STG Series WMBus (Wireless MBus) – Modbus TCP/RTU Gateway with 2 x 10/100Base-T(x) Ports, 1 x RS232 and 1 x RS485 Serial Ports and/or BPL (Broadband Power Line Link) User Manual

1. About STG Series Wireless Mbus Gateways

STG Series Wireless MBus (WMBus) Gateways are designed for industrial-grade Radio Frequency (RF) communication. STG Series WMBus Gateways are tailored to perform various features such as wide temperature range, wide power input range and several connectivity ports. Thus, STG Series WMBus Gateways are the best choice for smart metering, power utility, telecommunication and all other applications that require industrial Wireless MBus Radio Frequency connectivity.

STG Series which have REDZ Broadband Power Line (BPL) link allows devices to communicate with full transparent TCP/IP standard over Low Voltage power lines and allows easy connection between TCP/IP based terminals without use of extra cables.

STG Series WMBus Gateways can listen WMBus RF network and connect Serial and/or ETH based devices with Wireless MBus devices. All communication can be done over Radio Frequency network based on WMBus standard. STG Series WMBus Gateways are all in one devices and can operate in 3 main modes: Transparent Mode, WMBus OMS Converter to Modbus RTU packages (in this mode meter data can also be sent to MQTT Server in several formats), WMBus OMS Converter to Modbus TCP packages. Typical applications: Automated Meter reading, Home – Building – Industrial Automation, Wireless Sensors, Telemetry...

2. Hardware Features

STG Series WMBus Gateways have the versions with and without BPL (Broadband Power Line) Link.

2.1 Features

- Supports 2 x 10/100Base-T(X) ports

- Supports Full/Half-Duplex, auto MDI/MDI-X on each port

- DHCP Server Capability

- Supports 1 x RS232 and 1 x RS485 Serial Connection up to 460800 Baud

- Embedded web interface for ease of use

- 868MHz Wireless MBus (WMBus) Radio Frequency (RF) Communication

- 3 Main Device Functions:

  WMBus OMS to Modbus TCP Converter (and/or send Meter Data to MQTT Server)

  WMBus OMS to Modbus RTU Converter

  Transparent Mode (Sends/Receive WMBus Packages to/from TCP/IP or Serial Side)

- Up to 20 Modbus device connection in Modbus TCP or RTU Conversion Modes

- MQTT Publisher with different data transfer options

  Raw WMBus Decrypted Frame

  Parsed WMBus Frame As Objects

  Parsed WMBus Frame As Modbus Frame

- WMBus link mode Configurable (S - Mode, T - Mode, C - Mode, C/T - Mode together)

- AES Decryption of Received Frames for up to 40 WMBus Devices (Mode 5, Mode 7, Mode 128 and custom modes)

- Supported CI Values: 53h, 5Bh, 60h, 6Ch, 6Dh, 6Eh, 6Fh, 72h, 74h, 75h, 78h, 7Ah, 7Ch, 7Dh, 80h, 8Ah, 8Bh, 8Ch, 8Dh, 8Eh, 8Fh, C3h, C4h, C5h

- Unlimited Numbers of WMbus device data can be listened over air andWMBus frames can be sent to remote server in Transparent Mode

- WMBus Radio Power Level Configurable (-1 dBm to 13 dBm) when sending WMBus frames in Transparent Mode

- AES Encryption of Transmitted frames is available in Transparent Mode

- WMBus device mode Configurable (Meter - send frames or Other Device - listen frames)

- Easy to follow WMBus data packages on web interface

- Easy monitor of parsed WMbus OMS Parsed data on web interface

- Easy to follow Device Status on web interface

- Black List and White List based WMBus package filter

- Firmware Upgrade over Web

- 2 firmware storage capability on same device (1 active only)

- AC or DC wide range power options

- Wide operating temperature range from -25 to 70 °C AC and -40 to 85 °C DC power input versions

- Rugged Metal IP-40 housing design

- DIN-Rail mounting

2.2 Extra Features for Models with BPL

- Supports 2 x 10/100Base-T(X) ports + 1 x BPL link

- Wide range 3 phase AC input

- Supports up to 30Mbps PHY rate on BPL with Up to 10 hops and 1000 nodes

- Up to 432 sub-carriers from 2 to 28MHz analog bandwidth

- Support LDPC-C FEC with 128-bit AES core

- Plug and play with Master/Slave selection via web interface

3. Installation

Each device has a Din-Rail kit on rear panel. The Din-Rail kit helps device to fix on the Din-Rail. Slant the switch and mount the metal spring to Din-Rail.

Then Push the switch toward the Din-Rail until you heard a "click" sound.

4. Front Panel Description

4.1 STG154 & STG254

1. Standard SMA female Antenna interface, 50 ohm.

2. Micro USB or USB Type-C Console port for LOG in 115200 baud.

   Console Tx and Rx Blinks when data transmission occurs.

3. WMBus Activity LEDs

   ALIVE: Blinks during normal operation.

   WMBus Tx and Rx Blinks when data transmission occurs.

4. Device Status LEDs

   STATUS: Blinks based on device operation.

   - When TCP line used blinks during no connection and keeps ON after TCP connection.

   - When Serial line used keeps ON.

   Device Tx and Rx Blinks when data transmission occurs.

   SERVER: Keeps ON after selecting Server from Server-Client Operating Modes. Keeps OFF if Client operating

   Mode selected.

5. ETHERNET Activity LEDs for port 1, 2 and STG device itself. Blinks during ethernet activity.

6. 5 pin Terminal Block

   RS232: Tx, Rx and GND pins

RS485: A, B and GND pins

Can be activated over web interface and baud rate/data type configurable.

7. 10/100Base-T(X) Ethernet ports

4.2 STG655

1. Standard SMA female Antenna interface, 50 ohm.

2. Micro USB or USB Type-C Console port for LOG in 115200 baud.

   Console Tx and Rx Blinks when data transmission occurs.

3. WMBus Activity LEDs

   ALIVE: Blinks during normal operation.

   WMBus Tx and Rx Blinks when data transmission occurs.

4. Device Status LEDs

   STATUS: Blinks based on device operation.

   – When TCP line used blinks during no connection and keeps ON after TCP connection.

   – When Serial line used keeps ON.

   Device Tx and Rx Blinks when data transmission occurs.

   SERVER: Keeps ON after selecting Server from Server–Client Operating Modes. Keeps OFF if Client operating

   Mode selected.

5. ETHERNET Activity LEDs for port 1, 2 and STG device itself. Blinks during ethernet activity.

6. 5 pin Terminal Block for Serial Line

   RS232: Tx, Rx and GND pins

   RS485: A, B and GND pins

   Can be activated over web interface and baud rate/data type configurable

7. BPL Status LEDs

   ACTIVITY: Blinks during BPL Ethernet activity

   LINK: LED Turns ON if the link can be established over BPL

   MASTER INDICATION: LED Turns ON if the device is configured and powered as "BPL Master" device

8. 10/100Base-T(X) Ethernet ports

5. Top Panel Description

5.1 STG154
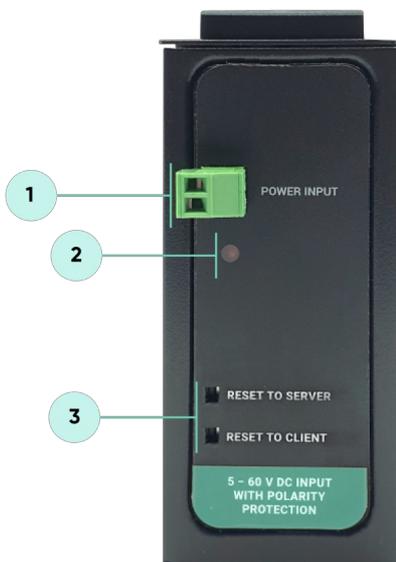
1. Power Input DC: 5-48V DC (max. 60V). Polarity protected so that the power input can be connected in any direction
2. Power LED: Turns ON when there is power in device
3. Reset Buttons

   RESET TO SERVER: Resets the device to factory setting as Server from Server-Client Operating Modes

   RESET TO CLIENT: Resets the device to factory setting as Client from Server-Client Operating Modes

   Reset can be done at any time by pushing any of the button for 5 seconds.

5.2 STG254
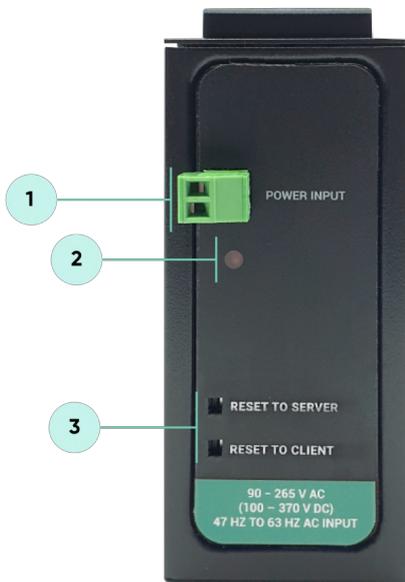
1. Power Input AC: 100 - 240V AC (120 – 370V DC), 50Hz to 60Hz AC Input
2. Power LED: Turns ON when there is power in device
3. Reset Buttons

   RESET TO SERVER: Resets the device to factory setting as Server from Server-Client Operating Modes

   RESET TO CLIENT: Resets the device to factory setting as Client from Server-Client Operating Modes

   Reset can be done at any time by pushing any of the buttons for more than 5 seconds.

5.3 STG655
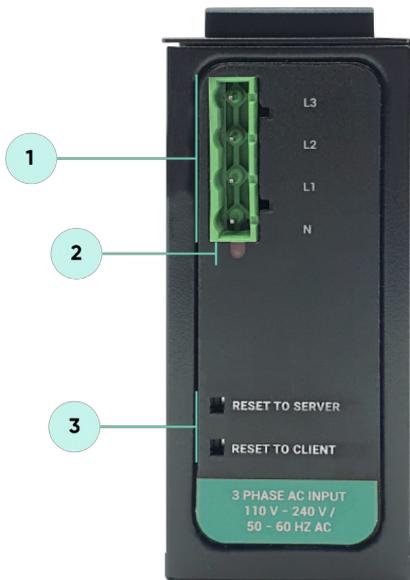
1. Power Input AC: 3 phase input, 110V–240V/50-60Hz. It is also ok to connect only single phase to the device such as L1-N connection only.

   AC Power supply use L1-N only. Phase 2-3 connections are used to BPL signal transmission.

2. Power LED: Turns ON when there is power in device

3. Reset Buttons

   RESET TO SERVER: Resets the device to factory setting as Server from Server-Client Operating Modes

   RESET TO CLIENT: Resets the device to factory setting as Client from Server-Client Operating Modes

   Reset can be done at any time by pushing any of the buttons for more than 5 seconds.

☝️

NOTE1 : BPL Model can be purchased in 2 versions:

1. P-N Model: Phase to neutral model (Standart Model). That version gets power from terminal pins 1 and 2 from phase and neutral. It can also transmit data from that pins and other pins usage is optionAl (Ex: Master can be connected to all phases and slaves can be connected to relevant phases)

2. P-P Model: Phase to phase model. That version also gets power from terminal pins 1 and 2 from phase and neutral. Data transmission only done through terminal pins 3 and 4. Phase to phase connection can be done to data transmission pins for better performance.
If not used then phase and neutral can still be connected for data transmission for terminal pins 3 and 4.

☝️

NOTE2: BPL Model can be purchased in DC model as well:

This model will be same as "P-P Model"(Phase to phase model) on data connection and gets **12V DC power** from terminal pins 1 and 2. Data transmission only done through terminal pins 3 and 4.

6. Ethernet Cables

STG Series WMBus Gateways have standard Ethernet ports. According to the link type, the switches use CAT 3, 4, 5, 5e UTP cables to connect to any other network device (PCs, servers, switches, routers, or hubs).

6.1 Cable Type and Specifications

Cable	Type	Max. Length	Connector
10BASE-T	Cat. 3, 4, 5 100-ohm	UTP 100 m (328 ft)	RJ-45
100BASE-TX	Cat. 5 100-ohm UTP	UTP 100 m (328 ft)	RJ-45

6.2 ETH Cable Pin Assignments

With 100BASE-TX/10BASE-T cable, pins 1 - 2 are used for transmitting data and pins 3 - 6 are used for receiving data.

Pin Number	Description
1	TD+
2	TD-
3	RD+
4	Not Used
5	Not Used
6	RD-
7	Not Used
8	Not Used

7. System Comparison Between CAT5 and BPL Links

	CAT5 Based System	BPL Link Based System
Media	CAT5	Power Line
Bandwidth	100Mbps	Up to 30Mbps
Re-Wire	Yes	No, Using existing Power Line
Span	<100m	<600m
Multiple Nodes	N/A	Up to 10 hops/1000 nodes
Encryption	Yes, but difficult to configure	Yes, Plug & Play
Installment	Difficult	Easy, simply user power line
Installment Cost	High	Low
Total Cost	High	Low

☝

NOTE1 : BPL Model can be purchased in 2 versions:

1. P-N Model: Phase to neutral model (Standart Model). That version gets power from terminal pins 1 and 2 from phase and neutral. It can also transmit

data from that pins and other pins usage is optionAl (Ex: Master can be connected to all phases and slaves can be connected to relevant phases)

2. P-P Model: Phase to phase model. That version also gets power from terminal pins 1 and 2 from phase and neutral. Data transmission only done through terminal pins 3 and 4. Phase to phase connection can be done to data transmission pins for better performance.
If not used then phase and neutral can still be connected for data transmission for terminal pins 3 and 4.

☝️

NOTE2: BPL Model can be purchased in DC model as well:

This model will be same as "P-P Model"(Phase to phase model) on data connection and gets 12V DC power from terminal pins 1 and 2. Data transmission only done through terminal pins 3 and 4.

8. Serial Cables

STG Series WMBus Gateways have 1 x RS232 and 1 xRS485 port. Serial line can be connected other serial devices such as RTUs, PLCs, energy meters or any other field device.

8.1 RS232 Cable Pin Assignments
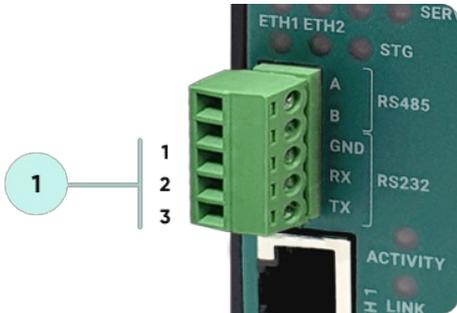
1. Terminal connector for 3 wire Tx-Rx-GND RS232 data transmission

Pin Number	Description
1	GND
2	Rx
3	Tx

8.2 RS485 Cable Pin Assignments
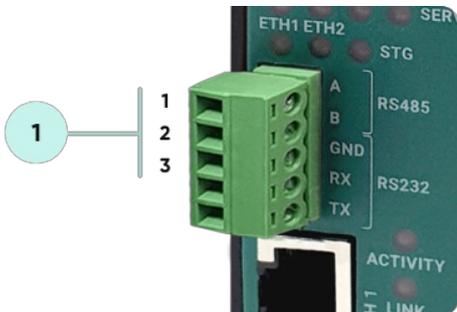
1. Terminal Connector for 2 wire RS485 connection and GND (if needed)

Pin Number	Description
1	A
2	B
3	GND (Suggested to use)

9. Usage Scenarios and Connection Diagrams

Some of the usage scenarios of STG Series WMBus Gateways are described below. Usages are not limited to that examples and user may create their own usage scenario.

9.1 Convert WMBus OMS Packages to Modbus TCP

STG Series WMBus Gateways can receive WMBus OMS packages and convert them to Modbus TCP packages. This way WMBus packages of many meters can directly be collected on a remote or local Data Server or RTU device as Modbus registers and can be used for other applications.

STG Configuration
- Server Mode
- Device Function: WMBus - Modbus TCP Gateway
- Wmbus Device Mode Other
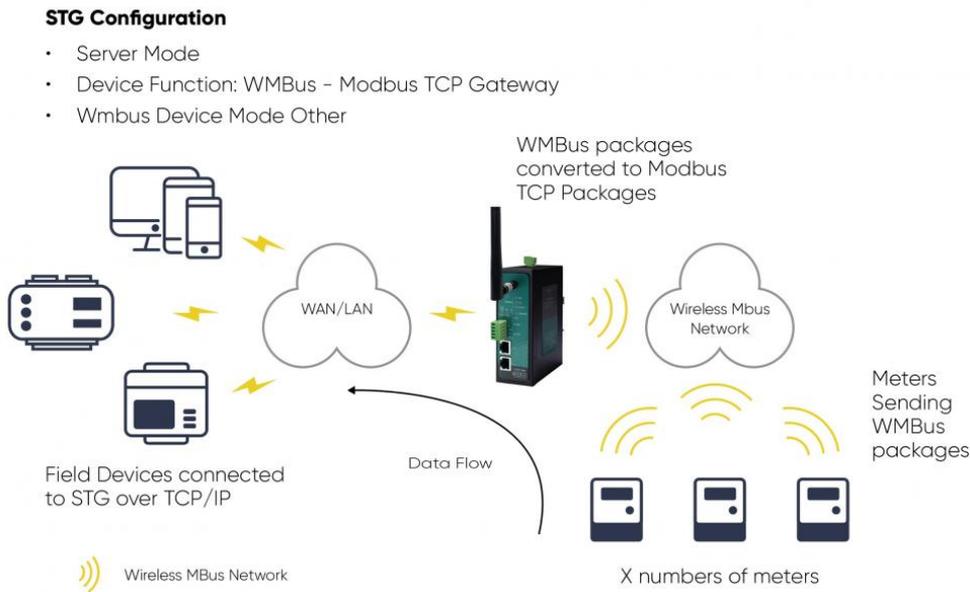
STG set in Server mode and Device function set as WMBus – Modbus TCP Gateway. Manufacturing IDs of field devices can be listed with related Modbus addresses. If needed decryption list can be set and White list can be enabled. Received WMBus OMS messages will be converted to Modbus Registers and field or remote Modbus RTU devices can query that data from related Modbus addresses.

9.2 Convert WMBus OMS Packages to Modbus RTU

STG Series WMBus Gateways can receive WMBus OMS packages and convert them to Modbus RTU packages. This way WMBus packages of many meters can directly be collected on a remote or local Data Server or RTU device as Modbus registers and can be used for other applications.

STG Configuration

- Client Mode
- Device Function: WMBus – Modbus RTU Gateway
- Wmbus Device Mode Other

WMBus packages
converted to Modbus
RTU Packages

Wireless Mbus
Network

Meters
Sending
WMBus
packages

Data Flow

—— RS485 Connection
—— RS232 Connection
))) Wireless MBus Network

X numbers of meters
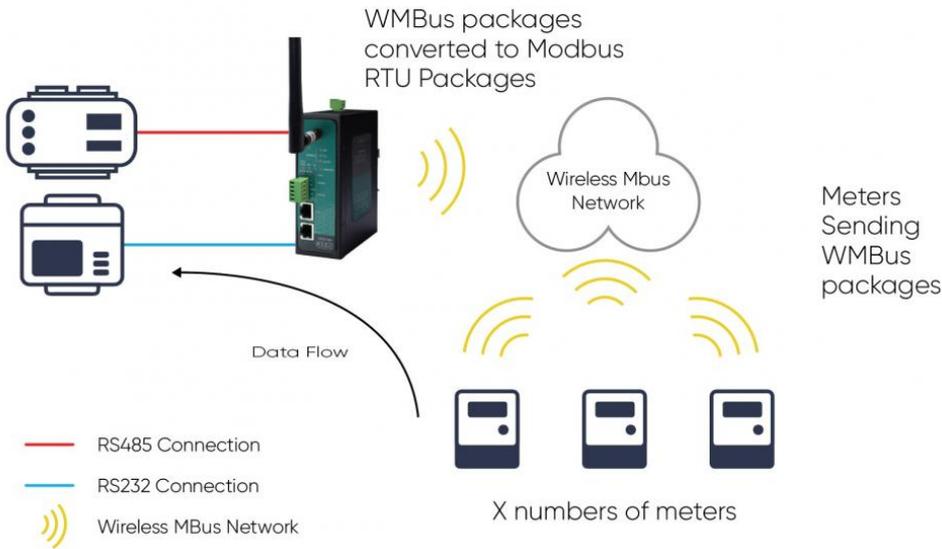
STG set in Client mode and Device function set as WMBus – Modbus RTU Gateway. Manufacturing IDs of field devices can be listed with related Modbus addresses. If needed decryption list can be set and White list can be enabled. Received WMBus OMS messages will be converted to Modbus Registers and field or remote Modbus RTU devices can query that data from related Modbus addresses.

9.3 Transmit Received and Parsed WMBus Packages to MQTT Server

STG Series WMBus Gateways can receive WMBUS packages, decrypt and parse. All parsed data can be sent to MQTT server for web based applications in several different formats.

STG Configuration

- Server Mode
- Device Function: WMBus – Modbus TCP Gateway and MQTT Publisher
- Wmbus Device Mode Other

WMBus packages
converted to Modbus
TCP Packages and
WMBus data prepared
for MQTT Server

MQTT based
Applications

WAN/LAN

Wireless Mbus
Network

Meters
Sending
WMBus
packages

MQTT Server

Data Flow

))) Wireless MBus Network

X numbers of meters
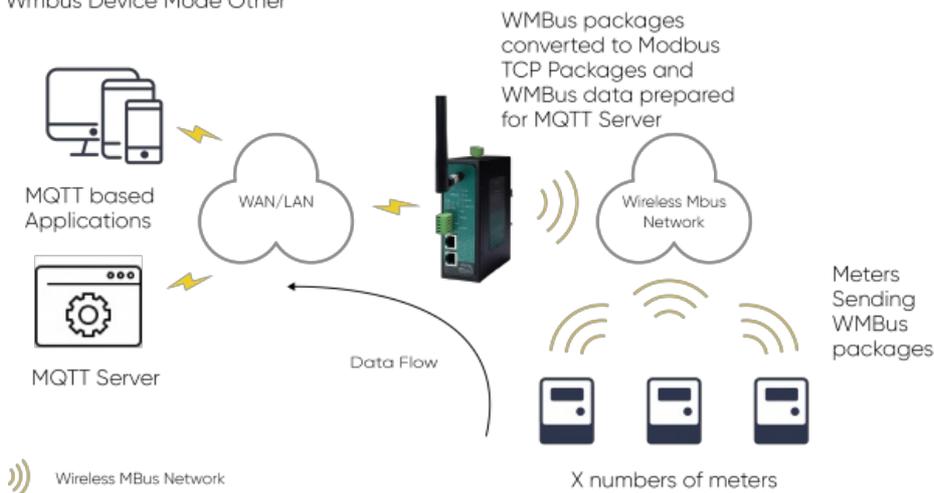
STG act as TCP Server to WMBus Gateway configured in Server Mode. Field devices still can connect to STG device and read WMBus packages in Modbus TCP protocol. STG will get WMBus frames, decrypt them, parse them serve as Modbus TCP packages if needed and send WMBus data to MQTT Server simultaneoulsy.

