TABLE OF CONTENTS

TRADEMARK INFORMATION

LON, LONWORKS, and Neuron are trademarks of Echelon Corporation registered in the United States and other countries.

APPLICABLE PRODUCT LITERATURE

The following is a list of documents containing information related to the EAGLE Controller that are available on the Centraline Dokuserver and on the E-catalogue.

Title	Literature Number
EAGLE – M-Bus and Modbus White List	EN2Z-0995GE51
EAGLE – Networking White Paper	EN2Z-0992GE51
EAGLE – Product Data	EN0Z-0970GE51
EAGLE – Installation and Commissioning Instructions	EN1Z-0970GE51
EAGLE – Web Interface User Guide	EN2Z-0970GE51
EAGLE – Onboard HMI User Guide	EN2Z-0971GE51
EAGLE – Mounting Instructions	MU1Z-0970GE51
IF-LON – Mounting instructions	MU1B-0545GE51
IF-LON – Installation Instructions	MU1B-0538GE51
EAGLE External HMI – Product Data	EN0Z-0988GE51
EAGLE External HMI – Mounting Instructions	MU1Z-0988GE51
EAGLE – BACnet PICS (Protocol Implementation Conformance Statement)	EN0Z-0978GE51
EAGLE – LON-Bus Traffic Calculator	Excel Web_LonBus_Traffic_Calc_0.2.xls
EAGLE – External HMI Guide Specification	CLEAHMI21_GuideSpecification_english_v1.doc
EAGLE – Guide Specification	EAGLE_GuideSpecification_english_v7.doc

RS485 INTERFACES

The RS485 Standard

According to the RS485 standard (TIA/EIA-485: "Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems"), only one driver communicating via an RS485 interface may transmit data at a time. Further, according to U.L. requirements, each RS485 interface may be loaded with a max. of 32 unit loads. E.g., CentraLine devices have as little as ¼ unit load each, so that up to 128 devices can be connected.

BACnet MS/TP connections to the RS485 interfaces must comply with the aforementioned RS485 standard. Thus, it is recommended that each end of every connection be equipped with one termination resistor having a resistance equal to the cable impedance (120 Ω / 0.25 – 0.5 W).

RS485 systems frequently lack a separate signal ground wire. However, the laws of physics still require that a solid ground connection be provided for in order to ensure error-free communication between drivers and receivers – unless all of the devices are electrically isolated and no earth grounding exists.

IMPORTANT

In the case of new EAGLE controller installations, we strongly recommend using a separate signal ground wire. Doing otherwise may possibly lead to unpredictable behavior if other electrically non-isolated devices are connected and the potential difference is too high.
In the case of the installation of EAGLE controllers in already-existent RS485 two-wire systems (e.g., when replacing PANTHER or LION controllers with EAGLE controllers), not using a separate signal ground wire will probably have no undesirable effects.

The cable length affects the baud rate. The following table provides a few examples.

Baud rate	**Max. cable length (L)**
9.6 - 76.8 kbps	1200 m
*115.2 kbps	1000 m
* In the case of configuration of RS485-2 for Panel Bus, the communication rate is set to 115.2 kbps.	

Table 1. Baud rate vs. max. cable length for RS485

For information on wire gauge, max. permissible cable length, possible shielding and grounding requirements, and the max. number of devices which can be connected to a bus, see following pages, and refer to standard EIA-485.

EIA 485 Cable Specifications and Wiring

The following cable specification is valid for all EIA 485 busses (e.g., BACnet MS/TP, Modbus RTU, and Panel Bus)

max. length	1000-1200 m, see Table 1 above.
cable type	twisted pair, shielded (foil or braided shields are acceptable)
characteristic impedance	100…130 Ω
distributed capacitance between conductors	Less than 100 pF per meter (30 pF per foot)
distributed capacitance between conductors and shield	Less than 200 pF per meter (60 pF per foot)

Table 2. EIA 485 cable specifications

The following cables fulfill this requirement:

- AWG 18
- Shielded, twisted pair cable J-Y-(St)-Y 2 x 2 x 0.8
- CAT 5,6,7 cable (use only one single pair for one bus)
- Belden 9842 or 9842NH

Wiring topology

- Daisy-chain topology, only.

Max. number of BACnet MS/TP devices

- See section "BACnet MS/TP" on pg. 8.

RS485-2 (non-isolated)

- Wiring should not extend beyond a single building.
- Ground noise should not exceed the EIA-485 common mode voltage limit

Routers

- MS/TP routers conforming to the EIA-485 standard are allowed.

LEDs

The EAGLE Controller features the following LEDs:

Fig. 1. EAGLE Controller LEDs

symbol	color	function, description
L1	yellow	Application-specific LED indicating status information ("Cooling Mode", "Heating Mode" "Service Interval", etc.)
L2	--	Reserved.
Tx	yellow	RS485-1 status LED indicating transmission of communication signals.
Rx	yellow	RS485-1 status LED indicating reception of communication signals.
⚠	red	status LED indicating hardware problems, lack of application, sensor failure, or Panel Bus failure
⏻	green	power LED

Table 3. EAGLE Controller LEDs

LED behavior	meaning	remedy
Both Tx and Rx are flashing	Normal operation; RS485-1 is functioning properly.	No action necessary
Both Tx and Rx are OFF	No communication on RS485-1.	Switch ON communication on RS485-1. L1 should then flash. Further handling like case 4 (below).
Rx is flashing and Tx is OFF	Communication on RS485-1 has been switched OFF, but the EAGLE is receiving data from other controllers.	Switch ON communication on RS485-1. If this proves unsuccessful, the hardware may be defective.
Tx is flashing and Rx is OFF	The EAGLE Controller is attempting to establish communication on RS485-1, but there is no answer.	The communication rate (Kbaud) on RS485-1 has not been correctly set; other controllers on the bus may have been incorrectly assigned the same device number; wiring problem or hardware defect.

Table 4. Tx and Rx LED behavior

EAGLE RS485 INTERFACES

General

The EAGLE Controller features two RS485 interfaces:

- RS485-1 (consisting of push-in terminals 24 [GND-1], 25, and 26) is isolated and can be used for BACnet MS/TP bus, Panel Bus, or Modbus RTU Master communication.
  The RS485-1 interface provides a switch for termination and bias selection. See Fig. 2 below.
- RS485-2 (consisting of push-in terminals 29, 30, and 31 [GND-2]) is non-isolated (i.e., GND-2 is internally connected with terminal 1 [24V~0] and terminals 19+37 [system ground]) and can be used for BACnet MS/TP bus, Panel Bus, or Modbus RTU Master communication.
  The RS485-2 interface does NOT have a switch for bias and termination. Thus, the bias and termination resistors are always ON (like for RS 485-1 "End position").

Restriction Applying to CLEA2014B21 and CLEA2014B31

In the case of the CLEA2014B21 and CLEA2014B31, CARE automatically assigns a minimum of one of the two RS485 interfaces to BACnet MS/TP. Although the user has the option of shifting this assignment from the automatically assigned RS485 interface to the other, the user cannot alter the fact that a minimum of one of the two RS485 interfaces will be assigned to BACnet MS/TP. Thus, in the case of the CLEA2014B21 and CLEA2014B31, the total max. no. of Panel Bus I/O modules is reduced from 128 to only 64.

RS485-1

RS485-1 Bias and Termination Resistors

RS485-1 is equipped with a three-position slide switch which can be used to switch its bias resistors OFF (position "MID" – this is the default), switch its bias resistors ON (position "BIAS"), and switch its bias resistors on with an additional 150Ω termination resistor (position "END").

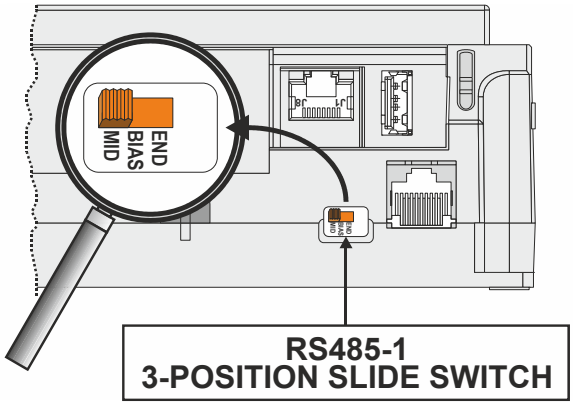


Fig. 2. RS485-1 three-position slide switch

The recommended slide switch setting depends upon the location and usage of the given EAGLE – see Fig. 3 through Fig. 5; it also depends upon the selected communication protocol (BACnet MS/TP, Panel Bus, or Modbus RTU Master communication, respectively).

setting	remarks
END	Termination = ON, Bias = ON Controllers located on either end of bus should have this setting.
BIAS	Termination = OFF, Bias = ON In small bus networks, a min. of one and a max. of two controllers should have this setting.
MID	Termination = OFF, Bias = OFF All other controllers (not set to "END" or "BIAS") on bus should have this setting (which is the default).

Table 5. Recommended slide switch settings

NOTE: According to BACnet standards, a minimum of one and a maximum of two BACnet devices must have its/their bias resistors switched ON. In the case of the RS485-1 interface of the EAGLE, setting its slide switch to either "BIAS" or "END" fulfills this requirement.

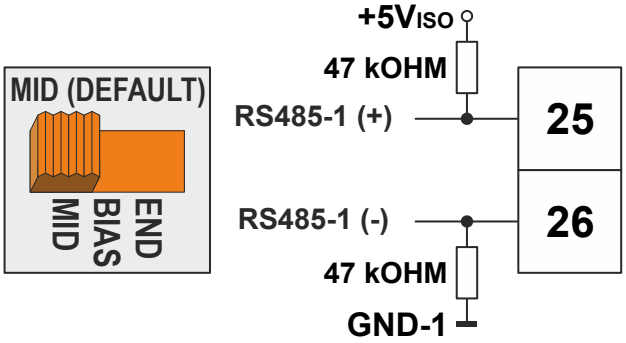


Fig. 3. RS485-1 three-position slide switch setting MID

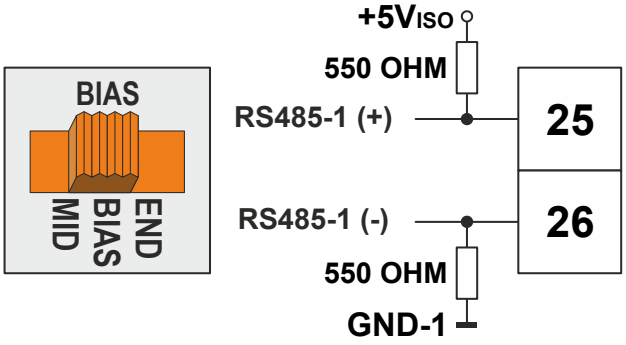


Fig. 4. RS485-1 three-position slide switch setting BIAS

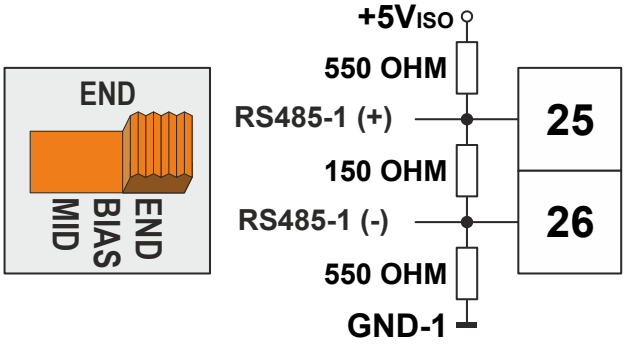


Fig. 5. RS485-1 three-position slide switch setting END

NOTE: All terminals are protected (up to 24 Vac) against short-circuiting and incorrect wiring – except when the 3-position slide switch is set to "END," in which case the terminals of the RS485-1 bus (24, 25, and 26) have no such protection. Higher voltages may damage the device.

RS485-2

RS485-2 Bias and Termination Resistors

The RS485-2 interface is not affected by the aforementioned three-position slide switch. The 550Ω bias resistors and 130Ω termination resistor of the RS485-2 are thus always ON.

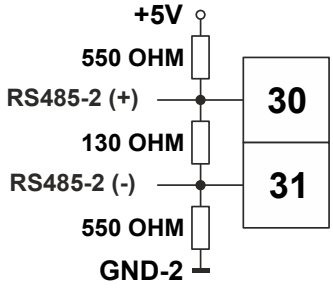


Fig. 6. RS485-2 bias and termination resistors

NOTE: GND-2 is internally connected with 24V-0 (terminal 1) and system GND (terminals 19+37)

BACNET MS/TP

BACnet MS/TP devices can be connected to RS485-1 (consisting of push-in terminals 24 [GND-1], 25, and 26) and/or RS485-2 (consisting of push-in terminals 29, 30, and 31 [GND-2]).

The max. number of BACnet MS/TP devices per RS485 interface depends on the given traffic. The current limit is 450 messages per minute and per channel. The max. number of room controllers is 45, assuming a typical traffic of 10 messages per minute and per channel.

BACnet MS/TP on RS485-1

Powering, Bias and Termination

With regards to Fig. 7 through Fig. 9, please note the following:

NOTE: Always power each EAGLE and the connected BACnet MS/TP devices via separate transformers.

NOTE: For the length of the RS485 bus (marked "L" in the following figures), see section "The RS485 Standard" on pg. 3.

NOTE: If any of the devices are electrically isolated, it is recommended that those devices be connected to signal ground. See section "The RS485 Standard" on pg. 3.

Example 1: Single EAGLE Controller and Connected BACnet devices (with inserted termination resistor)

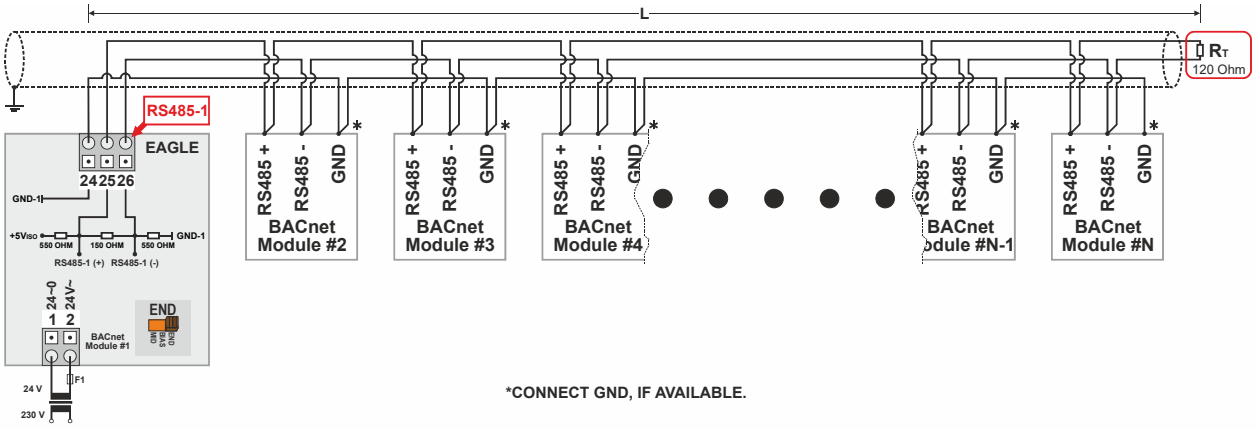


Fig. 7. Connection of RS485-1 to a BACnet MS/TP Bus

The termination resistor must be inserted directly into the terminals of the last BACnet MS/TP device.

