

# Опрос Modbus RTU устройства по Modbus TCP и MQTT через ioThinX 4510

## 1. Настройки последовательного порта

Должны быть такие же как у конечного устройства, подключаемого по каналу RS

The screenshot shows the Moxa ioThinX 4510 web interface. The left sidebar contains a menu with items: Dashboard, System, Security, Network, Module, I/O, **Serial Port** (highlighted with a red box), Internal Register, and Protocol. The main content area is titled 'Port Settings' and is divided into two tabs: 'Port 1' (selected) and 'Port 2'. The 'Port 1' tab contains a form with the following fields: Mode (RS-485 2-Wire), Baudrate (9600), Parity (NONE), Data Bits (8), Stop Bits (1), and Flow Control (None). These fields are grouped within a red rectangular box. Below the form, there is a green 'Service Enabled' button and a note: 'Note: enable/disable this service through [Security Service Settings](#)'.

## 2. Добавление профиля конечного устройства (в том же разделе Serial Port)

The screenshot shows the 'Device Setting' and 'Profile Setting' sections of the Moxa ioThinX 4510 web interface. The 'Device Setting' section has a dropdown menu for 'ModbusRTUDev\_1' and a 'Device ID' field with the value '3'. The 'Profile Setting - Profile-00' section has several fields: 'Profile Name' (Profile-00), 'Start Address' (30026), 'Scan Rate (Unit: 100ms)' (10), 'Point Type' (04: Input Register (R)), 'Length' (1), 'IR type' (WORD), 'IR Start Index' (0), 'Swapped Value' (HighByte & LowByte), and 'Exception Code Setting - WORD IR Index' (—). The 'Start Address', 'Scan Rate', 'Point Type', 'Length', 'IR type', and 'IR Start Index' fields are grouped within a red rectangular box. At the bottom of the 'Profile Setting' section, there is a red button labeled 'DELETE this Profile'.

Важно указать правильные параметры для Modbus RTU – выделены красным.

- Device ID – Modbus RTU идентификатор
- Point type – код функции, которой нужно опрашивать конечное устройство
- Start Address – адрес регистра Modbus RTU устройства с которого начинается опрос
- Length – длина опроса (количество регистров)
- Scan Rate – частота опроса
- IR type – тип данных внутреннего регистра, куда сохраняются данные после опроса
- IR Start Index – стартовый адрес ячейки внутреннего регистра, в который сохраняются данные

Например, у нас в примеры в качестве Modbus RTU устройства выступал [ioLogik R1212](#) - Модуль удаленного ввода/вывода по RS-485: 8 цифровых входов и 8 настраиваемых цифровых входов/выходов.

Modbus RTU идентификатор взяли из настроек ioLogik R1212

R1200 Series RS485 I/O Server			
online			
<span>I/O Configuration</span> <span>Server Info.</span> <span>Server Settings</span> <span>Watchdog</span>			
Address	Value/Status	Access	Description
330026	0x1393	Read	Vendor ID
330027	0x0003	Read	Unit ID for MODBUS/RTU
330101	Moxa Technologies Inc.	Read	Vendor Name
330038	R1212	Read	Product Name
330022	V1.0	Read	Firmware Revision
330024	Build13031519 (03/15/2013)	Read	Firmware Release Date
330059	Run Mode and Dual RS485	Read	Mode of Operation

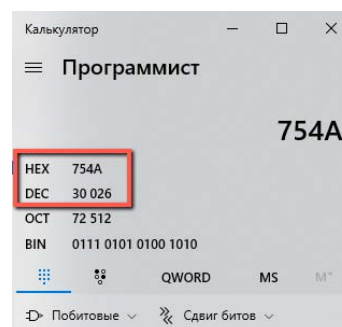
А остальные параметры по Modbus опросу в таблице в документации (страница 44)

[https://moxa.ru/files/manuals\\_ioLogik/iologik-r1200-series-manual-v3\\_4.pdf](https://moxa.ru/files/manuals_ioLogik/iologik-r1200-series-manual-v3_4.pdf)