9.4 Transmit Received WMBus Packages to Remote or Local Server over TCP/IP

STG Series WMBus Gateways can receive WMBus packages and send them to remote or local server over TCP/IP link. This way WMBus packages of many meters can directly be collected on a remote or local Data Server and that data can be used for other applications.

WMBus data server with Static IP and Listening port

STG Configuration
- Client Mode
- Connect to NON STG Server IP and port
- Wmbus Device Mode Other
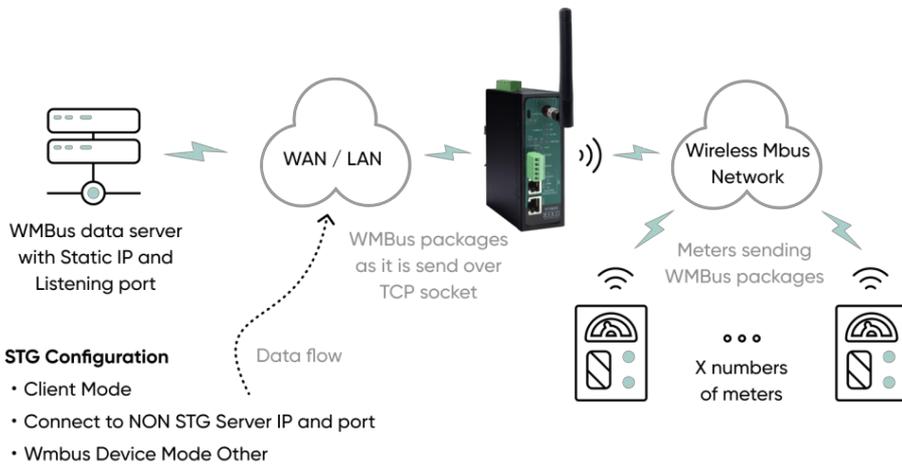
STG act as TCP Client to WMBus Gateway configured in Client Mode, Server device listening the connections and its IP and port must be configured in STG. WMBus configured to operate in Other Mode to listen WMBus packages and all received WMBus packages from the field devices will be transferred to remote server.

9.5 Transmit Received WMBus Packages to Local or Remote STG Clients over TCP/IP to Regenerate WMBus packages (Extend WMBus range)

STG Series WMBus Gateways can receive WMBUS packages and send them to remote or local clients over TCP/IP link. STG can act as server and up to 20 clients can connect. This way WMBus packages of meters can directly be shared with local or remote devices such as in-home displays.
STG can act as client and can regenerate the WMBus received over TCP/IP line. STG server can receive WMBus packages and send to STG client over TCP/IP network and WMBus packages can be regenerated on STG client side. This way WMBus range can be extended over TCP/IP network.

STG Server Mode Configuration
- Connections from STG and NON STG Clients are accepted at port 9760 with static IP for STG devices and 9770 for NON STG devices
- WMBus Device Mode Other

STG Client Mode Configuration
- Connect to Remote STG Server, Remote IP fixed or defined by Server if communicate in LAN
- WMBus Device Mode Meter
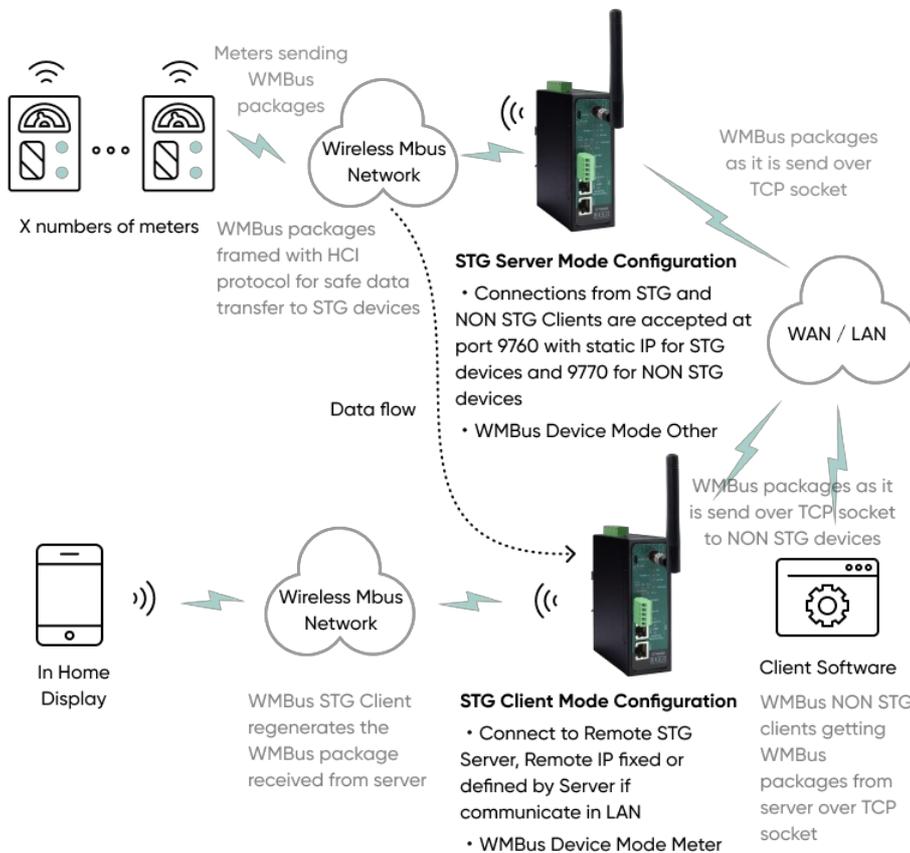
9.6 Transmit Received WMBus Packages to Local Serial Devices over RS232 or RS485

STG Series WMBus Gateways can receive WMBUS packages and send them to local serial devices over RS232 or RS485 serial connection. This way WMBus packages of many meters can directly be shared with local serial devices and that data can be used for other applications.
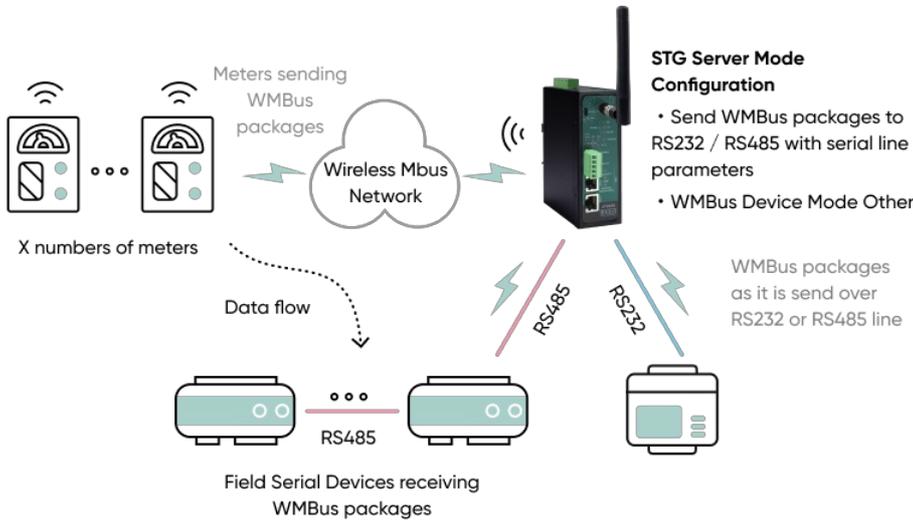
STG act as Serial to WMBus Gateway configured in Server Mode and listens WMBus packages. WMBus configured to operate in Other Mode to listen WMBus packages and all received WMBus packages from the field WMBus devices will be transferred to field serial devices.

9.7 Generate WMBus Packages for WMBus Devices over RS232 or RS485

STG Series WMBus Gateways can receive WMBUS messages from local serial devices over RS232 or RS485 serial connection and generate WMBus package based on those messages. This way WMBus packages for WMBus devices can directly be generated by local serial devices over RS232 or RS485.
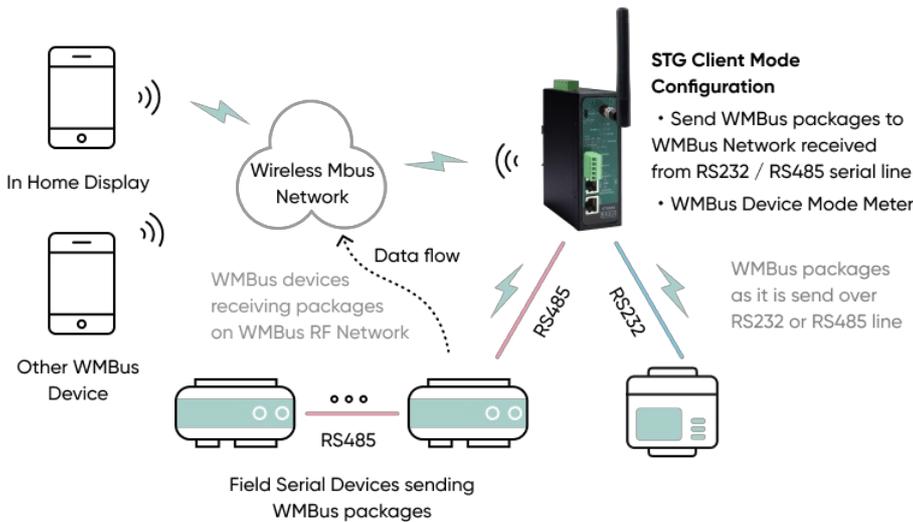
STG act as Serial to WMBus Gateway configured in Client Mode and listens serial packages. WMBus configured to operate in Meter Mode to be able to send WMBus packages and all received WMBus packages from the field serial devices will be transferred to field WMBus devices.

☝️

NOTE: Sent message must be a valid WMBus frame.

10. Configuration via WEB Interface

STG Series WMBus Gateways can be configured over web interface.
Device will get IP from DHCP client when connected to a network. User can use discovery tool to see IP of the device.

tcpip_discoverer.jar

Once the IP of the device is set, user may login the device by simply typing the Ip address of device.
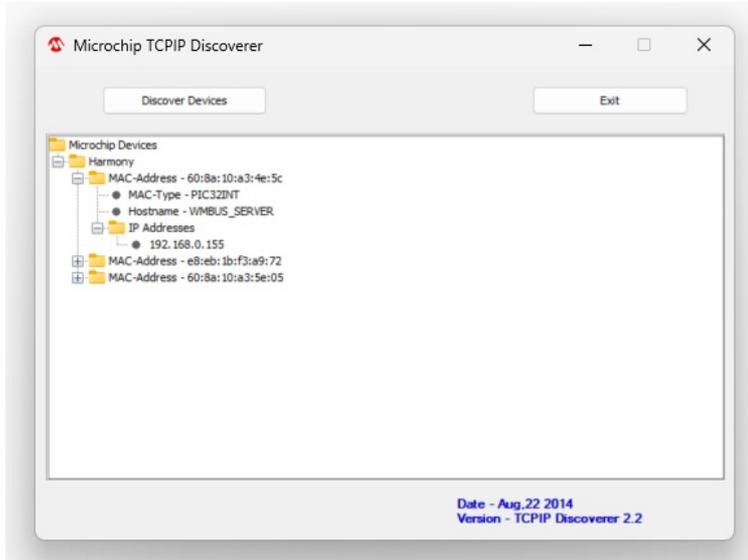
☝ NOTE 1: STG default firmware runs with DHCP off and expects an IP lease. If user need static IP or prefers DHCP on during start up, additional firmware is available.

☝ NOTE 2: If there is no DHCP server in LAN, REDZ device will get default 192.168.1.1 IP if it is set as Server Mode. It will get default 192.168.1.100 IP if it is set as Client mode.

10.1 Connecting Web Interface

Simply write IP of the device to the http client. Google Chrome is suggested to use. Login screen will pop up.
Default user name: **admin**
Default password: **admin**

Main screen of device will appear with following information:
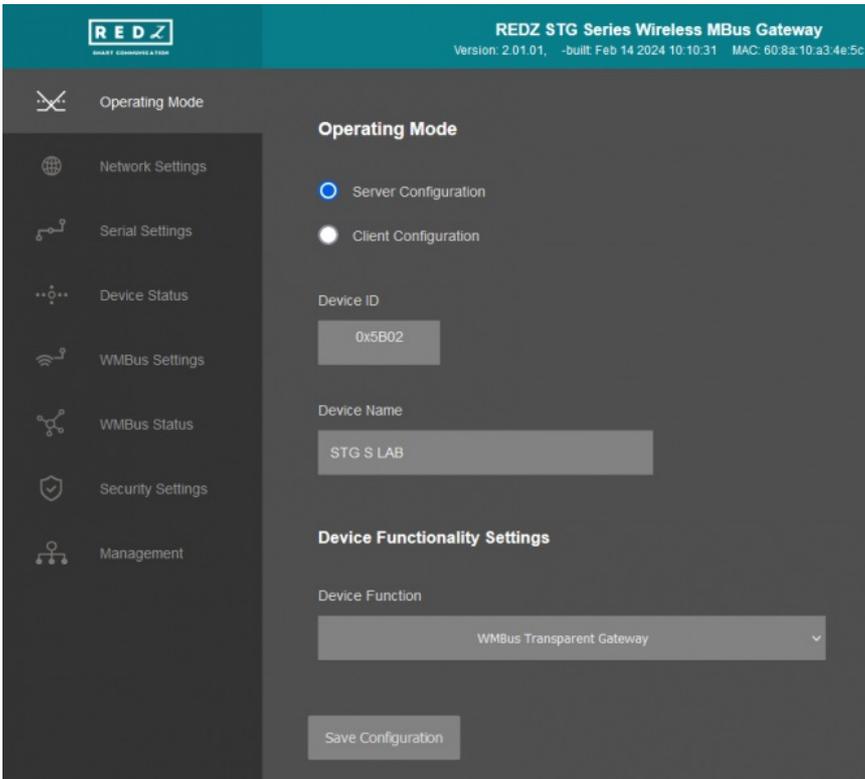
Firmware Info, MAC details and Device Name on top

Menu Items on left

Menu Item details in center

10.2 MENU: Operating Mode

From this menu user may select the operating mode and device function of the device.

STG has 3 Main Device Function:

1. WMBus OMS to Modbus TCP Converter (and/or send Meter Data to MQTT Server):

   Set "Server Configuration" and select Device Function "WMBus – Modbus TCP Gateway (and MQTT Publisher)"
   to get Wireless MBus frames, decrypt and parse them and read data as Modbus TCP frames and/or send
   data to MQTT Server.

2. WMBus OMS to Modbus RTU Converter:

Set "Client Configuration" and select Device Function "WMBus – Modbus RTU Gateway" to get Wireless MBus frames, decrypt and parse them and read data as Modbus RTU frames.

3. Transparent Mode (Sends/Receive WMBus Packages to/from TCP/IP or Serial Side):

Set "Server Configuration" and select Device Function "WMBus Transparent Gateway" to get Wireless MBus frames and send data to TCP/IP client or serial device (RS232 or RS485). Set "Client Configuration" and select Device Function "WMBus Transparent Gateway" to get Wireless MBus frames and send data to TCP/IP server. In this mode STG can send frame as it is or still can decrypt if device is listed on "AES Decryption List" defined under "WMBus Settings" menu.

Also in Device Function "WMBus Transparent Gateway" user can change "WMBus reception Mode" under "WMBus Settings" menu and if receiver is closed, STG will send frames to WMBus RF network, thus STG can be used to create or recreate WMBus frames.
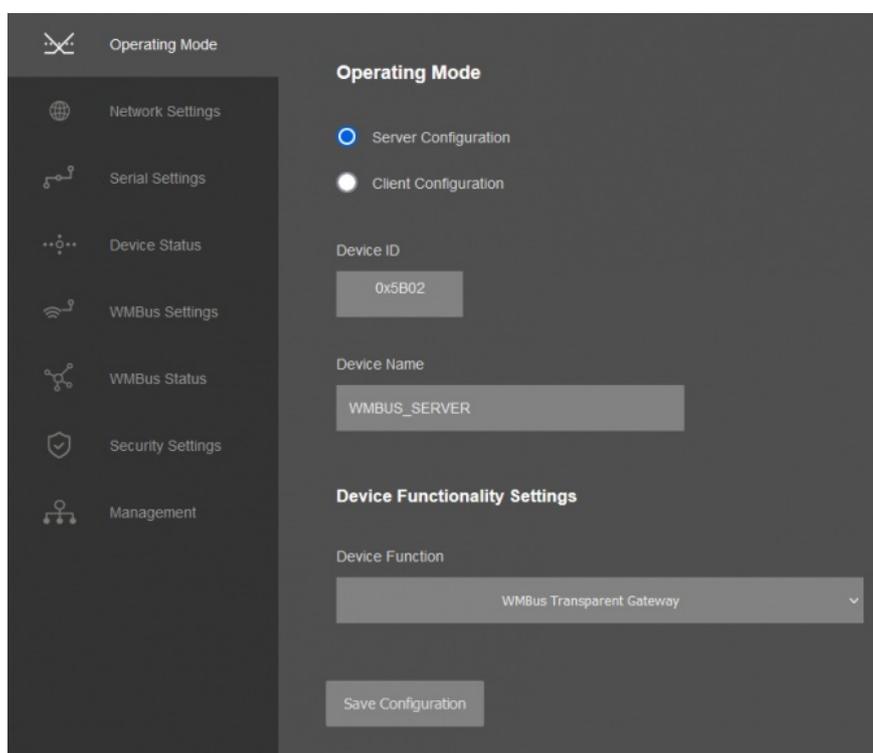
"Device ID" field is the unique Device ID of STG device itself, based on WMBus Module serial number.

"Device Name" field is used to identify device.

"Device Function" field is used to select device behaviour.

STG can act as WMBus to Modbus TCP Gateway and make conversion between WMBus and Modbus TCP protocols and if enabled can also send WMBus data to MQTT Server. STG is set to Server Mode and "Device Function" is "WMBus – Modbus TCP Gateway (and MQTT Publisher)".

STG can act as WMBus to Modbus RTU Gateway and make conversion between WMBus and Modbus RTU protocols. STG is set to Client Mode and "Device Function" is "WMBus – Modbus RTU Gateway".

STG can act as WMBus Transparent Gateway and transfer data between WMBus RF Network and field TCP/IP or Serial (RS232 or RS485) devices. STG is set to Server Mode or client Mode based on TCP/IP connectivity need (do not matter for serial connection, Server Mode can be used) and "Device Function" is "WMBus Transparent Gateway". User can change "WMBus reception Mode" under "WMBus Settings" menu and if receiver is closed, STG will send frames to WMBus RF network, if opened then STG will receive frames from WMBus RF Network..

If Device Function is set as "WMBus – Modbus TCP Gateway (and MQTT Publisher)" or "WMBus – Modbus RTU Gateway", following settings will be shown.

Users may enter Modbus Address that they want to use with field device in this List. They can match the Modbus Address with field WMBus device Manufacturing ID and they can enable the entry they use.

"Modbus Address" Modbus Address to be used to Access Modbus registers for field WMBus Device

"WMBus Device Address" Manufacturing ID of field WMBus Device that matched with Modbus Address

"Enable" Option to enable or disable the entry

☝

NOTE: STG Series WMBus Gateways have built in mechanism to get WMBus frames and Auto Configure which is described in "WMBus Settings" menu.
This list can be auto filled by Auto Configuration mechanism, thus user do not have to enter manually.

STG has capability to decrypt and convert WMBus data to Modbus protocol for up to 40 devices in list

There is also a version with up to 20 devices for Modbus conversion and up to 16 devices for decryption.

Once the setting has been changed, "Save Configuration" button will be enabled.
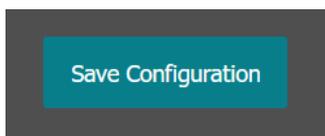
Save Configuration

After clicking button system will tell if the settings applied successfully or not.

Operating Mode Changed.

OK

☝

NOTE 1: STG Series WMBus Gateways can keep configuration of 2 different modes in its memory and once the configuration enabled, its already saved settings will be applied. Device can act as Server or Client at a time.

☝

NOTE 2: Settings will be applied once the device is rebooted from web interface or repowered manually.

☝
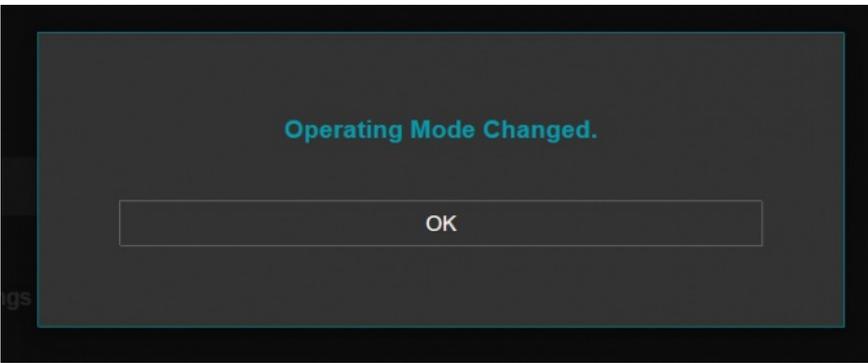
NOTE 3: If STG will be used to generate WMBus packages and TCP/IP connection will be used for that, there are 2 options:

1. "Device Function" is set to "WMBus Transparent Gateway" in "Server configuration" and serial line is not enabled.

