF],<[broker:/test:st ate:MAP(r14.map)]",autoupdate="false"}

Path of map file: /etc/openhab2/transform/r4.map

Code of map file r4: R04isON=ON R04isOFF=OFF

Path of map file: /etc/openhab2/transform/r12.map

Code of map file r12: R12isON=ON R12isOFF=OFF

Path of map file: /etc/openhab2/transform/r13.map

Code of map file r13: R13isON=ON R13isOFF=OFF

Path of map file: /etc/openhab2/transform/r14.map

Code of map file r14: R14isON=ON R14isOFF=OFF

![](_page_29_Picture_0.jpeg)

<b>⊥</b> *	<b>DOCUMENT #:</b> A0011					
	DOCUMENT REV: A					
DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.						
4. Click on Tools Tab, move mouse pointer on "Board: xxxxxxxxxx" and click on "NodeMCU-						
32S" as shown in figure 10.						
es 🥯 Arduino IDE 🕈	Fri 20:13					
	Node_Mcu_32_Four_Relay_WiFiSwitch_with_alexa_dynamicNa					
File Edit Sketch Tools Help						
Auto Format Ctrl+1						
Node_Mcu_32_Fq Fix Ferendian & Delend	ConfigFunctions Network helperFunctions mqttFunctions se					
/* Choice						
* Date Edited : Serial Monitor Curl+Shifty						
Previous Edited WH101 HIRMWARE Updater						
#define OUTETN OA ESPassivel Device Here	<u>ـــــ</u>					
#define OUTPIN_04 ESP32 Sketch Data Upload #define OUTPIN_12						
#define OUTPIN_13 Board: "NodeMCU-32S"	Electronic SweetPeas - ESP320					
#define OUTLED 2	Nano32					
#define INPTN 0	WEMOS LOLIN32					
#define INPIN_04 Port	Dongsen Tech Pocket 32					
#define INPIN_12 Get Board InFo #define INPIN_13	"WeMos" WiFi&Bluetooth Battery					
#define INPIN_14 Programmer: "AVRISP mkII"	ESPea32					
* This sketch is set of the set o	Noduino Quantum le GPIO to switch a light/LED					
* Also it supports to change the state of the light via MQ * The puch button has to switch to ground. It has following	Node32s change.					
<ul> <li>While a WiFi config is not set or can't connect;</li> </ul>	Hornbill ESP32 Dev but more than Some-> switch					
* <u>http://server_ip</u> will give a config page with * While a WiEi config is set;	FireBeetle-ESD32					
* <u>http://server_ip/gpio</u> -> Will display the GIPIO state a						
* <u>http://server_ip/qpio?state04=x</u> -> Will change the GPI0 * http://server_ip/qpio?state12=x -> Will change the GPI0	Opehorse FSP32 Dev Module will be 0-for off 1-for on will be 0-for off 1-for on					
<pre>* <u>http://server_ip/gpio?statel3=x</u> -&gt; Will change the GPI(</pre>	Adafruit ESP32 Feather will be 0-for off 1-for on					
* <u>http://server_ip/gpio?state14=x</u> -> Will change the GPIO * <u>http://server_ip/gpio?state05=x</u> -> Will change the GPIO	NodeMCU-32S     Will be 0-for off 1-for on     will be 0-for off 1-for on					
* MQTT Commands	MH ET LIVE ESP32DevKIT					
* R12_ON , R12_OFF for on and off	MH ET LIVE ESP32MiniKit					
* R13_ON ,R13_OFF for on and off	ESP32vn IoT Uno					
* R5_ON ,R5_OFF for on and off	DOIT ESP32 DEVKIT V1					
* http://server_ip/cleareeprom -> Will reset the WiFi ser	OLIMEXESP32-EVB					
* server_ip is the IP address of the ESP8266 module, will i	OLIMEXESP32-GATEWAY					
* printed to Serial when the module is connected. (most like * To force AP config mode, press button 20 Secs!	ThaiEasyElec's ESPino32					
(+(	M5Stack-Core-ESP32					
	FSDectro32					

