

Modbus / MQTT Gateway

GT100-MQ-RS

User Manual

V 1.2

Rev A



SST Automation

E-mail: support@sstautomation.com
www.SSTAutomation.com



Important Information

Disclaimer

No part of this document may be reproduced without the express written permission of SST Automation. SST Automation reserves the right to make changes to the product described within this manual at any time without notice.

The examples and diagrams in this manual are included solely for illustrative purposes and are not intended as a substitute for determining the suitability of the product for specific user applications. It is the responsibility of any such user to ensure that all operations, evaluations, and testing comply with all safety requirements such as laws, rules, codes and standards applicable to the relevant fields. SST Automation shall not be responsible or liable for misuse of the information contained herein.

Copyright

Copyright © 2025 by SST Automation. All rights reserved.

Trademark



is the registered trade mark of SST Automation.

Technical Support Contact Information

www.sstautomation.com

E-mail: support@sstautomation.com

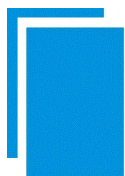
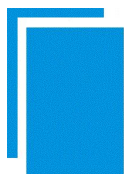


Table of Contents

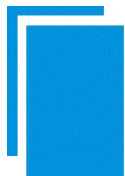
1 Product Overview	3
1.1 Product Introduction	3
1.2 Key Features	4
1.3 Technical Specifications	5
1.4 Related Products	7
1.5 Revision History	8
2 Hardware Overview	9
2.1 Product Appearance	9
2.2 LED Status Indicators	10
2.3 Configuration Button	11
2.4 Interfaces	12
2.4.1 Power Interface	12
2.4.2 RS-485 Serial Interface	12
2.4.3 Ethernet Interface	14
3 Hardware Installation	15
3.1 Mechanical Dimensions	15
3.2 Installation Method	16
4 Quick Start Guide	17
4.1 Hardware Connection	17
4.2 Software Configuration	18
5 Software Instructions	21
5.1 Software Interface Description	21
5.2 MQTT	24
5.2.1 Client	24
5.2.2 Advanced Config	26
5.2.3 Topic	28
5.2.4 Connecting to Microsoft Azure	29
5.2.5 Connecting to Amazon AWS	32
5.3 Ethernet	33
5.4 Subnet	34
5.4.1 Modbus Master Interface Configuration	36
5.4.2 Modbus Slave Interface Configuration	39
5.5 Tools	40
5.5.1 Upload Config and Download Config	41



GT100-MQ-RS Modbus / MQTT Gateway

User Manual

5.5.2 Conflict Detection	44
5.5.3 Export Excel	45
5.5.4 Communication Debug	45
5.5.5 AutoMap	47
5.6 Property Operations	48
5.6.1 Property Configuration	48
5.6.2 Name Properties in Order	49
5.6.3 Properties Name Conflict Detection	49
6 Working Principle	50
6.1 Connection Process	50
6.2 Data Exchange	50
Appendix A: MQTT Message Format	51
A.1 Default Format	51
A.1.1 Publish Message	51
A.1.2 Subscribe Message	52
A.2 Custom Message Format	53
Appendix B: Connecting to AWS IoT Application Notes	55
Appendix C: Connecting to Azure IoT via Self-Signed Certificate Method Application Notes	63
Azure CA Certificate Conversion:	73

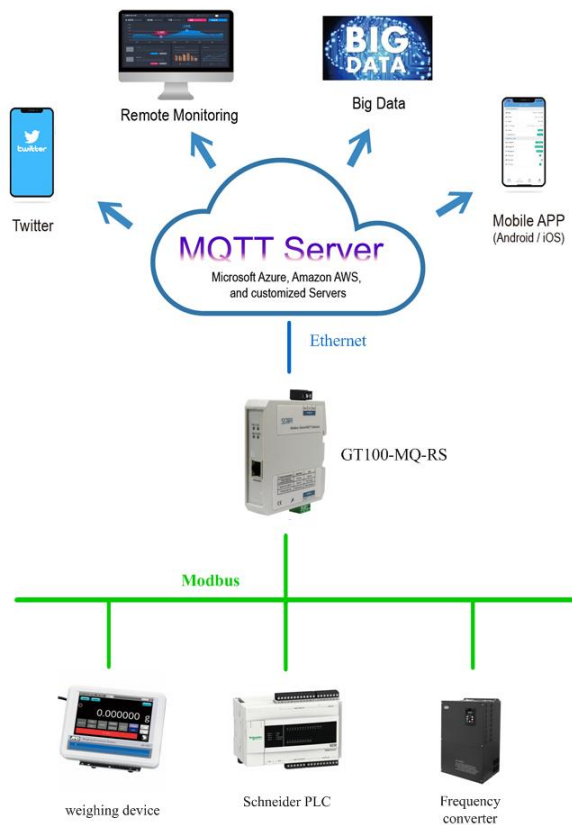


1 Product Overview

1.1 Product Introduction

The **GT100-MQ-RS** is an industrial gateway designed to provide seamlessly connectivity between Modbus RS-485 Serial (RTU/ASCII) devices and an MQTT Broker over a wired WAN connection. It can connect Modbus RTU/ASCII Master or Slave devices to the IoT Cloud via the MQTT protocol. The wired Ethernet interface enables MQTT connections to various IoT cloud platforms such as Microsoft Azure IoT, Amazon AWS IoT, Mosquitto and generic MQTT Brokers.

Note: When using authentication to connect to AWS or Azure IoT, a valid **security certificate** is required.

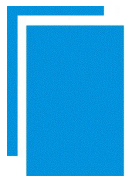




1.2 Key Features

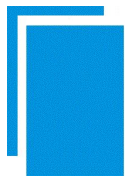
- One RS-485 Interface - Modbus Master/Slave
 - ✧ **As a Modbus Master:** Connects field devices to the IoT Cloud Platform, enabling data acquisition and device control via the Ethernet interface.
 - ✧ **As a Modbus Slave:** Integrates PLCs, DCS systems, and other Modbus master stations into an IoT Cloud Platform via the Ethernet interface.
- **Reliable Ethernet Connectivity:** Always-online Ethernet connection with automatic reconnection and connection status detection.
- Supports MQTT communication over a wired WAN network.
- **Security:** TLS V1.2 supported. Supports custom Client ID.
- **Quality of Service (QoS):** Supports QoS levels 0 and 1.
- **Debugging functionality** available for both Modbus and MQTT interfaces.
- **Timestamp publishing** supported with configurable time zone settings.
- Two MQTT publishing modes:
 1. **Change of Value Mode:** Publishes data only when values change.
 2. **Cyclic Mode:** Publishes data periodically at user-defined intervals to reduce data traffic.
- Real-time device status monitoring on the IoT Cloud Platform.
- Ethernet connection status display.
- **High reliability** for industrial applications, supporting real-time monitoring and automatic reset.
- **Remote firmware upgrade** capability.

Note: Network firewalls may interfere with the remote update process and can cause firmware updates to fail. Please ensure that the necessary network ports are open to allow for proper operation.



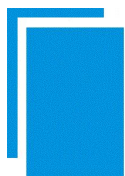
1.3 Technical Specifications

Modbus Serial Interface Features	
Serial Interface	One (1) RS-485 interface
Interface Connector	3.81 mm 3-pin terminal block
Galvanic Isolation	1kV photoelectric isolation
Baud Rates	600 bps, 1200 bps, 2400 bps, 4800 bps, 9600 bps, 19.2 kbps, 38.4 kbps, 57.6 kbps, 115.2 kbps
Parity Options	None, Odd, Even, Mark, Space
Stop Bits	1 or 2
Serial Operation Modes	Modbus RTU Master Modbus RTU Slave Modbus ASCII Master Modbus ASCII Slave
Modbus Function Codes Supported	01: Read Coils 0x 02: Read Discrete Inputs 1x 03: Read Holding Registers 4x 04: Read Input Registers 3x 05: Single Coil Write 0x 06: Single Register Write 4x 15: Multiple Coil Write 0x 16: Multiple Register Write 4x
Maximum Number of Modbus Serial Slave Devices	Without Repeater: Up to 31 Slave devices With Repeater: Up to 127 Slave devices
Maximum Input Size	Up to 512 Bytes
Maximum Output Size	Up to 512 Bytes
Maximum Number of Modbus Commands (As Master)	Up to 100 Modbus commands
Byte Swap Modes	No swap Two Byte Swap (0x1234 => 0x3412) Four Byte Swap (0x12345678 => 0x78563412)



GT100-MQ-RS Modbus / MQTT Gateway User Manual

MQTT Ethernet Interface Features	
Ethernet Interface	One (1) RJ45 10/100M full-duplex / half-duplex auto-negotiation Ethernet port.
IP Address Assignment Modes	Static IP Addressing (Manual Configuration) DHCP (Auto IP)
Ethernet Protocol	MQTT Version 3.1.1
Security	TLSv1.2 Encryption
Maximum Number of MQTT Data Tags	Up to 1000 Data Tags
Supported Data Types	16-Bit: • UINT16 • INT16 32-Bit: • UINT32 • INT32 • FLOAT • UINT32 Word Swap (0x01234567 => 0x45670123) • INT32 Word Swap (0x01234567 => 0x45670123) • FLOAT Word Swap (0x01234567 => 0x45670123) 64-Bit: • UINT64 • INT64 • DOUBLE • UINT64 Word Swap (0x0123456789ABCDEF => 0xCDEF89AB45670123) • INT64 Word Swap (0x0123456789ABCDEF => 0xCDEF89AB45670123) • DOUBLE Word Swap (0x0123456789ABCDEF => 0xCDEF89AB45670123)
Maximum Number of MQTT Topics	Up to 25 Topics



GT100-MQ-RS

Modbus / MQTT Gateway

User Manual

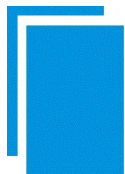
Electrical/Environmental	
Power Supply Input Voltage	24VDC (Range 11V to 30V)
Power Consumption	<1W
Operating Temperature	-4°F to 140°F (-20°C to 60°C)
Relative Humidity	5% to 95% RH (non-condensing)
Certification	CE Compliant to EMC Test Standards
Hardware	
Dimensions (W × H × D)	1.0in × 4.0in × 3.6in (25mm × 100mm × 90mm)
Mounting	35mm DIN Rail
Enclosure Protection Rating	IP20
Power Interface	One (1) 3-Pin Terminal Block

1.4 Related Products

The related products include:

- IOT600-TWX-TS – Modbus / ThingWorx Gateway
- GT200-MQ-IE – Modbus RS485/TCP to **4G** Cellular MQTT Gateway
- GT100-MQ-IE – Modbus RS485/TCP to **IoT** Cellular MQTT Gateway

To get more information about these and other related products, please visit SST Automation's website:
www.sstautomation.com.



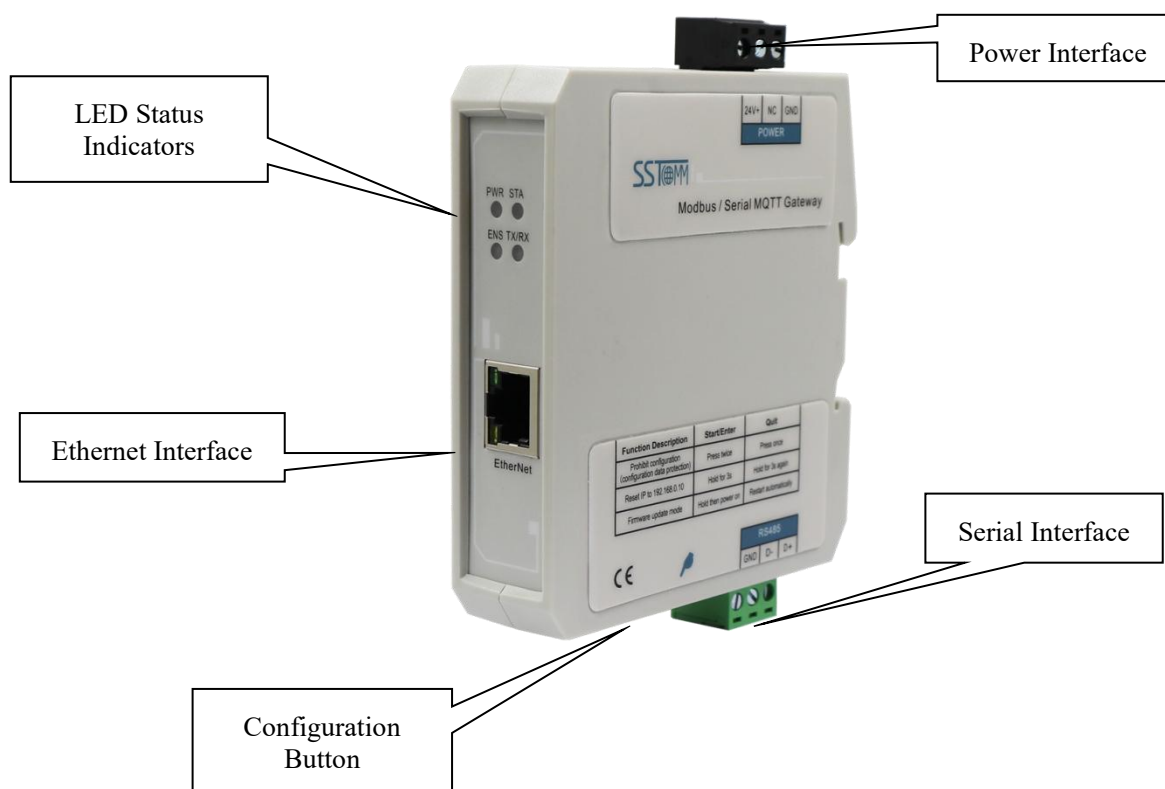
1.5 Revision History

Revision	Date	Chapter	Description
V1.0	2/16/2023	ALL	New release
V1.2	6/7/2023	MQTT Chapters 5.2.4 and 5.2.5.	Added details on how to configure Microsoft Azure and Amazon AWS.
V1.2 Rev A	11/6/2025	ALL	Updated the document format. Added diagrams for the Configuration Button in Chapter 2.3 Added Appendix B and Appendix C for application notes on connecting to AWS IoT and Azure IoT

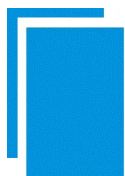


2 Hardware Overview

2.1 Product Appearance

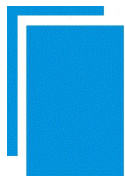


Note: This image is for reference only. The actual product may differ in appearance



2.2 LED Status Indicators

LED	State	State Description
PWR	OFF	The device is not powered on.
	Solid Green	The device is powered on.
STA	Solid Green	MQTT connection established successfully.
	Flashing Green	MQTT connection failed.
TX/RX	Flashing Red	The RS-485 interface is transmitting data.
	Flashing Green	The RS-485 interface is receiving data.
ENS	Flashing Green (Slow)	Failed to connect to the Internet.
	Flashing Green, (Fast)	Operating with a fixed IP Address: 192.168.0.10
	Flashing Red	Running the Locate function.
	Solid Green	Successfully connected to the Internet.



2.3 Configuration Button

The configuration button is located at the bottom of the GT100-MQ-RS. It is used to reset the IP address and to enable/disable configuration data protection.

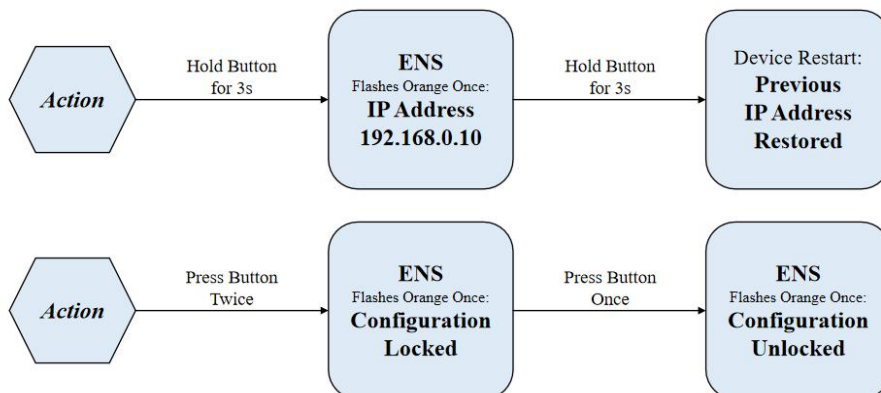
1. IP Address Reset:

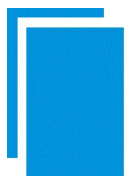
To reset the IP address, press and hold the configuration button for 3 seconds while the GT100-MQ-RS is powered on. The IP address will be reset to 192.168.0.10 and the ENS indicator will flash orange once to confirm. This function is useful when the GT100-MQ-RS cannot be detected by the configuration software. Press and hold the button again for 3 seconds to restart the device and return it to normal operation.

2. Lock/Unlock Configuration:

To toggle the configuration lock, double-press the configuration button while the gateway is powered on. The ENS indicator will flash orange once to confirm. When locked, the GT100-MQ-RS can still be discovered but cannot be configured. This function helps prevent unauthorized configuration changes. To unlock and restore normal operation, press the button once. The ENS indicator will again flash orange once to confirm.

Button Action	Description
Hold for 3 Seconds	IP Address Reset: Resets the IP address to 192.168.0.10, and the ENS indicator will flash green rapidly to confirm. Hold the button again for 3 seconds to restore the previous IP address.
Press Twice	Configuration Lock: Disables configuration upload/download. When locked, the GT100-MQ-RS can still be discovered but cannot be configured. Press the button once to unlock and restore normal operation.



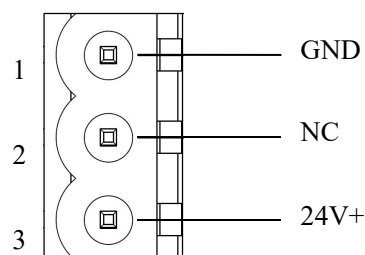


2.4 Interfaces

2.4.1 Power Interface

The GT100-MQ-RS gateway operates on a 24V DC power supply.

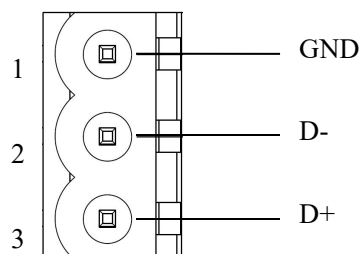
Pin	Function
1	Power GND (Negative)
2	NC (Not Connected)
3	24V+, (DC 24V Positive)



2.4.2 RS-485 Serial Interface

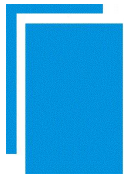
The GT100-MQ-RS gateway features an RS-485 serial interface with a 3-pin terminal connector. The pin definitions are as follows:

Pin	Function
1	GND
2	D- (RS-485 Data Negative)
3	D+ (RS-485 Data Positive)



RS-485 Serial Characteristics:

- Network topology: Linear bus with active terminal resistors installed at both ends. If the communication quality is unstable, it is recommended to add terminal resistors (120Ω, 1/2W) at both ends.
- Baud rate: 600 bps – 115.2 Kbps.
- Recommended Transmission Medium: Shielded twisted-pair cable (shielding optional depending on environmental EMC conditions).
- Number of Nodes: Up to 31 nodes per segment (without repeater), and expandable up to 127 nodes through the use of an RS485 repeater).
- Connector Type: 3-pin pluggable terminal block.

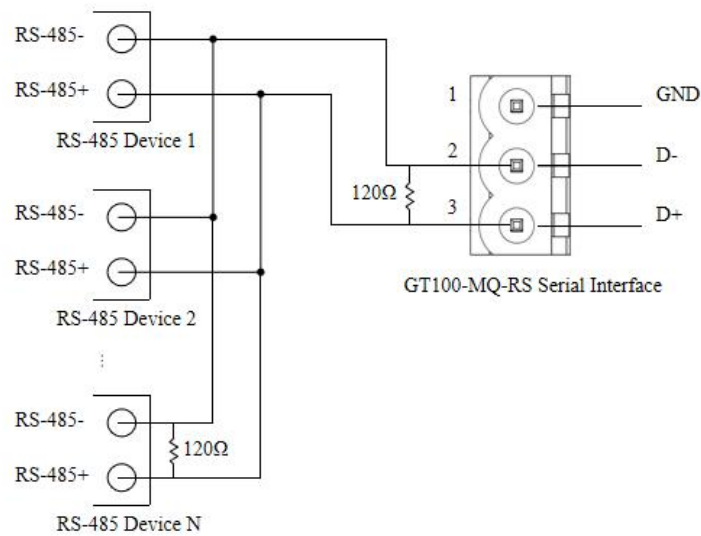


GT100-MQ-RS Modbus / MQTT Gateway User Manual

Key Points for RS-485 Equipment Installation:

- All devices must be connected to the RS485 bus.
- Each segment can connect up to 31 devices.
- Install a 120Ω 1/2W terminal resistor at each end of the bus to ensure reliable network operation.

