

# Modbus - TCP Master MQTT Gateway

---

## Table of Contents

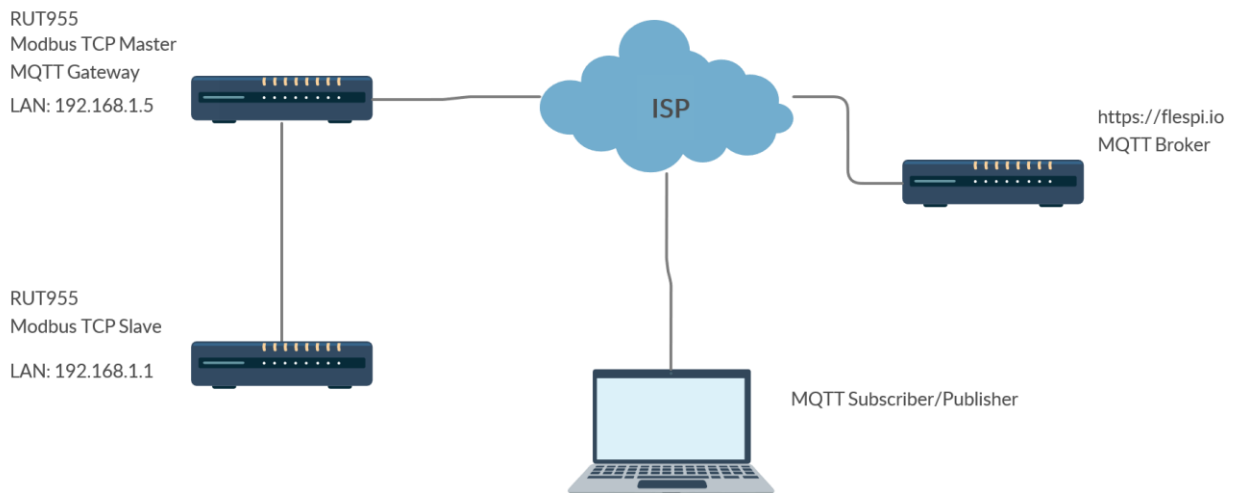
Topologies .....	2
Topology 1 .....	2
Configuring RUT955 Modbus TCP Slave .....	2
Configuring RUT955 Modbus TCP Master .....	3
Configuring Flespi.io MQTT Broker .....	4
Configuring MQTT Gateway on RUT955 Modbus TCP Master router .....	5
Message format for MQTT publisher .....	5
Testing MQTT Publisher and Subscriber on flespi.io.....	6
Topology 2 .....	7
Configuring Modbus TCP Master, MQTT Gateway MQTT Broker on RUT240 and MQTT Publisher/Subscriber on PC .....	7
Configuring RUT240 MQTT Broker .....	8
Check if MQTT Gateway is working.....	8

## Topologies

In this manual Modbus TCP master MQTT Gateway function will be configured with two different types of MQTT Brokers. First using third party MQTT Broker services (in this example Flespi.io). Second using RUT240 router as MQTT Broker. Two RUT955 routers will be used as Modbus TCP Master and Slave, and PC acts as MQTT Publisher and Subscriber.

### Topology 1

Using <https://flespi.io> as Broker



### Configuring RUT955 Modbus TCP Slave

Go to **Services > Modbus > Modbus TCP Slave**:

1. Check **Enable**
2. Enter port that will be used, for this example default port 502 will be used.
3. Enter device ID
4. Press **Save**

*\*In this config LAN port is used so "Allow Remote Access" is not needed*  
 Modbus TCP Slave RUT955 is now configured.

## Configuring RUT955 Modbus TCP Master

Go to **Services > Modbus > Modbus TCP Master**

1. Enter name for Slave device
2. Slave ID **must match** ID that was chosen in previously configured Slave device
3. Enter IP address of Modbus TCP Slave device
4. Choose same Port as in Slave device
5. Press **Add**

**New slave device**

Name <span style="color: blue;">1</span>	Slave ID <span style="color: blue;">2</span>	IP address <span style="color: blue;">3</span>	Port <span style="color: blue;">4</span>	
<input type="text" value="TestSlave"/>	<input type="text" value="1"/>	<input type="text" value="192.168.1.1"/>	<input type="text" value="502"/>	<input style="color: blue; border: 1px solid blue;" type="button" value="Add"/>