Example 2: Multiple EAGLE Controllers and Connected BACnet devices

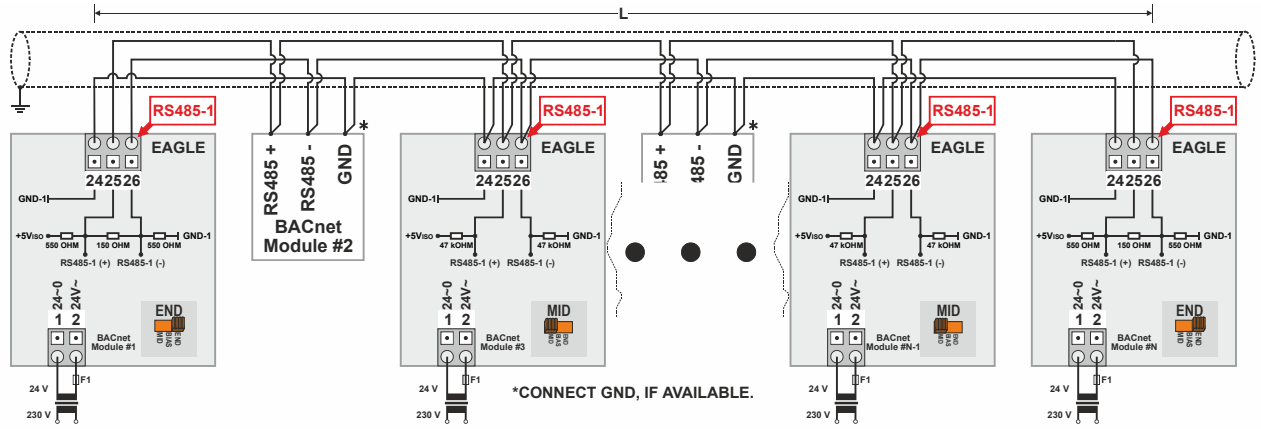


Fig. 8. Connection of RS485-1 to a BACnet MS/TP Bus

Example 3: Multiple EAGLE Controllers and Connected BACnet devices (with inserted termination resistor)

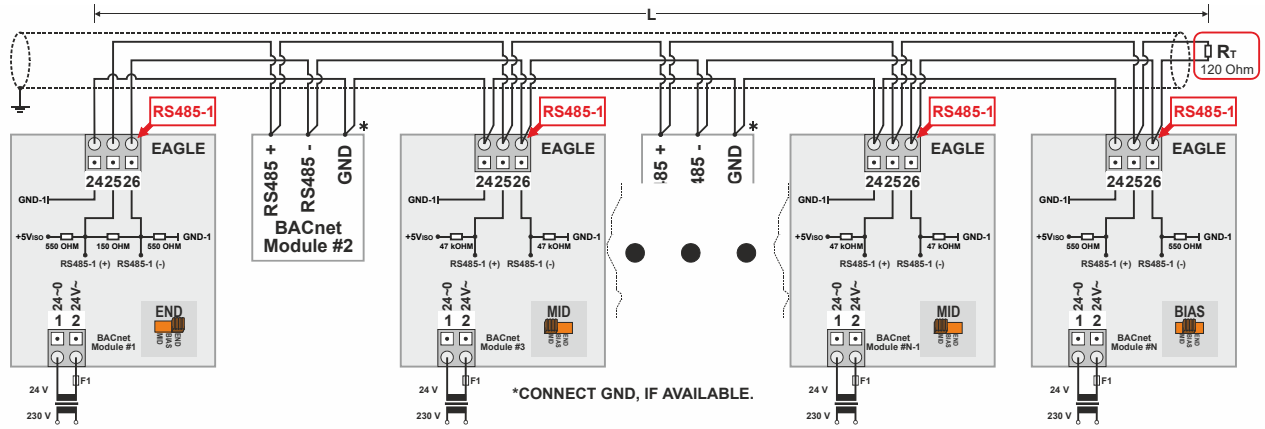


Fig. 9. Connection of RS485-1 to a BACnet MS/TP Bus

The termination resistor must be inserted directly into the terminals of the last BACnet MS/TP device (in this example, that is the rightmost EAGLE, the 3-position slide switch of which has been set to "BIAS.")

BACnet MS/TP on RS485-2

NOTE: GND-2 is internally connected with 24V-0 (terminal 1) and system GND (terminals 19+37)

Powering, Bias, and Termination

With regards to Fig. 10 and Fig. 11, please note the following:

NOTE: Always power each EAGLE and the connected BACnet MS/TP devices via separate transformers.

NOTE: For the length of the RS485 bus (marked "L" in the following figures), see section "The RS485 Standard" on pg. 3.

NOTE: If any of the devices are electrically isolated, it is recommended that those devices be connected to signal ground. See section "The RS485 Standard" on pg. 3.

NOTE: Between devices equipped with non-isolated RS485 bus interfaces, potential differences of max. ±7 V are allowed. Further, this bus should not extend beyond a single building.

Example 1: Single EAGLE Controller and Connected BACnet Devices (with inserted termination resistor)

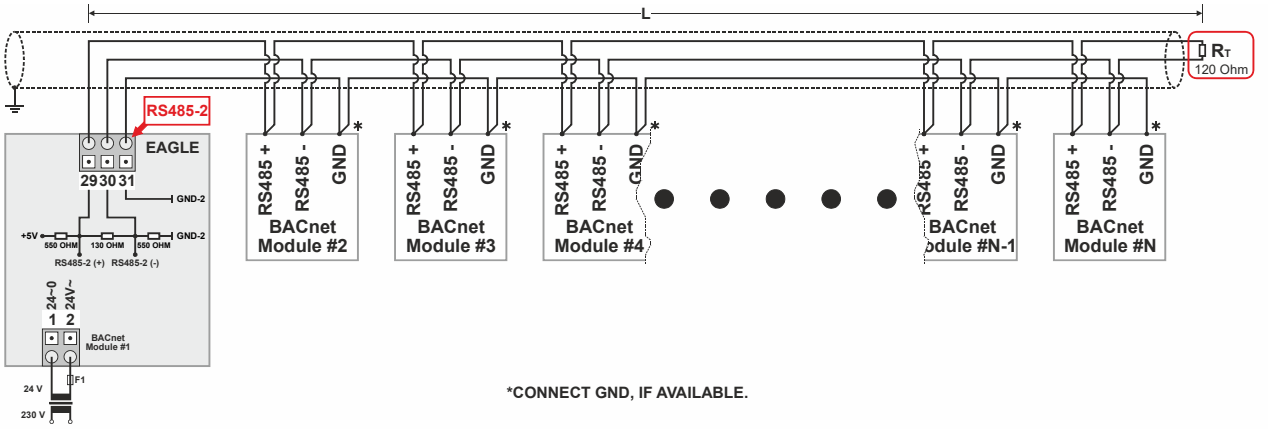


Fig. 10. Connection of RS485-2 to a BACnet MS/TP Bus

The termination resistor must be inserted directly into the terminals of the last BACnet MS/TP device. BIAS is not required because this is automatically done on the first EAGLE.

Example 2: Multiple EAGLE Controllers and Connected BACnet Devices

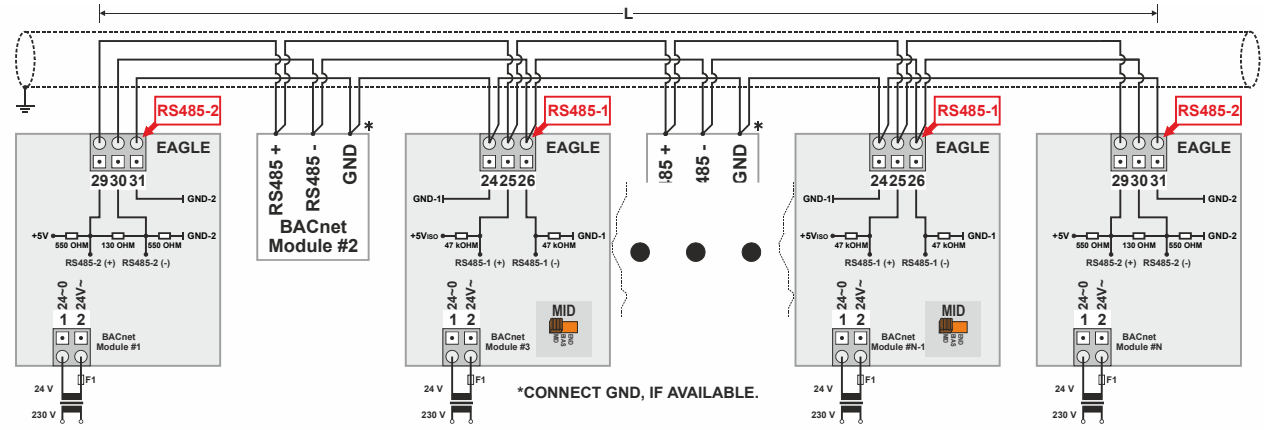


Fig. 11. Connection of RS485-2 to a BACnet MS/TP Bus

BACnet MS/TP Addressing and Wiring Rules

1) Only use BACnet MS/TP Master devices. BACnet MS/TP slaves are not supported.
2) Device ID:
   - Must be unique system-wide.
3) MAC address
   - Can be from 0 to 127.
   - Per device, must be unique on the same BACnet MS/TP Network (network number).
   - Must be sequential on the same BACnet MS/TP network (network number), and without gaps:
     Example: 1, 2, 3, 4, 5, 6, 7 Do NOT do this: 1, 2, 4, 5, 6, 7. In the event that you are using your EAGLE controller as a router to MS/TP and are using MERLIN controllers as MS/TP controllers, do NOT use the MERLIN controller's automatic MAC address assignment option.
   - Per RS 485 channel EAGLE should have the highest MAC address.
   - Dip switch setting must be correct (depending on its physically connection) for every BACnet MS/TP device.
4) Max Master setting
   - Set in CARE under "Max Masters" the maximum number of the devices that are connected per RS485 channel to EAGLE.
   - The BACnet MS/TP controller with highest MAC Address should support Max Master setting in order to prevent polling up to 127, which would slow down the bus communication.

NOTE: Unitary controllers and field devices like LYNX and SmartDrive typically do not support a Max Master setting. Please refer to the following application examples 1 through 4 for the necessary actions to compensate for this issue.

NOTE: Do NOT use the automatic MAC address assignment option featured by MERLIN controllers which may be present in your system.

5) Termination and Bias
   - Must be correct for every BACnet MS/TP device.

Start-Up Sequence

1) Disconnect all EAGLE MS/TP controllers.
2) Connect the first BACnet MS/TP controller on the RS485-1 channel of the controller. This allows the communication status of this BACnet MS/TP controller to be checked using the LEDs.
3) In the case of the Lynx controller, the blinking interval must be 2 sec off followed by 2 fast active LED blinks.
4) Check the Tx and Rx LED on the EAGLE acting as gateway for permanent flickering.
5) Connect to the EAGLE via IP.
6) Start BACshark and check if all devices will be found.
7) Add the next BACnet MS/TP controller and check its LEDs.

NOTE: As long as it is not possible to see all devices in BACshark, it does not make sense to start Quick Builder and try to discover the controllers.

Application Example 1: CORRECT installation

1) Device IDs: OK
2) MAC address on RS485–2: EAGLE has highest MAC address number: OK.
3) Max Master setting for RS485–2: EAGLE at the end of the bus has Max Masters set, so when it has the token, it will pass it on to the EAGLE with MAC address 0: OK.
4) Termination and Bias for RS485–1: Is done via 1st and last EAGLE: OK.
5) Termination and Bias for RS485–2: Is done via 1st and last EAGLE: OK.

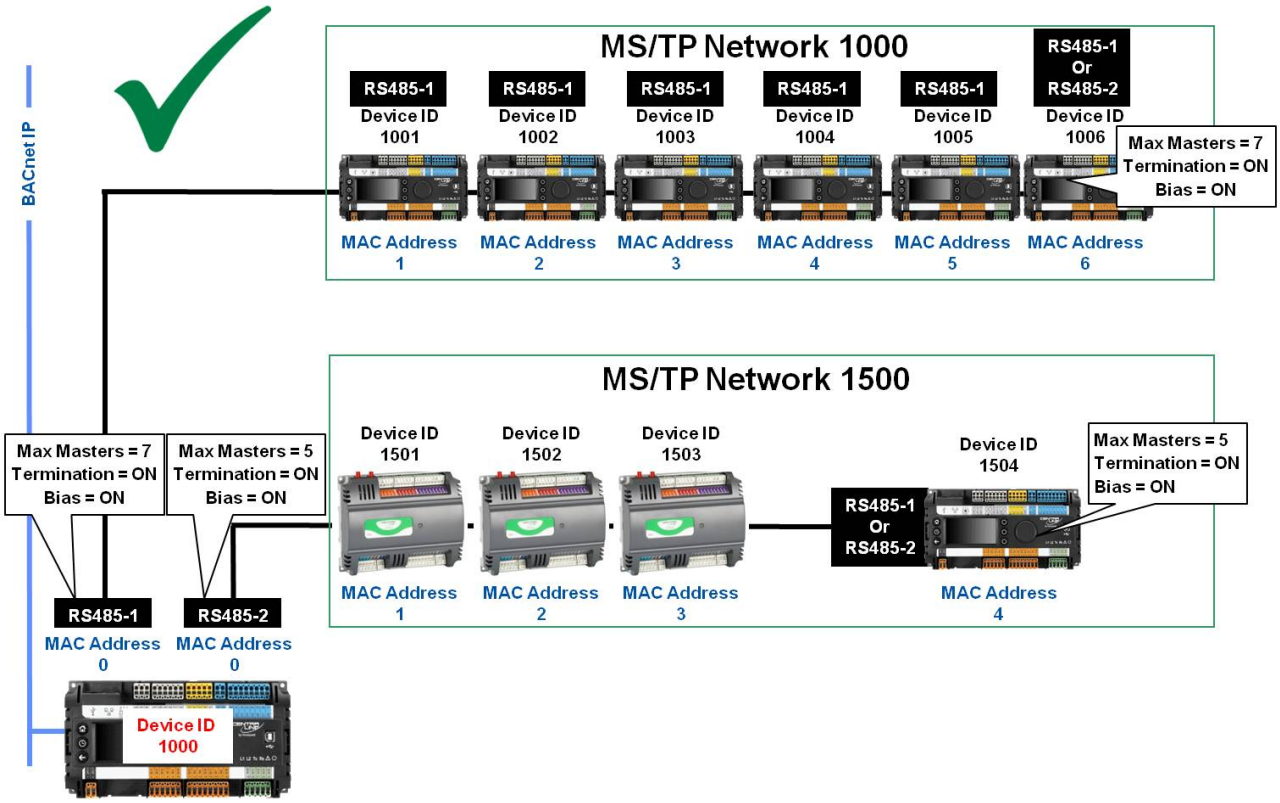


Fig. 12. Example 1: Correct BACnet MS/TP installation

Application Example 2: INCORRECT installation: Wiring of RS485-2 connected in middle of the channel

Incorrect: RS485-2 of Device ID 1000 is connected to the middle of the bus.

On RS485-2 the termination and bias are set by default and cannot be deactivated. It is therefore not allowed to place Device ID 1000 at this position. Instead, it must be placed at the beginning of the bus.