Note: The GT100-MQ-RS gateway does **not** include an internal terminal resistor on its RS-485 interface.



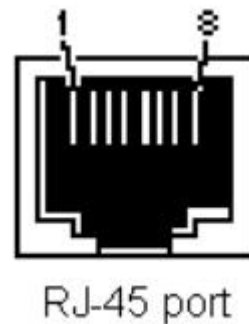


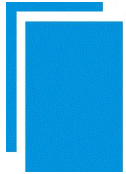
2.4.3 Ethernet Interface

The Ethernet interface is used for MQTT communication and for configuring the gateway via the SST-MQT-CFG configuration software. The Ethernet interface uses one (1) RJ45 connector and follows the IEEE 802.3u 100BASE-T standard.

The pin assignments for the Ethernet interface are as follows:

Pin	Signal Description
S1	TXD+ (Transmit Data+)
S2	TXD- (Transmit Data-)
S3	RXD+ (Receive Data+)
S4	-
S5	-
S6	RXD- (Receive Data-)
S7	-
S8	-



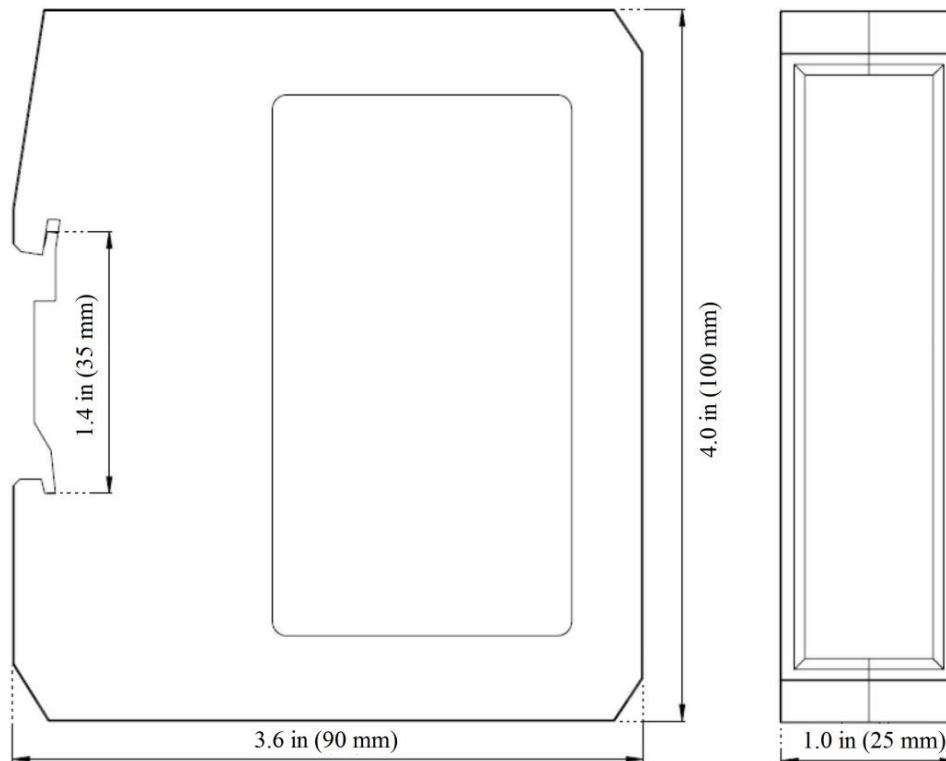


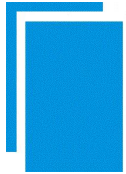
3 Hardware Installation

3.1 Mechanical Dimensions

Dimensions (Width × Height × Depth):

1.0 in × 4.0 in × 3.6 in (25 mm × 100 mm × 90 mm)

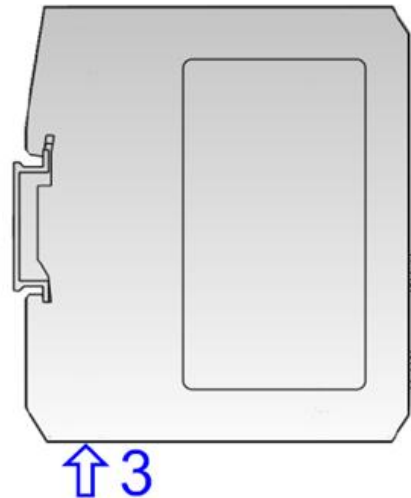
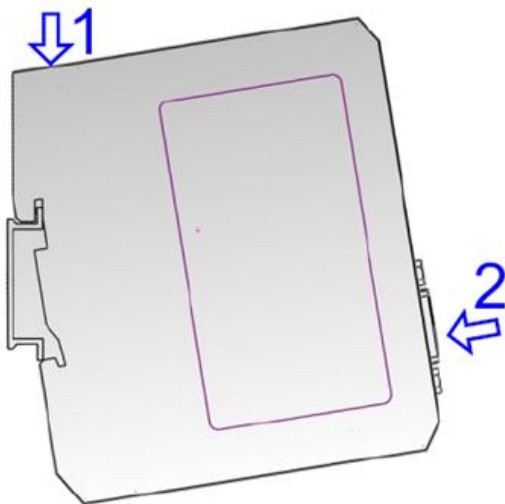




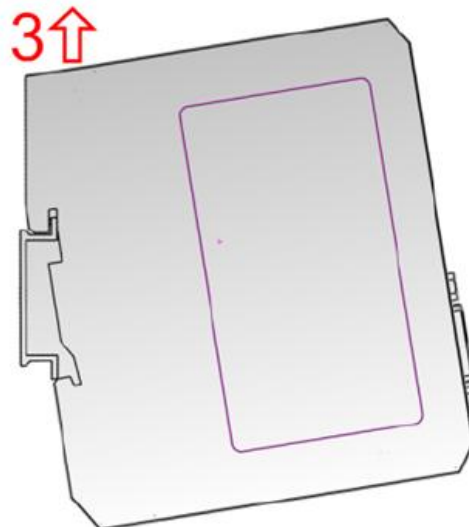
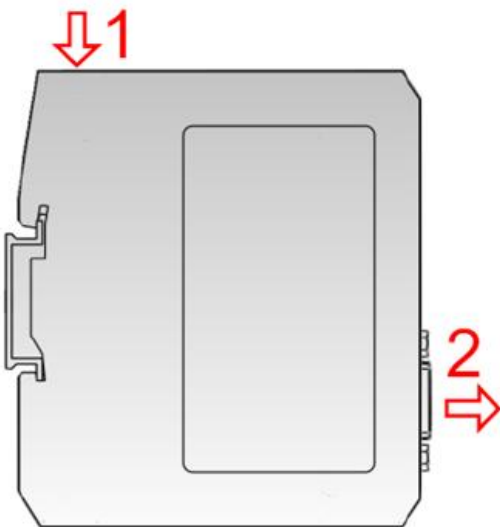
3.2 Installation Method

Mount the gateway onto a 35mm (1.4in) DIN rail. Ensure adequate ventilation and secure mounting.

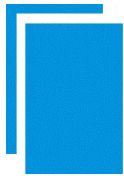
Install



Uninstall



Safety Tip: Avoid installing the device near high electromagnetic interference (EMI) sources.

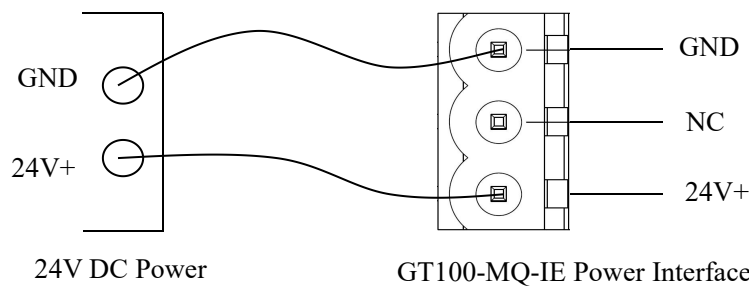


4 Quick Start Guide

4.1 Hardware Connection

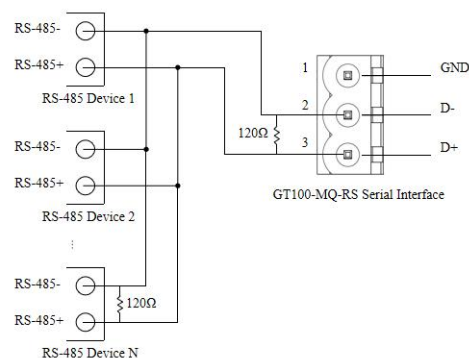
1. Connect the power supply to the GT100-MQ-RS as shown below.

Note: Do not power on the device until all connections have been completed and verified.



2. Connect the serial devices to the GT100-MQ-RS. For detailed wiring instructions, see [Chapter 2.4.2](#).

Note: The GT100-MQ-RS does not include a built-in terminal resistor on the RS-485 interface. When connecting multiple RS-485 devices, it is recommended to install terminal resistors (120Ω, 1/2W) in parallel at both ends of the communication line to ensure stable communication.



3. Connect the GT100-MQ-RS to the PC using an Ethernet cable. The GT100-MQ-RS can be connected directly to the PC or through a network router.

Note: Ensure that the GT100-MQ-RS network segment matches the PC's network segment.

4. Power on the GT100-MQ-RS.

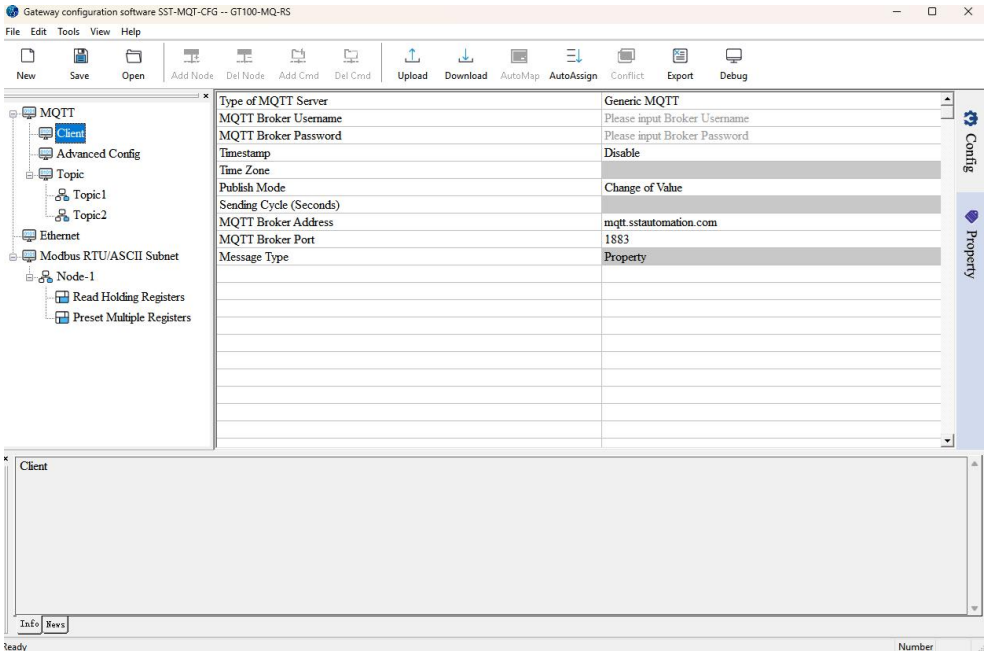
4.2 Software Configuration

Download, install, and run the configuration software SST-MQT-CFG. For more details, please see [Chapter 5](#).

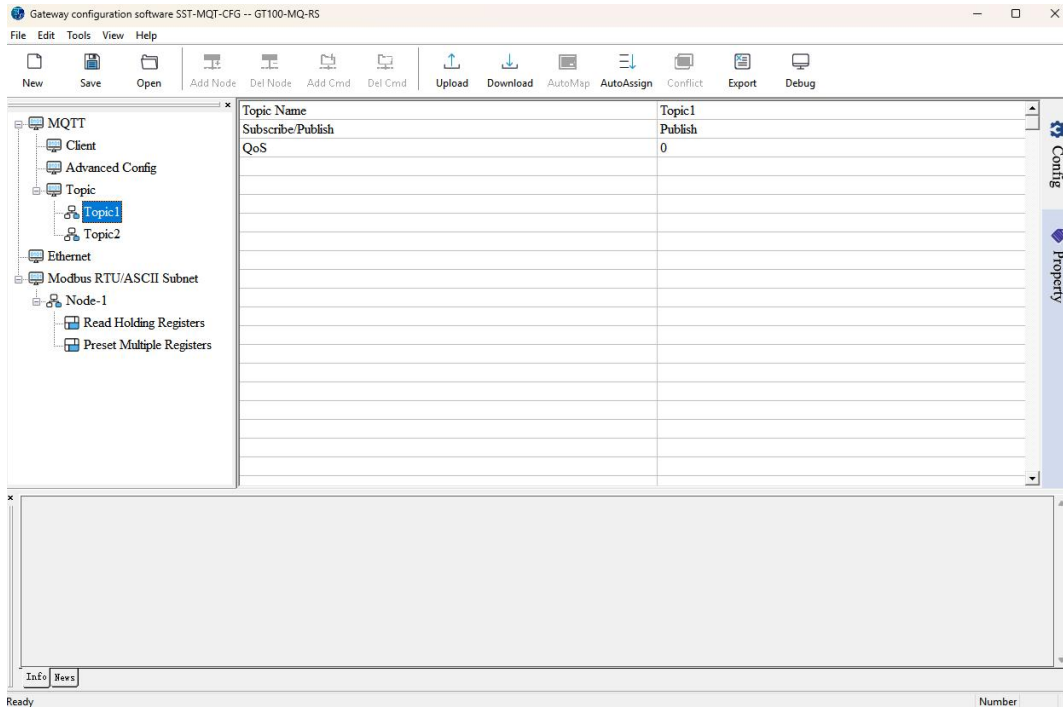
1.
- Download the configuration software SST-MQT-CFG from www.sstautomation.com and install it. Open the software and select the GT100-MQ-RS product.



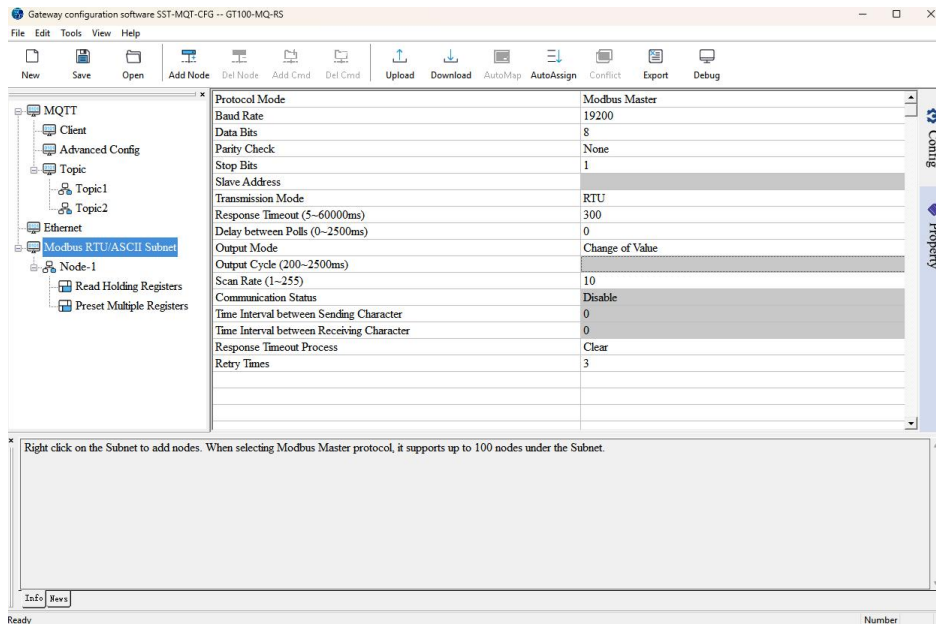
2.
- Configure the Client (MQTT Client configuration).



- 

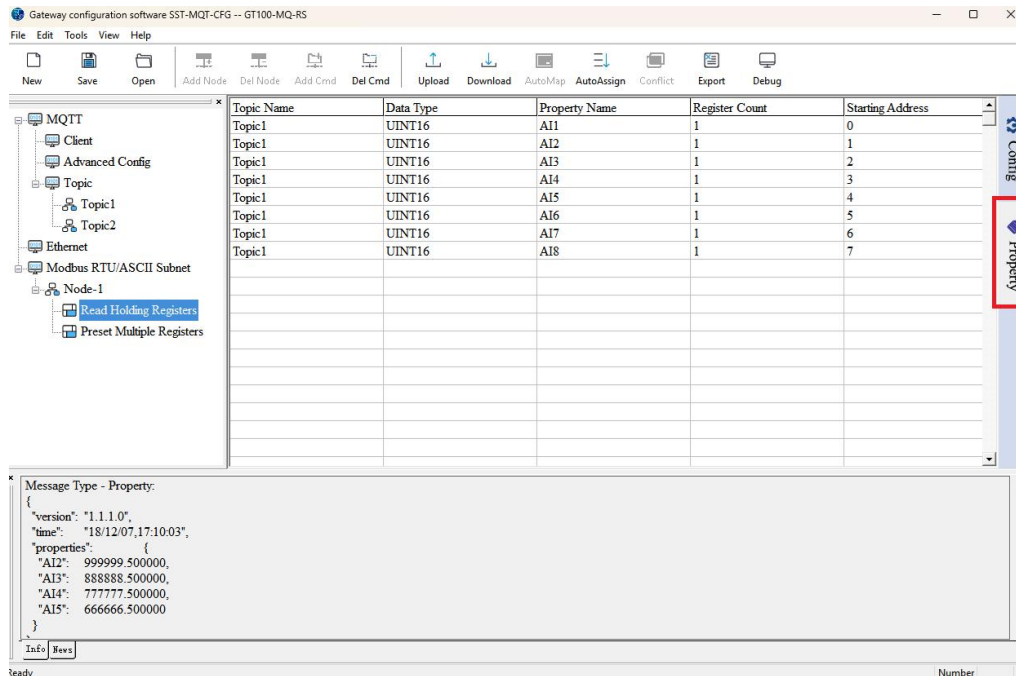


- 



GT100-MQ-RS Modbus / MQTT Gateway User Manual

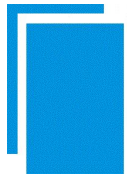
5. Configure the properties.



6. Check the mapping buffer and the property names. Use the “Auto Mapping” and “Name Properties in Order” functions.
7. Save the configuration and download to GT100-MQ-RS through network cable configuration.

Note:

1. Make sure that the GT100-MQ-RS and your computer are in the same network segment.
2. If you can’t discover any gateways, please test the network connection first. Please refer to the note “[How to Use the Ping Command](#)” located on our Support page on the sstautomation.com website.
3. If you can’t discover any gateways, press and hold the configuration button for 3 seconds after powering on. At this time, the IP address is restored to 192.168.0.10. The device can be searched under the same network segment as the gateway and computer, and the configuration can be uploaded and downloaded.



5 Software Instructions

SST-MQT-CFG is the configuration software which can be used to configure GT100-MQ-RS.

It can be downloaded from the product page or Download page (www.sstautomation.com/Download1/) of the website.

The software is based on Windows OS.

Operating System: Win7, Win10, Win11.