After clicking Add Advanced device settings for new Slave device

1. Select **Enabled**
2. Enter **Period** in seconds, how often requests will be sent to Slave device
3. Press **Add** in Requests configurations
4. Choose Data type and Function, enter First register/coil/input to be read or written and Register count. You can name each individual configuration, and then select enable on configurations that you want to use. Press Test button to see if Slave device respond to requests, response like this should be shown:

```
Request sent (12 bytes)
Response:
  Unit ID:1, Message length:19, Function code:3, Data count:16
Data:
  1:[4505] 2:[0] 3:[-61] 4:[0] 5:[460] 6:[21605] 7:[27764] 8:[28526]
```

5. Click **Save** to finish configuration

**Advanced device settings**

Here you can add and configure request parameters and alarms for this TCP slave device

**Slave device configuration**

1  Enabled

Name:

Slave ID:

IP address:

Port:

2  Period

Timeout:

Name	Data type	Function	First register	Register count / Values	Enabled	
<span style="color: blue;">4</span> Unnamed	16bit INT, high byte	Read holding registers (3)	<input type="text" value="3"/>	<input type="text" value="8"/>	<input checked="" type="checkbox"/>	<input type="button" value="Test"/> <input type="button" value="Delete"/>
Unnamed	16bit INT, high byte	Read holding registers (3)	<input type="text" value="4"/>	<input type="text" value="5"/>	<input checked="" type="checkbox"/>	<input type="button" value="Test"/> <input type="button" value="Delete"/>

