

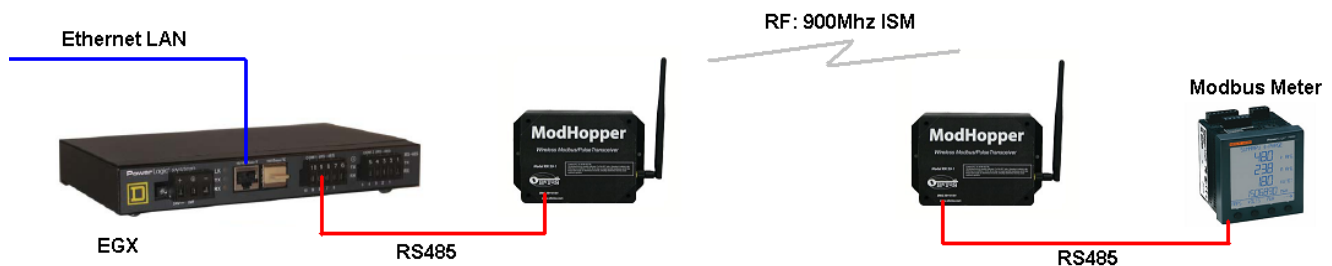
# Technote 19 – Using the POWERLOGIC® EGX with ModHoppers.®

Author: Stephen Herzog  
Copyright 2008, by Obvius Holdings LLC  
Revised: January 30, 2009

This document discusses the use of Obvius ModHopper with the POWERLOGIC EGX Ethernet gateway from Schneider Electric.

## Concept:

The POWERLOGIC EGX provides an Ethernet to RS485 Modbus gateway service. The Obvius ModHopper serves as a wireless Modbus Mesh network and can connect multiple Modbus meters together with a Modbus master system. The ModHopper can be used to extend the RS485 Modbus terminal on the EGX with remote Modbus meters via a wireless connection.



## Dual master problem:

The ModHopper provides a wireless mesh network to relay Modbus queries and replies between two devices attached to the system. To be able to route Modbus queries to the correct destination, the ModHopper must know what Modbus devices are attached to the RS485 port. An inventory of attached meters is built by sending out queries to all Modbus addresses to see what devices respond. When a device responds, the address is added to the ModHopper routing table.

On the ModHopper that is connected to the Modbus Master system, the ModHopper will detect queries from the master, and automatically switch from auto-detect mode into slave mode. All further queries from the Modbus master are forwarded as needed through the ModHopper mesh network.

When the ModHopper is attached to the EGX, several things happen. First, the ModHopper starts sending queries. The EGX receives the queries and responds with a Modbus Exception, indicating a gateway configuration error. This reply is received by the ModHopper. Because the Modbus exception response is a valid Modbus reply packet, the ModHopper adds the address to the routing table as a local RS485 attached device.

The following shows the Modbus queries and responses in hex format.

```
05 03 00 00 00 01 85 8E    (ModHopper query for address 5)
05 83 0A 80 F6             (EGX response, address 5, Gateway Configuration Exception)

06 03 00 00 00 01 85 BD    (ModHopper query for address 6)
06 83 0A 70 F6             (EGX response, address 6, Gateway Configuration Exception)

07 03 00 00 00 01 84 6C    (ModHopper query for address 7)
07 83 0A 21 36             (EGX response, address 7, Gateway Configuration Exception)
```

Within a few minutes, every Modbus address between 1 and 255 are added to the ModHopper routing table for the local RS485 port. This false information prevents the ModHoppers from properly routing Modbus queries to the proper destination. In addition, the EGX does not appear to forward any Modbus/TCP queries through the RS485 port, most likely due to having detected a Modbus master on the port.

In this situation, the system will not function properly.

### ***Proper ModHopper configuration:***

The ModHopper has a feature that allows configuration of the probe/detect feature. This can be set to one of three modes.

- **Automatic:** On power up, the ModHopper sends queries to locate attached Modbus slave devices. If a Modbus query from another Master device is received by the ModHopper, the ModHopper switches into slave mode. The ModHopper will stay in slave mode as long as a query from another Master device is received by the ModHopper at least every 10 minutes. Automatic mode is the default setting on the ModHopper.
- **Force-slave:** The ModHopper does not send queries on the RS485 port. It only processes queries sent from an external Modbus master device.
- **Force-master:** The ModHopper will send queries to search for attached Modbus devices. If a query is received from an external Modbus master device on the RS485 port, it will be ignored.

The default mode of “automatic” is incompatible with the EGX. To use the ModHopper with the EGX, the ModHopper must be configured for “force-slave” mode.

Note: the force-slave mode should only be configured for the ModHopper that is directly wired to the RS485 port of the EGX. All other ModHoppers in the system may be left on the default setting of automatic.