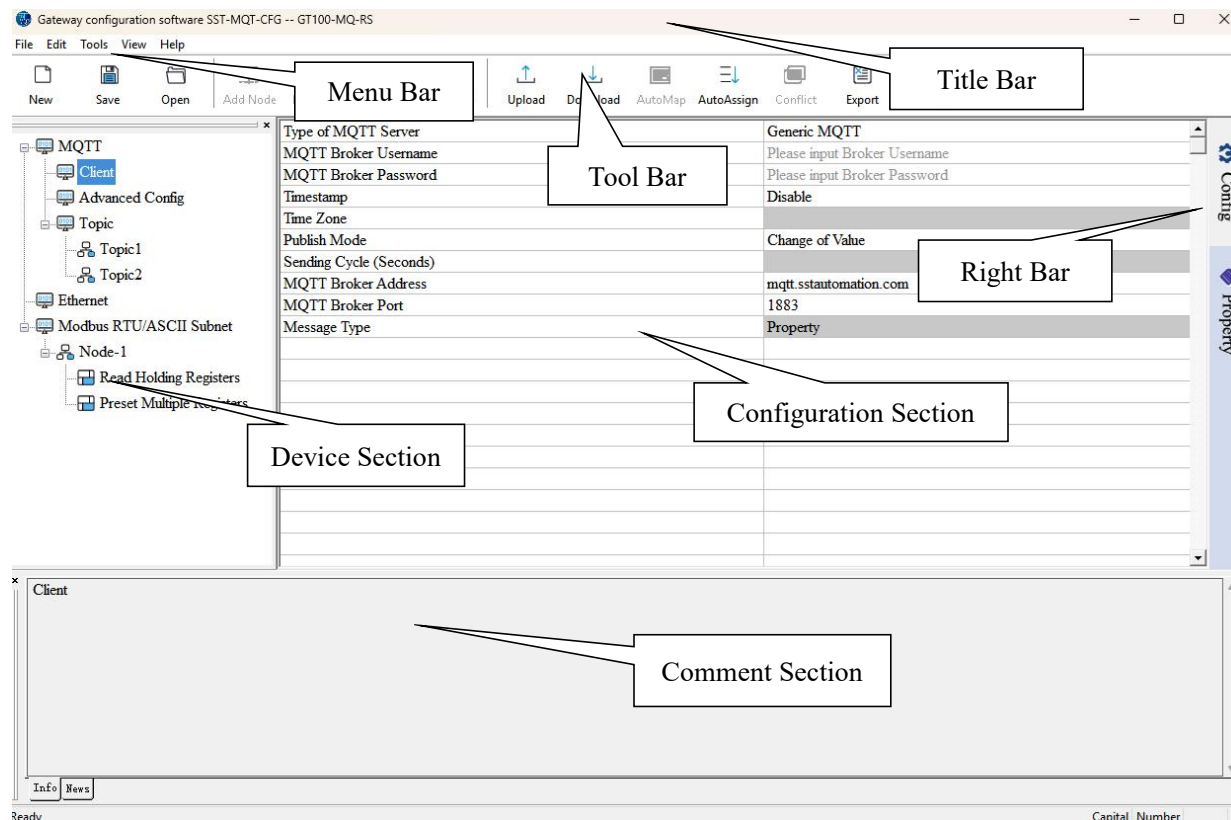
5.1 Software Interface Description

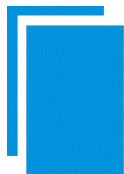
Double click on the SST-MQT-CFG icon to enter Device Select interface.



GT100-MQ-RS Modbus / MQTT Gateway User Manual

Select “GT100-MQ-RS” to enter the corresponding configuration interface:





GT100-MQ-RS

Modbus / MQTT Gateway

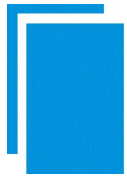
User Manual

Toolbar:



The function from left to right is: New, Save, Open, Add Node, Delete Node, AddCommand, Delete Command, Upload, Download, AutoMap, Conflict Detection, Export XLS, and Debug.

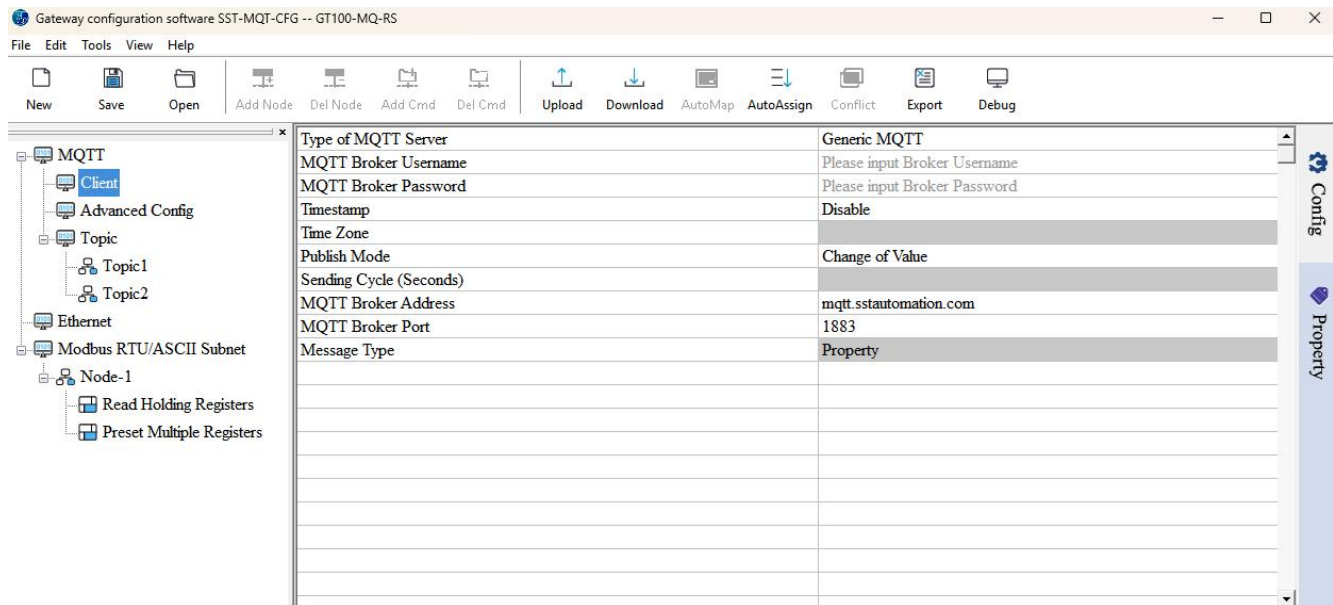
 New	New: Create a new configuration project.
 Save	Save: Save the current configuration.
 Open	Open: Open a configuration project.
 Add Node	Add Node: Add a Modbus node.
 Del Node	Delete Node: Delete a Modbus node.
 Add Cmd	Add Command: Add a Modbus command.
 Del Cmd	Delete Command: Delete a Modbus command.
 Upload	Upload: Upload the configuration from the gateway.
 Download	Download: Download the configuration to the gateway.
 AutoMap	Auto Map: Automatically calculate the mapped memory addresses for each command without conflicts.
 AutoAssign	Auto Assign: Automatically assign property names for each command without conflicts.
 Conflict	Conflict Detection: Check whether there are mapped data address conflicts with the configured commands in the gateway memory data buffer.
 Export	Export Excel: Export current configuration as a table and save it as *.xls file.
 Debug	Debug: Debug Modbus communication and view MQTT connection information.



5.2 MQTT

5.2.1 Client

Select the “Client” in the left tree view and enter the MQTT Client configuration window.



Type of MQTT Server: Supports Generic MQTT.

MQTT Broker Username: The username is used for authentication on the MQTT broker. Please refer to the corresponding MQTT server guide. The length should be less than 512bytes

MQTT Broker Password: The password is used for authentication on the MQTT broker. Please refer to the corresponding MQTT server guide. The length should be less than 512bytes

Timestamp: The time when the data is collected by the gateway, and will be published together with the data when enabled.

Time zone: Please select the local time zone to obtain the accurate time.

Publish Mode: Supports two publish mode: Change of Value and Cyclic.

Change of Value: Publish messages when value changes.

Cyclic: Publish Messages periodically.

Sending Cycle: The cycle time when publishing messages periodically. Range: 5–100000s. The default value is 60s. It's valid when the Publish Mode is “Cyclic”.

MQTT Broker Address: The URL or IP Address to connect with the MQTT broker. Please refer to the guide of corresponding MQTT Server. The length should be less than 512bytes

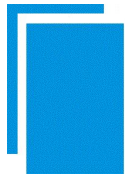
MQTT Broker Port: The access port of the MQTT broker. Please refer to the guide of corresponding MQTT Server.



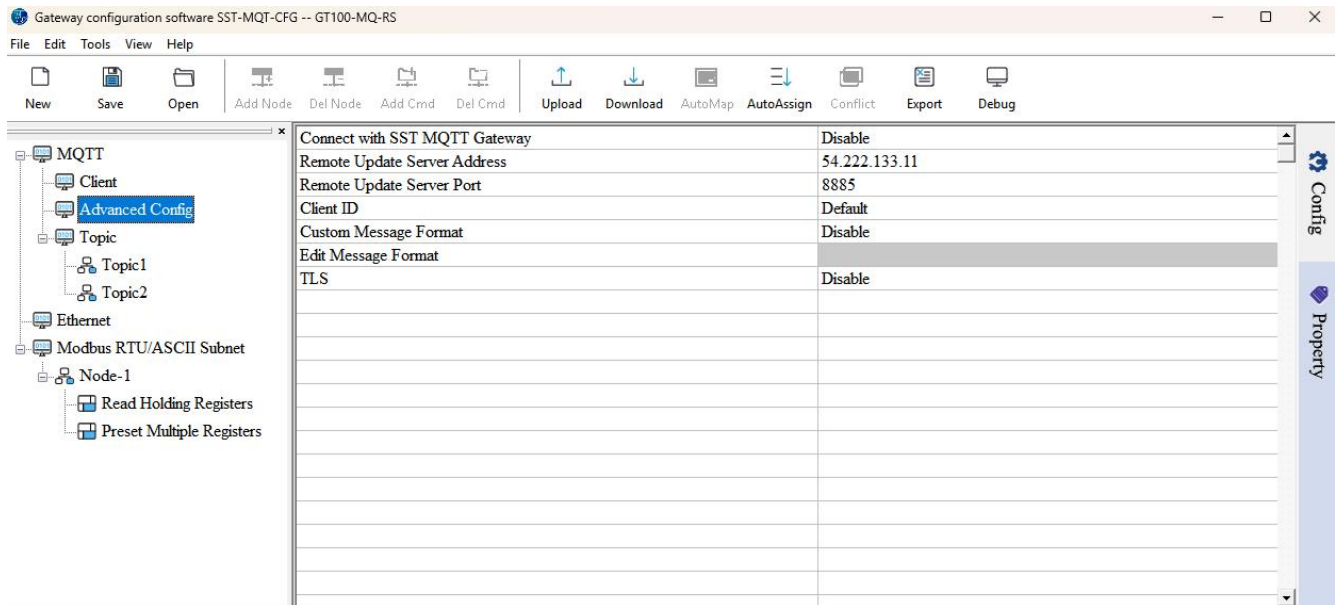
Message Type: Property type.

Property: The data is transmitted with the properties in the following message format:

```
{  
  "version":      "1.1.1.0",  
  "time":         "19/03/07,17:04:07",  
  "properties": {  
    "BI3":        false,  
    "BI4":        true,  
    "BI5":        false,  
    "BI6":        false  
  }  
}
```



5.2.2 Advanced Config



Select the “Advanced Config” in the left tree view and enter the Advanced configuration window.

Connect with SST MQTT Gateway: If enabled, the GT100-MQ-RS is able to publish or subscribe topics of other SST gateways, that allows the gateways to communicate with each other. The interconnecting gateways should connect to the same MQTT broker.

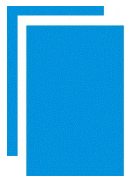
Remote Update Server Address: The default remote update service IP Address is 54.222.133.11. The gateway can be updated remotely by SST Automation. Please do not change this IP address.

Note: Remote update services may be affected by your network firewall, and firmware updates may fail.

Remote Update Server Port: The default remote update server port is 8885. The gateway can be updated remotely by SST Automation. Please do not change this port.

Client ID: The Client ID is specified for the gateway. The default ID is the serial number of the gateway. When connecting to a MQTT Server, each Client should have a unique ID. It’s recommended to use the default ID.

Custom Message Format: If enabled, the message format is set by “Edit Message Format”.



GT100-MQ-RS Modbus / MQTT Gateway User Manual

Edit Message Format:

```
{
  "timestamp": |TIMESTAMP|,
  "values": {
    |#each VALUES|
    "name": "|PROPERTIESNAME|", "v": |VALUE||#unless @last|,/unless|
  }
}
```

Buttons: Verify, Preview, OK, Cancel

For example:

```
{
  "timestamp": |TIMESTAMP|,
  "values": {
    |#each VALUES|
    "name": "|PROPERTIESNAME|", "v": |VALUE||#unless @last|,/unless|
  }
}
```

|PROPERTIESNAME| is the property name, |VALUE| is the value of the tag, and |TIMESTAMP| is the time to read the tag. Please see [Appendix A.2](#) for more details.

Click “Verify” to verify the JSON format. Click “Preview” to see a message preview.

```
{
  "timestamp": 1636005970,
  "values": {
    "name": "AI_1", "v": 11,
    "name": "AI_2", "v": 22,
    "name": "AI_3", "v": 33,
    "name": "AI_4", "v": 44,
    "name": "AI_5", "v": 55,
    "name": "AI_6", "v": 66,
    "name": "AI_7", "v": 77,
    "name": "AI_8", "v": 88
  }
}
```

Buttons: OK, Cancel

TLS: If enabled, the gateway supports the connection under TLS. Please refer to the corresponding MQTT Server guide. TLS 1.2 is supported.

CA File: Select a CA certificate and download it .(Only supported when TLS is enabled)

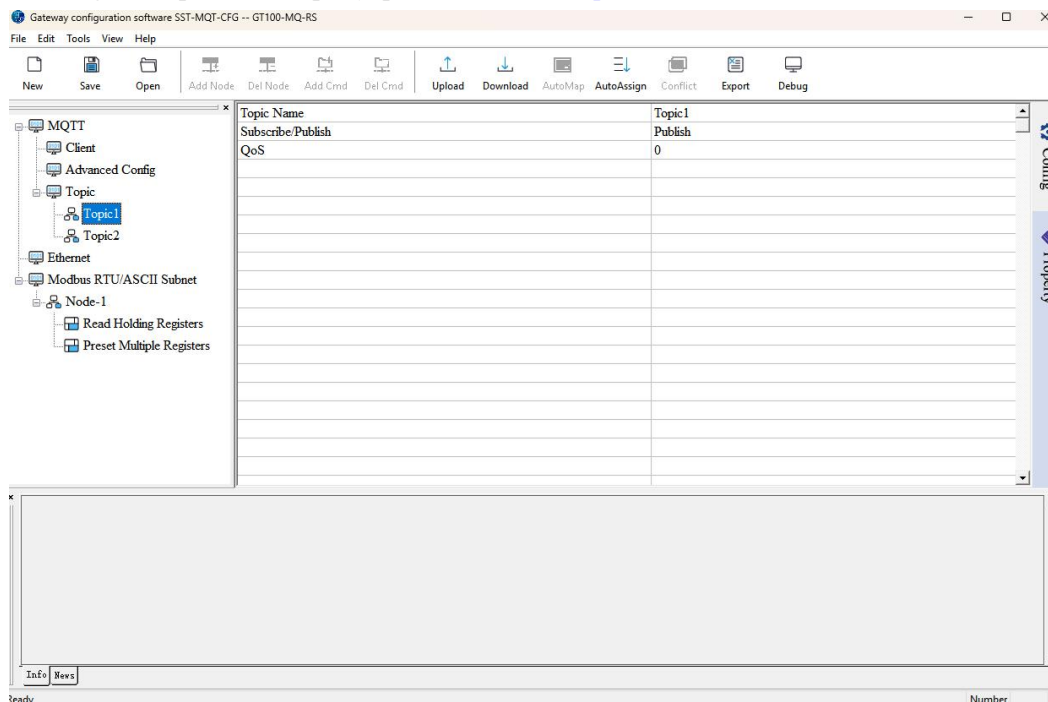
Client Key File:Select a Client Key certificate and download it.(Only supported when TLS is enabled)

5.2.3 Topic

Right click “Topic” to add a MQTT topic. If you want to delete a topic, please right-click the topic and click “Delete Topic”.

After adding a Topic, select the Topic in the left tree view to enter Topic configuration window:

Note: For the setting description of Property, please refer to [Chapter 5.6](#).



Topic Name: The topic name can be composed of characters and numbers, but does not include+and #. The length should be less than 512bytes. It supports up to 25 Topics.

Subscribe/Publish: There are three options: “Publish”, “Subscribe”, “Publish and Subscribe”.

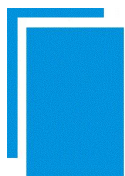
Publish: The gateway publishes the data to the MQTT Server.

Subscribe: The gateway subscribes the data from the MQTT Server.

QoS: Supports QoS 0 and QoS 1.

QoS 0: The gateway will send the data package only once, whether it is received or not. It applies to the less important data.

QoS 1: There will be an ACK to ensure that the Client or Server received the data. Data will be sent until successfully received or sent 3 times.



5.2.4 Connecting to Microsoft Azure

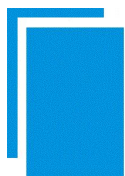
The GT100-MQ-RS supports the following device authentication methods to connect to Azure IoT:

1. Symmetric Key / Shared Access Signature (SAS) Token:

Simply configure the correct username and password, and enable TLS. A certificate file does not need to be imported. The following fields in the SST-MQT-CFG configuration software will need to match the following format to connect to Azure:

Property	Format
Client ID	<deviceID>
MQTT Broker Username	<IoT Hub Name>/<deviceID>/?api-version=2018-06-30
MQTT Broker Password	<SAS Token>
MQTT Broker Address	<IoT Hub Name>.azure-devices.net
MQTT Broker Port	8883
Publish Topic	devices/<deviceID>/messages/events/
Subscription Topic	devices/<deviceID>/messages/devicebound/#

2.



GT100-MQ-RS

Modbus / MQTT Gateway

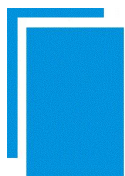
User Manual

Self-Signed X.509 Certificate:

When creating a device in Azure IoT, it is necessary to fill in the fingerprint information of the certificate along with the Device ID. In the SST-MQT-CFG configuration software, configure the correct username, enable TLS, and import the Client Certificate and Client Key. The following fields in the SST-MQT-CFG configuration software will need to match the following format to connect to Azure:

Property	Format
Client ID	<deviceID>
MQTT Broker Username	<IoT Hub Name>/<deviceID>/?api-version=2018-06-30
MQTT Broker Password	No password is required
MQTT Broker Address	<IoT Hub Name>.azure-devices.net
MQTT Broker Port	8883
CA File	No CA file is required
Client Certificate File*	<Device Certificate.pem>
Client Key File*	<Private Key.pem.key>
Publish Topic	devices/<deviceID>/messages/events/
Subscription Topic	devices/<deviceID>/messages/devicebound/#

***Note:** Certificates and keys must be in PEM format.



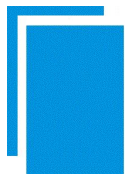
3. Signed X.509CA Certificate:

When creating a device in Azure IoT, it is necessary to register the X.509 CA certificate to your IoT Hub. Detailed instructions can be found on [Microsoft's website](#).

In the SST-MQT-CFG configuration software, configure the correct username, enable TLS, and import the correct certificates (CA Certificate, Client Certificate, and Client Key). The following fields in the SST-MQT-CFG configuration software will need to match the following format to connect to Azure:

Property	Format
Client ID	<deviceID>
MQTT Broker Username	<IoT Hub Name>/<deviceID>/?api-version=2018-06-30
MQTT Broker Password	No password is required
MQTT Broker Address	<IoT Hub Name>.azure-devices.net
MQTT Broker Port	8883
CA File*	<X.509 CA.pem>
Client Certificate File*	<Device Certificate.pem> Note: The certificate subject Common Name (CN) must match the <deviceID>.
Client Key File*	<Private Key.pem.key>
Publish Topic	devices/<deviceID>/messages/events/
Subscription Topic	devices/<deviceID>/messages/devicebound/#

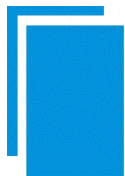
***Note:** Certificates and keys must be in PEM format.