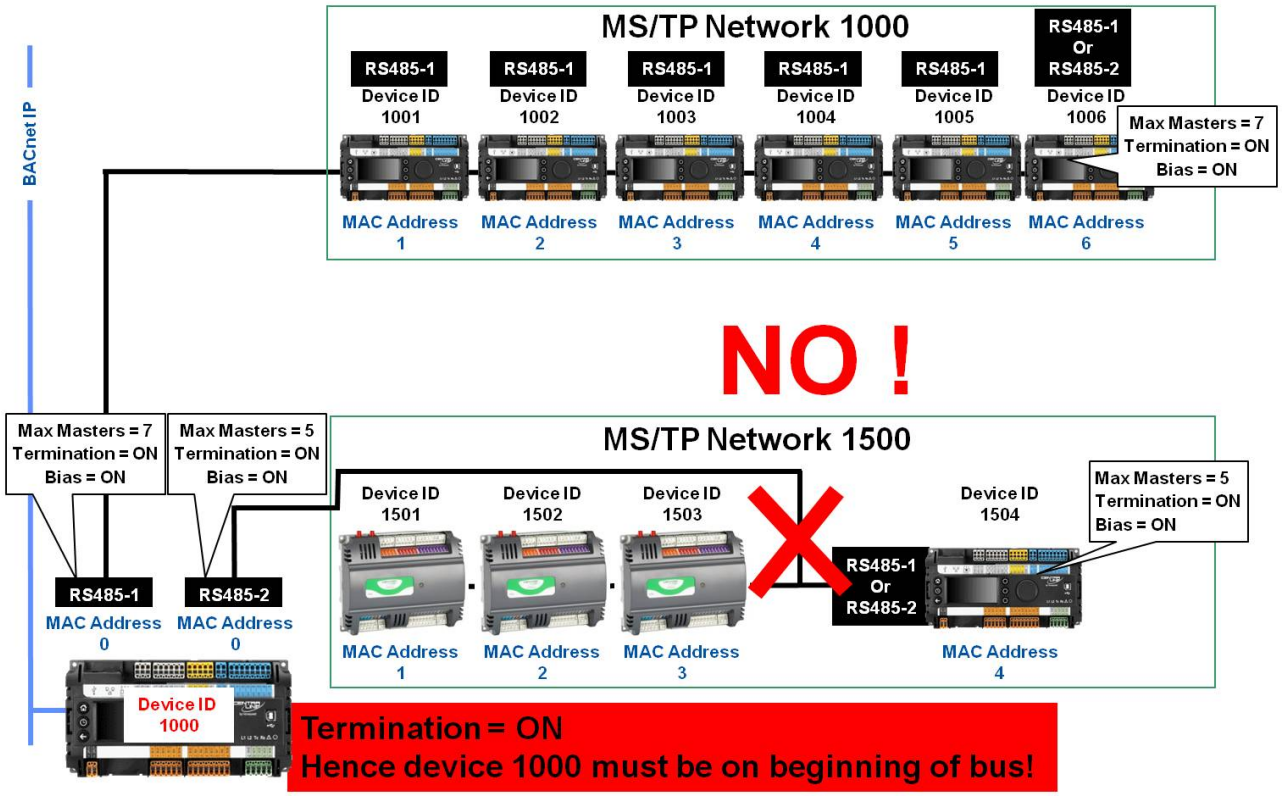


Fig. 13. Example 2: Wrong BACnet MS/TP installation

Application Example 3: Devices w/o "Max Master", Bias and termination on RS485-2

- No "Max master" setting for Device ID 1503
  → Result: Slow communication: Device 1503 will poll for max master up to 127 with each token it gets.
- No Termination for Device ID 1503
  → Action: Termination needs to be done using a 120 Ohm resistor.

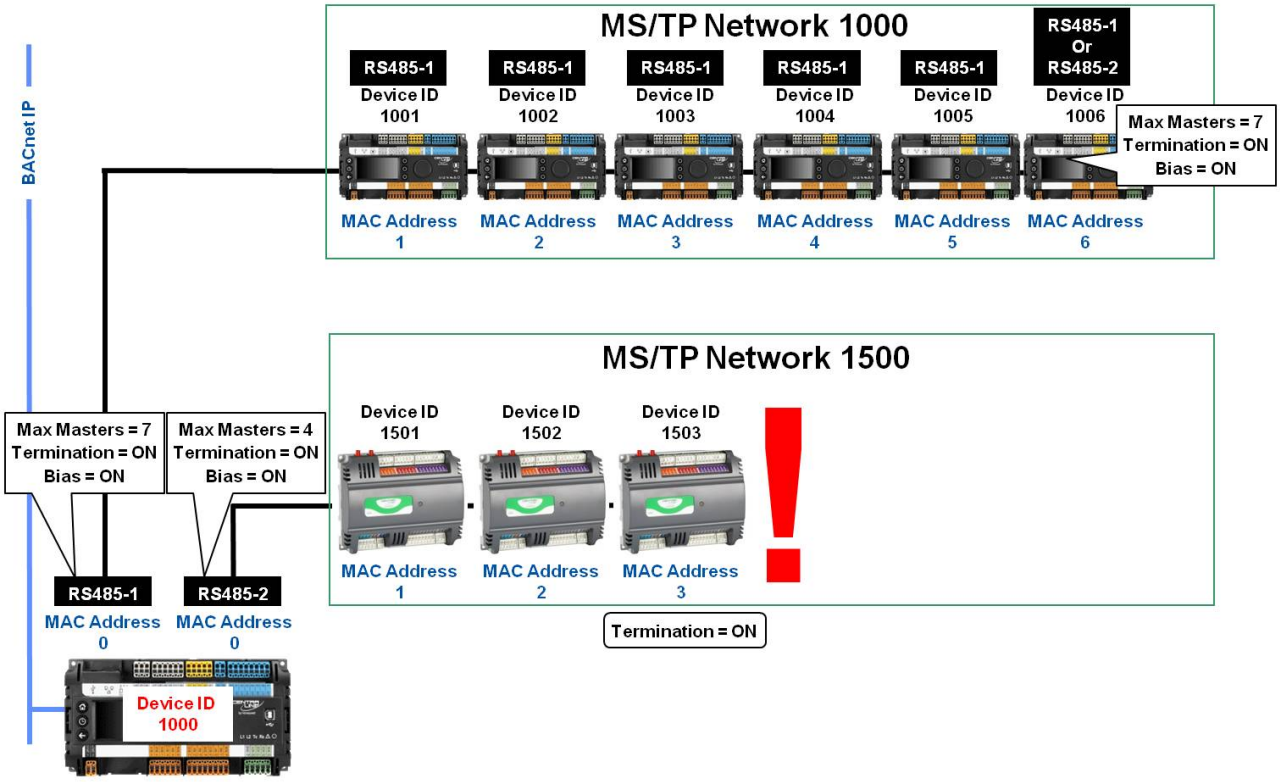


Fig. 14. Example 3: Incorrect BACnet MS/TP installation

Application Example 4: INCORRECT installation: EAGLE with RS485-2 connected in the middle of a channel

RS485-2 of Device ID 1002 is connected to the middle of the RS485-1 channel.

Because the termination and bias of the RS485-2 interface of the Device ID 1002 cannot be deactivated, it is not allowed to place it in the middle of the channel. Instead it should be connected using its RS485-1 interface.

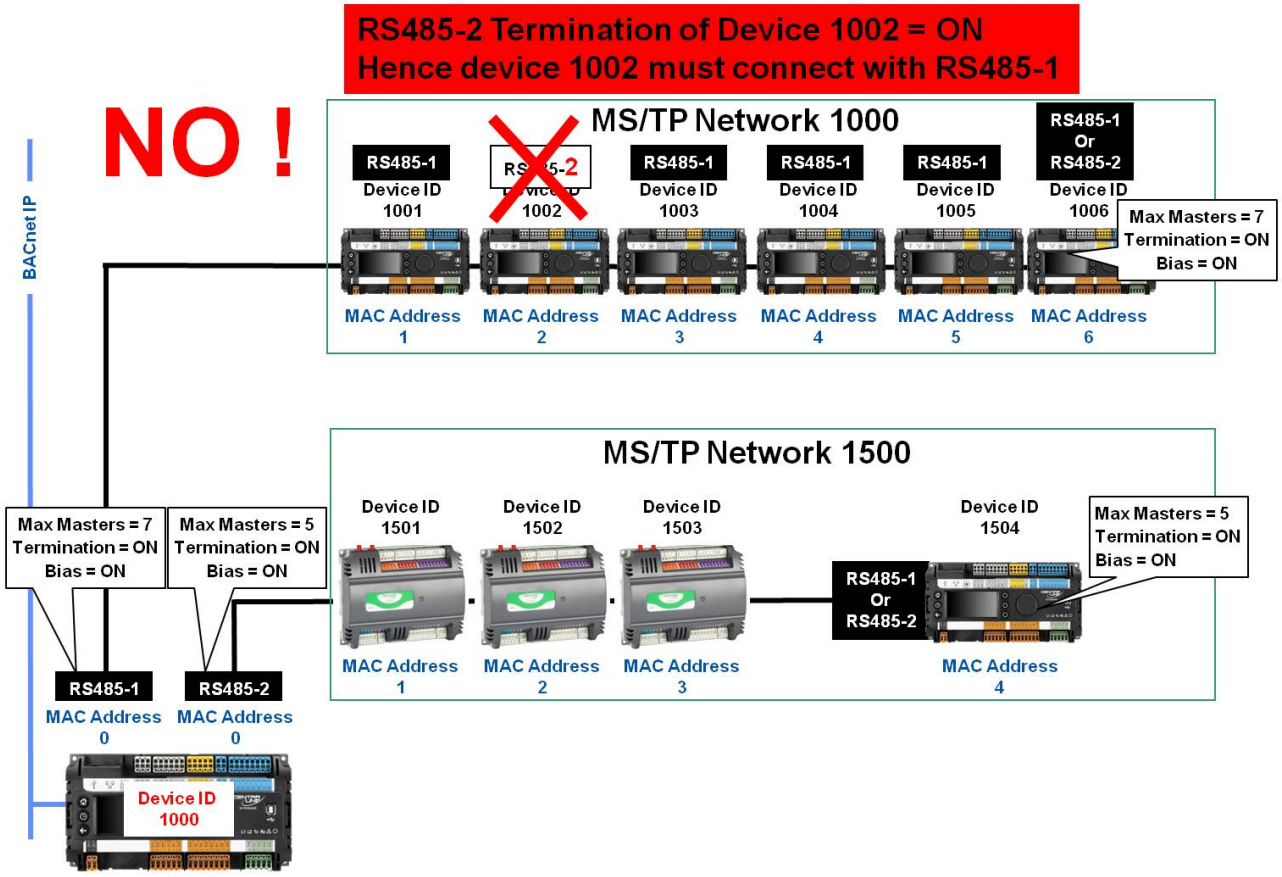


Fig. 15. Example 4: Incorrect BACnet MS/TP installation

Optimizing Communication

1. Use latest EAGLE and CARE versions

2. APDU & APDU retry setting

In order to give the gateway controller the change to have a higher communication rate per token, the following CARE settings for the Router controller must be optimized.

NOTE: The BACnet MS/TP controller should remain with the default settings.

You will find the BACnet/IP timing settings in the BACnet node in CARE:

- Change "APDU Segment Timeout" from 2000 to 5000 sec
- Change "APDU Timeout" from 3000 to 6000 sec
- Change "Number of APDU Retries" from 4 to 1

With CARE 10.03 and higher, there is a new option to disable the private transfer messages. Especially for big jobs with more than one CARE project, the synchronization of the users and calendars is not required (object synchronization max send time).

In the web interface of the EAGLE controller, the controller drop-down list has information about the alarm status from other controllers. The controllers send for this purpose the number of alarms in a Private Transfer message. This is can be disabled if this feature is not required (i.e., if the customers don’t use web interface) with the "Alarm synchronization max send time".

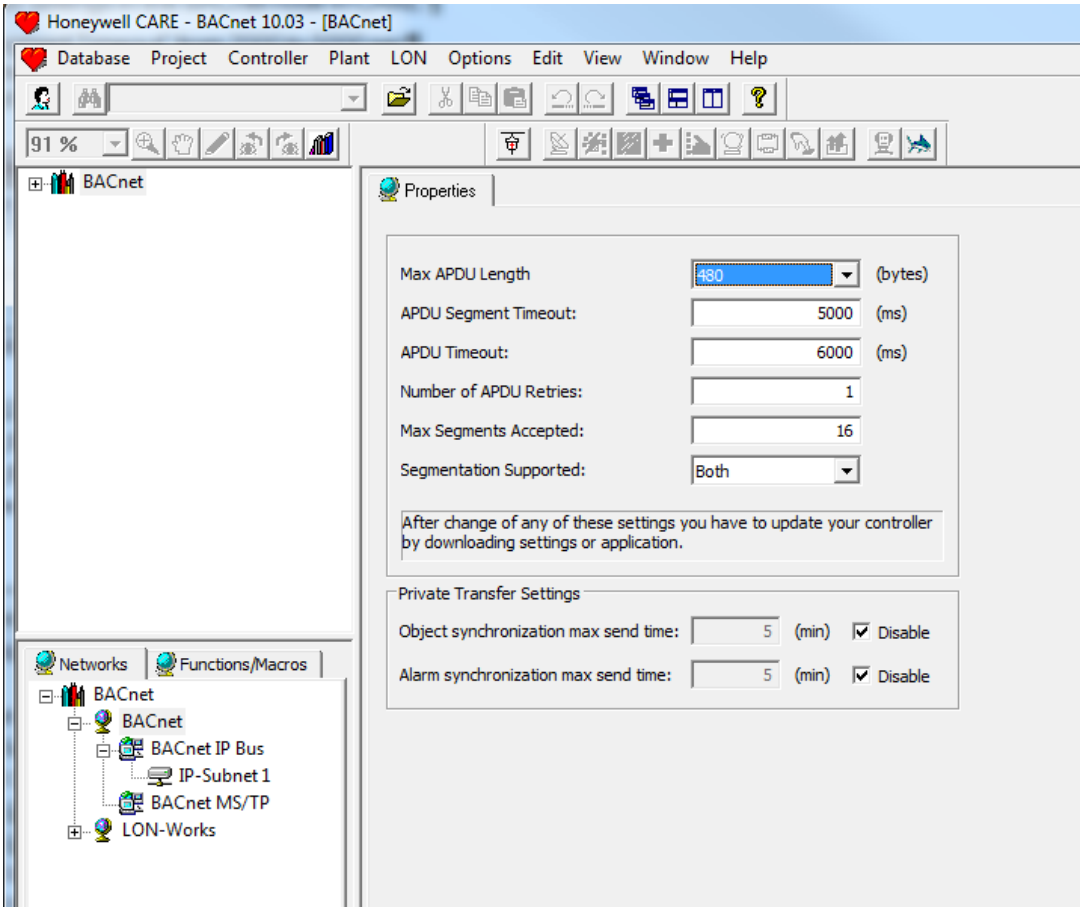


Fig. 16. BACnet/IP timing settings in CARE

You will find the BACnet MS/TP device settings in the BACnet MS/TP node in CARE:

- "Max. Info Frames Router" should be changed to 50. This will allow the BACnet MS/TP routing controller to process more messages.
- "Max. Info Frames Device" should be set to 20

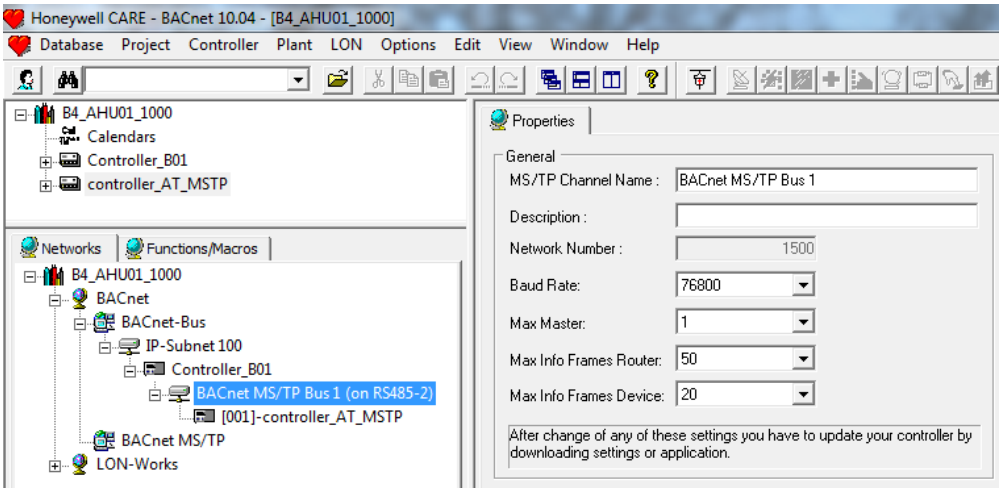


Fig. 17. BACnet MS/TP timing settings in CARE

NOTE: In order to activate these changes you have to connect to the controller and download the settings. Please stop and start the controller after download via CARE.

The result of the modification can be seen in the log file "BACnetConfig.xml"

```
<?xml version="1.0" encoding="utf-8" ?>
- <falcon_bacnet_config>
    <parameter type="max_apdu_length" value="480" />
    <parameter type="apdu_segmented_timeout" value="4000" />
    <parameter type="apdu_timeout" value="5000" />
    <parameter type="number_of_apdu_retries" value="1" />
    <parameter type="max_segments_accepted" value="16" />
    <parameter type="segmentation_supported" value="Both" />
    <parameter type="home" value="IP" sequential_number="1" />
  - <networks>
      <network type="IP" sequential_number="1" ip_portnum="47808" network_number="1" />
      <network type="MSTP" sequential_number="1" com="1" network_number="2000" max_master="15" mac_address="0" max_info_frames="50" baud_rate="76800" />
      <network type="MSTP" sequential_number="2" com="2" network_number="2500" max_master="13" mac_address="0" max_info_frames="50" baud_rate="76800" />
    </networks>
  </falcon_bacnet_config>
```

3. Delete unused BACnet MS/TP channels

If only one BACnet MS/TP channel is necessary, it is recommended to use the RS485-1 because it is optically isolated. In CARE, delete the RS485-2 if it is not used for BACnet MS/TP, Modbus RTU, or Panel Bus. This improves the controller performance because each configured interface consumes CPU load.