2. "Device Function" is set to "WMBus Transparent Gateway" in "Client configuration" and serial line is not enabled.

STG will act as WMBus to TCP/IP Transparent Gateway and can send data from WMBus to TCP/IP or send TCP/IP data to WMBus RF Network (generate WMBus frames)

STG has 2 versions as hardware as well:

STG has capability to decrypt and convert WMBus data to Modbus protocol for up to 40 devices in list. This version can receive WMBus frame from TCP/IP as it is and send to WMBus RF Network.

There is also a version with up to 20 devices for Modbus conversion and up to 16 devices for decryption. If that version is used, HCI protocol must be used to generate WMBus frames.

HCI protocol:

SOF	Msg Header Field	Payload Field	FCS (optional)
8 Bit	24 Bit	n * 8 Bit	16 Bit

SOF (Start of Frame): 0xA5
Message Header Field:

Control Field	Endpoint ID	Msg ID Field	Length Field
4 Bit	4 Bit	8 Bit	8 Bit

Control Field: User may use 0x0 for this part to ignore CRC16 attachment at

end
Endpoint ID: It is 0x2
Message ID Field: It is 0x01 for sending data
Length Field: It is the payload length which means data size of framed
WMBus package

Example WMBus Message:

Frame in Hex
0E 06 23 12 67 45 23 01 01 01 80 54 65 73 74

Example encapsulated WMBus Message with HCI protocol:

Frame in Hex
A5 02 01 0E 06 23 12 67 45 23 01 01 01 80 54 65 73 74

Please contact our company if you need full implementation of protocol
including CRC16 calculation.

10.3 MENU: Network Settings – Server

From this menu user may change the network settings of the device.

- User can change NTP Server address that used in MQTT data transmission.
- User can activate MQTT Publisher mode and change settings for MQTT data tranmission.
- User can change TCP Listening port for field devices.
- User can select maximum number of clients allowed to connect device.
- User can activate DHCP server.
- User can force device to a static IP.
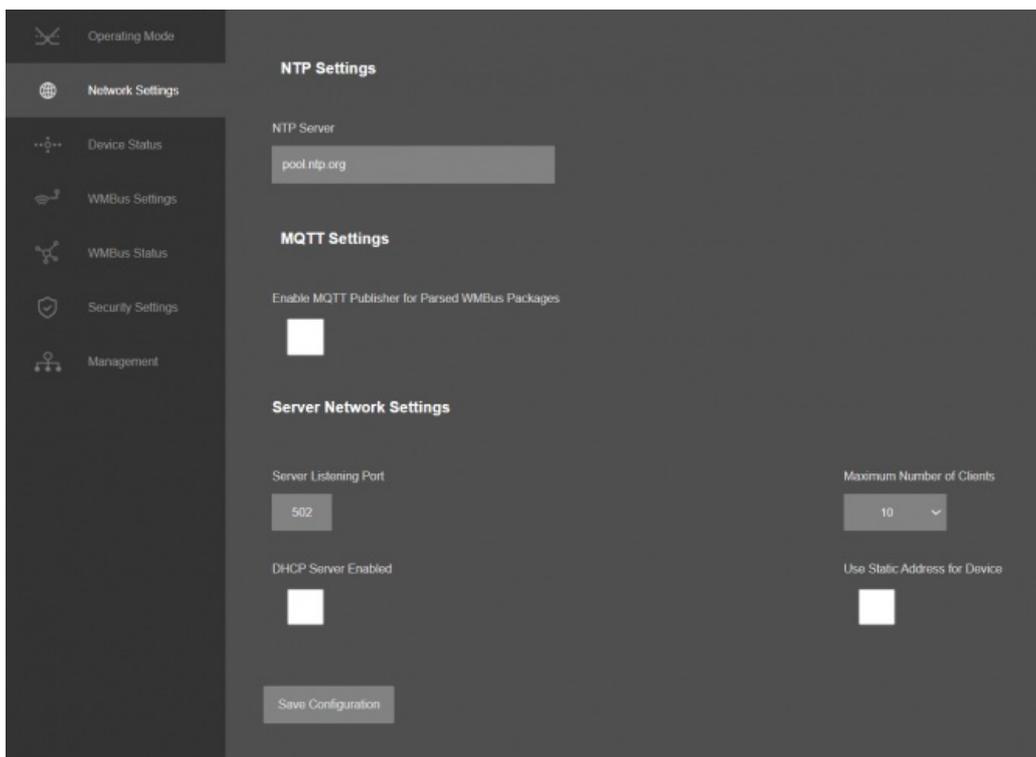
"NTP Server": NTP Server address that used in MQTT data transmission.

"Enable MQTT Publisher for Parsed WMBus Packages": Click to enable MQTT Publisher. STG will send data read from WMBus meters for enabled devices (there are up to 40 available) to MQTT Server. Data can aonly be sent if the WMBus frame can be (decrypted and) parsed. This Option is not availble if "Device Function" is set to "WMBus Transparent Gateway".

If "Enable MQTT Publisher for Parsed WMBus Packages" is checked, following settings will be shown.

"MQTT Broker IP": TCP IP of the MQTT Server. User must enter IP value
Ex: 75.2.83.130 is for "https://tago.io/" web address

"MQTT Broker Port": TCP Port of the MQTT Server.
Ex: 1883 is for "https://tago.io/" web address

"Client ID": MQTT Publisher client ID. Default is MQTT_STG_Client.
Maximum length for this field is 32.

"User Name": MQTT Publisher user name. This must be entered based on MQTT server settings.
Maximum length for this field is 64.

"Password": MQTT Publisher password. This must be entered based on MQTT server settings.
Maximum length for this field is 48.

"Publish Topic": MQTT Publisher topic value. Default is MQTT_STG_Publish_topic.
Maximum length for this field is 32.

"Subscribe Topic": MQTT Publisher subscribe topic value. Default is MQTT_STG_Subscribe_topic.
Maximum length for this field is 32.

"Minimum Send Interval for a Parsed Data (in seconds)": Minimum value to send meter data to MQTT Server. This time is the minimum time to send data to server, if WMBus meter sends frames in longer periods then it will be basis of the sending interval.

"Data Format": Options for how data is shared by STG with MQTT server. There are 2 options:

RAW WMBUS DECRYPTED DATA

PARSED DATA AS OBJECTS

PARSED DATA AS MODBUS FRAME

When selected as "Raw WMBus Decrypted Data", STG will share WMBus frame as it is but decrypted as follows as an example:

13:58:03:

[MQTT] Device publish

{ "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710413878\",\"Data\":
{\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-
51\",\"AcNu\":\"45\",\"RawDt\":\"202D900F002C2540430000CF464FAE2BBFB78B7239471400952642032D181007102F2F04953A6F4400002F2F2F2…
"qos": 0, "device": "65e076784d9b150010876950", "messageId": null, "isHex": false }

{ "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710413878\",\"Data\":
{\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-
51\",\"AcNu\":\"45\",\"RawDt\":\"202D900F002C2540430000CF464FAE2BBFB78B7239471400952642032D181007102F2F04953A6F4400002F2F2F2F2F2F\"}}", "qos": 0, "device":
"65e076784d9b150010876950", "messageId": null, "isHex": false }

MQTT message shows STG device name, device id, NTP time and data itself

Data has values of WMBus Device Id, Manufacturer Id, Device Version, Device Type, RSSI Value of recevied signal, Access Number value and Decrypted frame Data Block.

<u>When selected as "Parsed Data As Objects", STG will share WMBus data as parsed objects as follows as an example:</u>

14:04:57:

[MQTT] Device publish

{ "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710414288\",\"Data\":
{\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-51\",\"AcNu\":\"45\",\"DtCnt\":\"1\",\"Blcks\":
[{\"S\":\"0\",\"F\":\"1\",\"T\":\"3\",\"V\":\"1751900.0\"}]}}", "qos": 0, "device": "65e076784d9b150010876950", "messageId": null, "isHex": false }

{ "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710414288\",\"Data\":
{\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-51\",\"AcNu\":\"45\",\"DtCnt\":\"1\",\"Blcks\":[{\"S\":\"0\",\"F\":\"1\",\"T\":\"3\",\"V\":\"1751900.0\"}]}}", "qos": 0,
"device": "65e076784d9b150010876950", "messageId": null, "isHex": false }

MQTT message shows STG device name, device id, NTP time and data itself

Data has values of WMBus Device Id, Manufacturer Id, Device Version, Device Type, RSSI Value of recevied signal, Access Number value.

Parsed Data is shared in blocks and Data Count value shows how many blocks are there.

In this example there is 1 block

{\"S\":\"0\",\"F\":\"1\",\"T\":\"3\",\"V\":\"1751900.0\"}

Here are explanations:

S: Storage Number (of data from meter)

F: Function Field (of data from meter)

T: Data Type (STG uses this value to identify type of Data)

V: Data Value (can be float or UINT - 4 byte data, matched with Data Type)

☝️

NOTE: Parsed data shared as data blocks. This is valid for both Modbus Frames and web page visualization. Each data block has following values:

– Data Type specified by STG

– Data Value (can be a float or long value)

– Function, available values are:

  INSTANTANEOUS, MAXIMUM, MINIMUM or ERROR STATE

– Data Storage Number as an integer

details are explained in "Chapter 11: WMBus Data Parsing Format".

<u>When selected as "Parsed Data As Modbus Frame", STG will share WMBus data as Modbus like frame as follows as an</u>

example:

14:22:03:
[MQTT] Device publish
{ "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710415316\",\"Data\":
{\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-
51\",\"AcNu\":\"45\",\"ModFr\":\"1A00144739269500420037000100000001000349D5DAE0002DFFCD\"}}", "qos": 0, "device":
"65e076784d9b150010876950", "messageId": null, "isHex": false }
    { "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710415316\",\"Data\":
    {\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-51\",\"AcNu\":\"45\",\"ModFr\":\"1A00144739269500420037000100000001000349D5DAE0002DFFCD\"}}",
    "qos": 0, "device": "65e076784d9b150010876950", "messageId": null, "isHex": false }

MQTT message shows STG device name, device id, NTP time and data itself

Data has values of WMBus Device Id, Manufacturer Id, Device Version, Device Type, RSSI Value of recevied signal, Access Number value.

Parsed Data is shared in Modbus like frame, in this example as follows.

1A00144739269500420037000100000001000349D5DAE0002DFFCD

1A: 1 byte frame length, which is 1Ah and which is 26 in decimal and WMBus data shared as blocks

00144739: 4 bytes of WMBus device id

2695: 2 bytes of WMBus device manufacturer id

0042: 2 bytes of WMBus device version

0037: 2 bytes of WMBus device type

0001: 1 byte of Data count value. Data shared in blocks and this value shows how many blocks are there

00000001000349D5DAE0: Each data blocks has following values in 6 registers and 12 bytes

Storage Number (of data from meter) – 2 bytes

Function Field (of data from meter) – 2 bytes

Data Type (STG uses this value to identify type of Data that is shown in previous example) – 2 bytes, Data Type is Volume in this example

Data Value (float data based on Data Type in this example) – 4 bytes, 1751900.0 in this example

002D: 1 byte of Access Number value as unsigned integer. 45 in this example.

FFCD: 1 byte of RSSI value as signed integer. –51 in this example.

☝

NOTE: Parsed data shared as data blocks. This is valid for both Modbus Frames and web page visualization. Each data block has following values:

– Data Type specified by STG

– Data Value (can be a float or long value)

– Function, available values are:

  INSTANTANEOUS, MAXIMUM, MINIMUM or ERROR STATE

– Data Storage Number as an integer

details are explained in "Chapter 11: WMBus Data Parsing Format".

There is also 2 registers at the end of each frame (service registers) to show Access Number and RSSI values of frame.

Following parameters and static IP settings available for "Server Network Settings" part.

"Listening Port": TCP Port that STG uses for incoming connections. Remote devices can use STG IP and this port to connect to STG for Modbus TCP query. This Option is not availble if "Device Function" is set to "WMBus Transparent Gateway" and one of the serial line (RS232 or RS485) is activated.

"Maximum Number of Clients": Maximum numbers of incoming connections accepted. STG can accept up to 10 simultaneous connection and all devices can query Modbus TCP. This Option is not availble if "Device Function" is set to "WMBus Transparent Gateway" and one of the serial line (RS232 or RS485) is activated.

"Use Static Address for Device": Set a static TCP IP for STG from this part. Enable and enter network settings and STGwill be available to connect from this static IP locally or remotely (gateway must be set properly for remote WAN connection).

Following parameters available if "DHCP Server" setting is enabled. This is used if DHCP server is needed in network. STGcan distribute IP to field devices connected to it in this way.

Also if the device has Broadband Power Line (BPL) option:

> User can select operating mode of BPL either MASTER or NODE.

☝

NOTE: Standard firmware of REDZ BPL supports up to 10 hops and 1000 nodes. Only 1 device can be MASTER in same network.

Once the setting has been changed, "Save Configuration" button will be enabled.

After clicking button system will tell if the settings applied successfully or not.

👆

NOTE 1: STG Series WMBus Gateways can keep configuration of 2 different modes in its memory and once the configuration enabled, its already saved settings will be applied. Device can act as Server or Client at a time.

👆
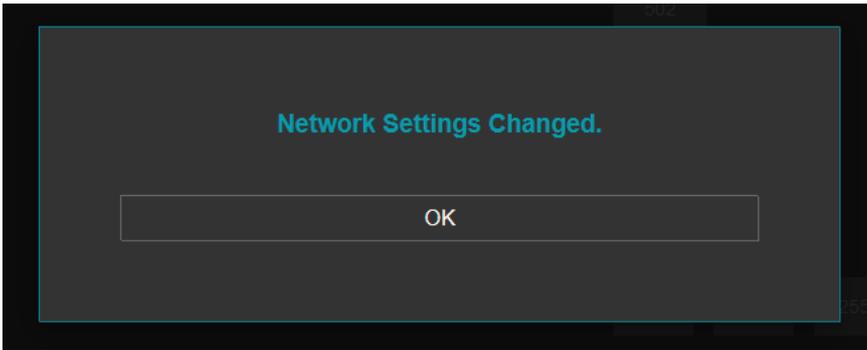
NOTE 2: Settings will be applied once the device is rebooted from web interface or repowered manually.

👆

NOTE 3: If "Device Function" is set to "WMBus Transparent Gateway" and one of the serial line (RS232 or RS485) is activated the device will act as Serial to WMBus Gateway and only following menu items will be available.
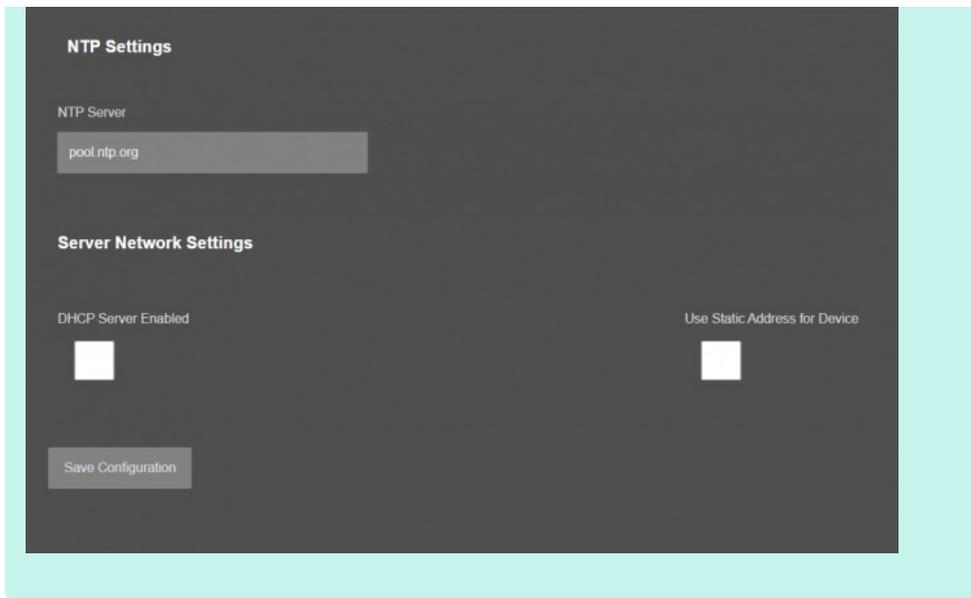
10.4 MENU: Network Settings – Client

From this menu user may change the network settings of the device.

- User can change NTP Server address.
- User can force device to a static IP.
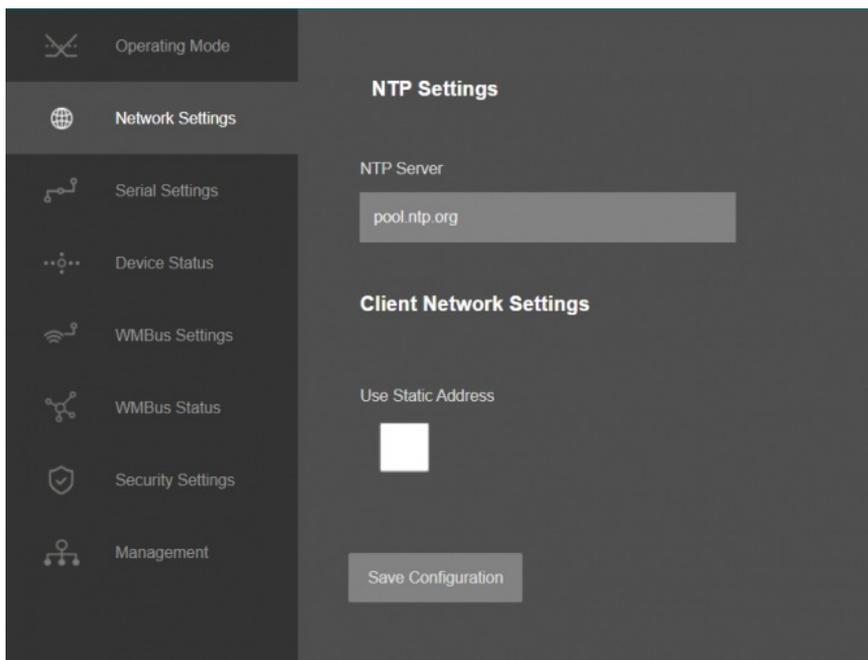
"Use Static Address": Set a static TCP IP for TLM from this part. Enable and enter network settings and TLM will be available to connect from this static IP locally or remotely (gateway must be set properly for remote WAN connection).

Following parameters and static IP settings available for "Use Static IP Address" setting.

Also if the device has Broadband Power Line (BPL) option,

> User can select operating mode of BPL either MASTER or NODE.

☝️

NOTE: Standard firmware of REDZ BPL supports up to 10 hops and 1000 nodes. Only 1 device can be MASTER in same network. If the device is in client mode, it is suggested to use "NODE" as setting.

Once the setting has been changed, "Save Configuration" button will be enabled.

After clicking button system will tell if the settings applied successfully or not.

☝️

NOTE 1: STG Series WMBus Gateways can keep configuration of 2 different modes in its memory and once the configuration enabled, its already saved settings will be applied. Device can act as Server or Client at a time.

☝️

☝️

10.5 MENU: Serial Settings

This menu is available only when

- "Device Function" is set to "WMBus Transparent Gateway" in "Server configuration" and one of the serial line is enabled (RS232 or RS485)
  STG can will as WMBus to Serial Gateway and can send data from WMBus to serial (RS232 or RS485) line or send serial data to WMBus RF Network (generate WMBus frames)

- "Device Function" is set to "WMBus Transparent Gateway" in "Client configuration" and one of the serial line is enabled (RS232 or RS485)
  Again STG will act as WMBus to Serial Gateway and can send data from WMBus to serial (RS232 or RS485) line or send serial data to WMBus RF Network (generate WMBus frames)

- "Device Function" is set to "WMBus –Modbus RTU Gateway" in "Client configuration".
  STG will act as WMBus to Modbus RTU Gateway and receive, decrypt WMBus frames and convert to Modbus RTU frames.

From this menu user may activate RS232 or RS485 connection.

- User can select RS232 line and can set baud rate and data type for serial line.

- User can select RS485 line and can set baud rate and data type for serial line.

Here are menu options:
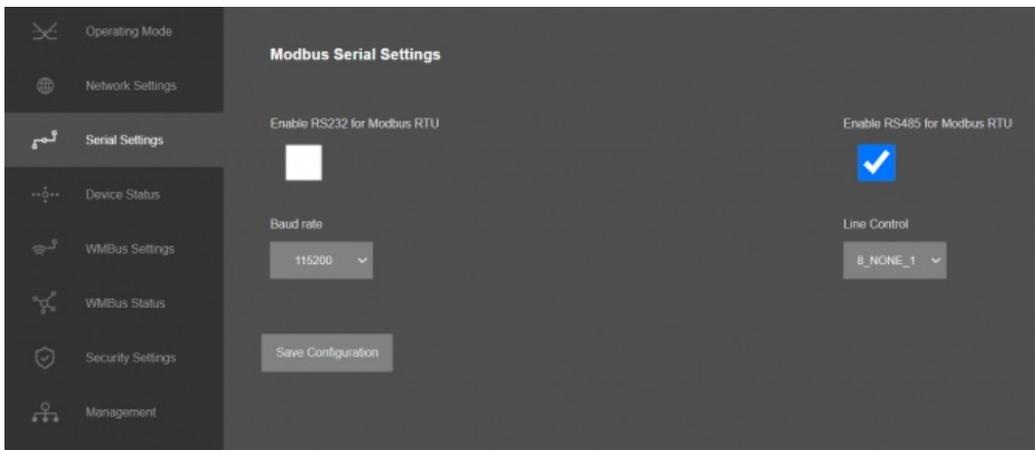
"Enable RS232 for Modbus RTU" Enable and set the baud rate and data type of RS232 serial line for Modbus RTU communication.
– If "Device Function" is set to "WMBus Transparent Gateway" and "Rx Package Reception Mode" is set to any of listening modes (S, T, C/T Modes) under "WMBus Settings", the text will be "Send WMBus Messages to RS232"
– If "Device Function" is set to "WMBus Transparent Gateway" and "Rx Package Reception Mode" is set to "OFF" under "WMBus Settings", text will be "Receive WMBus Messages from RS232"

"Enable RS485 for Modbus RTU" Enable and set the baud rate and data type of RS485 serial line for Modbus RTU communication.
– If "Device Function" is set to "WMBus Transparent Gateway" and "Rx Package Reception Mode" is set to any of listening modes (S, T, C/T Modes) under "WMBus Settings", the text will be "Send WMBus Messages to RS485"
– If "Device Function" is set to "WMBus Transparent Gateway" and "Rx Package Reception Mode" is set to "OFF" under "WMBus Settings", text will be "Receive WMBus Messages from RS485"

"Baud rate": Serial communication baud rate selection.