5.2.5 Connecting to Amazon AWS

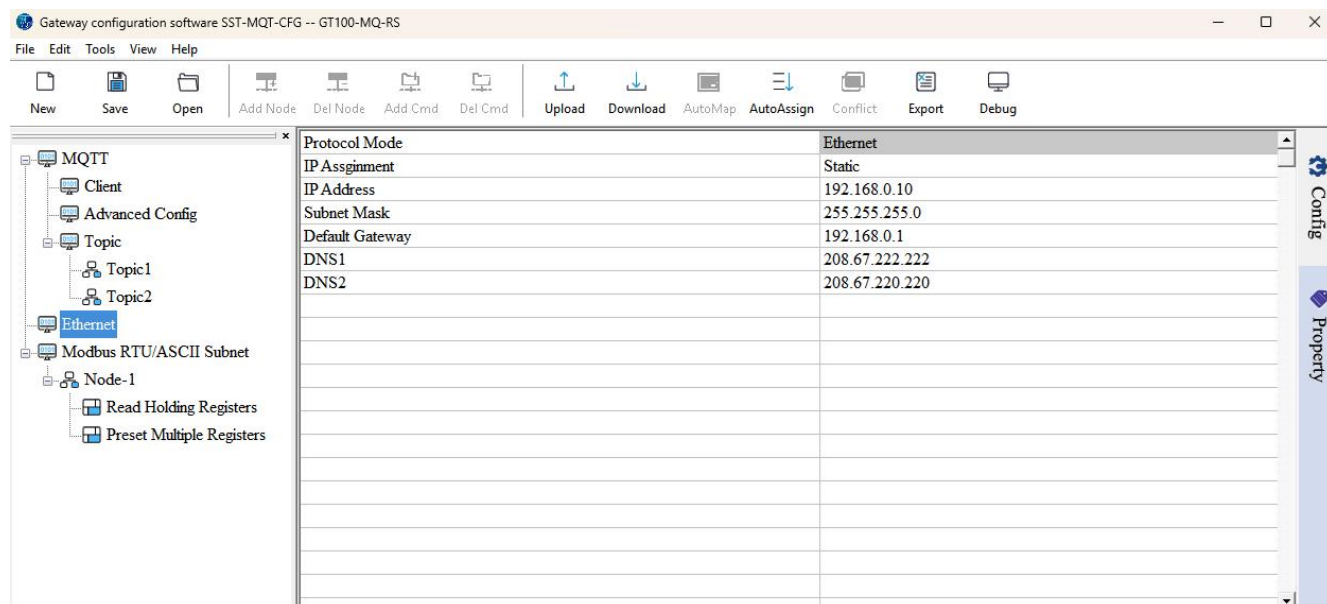
To connect the GT100-MQ-RS to AWS IoT, configure the MQTT broker address and port, enable TLS and import the correct certificates (CA Certificate, Client Certificate, and Client Key). The following fields in the SST-MQT-CFG configuration software will need to match the following format to connect to AWS:

Property	Format
MQTT Broker Username	No username is required
MQTT Broker Password	No password is required
MQTT Broker Address	<AWS Endpoint URL String>.amazonaws.com
MQTT Broker Port	8883
CA File	<Amazon Root CA.pem>
Client Certificate File	<Device Certificate.pem.crt>
Client Key File	<Private Key.pem.key>



GT100-MQ-RS Modbus / MQTT Gateway User Manual

5.3 Ethernet



Select Protocol: Ethernet.

Assign IP Mode: Supports Manually Assign / DHCP. In DHCP mode, the gateway needs to connect to a DHCP server or switch to get a valid IP address.

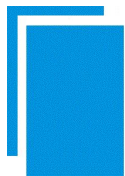
IP Address: IP Address of the GT100-MQ-RS. The device can be searched when the IP address is in the same network segment with the computer.

Note: Whether you use this protocol or not, please assign an IP address to the gateway. The configuration software uses the IP address to search for the gateway when uploading/downloading the configuration.

Subnet Mask: Subnet mask of the GT100-MQ-RS.

Default Gateway: Gateway address of the GT100-MQ-RS.

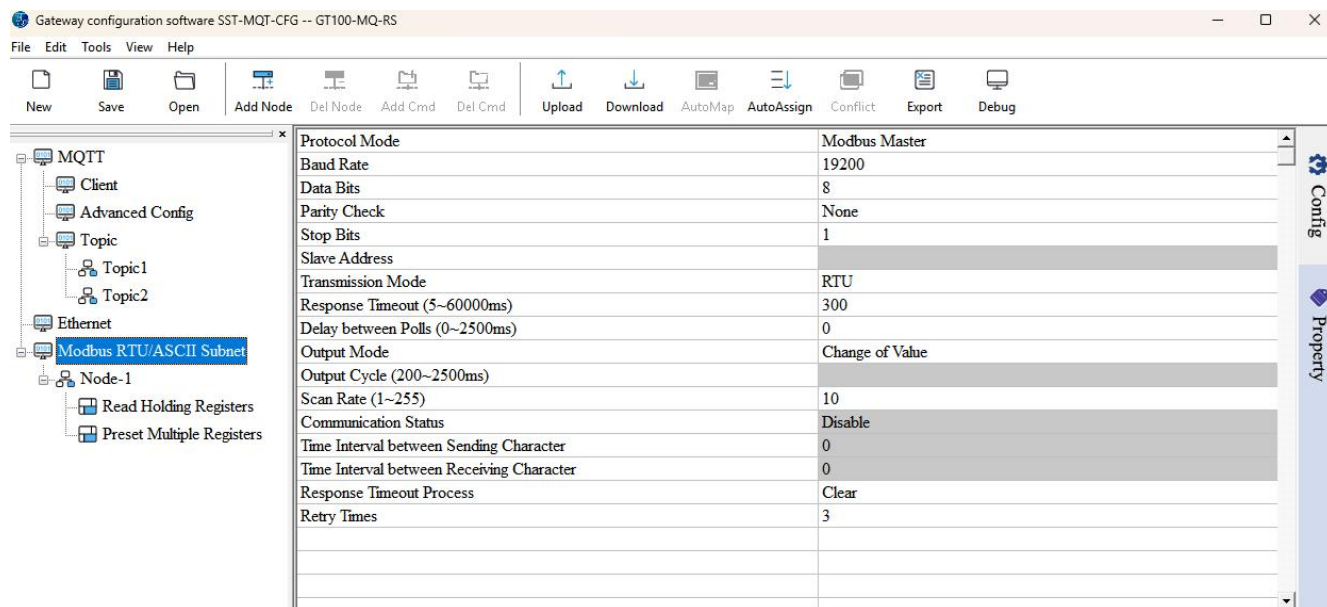
DNS1, 2: DNS (Domain Name System) is a service of the Internet. It acts as a distributed database that maps domain names and IP addresses to each other.



GT100-MQ-RS Modbus / MQTT Gateway User Manual

5.4 Subnet

Select the “Subnet” in the left tree view to enter Modbus interface configuration window.



Select Protocol: The current protocol.

Baud Rate: Baud Rate of the serial interface. Supports 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.

Data Bits: The number of data bits of each byte.

Parity Check: Supports Odd, Even, None, Mark and Space.

Stop Bits: 1 or 2.

Slave Address: It's invalid when selecting “Modbus Slave” protocol.

Transmission Mode: Modbus transmission mode. Supports RTU and ASCII.

Response Timeout: Time to wait for response from Modbus Slave after Modbus Master sends a command. Range: 5–60000ms. Invalid for “Modbus Slave” protocol.

Delay between Polls: Delay time before Modbus master send a new command after the previous command is sent and Modbus slave gives correct response or response times out. Range: 0–2500ms. The actual polling delay time is in multiples of 10 and will be rounded down from the input value. If the input value is 9, the actual polling delay time is 0ms. If the input value is 15, the actual polling delay time is 10ms. Invalid for “Modbus Slave” protocol.

Output Mode: Three options: Cyclic, Forbidden, Change of Value. Invalid for “Modbus Slave” protocol.

Cyclic: The write command will be sent periodically.

Forbidden: The write command won't be sent.

Change of Value: When the output data change, the write command will be sent.

Output Cycle: The cycle time to send write command. Range: 200–2500ms. Invalid for “Modbus Slave” protocol.



Scan Rate: The ratio of fast-scan command to slow-scan command. If the value is set to 10, a slow-scan command will be sent once every ten fast-scan commands are sent. Invalid for “Modbus Slave” protocol.

Communication Status: Unused. Modbus command status. It’s off by default and can’t be modified. The first few bytes indicate the Modbus command status. Bit0 of Byte1 represents the first command and Bit1 of Byte2 represents the tenth command.

The initial value is 0. When the communication is normal, the value is 1.

Time Interval between Sending Character: Unused. Every byte will be sent at this interval.

Time Interval between Receiving Character: Unused. The interval determines when to stop receiving a frame of data.

Response Timeout Process: When the gateway doesn’t receive the response within the time set in “Response Timeout”. “Modbus slave” protocol is not supported.

Clear: Sets the response data in the input buffer to zero.

Hold: The data in the input buffer remains the same.

Retry Times: The number of retries to send the command when response error. Range: 2–254. The default value is 3. “Modbus slave” protocol is not supported.

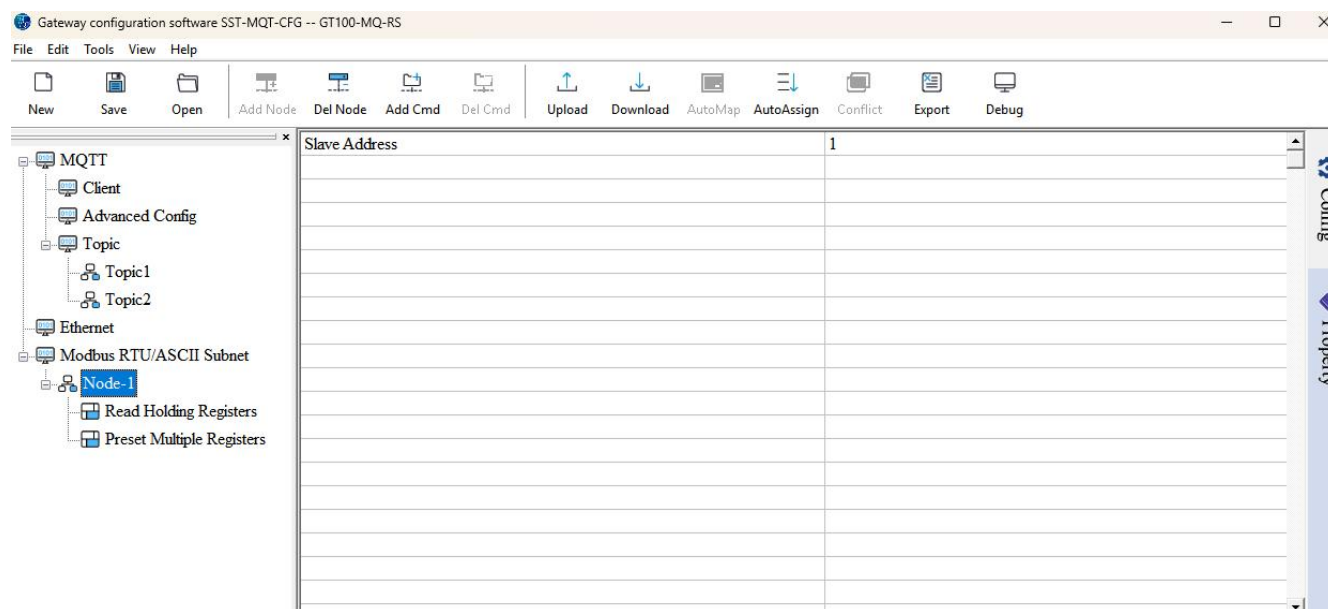
5.4.1 Modbus Master Interface Configuration

1. Node

The Configuration view supports three types of operation: Edit Menu, Edit Toolbar and Right click edit Menu.

- (1) Add node: Select the “Subnet” and add a new node. A new node named “Node-X” will be added under the Subnet. The maximum number of nodes is 100.
- (2) Delete node: When deleting a node, all commands under the node will be deleted.
- (3) Copy node: Select an existing node and copy the node. The commands under the node will be copied at the same time.
- (4) Paste node: Select the Subnet or an existing node and paste the node. The pasted node has the same parameters with the copied node.

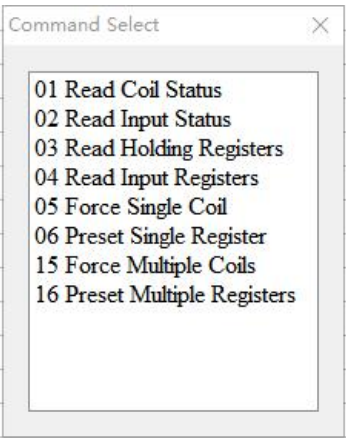
The node is the Modbus slave address, ranging from 0 to 247. The node interface is displayed as follows:



2. Modbus Command

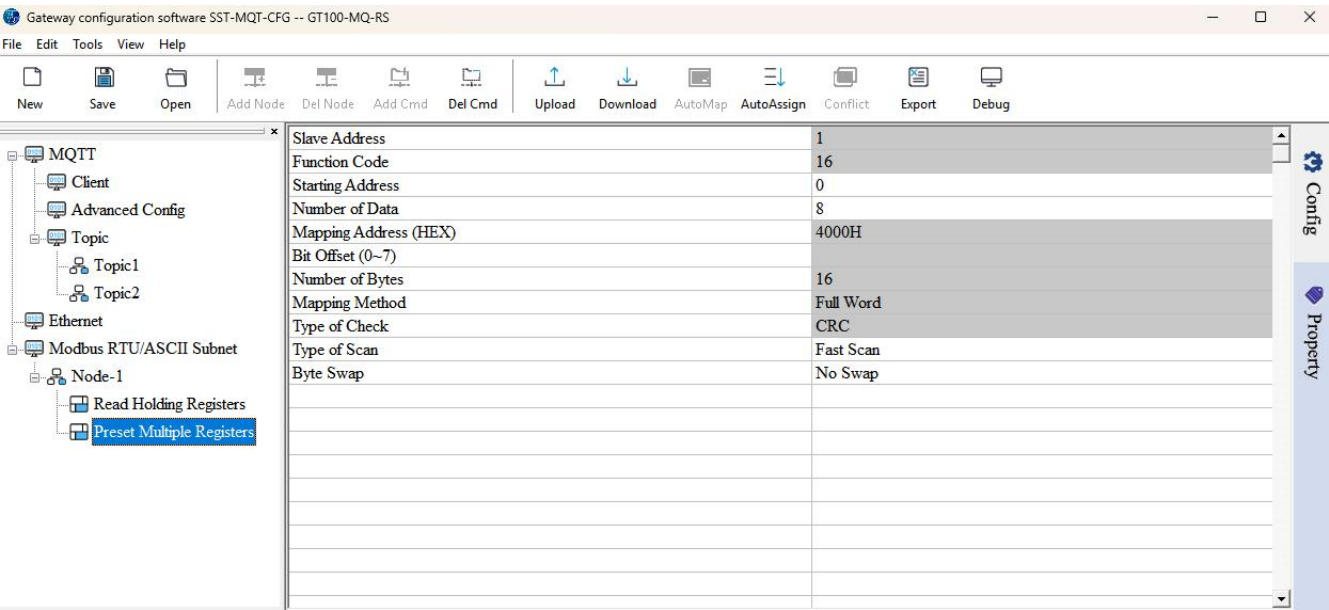
The Configuration view supports three types of operation: Edit Menu, Edit Toolbar and Right click edit Menu.

- (1) Add command: Select a node and add a command. The maximum command number is 100.
 - (2) Delete command: Select a command and then delete it.
- It supports the commands: 01, 02, 03, 04, 05, 06, 15 and 16. Double click the command to select the command.



The command interface is displayed as follows:

Note: For the setting description of Property, please refer to [Chapter 5.6](#).

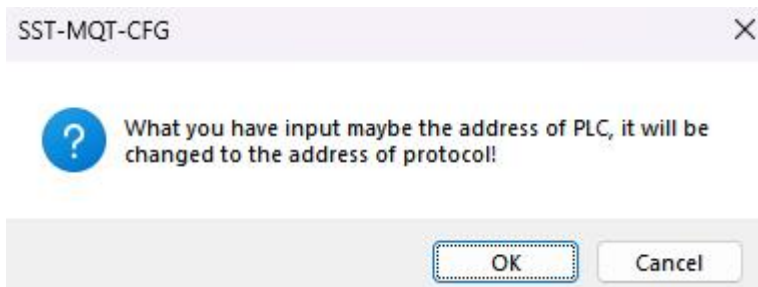


Slave Address: Modbus Slave address.

Function Code: Modbus Function Code.

Starting Address: The starting address of status or register. Range: 0–65535.

Note: This address refers to the protocol address. When the user enters the PLC address, there will be a pop-up window asking for confirmation. Click “OK” to convert the input address into a protocol address. Click “Cancel” to refuse the converting.



Examples of PLC addresses and corresponding protocol addresses.

Block	PLC Address	Protocol Address
Coil Status	00001–00010	00000–00009
Input Status	10001–10010	00000–00009
Holding Register	40001–40010	00000–00009
Input Register	30001–30010	00000–00009

Number of Data: The number of data(registers or coils or status). The range is 1–122 (Modbus function code 3,4,16) or 1–244 (Modbus function code 1,2,15).

Mapping Address (HEX): The starting address of data in the buffer.

Read command: 0x0000–0x01FF.

Write command: 0x4000–0x41FF.

When users want to exchange the data between slave devices, the write command can be mapping in 0x0000–0x01FF.

Bit Offset (0-7): When reading/writing a coil or reading input status, its value will be mapped into bits of input or output buffer.

Number of Bytes: The number of the data bytes.

Mapping Method: Unused. Support only full-word mapping now. When reading or writing a register, you can map a full word into the buffer, or only map MS/LS byte into the buffer.

Type of Check: Data communication error check.

Type of Scan: Every Modbus command can be set to fast scan or slow scan. The gateway will send Modbus command according to the Scan Rate. Scan Rate is the ratio of fast-scan command to slow-scan command.

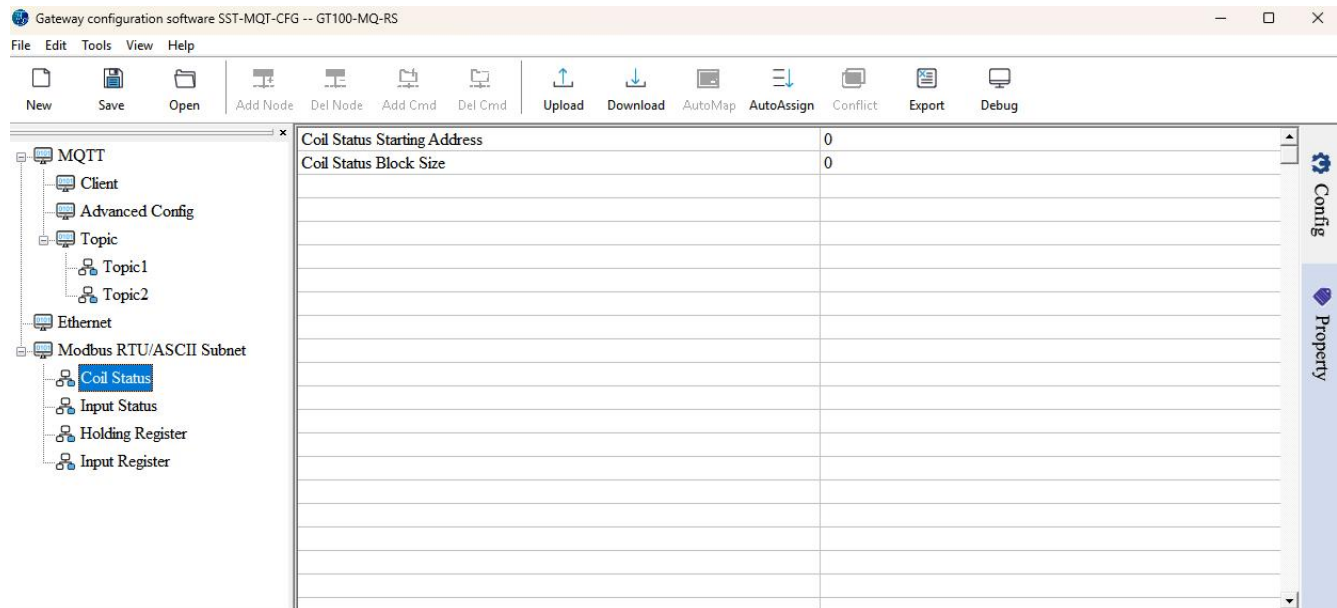
Byte Swap: There are three modes of Byte Swap: No Swap, Two Bytes Swap and Four Bytes Swap. It’s valid for registers.

