WM-I3® metering modem
MQTT Protocol Technical Description
v1.0

2021-03-26

WM Systems LLC 8 Villa str., Budapest H-1222 HUNGARY
Phone: +36 1 310 7075
Email: sales@wmsystems.hu Web: www.wmsystems.hu

Document specifications

This documentation was made for presenting the MQTT configuration of the **WM-I3**® pulse counter / MBUS data collector and transmitter device.

Document Version:	REV 1.0
Hardware Type/Version:	WM-I3® pulse counter modem, MQTT Protocol Technical description
Hardware Version:	V 3.0
Bootloader Version:	V 1.80
Firmware Version:	V 3.01.01
WM-E Term® configuration software version:	V 1.3.51
Pages:	11
Status:	Final
Created:	26-03-2021
Last Modified:	26-03-2021

Chapter 1. Foreword

The WM-I3® device is also capable of using for MQTT compatible communication, which can be operated by performing the following settings.
Currently, the data transmission part has been implemented within the MQTT protocol.

For demo and test purposes, we recommend to use the MQTT Explorer software, which can be downloaded from the following URL: http://mqtt-explorer.com/

To make the necessary settings, you need to use the WM-E Term v1.3.51 or later configuration program. The usage of the software is described in the product's *User Guide* documentation.

The configuration software is also required for the configuration, which can be downloaded from the following link: https://www.m2mserver.com/m2m-downloads/WM-ETerm_V1_3_51.zip