Figure 33: Board Selection

DOCUMENT NAME: DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.							
5. Select Unload Speed as "11	5200".						
es ⊚ Arduino IDE <del>▼</del>	5200 .			Fri 20:13			
		Node_Mcu	_32_Four_R	elay_WiFiSwitch_w	vith_alexa_dynamic		
<u>File Edit Sketch Tools Help</u>	CEduT						
Acchive Sketch	Cuttri						
Node_Mcu_32_Fd Fix Encoding & Reload		ConfigFunctions	Network	helperFunctions	mqttFunctions		
* Date Edited : Serial Monitor Edited By:Naren Serial Plotter	Ctrl+Shift+M Ctrl+Shift+L						
Previous Edited WiFi101 Firmware Updater Previously Edite ESP Exception Decoder							
#define OUTPIN_04 ESP32 Sketch Data Upload #define OUTPIN 12							
#define OUTPIN_13 Board: "NodeMCU-32S" #define OUTPIN_14 Elash Erequency: "80MHz"	۲ ۲						
#define OUTLED 2 Upload Speed: "115200"	· · ·	921600					
#define INPIN 0 #define INPIN_04	•	• 115200					
#define INPIN_12 Get Board Info #define INPIN_13		230400					
#define INPIN_14 Programmer: "AVRISP mkll" Burn Bootloader	,	460800					
<ul> <li>This sketch is tanking a way benefit of the state bush button has to switch to ground. If while a WiFi config is not set or can't contain the state of the state</li></ul>	ne light via MQT It has following nnect: with	if can't connect o T message and give functions: Normal	or for contro es back the s . press less	olling of one GPIO state after change. than 1 sec but mor	to switch a light/L& re than 50ms-> Switc}		
<pre>* While a WiFi config is set: * While a WiFi config is set:</pre>	a GIPIO state a	nd a switch form f	or it				
<pre>* http://server_ip/qpio?state04=x -&gt; Will * http://server_ip/qpio?state12=x -&gt; Will</pre>	change the GPIO	directly and disp	lay the abo	ve aswell x will be ve aswell x will be	e O-for off l-for on O-for off l-for on		
<pre>* http://server_ip/gpio?statel3=x -&gt; Will * http://server_ip/gpio?statel3=x -&gt; Will</pre>	change the GPIO	directly and disp directly and disp	lay the abo	ve aswell x will be ve aswell x will be	e O-for off 1-for on e O-for off 1-for on		
<pre>* http://server_ip/gpio/state05=x -&gt; Will * MOIT Commends</pre>	change the GPIO	directly and disp	lay the abo	ve aswell x will be	e O-for off 1-for on		
<ul> <li>R4_ON , R4_OFF for on and off</li> <li>812 ON B12 OFF for on and off</li> </ul>							
* RI3_ON , RI3_OFF for on and off * RI4_ON RI4_OFF for on and off							
* R5_0N ,R5_0FF for on and off							
http://server_ip/cleareeprom -> Will res * server in is the TP address of the ESP2266	et the WiFi set	ting and rest to c	onfigure mo	de as AP			
<ul> <li>* printed to Serial when the module is conne</li> <li>* To force AP config mode press button 20 Set</li> </ul>	ected. (most lik	ly it will be 192.	168.4.1)				
(	.0.51				11		
F	igure 34· Rai	udrate selectio	n				
•							
6. Click on tools tab, move mo	use pointer t	o "Programme	r: "Arduir	io as ISP'", unde	er this click on		
"Arduino as ISP".	uco pointort	o "Dort: "CON	v" undor	this click on "C	OMv'' + c		
<ol> <li>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).</li> </ol>							
8. Run the program. Refer to Figure 13.							
	-						

![](_page_32_Picture_0.jpeg)

# DOCUMENT #: A0011

DOCUMENT REV: A

**DOCUMENT NAME:** DESIGN DESCRIPTION, WIFI/BT QUAD RELAY BOARD.

## IMPORTANT NOTICE

ARMtronix Technologies LLP and its subsidiaries reserve the right to make corrections, enhancements, improvements and other changes to its products and services 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-----