"Line Control": Serial communication data type selection in form of Data bits-Parity-Stop bits. Available options are:
8_NONE_1
9_NONE_1
8_EVEN_1
8_EVEN_2
8_ODD_1
8_ODD_2
8_NONE_2
9_NONE_2

Once the setting has been changed, "Save Configuration" button will be enabled.
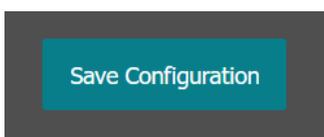
After clicking button system will tell if the settings applied successfully or not.

☝️
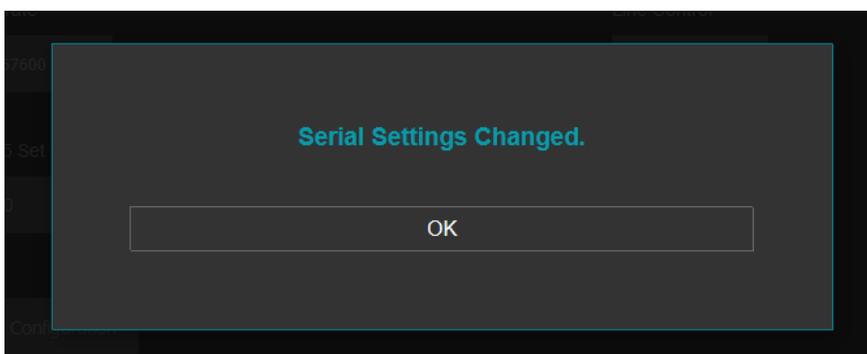
NOTE 1: STG Series WMBus Gateways can keep configuration of 2 different modes in its memory and once the configuration enabled, its already saved

settings will be applied. Device can act as Server or Client at a time. Lets say TCP Server enabled in Server operating mode and RS232 serial line enabled in Client operating mode on same device, the device can switch between to settings simply by changing the mode.

☝️

NOTE 2: Settings will be applied once the device is rebooted from web interface or repowered manually.

☝️

NOTE 3: If STG will be used to generate WMBus packages and one of serial line (RS232 or RS485) will be used for that, there are 2 options:

1. "Device Function" is set to "WMBus Transparent Gateway" in "Server configuration" and one of the serial line is enabled (RS232 or RS485)

2. "Device Function" is set to "WMBus Transparent Gateway" in "Client configuration" and one of the serial line is enabled (RS232 or RS485)

STG will act as WMBus to Serial Transparent Gateway and can send data from WMBus to serial (RS232 or RS485) line or send serial data to WMBus RF Network (generate WMBus frames)

STG has 2 versions as hardware as well:

STG has capability to decrypt and convert WMBus data to Modbus protocol for up to 40 devices in list. This version can receive WMBus frame from one of serial line (RS232 or RS485) as it is and send to WMBus RF Network.

There is also a version with up to 20 devices for Modbus conversion and up to 16 devices for decryption. If that version is used, HCI protocol must be used to generate WMBus frames.

HCI protocol:

SOF	Msg Header Field	Payload Field	FCS (optional)
8 Bit	24 Bit	n * 8 Bit	16 Bit

SOF (Start of Frame): 0xA5
Message Header Field:

Control Field	Endpoint ID	Msg ID Field	Length Field
4 Bit	4 Bit	8 Bit	8 Bit

Control Field: User may use 0x0 for this part to ignore CRC16 attachment at end
Endpoint ID: It is 0x2
Message ID Field: It is 0x01 for sending data
Length Field: It is the payload length which means data size of framed WMBus package

Example WMBus Message:

Frame in Hex
0E 06 23 12 67 45 23 01 01 01 80 54 65 73 74

Example encapsulated WMBus Message with HCI protocol:

Frame in Hex

```
A5 02 01 0E 06 23 12 67 45 23 01 01 01
80 54 65 73 74
```

Please contact our company if you need full implementation of protocol including CRC16 calculation.

10.6 MENU: Device Status - Server

From this menu user may monitor device status and statistics based on operating mode of device. The page also helps users to check device health.

"IP Address": TCP/IP address of STG itself.

"App Status": It shows current status of STG application. "SERVING CONNECTIONS" means device is ready for normal operation.

"Log Status": It is only available when "Log" is enabled from "Management" menu and shows current status of STG

logging. "LOG_STATE_TRYWRITELOG" means normal operation.

"Log Counter": It is only available when "Log" is enabled from "Management" menu and shows how many log lines has been transfered till now.

"MQTT Sent Messages": It is only available when "MQTT" is enabled from "Network Settings" menu and shows how many MQTT messages has been transfered till now.

"MQTT Status": It is only available when "MQTT" is enabled from "Network Settings" menu and shows current status of LKM MQTT Publisher. "APP_MQTT_STATE_SUBSCRIBED" means MQTT publisher is ready for normal operation.

"N. of Module UART Errors": Number of module uart errors. Device will enter "Reboot State" if this number is above 20.

"N. of Module Incorrect Chars Received" Number of incorrect packages length that does not fit Protocol Data Unit that device uses.

"N. of Module UART Input Overflow Errors": Number of module input overflow errors.

"N. of Module UART CRC Errors" Number of CRC errors based on Protocol Data Unit that device uses.

"N. of Module Message Retries" Number of WMBUS packages tried to send again due to a previous failure.

"N. of TCP Disconnections": Number of TCP disconnections from STG. Not available if one of serial line is enabled.

"N. of TCP Output Full Errors": Number of TCP output full errors during trying to send data to TCP side. Device will enter "Reboot State" if this number is above 5. Not available if one of serial line is enabled.

"N. of Gateway Side Message Retries" Number of tries to send data to TCP/IP.

"N. of Gateway Side Input Overflow Errors": Number of input overflow errors on TCP/IP. Not available for transparent gateway mode.

"N. of Module Incorrect Chars Received" Number of incorrect packages length that does not fit Modbus Protocol Data Unit on TCP/IP side. Not available for transparent gateway mode.

"N. of WMBus Messages Received Correctly": Number of WMBus packages received successfully over the WMBus Network.

"N. of WMBus Messages Sent Correctly": Number of WMBus packages sent to WMBus Network successfully.

"N. of Modbus Messages Received Correctly": Number of Modbus data packages received from TCP side. Not available for transparent gateway mode.

"N. of Modbus Messages Sent Correctly": Number of Modbus data packages sent to TCP side based. Not available for transparent gateway mode.

In "Modbus TCP Status" part (Available when "Device Function" is Set to "WMBus - Modbus TCP Gateway (and MQTT Publisher)":

Device will show 40 lines in this part to show modbus communciation status of each WMBus device available for decryption and parsing.

Modbus TCP Status

Modbus Slave Address	WMBus Device ID	Last Query Time	Last Modbus Package	Status
01.	1	0x144739	27-09-2023 06:28:47 00 47 00 00 00 1D 01 03 1A 00 14 47 39 26 95 00 42 00 37 00 01 00 00 00 01 00 03 49 D5 DA E0 00 2D FF CD	READY
02.	2	0x19142190	N.A. N.A.	READY
03.	3	0x3874	N.A. N.A.	READY
04.	4	0xDCBFB	N.A. N.A.	READY
05.	5	0x24095	N.A. N.A.	READY
06.	6	0x0	N.A. N.A.	DISABLED
07.	7	0x0	N.A. N.A.	DISABLED
08.	8	0x0	N.A. N.A.	DISABLED
09.	9	0x0	N.A. N.A.	DISABLED
10.	10	0x0	N.A. N.A.	DISABLED
11.	11	0x0	N.A. N.A.	DISABLED
12.	12	0x0	N.A. N.A.	DISABLED
13.	13	0x0	N.A. N.A.	DISABLED
14.	14	0x0	N.A. N.A.	DISABLED
15.	15	0x0	N.A. N.A.	DISABLED
16.	16	0x0	N.A. N.A.	DISABLED
17.	17	0x0	N.A. N.A.	DISABLED
18.	18	0x0	N.A. N.A.	DISABLED
19.	19	0x0	N.A. N.A.	DISABLED
20.	20	0x0	N.A. N.A.	DISABLED
21.	21	0x0	N.A. N.A.	DISABLED
22.	22	0x0	N.A. N.A.	DISABLED
23.	23	0x0	N.A. N.A.	DISABLED
24.	24	0x0	N.A. N.A.	DISABLED
25.	25	0x0	N.A. N.A.	DISABLED
26.	26	0x0	N.A. N.A.	DISABLED
27.	27	0x0	N.A. N.A.	DISABLED
28.	28	0x0	N.A. N.A.	DISABLED
29.	29	0x0	N.A. N.A.	DISABLED
30.	30	0x0	N.A. N.A.	DISABLED
31.	31	0x0	N.A. N.A.	DISABLED
32.	32	0x0	N.A. N.A.	DISABLED
33.	33	0x0	N.A. N.A.	DISABLED
34.	34	0x0	N.A. N.A.	DISABLED
35.	35	0x0	N.A. N.A.	DISABLED
36.	36	0x0	N.A. N.A.	DISABLED
37.	37	0x0	N.A. N.A.	DISABLED
38.	38	0x0	N.A. N.A.	DISABLED
39.	39	0x0	N.A. N.A.	DISABLED
40.	40	0x0	N.A. N.A.	DISABLED

"Modbus Slave Address": Is the Modbus Slave address for the device in entry.

"WMBus Device ID": Is the device id of WMBus meter that is queried.

"Last Query Date": Is the last time the modbus query is executed.

"Last Modbus Package": Is the last modbus frame sent for the WMBus meter in entry.

"Status": Is the status for communication.

After clicking "Refresh Status" button, system will reload data only and will not reload page. Button will be disabled during reload for an instance. If timeout occurs during the reload, the button will be enabled again with warning of timeout. In normal operation reload of status data will be done immediately.

☝️

NOTE 1: Following screen will show up if "WMBus Transparent Gateway" is selected as "Device Function"

In "Connections" part :

Device can show up to 10 lines in this part based on "Maximum Number of Clients" setting under "Network Settings" Menu.

"Ip Address": Is the TCP IP address of client connected to STG.

"Incoming Port": Is the TCP Port of socket used by client connected to STG.

☝

NOTE 2: Following screen will show up if "WMBus Transparent Gateway" is selected as "Device Function" and if one of serial (RS232 or RS485) line is activated.

10.7 MENU: Device Status - Client

From this menu user may monitor device status and statistics based on operating mode of device. The page also helps users to check device health.

"IP Address": TCP/IP address of STG itself.

"App Status": It shows current status of STG application. "SERVING CONNECTIONS" means device is ready for normal operation.

"Log Status": It is only available when "Log" is enabled from "Management" menu and shows current status of STG logging. "LOG_STATE_TRYWRITELOG" means normal operation.

"Log Counter": It is only available when "Log" is enabled from "Management" menu and shows how many log lines has been transfered till now.

"N. of Module UART Errors": Number of module uart errors. Device will enter "Reboot State" if this number is above 20.

"N. of Module Incorrect Chars Received" Number of incorrect packages length that does not fit Protocol Data Unit that device uses.

"N. of Module UART Input Overflow Errors": Number of module input overflow errors.

"N. of Module UART CRC Errors" Number of CRC errors based on Protocol Data Unit that device uses.

"N. of Module Message Retries" Number of WMBUS packages tried to send again due to a previous failure.

"N. of Serial UART Errors": Number of serial uart errors. Device will enter "Reboot State" if this number is above 20. Not available for transparent gateway mode when serial line (RS232 or RS485) is not activated.

"N. of Gateway Side Incorrect Chars Received": Number of incorrect bytes recived on Serial line (RS232 or RS485) that does not fit Modbus protocol. Not available for transparent gateway mode.

"N. of Gateway Side Input Overflow Errors": Number of input overflow errors on Serial line (RS232 or RS485). Not available for transparent gateway mode.

"N. of Gateway Side CRC Errors": Number of CRC errors on Serial line (RS232 or RS485) that does not fit Modbus protocol. Not available for transparent gateway mode.

"N. of WMBus Messages Received Correctly": Number of WMBus packages received successfully over the WMBus Network.

"N. of WMBus Messages Sent Correctly": Number of WMBus packages sent to WMBus Network successfully.

"N. of Modbus Messages Received Correctly": Number of Modbus data packages received from Serial RS232 or RS485 line. Not available for transparent gateway mode.

"N. of Modbus Messages Sent Correctly": Number of Modbus data packages sent to Serial RS232 or RS485 line. Not available for transparent gateway mode.

In "Modbus RTU Status" part (Available when "Device Function" is Set to "WMBus – Modbus RTU Gateway":

Device will show 40 lines in this part to show modbus communciation status of each WMBus device available for decryption and parsing.

Modbus RTU Status

	Modbus Slave Address	WMBus Device ID	Last Query Time	Last Modbus Package	Status
01.	1	0x144739	N.A.	N.A.	READY
02.	2	0x19142190	N.A.	N.A.	READY
03.	3	0x3874	N.A.	N.A.	READY
04.	4	0xDCBFB	N.A.	N.A.	READY
05.	5	0x24095	N.A.	N.A.	READY
06.	6	0x0	N.A.	N.A.	DISABLED
07.	7	0x0	N.A.	N.A.	DISABLED
08.	8	0x0	N.A.	N.A.	DISABLED
09.	9	0x0	N.A.	N.A.	DISABLED
10.	10	0x0	N.A.	N.A.	DISABLED
11.	11	0x0	N.A.	N.A.	DISABLED
12.	12	0x0	N.A.	N.A.	DISABLED
13.	13	0x0	N.A.	N.A.	DISABLED
14.	14	0x0	N.A.	N.A.	DISABLED
15.	15	0x0	N.A.	N.A.	DISABLED
16.	16	0x0	N.A.	N.A.	DISABLED
17.	17	0x0	N.A.	N.A.	DISABLED
18.	18	0x0	N.A.	N.A.	DISABLED
19.	19	0x0	N.A.	N.A.	DISABLED
20.	20	0x0	N.A.	N.A.	DISABLED
21.	21	0x0	N.A.	N.A.	DISABLED
22.	22	0x0	N.A.	N.A.	DISABLED
23.	23	0x0	N.A.	N.A.	DISABLED
24.	24	0x0	N.A.	N.A.	DISABLED
25.	25	0x0	N.A.	N.A.	DISABLED
26.	26	0x0	N.A.	N.A.	DISABLED
27.	27	0x0	N.A.	N.A.	DISABLED
28.	28	0x0	N.A.	N.A.	DISABLED
29.	29	0x0	N.A.	N.A.	DISABLED
30.	30	0x0	N.A.	N.A.	DISABLED
31.	31	0x0	N.A.	N.A.	DISABLED
32.	32	0x0	N.A.	N.A.	DISABLED
33.	33	0x0	N.A.	N.A.	DISABLED
34.	34	0x0	N.A.	N.A.	DISABLED
35.	35	0x0	N.A.	N.A.	DISABLED
36.	36	0x0	N.A.	N.A.	DISABLED
37.	37	0x0	N.A.	N.A.	DISABLED
38.	38	0x0	N.A.	N.A.	DISABLED
39.	39	0x0	N.A.	N.A.	DISABLED
40.	40	0x0	N.A.	N.A.	DISABLED

"Modbus Slave Address": Is the Modbus Slave address for the device in entry.

"WMBus Device ID": Is the device id of WMBus meter that is queried.

"Last Query Date": Is the last time the modbus query is executed.

"Last Modbus Package": Is the last modbus frame sent for the WMBus meter in entry.

"Status": Is the status for communication.

After clicking "Refresh Status" button, system will reload data only and will not reload page. Button will be disabled during reload for an instance. If timeout occurs during the reload, the button will be enabled again with warning of timeout. In normal operation reload of status data will be done immediately.

☝

NOTE 1: Following screen will show up if "WMBus Transparent Gateway" is selected as "Device Function"

"N. of TCP Disconnections": Number of TCP disconnections from STG.

"N. of TCP Output Full Errors": Number of TCP output full errors during trying to send data to TCP side. Device will enter "Reboot State" if this number is above 5.

"N. of Gateway Side Message Retries" Number of tries to send data to TCP/IP.

Also an extra item will appear to show remote IP of the target TCP device.
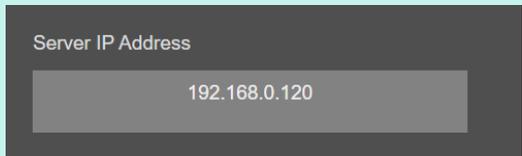
☝

NOTE 2: Following screen will show up if "WMBus Transparent Gateway" is selected as "Device Function" and if one of serial (RS232 or RS485) line is activated.
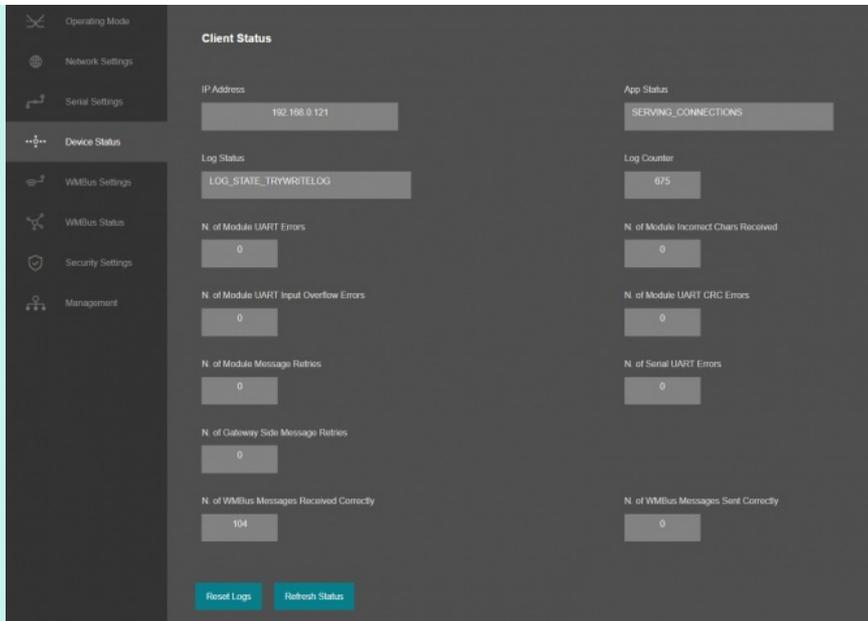
"N. of Gateway Side Message Retries" Number of tries to send data to Serial line (RS232 or RS485).

10.8 MENU: WMBus Settings

The Wireless MBus protocol stack implemented on STG is compliant the European standard 13757 part 4: "Communication systems for meters and remote reading of meters". It describes the wireless communication of water, heat, electricity and gas meters with data concentrators. For sake of convenience in this manual such meter devices are called "Meter", the communications partner devices like concentrators are called "Other".
From this menu user may change following WMBus RF parameters:

- "Rx Package Reception Mode": Determines if the module operates in Meter (only sends WMBus packages) or

  Other Mode (receives and sends WMBus packages).

  Device Mode is always Other Mode when device Function is set to Wmbus – Modbus Gateway.

  Available options are: S – Mode, T – Mode, C/T – Mode (combined reception of T and C Mode messages) and

  OFF

  OFF is used to turn of reception and set the device to Meter mode.

- "Tx Package Transmission Mode": Determines the message sending format of STG while sending WMBus

  messages, used only if reception mode is "OFF"

  Available options are: S – Mode Packet Format A, T – Mode, Packet Format A, C – Mode 50 000 bps Packet

  Format A, C – Mode 100 000 bps Packet Format A, C – Mode 50 000 bps Packet Format B, C – Mode 100 000 bps

  Packet Format B

- "Tx Power Level": Determines the power level of STG while sending WMBus messages, used only if reception mode

  is "OFF"

  Available options are from –1dBm to 13dBm

- "AES Decryption": The list of devcies for AES decryption. STG can decrypt up to 40 WMBus devices messages and

  in this list Manufacturer Id, Device Id, Device Version, Device Type and Decryption Key must be entered for each

  WMBus device that is needed to be read, decrypted, parsed and data converted to Modbus and/or sent to

  MQTT Server.

- "Auto Configure" option can be used to fill this list (besides the key) and other parts of device settings

automatically. When this button clicked, STG will show received frames with latest RSSI values for all WMBus devices since the device powered up. If there is no RSSI value, that means STG did not receive any frame for that entry yet, user may wait and click button again. If a line has notation "Recently Dedected" on the right side of entyr and framed with green line, that means this WMBus frame received and this device is not in configuration (or not in Black List), user may select it to "Add to Config" or "Add to Black List" based on needs (and click "Save Configuration" button). That process will also put the device id to Modbus list automatically thus ease up configuration process dramatically.

☝️

NOTE: STG Series WMBus Gateways have built in mechanism to get WMBus frames and Auto Configure. AES Decryption, Modbus Adress mapping and Black list items can be auto filled by Auto Configuration mechanism, thus user do not have to enter manually each item besides AES Key.

STG has capability to decrypt and convert WMBus data to Modbus protocol for up to 40 devices in list

There is also a version with up to 20 devices for Modbus conversion and up to 16 devices for decryption.

