ModbusRTU Master
RS-485 or TCP

Упътване за употреба

rev. 1.2

01.07.2025

СЪДЪРЖАНИЕ

Версии на документа

Версия	Дата	Кратко описание на въведените промени
1.02	1.07.2025 г.	Промени в раздел 3.2, свързани с организацията на MQTT съобщенията от v.5.78
1.01	20.05.2025 г.	Цялостна преработка за отразяване на нововъденията от v5.72 (команди за запис)
1.00	7.03.2025 г.	Начална версия на документа

Легенда:

Текстът съдържа допълнителна и полезна информация, която разяснява специфични ситуации и особености.

Текстът съдържа информация от съществена важност, с която непременно трябва да се запознаете!

1. Въведение

В тази специална версия на фърмуера е с вграден модул за комуникация по ModbusRTU протокол с отдалечени устройства, като са възможни два метода на комуникация: TCP или RS-485. За втория е необходимо да разполагате с модел на NetControl, който има RS-485 хардуерен порт (“RS” в означението на модела). TCP режима може да се използва с всеки предлаган модел.

С този софтуерен модул можете да четете/записвате до 16бр. регистри (16 или 32bit) от/към „slave“ устройство. Извлечените стойности се предават през MQTT, а могат да бъдат прикачени към някой от вградените виртуални портове, така че да могат да участват в задачи за автоматизация.

Стойностите, които трябва да се запишат също се подават през виртуалните портове.

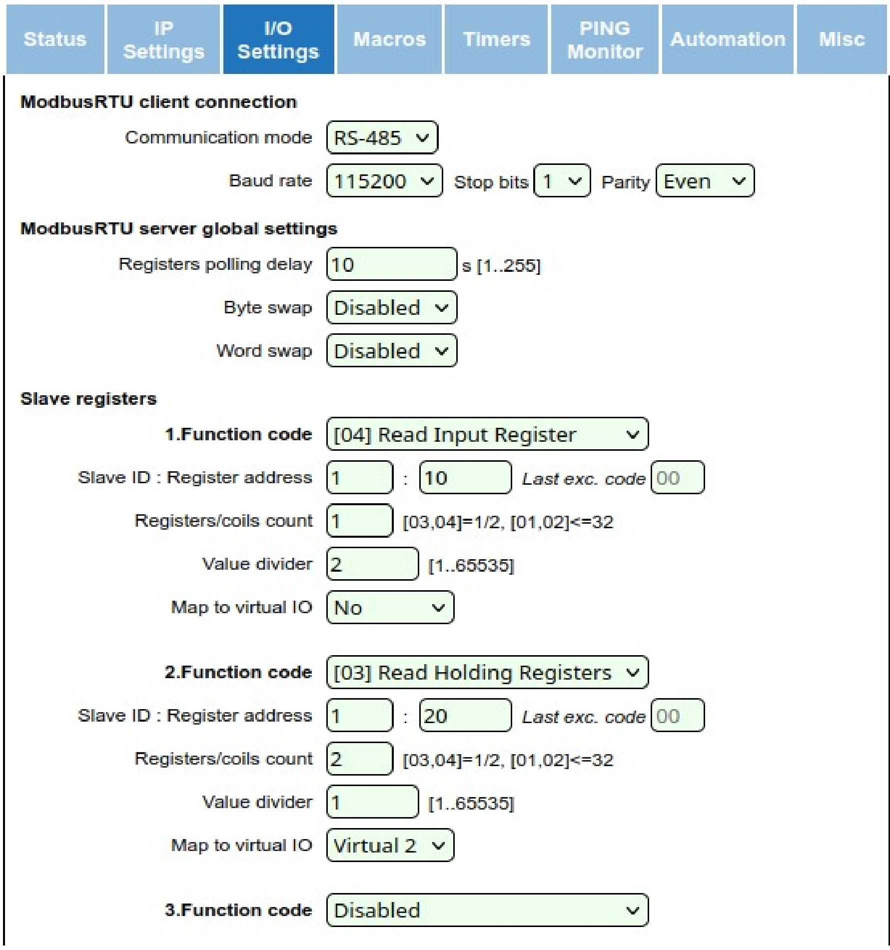
Чрез този софтуерен модул може да използвате NetControl в системи за дистанционно управление и мониторинг на електромери, инвертори за фотоволтаични системи и др.