4. Disabling broadcast routing between BACnet/IP and BACnet MS/TP

Broadcast messages are "Who is", "I am", "Who has", "I have", and synchronizations between devices on different channels.

Disabling these broadcasts may be desired – or required - to reduce the network traffic on the BACnet MS/TP network for the purpose of optimizing the pure BACnet MS/TP communication speed.

Beginning with firmware 3.04.03, broadcasts between devices on BACnet/IP on one hand and devices on BACnet MS/TP on the other can be disabled in the SVGA web browser interface.

Optionally, this disabling can begin after an adjustable time ("count-down").

See screenshots below.

Important notes about implications (side-effects):

1) This setting is lost after application download! (The goal is to have this setting included in a future CARE version.)
2) As long as broadcast routing is disabled, BTL conformance is invalid.
3) BACnet front-ends with dynamic object bindings like EBI will no longer be able to identify BACnet MS/TP devices and their objects. This will become a problem after network disruptions and after controller restarts. (This does not apply to Niagara, since with Niagara, the network and MAC address is stored.)
4) The following synchronizations will no longer take place:
   - Calendar synchronization between the BACnet/IP channel and the BACnet MS/TP channel.
   - User synchronization between the BACnet/IP channel and the BACnet MS/TP channel.
   - Display of all bus-wide alarms of all controllers in the BACnet MS/TP channel in the web browser interface.

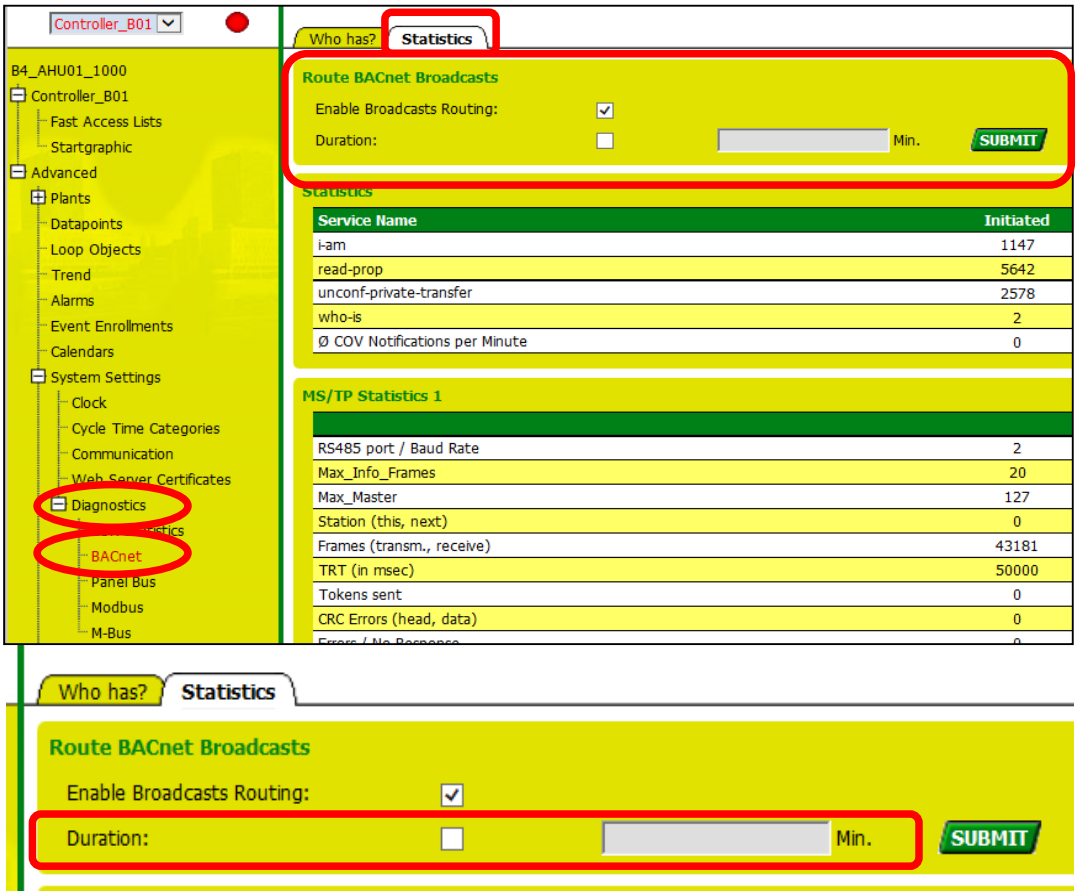


Fig. 18. BACnet MS/TP Statistics in the SVGA web pages

Checking and Verifying

BACnet Statistics in the Web Pages

In the Diagnostics / BACnet Statistics Tab, it is possible to check the BACnet MS/TP settings as well as to identify the last device that has been communicating with the EAGLE.

In order to check single devices, disconnect all BACnet MS/TP devices except the one for which the diagnosis needs to be done. The result in the web page should be that the MAC address of this BACnet MS/TP device is listed under "Station (this, next)"

Note the pink highlighted numbers in MS/TP Statistics 1 in the example screenshot below:

This station (BACnet MS/TP controller) = MAC address 0, Next station (BACnet MS/TP controller) = MAC address 14.

This means that controller 0 will scan the BACnet MS/TP bus every time it gets the token, which means a slow-down of BACnet MS/TP communication!

In order to remedy this, the controller with MAC address 14 should be given MAC address 2, and the remaining devices (stations) should be checked for sequential MAC addresses or sequential wiring.

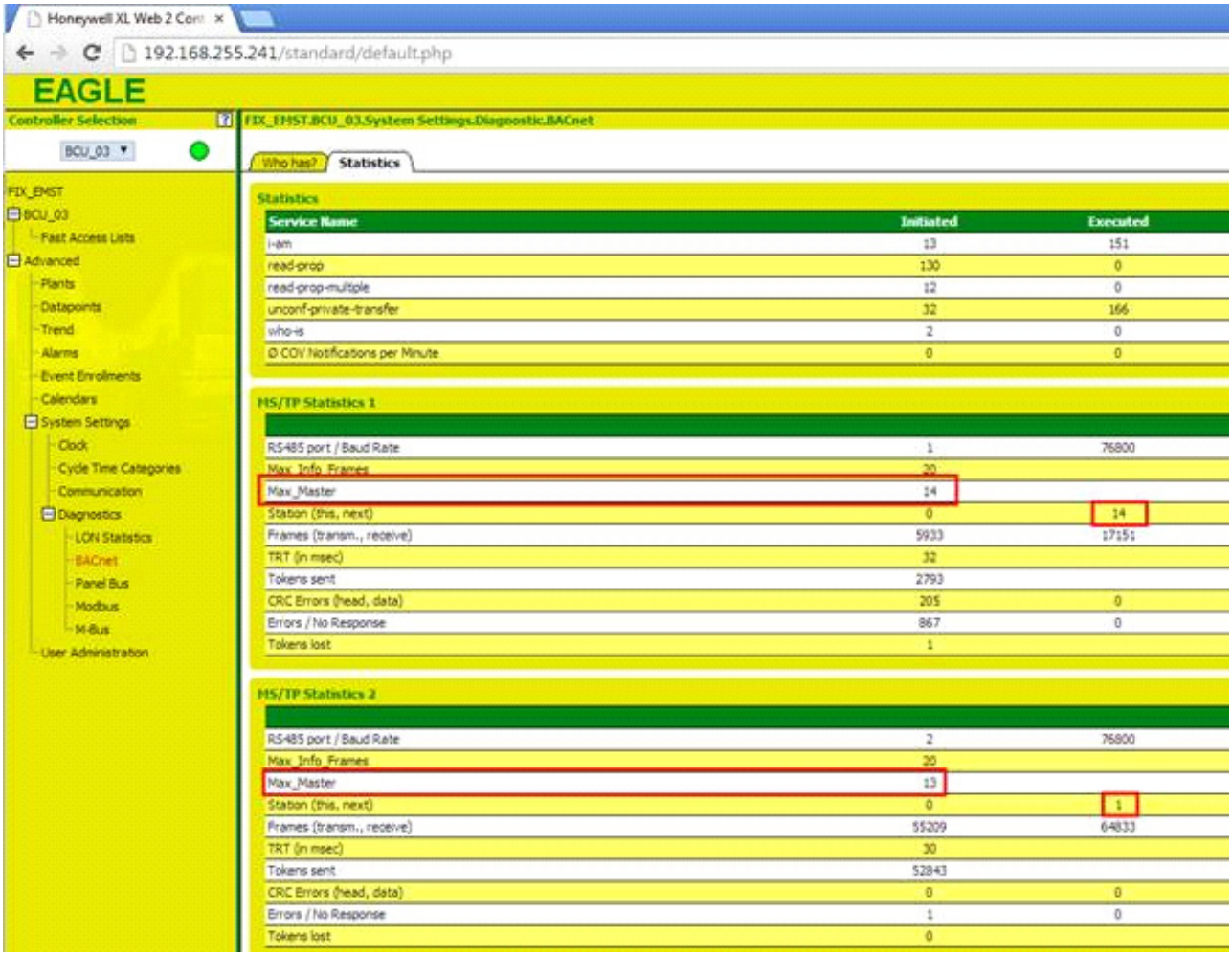


Fig. 19. BACnet MS/TP Statistics in the SVGA web pages

Debugging (using controller firmware 3.04.03 and CARE software 10.06 and higher)

In order to debug a BACnet MS/TP network, the following steps are recommended:

1. Upload BACnet statistic files from the controller

In order to check the BACnet communication, the log-files of the controller can be uploaded using CARE. Connect to the controller via USB or Ethernet in the setup tab and click on "Log files – Configure" and upload the logfiles.

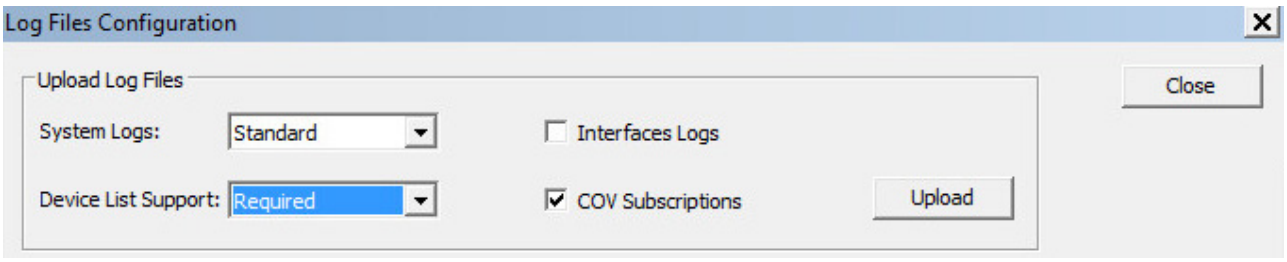


Fig. 20. Upload controller logs (controller firmware 3.04.03 and higher)

After the logs are uploaded, you will get a notification including the path:

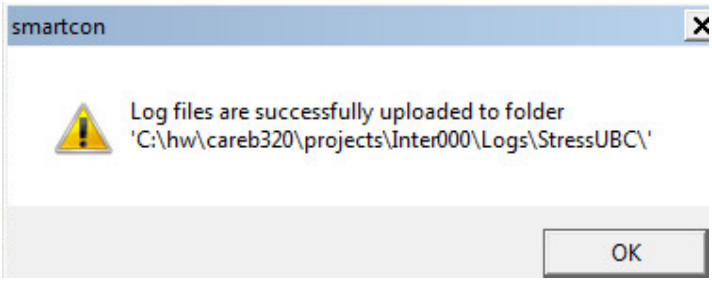


Fig. 21. Confirmation for uploaded log files

2. Record BACnet MS/TP and BACnet/IP packets on the controller

In order to check the BACnet MS/TP and the BACnet/IP communication, the capture of the traffic can initiated with CARE.

For BACnet/IP, please start the Ethernet log. Please check if the correct BACnet port is entered in the filter column. For BACnet MS/TP, start the capture on the correct RS485 interface of the controller.

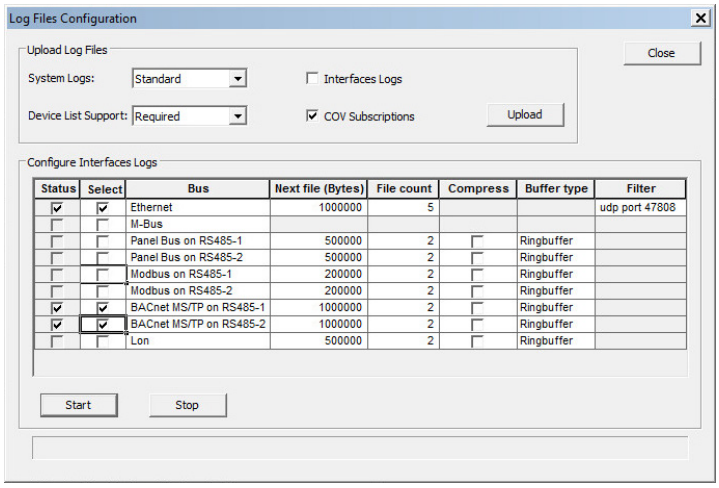


Fig. 22. Enable BACnet logging

After the error occurs in the controller, it is possible to upload the logs. For this select the "Interfaces Logs" and click the upload button

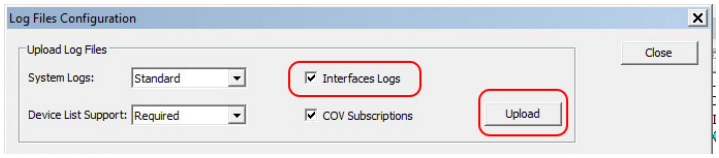


Fig. 23. Upload BACnet logs

NOTE: After finishing the diagnostics, the logging must be stopped with CARE in order to not deteriorate the controller performance.

NOTE: For BACnet MS/TP related issues, it is better to record the BACnet MS/TP traffic using the "Nientec" serial sniffer: How To: Record MS/TP packets using wireshark.

MODBUS RTU

References

The Modbus protocol is described in the following two documents, which can be found at www.modbus.org:

- Modbus_over_serial_line_V1_02.pdf
- Modbus_Application_Protocol_V1_1b.pdf

Modbus RTU slaves can be connected to RS485-1 (consisting of push-in terminals 24 [GND-1], 25, and 26) and/or RS485-2 (consisting of push-in terminals 29, 30, and 31 [GND-2]).

Modbus RTU on RS485-1

Powering, Bias and Termination

With regards to Fig. 24, please note the following:

NOTE: Always power each EAGLE and the connected Modbus RTU slave via separate transformers.

NOTE: For the length of the RS485 bus (marked "L" in the following figures), see section "The RS485 Standard" on pg. 3.

NOTE: If any of the devices are electrically isolated, it is recommended that those devices be connected to signal ground. See section "The RS485 Standard" on pg. 3.