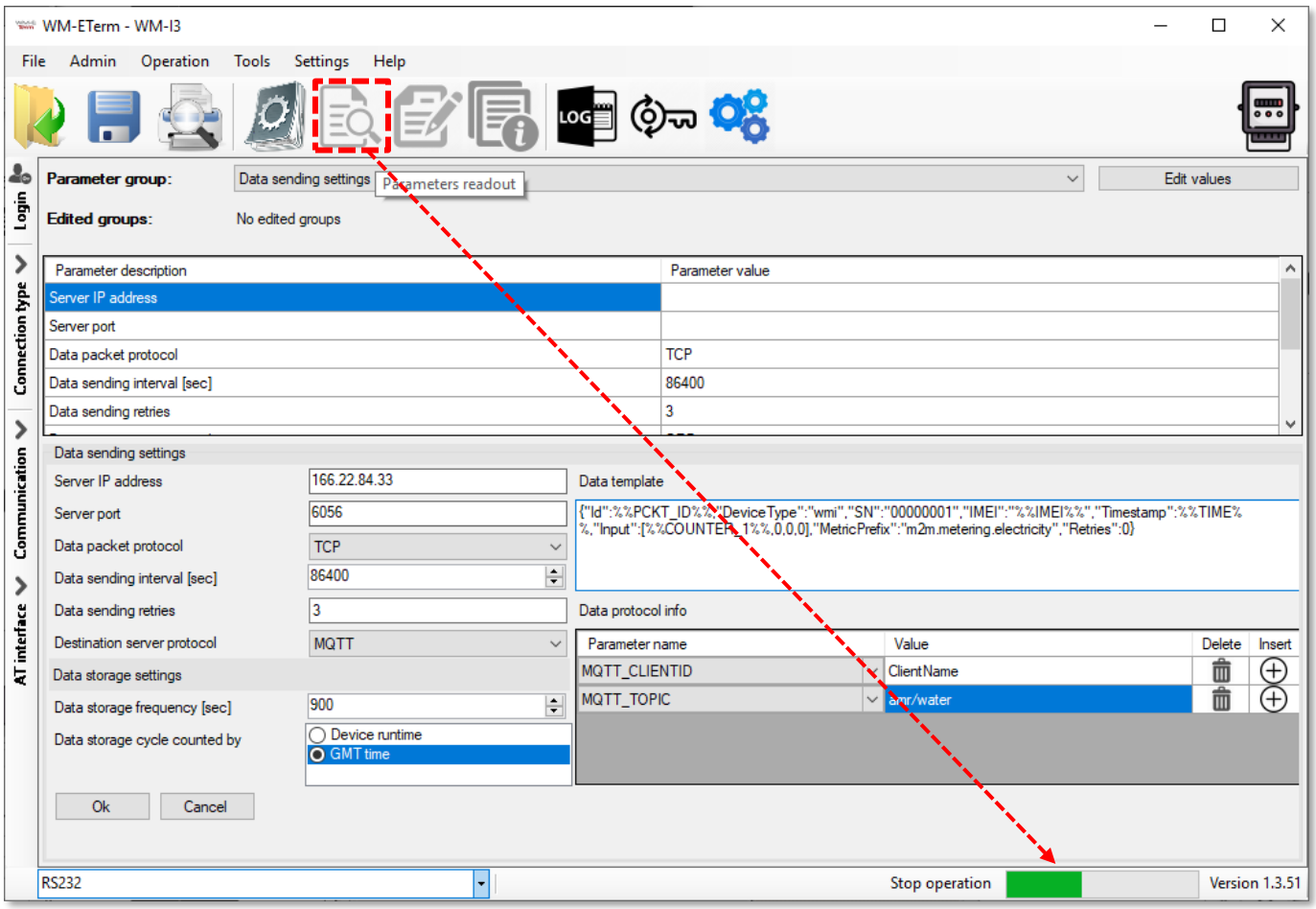
Chapter 2. Settings

1. An MQTT template must be created on both the MQTT side and the WM-I3® client side.

2. Open the *WM-E Term* device configuration software (**WM-ETERM.exe** file), select **WM-I3** device, then click to **Login**.

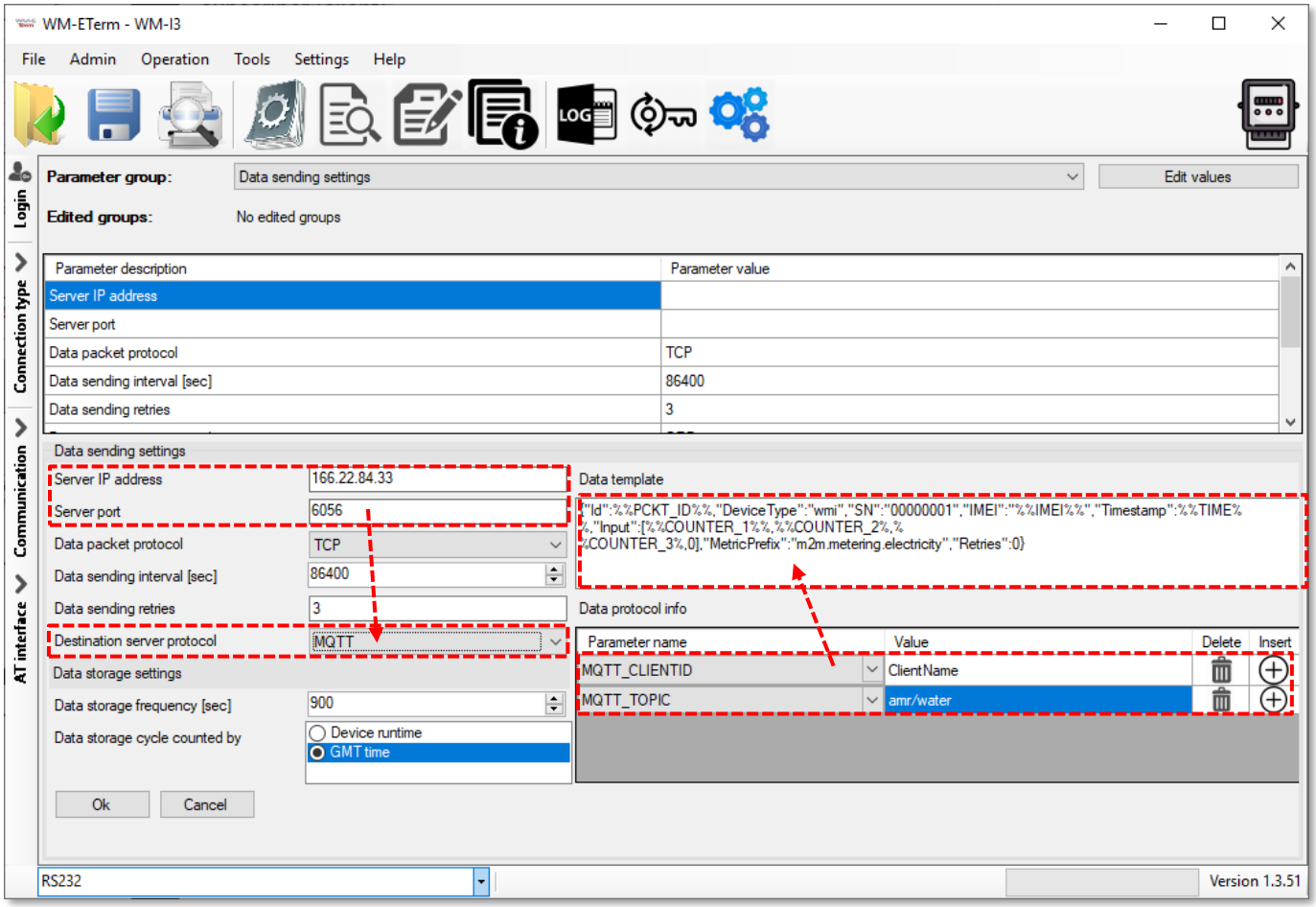
3. Connect to the *WM-I3®*, and download the device settings (click tot he **Read parameters** icon in the menu) - or download the device's factory configuration file and opent it.

