

# Template:Modbus RTU Master MQTT Serial Gateway

The information on this page is updated in accordance with the [00.07.4](#) firmware version .

□

## Contents

- [1 Summary](#)
- [2 Configuration overview & prerequisites](#)
- [3 RUT2 configuration](#)
  - [3.1 Configuring Modbus RTU Slave](#)
- [4 RUT1 configuration](#)
  - [4.1 Configuring MQTT Gateway](#)
  - [4.2 Configuring Serial Gateway](#)
  - [4.3 Configuring Flespi.io MQTT Broker](#)
  - [4.4 Message format for MQTT publisher](#)
    - [4.4.1 Examples](#)
- [5 Testing MQTT Publisher and Subscriber on flespi.io](#)
  - [5.1 Adding Flespi Subscriber](#)
  - [5.2 Adding Flespi Subscriber](#)
  - [5.3 Flespi Subscriber output](#)
- [6 See Also](#)
- [7 External links](#)

## Summary

In this guide, the MQTT Serial Gateway function will be configured using third-party MQTT Broker services (in this example, *Flespi.io*).

## Configuration overview & prerequisites

- Two devices with serials ports - one acts as Modbus RTU Master, another as Modbus RTU Slave;
- Flespi.io account to act as an MQTT Broker/Publisher/Subscriber (for first configuration example);

✘

RUT2 will act as a Modbus RTU slave and RUT1 as a Modbus RTU Master. On RUT1, MQTT Serial Gateway will be configured to transfer Modbus data over MQTT. Flespi.io platform will serve as an MQTT Broker

# RUT2 configuration

## Configuring Modbus RTU Slave

---

Go to Services → Modbus → Modbus RTU Slave and create a new instance.

- Enter the **desired instance name**;
- Select the **desired serial interface**.



# RUT1 configuration

## Configuring MQTT Gateway

---

Go to **Services** → **Modbus** → **MQTT Gateway** and there:

1. **Enable** the **instance**;
2. **Enter Host** (copied from flespi connection settings without 'wss://' and port);
3. **Enter Username** (Copied from flespi Connection settings generated **token**);
4. **Enter Password**.



**Note:** *Everything else can be left as default or changed according to your needs.*

## Configuring Serial Gateway

---

Under the MQTT Gateway configuration, create the Serial Gateway:

1. Enter the **desired device ID**;
2. Select the **desired serial interface**.



## Configuring Flespi.io MQTT Broker

---

**Log in** or **create an account** on <https://flespi.io>;

1. Navigate to **MQTT Board** on the **left side** menu;
2. On the right-hand panel, top right corner, next to the name of the MQTT board, **press the cogwheel-looking icon** to open *Connection Settings*;
3. In the opened window, press "**Get flespi token**" to generate a username;
4. Enter the **Client name**;
5. Copy the Host address;

6. Copy **Username**;
7. Create a **password**.

Once done, save all the changes.



## Message format for MQTT publisher

---

Modbus request data sent in the MQTT payload should be generated in accordance with the following format:

```
1 <COOKIE> <SERIAL_DEVICE_ID> <TIMEOUT> <SLAVE_ID> <MODBUS_FUNCTION>
<FIRST_REGISTER> <REGISTER_COUNT>
```

The table below explains what each option means:

1. Format version **1**
2. Cookie from **0** to **2<sup>64</sup> - 1**
3. Serial device ID a string used to identify a serial device. Must match with Device ID field in MQTT Gateway page Serial gateway configuration section
4. Timeout timeout for Modbus connection, in seconds. Range [1..999].
5. Slave ID Indicates to which slave request is sent  
Modbus task type that will be executed. Possible values are:
  - **1** - read coils;
  - **2** - read input coils;
  - **3** - read holding registers;
  - **4** - read input registers;
  - **5** - set single coil;
  - **6** - write to a single holding register;
  - **15** - set multiple coils;
  - **16** - write to multiple holding registers.
7. First register number (not address) of the first register/coil/input (in range [1..65536]) from which the registers/coils/inputs will be read/written to.
  - **1** - coil count (in range [1..2000]); must not exceed the boundary (first coil number + coil count <= 65537);
  - **2** - input count (in range [1..2000]); must not exceed the boundary (first input number + input count <= 65537);
  - **3** - holding register count (in range [0..125]); must not exceed the boundary (first register number + holding register count <= 65537);
  - **4** - input register count (in range [0..125]); must not exceed the boundary (first register number + input register count <= 65537);
  - **5** - coil value (in range [0..1]);
  - **6** - holding register value (in range [0..65535]);
  - **15** - coil count (in range [1..1968]); must not exceed the boundary (first coil number + coil count <= 65537); and coil values separated with commas, without spaces (e.g., *1,2,3,654,21,789*); there must be exactly as many values as specified (with coil count); each value must be in the range of [0..1].
8. Registry count

## Examples

---

Setting relay (on) (Relay address is 202, which means 'Number of first register will be 203) **1 1 1 1 1 6 203 1**  
Getting temperature **1 1 1 1 1 3 6 2**

Modbus parameters are held within registers. The register numbers and corresponding system values can be found [in this article](#).

## Testing MQTT Publisher and Subscriber on flespi.io

### Adding Flespi Subscriber

---

To test the Modbus Serial Gateway functionality, **log into your Flespi account** → **MQTT Board** and **add a Subscriber**:

1. Press '+' button on the top right corner
2. Select '**Subscriber**'
3. In the topic field enter '**response**'
4. Press '**Subscribe**' button



### Adding Flespi Subscriber

---

Also, you will need to **add a Publisher**:

1. Press '+' button on the top right corner
2. Select '**Publisher**'
3. In the topic field enter '**request**'
4. In the message field enter message, for this example '**Getting temperature**' is used
5. Press '**Publish**' button



### Flespi Subscriber output

---

Check the response in the '**Subscriber**' tab, you should receive a message similar to the one below.



In the output, we can see that router's **temperature** is **44 degrees Celsius**.

## See Also

- [RUT955 Monitoring via Modbus#Get Parameters](#)
- [MQTT Gateway and Modbus](#)

## External links

[Flespi.io](#)