Example 1: Single EAGLE Controller and Connected Modbus RTU slaves (with inserted termination resistor)

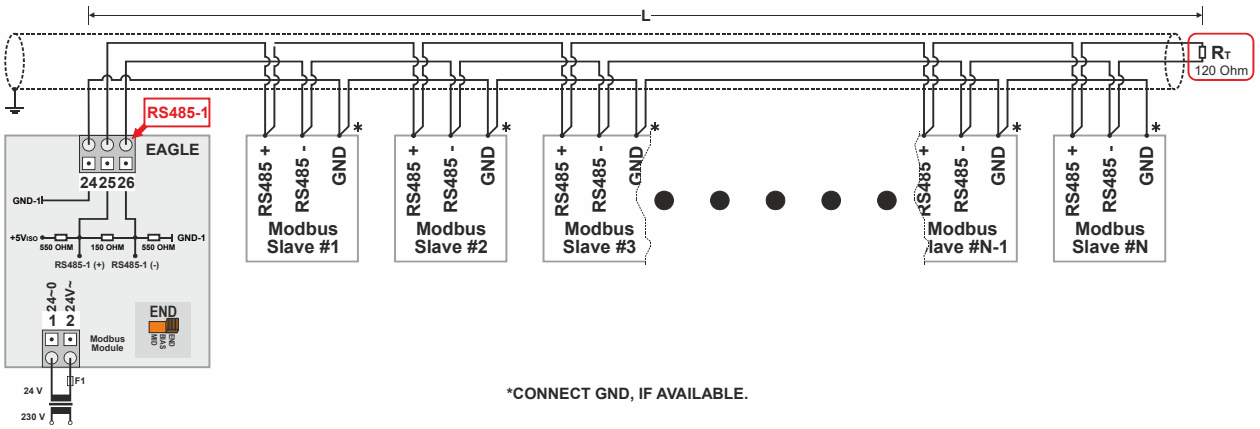


Fig. 24. Connection of RS485-1 to Modbus

The termination resistor must be inserted directly into the terminals of the last Modbus RTU slave.

Modbus RTU on RS485-2

NOTE: GND-2 is internally connected with 24V-0 (terminal 1) and system GND (terminals 19+37)

Powering, Bias, and Termination

With regards to Fig. 25, please note the following:

NOTE: Always power each EAGLE and the connected Modbus RTU slaves via separate transformers.

NOTE: For the length of the RS485 bus (marked "L" in the following figures), see section "The RS485 Standard" on pg. 3.

NOTE: If any of the devices are electrically isolated, it is recommended that those devices be connected to signal ground. See section "The RS485 Standard" on pg. 3.

NOTE: Between devices equipped with non-isolated RS485 bus interfaces, potential differences of max. ±7 V are allowed. Further, this bus should not extend beyond a single building.

Example 1: Single EAGLE Controller and Connected Modbus RTU slaves (with inserted termination resistor)

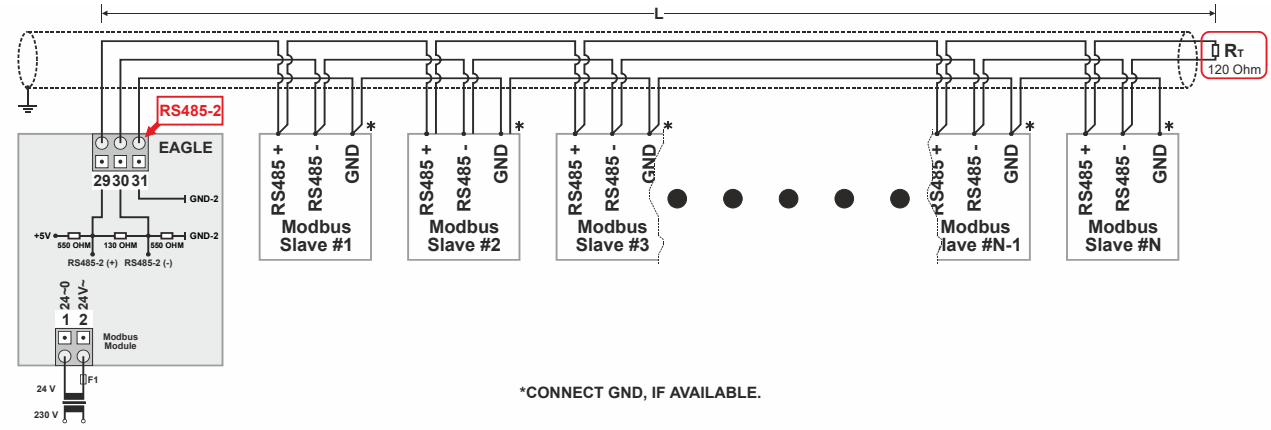


Fig. 25. Connection of RS485-2 to Modbus

The termination resistor must be inserted directly into the terminals of the last Modbus RTU slave. BIAS is not required because this is automatically done on the first EAGLE.

Modbus Addressing and Wiring Rules

1) Use ONE Modbus RTU master, only - e.g., the EAGLE controller.
2) Maximum number of devices: 32, including the Modbus RTU master.
3) Address: Can be from 1 to 247.

NOTE: Never use address 0, because 0 is used by the Modbus RTU master exclusively for broadcast addressing!

   - Per device, the address must be unique on the same Modbus.
   - Dip switch setting (when available) must be correct (depending on its physical connection) for every BACnet MS/TP device.

4) Termination and Bias: Must be correct for every Modbus RTU device.
5) Naming of Modbus terminals on Modbus RTU slave. Modbus RTU devices may use different naming and descriptions for the wire connections. Possible synonyms can be as follows:
   - Transmit Data: TxD, Tx, T+/R+, D1, B
   - Receive Data: RxD, Rx, R-/T-, D0, A
   - Signal Ground: GND, COMMON

Start-Up Sequence

For a first test of the communication, connect only one Modbus RTU device to the EAGLE, and use short cables (i.e., cables less than 3 meters in length).

Connect the first Modbus RTU controller on the RS485-1 channel of the controller. This allows checking the communication LEDs on this Modbus RTU controller for the communication status.

Create one register to read; this register you create must have a known value.

Check the Tx and Rx LEDs on the EAGLE for permanent flickering.

Once you have verified that the TX and RX LEDs are permanently flickering (indicating that data communication is taking place), you can now extend the interface to all required registers. Use XW-Online or the EAGLE's onboard HMI or the web pages to verify that the Modbus data is, indeed, being received as expected, and that this data is being successfully written.

Once this is working properly, add the next Modbus RTU slave.

Best Practices

The following order of investigation and checks is recommended:

1. Get Modbus RTU slave device documentation

It is mandatory to have the vendor’s documentation of the Modbus functionality for every Modbus RTU slave that is to be connected to the controller.

This will allow obtaining information about most of the required information, but most likely not all of it.

Information that is missing can be obtained from the support function of the vendor.

2. Verify use and activation of the correct Modbus RTU interface

Make sure that the desired RS485 interface is used and active:

- For the EAGLE, check in CARE:

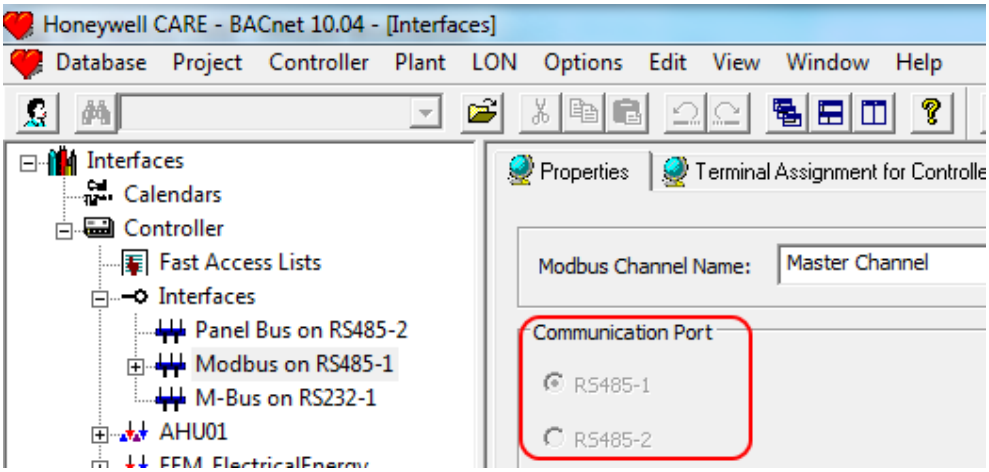


Fig. 26. Modbus RTU communication port indication in CARE

For Modbus RTU slaves: Modbus RTU slaves may not have the Modbus interface activated by default. Activation may be necessary via software, e.g., setting a parameter via the Modbus device's user interface. In addition, activation may be necessary via hardware, e.g., shorting an input of the Modbus RTU slave.

When using a MVC or W858xxx controller as a Modbus RTU slave, verify that you have the correct hardware variant which supports Modbus on the RS485 interface as expected. Consult the MVC documentation for this purpose.

3. Verify RTU Transmission Mode

All Modbus RTU slaves must use transmission mode RTU.

No other transmission mode (e.g., ASCII) is supported.

4. Ensure only one Modbus RTU master is on the bus

On any given Modbus, there can be only one Modbus RTU master controller – and up to 63 Modbus RTU slaves.

The EAGLE controller does not support "Multi-Master" nor "Modbus plus" devices nor does it support such configurations on the Modbus.

5. Check Modbus Channel Communication Parameters

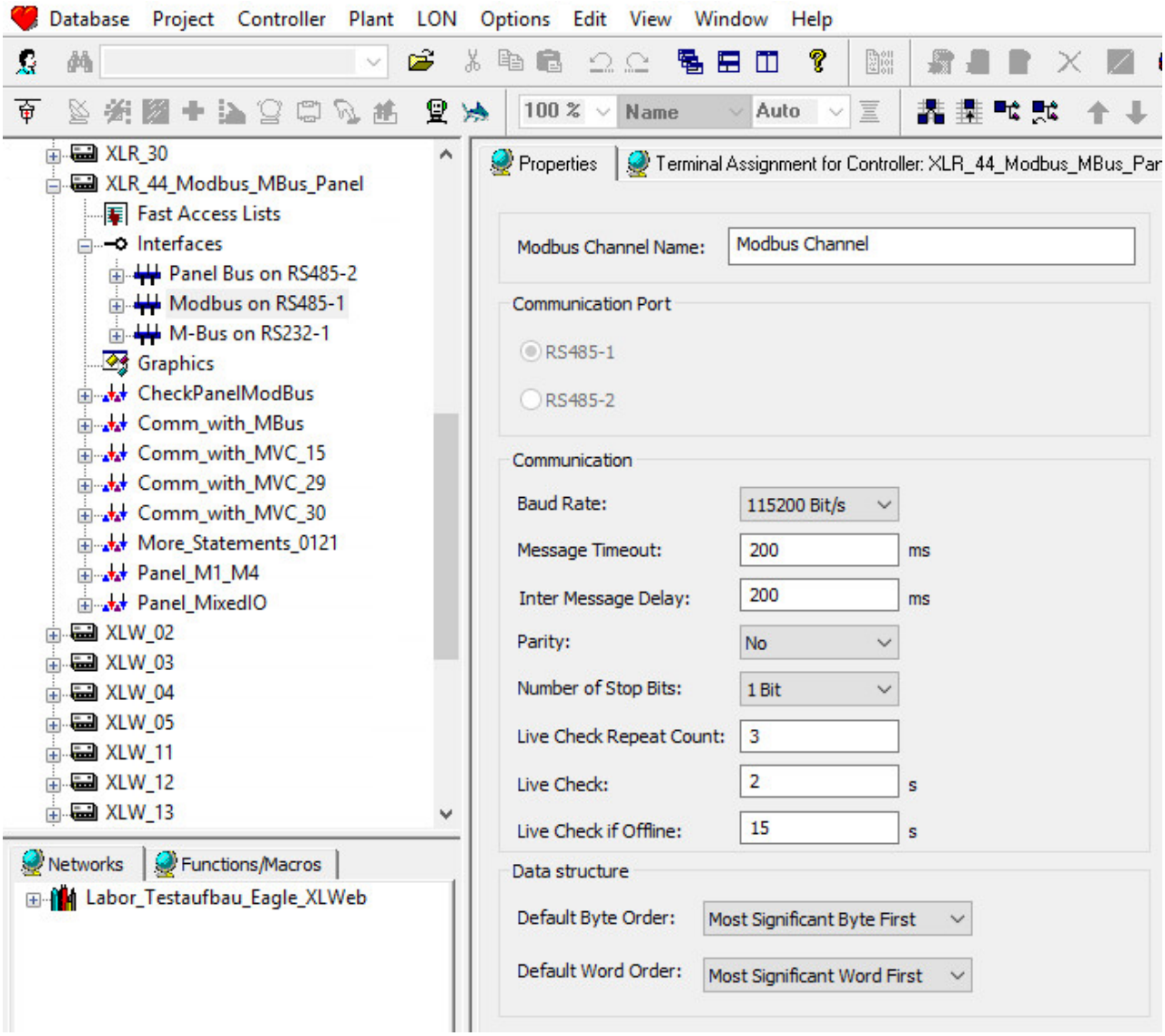


Fig. 27. Modbus RTU Channel communicating settings in CARE

No device ID for the Modbus Master = EAGLE!

As per the Modbus standard, the Modbus RTU master does not have an ID.

Baud Rate:

All Modbus devices must have an identical Baud rate, an identical Parity, and an identical number of stop bits.

Typically, the Baud rates of Modbus devices are set and changed via their user interfaces or via their separate configuration software.

Message Timeout:

If all other attempts to get Modbus communication running have failed, vary the Timeout of the Modbus messages – though only in the last place. Use a Modbus message analyzer ("sniffer") in order to verify that a Modbus RTU slave does not react to the Modbus commands of the MVC, or which Modbus error codes are returned. In this case, the timeout may be increased in order to give the Slave device more time to respond.

Inter Message Delay:

In Inter Message Delay (0 …1000 ms), leave the default setting or change it if you want to remedy response issues with the Modbus slaves. The inter message delay is the time in ms which the master waits before it sends the next request. The set value affects bus performance and stability with reciprocal effect. Entering values between 0 ms and 200 ms is suitable for experimenting with performance and stability. When entering 0 ms, you will have the best performance but irregularities in the master-slave communication may occur which result in worse stability. When entering values greater than 200 ms, you will have a high stability but the performance might slow down to an inappropriate level. The default value of 200 ms can be a good value considering both performance and stability.

Parity:

Modbus devices may have a configurable Parity, but in all likelihood, they will not have a configurable number of stop bits.

If a Modbus device has an un-matching number of stop bits, this device cannot be used!

NOTE: Not all Modbus products comply with the Modbus standard. Modbus devices that have implemented "no parity" may have 1 or 2 stop bits.

Live Check:

The Live Check settings should be adapted if the time behavior of the connected Modbus RTU slaves is known and suggests to do so. See section "Debugging" on pg. 38 for more information.

6. Check Modbus Channel Data Structure

The Modbus standard defines the serial bit order to be Least Significant Bit First and Most Significant Bit Last.

However, the Modbus standard does not define the Byte and Word order of messages. Hence, there are Modbus devices that will transmit Most Significant Byte First or Lower Significant Byte First.

For 32-bit values, Most Significant Word First or Lower Significant Word first may be implemented.

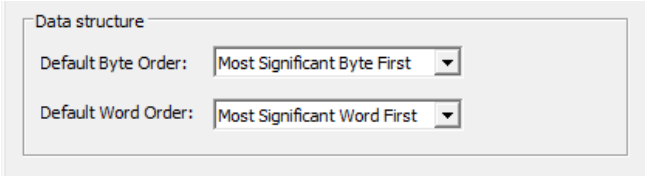


Fig. 28. Modbus RTU Data Structure settings in CARE

The default settings in CARE are shown above, and match the most common implementation among Modbus devices. These settings can be changed for the Modbus Master Channel as a whole, and for every Modbus device individually.

7. Check Modbus RTU Slave communication settings

Device ID:

Modbus RTU slaves can have IDs between 1 and 247.

All Modbus RTU slaves must have a different ID.

Typically, the Modbus IDs of 3rd-party Modbus RTU slaves are set and changed via their user interfaces or via their separate configuration software.