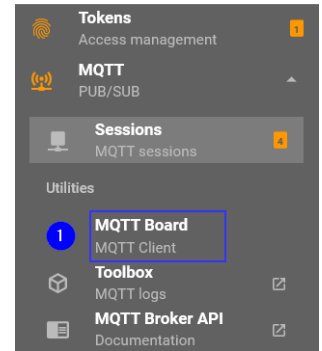
3

5

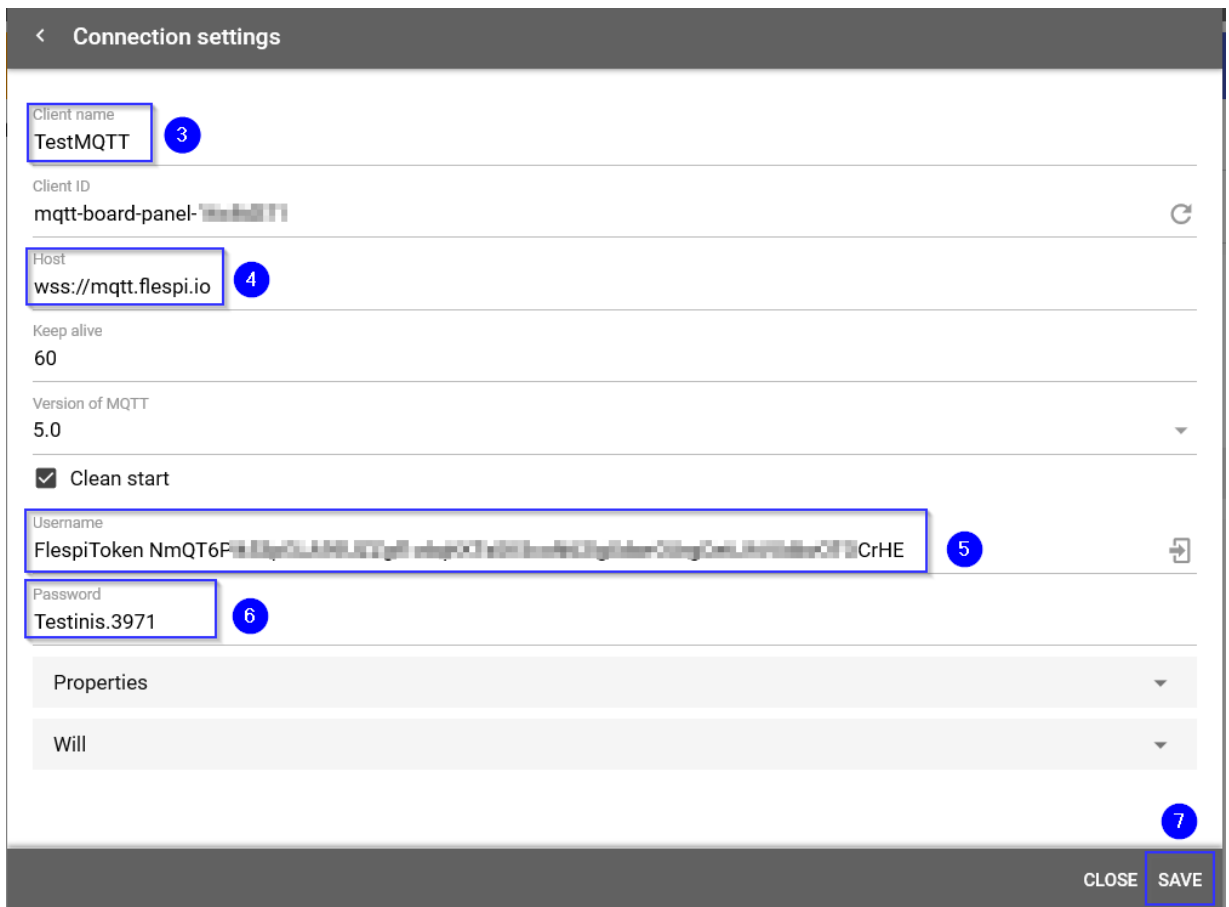
## Configuring Flespi.io MQTT Broker

Get username and password, open <https://flespi.io> Log in or create an account, then:

1. Navigate to MQTT Board on the left side of the screen
2. Press cogwheel in the top right, above Publisher



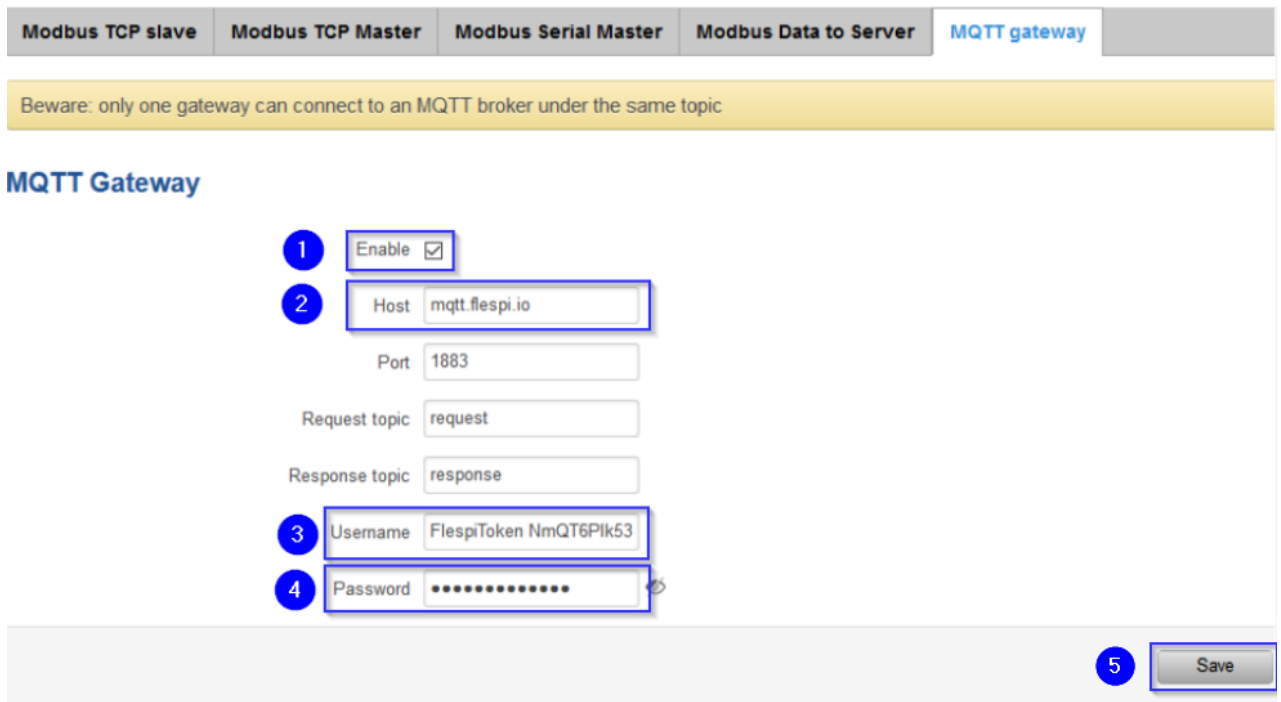
3. Enter Client name
4. Copy Host address
5. Copy the Token (it will be username for configuration)
6. Enter Password
7. Press Save