Two Bytes Swap: The high and low bytes of 2 bytes (one register) are exchanged, e.g. 0x1234 to 0x3412.

Four Bytes Swap: The first two bytes are exchanged with the last two bytes, e.g. 0x12345678 to 0x78563412.

5.4.2 Modbus Slave Interface Configuration

Select a Status/Register block to enter Modbus slave block configuration window.



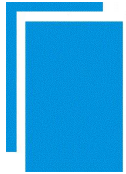
Coil/Input Status Starting Address: The start address of 0/1XXXX area. The range is 0–65535, the default is 0.

Coil/Input Status Block Size: The block size of the 0/1XXXX area. The range is 0–512, and the default is 0.

Holding/Input Register Starting Address: The start address of the 4/3XXXX area. The range is 0–65535, and the default is 0.

Holding/Input Register Block Size: The block size of the 4/3XXXX area. The range is 0–256, and the default is 0.

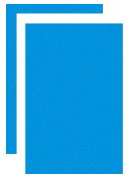
For the setting description of Property, please refer to [Chapter 5.6](#).



5.5 Tools

The “Tool” option on the menu bar contains the following functions:

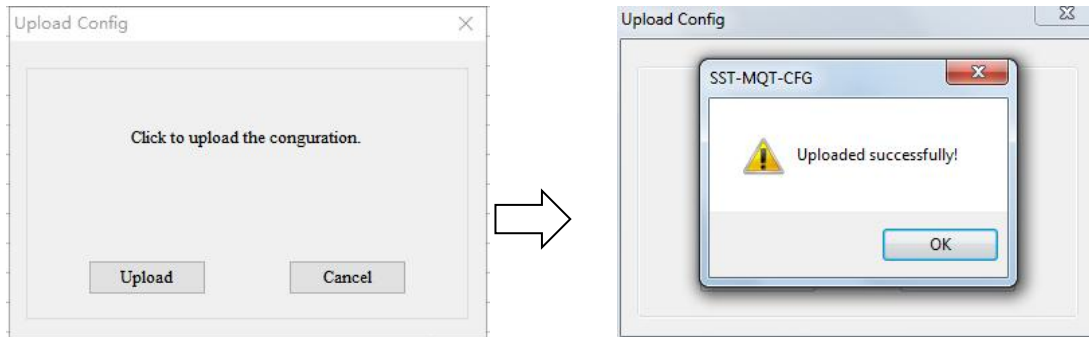
- Upload Config
- Download Config
- Conflict Detection
- Export Excel
- Communication Debug
- AutoMap
- Name Properties in Oder
- Properties Name Conflict Detection

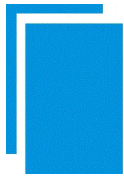


2. Upload

Select “Upload” to upload the gateway configuration information from the device to the software.

When uploading the configuration from the gateway to the software, the pop-up window will be shown as below:

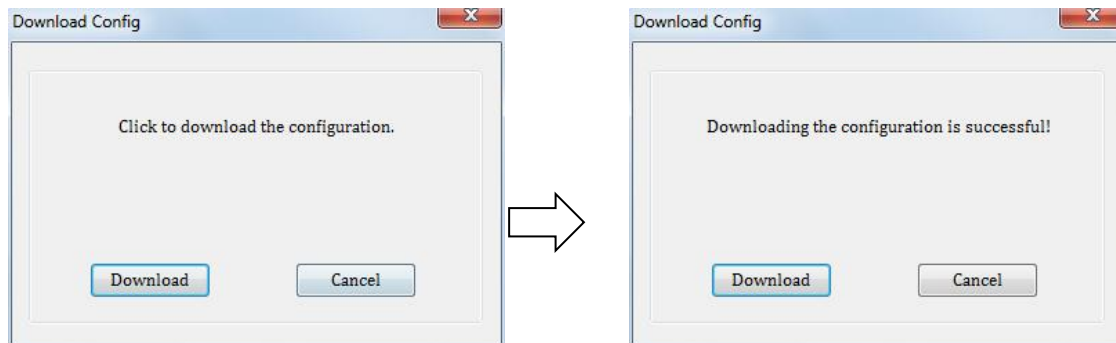




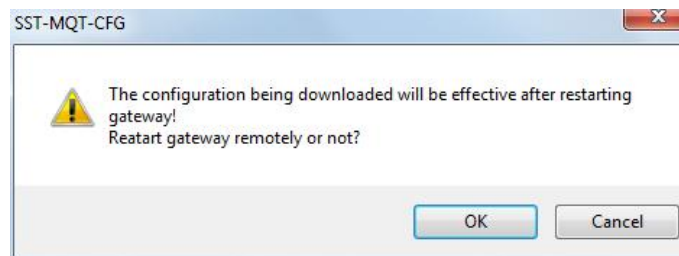
3. Download

Select “Download” to download the configured gateway information to the gateway device.

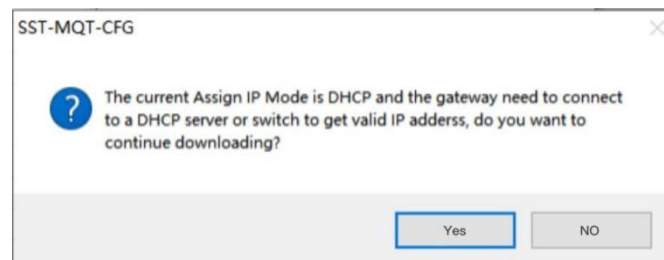
When downloading the configuration to the gateway, the pop-up window will be shown as below:



Click “Cancel” and the pop-up window will ask for remote restart. The new configuration is effective after restarting. Click “OK” to restart the gateway. Click “Cancel” to refuse to restart.



Note: If Modbus TCP Assign IP Mode is set to DHCP, the following dialog box is displayed.



5.5.2 Conflict Detection

To detect whether there are address conflicts in the mapping buffer. The configuration cannot be downloaded to the gateway if there are conflicts. This function is used for Modbus commands.

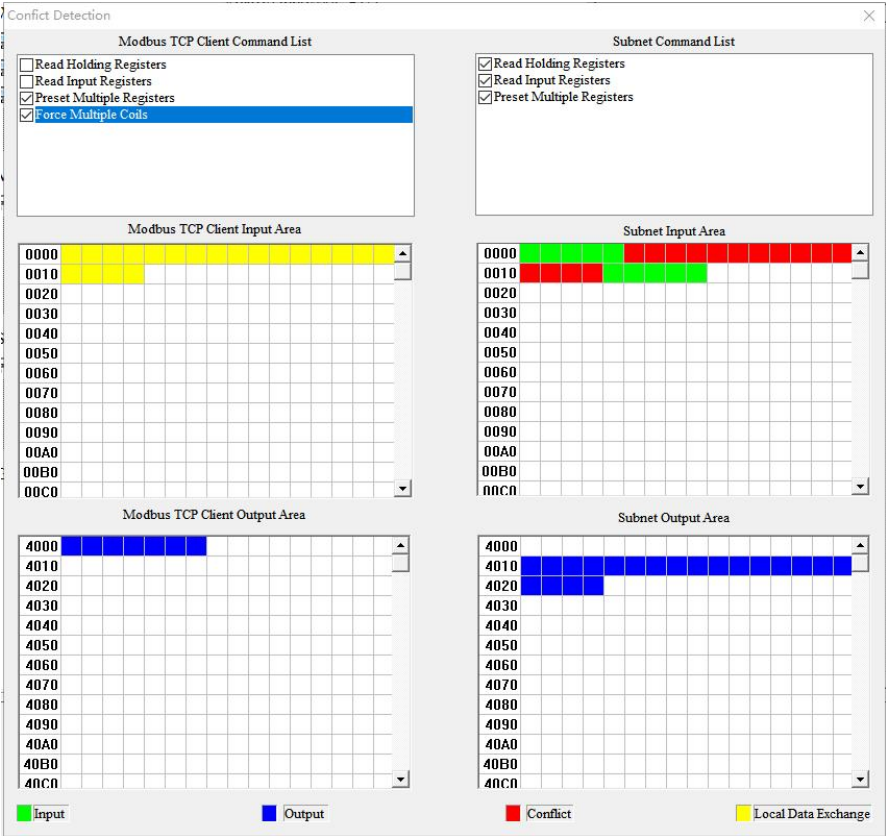
The mapping buffer is divided into input area and output area.

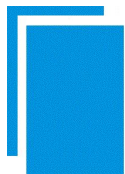
Input area address is from 0x0000 to 0x3FFF.

Output area address is from 0x4000 to 0x7FFF.

Each square represents a byte. The colors indicate the status of the address.

- Green:** No conflict of read commands in input area.
- Yellow:** The write commands are used to exchange data between slave devices and there is no conflict.
- Blue:** No conflict of write commands in output area.
- Red:** Address conflicts.
- White:** None data is mapped in this address.



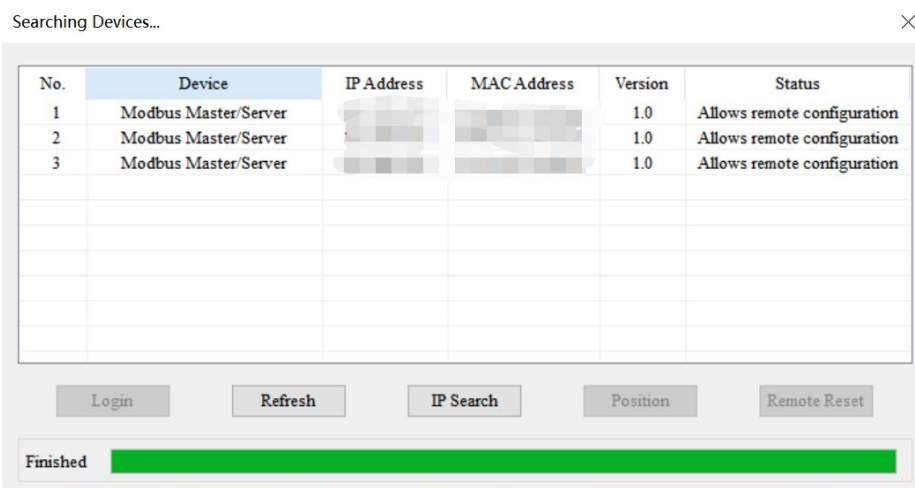
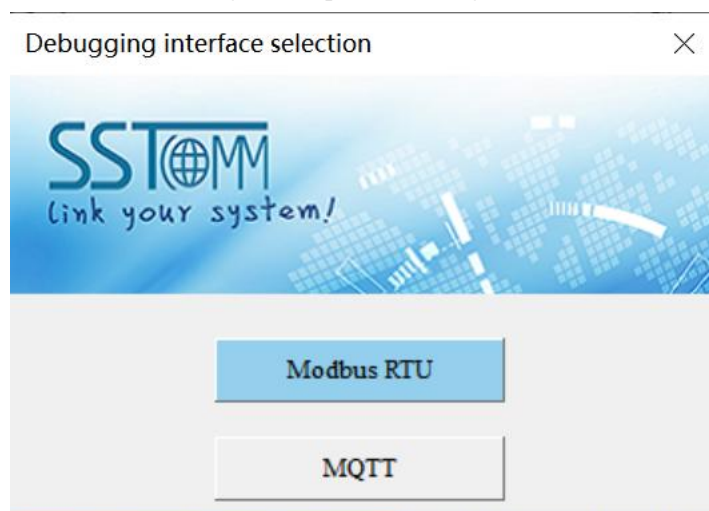


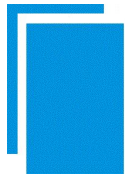
5.5.3 Export Excel

Excel documents help users examine the configuration. Save the configuration as excel document and choose the right path.

5.5.4 Communication Debug

The Communication Debug function is to Debug modbus communication and view MQTT connection information. Select “Tool” >> “Communication Debug” to open the debug window.





GT100-MQ-RS Modbus / MQTT Gateway User Manual

MQTT: Display MQTT network connection information

Communication Traffic ×

Exit Continue Clear **Save**

Index	Time Stamp	Status
5	10:34:06.415	Data published successfully
4	10:34:05.906	Data published successfully
3	10:34:05.477	Data published successfully
2	10:34:05.164	Data published successfully
1	10:34:04.849	Data published successfully

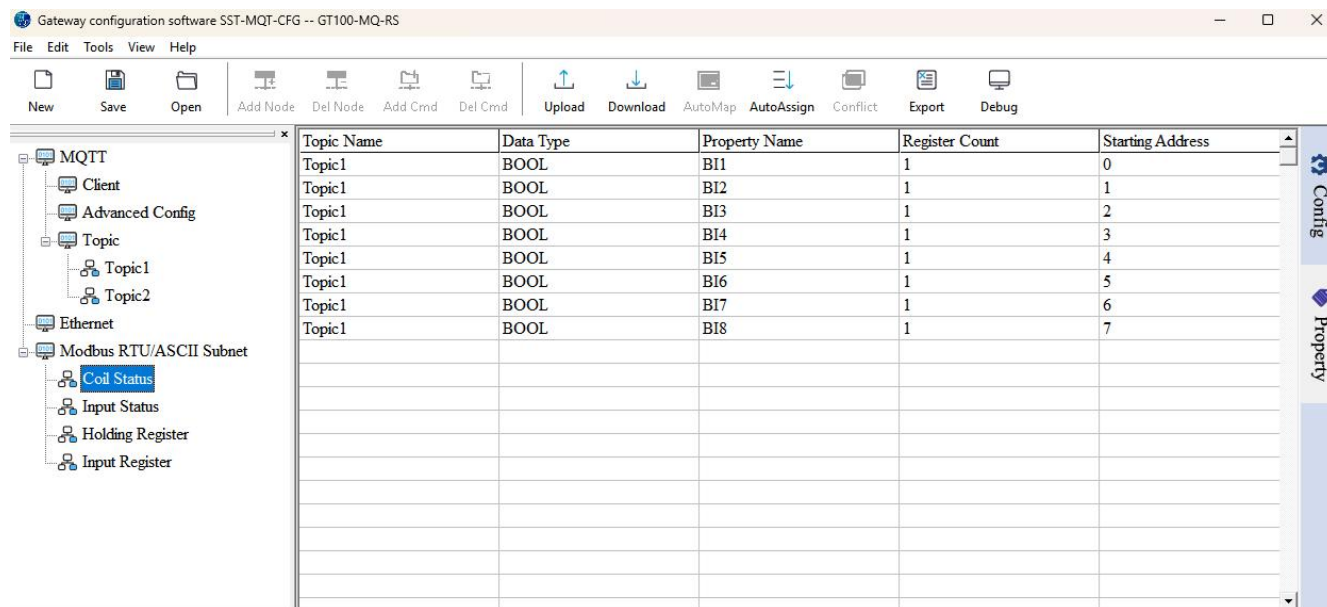
5.5.5 AutoMap

Click “AutoMap” to automatically calculate the non-conflicting mapped memory address for each command.

5.6 Property Operations

5.6.1 Property Configuration

Click “Property” in the right bar to enter the property window.



Topic Name: Select the topic for properties.

Data Type:

For command 1, 2, 5 and 15: The data type can be BOOL.

For command 3, 4, 6 and 16:

When using 1 register, the data type can be UINT/INT16.

When using 2 registers, the data type can be UINT32/INT32/Float/UINT32V/INT32V/FloatV.

When using 4 registers, the data type can be UINT64/INT64/Double/UINT64V/INT64V/DoubleV.

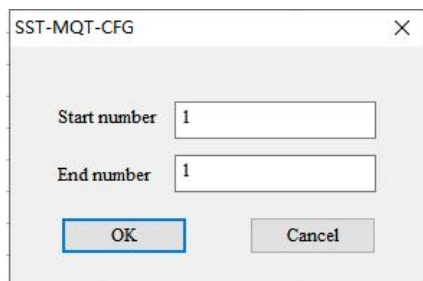
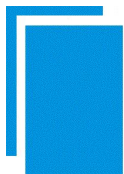
“V” refers to exchange of the bytes.

For example: Float 12.56 corresponds to HEX F5 C3 41 48. FloatV 12.56 corresponds to HEX 41 48 F5 C3.

Double 3.38 corresponds to HEX D7 0A 70 A3 0A 3D 40 0B. DoubleV 3.38 corresponds to 40 0B 0A 3D 70 A3 D7 0A.

Property Name: The property can be named automatically in orders or as custom names.

Copy Property Name: Click the right mouse button on any command option, and then select copy property name selection. Click the left mouse button, and fill in the start and end numbers of the properties to be copied in the prompt box.



Paste Property Name: Click the right mouse button on any command option, then select paste property name selection, fill in the prompt box to paste attribute name start number and end number to complete the paste attribute name.

Copy scope = Paste scope.

Copy scope > Paste scope, modifications are subject to pasting.

Copy scope < Paste scope, modifications are subject to copying.

Register Count: The number of the register.

Starting Address: The starting address of data.

5.6.2 Name Properties in Order

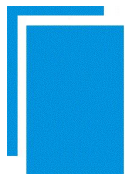
Select “Tool” >> “Name Properties in Order” to use this function.

The “Name Properties in Order” function is to automatically sort the properties of one topic and name them in order, with no conflict.

5.6.3 Properties Name Conflict Detection

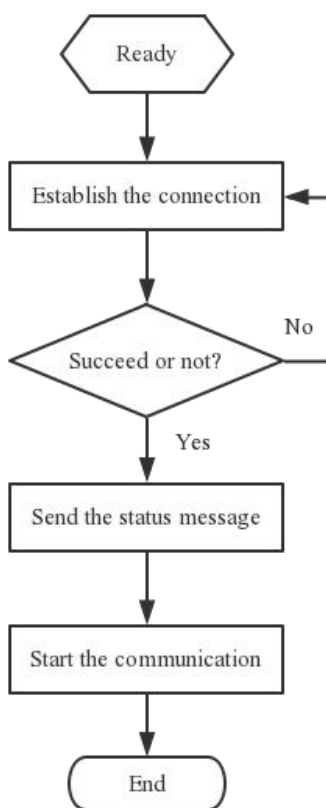
Select “Tool” >> “Properties Name Conflict Detection” to use this function.

The “Properties Name Conflict Detection” function is to automatically detect whether there are repeated property names in the same Topic and pop up the prompt.



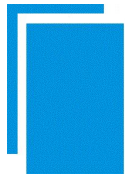
6 Working Principle

6.1 Connection Process



6.2 Data Exchange

The GT100-MQ-RS is able to connect Modbus RTU/ASCII devices to MQTT Servers, such as Amazon AWS IoT, EMQ and custom Servers.



Appendix A: MQTT Message Format

A.1 Default Format

A.1.1 Publish Message

Note: "//" indicates the comments part.

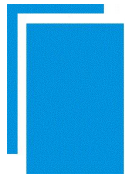
Property type:

Read Coil Status / Input Status:

```
{  
  "version":    "1.1.1.0",      //Version information  
  "time":      "19/03/07,17:04:07",  //Time stamp  
  "properties": {  
    //When publishing messages in property type, the GT100-MQ-RS can publish 8 properties at most in one  
    //message. If more than 8 properties need to be transmitted, the GT100-MQ-RS will publish multiple messages of 8  
    //or less properties.  
    "BI3":    false,  
    "BI4":    true,  
    "BI5":    false,  
    "BI6":    false  
  }  
}
```

Read Holding Register / Input Register:

```
{  
  "version":    "1.1.1.0",  
  "time":      "19/03/07,17:10:03",  
  "properties": {  
    "AI2":    9999,    //The data type (for example, Float) consists with the property configuration  
    "AI3":    8888,  
    "AI4":    7777,  
    "AI5":    6666  
  }  
}
```



A.1.2 Subscribe Message

Note: "/" indicates the comments part.