Тъй-като виртуалните портове могат да се използват от вградения модул за автоматизация (с някои ограничения) може лесно да се реализира типична задача при фотоволтаичните системи: включване на товари при наличие на фотоволтаична мощност над определени прагове или ограничаване на мощността при дадени условия.

2. Конфигуриране на „ModbusRTU Master“

След качване на фърмуера в менюто „IO Settings“ ще се появи линк *„ModbusRTU Master Parameters”*.

Конфигурацията на ModbusRTU модула не се запазва при обновяване на фърмуера. Не се запазва и през командите за запазване на конфигурацията във файл от менюто “Misc”



2.1. Раздел “ModbusRTU client connection“

В тази група са всички общи настройки за софтуерния модул.

След промяна на параметрите в този раздел и “Save parameters” се извършва рестартиране на цялото устройство!

Communication mode	Поддържа се RS-485 или TCP.
Baud rate, Stop bits, Parity	Стандартни настройки за параметрите на серийната комуникация. Трябва да съвпадат с тези на отсрещното (slave) устройство.
IP address / Port	В режим TCP задавате IP адрес (или DNS име) на „slave“ устройството и номер на TCP порт. При установена връзка до полето за порт се появява надпис „CONNECTED“

2.2. Раздел “ModbusRTU server global settings“

В тази група са всички общи настройки за софтуерния модул.

Registers polling delay	Този параметър определя времето на изчакване (секунди) преди да се започне нова последователност от изчитане на зададените адреси.
Byte swap	Разрешаването на тази опция води до размяна местата на байтовете във всяка група от 16 бита.
Word swap	Разрешаването на тази опция води до размяна подредбата на думите от 16 бита във всяка група от 32 бита. Опцията е валидна само при в режимите 03,04,06,16 на Function Code и брой на изчитаните регистри = 2 (32bit).

2.3. Раздел “Slave registers”

Тук можете да зададе до 16 регистъра (които могат да са на различни устройства), които да бъдат периодично изчитани от NetControl.

Function code	Задава се типа команда, съгласно спецификацията на Modbus протокола. В случай, че искате да деактивирате блока с настройки – избирате „Disabled“
Slave ID : Register address	Всяко устройство има адрес (Slave ID) и регистри, достъпни за четене. Можете да четете множество устройства, които са едновременно свързани на RS-485 шината, но трябва всяко от тях да е с уникален Slave ID.
Registers/coils count	Брой на регистрите (1 или 2 по 16bit) или coils/discrete inputs (1 до 32), които искате да изчетете с една команда. Ограниченията се налагат тъй-като крайният резултат е ограничен до 32bit
Value divider/multiplier	Тук можете да зададете коефициент на делене (цяло, положително число), който да се приложи върху данните, получени от командите за четене. За регистри в режим запис – стойността на това поле е множител, който се прилага към подадената във виртуалния порт стойност и тогава се изпраща по Modbus.
Link to virtual IO	Получената стойност (режими за четене) може да се „прехвърли“ вътрешно към избран виртуален порт. Това позволява да се настрои автоматична задача, която да се активира при дадена стойност на Modbus данните. В режимите за запис това поле показа, стойността на кой виртуален порт се взема за запис.
Last exc. code	В това поле се вижда последния код за грешка от комуникацията. 0 = няма грешки и са получени/потвърдени данните.

Номера на регистъра, който задавате тук е и стойността, който се предава по комуникационния канал. Някои устройства и софтуери използват номерация на регистрите, започваща от 1 и това води до отместване на адресите.

3. Указания за използване на модула

3.1. Характеристики и ограничения

Софтуерният модул поддържа само следния набор от команди от Modbus стандарта:

F.code	Команда	Описание
01	ReadCoils	Изчитане на дискретни (0/1) входове до 32 броя. Връща цяло число до 32bit, съдържащо бит маска със стойностите на входовете.
02	Read Discrete Inputs	Аналогично на предходното
03	Read Holding Registers	Изчитане на регистър 16 или 32bit (1 или 2 броя)
05	Write Single Coil	Записване на стойност на дискретни изходи до 32броя. Подава се цяло чисо, което е бит маска на състоянието на изходите
06	Write Single Register	Запис на цяло число (16bit) в единичен регистър.
16	Write Multiple Registers	Запис на цяло число (16bit или 32bit) в един или два последователни регистъра.