You can download the sample configuration file from here:
https://www.m2mserver.com/m2mdownloads/SampleConfigWMI3.cfg



4. Wait for the parameters to load, then select the ***Data sending settings*** group at the top of the screen, at the **Parameter groups** field and then press to the **Edit values** button at right.

Then the following editable interface will be loaded.



5. At left, at the *Data Send Settings* part, enter the **Server IP address**, **Server port** number values, and change the **Destination server protocol** type to *MQTT*.

6. At bottom right, at the *Data protocol info* section, select and **Parameter name** of the required MQTT field. (You can add more fields by **Insert +**, button). Then enter the parameter **Value** to every **Parameter Name** according to the following syntax.

MQTT connection parameters:

- **MQTT_CLIENTID** – MQTT client identifier
- MQTT_QOS – Service quality level
- MQTT_RET – retain level
- **MQTT_TOPIC** – topic name

The required MQTT server-client identifiers tot he connection, can be added at the protocol information, in plain text string format. The green, marked parameters are obligatory to use and define.

Example:

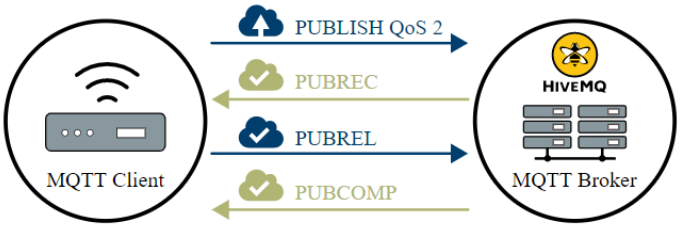
MQTT_CLIENTID:ClientName **MQTT_TOPIC**:amr/water

For the **MQTT_QOS** parameter you can specify the quality of service (QoS) level. It's a level of agreement between the sender of the message and the recipient of the message, which determines the guarantee of delivery of the given message. There are 3 levels:

- Up to once (0)

- At least once (1)

- Exactly once (2)



The two sides of the message delivery must also be considered:

1. Delivery of the message from the publishing client to the broker.

2. Delivery of a message from the broker to the subscriber client.

If the subscriber client specifies a lower QoS than the publishing client, the broker forwards the message with a lower quality of service.

