

# Serial and USB485

General USB485 and serial information

- [Remote device via eGauge USB485 adapter](#)
- [Using the USB485 converter with Windows 10](#)

# Remote device via eGauge USB485 adapter

The EG4xxx has two USB ports which may be utilized for communicating with third-party serial devices via the eGauge RS485-USB converter (USB485). This effectively replaces the BF-430 used with older model eGauges. In addition, a network connection is not required, as data is read from the third-party serial device through the RS485-USB converter (which is physically connected to the eGauge). At the end of this document is a wiring diagram illustrating the required connections between an eGauge and a single third-party serial device using the RS485-USB converter. Specific wiring diagrams for the commonly used IMT SI-RS485TC irradiance sensor [are also available](#).

The RS485-USB Converter features include:

- Micro-USB interface
- 3-wire RS485 terminal interface (data+, data-, ground)
- Termination switch
- 2 LEDs, to indicate TX and RX activity

image not found or type unknown



The USB ground is tied to the eGauge DC voltage ground terminal. If the DC voltage terminal is wired in reverse, a significant ground potential may form and damage will occur!

# Topics in this article include:

- RS485 Interface Grounding
- Termination
- eGauge configuration
- Configuring additional serial parameters
- Indicator LEDs
- Standard wiring diagram

## RS485 interface grounding

A ground between the RS485-USB converter and serial device's power supply may be required. If the ground potential between the sender and receiver are too far off, the communication voltage can run outside of acceptable limits resulting in communication degradation or even damage to connected equipment.

It is recommended to connect the ground connection before the data cables.

If the serial device is powered via one of the eGauge USB ports, the ground connection is not needed.

## Termination

The RS485-USB Converter has a termination toggle switch. This should be enabled if the converter is the last device on the chain, or if there is only a single serial device connected. In most cases this termination switch should be in the ON position. Failure to enable the termination switch may

result in communication degradation or failure.

# eGauge Configuration

The configuration for an eGauge to communicate with a remote serial device using the RS485-USB Converter is similar to configuring other remote devices. Navigate to Settings -> Installation and add a new remote device using protocol "Serial". The device address is either USB1 or USB2, depending on the USB input in use. All other remote device configuration is standard, and more information on the device address syntax for custom Modbus definitions can be found in the [third party device section of eGauge.net](#).

Device name:	Protocol:	Device address:	
<input type="text" value="IMT"/>	<input type="text" value="Serial"/>	<input type="text" value="modbus://imt_si.1@USB1"/>	<input type="button" value="Edit"/>

*Example remote device address for a Modbus IMT irradiance sensor with serial address 1 via RS485-USB converter in USB port 1*

## Configuring additional serial parameters (i.e., baud rate, data bits, parity, stop bits)

If the default serial parameters for a supported device change, or if an unsupported Modbus device with in-line definitions is used, serial parameters may need to be defined in the device address. To specify serial parameters, append to the end of the remote device address (do not include brackets):

:[BAUD\_RATE]/[DATA\_BITS][PARITY][STOP\_BITS]

For example, `modbus://imt_si.1@USB1:19200/8e2` will look for a Modbus IMT sensor with serial address 1 on USB port 1, with 19200 baud, 8 data bits, even parity, and 2 stop bits.

Similarly, `modbus://temperature=2,s32,degC.3@USB2:1200/8n1` will read a signed 32-bit (s32) temperature value from register 2 from a Modbus device at serial address 3, connected via an RS485-USB converter in USB port 2, running at 1200 baud, 8 data bits, no parity, and one stop bit.

Parity options are 'e' for even, 'n' for none, and 'o' for odd.

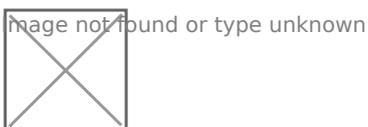
## Indicator LEDs

The eGauge RS485-USB Converter has 2 indicator LEDs. The left-side green LED flashes to indicate serial data was received (responses from the remote serial device), while the right-side red LED flashes to indicate data was received from the eGauge (requests from the eGauge to the serial device).

If there is only activity on the red LED, and no green LED flashes, it is an indication that the serial device is not responding or is not receiving the requests from the eGauge. This may be due to incorrect wiring or incorrect configuration of the eGauge remote device address, or of the serial device itself.

## Standard wiring diagram

Below is the standard wiring diagram for an eGauge communicating to a single serial device via the RS485-USB Converter. An external power supply for the serial device is generally needed, but if the serial device can be powered via USB, an unused USB port on the eGauge may be connected for the power, and voltage and ground wires in the diagram ignored. It is recommended to connect the ground connection before the data cables, unless the serial device is powered from one of the USB ports on the eGauge.



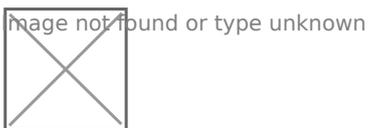
# Using the USB485 converter with Windows 10

The eGauge USB485 converter may be used as a standard serial-to-USB converter on a computer to read data from an eGauge or communicate with other third party devices besides the eGauge. The eGauge USB485 uses the FT230X chipset from FTDI. Up to date systems should already have the FTDI drivers available for use. If the FTDI drivers are not installed, drivers and information are available from the FTDI website at <http://www.ftdichip.com/FTDrivers.htm>. COM port downloads can be found at <https://www.ftdichip.com/Drivers/VCP.htm>.

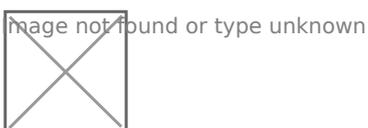
This article describes how to verify the USB485 converter is recognized by a Windows 10 machine, and uses QModBus to show data being read from an eGauge via two USB485 converters.

Note that support for third party devices is not guaranteed.

1) Connect the eGauge USB485 via USB to the Windows 10 machine. After a minute or several, a notification indicating the serial converter drivers are ready will be displayed at the bottom right-hand corner. If this does not appear after 5 minutes, verify the USB cable is secure and move to the next step.

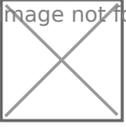


2) Open the device manager by pressing the Windows Key and R (Win+R) to open the "Run" prompt. Alternatively open the start menu, and type "run" and press enter to open the run prompt. Enter devmgmt.msc in the text box and press "OK":



3) Under "Universal Serial Bus controllers" you should see "USB Serial Converter":

image not found or type unknown



---

4) You can determine the COM port by expanding "Ports (COM & LPT)". Here, it is COM3:

image not found or type unknown



---

5) The QModBus program may be installed from <http://qmodbus.sourceforge.net>. eGauge Systems does not endorse and cannot validate the legitimacy of the QModBus program. When the QModBus program is installed and executed, if no serial converters are found it will warn you:

image not found or type unknown



---

6) If a serial converter is detected, the QModBus will default to use one that is found (in our case, COM3). Here, we have configured the serial settings to match with an eGauge Modbus slave connected via serial to this Windows computers USB485 converter. In the screenshot below, we read registers 1012 and 1013, as DC voltage is a 32-bit float starting at address 1012. When converting 0x4143 and 0xB07B as a big endian float, we see the DC voltage is about 12.23.

image not found or type unknown