## Configuring MQTT Gateway on RUT955 Modbus TCP Master router

On RUT955 Modbus TCP Master go to **Services > Modbus > MQTT Gateway**

1. Select **Enable**
2. Enter **Hostname** (copied from flespi settings without 'wss://')
3. Enter **Username** (flespi token)
4. Enter **Password**



You can change Request and Response topics that you will have to publish and subscribe to get information from Modbus TCP Master through MQTT Gateway, but for this example they are left on default topics

## Message format for MQTT publisher

Format in text. Heavier slower, but easier to edit.

Format version (Must be 0)

- Cookie (number from 0 to  $2^{64}-1$ )
- IP type
  - 0 means IPv4
  - 1 means IPv6
  - 2 means hostname
- IP
  - IPv6 must be in full format (example.: 2001:0db8:0000:0000:0000:8a2e:0370:7334)
- Port
- Timeout (how long will be waiting for response). In seconds. from 1 to 999.
- Slave ID – indicates to which slave request is sent. From 1 to 255.
- Function.
  - 3 means read registers
  - 6 means write single register
  - 16 means write multiple registers

- **Number** of first register - from 1 to 65536, from which will information will be read or written
- If function is 3
  - Registry number (from 1 to 123, but number of first register. + number can't go out of range)
- If function is 6
  - Registry value (from 0 to 65535)
- If function is 16
  - Registry number (from 0 to 123, but number of first register. + number can't go out of range)
  - Registry values separated by commas without spaces. Example.: 1, 2, 3, 654, 21, 789. There has to be as many values, as specified number of registers, and each value must be between 0 and 65535.
    - If number of registries is 0, there should be no registry values

### Examples

Setting relay (on). Because relay **address** is 202, it means number will be 203.

```
0 65432 0 192.168.1.1 502 5 1 6 203 1
```

65432 OK

Setting relay (off).

```
0 65432 0 192.168.1.1 502 5 1 16 203 1 0
```

65432 OK

Getting uptime

```
0 65432 0 192.168.1.1 502 5 1 3 2 2
```

65432 OK 0 5590

Examples are with modbus already set up in other case you will get reply: 65432 ERROR: Failed to open socket

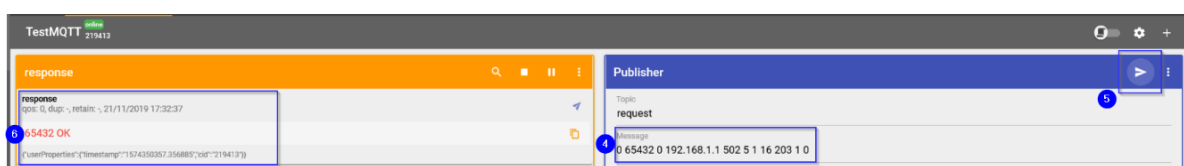
### Testing MQTT Publisher and Subscriber on flespi.io

Navigate to MQTT Board in <https://flespi.io>

1. Enter response topic in subscriber side (in this example **response**)
2. Enter request topic in Publisher side (in this example **request**)
3. Press Subscribe