Property type: (Up to 8 properties can be changed at one time)

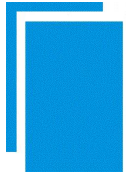
Modbus Master/Slave / Modbus TCP Server:

Change Coil/Input Status:

```
{  
  "properties": {  
    "BO3": false,  
    "BO4": true,  
    "BO5": false  
  }  
}
```

Change Holding/Input Register:

```
{  
  "properties": {  
    "AO3": 123,  
    "AO4": 321,  
    "AO5": 555  
  }  
}
```



A.2 Custom Message Format

```
{
  "timestamp": |TIMESTAMP|,
  "values": {
    |#each VALUES|
    "name": "|PROPERTYNAME|", "v": |VALUE||#unless @last|,/unless|
  }
}
```

(1) Variable description

|PROPERTYNAME|: Property Name. For example, AI1.

|VALUE|: The value of the property.

|TIMESTAMP|: The time when the property was read.

(2) Grammar

```
|#each VALUES|
  |PROPERTYNAME|, |VALUE|, |TIMESTAMP|,
|/each|
```

The "each" is to generate text for each property. Users just need to edit one single property template between "|#each VALUES|" and "|/each|" and it will be copied to every property. Up to 8 properties can be published in one message.

Template example:

```
{
  |#each VALUES|
    "|PROPERTYNAME|_value": |VALUE|,
    "|PROPERTYNAME|_timestamp": |TIMESTAMP|,
  |/each|
}
```



Message example:

```
{  
  "AI1_value": 23,  
  "AI1_timestamp": 1456150184825,  
  "AI2_value": 17,  
  "AI2_timestamp": 1456150184984,  
  ...  
}
```

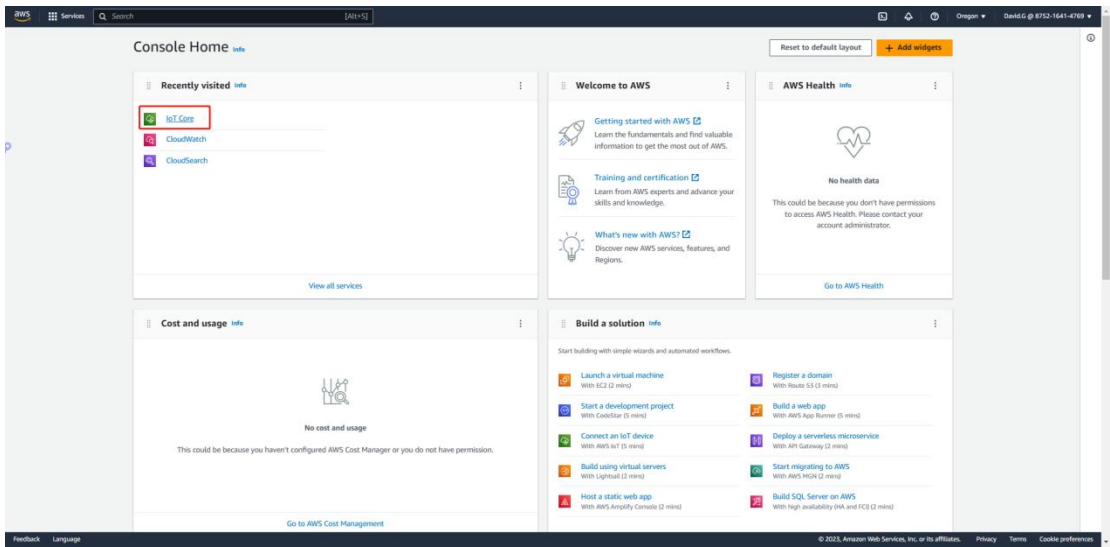
(3) Additional grammar description:

|#unless @last|,/unless|

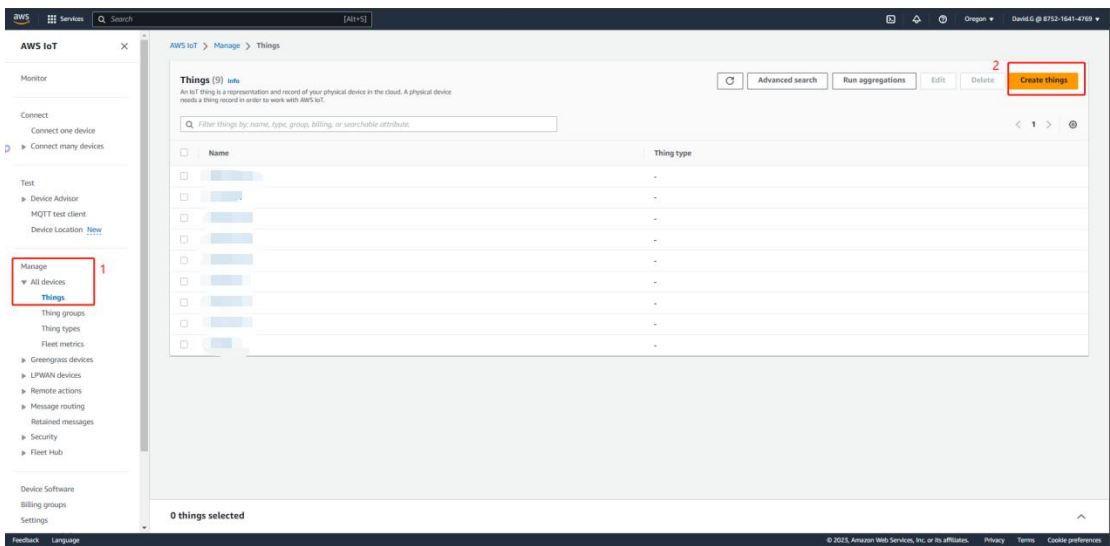
This line means "Must insert a comma at the end of each line, unless it's the last line". It cancels the comma in the last line. If users regard the comma in the last line as a syntax error, the template must contain this line.

Appendix B: Connecting to AWS IoT Application Notes

Log in to the AWS homepage and click IoTCore:



Click "Manage ->All Devices ->Things", and click "Create things" on the right,



GT100-MQ-RS Modbus / MQTT Gateway User Manual

Select to Create single thing, click Next,

The screenshot shows the AWS IoT console interface. On the left is a navigation sidebar with categories: Monitor, Connect, Test, and Manage. The 'Manage' section is expanded, showing 'All devices' and 'Things'. The 'Things' section is further expanded, listing 'Thing groups', 'Thing types', and 'Fleet metrics'. The main content area is titled 'Create things' and includes a sub-header 'Number of things to create'. There are two radio button options: 'Create single thing' (selected) and 'Create many things'. Below the options are 'Cancel' and 'Next' buttons.

Enter Thing name and click Next,

The screenshot shows the 'Specify thing properties' page in the AWS IoT console. The page is divided into three steps: Step 1 (Specify thing properties), Step 2 (optional: Configure device certificate), and Step 3 (optional: Attach policies to certificate). The 'Thing name' field is highlighted with a red box and contains the text 'DavidG-test01'. Below the name field is a note: 'Thing names can contain only letters, numbers, hyphens, colons, or underscores. A thing name can't contain any spaces.' The 'Additional configurations' section includes options for 'Thing type', 'Searchable thing attributes', 'Thing groups', and 'Billing group'. The 'Device Shadow' section has three radio button options: 'No shadow' (selected), 'Named shadow', and 'Unnamed shadow (classic)'. At the bottom right, there are 'Cancel' and 'Next' buttons, with the 'Next' button highlighted by a red box.

GT100-MQ-RS Modbus / MQTT Gateway User Manual

Select Auto-generate a new certificate and click Next,

The screenshot shows the AWS IoT console interface. On the left is a navigation menu with sections like Monitor, Connect, Test, and Manage. The main content area is titled 'Configure device certificate - optional' and includes a description: 'A device requires a certificate to connect to AWS IoT. You can choose how to register a certificate for your device now, or you can create and register a certificate for your device later. Your device won't be able to connect to AWS IoT until it has an active certificate with an appropriate policy.' Below this, there are four radio button options under the heading 'Device certificate':

- ☒ Auto-generate a new certificate (recommended) - Generate a certificate, public key, and private key using AWS IoT's certificate authority. (This option is highlighted with a red box)
- ☐ Use my certificate - Use a certificate signed by your own certificate authority.
- ☐ Upload CSR - Register your CA and use your own certificates on one or many devices.
- ☐ Skip creating a certificate at this time - You can create a certificate for this thing and attach a policy to the certificate at a later time.

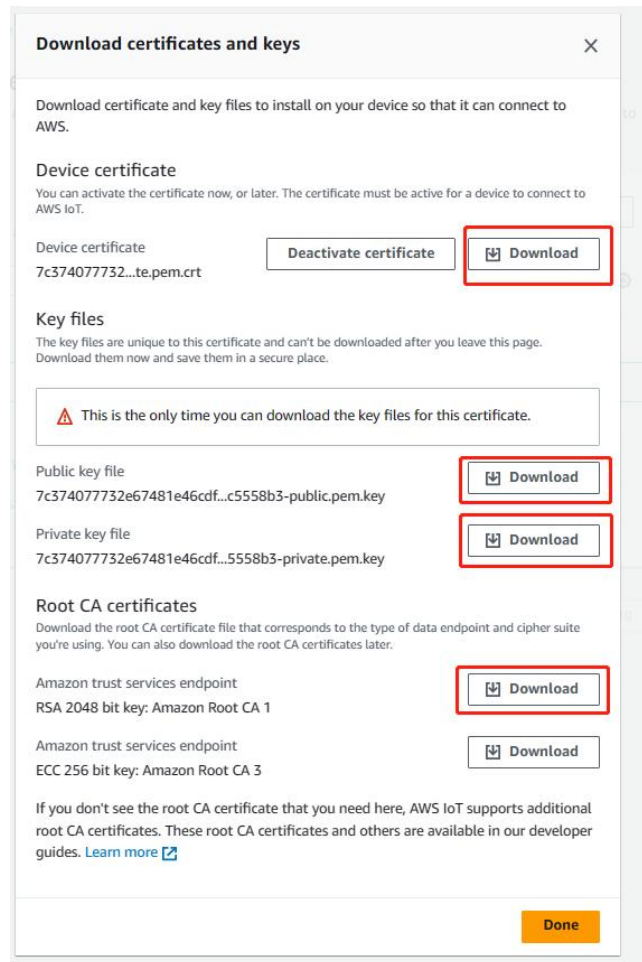
At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Next'. The 'Next' button is highlighted with a red box.

A certificate attachment policy that determines the permissions for subscription and publication,

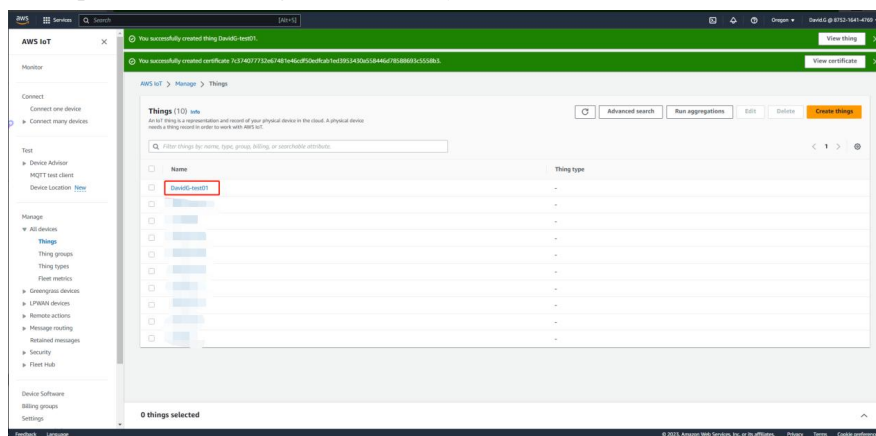
The screenshot shows the AWS IoT console interface for the 'Attach policies to certificate' step. The main content area is titled 'Attach policies to certificate - optional' and includes a description: 'AWS IoT policies grant or deny access to AWS IoT resources. Attaching policies to the device certificate applies this access to the device.' Below this, there is a section titled 'Policies (1/5)' with a search bar and a list of policies. The first policy, named 'test', is selected with a checkbox and highlighted with a red box. At the bottom right, there are three buttons: 'Cancel', 'Previous', and 'Create thing'. The 'Create thing' button is highlighted with a red box.

GT100-MQ-RS Modbus / MQTT Gateway User Manual

Download the four certificates in the following figure,

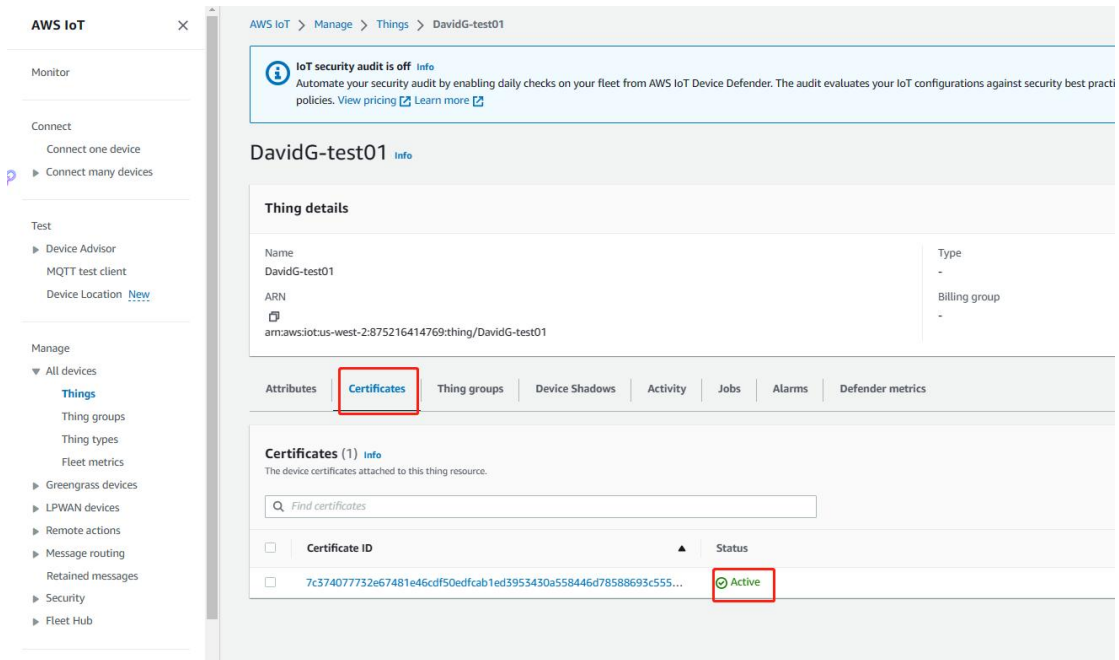


After completing these steps, the AWS Thing is successfully created.

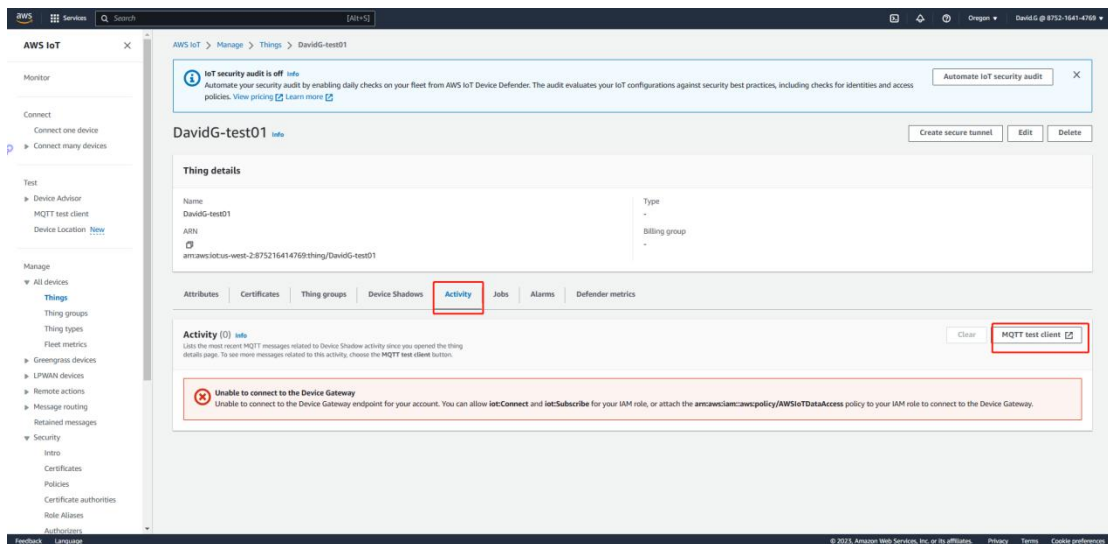


GT100-MQ-RS Modbus / MQTT Gateway User Manual

After clicking on the device, confirm whether the certificate has been activated in Certificates,



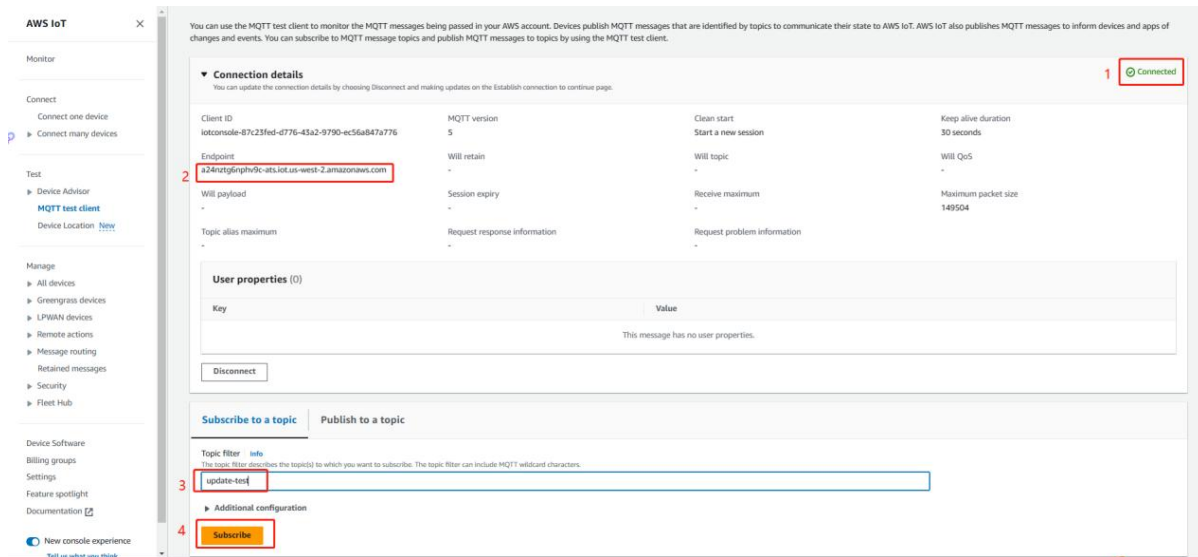
Select the Activity and click to enter the MQTT test client,