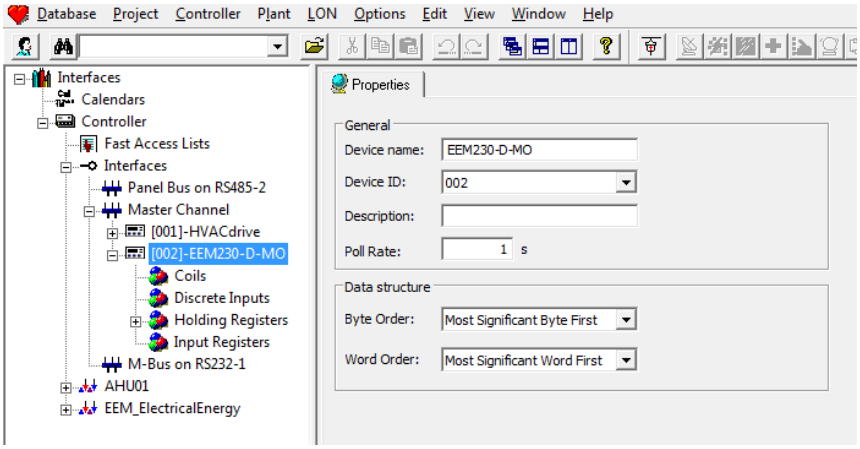


Fig. 29. Modbus RTU slave communicating settings in CARE

Poll Rate:

This parameter allows adapting to the communication and processing performance of the Modbus RTU slave.

If the Modbus RTU slave does not respond fast enough, increase the poll rate.

NOTE: Beginning with controller firmware version 4.02.00, poll rates of "0" can now also be entered. A value of "0" corresponds to a poll rate of 0.1 seconds.

Data Structure:

The default settings in CARE are shown above, and match the most common implementation among Modbus RTU devices. These settings are derived from the Modbus Master Channel settings, and can be set for every Modbus device individually.

Heartbeat:

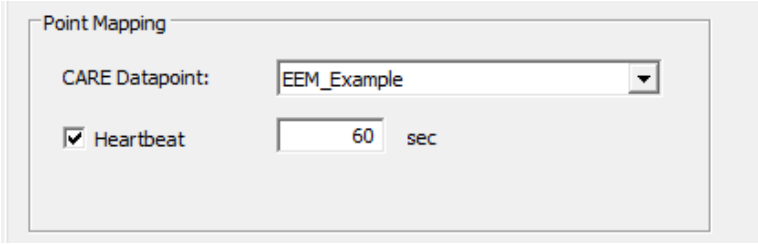


Fig. 30. Modbus heart-beat settings in CARE

Heartbeat means a repeated sending of the output value of the mapped datapoint.

It can be set individually for every Modbus Coil and holding register.

Too many and too frequent heartbeats can overload the EAGLE controller. This overload can be caused by one, several, or all of the following reasons:

- Too many heartbeats are active;
- One or several heartbeats are too short;
- The Modbus communication load is too high;
- The application is too big, causing an excessively long cycle time.

In order to correct this problem, you must find out and decide which of the above reasons is valid, and which of them can be eliminated.

8. Finding the correct Modbus register type

See also http://xl5kfaq.ge51.honeywell.de/index.php?action=artikel&cat=271&id=5826&artlang=en

If the specification of a Modbus RTU slave does not explicitly state the type of register that is used for a Modbus parameter, then apply the following approach:

- Most commonly, Modbus RTU devices use holding registers for "Read Only" as well as for "Read and Write" parameters. However, a Modbus device may use input registers for "Read Only" parameters, and holding registers for "Read and Write" parameters.
- If you cannot find out the register type, try holding registers (function Code 03) first, as these are used most commonly.
- In CARE, implement one holding register only. Download the application and check the communication. When the Modbus communication is running fine, add more registers and continue testing.
- Even if the Modbus parameter supports "Read and Write", try out message type "Read Only" first. When the Modbus communication is running fine, you can switch the message type from "Read Only" to "Read and Write".
- If holding registers do not work for "Read Only" Modbus parameters, try input registers.
- Make sure that you do not write to "Read Only" Registers, because this can kill Modbus communication.

Testing Modbus Slave Registers with a 3rd-Party Tool

The following hardware and software tools allow testing communication of Modbus RTU slaves:

Hardware: USB to RS485 converter:

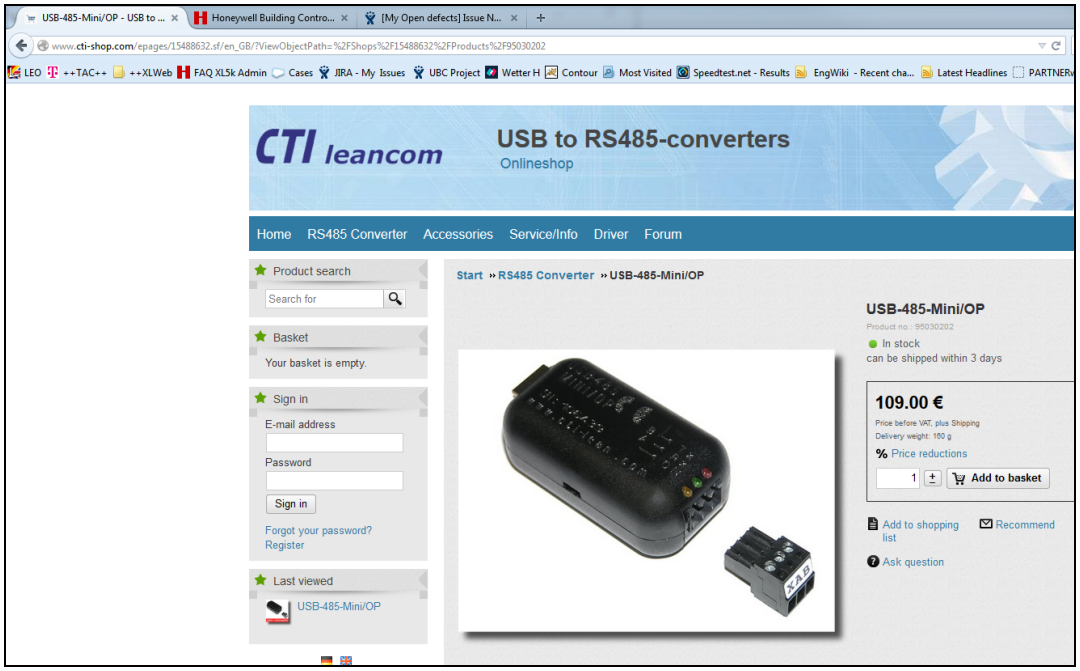


Fig. 31. Screenshot of CTI leancom USB to RS485 converter (http://www.cti-shop.com/epages/15488632.sf/en_GB/?ViewObjectPath=%2FShops%2F15488632%2FProducts%2F95030202)

Software: "QModMaster", download from Sourceforge:

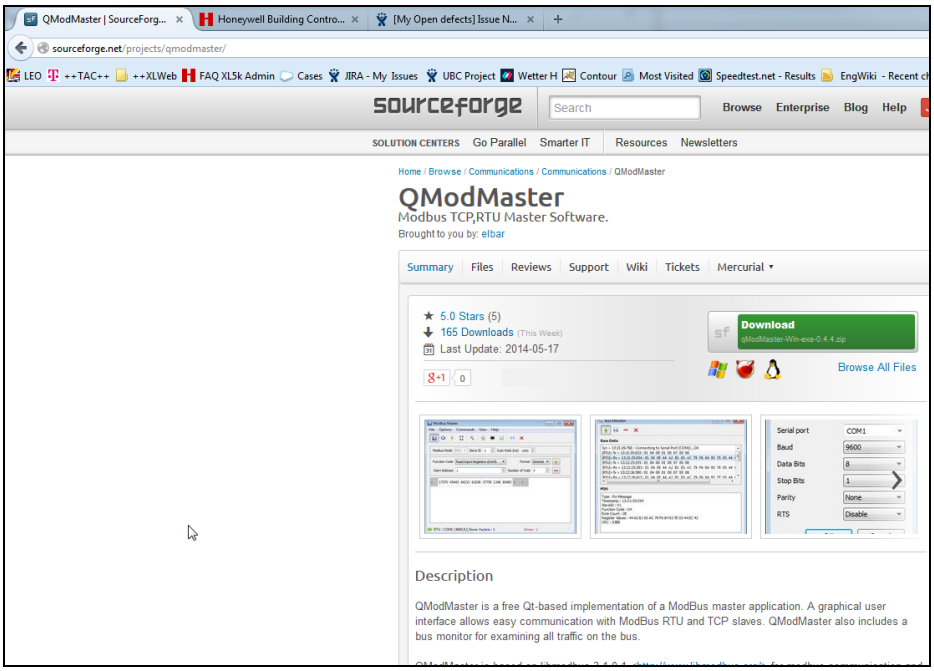


Fig. 32. Screenshot of QModMaster software (http://www.cti-shop.com/epages/15488632.sf/en_GB/?ViewObjectPath=%2FShops%2F15488632%2FProducts%2F95030202)

The step-by-step procedure is as follows:

1) Configure baud rate – same as setup in CARE:

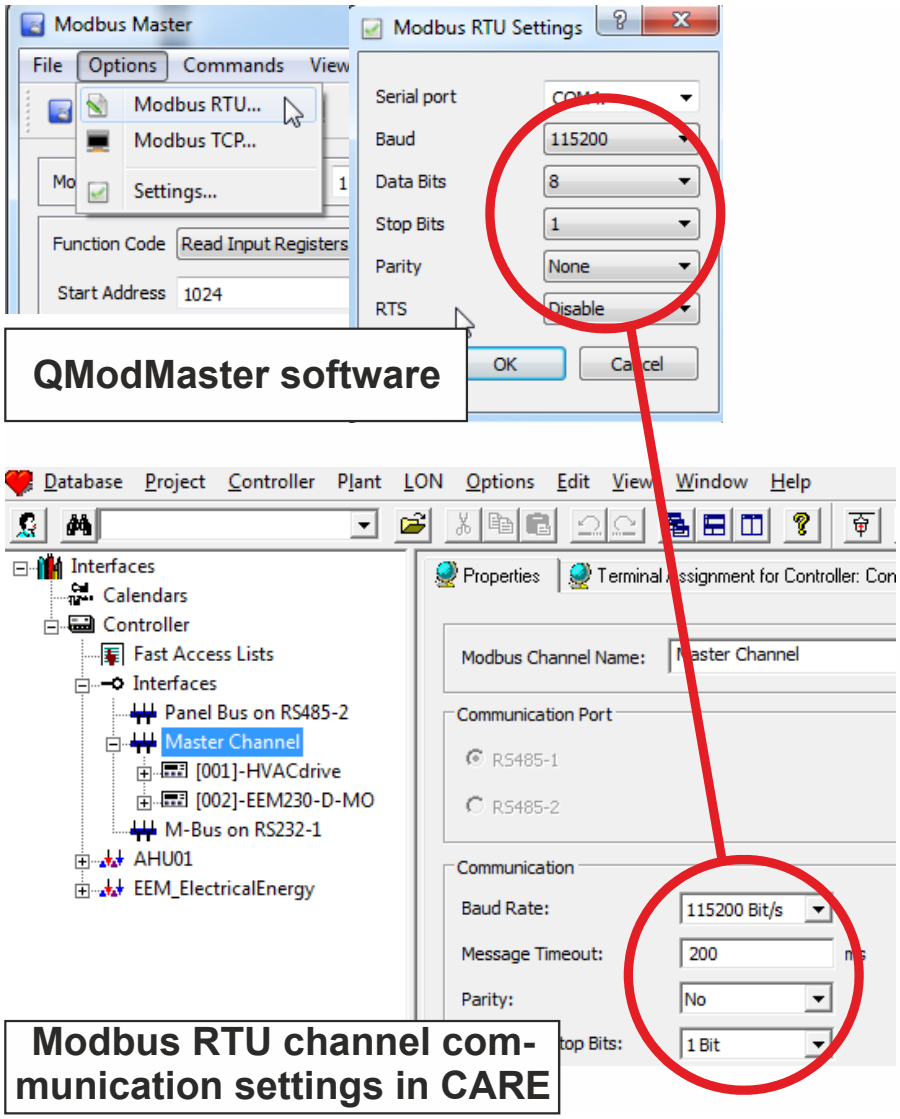


Fig. 33. Screenshots of QModMaster software and CARE with Modbus RTU channel communication settings

2) Enter physical address of a selected Modbus register that you want to address and test:

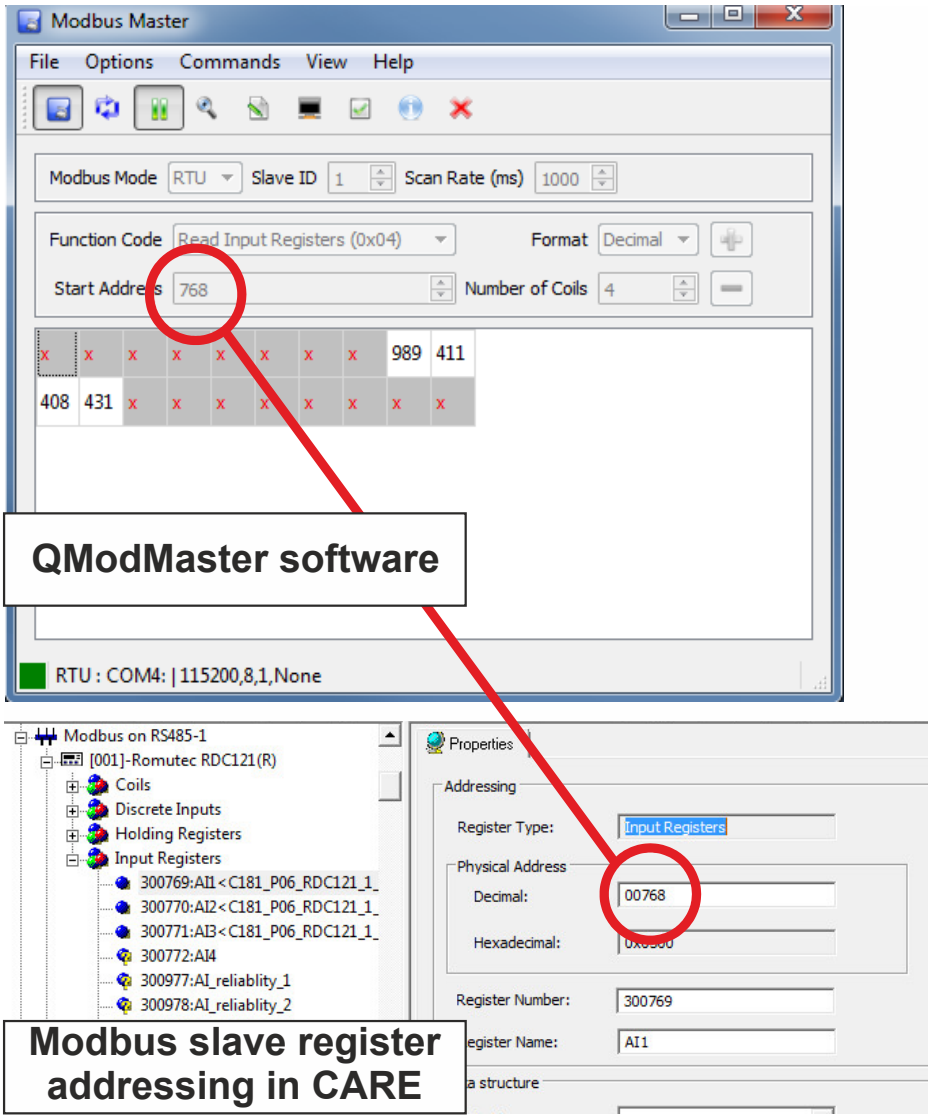


Fig. 34. Screenshots of QModMaster software and CARE with Modbus RTU slave register addressing

9. Modbus register address numbering:

The physical addressing on Modbus starts with the first address being "0".