Supported Link modes are:

	Frequency	Coding	Chiprate	Bitrate	Frame Format

S - Mode	868.30 MHz	Manchester	32786 cps	16384 bps	A
T - Mode (Rx) (Meter to Other)	868.95 MHz	3-Out-Of-6	10000 cps	66666 bps	A
T - Mode (Tx) (Other to Meter)	868.30 MHz	Manchester	32768 cps	16394 bps	A
C - Mode (Rx) (Meter to Other)	868.95 MHz	NRZ	100000 cps	100000 bps	A, B
C - Mode (Tx) (Other to Meter)	869.525 MHz	NRZ	50000 cps	50000 bps	A, B
C/T - Mode (Rx) (Meter to Other)	868.95 MHz	NRZ	10000 cps	10000 bps	A, B
		3-Out-Of-6	10000 cps	66666 bps	A

Supported CI Values are:

CI-Field	Function / Layer	Up- or Down-link	TPL header - Type	Protocol / Service
53h	Application Reset or Select	Down	Long	Application Reset or Select
5Bh	Command	Down	Long	M-Bus
60h	Command	Down	Long	DLMS
6Ch	Time Sync	Down	Long	Generic
6Dh	Time Sync	Down	Long	Generic
6Eh	Application Error	Up	Short	Generic
6Fh	Application Error	Up	Long	Generic
72h	Response	Up	Long	M-Bus
74h	Alarm	Up	Short	Generic
75h	Alarm	Up	Long	Generic
78h	Response	Up	None	M-Bus
7Ah	Response	Up	Short	M-Bus
7Ch	Response	Up	Long	DLMS
7Dh	Response	Up	Short	DLMS
80h	Pure Transport Layer	Down	Long	None
8Ah	Pure Transport Layer	Up	Short	None
8Bh	Pure Transport Layer	Up	Long	None
8Ch	Extended Link Layer	Up / Down	Short	Lower Layer Service (2 Byte)
8Dh	Extended Link Layer	Up / Down	Long	Lower Layer Service (8 Byte)
8Eh	Extended Link Layer	Up / Down	Long	Lower Layer Service (10 Byte)
8Fh	Extended Link Layer	Up / Down	Long	Lower Layer Service (16 Byte)
C3h	Command	Down	Long	Security Information Transport
C4h	Response	Up	Short	Security Information Transport
C5h	Response	Up	Long	Security Information Transport

Packet Decryption and Encryption Support List is:

Mode	Encryption Type	Authentication	Supported
0	None	None	YES
2	DES CBC	None	NO
3	DES CBC	None	NO
5	AES-128 CBC	None, MIC	YES
7	AES-128 CBC, dynamic key	CMAC	YES
8	AES-128 CTR	CMAC	NO
9	AES-128 GCM	GCM/GMAC	NO
10	AES-128 CCM	CCM	NO
128	ELL AES-128 CTR	None, CRC	YES
129	Custom Modes	None	e.g. AT-WMBUS-NA-1

Here is an example for "AES Decryption" part: After clicking of "Refrefsh Auto Config List", STG will fill WMBus device address, Manufacturer ID, Device version and Device Type based on received frames.

In this example entries 1 to 5 are from previous configuration (already saved in configuration) and entries 6 & 7 are new frames and marked with "Recently Dedected" sign. User can mark those devices as "Add to Config" (and enter AES Key) or "Add to Black List".

If "Add to Config" checked, user must enter AES Key (if Key is used). After clicking "Save Configuration", STG will save this entry and add that device to "Modbus Scheduler Settings" under "Operating Mode" menu.

If "Add to Black List" checked, after clicking "Save Configuration", STG will save this device id under "Black List Table" (and enables list by checking "Black List Enabled" checkbox, if not enabled already) under "Security Settings" menu. This entry also will be freed up from decryption list to be used for new coming frames.

Once the setting has been changed, "Save Configuration" button will be enabled.
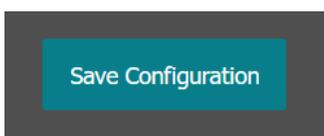
After clicking button system will tell if the settings applied successfully or not.

☝️
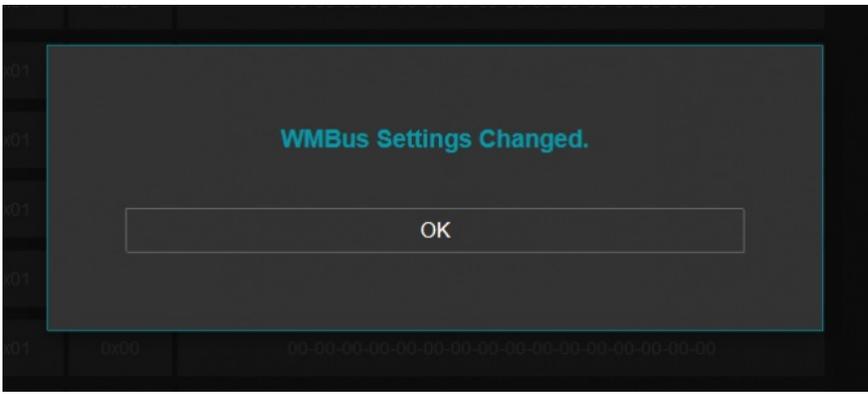
NOTE 1: STG Series WMBus Gateways can keep configuration of 2 different modes in its memory and once the configuration enabled, its already saved settings will be applied. Device can act as Server or Client at a time. This way different WMBus settings can be stored in 2 different operating modes.

☝️

NOTE 2: Settings will be applied once the device is rebooted from web interface or repowered manually.

☝️

NOTE 3:

STG has capability to decrypt and convert WMBus data to Modbus protocol for up to 40 devices in list

There is also a version with up to 20 devices for Modbus conversion and up to 16 devices for decryption. In that version, menu items are different:
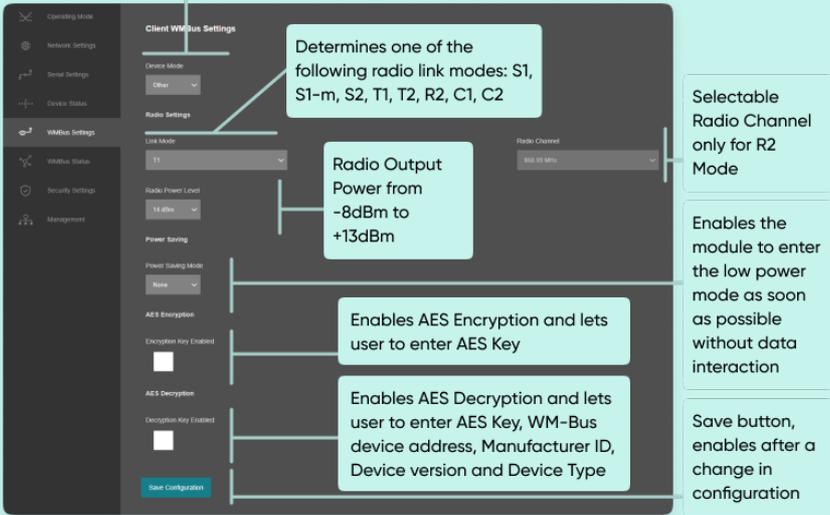
- "Device Mode": Determines if the module operates in Meter (only sends WMBus packages) or Other Mode (receives and sends WMBus packages). Device Mode is always Other Mode when device Function is set to Wmbus – Modbus Gateway

- "Link Mode": Determines one of the following radio link modes: S1, S1-m, S2, T1, T2, R2, C1, C2.

- "Radio Channel": Selectable Radio Channel only for R2 Mode.

- "AES Encryption": User can Enable or Disable Encryption and determine its key

- "AES Decryption": User can Enable or Disable Decryption and determine its details

If Enabled STG supports automatic AES-128 encryption and decryption of radio link messages.

There is the chance to configure up to 16 decryption keys for 16 different devices. The keys can be stored in a RAM table together with the complete WM-Bus device address, Manufacturer ID, Device version and Device Type.

Following table shows the RF channel setup. These channels are available in R-Mode for transmissions from Meter to Other devices. The opposite direction is always done in 868.33 MHz (channel 5).

Channel	Frequency (MHz)
1	868,09 MHz
2	868,15 MHz
3	868,21 MHz
4	868,27 MHz
5	868,33 MHz
6	868,39 MHz
7	868,45 MHz
8	868,51MHz

- "Radio Power Level": Radio Output Power from -8dBm to +13dBm

- "Power Saving Mode": Enables the module to enter the low power mode

  as soon as possible without data interaction.

This version of STG is a bidirectional radio device for the 868 MHz frequency band. STG is optimal suited for use in Smart Metering Applications, which are compliant to EN 13757 part 4.

This version of STG supports all link modes according to EN 13757-4: S (stationary), T (frequent transmit), R (frequent receive), C (compact operation). These four main modes are divided into further sub-modes for dedicated applications. All modes are available in web interface. Following table gives an overview over all WM-Bus modes and their physical parameters.

Mode	Direction	Data Rate	Coding	Frequency	Preamble + Synchronization
S1	Meter => Other	32,768 kcps	Manchester	868,3 MHz	582 chips
S1-m	Meter => Other	32,768 kcps	Manchester	868,3 MHz	56 chips
S2	Meter => Other	32,768 kcps	Manchester	868,3 MHz	56 chips
	Other => Meter	32,768 kcps	Manchester	868,3 MHz	56 chips
T1	Meter => Other	100 kcps	3 out of 6	868,95 MHz	56 chips
T2	Meter => Other	100 kcps	3 out of 6	868,95 MHz	56 chips
	Other => Meter	32,768 kcps	Manchester	868,3 MHz	56 chips
R2	Meter => Other	4,8 kcps	Manchester	868,03 MHz + n*60 kHz	104 chips
	Other => Meter	4,8 kcps	Manchester	868,33 MHz	104 chips
C1	Meter => Other	100 kcps	NRZ	868,95 MHz	64 chips
C2	Meter => Other	100 kcps	NRZ	868,95 MHz	64 chips
	Other => Meter	50 kcps	NRZ	868,525 MHz	64 chips

10.9 MENU: WMBus Status

From this menu user may monitor WMBus status and package details as well as WMBus OMS parsed data if available.

The page also helps users to see decryption status of received frames and check STG protocol specific and WMBus protocol specific parameters.

"System Time": Shows the system of STG.

The page has several parts.

"Latest Message Received – Module protocol fields":
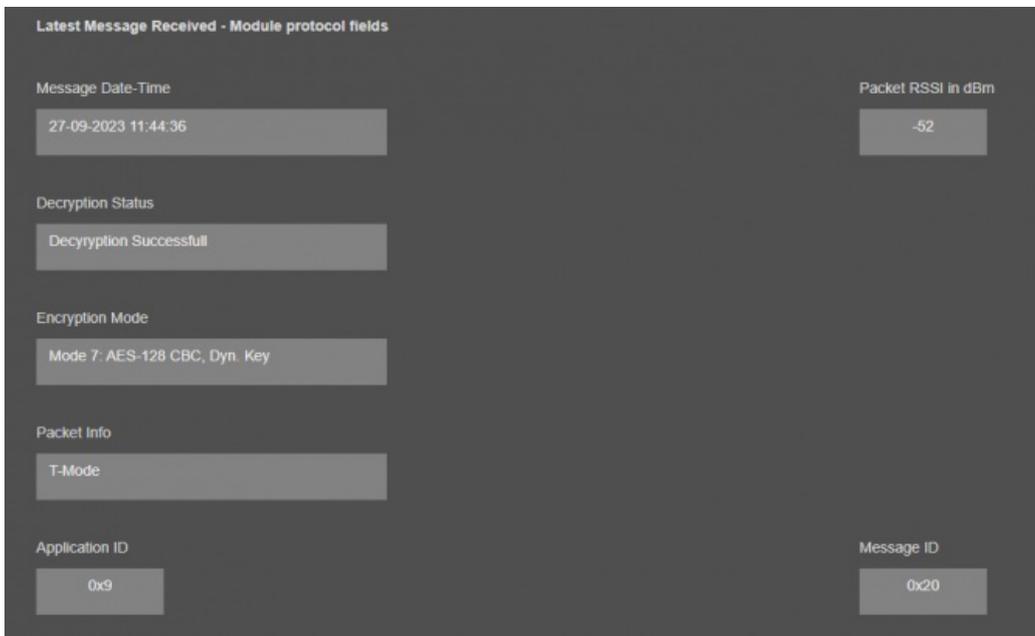
"Message Date-Time": Shows the system time when latest WMBus frame received.

"Packet RSSI in dBm": Received Signal Strenght Indicator (dBm) of last package received from WMBus side.

"Decryption Status": Decryption status of received WMBus frame. Available values are:

- "Not Encrypted"
- "Decyryption Successfull"
- "No Key Found for WM-Bus Address"
- "Decryption Mode Not Supported"
- "Decryption Error, Key Might Be Wrong"
- "UNKNOWN"

"Encryption Mode": Dedected encryption mode of received WMBus frame. Available values are:

- "None"
- "Mode 5: AES-128 CBC"
- "Mode 7: AES-128 CBC, Dyn. Key"
- "Mode 128: ELL AES-128 Counter Mode"
- "Mode 129: Custom Encryption Mode"
- "UNKNOWN: " (gives also number of encryption mode)

"Packet Info": Dedected packet type for received WMBus frame.Available values are:

- "S-Mode"
- "T-Mode"
- "C-Mode, 50.000 bps, Format A"
- "C-Mode, 100.000 bps, Format A"
- "C-Mode, 50.000 bps, Format B"
- "C-Mode, 100.000 bps, Format B"
- "UNKNOWN"

"Application ID": STG built in protocol specific parameter.

"Message ID": STG built in protocol specific parameter.

"Latest Message Received - WMBUS Link Layer fields":
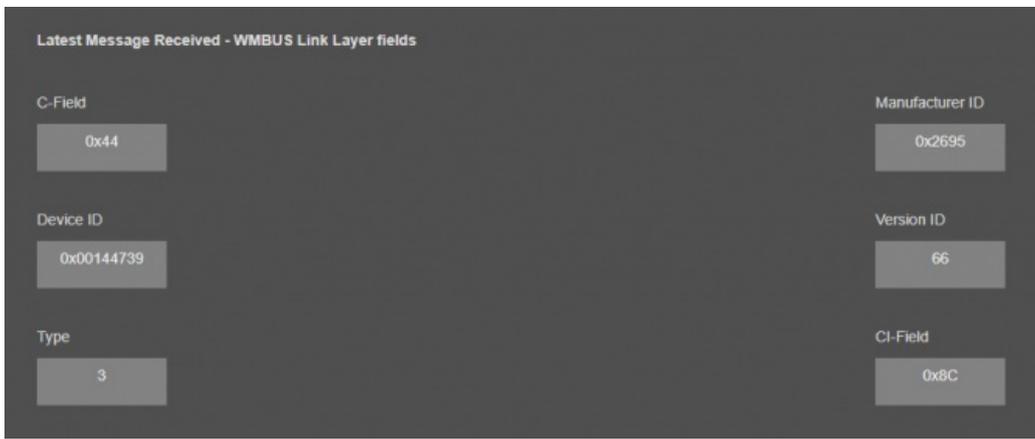
"C-Field": Control Field of WMBus Frame.

"Manufacturer ID": Manufacturer ID which is parsed from received WMBus Frame.

"Device ID": Device ID which is parsed from received WMBus Frame.

"Version ID": Device version which is parsed from received WMBus Frame.

"CI-Field": CI-Field of received WMBus Frame.

"Latest Message Received - Payload":

"Message Received Time": Shows the system time when this specific WMBus frame received.

"Payload" WMBus frame (if decryption is successfull it is also decrypted) after CI-Field part.

"Message Sent Time" Latestw WMBus package received time (based on STG system time)

"Poyload" Latest WMBus package payload in HEX format

"OMS Parsing" OMS parsed data (if available)

Parsed data shared as data blocks. This is valid for both Modbus Frames and web page visualization. Each data block has following values.

- Data Type specified by STG
- Data Value (can be a float or long value)
- Function, available values are:
  INSTANTANEOUS, MAXIMUM, MINIMUM or ERROR STATE
- Data Storage Number as an integer

details are explained in "Chapter 11: WMBus Data Parsing Format".

After clicking "Refresh WMBus Status" button, system will reload data only and will not reload page. Button will be disabled during reload for an instance. If timeout occurs during the reload, the button will be enabled again with warning of timeout. In normal operation reload of status data will be done immediately.

10.10 MENU: Security Settings

From this menu user may activate WMBUs RF communication package filter based on White list (accepted packages from Device ID field of WMBus packages) or Black list (rejected packages from Device ID field of WMBus packages).

Up to 40 entry can be defined for each filter list.

User can get ID of device from "Device Status" menu page.

- User can select to activate White List and STG device will accept WMBus packages only from the devices with Device ID stated in the list.
- User can select to activate Black List and STG device will accept all WMBus packages except from the devices with Device ID stated in the list.
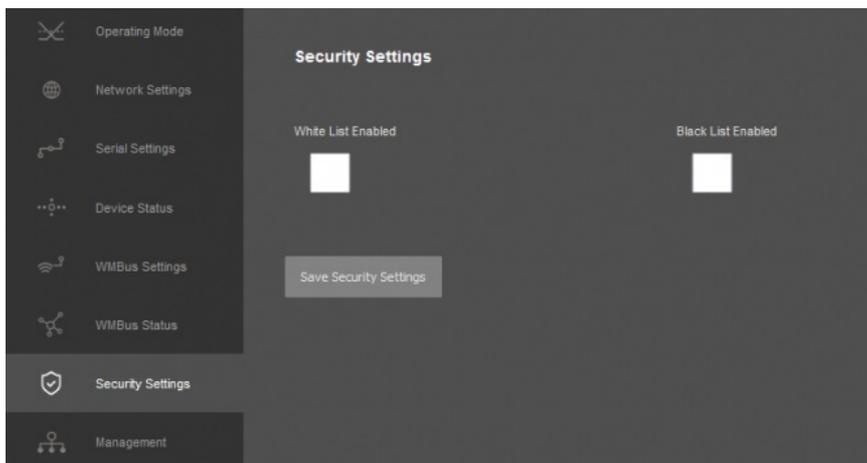
"While List Enabled" Enable to set the White List

"Black List Enabled" Enable to set the Black List

Following settings are available for any of the list:

White List Address Table			
	Filter Device ID		**Filter Device ID**
01.	0x00000000	02.	0x00000000
03.	0x00000000	04.	0x00000000
05.	0x00000000	06.	0x00000000
07.	0x00000000	08.	0x00000000
09.	0x00000000	10.	0x00000000
11.	0x00000000	12.	0x00000000
13.	0x00000000	14.	0x00000000
15.	0x00000000	16.	0x00000000
17.	0x00000000	18.	0x00000000
19.	0x00000000	20.	0x00000000

In this page user can enter decimal values and page will format it to hexadecimal automatically after cursor is moved out of scope.

02.	0x0000000C

Once the setting has been changed, "Save Configuration" button will be enabled.
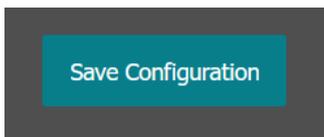
Save Configuration

After clicking button system will tell if the settings applied successfully or not.

Security Settings Changed.

OK

☝️
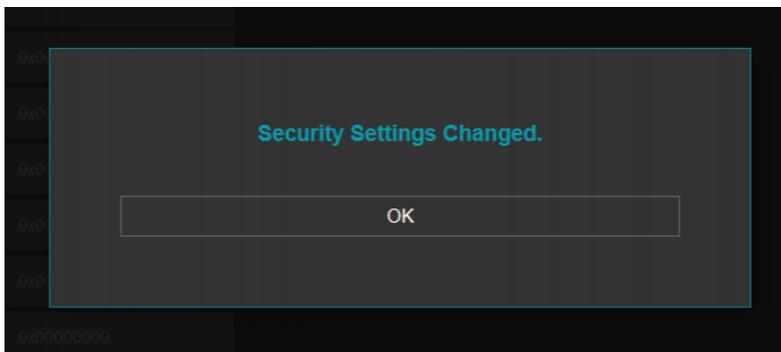
NOTE 1: STG Series WMBus Gateways can keep configuration of 2 different modes in its memory and once the configuration enabled, its already saved settings will be applied. Device can act as Server or Client at a time. That means users can configure different configuration settings for different operating modes and switch between each other in an instant.

☝️

NOTE 2: Settings will be applied once the device is rebooted from web interface or repowered manually.

10.11 MENU: Management

From this menu user may change parameters or send command to device.

- User can change system time. There is no battery inside the device so clock will be lost after a power down and will set to default.

> The device restarts itself every 86400 seconds. (Which means every 24 hours.)
> There are also timeout restart routines in Server mode during listening clients
> and in Client Mode trying to connect to the server. (Both preset to 10 minutes
> which means device will restart system if fails to connect a server in Client
> mode or a client do not connect in preset time in Server mode.)

- User can change firmware of device. REDZ STG WMbus Gateways has easy to use firmware capability and system can store 2 firmware at memory. After a firmware change, user can restore back old firmware anytime needed.

> After a firmware change old configuration will be used for minor changes. If a
> major change occurs system will restore to factory default configuration.

- User can change the login information.

- User can change the log level of the device. REDZ STG Series WMBus Gateways series has micro USB or USB Type-C and gives log in 115200 - 8N1 format.

  Any terminal program can be used to listen the LOG over USB type-C or micro USB port of the device which is recognized as Virtual COM port in PC.
  LOG to remote UDP server is also available. If set to UDP server, then TLM will send LOG data to remote UDP server device.