Other examples, explanation:

https://www.hivemq.com/blog/mqtt-essentials-part-6-mqtt-quality-of-service-levels/

The MQTT_RET retention value field specifies whether the server retains the value after sending it to the client. Possible values:

- 0 - The server will not keep the message after sending it to the subscriber (client)
- 1 - The server saves the message after it has been sent to the subscriber (client)

7. In the *WM-E Term* window, at the **Data protocol template** field define the MQTT template with the data sending parameters (a list of parameter names and their values in JSON format with the following syntax.)

The template content can be similar to this:

```
{"Id":%%PCKT_ID%%,"DeviceType":"wmi","SN":"00000001","IMEI":"%%IMEI%%"
,"Timestamp":%%TIME%%,"Input":[%%COUNTER_1%%,%%COUNTER_2%%,%%COUNTER_3
%%,0],"MetricPrefix":"m2m.metering.electricity","Retries":0}
```

The whole message max. can be 255 bytes long.

Available parameters:

"**Id**" - package identifier, automatically increments each time it is sent

"**DeviceType**" - device type

"**SN**" - product serial number (for hardware identification)

"**IMEI**" - module ID (optional to use)

"**Timestamp**" - the time of the pulse value saved on the device

"**Input**" - the value of the measured pulse (counting of input nr. 1, input nr. 2, input nr. 3 - comma separated), the input nr. 4 must be 0 (since there are only 3 available inputs).

"**MetricPrefix**" - free to use field (optional to use)

"**Retries**" - number of data sending retries (currently inactive) - so its value is fixed: 0

For the fields, the parameter assignment is %%VALUE%%. During the real data sending, the value will be changed corresponding to the sent variable.

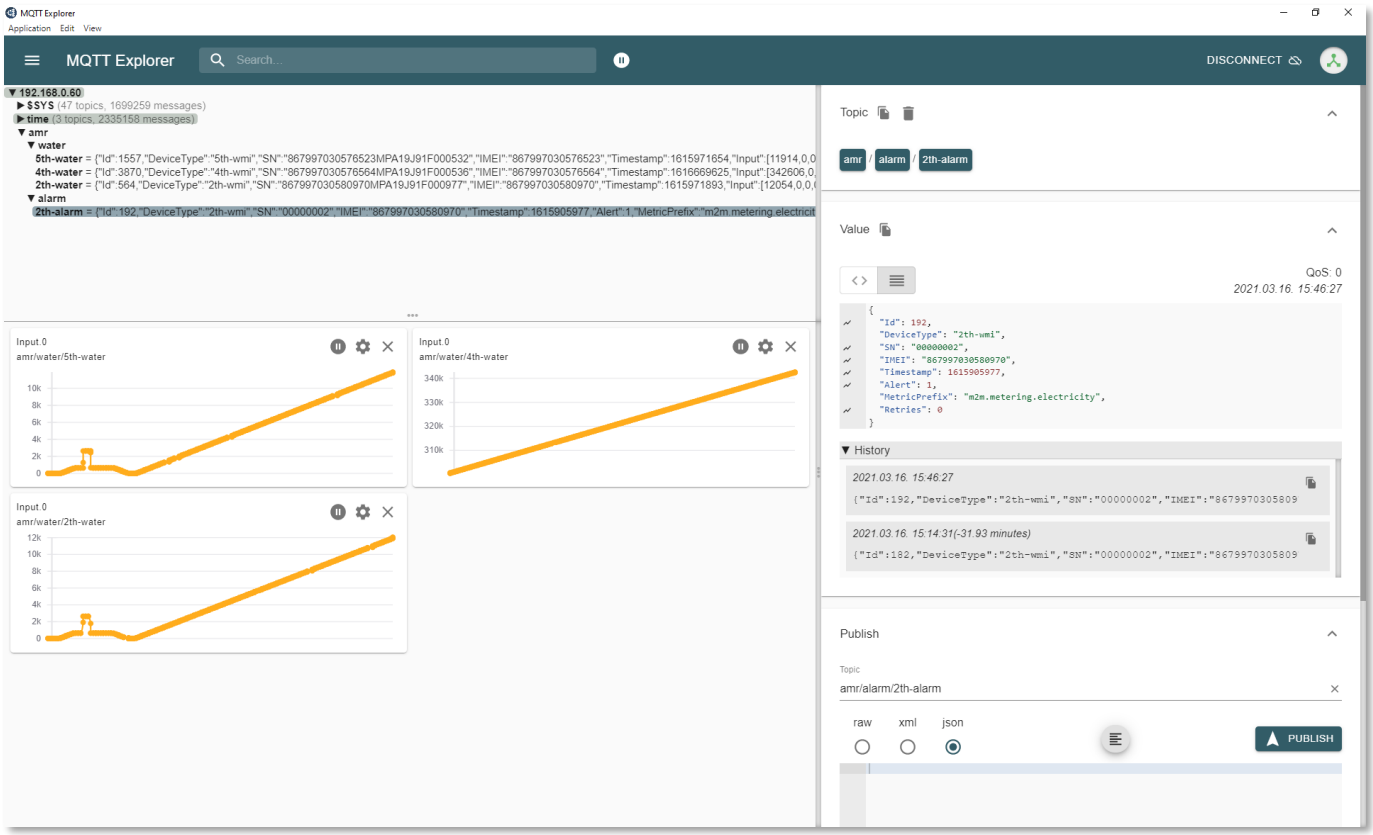
Value assignment for fields:

- "Id":%%PCKT_ID%%" – package identifier
- "DeviceType":"wmi" – device type
- "SN":"00000001" – product serial number (optional to use)
- "IMEI":"%%IMEI%%" – module identifier (optional to use)
- "Timestamp":%%TIME%% - pulse counter value timestamp (which is saved) on the device
- "Input":[%%COUNTER_1%%,%%COUNTER_%%,%%COUNTER_3%%,0] - the value of the measured pulse (input counter nr.1), value of the measured pulse (input counter nr.2), value of the measured pulse (input counter nr.3) and the value 4 must be 0 (since there are no more inputs)
- "MetricPrefix":"m2m.metering.electricity" – free to use field (optional)
- "Retries":0 - number of data sending retries (currently inactive) - optional

8. In the program window, press the **OK** button and press to the **Write Parameters** button in the menu and the device will be send the settings to the device. The *WM-I3®* also sends the template to the MQTT server/address at the next data sending cycle.
Of course, you can also use the **Save** button to save the MQTT template in JSON format.

9. On the MQTT server side, or in the case of testing the data in *MQTT Explorer* – see the next screenshot **-** at the **Value** field (on the right), the MQTT template must be defined in a similar way - in JSON format.

In the **Publish** field, enter the name of the *Topic* (according to the value of the **MQTT_TOPIC** field) and set the format to *json*.



10. After data transmissions, the graphs and numbers will show the amount of counted delta pulses received from the WM-I3 and the consumption tendency, which will be also displayed on the MQTT server side (or in *MQTT Explorer*).

Chapter 3. Support

If you have any questions concerning the usage of the device, contact us at the following contact:

E-mail: iotsupport@wmsystems.hu **Phone**: +36 20 3331111

Online product support can be required here at our website:
https://www.m2mserver.com/en/support/

For the proper identification of your device, use router sticker and its information, which contains important information for the call center.

Due to the support questions, the product identifier is important for resolve your problem. Please, when you are attempting to tell us an incident, please send us the IMEI and SN (serial number) information from the product warranty sticker (located on the front face of the product housing).

The documentation and software release for this product can be accessed via the following link:
https://m2mserver.com/en/product/wm-i3/

Chapter 4. Legal notice

The pictures in the user guide are only for illustration purposes.
WM Systems LLC. does not confirm or accept responsibility for any mistakes in the information contained in the user guide.
The published information in this document is subject to *change without notice*.
All data contained in the user guide is for information purposes only. For further information, please, contact our colleagues.

Warning

Any errors occurring during the program update process may result in failure of the device.