However, the Modbus devices on the market use two different address numberings in their documentation:

- Address numbering starts with "0"; this represents the physical address numbering in decimal:

Example: Carel Power+ inverter

ModBus® address	Parameter	Def	Min	Max	U.M.	R/W	
0	Motor control mode 0 = PM brushless motor 1 = asynchronous motor with vectorial control 2 = asynchronous motor with V/f control	0	0	2	-	R/W	S
1	Motor base frequency	500 (50.0Hz)	0	5000 (500.0Hz)	0.1Hz	R/W	S a n

Fig. 35. Screenshot of a Carel Power+ Inverter Nodbus address list

In this case, either enter 00000 into the "Decimal" field of the "Physical Address", or enter 400001 into the Register Number" field:

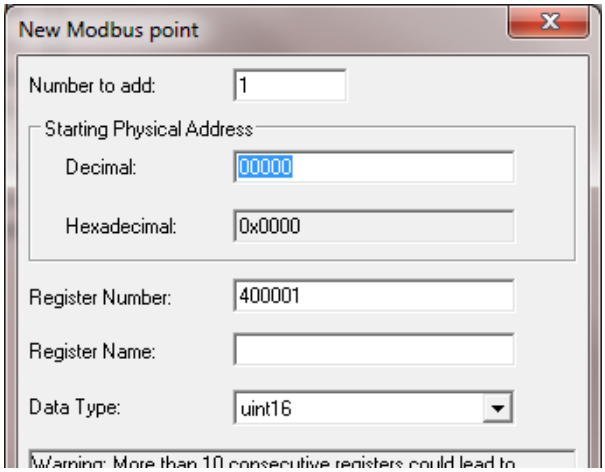


Fig. 36. CARE screenshot of a Modbus RTU slave register addressing

- Address numbering starts with "1"; this represents the physical (decimal) address numbering PLUS 1:

Example: Honeywell NXL inverter

Addr.	Modbus register	Name	Scale
2101	32101, 42101	FB Status Word	-
2102	32102, 42102	FB General Status Word	-
2103	32103, 42103	FB Actual Speed	0,01
2104	32104, 42104	Motor speed	0,01
2105	32105, 42105	Motor speed	1
2106	32106, 42106	Motor current	0,1
2107	32107, 42107	Motor Torque	0,1

Fig. 37. Screenshot of a Honeywell NXL Inverter Modbus address list

In this case, either enter 2100 into the "Decimal" field of the "Physical Address" or enter 402101 into the "Register Number" field:

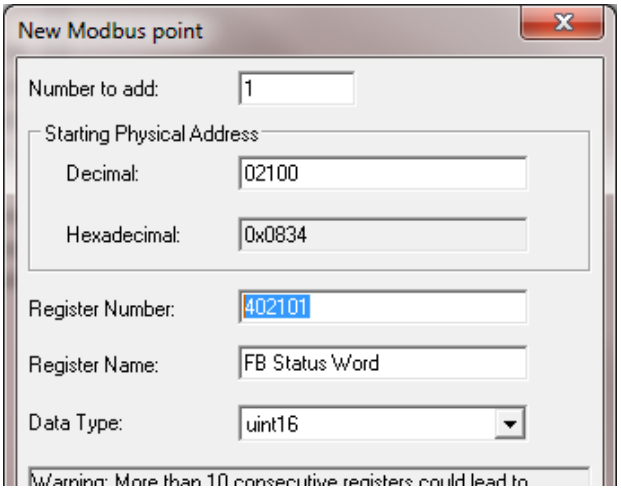


Fig. 38. CARE screenshot of a Modbus RTU slave register addressing

NOTE: The CARE printout uses physical addresses in decimal format.

10. Writing single bits of 16-bit holding registers

A Modbus device may allow writing single bits of a 16-bit holding register.

Example:

The following is an example of an inverter which allows start/stop and setting the sense of rotation with two bits of one holding register:

- Bit 0 (First bit): 0=stop, 1=run
- Bit 1 (Second bit): 0=clockwise, 1=anticlockwise

Because all 16 bits of a holding register have to be transmitted, the EAGLE cannot send a single bit. For this reason, the workaround is to map an analog datapoint to the Modbus RTU register and to write the decimal value of the holding register.

For this example of the inverter, this will look like this:

- stop/clockwise = decimal value 0
- run/clockwise = decimal value 1
- stop/counterclockwise = decimal value 2
- run/ counterclockwise = decimal value 3

11. Setting decimal points for integer values received from a Modbus RTU device

Modbus parameters with decimal values are typically transmitted as 16-bit registers with INT16 or UINT16 Data Type, and a factor is used to define the decimal point.

Example

- °C Temperature range received via Modbus: -200 … 2000
- °C Temperature range after conversion table in CARE: -20.0 … 200.0

In the above example, a temperature is be defined as a 16-bit input register, with a factor of 0.1.

Solution 1 (all CARE versions): Use a conversion table in CARE which divides the original value by 10.

Solution 2 (CARE 10.06.01 and higher): Enter a corresponding value in the new property of the register named "scaling factor."

12. 16-bit and 32-bit registers

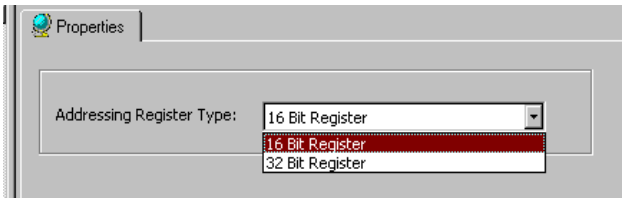


Fig. 39. CARE screenshot of Addressing Register Type setting

A Modbus RTU device may use 16-bit or 32-bit registers for its input registers or/and its holding registers.

The selection is done in CARE and it is valid for all input registers or/and holding registers of a specific Modbus device with which the EAGLE communicates. (The EAGLE can simultaneously communicate with several Modbus devices that may have 16-bit or/and 32-bit registers.)

13. Accessing 32-bit register values from two 16-bit registers (register type: 16-bit register)

For values greater than 65536 (equaling 16 bit), Modbus devices typically use two 16-bit registers. The addressing may use one of the two following implementations. You can identify this implementation by the fact that every second Modbus address for these registers will be missing: For example, the addresses might be in the following sequence: 100, 102, 104, 106, etc.

In CARE, you can choose which of the following two methods is employed to read values:

Single-Register Address Method

The single-register address method internally addresses two sequential 16-bit registers.

In this case, the set-up in CARE is straight forward, namely: Select "32 Bit Register" as the "Addressing Register Type".

Two-Register Address Method

In the two-register address method, the first address stands for the first 16-bit register, and the second address stands for the second 16-bit register.

In this case, in CARE select "16 Bit register" as the "Addressing Register Type", and read the two 16-bit registers individually into a separate datapoint:

- Read the "High" register ("High Word") into one datapoint.
- Read the "Low" register ("Low Word") into a separate datapoint.
- Create a small equation in a CARE control loop that calculates the 32-bit value from the "Low Word" and "High" Word".

Example: A Modbus meter provides Read access to the "Hours Run" information across two 16-bit registers:

- "Hours Run High Word" = Register 20
- "Hours Run Low Word" = Register 21

NOTE: The meter's Modbus documentation must specify which register is the "High Word" and which register is the "Low Word."

The CARE related application will appear as in the accompanying CARE screen shots.

- Create an Analog Input datapoint for Register 20, e.g., "Hours Run High"
- Create an Analog Input datapoint for Register 21, e.g., "Hours Run Low"
- Create a Pseudo Analog datapoint for the Hours Run, e.g., "Hours Run"
- Create a control loop which has the following formula:
  "Hours Run" = "Hours Run High" x 65536 + "Hours Run Low

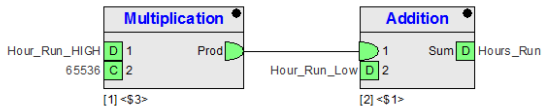


Fig. 40. CARE screenshot of control loop

14. Accessing real 32-bit register values from a single 32-bit register (register type: 32-bit register)

For values greater than 65536 (equaling 16 bit), in rare cases, Modbus slaves can provide data in "real" 32-bit registers. You can identify this implementation by the fact that Modbus address for these registers are continuous: For example, the addresses might be in the following sequence: 100, 101, 102, 103, etc.

15. Transfer of non-numeric register content

All registers in the EAGLE controller are converted into numerical values.

The valid range of register values is limited to ± 16,777,215.

If, e.g., character strings are transferred to the EAGLE, if they exceed the limit, they will be corrupted.

Checking and Verifying

Alarm LED "Modbus error"

There are four possible reasons for this alarm:

- The EAGLE is (still) powering up or starting the application.
- Physical disconnect or miswiring of the Modbus. After the physical Modbus connection is re-established, the EAGLE will report this with a red Alarm LED.
- Some or none of the Modbus RTU slaves on the bus are responding.
- Every Modbus RTU slave on the bus is responding with an error code.

Datapoints Show "Fault"

There are two possible reasons for this alarm:

- The Modbus RTU slave does not respond at all.
- The Modbus RTU slave responds with an error code. Unfortunately, it is not possible to allocate an error code to a specific Modbus register and put only the corresponding datapoint into "fault". This is because side by side registers are read in one package. Hence the error response from the Modbus RTU slave cannot refer to individual registers.

Modbus Statistics in the Standard SVGA Web Pages (800x600)

The standard web pages show the Modbus RTU slaves with their addresses and their status (offline/online)

When a Modbus RTU slave is offline, the associated datapoints will go to fault.

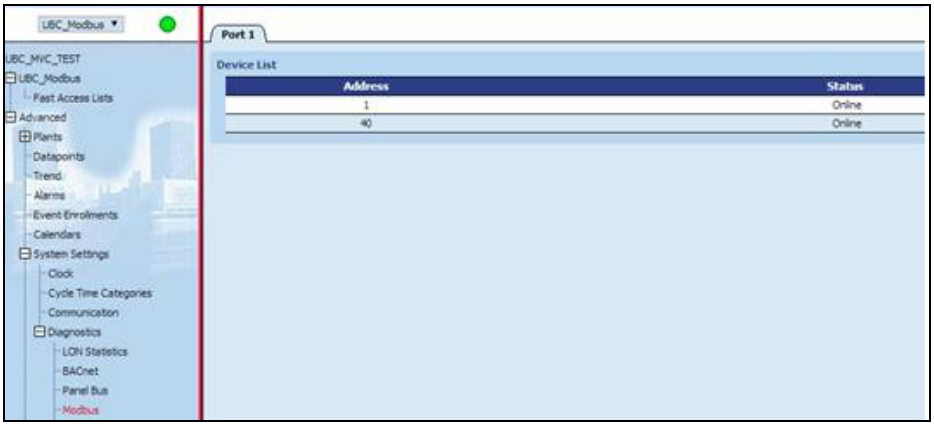


Fig. 41. Screenshot of Modbus Diagnostic screen in the SVGA web pages

Modbus Statistics in File (using controller firmware 3.04.00 and higher)

Beginning with controller firmware version 3.04.00, statistics on Modbus cycle times are no longer supported. Instead, there is a new logfile for the Modbus, which lists the errors per device. It is located in /tmp/logs/ModbusStat.txt.

ModbusStat.txt

```
Dev Pnts  State   InvFct InvAddr  InvVal DevFail Incompl Timeout     Crc   Other
  1:  45, Offline      0,      0,      0,      0;      0,      6,      0,      0;
  2:  45, Offline      0,      0,      0,      0;      0,      6,      0,      0;
  3:  45, Online       0,    164,      0,      0;      0,      4,      0,      0;
 20:  48, Offline      0,      0,      0,      0;      0,      5,      0,      0;
 50:   9, Offline      0,      0,      0,      0;      0,      5,      0,      0;
```

Fig. 42. Screenshot of Modbus statistics file (controller firmware 3.04.00 and higher)

It is intended to implement these statistics in the controller web pages at a later point in time.

Description of Modbus Statistics Columns

In the event of an error, Modbus slaves are able to return error codes. EAGLE is able to display these error codes. The meaning is dependent on the 3rd party implementation. The following description is based on the Modbus specification and provides a guideline.

InvFct / ILLEGAL FUNCTION

The function code received in the query is not an allowable action for the server (or slave). This may be because the function code is applicable only to newer devices, and was not implemented in the unit selected. It could also indicate that the server (or slave) is in the wrong state to process a request of this type, for example because it is unconfigured and is being asked to return register values.

If you get this message, please check if you have used the right kind of register.

InvAddr / ILLEGAL DATA ADDRESS

The data address received in the query is not an allowable address for the server (or slave). More specifically, the combination of reference number and transfer length is invalid. For a controller with 100 registers, the PDU addresses the first register as 0, and the last one as 99. If a request is submitted with a starting register address of 96 and a quantity of registers of 4, then this request will successfully operate (address-wise at least) on registers 96, 97, 98, 99. If a request is submitted with a starting register address of 96 and a quantity of registers of 5, then this request will fail with Exception Code 0x02 "Illegal Data Address" since it attempts to operate on registers 96, 97, 98, 99 and 100, and there is no register with address 100.

InvVal/ ILLEGAL DATA VALUE

A value contained in the query data field is not an allowable value for server (or slave). This indicates a fault in the structure of the remainder of a complex request, e.g., an incorrect implied length. It specifically does NOT mean that a data item submitted for storage in a register has a value outside the expectation of the application program, since the Modbus protocol is unaware of the significance of any particular value of any particular register.

DevFail / SERVER DEVICE FAILURE:

An unrecoverable error occurred while the server (slave) was attempting to perform the requested action. If this happens frequently, restart the Modbus RTU slave.

Incompl.

A response from a Modbus RTU slave was received but it was not complete. If this happens frequently, check the wiring.

Timeout:

The slave was not able to respond in time on some requests. There is no response from the slave. This might happen if the device is not connected or if too many requests are sent to the slave.

Remedy: You can increase the poll rate of the device or increase the "Live Check" or "Live Check Repeat Count" in CARE.

Alarm LED and Alarm Reason

If one or more Modbus devices are offline, the alarm LED on the controller is red.

The reason for the alarm LED is displayed in the HMI of the EAGLE and in XW-Online. See Fig. 43 below.

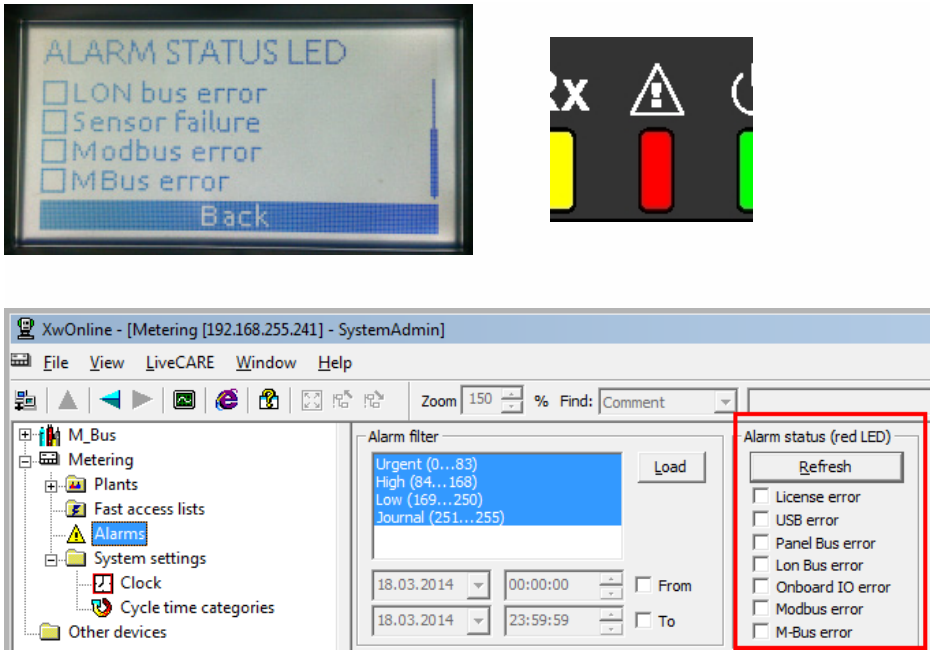


Fig. 43. Alarm LED and alarm reasons screens (Onboard HMI, EAGLE Alarm LED, XW-Online)

Debugging

Onboard Modbus Message Analyzer (using controller firmware 3.04.03 and CARE software 10.06 and higher)

Starting with this controller firmware version, the Modbus messages can be collected using CARE pages. Procedure:

1. Connect with the controller in the setup tab, then click on "Log Files – Configure".
2. Click in the "Select" column and the Modbus on the required RS485 port.
3. Click the Start button; after this, the Status should be enabled.