Для наглядности запрашивали идентификатор по коду функции 04:input register и адресу регистра 0x754A (в формате hex) перевели в десятичный и в ioThinx указали уже его 30026

## ioLogik R1200 System Modbus Address and Register Map

Address (hex)	Register (decimal)	Access	Type	Parameter Name	Function Code	Description
0x7530	330001	R	word	SYS_modelID	04:INPUT REGISTER	
0x7545	330022	R	word	SYS_firmwareVersion	04:INPUT REGISTER	2 words
0x7547	330024	R	word	SYS_firmwareReleaseDate	04:INPUT REGISTER	2 words
0x7549	330026	R	word	SYS_vendorID	04:INPUT REGISTER	
0x754A	330027	R	word	SYS_unitID	04:INPUT REGISTER	
0x754B	330028	R	word	SYS_serialNumber	04:INPUT REGISTER	6 words
0x7555	330038	R	word	SYS_productName	04:INPUT REGISTER	10 words
0x756A	330059	R	word	SYS_rs485PortMode	04:INPUT REGISTER	0: initial-repeater, 1: initial-repeater, 2: initial-repeater



### 3. Включение работы нужных протоколов

No.	Service	TCP/UDP	Port
<input checked="" type="checkbox"/> 1	Web Service via HTTP	TCP	80
<input type="checkbox"/> 2	Web Service via HTTPS <small>MUST import the self-signed certificate before enabling the web service via https, or the browsers may block the connection</small>	TCP	443
<input type="checkbox"/> 3	RESTful API via HTTP	TCP	80
<input type="checkbox"/> 4	RESTful API via HTTPS	TCP	443
<input type="checkbox"/> 5	SNMP Agent/Trap/Inform	UDP	161
<input checked="" type="checkbox"/> 6	Modbus/TCP Slave	TCP	502
<input checked="" type="checkbox"/> 7	Modbus/RTU Master	-	-
<input checked="" type="checkbox"/> 8	MQTT Client	TCP	-
<input checked="" type="checkbox"/> 9	IOexpress/MCC Tool/MXIO	TCP/UDP	10124/4800

Далее проверили, что ioThinX 4510 опрашивает конечное устройство R1212 и записывает данные в свой внутренний регистр. Для этого использовали программу modpoll и команду

**modpoll -m tcp -a 1 -r 3841 -c 1 -t 4 192.168.127.113**

```
C:\WINDOWS\system32\cmd.exe
.:Users\user\Desktop\ПО\modpoll-3.6\win>modpoll -m tcp -a 1 -r 3841 -c 1 -t 4 -i 192.168.127.113
modpoll 3.6 - FreeBASIC(tm) Modbus(R) Master Simulator
Copyright (c) 2002-2018 proconX Pty Ltd
Visit https://www.modbusdriver.com for Modbus libraries and tools.

Protocol configuration: MODBUS/TCP
Slave configuration...: address = 1, start reference = 3841, count = 1
Communication.....: 192.168.127.113, port 502, t/o 1.00 s, poll rate 1000 ms
Data type.....: 16-bit register, output (holding) register table

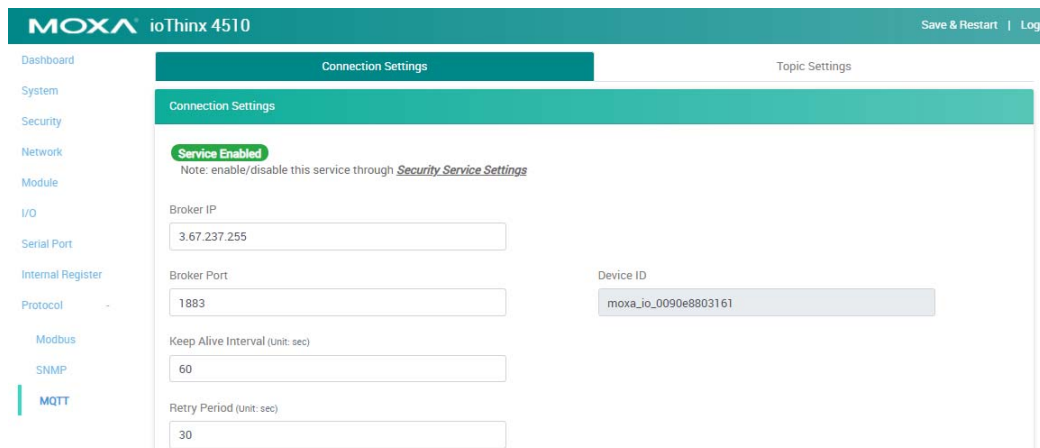
- Polling slave...
3841]: 3
C:\Users\user\Desktop\ПО\modpoll-3.6\win>
```