Според типа на командата са валидни съответни ограничения в параметъра “Registers/coils count” за всяка от групите с настройки.

Ако изолзвате Modbus модула в комбинация с макроси имайте предвид, че макросите поддържат само 8бит стойност за запис [0...255]. Това означава, че е необходимо да ползвате и параметъра “Value divider/multiplier” ако искате да запишете по-големи стойности към регистъра.

Например, ако искате да урпавлявате мощността на инвертор, който очаква стойности от тип 20000 за 20kW, то може да използвате “Value divider/multiplier=1000” – така всяка стойност записана през макрос към съответния виртуален порт ще съответства на 1kW, т.е. максимум 255kW.

При работа с блока за автоматични задачи имайте предвид, че там ограничението на стойността в полетата LOW/HIGH е 16bit (0...65535). Отново, ако използвате Modbus регистри, които връщат 32bit стойност – използвайте “Value divider/multiplier”, който да Ви даде практична стойност в диапазона до 65535.

3.2. Начин на работа на модула

Настроените регистри в режим за четене (които не са Disabled във “Function code”) се обхождат последователно от първия към последния. Заложено е време за изчакване от 2s за отговор на всяко запитване към даден регистър.

Ако такъв не постъпи се маркира в полето “Last exc. Code” съответната стойност (виж 3.3). За да виждате актуални стойности на това поле е необходимо да презареждате страницата (данните не се опресняват автоматично).

След последователно обхождане на всички зададени регистри се изчаква “Registers polling delay” и се започва ново обхождане от началото.

При свързване с витуален порт НЕ се проверява за дублиране – можете към един витуален порт да свържете няколко Modbus регистъра (няма да може да се разграничава данните от кой източник са), за това внимателно планирайте разпределението на виртуалните портове.

Също така е допустимо един и същ адрес да се изчита няколко пъти, да се използва различен “Value divider/multiplier” и стойностите да се присвоят на различни виртуални портове.

Регистрите, които са в режим на достъп “запис”, трябва задължително да се свържат с виртуален порт, тъй-като записът на стойност в него води до генериране на съответната комнда през Modbus. “Last exc. Code” и тук показва сатуса на командата. Командите за запис се изпълняват веднага и приоритетно спрямо тези за четене, като се изпълняват еднократно.

За всеки регистър (четене/запис) се генерира MQTT съобщение в стандартния JSON формат (показани са само различните полета от JSON съобщениет):

```
{"name" = "Modbus X"
“value.real” = “”
“value.raw” = [32bit unsigned register value]
"channel" = 63 + X
"type" = 252 (MODBUS_MASTER), ...}
```

, където X=1..16 е номера на настройката с регистрите

Полето ‘value.raw’ съдържа стойността след прилагане на “Value divider/multiplier” при запис и преди прилагането му при четене (т.е. реалната modbus стойност към/от регистъра).

Свързаните с виртуални портове регистри генерират и стандартното за виртуален порт MQTT съобщение (т.е. при тези за запис винаги има и съобщение за виртуалния порт), като там вече стойността е преди/след прилагането на “Value divider/multiplier” при запис/четене на регистър.

В “Misc→Events Log” не се записват събитията от този модул, но са налични тези на свързаните виртуални портове. В режим на четене, дори и да не Ви е необходима връзка с виртуален порт, можете да я ползвате временно за диагностика на постъпващите данни.

3.3. Списък стойности за “Last exc. code”

Кодове 00, FB-FE са системни кодове и не са част от спецификацията Modbus. Останалите кодове отговарят на спецификацията на протокола. Възможно е устройства да използват и допълнителен набор от кодове.

00	Получени са коректни данни
-	-
01	Modbus: Illegal function
02	Modbus: Illegal data address
03	Modbus: Illegal data value
04	Modbus: Slave device failure
05	Modbus: Acknowledge
06	Modbus: Slave device busy
07	Modbus: Negative acknowledgment
08	Modbus: Memory parity error
0A	Modbus: Gateway path unavailable
0B	Modbus: Gateway target device failed to respond
FB	Несъвпадащ TID в TCP MBAP хедъра
FC	Получен е отговор от друг Slave ID
FD	Получен е отговор с невалиден формат на днните
FE	Полчен е отгвор с грешен CRC (само за RS485)
FF	Не е получен отговор в допустимото време за изчакване