- User can restore to factory settings and force device to reboot. Factory settings restored for Client if the device in Client mode and factory settings are restored for Server if the device in Server mode.
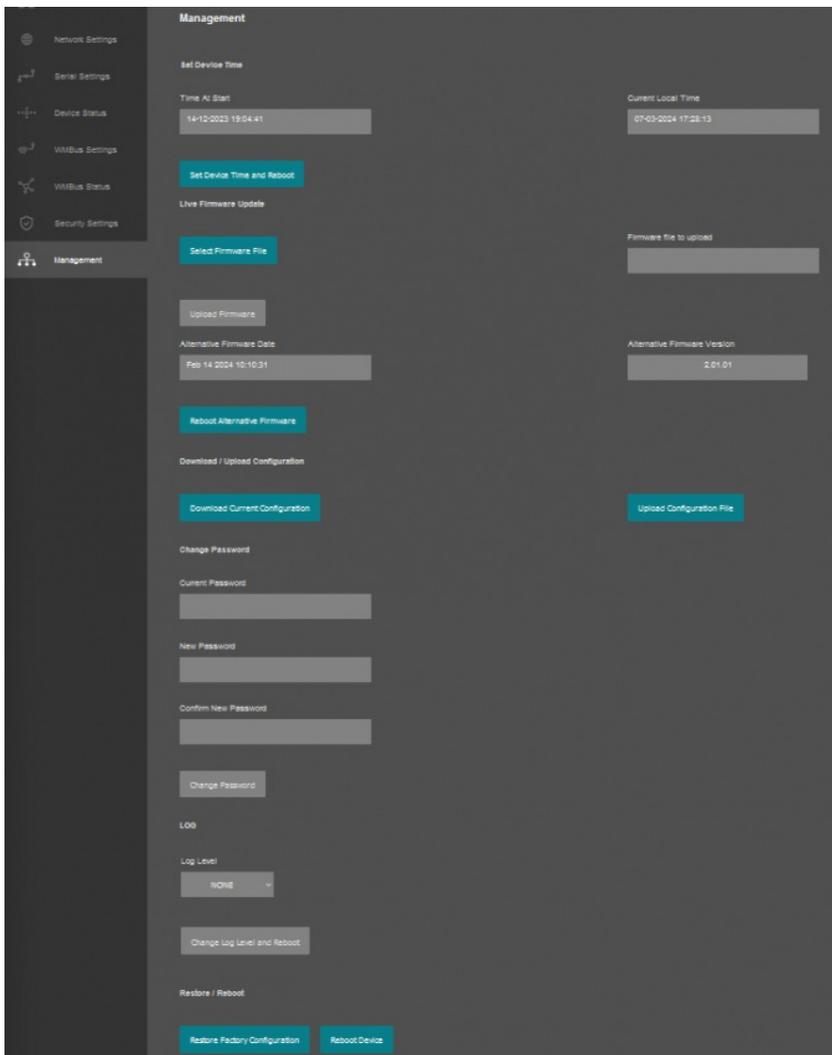
"Set Device Time" This part shows time at page load. Reload page to see updated time. Current local time shows the PC time. User can set the STG time based on shown PC time. The device sets the date and time and reboots.

In "Live Firmware Update" part:

Firmware upgrade is possible only with files that REDZ supplied. Once the file selected, TLM shows selected file:

Then "Upload Firmware" button must be clicked. STG will start to upload file and show status on pop up screen.
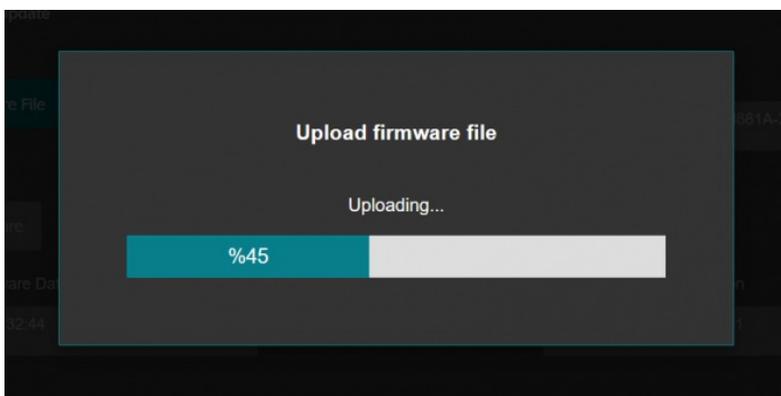
Click "Close" when finished. If somehow LKM fails to upload, refresh webpage and try again please.

After successful upload, TLM will show "Alternative Firmware Date" and "Alternative Firmware Version" data.

Click "Reboot Alternative Firmware" and STG will reboot with new firmware and show status on screen.
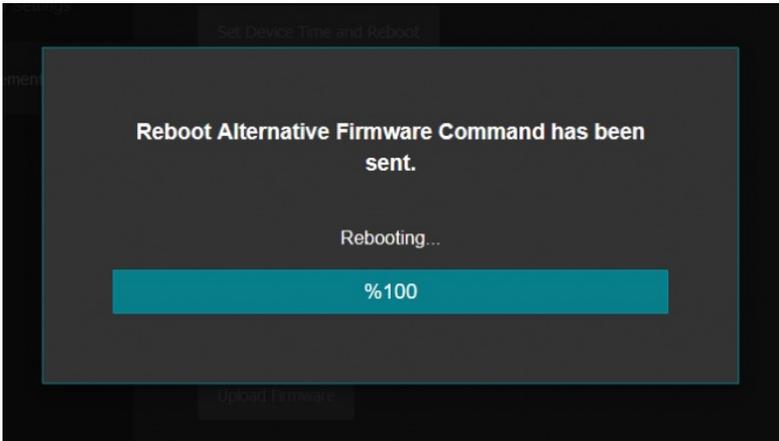
This will take 5 seconds only. Please wait .

Check firmware details from upper part of main screen please if the update firmware procedure finalized properly

☝ NOTE 1: User must refresh cache of their browser by clicking CTRL+F5 after a succesfull firmware change so that it will force browser to reload web interface (with latest updates/changes).

☝ NOTE 2: In major updates user must also reset device to factory settings.

In "Download / Upload Configuration" part:

User download current configuration of the device to a file or restore a previously defined configuration to device from file.

"Download Current Configuration": Downloads the configuration to a file. It uses "Device Name" for file name and the extensions will be "*.zcfg".
"Download Configuration File": Uploads the configuration from "*.zcfg" file.

In "Log" part:

User may activate Logging and see details of operation. There are different levels of Log with different amount of data.

"None": Logging is closed
"Error": Only errors in systems will be logged
"Info": General info and errors will be logged
"Debug": All details regarding device operation will be logged

If "Console" is selected as output of Log, then micro USB or USB Type-C port of device will be used for logging. Proper cable must be connected and a teminal should be used to receive Log data. As an example "RealTerm" tool can be used.

Simply select COM port and set baud rate 115200 and data type 8N1 and then click open. Device will send log data.
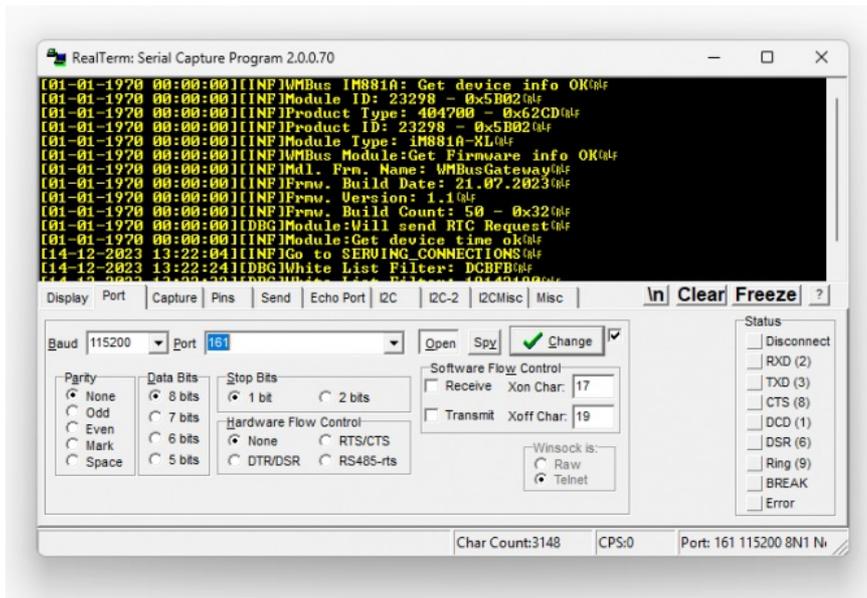
If "UDP Server" is selected as output of Log, then proper tool must be used to get log data. User must set "UDP Server IP" and "UDP Server Port". Device will send Log to that address. As an example "Package Sender" tool can be used.

Click "File" and then "Settings". Enable "UDP Server" and set the port. Device will send Log data to UDP server.
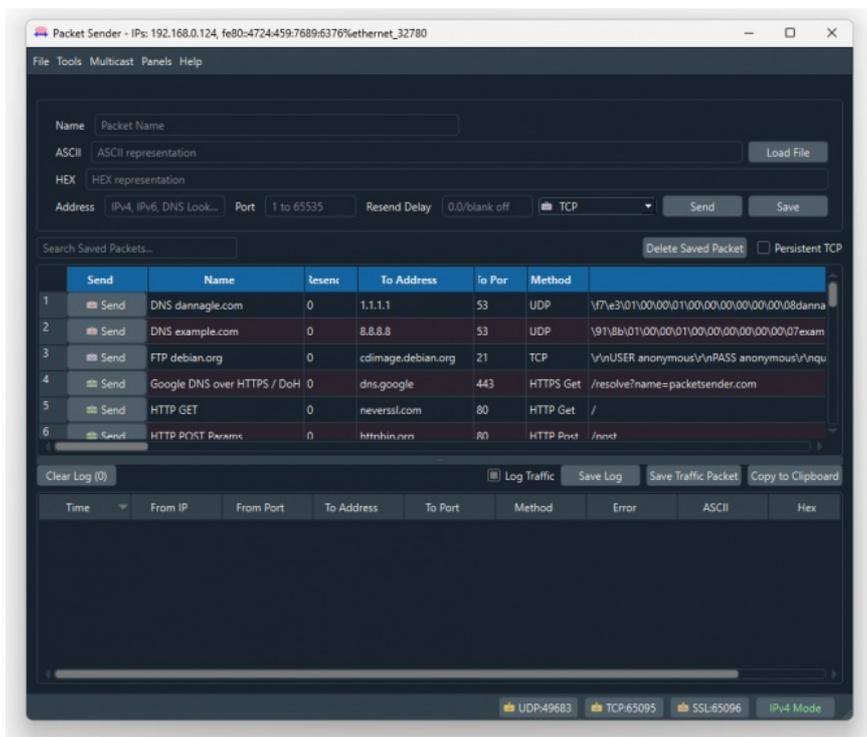
Here is a video example to enable UDP log and receive data via UDP Server software. Video is created with CKL series but applies to all series.

☝

NOTE: This page has same settings both for Server and Client operating modes.

11. WMBus Data Parsing Format

STG decrypts WMBus frames and convert data to Modbus registers Modbus is stored in three parts: Status Block, several Data Blocks depends on number measurements stored in WMBus device and finally the Service Block.

<u>Status Block, 6 Registers:</u>

– WMbus Device ID: 2 Registers

– WMbus Man ID: 1 Register

– WMbus Version: 1 Register

– WMbus Type: 1 Register

– Total Data Count: 1 Register (Represents how many data blocks exists)

<u>Data Block, each 5 Bytes total n bytes:</u>

– Storage Number: 1 Register

– Function Field: 1 Register

– Data Type: 1 Register

– Data Value: 2 Registers

<u>Service Block, 2 Registers:</u>

– Access Number: 1 Register

– RSSI Value: 1 Register

11.1 Parsed Data Details

Parsed data is shared by STG as data blocks. This is valid for both Modbus Frames and web page visualization. Each data block has following values:

- Data Type specified by STG
- Data Value itself (can be a float or long value)
- Function Field, available values are:
  INSTANTANEOUS, MAXIMUM, MINIMUM or ERROR STATE
- Data Storage Number as an integer

Here is complete list for Data types available in STG

Item	Data Type Name	Data Type Value	Data Lenght in bytes stored in MODBUS register	Data Type Format	Data Type description	Range
0	SPECIAL_FUNC	0	4			
1	ENERGY_WH	1	4	FLOAT32	Energy	0.001Wh to 10000Wh
2	ENERGY_J	2	4	FLOAT32	Energy	0.001kJ to 10000kJ
3	VOLUME_M3	3	4	FLOAT32	Volume	0.001l to 10000l
4	MASS_KG	4	4	FLOAT32	Mass	0.001kg to 10000kg
5	POWER_W	5	4	FLOAT32	Power	0.001W to 10000W
6	POWER_J_H	6	4	FLOAT32	Power	0.001kJ/h to 10000kJ/h
7	VOLUMEFLOW_M3_H	7	4	FLOAT32	Volume Flow	0.001l/h to 10000l/h

8	VOLUMEFLOW_M3_M	8	4	FLOAT32	Volume Flow ext.	0.0001l/min to 1000l/min
9	VOLUMEFLOW_M3_S	9	4	FLOAT32	Volume Flow ext.	0.001ml/s to 10000ml/s
10	MASSFLOW_KG_H	10	4	FLOAT32	Mass flow	0.001kg/h to 10000kg/h
11	FLOWTEMPERATURE_C	11	4	FLOAT32	Flow Temperature	0.001°C to 1°C
12	RETURNTEMPERATURE_C	12	4	FLOAT32	Return Temperature	0.001°C to 1°C
13	TEMPERATUREDIFF_K	13	4	FLOAT32	Temperature Difference	1mK to 1000mK
14	EXTERNALTEMP_C	14	4	FLOAT32	External Temperature	0.001°C to 1°C
15	PRESSURE_Bar	15	4	FLOAT32	Pressure	1mbar to 1000mbar
16	ON_TIME_S	16	4	LONG32	ON time	sec
17	ON_TIME_M	17	4	LONG32	ON time	min
18	ON_TIME_H	18	4	LONG32	ON time	hour
19	ON_TIME_D	19	4	LONG32	ON time	days
20	OPER_TIME_S	20	4	LONG32	Operating time	sec
21	OPER_TIME_M	21	4	LONG32	Operating time	min
22	OPER_TIME_H	22	4	LONG32	Operating time	hour
23	OPER_TIME_D	23	4	LONG32	Operating time	days
24	AVR_DUR_S	24	4	LONG32	Average duration time	sec
25	AVR_DUR_M	25	4	LONG32	Average duration time	min
26	AVR_DUR_H	26	4	LONG32	Average duration time	hour
27	AVR_DUR_D	27	4	LONG32	Average duration time	days
28	ACT_DUR_S	28	4	LONG32	Actuality Duration time	sec
29	ACT_DUR_M	29	4	LONG32	Actuality Duration time	min
30	ACT_DUR_H	30	4	LONG32	Actuality Duration time	hour
31	ACT_DUR_D	31	4	LONG32	Actuality Duration time	days
32	DURATION_LAST_CUMULATION_H	32	4	LONG32	Duration since last cumulation	hour
33	DURATION_LAST_CUMULATION_D	33	4	LONG32	Duration since last cumulation	day
34	DURATION_LAST_CUMULATION_M	34	4	LONG32	Duration since last cumulation	month
35	DURATION_LAST_CUMULATION_Y	35	4	LONG32	Duration since last cumulation	year
36	BATTERY_OP_TIME_H	36	4	LONG32	Battery operation time	hour
37	BATTERY_OP_TIME_D	37	4	LONG32	Battery operation time	day
38	BATTERY_OP_TIME_M	38	4	LONG32	Battery operation time	month
39	BATTERY_OP_TIME_Y	39	4	LONG32	Battery operation time	year
40	REMAINING_BATTERY_H	40	4	LONG32	Remaining battery time	hour
41	REMAINING_BATTERY_D	41	4	LONG32	Remaining battery time	day
42	REMAINING_BATTERY_M	42	4	LONG32	Remaining battery time	month
43	REMAINING_BATTERY_Y	43	4	LONG32	Remaining battery time	year
44	MODEL_VERSION	44	4	LONG32		
45	HUMIDITY_Percent	45	4	LONG32		
46	TIME_POINT_DATE	46	4	LONG32	2 bytes date (in 4 bytes register), Data Type G	YY:MM:DD
47	TIME_POINT_DATE_TIME	47	4	LONG32	4 bytes time and date, Data Type F	YY:MM:DD HH:MM:SS
48	BATTERY_CHANGE_DATE_TIME	48	4	LONG32	4 bytes time and date, Data Type F	YY:MM:DD HH:MM:SS
49	FABRICATION_NO	49	4	LONG32		
50	SPECIAL_SUPPLIER_INFO	50	4	LONG32		
51	REVERSEFLOW_VOLUME	51	4	FLOAT32		
52	ERROR_FLAGS	52	4	LONG32		
53	HUMIDITY_Relative	53	4	FLOAT32		
54	DIMENSIONLESS	54	4	LONG32		
55	VERSION	55	4	LONG32		
56	VOLUME_UNCORRECTED	56	4	FLOAT32	Volume	0.001l/h to 10000l/h
57	ENERGY_RESRVD_GJ	51	4	FLOAT32	Energy	
58	NOT_IMPLEMENTED	56	4	LONG32		
59	UNIT_FOR_HCA	57	4	LONG32		
60	RESERVED	58	4	LONG32		

61	ENHANCED_ID	59	4	LONG32		
62	BUS_ADDRESS	60	4	LONG32		

11.1 Modbus Frame Details

Parsed data is shared by STG as data blocks as the details are given above. There is also 2 registers at the end of each frame (service registers) to show Access Number and RSSI values of frame.

Here is an example Modbus data frame (without Modbus frame header part and CRC part for Modbus RTU)

1A00144739269500420037000100000001000349D5DAE0002DFFCD

1A: 1 byte frame length, which is 1Ah and which is 26 in decimal and WMBus data shared as blocks

00144739: 4 bytes of WMBus device id

2695: 2 bytes of WMBus device manufacturer id

0042: 2 bytes of WMBus device version

0037: 2 bytes of WMBus device type

0001: 1 byte of Data count value. Data shared in blocks and this value shows how many blocks are there

00000001000349D5DAE0: Each data blocks has following values in 6 registers and 12 bytes

Storage Number (of data from meter) - 2 bytes

Function Field (of data from meter) - 2 bytes

Data Type (STG uses this value to identify type of Data that is shown above) - 2 bytes, Data Type is Volume in this example

Data Value (float data based on Data Type in this example) - 4 bytes, 1751900.0 in this example

002D: 1 byte of Access Number value as unsigned integer. 45 in this example.

FFCD: 1 byte of RSSI value as signed integer. -51 in this example.

12. WMBus OMS – Modbus Conversion Example

STG series WMBus Gateways can get WMBus frames, decrypt the data and share data as Modbus values.

11.1 Selecting Modbus TCP or Modbus RTU

As a first step user should select "Device Function" under Operating Mode page:

"WMBus – Modbus TCP Gateway (and MQTT Publisher)" (available when device operation mode is server)

Or

"WMBus – Modbus RTU Gateway" (available when device operation mode is client)
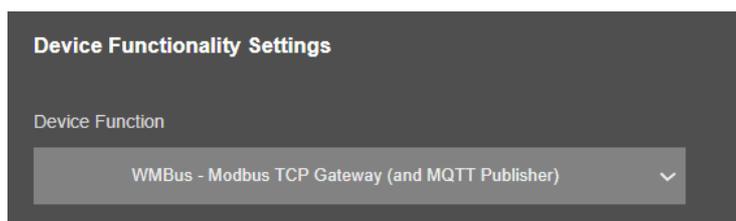
In this example we have selected "WMBus – Modbus TCP Gateway (and MQTT Publisher)" and saved configuration reboot device under "Management" menu.

"Modbus Scheduler Settings" is still empty in this stage.

11.2 Entering Field Device Details with Auto Configuration

STG will listen WMBus frames.

Click "Refresh Auto Config List" button under "WMBus Settings" menu. When this button clicked, STG will show received frames with latest RSSI values for all WMBus devices since the device powered up. If there is no RSSI value, that means STG did not receive any WMBus frame yet, user may wait and click button again.

If a line has notation "Recently Dedected" on the right side of entyr and framed with green line, that means this WMBus frame received and this device is not in configuration (or not in Black List), user may select it to "Add to Config" or "Add to Black List" based on needs.

If "Add to Black List" checked, after clicking "Save Configuration", STG will save this device id under "Black List Table" (and enables list by checking "Black List Enabled" checkbox, if not enabled already) under "Security Settings" menu. This entry also will be freed up from decryption list to be used for new coming frames.

> ☝
>
> NOTE: User must enter AES keys manually.

After all settings are done, click "Save Configuration" button. That process will save configuration and also put the device id to Modbus list (and black List if it is used) automatically thus ease up configuration process dramatically.

Here is "Modbus Scheduler Settings" view under "Operating Mode" menu after saving settings.
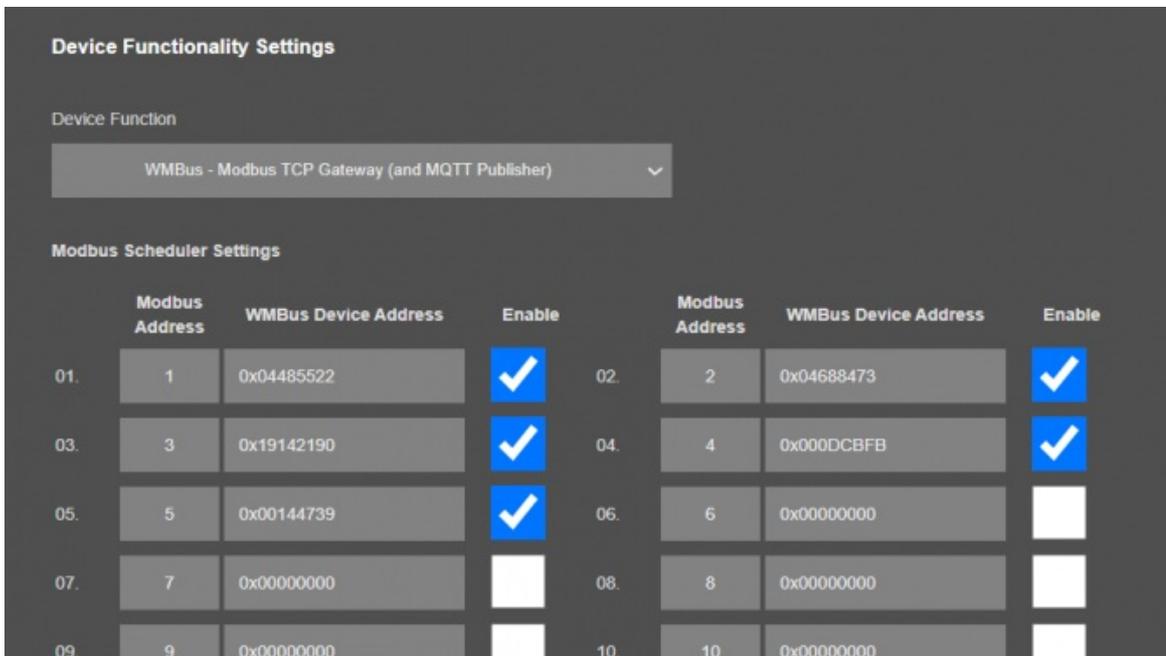
Each WMBus device is separate MODBUS device so user should matched Manufacturing IDs of field devices with selected Modbus Device Addresses. Then user should enable the entry that STG will expect the WMBus data and share info as Modbus register.