где 3841 – это стартовый адрес внутреннего регистра WIR, в который и писались данные. Программа modpoll имеет начальный адрес опроса 1, поэтому 3841 вместо 3840 как в настройках.

#	Slot	Module Name	Parameter	Point Type	Start Address (DEC)	Start Register (DEC)	Length	Data Type
1	R-01	45MR-1600-0	diCounterValue	0	1536	401537	8	DWORD
2	R-01	45MR-1600-0	diCounterStatusAll	0	48896	448897	1	WORD
3	R-01	45MR-1600-0	diCounterOverflowFlagClearAll	0	50944	450945	1	WORD
4	R-00	ioThinX4510	frValue	0	4608	404609	128	REAL
5	R-00	ioThinX4510	drValue	0	4096	404097	128	DWORD
6	R-00	ioThinX4510	wirValue	0	3840	403841	64	WORD

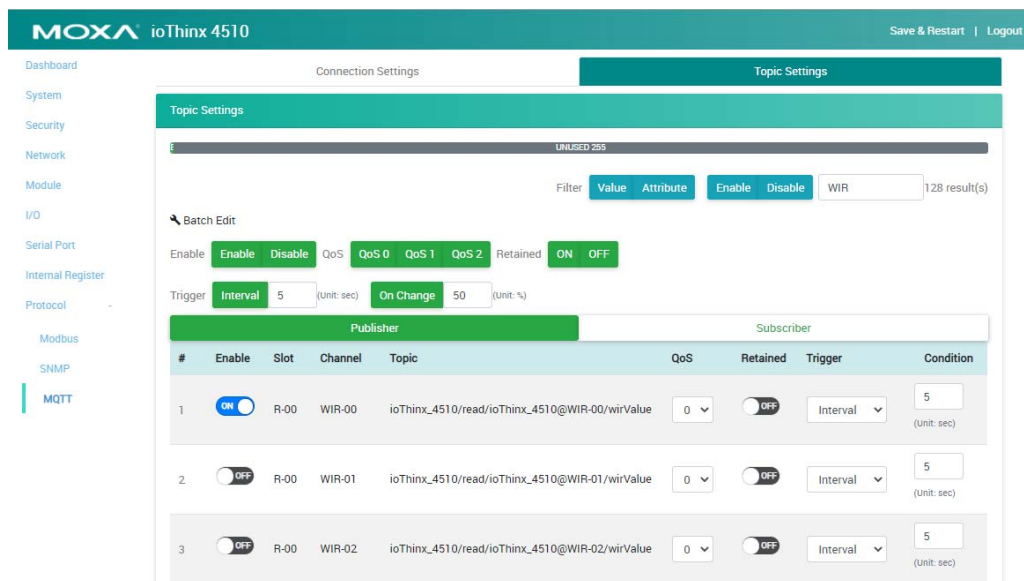
Параметр считался, значит соединение между ioThinx 4510 и R1212 по RS-485 установлено корректно.

#### 4. Настройка MQTT



ioThinx не воспринимает в качестве адреса брокера доменное имя, поэтому вместо broker.hivemq.com указали адрес 3.67.237.255

Включение нужных топиков (формируются автоматически):



Для опроса по MQTT использовали бесплатный брокер <https://www.hivemq.com/public-mqtt-broker/> В веб клиенте отображалось запрашиваемое значение