To configure the ModHopper for force-slave mode, a value of 2 must be written to register 40143.

The ModHopper must also have firmware version 1.17 or later installed to support this feature. If the ModHopper has an earlier firmware version, please contact Obvius technical support for upgrade instructions and firmware.



The following are three ways to configure this register.

Method 1: Use a third party windows software package such as Modscan32. Connect the ModHopper to the windows computer using an RS485 adapter. Use the Windows software to read holding register 40143 to confirm communications. The value read should be 0, the default setting for this feature. Use the preset holding register function to write a value of 2 to register 40143. Note: it does not appear to be possible to reconfigure this register with a Modbus/TCP query through the EGX to the ModHopper when the ModHopper is not already properly configured. Once the ModHopper has been configured for force-slave mode, all r/w register functions may be configured using Modbus/TCP software and the EGX.

Method 2: Attach the ModHopper to an AcquiSuite. When the ModHopper appears in the Modbus device list of the AcquiSuite, click the device address number, click Configure, and finally, click the Advanced Configure button. In the advanced configuration page, locate the RS485 port mode. Select “force-slave” from the dropdown menu, and click the “Save Advanced” button.

Method 3: ModHopper firmware v2.02 and later can modify this register with a combination of dipswitch settings on power up. Turn off all the dipswitches, and remove power from the ModHopper. Turn on Modbus 128, RF Channel 2, and Reserved. Power up the ModHopper. The TX485, Alive, Pulse 1, and Pulse 2 LEDs should start blinking together. Remove power from the ModHopper. Return the dipswitches to the desired positions, and turn OFF the reserved switch. On power up, register 40143 should be set to 2. Please read the full instructions on using the ModHopper Dipswitch Mode option for further information on this feature.

Baud Rate: The ModHopper has 9600 and 19200 baud rate options. Either baud rate may be used, as long as the EGX is configured for the same rate as the ModHopper. Use 19200 to improve performance if possible.

### ***Confirm proper ModHopper configuration:***

Power up the ModHopper with no external devices attached to the RS485 port. Observe the RS485 TX LED. The RS485 TX LED should not blink. If the RS485 TX LED is blinking at a rate of 1-2 times per second, the ModHopper is in either automatic or force-master mode. Note: you must power on the ModHopper with nothing attached to perform this test properly.

### ***EGX Configuration:***

The EGX 100, 200, and 400 have numerous features. Specifying all configuration options are beyond the scope of this document. Options noted below are specific to using the ModHopper with the EGX only. Consult the EGX user manual for further instruction.

Configure the EGX serial port options as follows:

- Baud Rate: 19200 (or 9600). This must match the ModHopper baud rate setting.
- Parity: none
- Mode: 2 wire



Configure the EGX Device List as follows:

- If the local ModHopper is attached to COM1: add the address of the attached ModHopper to the EGX device list, and specify the protocol as Modbus.
- If the local ModHopper is attached to COM2: add the address of the attached ModHopper and every remote device in the ModHopper network to the EGX device list. Specify the protocol as Modbus. This includes all Modbus meters that are attached to remote ModHopper RS485 ports.

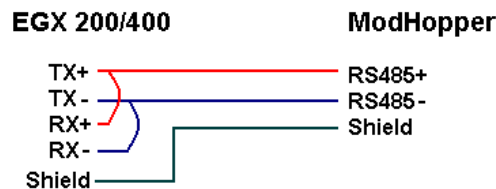
Configure the EGX Advanced configuration page:

- Timeout: The RS485 timeout for COM1 (or COM2) should be set long enough for queries to be relayed by the ModHopper to the remote Modbus device, and for the reply to return round trip. Typically, this should be set to 500ms for each radio hop through the ModHopper network, in addition to the timeout required for the remote Modbus device. For example, an Enercept meter typically needs a timeout of 200ms. If there are two ModHopper hops between the Enercept and the EGX, the timeout should be  $500\text{ms} + 500\text{ms} + 200\text{ms} = 1200\text{ms}$ .
- Number of viewable devices: this number may need to be increased depending on the total number of ModHopper and remote Modbus devices that are attached.

Note: if the ModHopper has been attached to the EGX prior to updating the ModHopper to force-slave mode, you may be required to reboot the EGX. It is recommended that after both the EGX and ModHopper have been properly configured, both are rebooted to clear any invalid data in the routing tables of both products.

## **Wiring:**

The ModHopper is a 2 wire RS485 device. When connecting the ModHopper to the EGX, the following wiring hookup should be used.



Wiring to the EGX 100 is slightly different due to the 2/4 wire configuration option. Refer to the EGX 100 install guide for 2 wire RS485 connections.