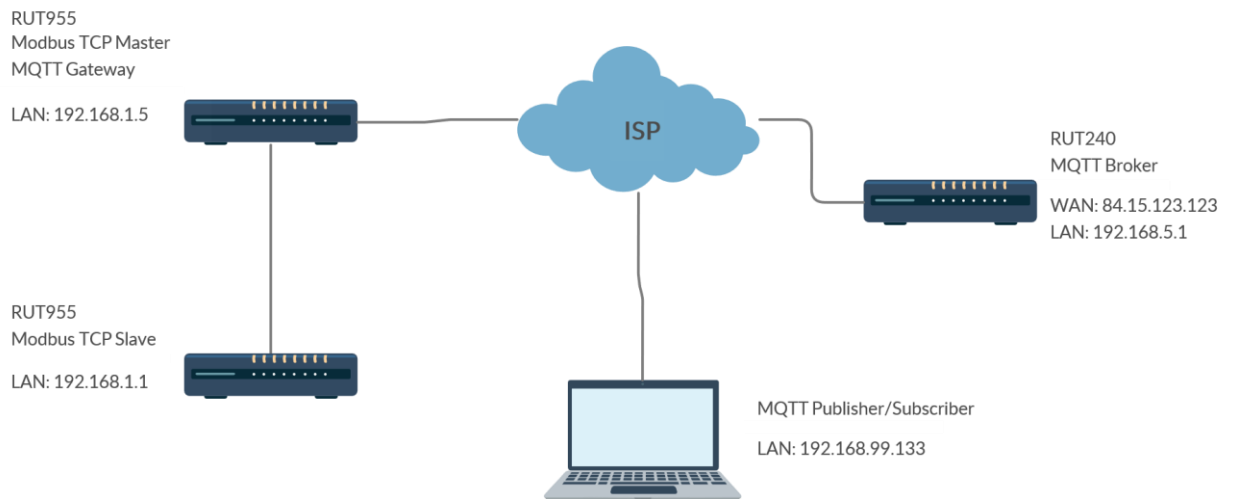
4. Enter message in Publisher side
5. Press **Publish**
6. You should receive answer in Subscriber side



*This Means that MQTT Gateway on Modbus TCP Master router is working correctly and Modbus TCP Slave receives requests.*

## Topology 2

Using RUT240 router as broker.

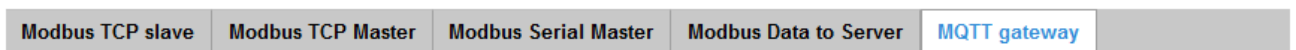


## Configuring Modbus TCP Master, MQTT Gateway MQTT Broker on RUT240 and MQTT Publisher/Subscriber on PC

Same configuration will be used for Modbus TCP Master and Slave RUT955 routers as in previous example, only settings in Modbus TCP Master router RUT955 will be changed to match MQTT Broker on RUT240 router

Go to **Services > Modbus > MQTT Gateway**

1. As Host enter Public IP of RUT240router that will host MQTT Broker
2. We are not using username this time so leave it empty (or delete if it was there from previous configuration)
3. Same with password
4. Press **Save**



Beware: only one gateway can connect to an MQTT broker under the same topic

### MQTT Gateway

Enable

1 Host

Port

Request topic

Response topic

2 Username

3 Password

4

## Configuring RUT240 MQTT Broker

Go to **Services > MQTT > Broker**

1. Select **Enable**
2. Select **Enable Remote Access**
3. Press **Save**

MQTT Broker configuration on RUT240 router is complete

## Check if MQTT Gateway is working

Open Terminal, if you are running Windows 10 you can get terminal following instruction in this link:  
<https://tutorials.ubuntu.com/tutorial/tutorial-ubuntu-on-windows#0>

Open two terminal windows

In first one (it will be your subscriber) type in: `mosquitto_sub -h [Host_address] -p [Port] -t [Topic]`

In this example: `mosquitto_sub -h 84.15.123.123 -t response`

```
root@teltonika-rut240:~# mosquitto_sub -h 84.15.123.123 -p 1883 -t response
```

In second Terminal window (this will be your publisher) type in: `mosquitto_pub -h [Host] -h [Host_address] -p [Port] -m ['Message'] -t [Topic]`

In this example: `mosquitto_pub -h 84.15.229.3 -p 1883 -m '0 65432 0 192.168.1.1 502 5 1 3 2 2' -t request`

```
root@teltonika-rut240:~# mosquitto_pub -h 84.15.123.123 -p 1883 -m '0 65432 0 192.168.1.1 502 5 1 3 2 2' -t request
```

In MQTT Subscriber side message like this should appear

```
root@teltonika-rut240:~# mosquitto_sub -h 84.15.123.123 -p 1883 -t response
65432 OK 0 1409
```

This means MQTT Gateway on Modbus TCP Master device functions properly.