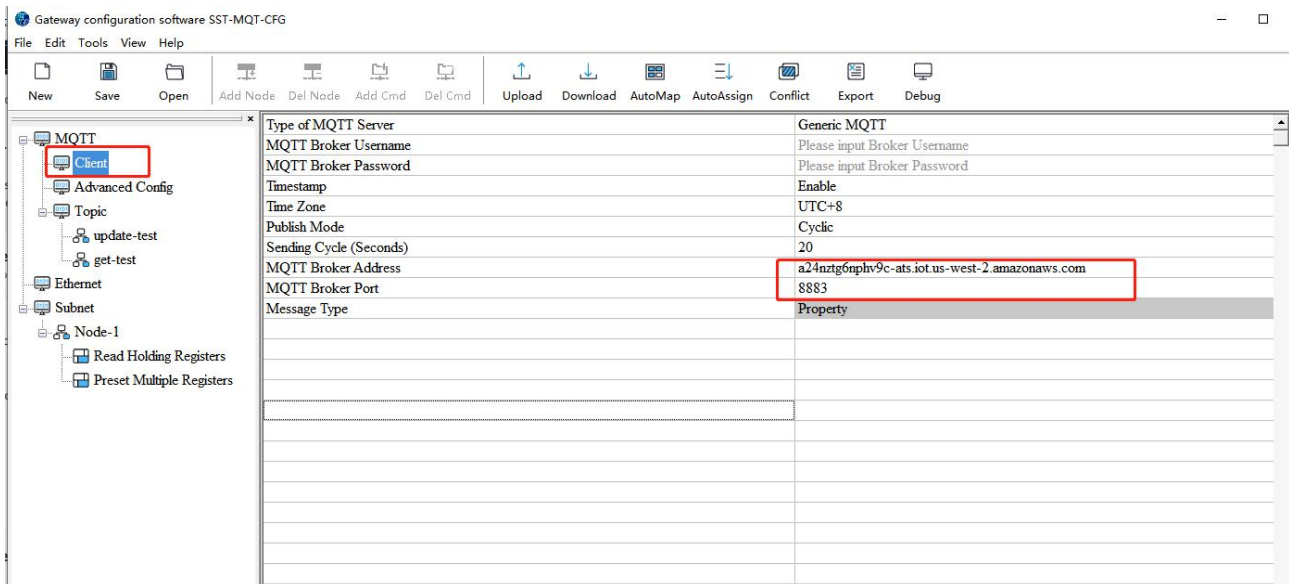
1. Confirm whether the connection is successful
2. Copy the URL in the endpoint to SST-MQT-CFG
3. Configure the topic to subscribe to, and the topic name must be consistent with that in SST-MQT-CFG

GT100-MQ-RS Modbus / MQTT Gateway User Manual

4. Click to subscribe

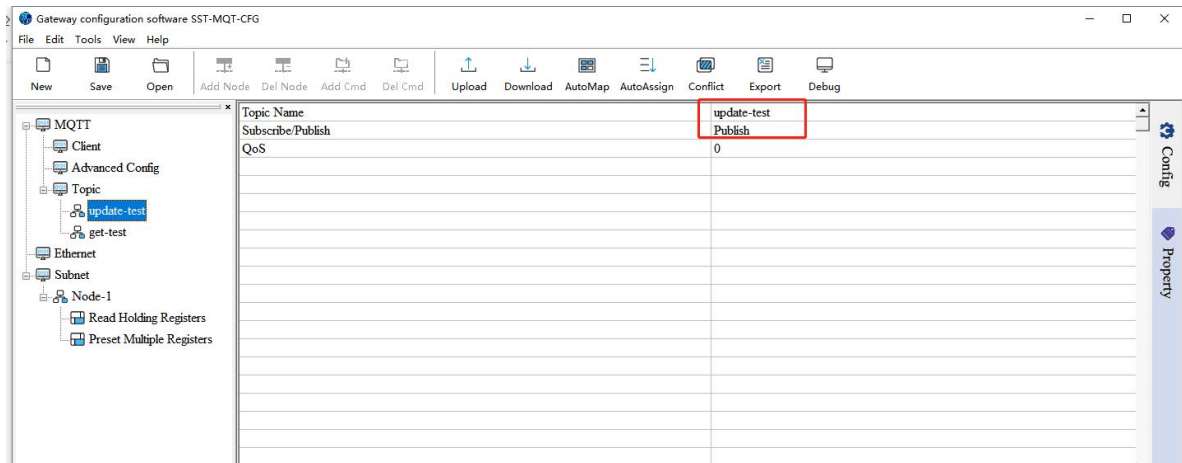


In SST-MQT-CFG, configure the Broker address and Port Number 8883:



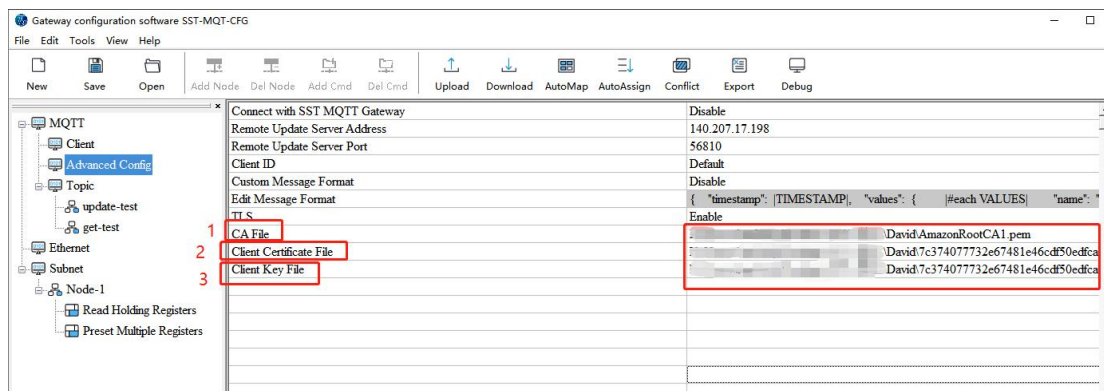
GT100-MQ-RS Modbus / MQTT Gateway User Manual

Configure topic name and publish permissions



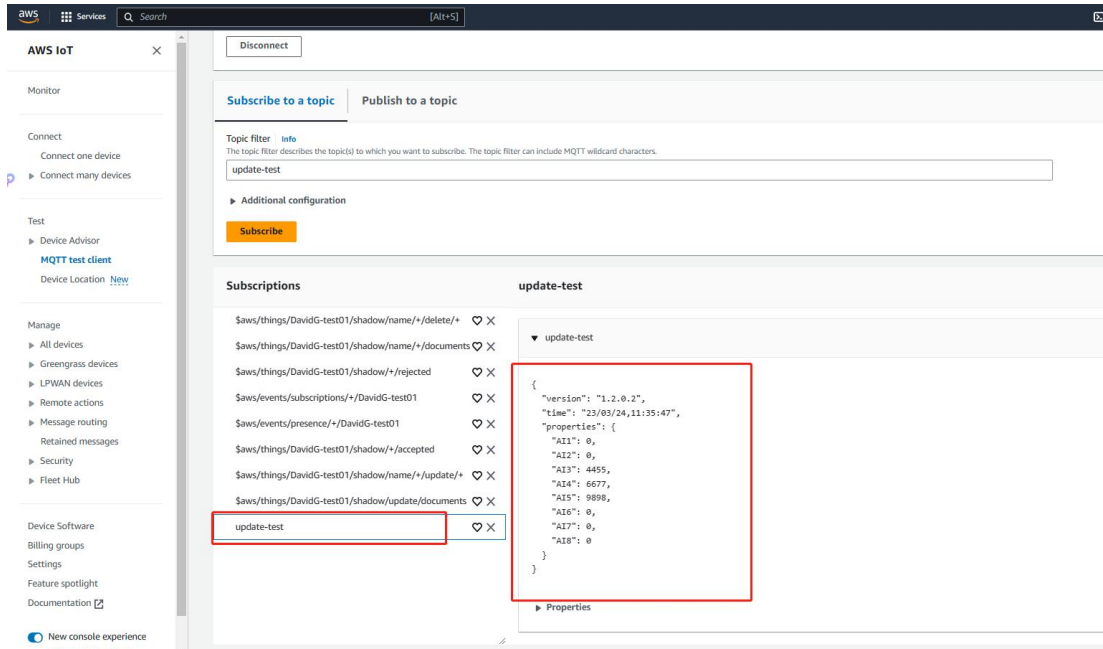
Select the certificate file, download the configuration, and restart the device

1. AmazonRootCA1.pem
2. 7c374077732e67481e46cdf50edfcab1ed3953430a558446d78588693c5558b3-certificate.pem.crt
3. 7c374077732e67481e46cdf50edfcab1ed3953430a558446d78588693c5558b3-private.pem.key

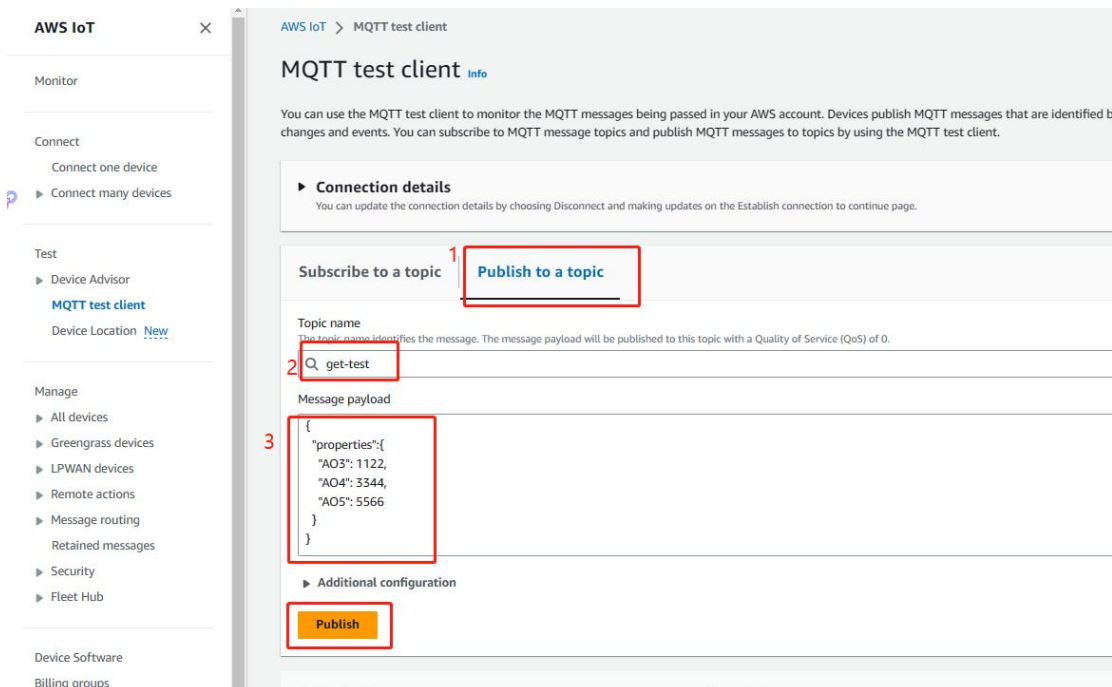


GT100-MQ-RS Modbus / MQTT Gateway User Manual

Once your device reboots and connects to AWS, if it's set to send data at regular intervals, just wait for the device to hit its scheduled data transmission period, and then you can view all the data it sends on the AWS IoT Core.

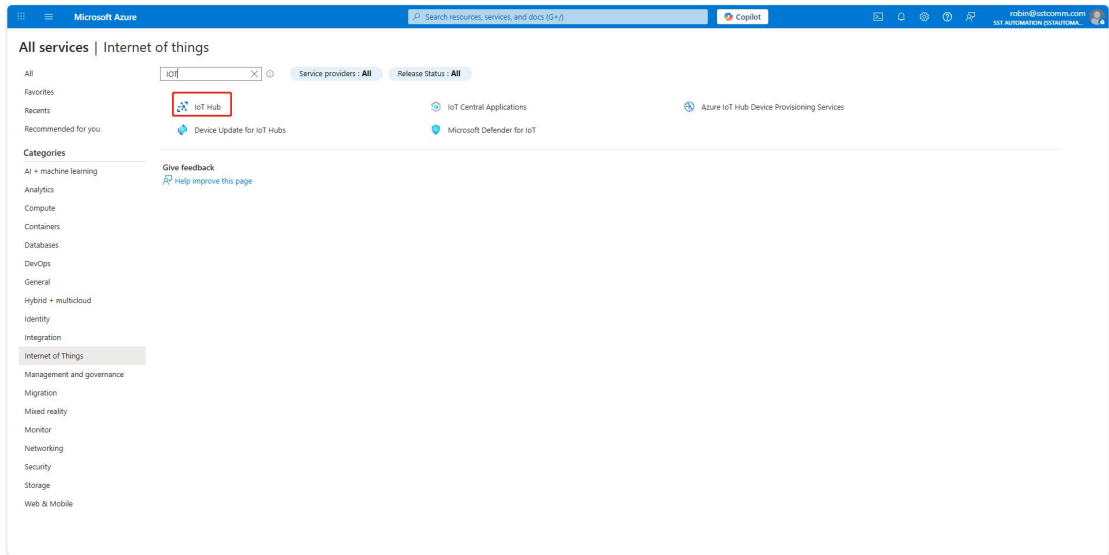


AWS IoT Core supports publishing data to devices. You need to configure topic in SST-MQT-CFG and have subscribe permission:

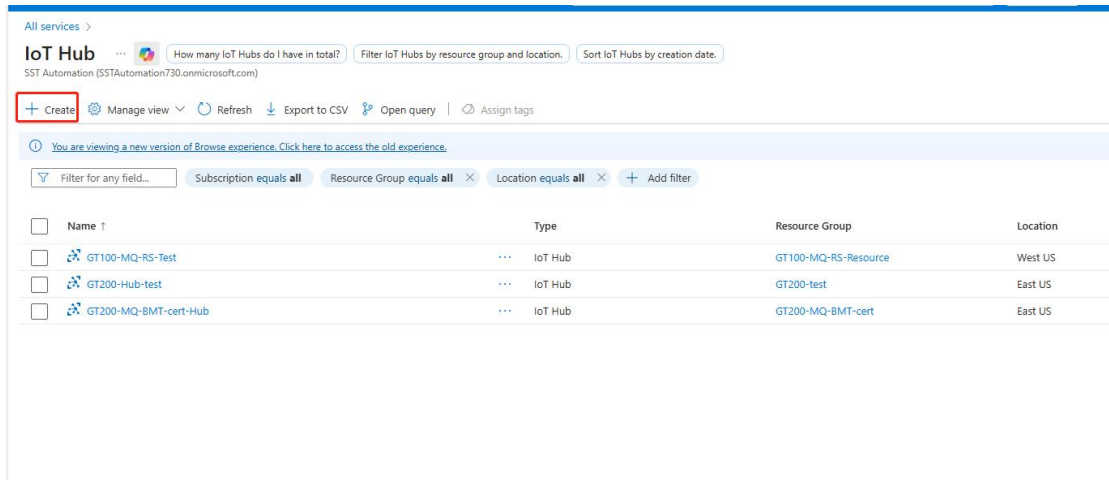


Appendix C: Connecting to Azure IoT via Self-Signed Certificate Method Application Notes

Log in to the Azure homepage and click IoT Hub:



Click "Create" and fill in the relevant information to set up your IoT Hub device:



GT100-MQ-RS Modbus / MQTT Gateway User Manual

This application note demonstrates how to connect to Azure IoT using X.509 certificates. To begin, you need to register the certificates.

You can download and install the OpenSSL tool to generate the certificates:

<https://slproweb.com/products/Win32OpenSSL.html>

acquire in bacon fat before delivery to your company's machines.

"We would like to transfer a donation to Win32 OpenSSL, but only without Paypal. Can we transfer directly? Please send me the bank details with a quote." My bank account is 000012345 and my routing number is 000012345. I look forward to your donation. Huh...that's amazing! I've got the same combination on my luggage!

Win32/Win64 OpenSSL Screenshots

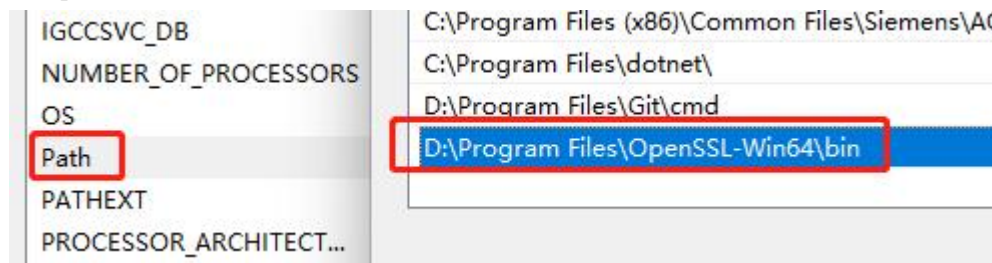
Screenshot of the Win32/Win64 OpenSSL Installer

Download Win32/Win64 OpenSSL

Download Win32/Win64 OpenSSL today using the links below:

File	Type	Description
Win64 OpenSSL v3.6.0 Light EXE MSI	6MB Installer	Installs the most commonly used essentials of Win64 OpenSSL v3.6.0 (Recommended for users by the creators of OpenSSL). Only installs on 64-bit versions of Windows and targets Intel x64 chips. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v3.6.0 EXE MSI	240MB Installer	Installs Win64 OpenSSL v3.6.0 (Recommended for software developers by the creators of OpenSSL). Only installs on 64-bit versions of Windows and targets Intel x64 chips. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win32 OpenSSL v3.6.0 Light EXE MSI	3MB Installer	Installs the most commonly used essentials of Win32 OpenSSL v3.6.0 (Only install this if you need 32-bit OpenSSL for Windows). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win32 OpenSSL v3.6.0 EXE MSI	207MB Installer	Installs Win32 OpenSSL v3.6.0 (Only install this if you need 32-bit OpenSSL for Windows). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v3.6.0 Light for ARM EXE MSI	6MB Installer	Installs the most commonly used essentials of Win64 OpenSSL v3.6.0 for ARM64 devices (Only install this VERY EXPERIMENTAL build if you want to try 64-bit OpenSSL for Windows on ARM processors). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v3.6.0 for ARM EXE MSI	207MB Installer	Installs Win64 OpenSSL v3.6.0 for ARM64 devices (Only install this VERY EXPERIMENTAL build if you want to try 64-bit OpenSSL for Windows on ARM processors). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
WinUniversal OpenSSL v3.6.0 Light for all architectures EXE	15MB Installer	Installs the most commonly used essentials of WinUniversal OpenSSL v3.6.0 for all devices (Only install this EXPERIMENTAL build if you need universal Windows binaries of OpenSSL for Windows for all architectures - Intel x86, Intel x64, and ARM - includes static binaries and a portable installer variant). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
WinUniversal OpenSSL v3.6.0 for all architectures EXE	174MB Installer	Installs WinUniversal OpenSSL v3.6.0 for all devices (Only install this EXPERIMENTAL build if you need universal Windows binaries and compiler libraries of OpenSSL for Windows for all architectures - Intel x86, Intel x64, and ARM - includes static binaries and a portable installer variant). Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v3.5.4 Light EXE MSI	6MB Installer	Installs the most commonly used essentials of Win64 OpenSSL v3.5.4 (Recommended for users by the creators of OpenSSL). Only installs on 64-bit versions of Windows and targets Intel x64 chips. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.
Win64 OpenSSL v3.5.4 EXE MSI	281MB Installer	Installs Win64 OpenSSL v3.5.4 (Recommended for software developers by the creators of OpenSSL). Only installs on 64-bit versions of Windows and targets Intel x64 chips. Note that this is a default build of OpenSSL and is subject to local and state laws. More information can be found in the legal agreement of the installation.

