ModbusTCP FAQ

Introduction

ModbusTCP is a feature on our Color Control GX, and also the Venus GX. ModbusTCP is an industry standard protocol, that can be used to interface PLCs or other equipment with Victron products. For example to enable a heat pump when there is excess solar power, or create a fully integrated glass bridge on a yacht.

Besides reading this FAQ, make sure to also read the ModbusTCP section in our Data communication whitepaper.

Supported ModbusTCP function codes

The supported function codes are:

- ReadHoldingRegisters = 3
- ReadInputRegisters = 4
- WriteSingleRegister = 6
- WriteMultipleRegisters = 16

Note that there is no difference between function code 3 and 4.

Product categories and registers

In the ModbusTCP Registerlist xlsx file, there is a column called 'dbus-service-name'. And it must be consulted to find out which set of registers are available for a product.

Note that besides device-specific registers, there is also a set of registers that represent the entire system: com.victronenergy.system. This service is also where the *System Overview* pages in the GUI, and the VRM Dashboard get their data.

Device class	Product types	Registers
com.victronenergy.vebus	Inverters (VE.Bus), Multis, and Quattros	3 to 60
com.victronenergy.solarcharger	Solar chargers (both VE.Can and VE.Direct connected)	771 to 790
com.victronenergy.pvinverter	Grid-tie PV Inverters (Fronius, or Energy Meter)	1026 to 1039
com.victronenergy.battery	BMVs, CAN-Bus BMS connected batteries Extra parameter set for Lynx Ion and Lynx BMS product family	259 to 319 1282 to 1301
com.victronenergy.charger	Skylla-i and Skylla-IP44	2307 to 2322
com.victronenergy.inverter	Inverters (with a VE.Direct connection)	3100 to 3128
com.victronenergy.tank	Tank levels	3000 to 3005
com.victronenergy.grid	Energy meters configured to measure the grid	2600 to 2609
com.victronenergy.gps	GPS	2800 to 2807
com.victronenergy.genset	Generators	3200 to 3223

Device class	Product types	Registers
com.victronenergy.temperature	Temperature sensors [1]	3300 to 3305

(1): Only the Venus GX is provided with temperature sensor inputs. The battery temperature as measured by a sensor connected to a VE.Bus device is not available. And the battery temperature as measured by a sensor connected to a BMV battery monitor is available under the com.victronenergy.battery register set.

General note: not all list registers are available for each device. For example a BMV Battery Monitor has lots of readings: historical data such as maximum/minimum, for example. A canbus-connected lithium battery might not have those parameters available.

Error codes

Code	Name	Description
0x01	IllegalFunction	The requested function is not supported.
0x02	IllegalDataAddress	The unit-id is available, but one or more of the requested register(s) does not exist.
0x03	IllegalDataValue	The requested quantity of registers is invalid. See the modbus specs, http://www.modbus.org/specs.php, for the limits per function code.
0x0A	GatewayPathUnavailable	Unit-id is defined in the mapping list, but there is no device found on the mapped port. Double check the unit-id mapping list in the excel sheet, and make sure that the device is properly connected to the CCGX, switched on, and that its data is available on the CCGX display.
0x0B	GatewayTargetDeviceFailedToRespond	Requested unit-id not found in the mapping list. Double check the unit id mapping list in the excel sheet.

Mapping example

The purpose of this example is to show how to interpret the information in the worksheets of the excel sheet containing the modbustcp mapping. This excel sheet is available for download from our website on the Color Control page.

To request the input voltage of a Multi or Quattro, connected to the VE.Bus sockets on the CCGX, use the following settings: unit-id 246, and set the data address of the first request register to 3.

The ModbusTCP reply will contain 2302. Divide it by 10, as specified in cell E3, and then you have the voltage: 230.2 Volts AC.

Enabling ModbusTCP on the CCGX

By default, ModbusTCP is disabled. To enable it, go into the settings menu, scroll down to 'Services', and then enable the ModbusTCP service.

What is a unit-id and what do I need to set it to?

The unit-id, sometimes called 'slave address', specifies how a product connected to the CCGX needs to be addressed. See the tab 'Unit ID mapping' tab in the Modbus-TCP excel sheet.

Which products can be interfaced via Modbus TCP?

All products that can be connected to the Color Control GX.

Can I change a configuration setting in a Multi - for example: the absorption voltage?

No.

Can I request multiple registers at the same time?

Yes, you can! Though make sure not to include non existing registers in your query. Querying register 1 to 100 for example will not work, since registers 1 and 2 do not exist.

Can I implement more features myself?

Yes, you can! We are, slowly, making the Color Control code open source, and the Modbus-TCP sources are already available on https://github.com/victronenergy.

In a parallel or three-phase system, how can I address a single unit?

You cannot interface with them individually if they are configured for parallel-working; or as a three-phase system. Neither is it necessary to address units individually as ModbusTCP provides system-totals as well as 'per phase' information.

Will unit-id's change after a reboot or loss of power?

It may change in the case of VEdirect Devices connected via USB - but even then our experience is that it will only change in the first reboot.

Unit-id's for products *not* connected via USB will not change.

Advanced debugging

If all unit-id's and register-addresses have been checked and re-checked, and still the connection doesn't work, you could try looking at the modbustcp log file on the Color Control GX.

Follow these steps:

1. Enable SSH
2. Login: use for example Putty in Windows - or simply type *ssh root@192.168.12.10* on a Linux or Mac terminal console. Replace that ip addresses with the one on the CCGX. Use the password as entered in step 1. (see here for creating the root password).
3. List the latest log file:

```
cat /log/dbus-modbustcp/current | tai64nlocal
```

In case of any errors, the output will look like this:

```
2016-01-08 16:34:24.658248500  INFO 2016-01-08T16:34:24.657 [Server] New
connecion:  "141.138.140.60:41792"
2016-01-08 16:34:24.695846500 ERROR 2016-01-08T16:34:24.689 "Error
processing function code 3, unit id 247, start address 258, quantity 42 :"
"Unit id is available, but start address does not exist"
2016-01-08 16:34:24.753921500  INFO 2016-01-08T16:34:24.753 [Server]
Disconnected:  "141.138.140.60:41792"
```

In most cases the problem will be found in either the unit's id address; or the register addresses. Double check the numbers shown in the output with the numbers as used in your modbustcp software or PLC.

Does Modbus TCP support Hub-4?

Yes. See the Hub-4 advanced control manual for more details.

Disqus

~~DISQUS~~

From:
https://www.victronenergy.com/live/ - **Victron Energy**

Permanent link:
https://www.victronenergy.com/live/ccgx:modbustcp_faq?rev=1522230743

Last update: **2018-03-28 11:52**