Here is "Black List Address Table" view under "Security Settings" menu after saving settings.

☝
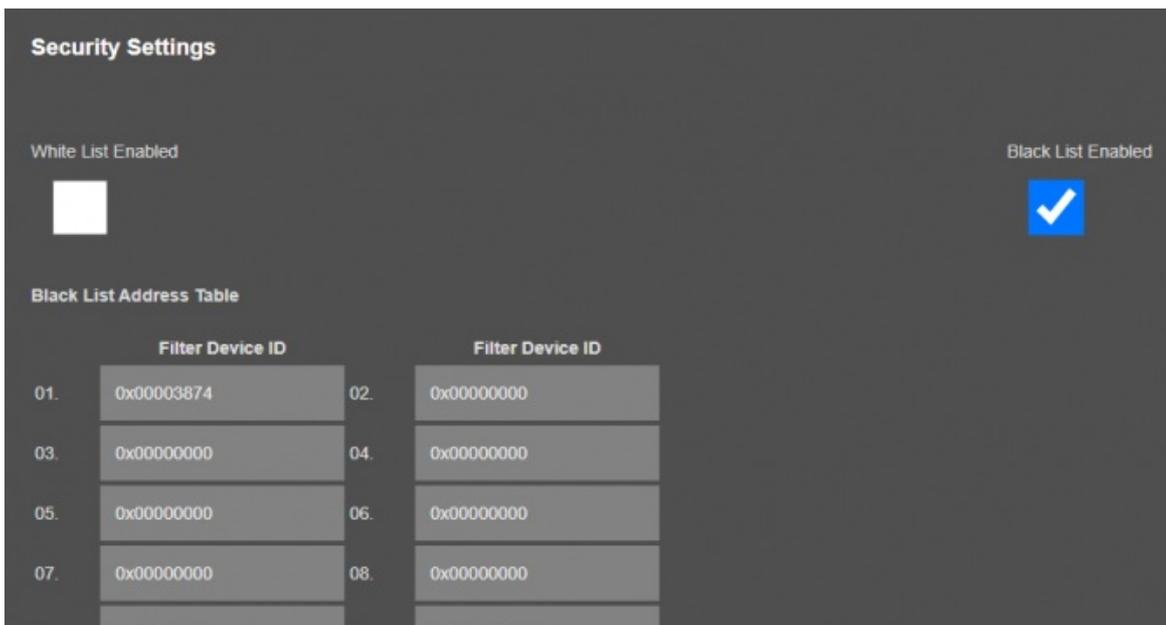
NOTE: All values in "AES-128 Decryption List", "Modbus Scheduler Settings" and "Black List Address Table" can also be entered/changed mnaually. In that case user may get help from "WMBus Status" menu to get WMBus frame details as follows:

Latest Message Received - WMBUS Link Layer fields

C-Field		Manufacturer ID
0x44		0x0601

Device ID		Version ID
0x04688473		20

Type		CI-Field
7		0xA0
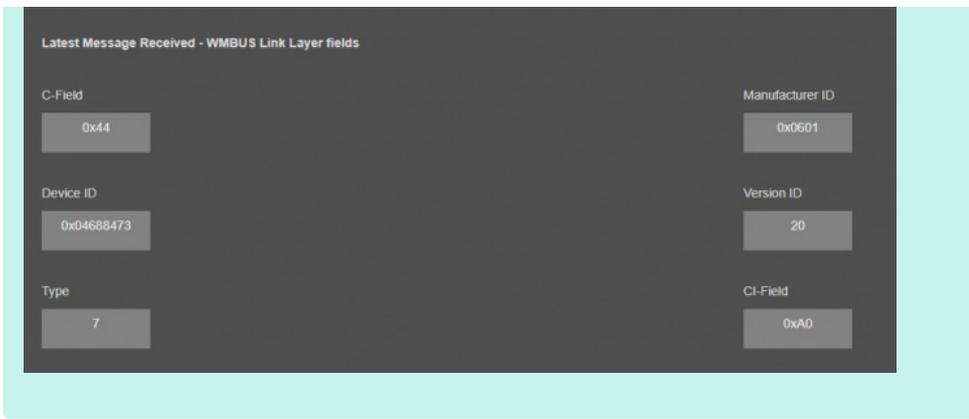
11.3 Setting Modbus Communication Parameters

If WMbus – Modbus RTU Gateway is set, user can make serial interface settings under menu Serial Settings.
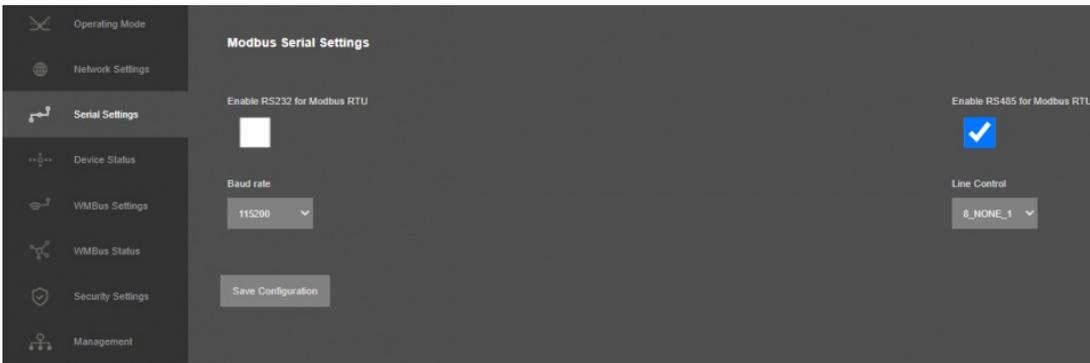
If WMbus – Modbus TCP Gateway is set, user can change Modbus TCP Port under menu Network Settings.
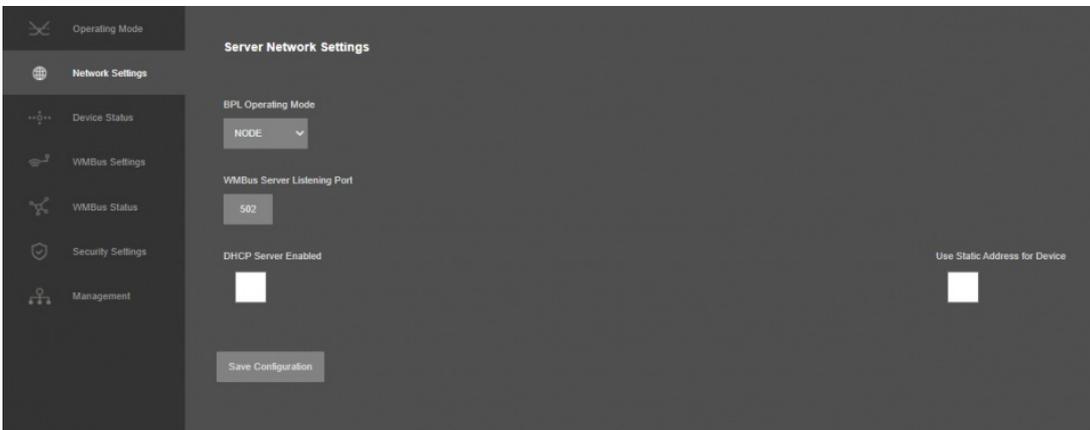
11.4 Reading Modbus Register – Rules

STG series WMbus – Modbus TCP/RTU Gateway devices can get WMBus OMS data and convert those to Modbus registers. Users or field devices can read WMBus data over Modbus registers. Here are basic rules:

- Each wMBUS device is separate MODBUS slave

Slave ID is possible to configure in STG web page "Operating Mode" in table "Modbus Scheduler Settings" column "Modbus address"

- MODBUS data is stored in three parts: Status Block, several Data Blocks depends of number measurements stored in WMBus device and finally the Service Block

Number of words needed to read all register is possible after counting length of data blocks. For example, if in status block "data count" value is equal to 2 you have to read 6+(2*5 (16))+2 words from address 0

- To restrict number of WMBUS devices it is possible use "White" or "Black" list of WMBus devices

See "Security Settings" in STG web configuration page

– To Decrypt WMBus devices you have to fill "AES-128 Decryption List"

See " WMBus Settings" in STG web configuration page

– Current WMBus frame details is possible to find in " WMBus Status" page

See " WMBus Status" in STG web configuration page

– Current Modbus frame details is possible to find in "Device Status" page

See "Device Status" in STG web configuration page

11.5 Reading Modbus Register – Modbus Register Structure

MODBUS data is stored in three parts: Status Block, several Data Blocks depends on number measurements stored in WMBus device and finally the Service Block.

Status Block, 6 Registers:

– WMbus Device ID: 2 Registers

– WMbus Man ID: 1 Register

– WMbus Version: 1 Register

– WMbus Type: 1 Register

– Total Data Count: 1 Register (Represents how many data blocks exists)

Data Block, each 5 Bytes total n bytes:

– Storage Number: 1 Register

– Function Field: 1 Register

– Data Type: 1 Register

– Data Value: 2 Registers

> ☝️
>
> NOTE: Please check **"Chapter 11: WMBus Data Parsing Format"** to see details of Data Types used by STG

Service Block, 2 Registers:

– Access Number: 1 Register

– RSSI Value: 1 Register

11.6 Reading Modbus Register – Modbus Function

In this example "Marcom OPC Server Modnet" software is used to read Modbus values and Modbus TCP protocol is used.

Any Modbus software can read following values with similar settings.

Enter TCP IP and TCP Port of STG for connection. In our example TCP IP is "192.168.0.121" and port is "502".
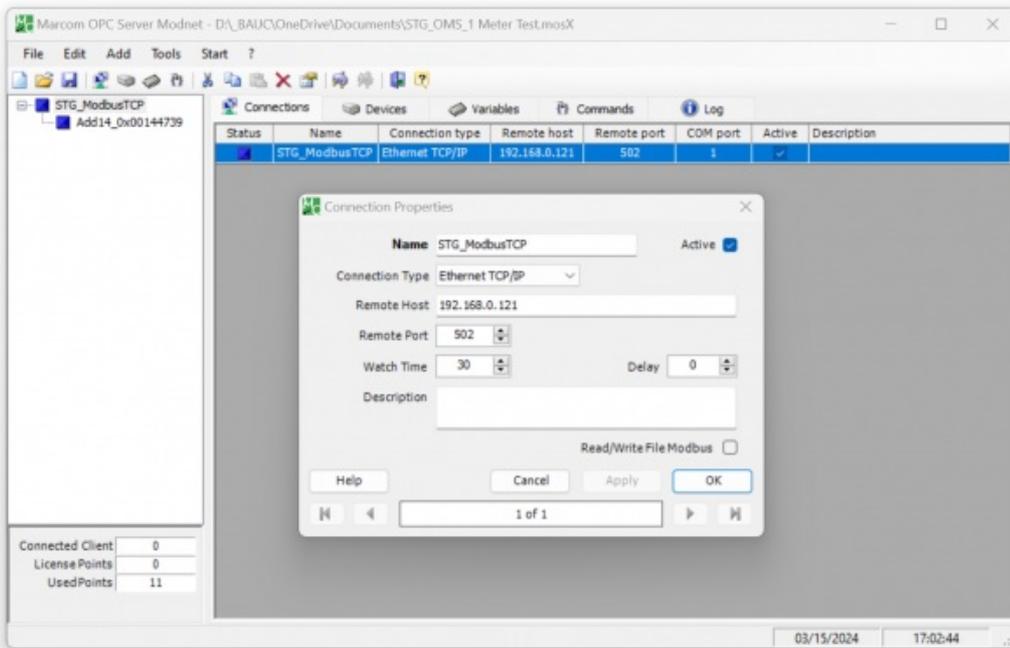
Then enter device Modbus ID.

In this example we want to read device id "0x00144739" and this is mapped with modbus address "5".
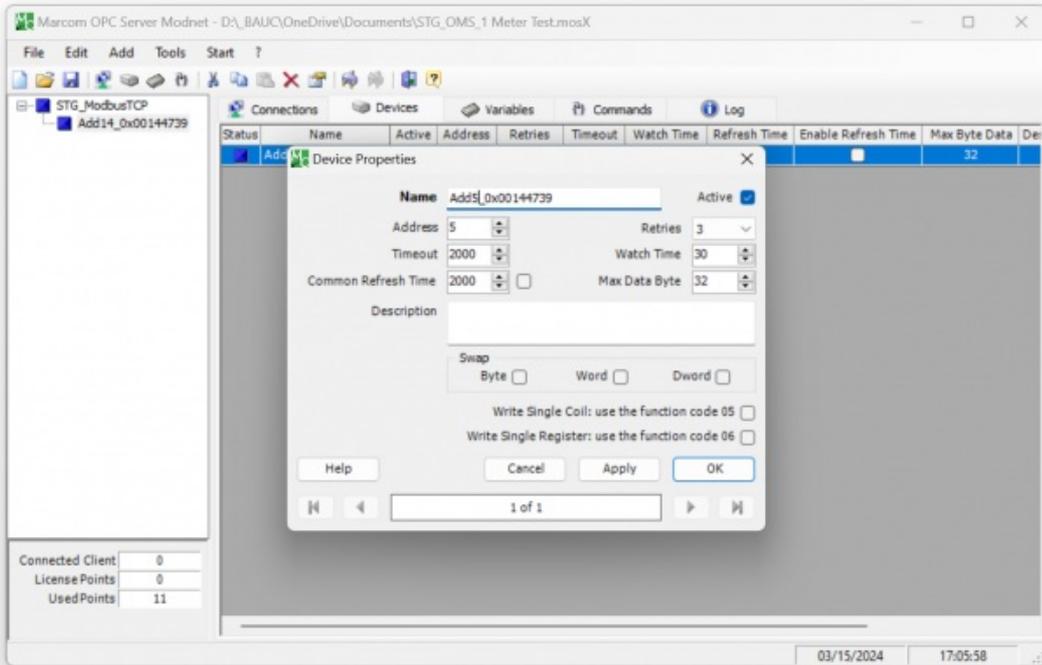
Then checked parsed data from "WMBus Status" page to see the parsed data. We will read 1 data blcok in this example.

OMS Parsing

Data Type - Data Value - Function - Storage No
VOLUME I------------1751900.0---------------------INSTANTANEOUS-------0

☝

NOTE: If data type part show "NOT IMPLEMENTED", data is not implemented as a valid type in STG and cannot be read via Modbus and that is not count in data blocks.

User may contact our company if that data is needed to be implemented and read via Modbus.

Enter register details and data types baesd on read data, in this example we will read Status Block, Data Block and Service Block (all blocks)
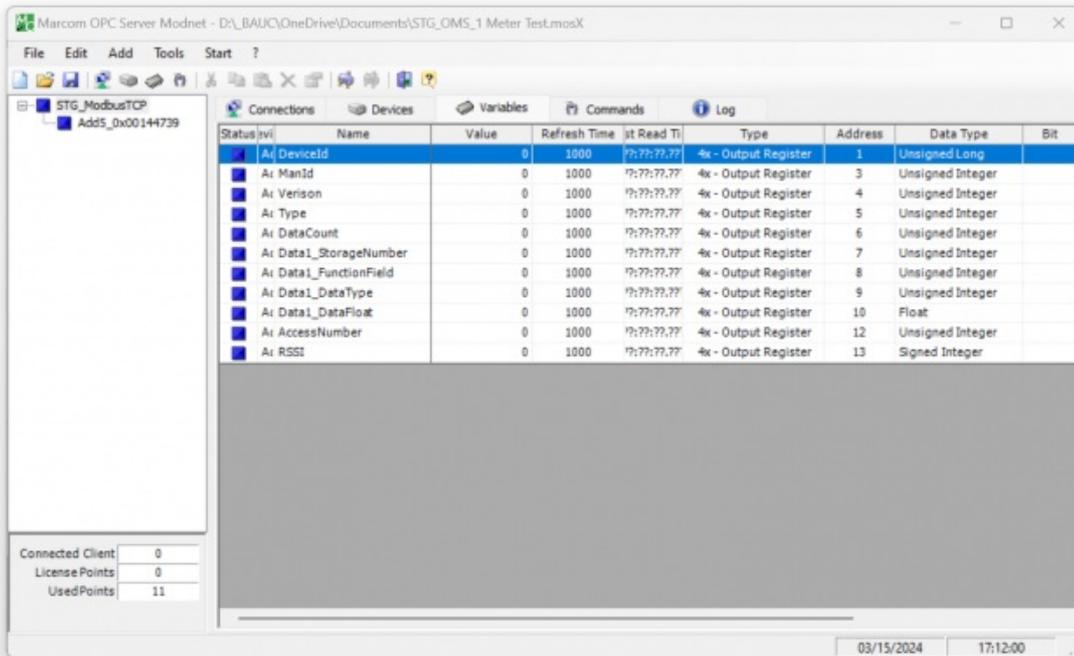
Status Block, 6 Registers:

- WMbus Device ID: 2 Registers, unsigned long

- WMbus Man ID: 1 Register, unsigned integer

- WMbus Version: 1 Register, unsigned integer

- WMbus Type: 1 Register, unsigned integer

- Total Data Count: 1 Register, unsigned integer

1 Data Block:

- Storage Number: 1 Register, unsigned integer

- Function Field: 1 Register, unsigned integer

- Data Type: 1 Register, unsigned integer

- Data Value: 2 Registers, Float 32 bytes

☝

NOTE: Please check **"Chapter 11: WMBus Data Parsing Format"** to see details of Data Types used by STG.

In this example, "Data Type" is "VOLUME l".

Service Block, 2 Registers:

- Access Number: 1 Register, unsigned integer

- RSSI Value: 1 Register, signed integer

Save settings and click "Start OPC Server" (or F9) button to read values.
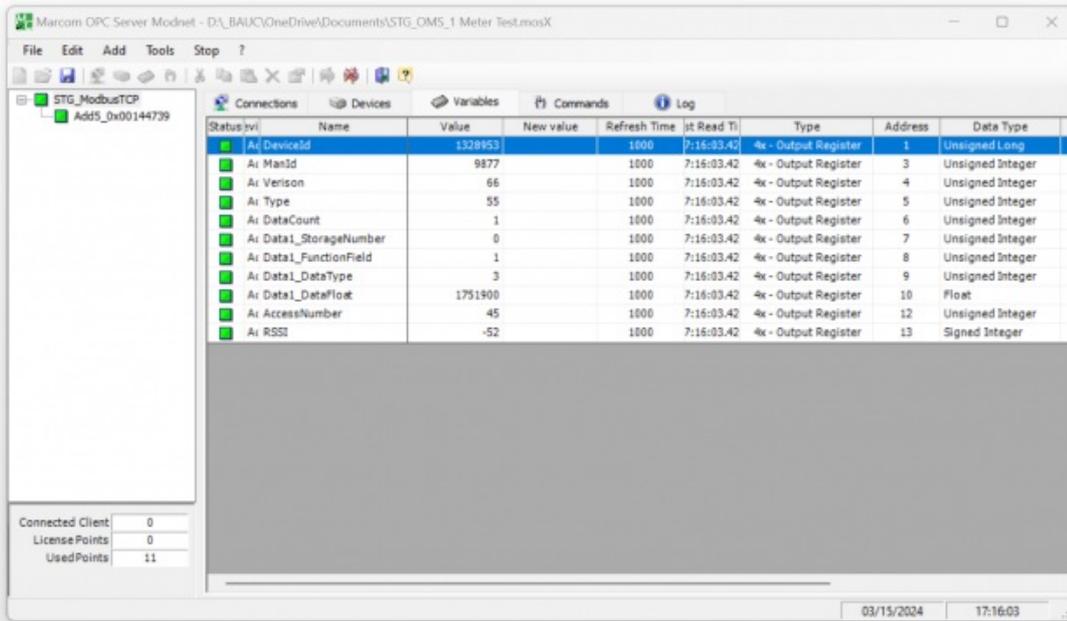
Here is log for sent and received frames
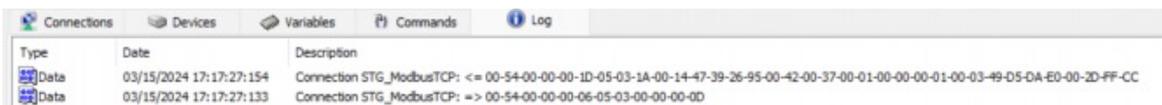
Received frame details are as follows:

00540000001D: 6 bytes of Modbus TCP protocol header

05: 1 byte of Modbus adress

03: 1 byte fo Modbus Function Code

1A: 1 byte data length, which is 1Ah and which is 26 in decimal and WMBus data shared as blocks

00144739: 4 bytes of WMBus device id

2695: 2 bytes of WMBus device manufacturer id

0042: 2 bytes of WMBus device version

0037: 2 bytes of WMBus device type

0001: 1 byte of Data count value. Data shared in blocks and this value shows how many blocks are there

00000001000349D5DAE0: Each data blocks has following values in 6 registers and 12 bytes

Storage Number (of data from meter) - 2 bytes

Function Field (of data from meter) - 2 bytes

Data Type (STG uses this value to identify type of Data that is shown in previous example) - 2 bytes, Data Type is Volume in this example

Data Value (float data based on Data Type in this example) - 4 bytes, 1751900.0 in this example

002D: 1 byte of Access Number value as unsigned integer. 45 in this example.