After installation, add the OpenSSL bin directory path to the system environment variables under "Path". For example:
D:\Program Files\OpenSSL-Win64\bin

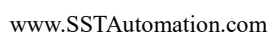
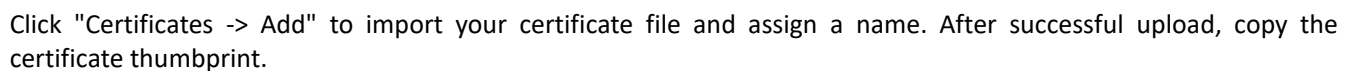


Open Command Prompt and enter the following command to verify if OpenSSL is installed successfully:

openssl version

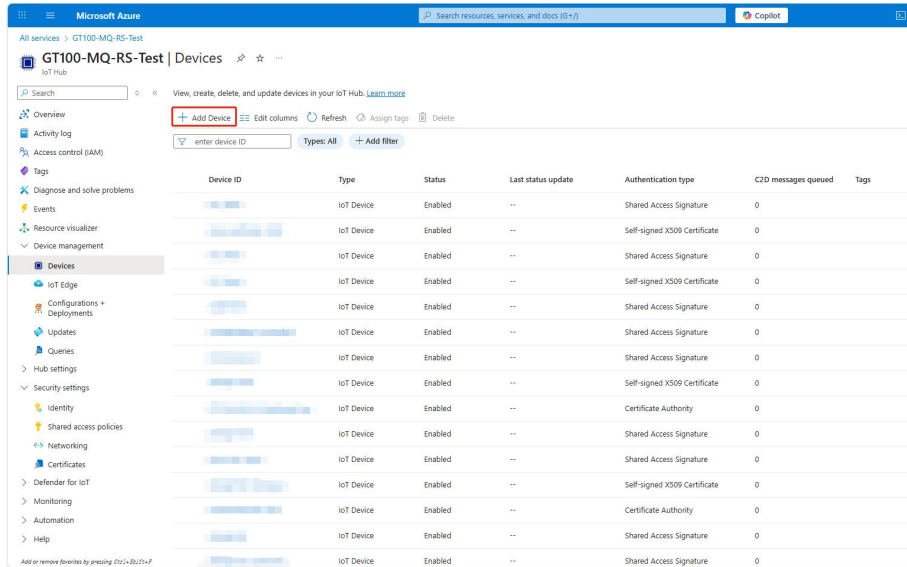
```
C:\Users\DELL>openssl version
OpenSSL 3.6.0 1 Oct 2025 (Library: OpenSSL 3.6.0 1 Oct 2025)
C:\Users\DELL>
```

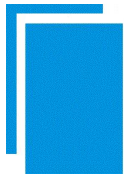
```
openssl req -x509 -sha256 -nodes -days 365 -newkey rsa:2048 -keyout privateKey.key -out certificate.crt
```



GT100-MQ-RS Modbus / MQTT Gateway User Manual

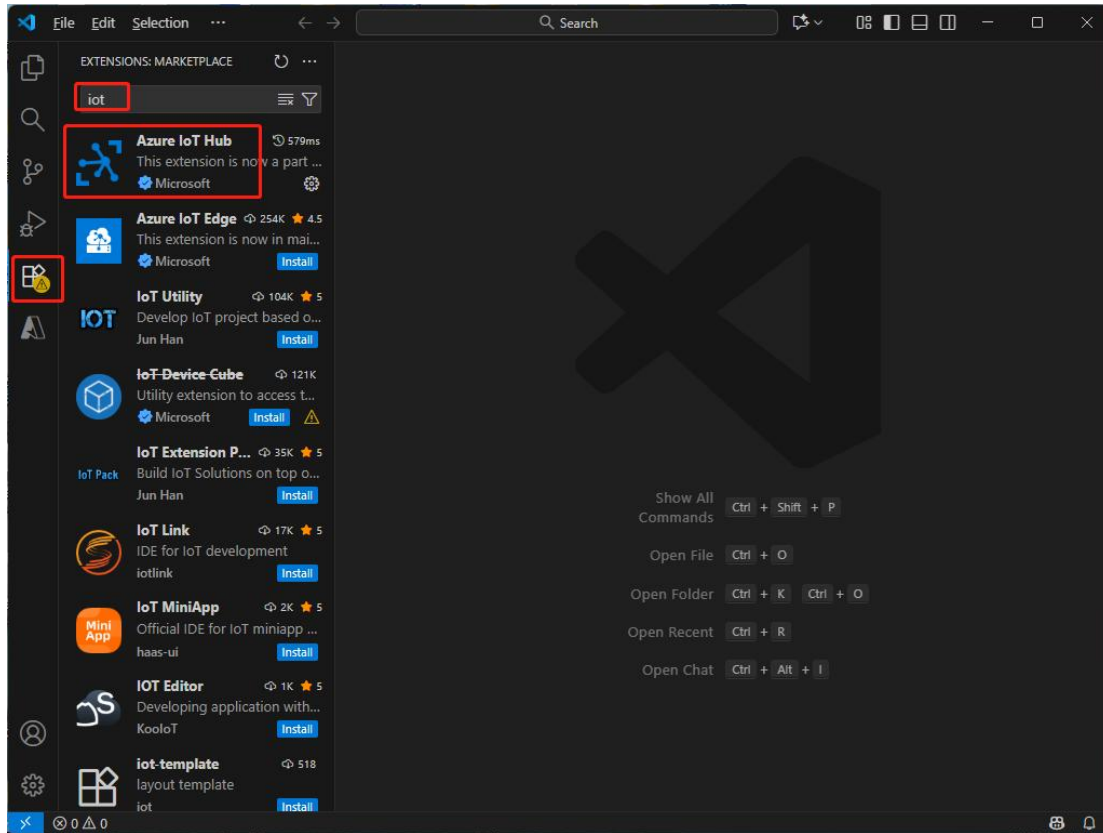
Click "Devices > Add Device" to create a new device named "GT100-MQ-IE-test". Select "X.509 Self-Signed" as the authentication type, then enter the certificate thumbprint in both the "Primary Thumbprint" and "Secondary Thumbprint" fields.





GT100-MQ-RS Modbus / MQTT Gateway User Manual

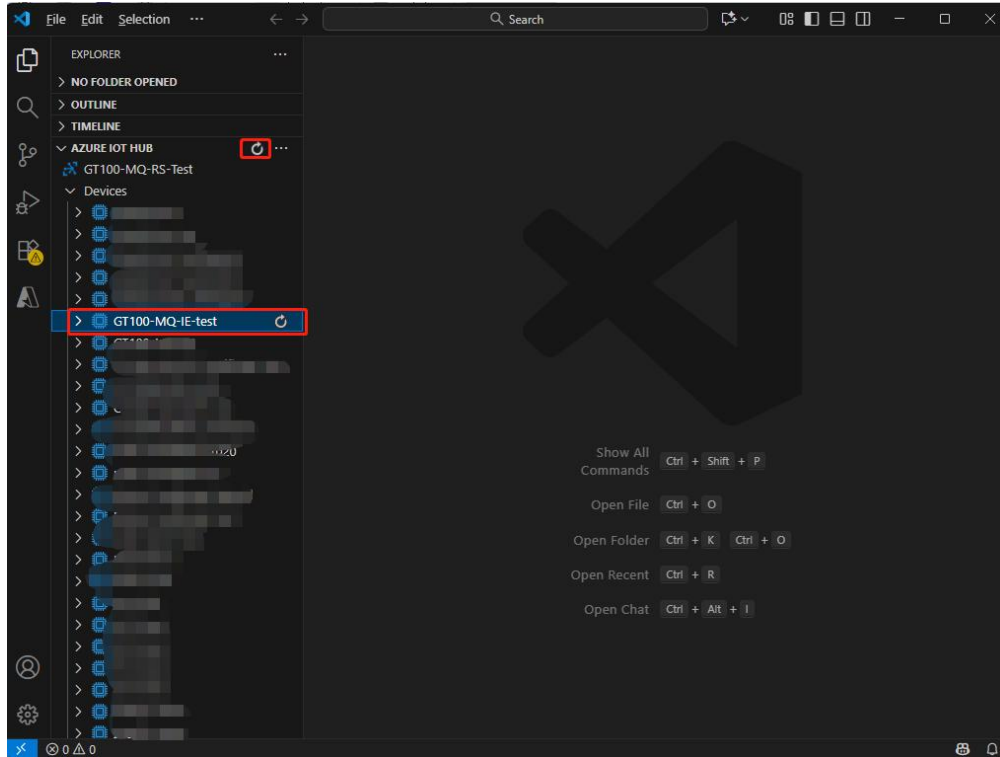
Download and install Visual Studio Code, then install the Azure IoT extension and log in with your Azure account:

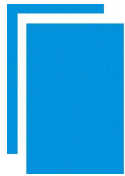




GT100-MQ-RS Modbus / MQTT Gateway User Manual

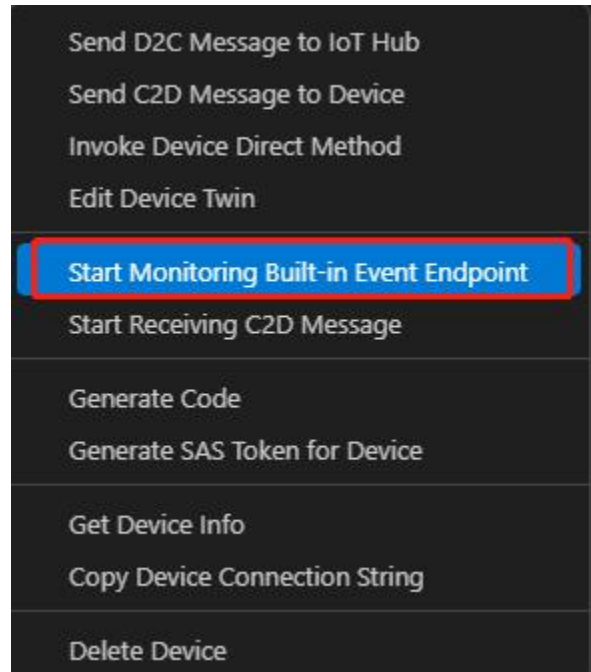
Click the refresh button next to "AZURE IOT HUB" to retrieve the newly added device.



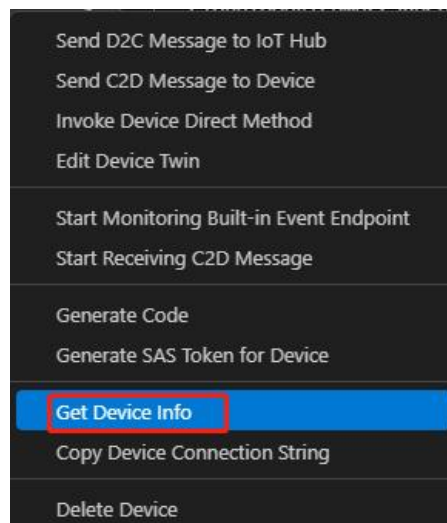


GT100-MQ-RS Modbus / MQTT Gateway User Manual

Right-click on the "GT100-MQ-IE-test" device and select "Start Monitoring Built-in Event Endpoint" to monitor the data published by the device.



The device has been successfully created in the Azure IoT platform. To proceed with configuring the GT100-MQ-IE, right-click on the "GT100-MQ-IE-test" device in VSCode and select "Get Device Info" to retrieve the connection details for this device:



GT100-MQ-RS Modbus / MQTT Gateway User Manual

```
"connectionString": "HostName=GT100-MQ-RS-Test.azure-devices.net;  
DeviceId=GT100-MQ-IE-test;x509=true"
```

MQTT Broker Username = **HostName/Device ID** =
GT100-MQ-RS-Test.azure-devices.net/GT100-MQ-IE-test

MQTT Broker Address = **HostName** = GT100-MQ-RS-Test.azure-devices.net
MQTT Broker Port = 1883

Client ID = **Device ID** = GT100-MQ-IE-test

TLS: Enable

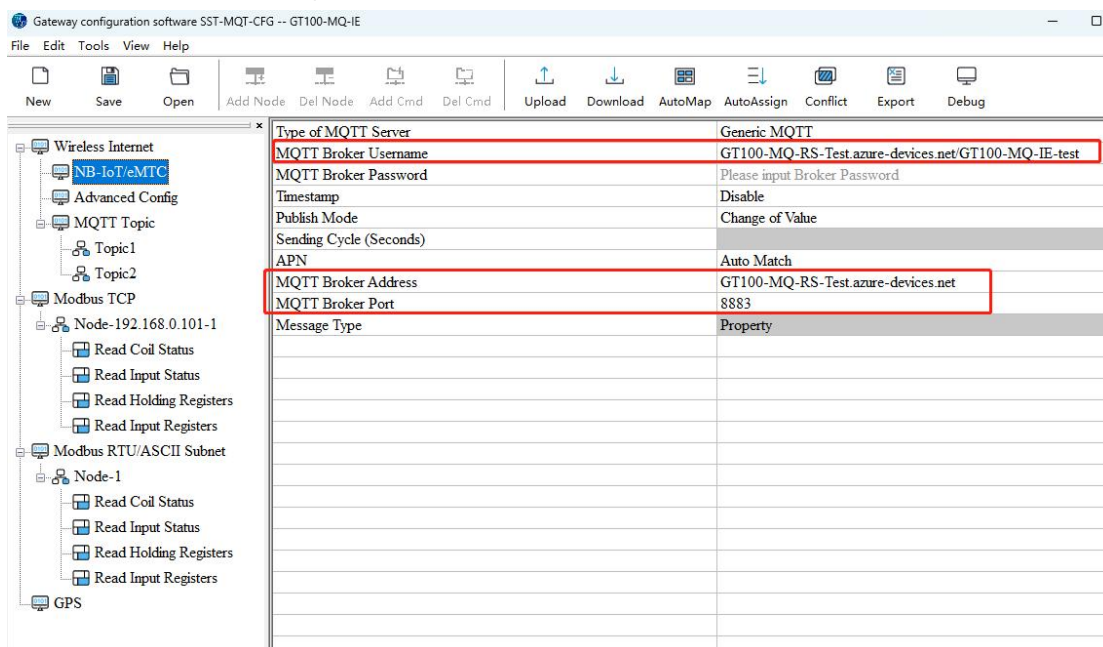
CA File=CA Certificate from Azure Platform - Converted PEM Certificate =
DigiCertGlobalRootG2.pem

For instructions on how to obtain the certificate, please refer to: [Azure CA Certificate Conversion:](#)

Client Certificate File = OpenSSL-Generated"certificate"

Client Key File= OpenSSL-Generated"privateKey"

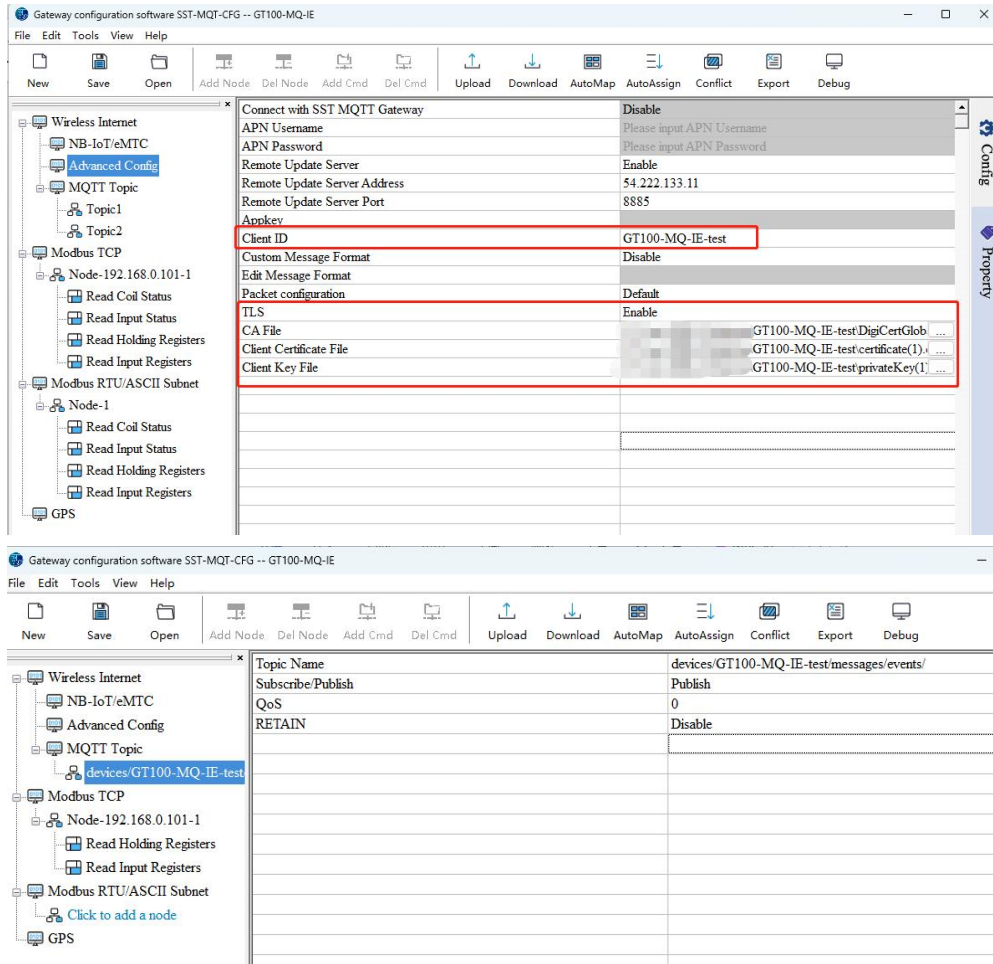
Topic Name =devices/**Device ID**/messages/events/ =
devices/**GT100-MQ-IE-test**/messages/events/

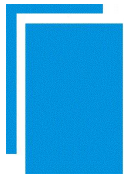


GT100-MQ-RS

Modbus / MQTT Gateway

User Manual



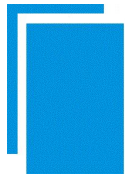


GT100-MQ-RS Modbus / MQTT Gateway User Manual

Data Received in VSCode After Device Data Change:

Mbslave1			
ID = 1: F = 03			
	Alias	00000	Alias
0		1234	
1		5678	
2		0	
3		0	
4		0	
5		0	
6		0	
7		0	
8		0	
9		0	

```
[Device] [Get] device info: {
  },
  "connectionString": "HostName=GT100-MQ-RS-Test.azure-devices.net;
  DeviceId=GT100-MQ-IE-test;x509=true"
}
[IoTHubMonitor] Start monitoring message arrived in built-in endpoint for device
[GT100-MQ-IE-test] ...
[IoTHubMonitor] Created partition receiver [0] for consumerGroup [$Default]
[IoTHubMonitor] Created partition receiver [1] for consumerGroup [$Default]
[IoTHubMonitor] Created partition receiver [2] for consumerGroup [$Default]
[IoTHubMonitor] Created partition receiver [3] for consumerGroup [$Default]
[IoTHubMonitor] [4:48:59 PM] Message received from [GT100-MQ-IE-test]:
{
  "version": "3.4.4.2",
  "properties": {
    "AI1": 1234
  }
}
[IoTHubMonitor] [4:49:02 PM] Message received from [GT100-MQ-IE-test]:
{
  "version": "3.4.4.2",
  "properties": {
    "AI2": 5678
  }
}
```



Azure CA Certificate Conversion:

Obtaining the CA Certificate from the Azure Platform:

https://learn.microsoft.com/en-us/azure/security/fundamentals/azure-ca-details?wt.mc_id=searchAPI_azureportal_inproduct_rmskilling&sessionId=888c8f88d23f481d98eed381da7b1e41&tabs=root-and-subordinate-cas-list

Root Certificate Authorities	
 Expand table	
Certificate Authority	Serial Number / Thumbprint
DigiCert Global Root CA 	0x083be056904246b1a1756ac95991c74a A8985D3A65E5EC4B2D7D66D40C6DD2FB19C5436
DigiCert Global Root G2 	0x033af1e6a711a9a0bb2864b11d09fae5 DF3C24F9BFD666761B268073FE06D1CC8D4F82A4
DigiCert Global Root G3 	0x055556bcf25ea43535c3a40fd5ab4572 7E04DE896A3E666D00E687D33FFAD93BE83D349E
DigiCert TLS ECC P384 Root G5 	0x09e09365acf7d9c8b93e1c0b042a2ef3 17F3DE5E9F0F19E98EF61F32266E20C407AE30EE
DigiCert TLS RSA 4096 Root G5 	0x08f9b478a8fa7eda6a333789de7ccf8a A78849DC5D7C758C8CE399856B3AAD0B2A57135
Microsoft ECC Root Certificate Authority 2017 	0x66f23daf87de8bb14aea0c573101c2ec 999A64C37FF47D9FAB95F14769891460EEC4C3C5
Microsoft RSA Root Certificate Authority 2017 	0x1ed397095fd8b4b347701eaabe7f45b3 73A5E64A3BFF8316FF0EDCCC618A906E4EAE4D74

Send the following command to convert the certificate format:

`openssl x509 -inform der -in DigiCertGlobalRootG2.crt -out DigiCertGlobalRootG2.pem`

```
C:\Users\DELL>openssl x509 -inform der -in DigiCertGlobalRootG2.crt -out DigiCertGlobalRootG2.pem
C:\Users\DELL>
```