FFCD: 1 byte of RSSI value as signed integer. -51 in this example.

👆

NOTE: This example is given for Modbus TCP Frame. Modbus RTU frame will

be similar in following manner :
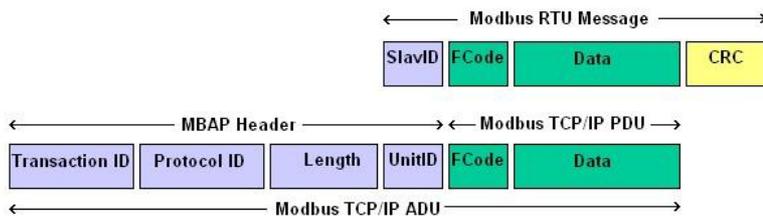
13. MQTT Communication Example

STG Series WMBus Gateways can get Wireless MBus (WMBus) frames, decrypt them and parse them. Field devices or applications can get meter WMBus data mapped to Modbus registers via Modbus TCP (or Modbsu RTU). Simultaneously all meter data can be sent to MQTT Server.

13.1 MQTT Data Transmission: Example

Enter meter reading details to STG and make sure all readings are ok.

Please follow details regarding settings and put into operation of STG based on explanations in"**Chapter 12: WMBus OMS – Modbus Conversion Example**" for adding WMBus meter details and parsing WMBus meter data.

Then go to "Network Settings" menu and "Enable MQTT Publisher for Parsed WMBus Packages" part.

NTP Settings

NTP Server

pool.ntp.org

MQTT Settings

Enable MQTT Publisher for Parsed WMBus Packages

☑

MQTT Broker IP

75.2.83.130

MQTT Broker Port

1883

Client ID

MQTT_STG_Client

User Name

MQTT_STG_User

Password

•••••••••••••••••••••••••••••••••

Publish Topic

MQTT_STG_Publish_topic

Subscribe Topic

MQTT_STG_Conf_topic

Minimum Send Interval for a Parsed Data (in seconds)

60

Data Format

RAW WMBUS DECRYPTED DATA

Server Network Settings

Server Listening Port

502

Maximum Number of Clients

2

DHCP Server Enabled

☐

Use Static Address for Device

☐

Save Configuration

In our example we will use "https://tago.io/" as MQTT Server.

Go to MQTT server panel and click "Add Device" to add STG to MQTT Server. We will also get password after adding device.

Select "Custom MQTT".

MQTT: The Standard for IoT Messaging

Custom MQTT

Details
Give a name for this device and learn about this network here.

Define the type of bucket to be used to store data for this device.

Device name

LKM

Data storage type

Device Data Optimized (Immutable)

Data Retention
The Data Retention feature automatically removes old data from the bucket after the period you define here. Learn more.

Period

Monthly

Retention

1

ⓘ This selection limits the storage for this device to **1 Million data registers per month.** Upgrade your plan to change the Period and Retention.

Main information
Set the initial configurations for this device.

Payload Type

Auto Parser (JSON or TEXT)

Description
Learn how this device works, and what its capabilities are.

This is quick setup to create new devices that can be used for any applications and functions to access the TagoIO MQTT broker.

Your device will only need to use the device-token to send data to your account. Don't use the Authorization feature when working directly our broker.

More information can be found in the *documentation link* shown above.

Cancel   Create my Device

Then enter "Device name" in pop up screen and click "Create My Device".

MQTT: The Standard for IoT Messaging

Custom MQTT

All done!

✓ Creating device
✓ Creating and linking a bucket

Create another device   Finish

Click "Finish" when all done.

Once the device is created, click "Copy Token" button. This will copy password value.

Then go back to STG and enter that value as password and click "Save Configuration" button.

"MQTT Broker IP": TCP IP of the MQTT Server. User must enter IP value
75.2.83.130 is IP of "https://tago.io/" web address.

"MQTT Broker Port": TCP Port of the MQTT Server.
1883 is TCP Port of "https://tago.io/" web address.

"Client ID": MQTT Publisher client id. Default is MQTT_STG_Client.
Maximum length for this field is 32.

"User Name": MQTT Publisher user name. This must be entered based on MQTT server settings.
Maximum length for this field is 64.

"Password": MQTT Publisher password. This must be entered based on MQTT server settings.
Maximum length for this field is 48.

"Publish Topic": MQTT Publisher topic value. Default is MQTT_STG_Publish_topic.
Maximum length for this field is 32.

"Subscribe Topic": MQTT Publisher subscribe topic value. Default is MQTT_STG_Subscribe_topic.
Maximum length for this field is 32.

"Minimum Send Interval for a Parsed Data (in seconds)": Minimum value to send meter data to MQTT Server. This time is the minimum time to send data to server, if WMBus meter sends frames in longer periods then it will be basis of the sending interval.

"Data Format": Options for how data is shared by STG with MQTT server. There are 2 options:

RAW WMBUS DECRYPTED DATA

PARSED DATA AS OBJECTS

PARSED DATA AS MODBUS FRAME

When selected as "Raw WMBus Decrypted Data", STG will share WMBus frame as it is but decrypted as follows as an example:

13:58:03:
[MQTT] Device publish
{ "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710413878\",\"Data\":
{\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-
51\",\"AcNu\":\"45\",\"RawDt\":\"202D900F002C2540430000CF464FAE2BBFB78B7239471400952642032D181007102F2F04953A6F4400002F2F2F2...
"qos": 0, "device": "65e076784d9b150010876950", "messageId": null, "isHex": false }

    { "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710413878\",\"Data\":
    {\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-
    51\",\"AcNu\":\"45\",\"RawDt\":\"202D900F002C2540430000CF464FAE2BBFB78B7239471400952642032D181007102F2F04953A6F4400002F2F2F2F2F2F\"}}", "qos": 0, "device":
    "65e076784d9b150010876950", "messageId": null, "isHex": false }

MQTT message shows STG device name, device id, NTP time and data itself

Data has values of WMBus Device Id, Manufacturer Id, Device Version, Device Type, RSSI Value of recevied signal, Access Number value and Decrypted frame Data Block.

When selected as "Parsed Data As Objects", STG will share WMBus data as parsed objects as follows as an example:

14:04:57:
[MQTT] Device publish
{ "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710414288\",\"Data\":
{\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-51\",\"AcNu\":\"45\",\"DtCnt\":\"1\",\"Blcks\":
[{\"S\":\"0\",\"F\":\"1\",\"T\":\"3\",\"V\":\"1751900.0\"}]}}", "qos": 0, "device": "65e076784d9b150010876950", "messageId": null, "isHex": false }

    { "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710414288\",\"Data\":
    {\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-51\",\"AcNu\":\"45\",\"DtCnt\":\"1\",\"Blcks\":[{\"S\":\"0\",\"F\":\"1\",\"T\":\"3\",\"V\":\"1751900.0\"}]}}", "qos": 0,
    "device": "65e076784d9b150010876950", "messageId": null, "isHex": false }

MQTT message shows STG device name, device id, NTP time and data itself

Data has values of WMBus Device Id, Manufacturer Id, Device Version, Device Type, RSSI Value of recevied signal, Access Number value.

Parsed Data is shared in blocks and Data Count value shows how many blocks are there.

In this example there is 1 block

{\"S\":\"0\",\"F\":\"1\",\"T\":\"3\",\"V\":\"1751900.0\"}

Here are explanations:

S: Storage Number (of data from meter)

F: Function Field (of data from meter)

T: Data Type (STG uses this value to identify type of Data)

V: Data Value (can be float or UINT - 4 byte data, matched with Data Type)

☝️

NOTE: Parsed data shared as data blocks. This is valid for both Modbus Frames and web page visualization. Each data block has following values:

  – Data Type specified by STG

  – Data Value (can be a float or long value)

  – Function, available values are:

    INSTANTANEOUS, MAXIMUM, MINIMUM or ERROR STATE

  – Data Storage Number as an integer

  details are explained in "Chapter 11: WMBus Data Parsing Format".

When selected as "Parsed Data As Modbus Frame", STG will share WMBus data as Modbus like frame as follows as an example:

14:22:03:
[MQTT] Device publish
{ "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710415316\",\"Data\":
{\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-
51\",\"AcNu\":\"45\",\"ModFr\":\"1A00144739269500420037000100000001000349D5DAE0002DFFCD\"}}", "qos": 0, "device":
"65e076784d9b150010876950", "messageId": null, "isHex": false }
    { "topic": "MQTT_STG_Publish_topic", "payload": "{\"STGid\":\"45FA\",\"STGNme\":\"WMBUS_SERVER\",\"Time\":\"1710415316\",\"Data\":
    {\"DevId\":\"144739\",\"ManId\":\"2695\",\"Ver\":\"42\",\"Typ\":\"37\",\"RSSI\":\"-51\",\"AcNu\":\"45\",\"ModFr\":\"1A00144739269500420037000100000001000349D5DAE0002DFFCD\"}}",
    "qos": 0, "device": "65e076784d9b150010876950", "messageId": null, "isHex": false }

MQTT message shows STG device name, device id, NTP time and data itself

Data has values of WMBus Device Id, Manufacturer Id, Device Version, Device Type, RSSI Value of recevied signal, Access Number value.

Parsed Data is shared in Modbus like frame, in this example as follows.

1A00144739269500420037000100000001000349D5DAE0002DFFCD

1A: 1 byte frame length, which is 1Ah and which is 26 in decimal and WMBus data shared as blocks

00144739: 4 bytes of WMBus device id

2695: 2 bytes of WMBus device manufacturer id

0042: 2 bytes of WMBus device version

0037: 2 bytes of WMBus device type

0001: 1 byte of Data count value. Data shared in blocks and this value shows how many blocks are there

00000001000349D5DAE0: Each data blocks has following values in 6 registers and 12 bytes

Storage Number (of data from meter) – 2 bytes

Function Field (of data from meter) – 2 bytes

Data Type (STG uses this value to identify type of Data that is shown in previous example) – 2 bytes, Data Type is Volume in this example

Data Value (float data based on Data Type in this example) – 4 bytes, 1751900.0 in this example

002D: 1 byte of Access Number value as unsigned integer. 45 in this example.

FFCD: 1 byte of RSSI value as signed integer. –51 in this example.

☝️

NOTE: Parsed data shared as data blocks. This is valid for both Modbus Frames and web page visualization. Each data block has following values:

- Data Type specified by STG

- Data Value (can be a float or long value)

- Function, available values are:

  INSTANTANEOUS, MAXIMUM, MINIMUM or ERROR STATE

- Data Storage Number as an integer

details are explained in "Chapter 11: WMBus Data Parsing Format".

When all settings are done click "Save Configuration".

Go back to "https://tago.io/" panel and click "Live Inspector" for the STG and click "Play button".

Data will be shown in real time.
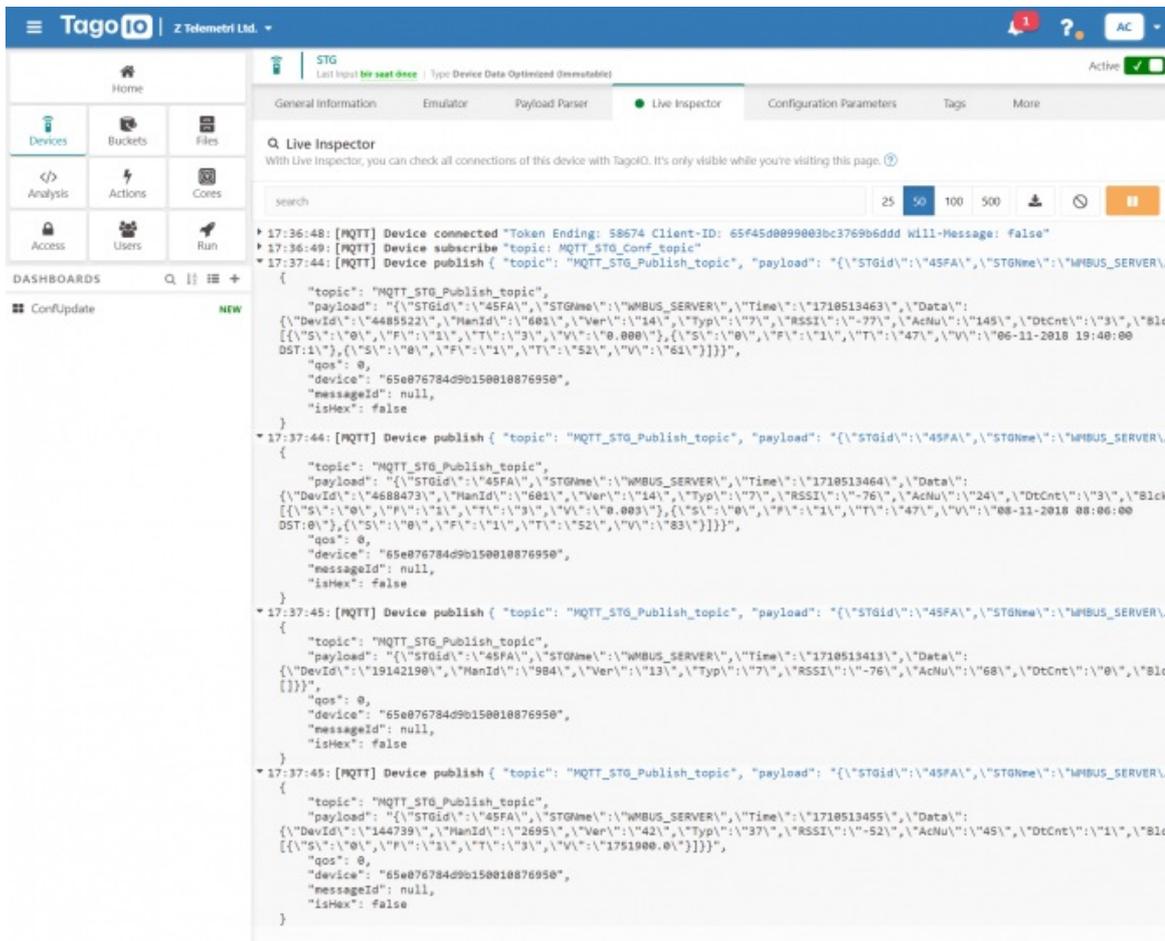
User has to parse and use that parsed data for visualization (for graphs for example) for their application. That is beyond scope of STG usage.

14. WMBus OMS – Modbus Conversion and Send Data to MQTT Server - Video Examples

STG Series WMBus Gateways have built in mechanism to get WMBus frames and Auto Configure. AES Decryption, Modbus Adress mapping and Black list items can be auto filled by Auto Configuration mechanism, thus user do not have to enter manually each item besides AES Key. STG has capability to decrypt and convert WMBus data to Modbus protocol for up to 40 devices in list.

14.1 Convert WMBus OMS Packages to Modbus TCP - Video Example

STG Series WMBus Gateways can receive WMBus OMS packages and convert them to Modbus TCP packages. This way WMBus packages of many meters can directly be collected on a remote or local Data Server or RTU device as Modbus registers and can be used for other applications.

STG Configuration

- Server Mode
- Device Function: WMBus – Modbus TCP Gateway
- Wmbus Device Mode Other
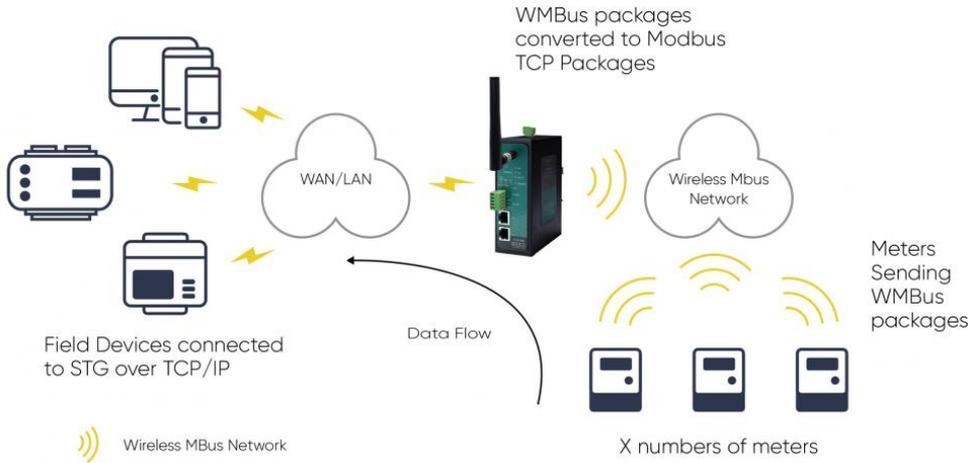
STG set in Server mode and Device function set as WMBus – Modbus TCP Gateway. Manufacturing IDs of field devices can be listed with related Modbus addresses. If needed decryption list can be set and White list can be enabled. Received WMBus OMS messages will be converted to Modbus Registers and field or remote Modbus RTU devices can query that data from related Modbus addresses.

Here is an example video for this application (this video follows steps described in Chapter 12. WMBus OMS – Modbus Conversion Example) :

14.2 Transmit Received and Parsed WMBus Packages to MQTT Server – Video Example

STG Series WMBus Gateways can receive WMBUS packages, decrypt and parse. All parsed data can be sent to MQTT server for web based applications in several different formats.

STG Configuration

- Server Mode
- Device Function: WMBus – Modbus TCP Gateway and MQTT Publisher
- Wmbus Device Mode Other
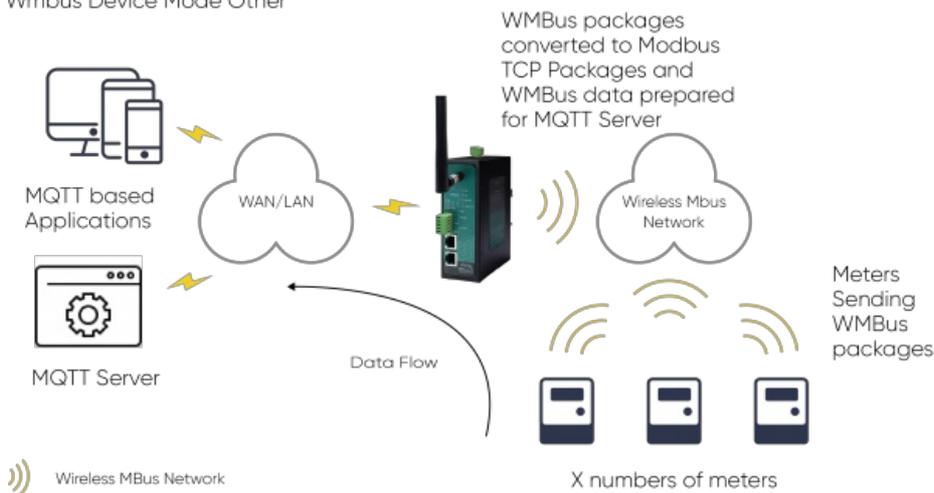
STG act as TCP Server to WMBus Gateway configured in Server Mode. Field devices still can connect to STG device and read WMBus packages in Modbus TCP protocol. STG will get WMBus frames, decrypt them, parse them serve as Modbus TCP packages if needed and send WMBus data to MQTT Server simultaneoulsy.

Here is an example video for this application (this video follows steps described in Chapter 13. MQTT Communication Example) :

15. Ordering Information

STG154: 868MHz WMBus – Modbus TCP/RTU Gateway with MQTT Publisher, 2x 10/100 T(x) ETH ports, 1 x RS232 & 1 x RS485, 5-48V (max. 60V) DC Power Input

STG254: 868MHz WMBus – Modbus TCP/RTU Gateway with MQTT Publisher, 2x 10/100 T(x) ETH ports, 1 x RS232 & 1 x RS485, 100 - 240V AC (120 – 370V DC), 50Hz to 60Hz AC Power Input

STG655: 868MHz WMBus – Modbus TCP/RTU Gateway with MQTT Publisher, 2x 10/100 T(x) ETH ports + 1 x BPL (Broadband Power Line) Link, 1 x RS232 & 1 x RS485, 3 Phase AC Power Input, 110V–240V/50-60Hz

STG154 - D16: 868MHz WMBus – Modbus TCP/RTU Gateway with MQTT Publisher, 2x 10/100 T(x) ETH ports, 1 x RS232 & 1 x RS485, 5-48V (max. 60V) DC Power Input

STG254 - D16: 868MHz WMBus – Modbus TCP/RTU Gateway with MQTT Publisher, 2x 10/100 T(x) ETH ports, 1 x RS232 & 1 x RS485, 100 - 240V AC (120 – 370V DC), 50Hz to 60Hz AC Power Input

STG655 - D16: 868MHz WMBus – Modbus TCP/RTU Gateway with MQTT Publisher, 2x 10/100 T(x) ETH ports + 1 x BPL (Broadband Power Line) Link, 1 x RS232 & 1 x RS485, 3 Phase AC Power Input, 110V–240V/50-60Hz

☝️

NOTE:

STG has capability to decrypt and convert WMBus data to Modbus protocol for up to 40 devices in list

There is also a version with up to 20 devices for Modbus conversion and up to 16 devices for decryption. Those version model names differantiated with - D16 extension.

16. Product Selection

Model	5-48V (max. 60V) DC Power Input	100 - 240V AC (120 – 370V DC), 50Hz to 60Hz AC Power Input	3 Phase AC Power input, 110V–240V/50-60Hz AC Power Input	2 x 10/100 T(x) ETH Ports	1 x RS232 and 1 x RS485 Serial Ports	BPL (Broadband Power Line) Link	Up to 40 WMBus device data decryption and parse	Up to 16 WMBus device data decryption and up to 20 WMBus device data parse
STG154	X			X	X		X	
STG254		X		X	X		X	
STG655			X	X	X	X	X	
STG154 - D16	X			X	X			X
STG254 - D16		X		X	X			X
STG655 - D16			X	X	X	X		X