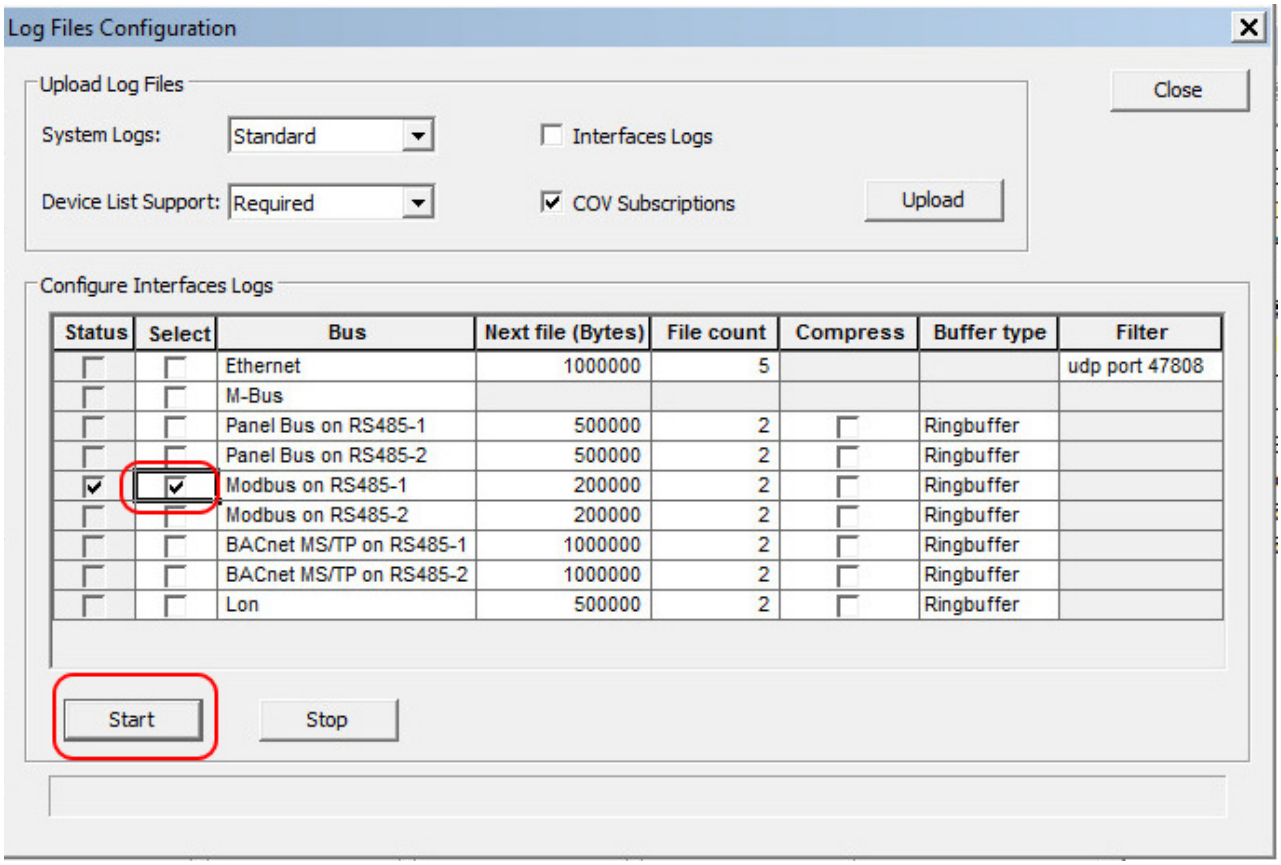


Fig. 44. Configure Modbus logging with CARE on RS485-1 port

After the error occurs in the controller, it is possible to upload the logs. To do this, select the "Interfaces Logs" and click the upload button

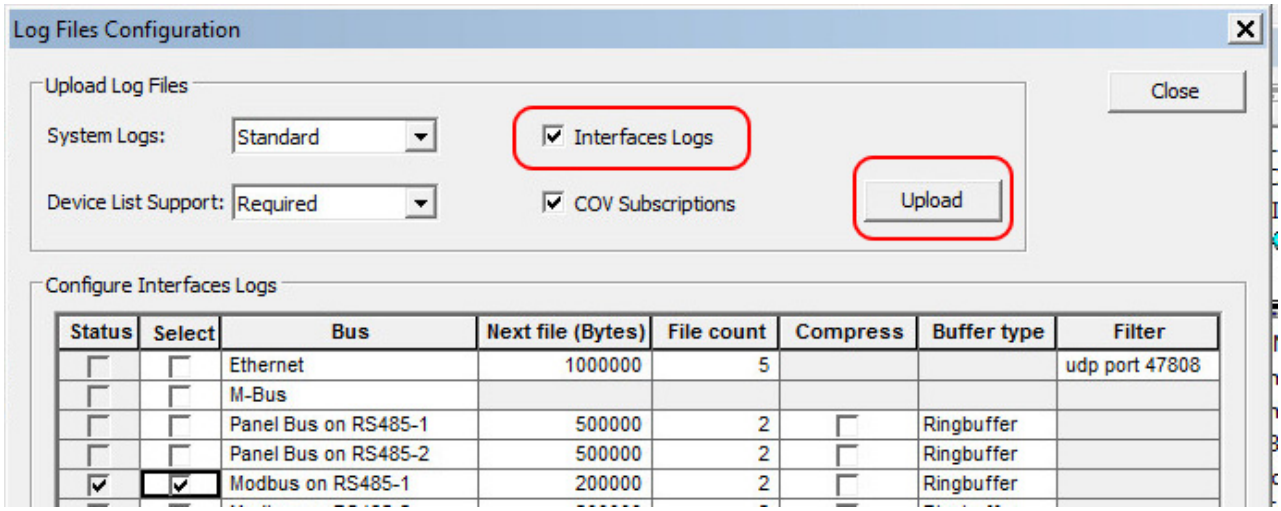


Fig. 45. Upload Modbus logs

After the logs are uploaded, you will get a notification including the path:

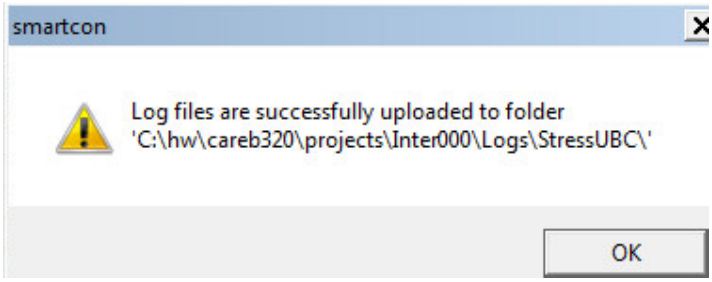


Fig. 46. Confirmation for uploaded Modbus logs

3rd-Party Modbus Message Analyzer ("Sniffer")

Honeywell Schoenaich employs and recommends the following Modbus analyzer from "Frontline", which has been evaluated to be the most helpful and useful analyzer available on the market: http://www.fte.com/products/serialanalyzers-RS422485.aspx

This analyzer is suitable for Windows XP & Windows 7 (32-bit and 64-bit), and it comes with a hardware-Interface, called USB-RS422/485 ComProbe.

Optimizing Communication: Compatibility

If the Modbus RTU slaves are slow in response, and the corresponding datapoints are showing "fault", the following measures can make the communication more effective:

- In the Modbus RTU master channel, set the "Live check" to a slower pace:

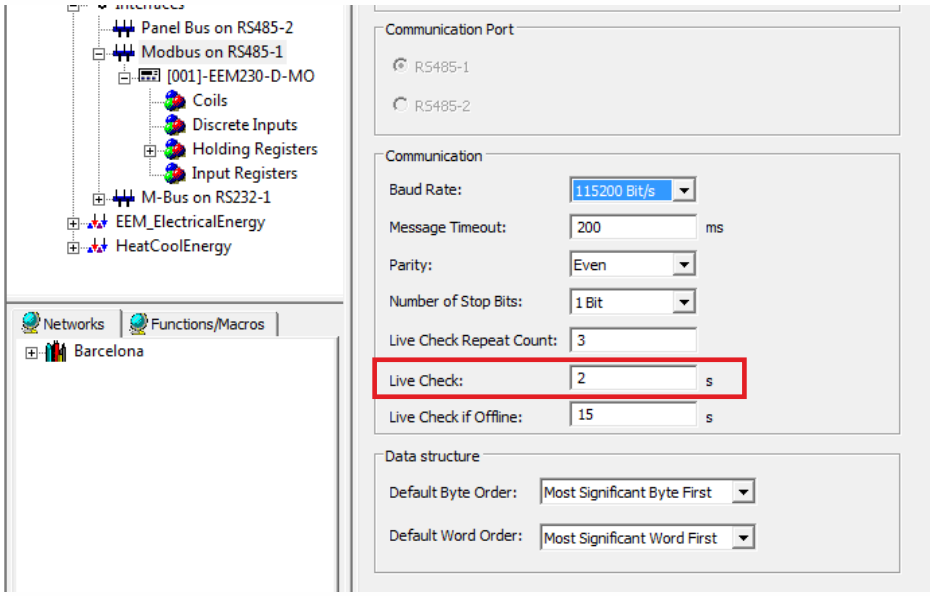


Fig. 47. Screenshot of Modbus Live Check settings in CARE

- In the Modbus datapoint mapping, disable the heartbeat (if not needed), or set it to a slower pace. Too many and/or too frequent heartbeats can overload the EAGLE controller and/or the Modbus RTU slave.

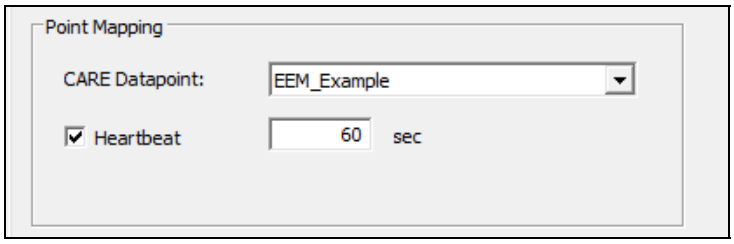


Fig. 48. Screenshot of Modbus point-mapping in CARE

- The application is too big, causing an excessively long cycle time. In this case, try to streamline the application.

We have learned that some Modbus devices do not follow the Modbus_over_serial_line_V1_02 specification. Some are very slow and some do not accept fast requests: If you have trouble integrating a Modbus device, setting higher values for those settings might help:

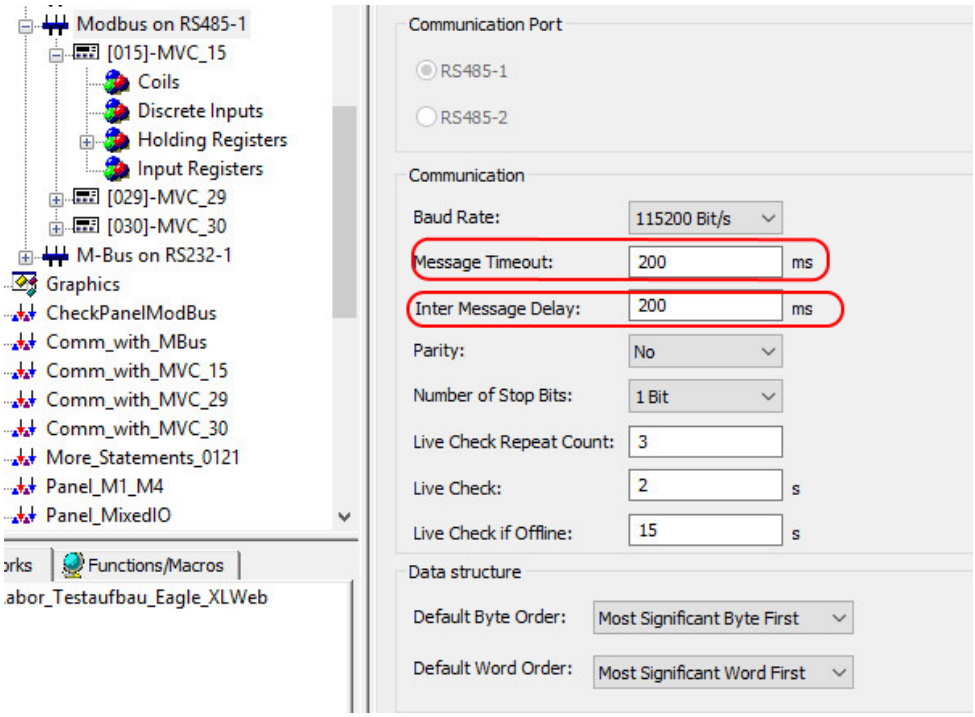


Fig. 49. Screenshot of Modbus Timeouts and delays in CARE

- Message Timeout = 2000 ms helps if the slave takes a long time to respond to the request (firmware 4.02.00 and higher)
- Inter Message Delay = 1000 helps if the wrong slave answers or messages are not accepted by the Modbus slave

Optimizing Communication: Speed

With CARE software 10.08. and higher, Inter Message Delay is introduced. This may slow down existing projects upgraded from CARE 10.07 or lower. To have the fastest Modbus communication, two properties must be taken into consideration:

- Inter Message Delay = 0 as Modbus Communication property
- Poll Rate = 0 as General Device Property

Both options work only for firmware 4.02.00 and higher.

Force Function Code 16 (0x10) Write Multiple Registers for 16-bit Registers (using controller firmware 3.04.04 and CARE software 10.06 and higher)

Firmware 3.04.04 or higher works with the function code 0x06 (write single register) as the default. In the case of jobs requiring the function code 0x10 (write multiple registers), you will need to upload a modified xwmain.ini to the controller. This file must contain the lines:

```
[Modbus]
WriteMultiple=1
```

This will work in all future versions of the firmware.

Procedure:

1. Connect with FTP client (Filezilla or Total Commander with FTP via USB)
2. Download /usr/local/config/xlwebcfg/xwmain.ini, if not present in the /usr/local/config/xlwebcfg/xwmain.ini.template.
3. Modify with supported editor (e.g., Notpad++, not Windows Notepad)
4. If xwmain.ini is present, delete it.
5. Copy xwmain.ini to /usr/local/config/xlwebcfg/xwmain.ini (see http://clfaq.ge51.honeywell.de/index.php?action=artikel&cat=71&id=858&artlang=en)
6. Restart controller with CARE Stop / Start

Supported Modbus RTU Slave Devices

NOTE: All of the following devices have 16-bit addressing registers.

Manufacturer	Device Type	Device Name	Note
ABB	Electrical Meter	B24	
ABB	Electrical Meter	M2M/DMTME	
Belimo	Modbus Automated Fire Damper	BKN230-24-MOD	
Buderus	Block heating station (Block Heat & Power Plant)	EM20	
Carel	Inverter Controller	Combo Drive for uPC	
Carel	Electronic expansion valve	ECD Evolution	
Carel	Inverter	Speed Drive Power +	
Carlo Gavazzi	Electrical Meter	EM24	
Carlo Gavazzi	Electrical Meter	WM3-96	Different word-order inside device.
Condair	Steam Humidifier	DL, ME, RS and EL series	
Emerson	Inverter	VSS Inverter	Modbus interface of Emerson requires external 5VDC supply!
HKW-Elektronik GmbH	Weather Station	WS-K xx Modbus	
Honeywell	Inverter	NXL HVAC SmartDrive HVAC	
IME	Electrical Meter	Nemo 96HDLe	Successfully site-tested by Honeywell partner.
JUMO	Temp. and Pressure converter	DI 308	
KSB	Pump	Calio	
KSB	Pump	PumpDrive BASIC-ADVANCED	
Pro Service		DINster 3xRS	Requires Inter Message Delay.
Produal	Modbus I/O module	MIO 12-PT	
Produal	Wall module	HDH-M-N CO2	
R SCHMITT ENERTEC GmbH	Generator		
Rhoss	Chiller	TCHEBY 122	
Reign	Air Handling Units	Corrigo EXOline v3.0 or higher	Wiring and termination is required.
Romutec	19" Remote I/O and manual override modules	RDC 121	
SAIA	3-phase energy meter	AWD3D5WD00C3A00	
SAIA	Single-phase energy meter	SAIA EEM230-D-MO	
Schako	Fire Damper bus gateway	Easy2-M - EasyBus-Master(1) Easy-M - EasyBus-Master(1)	
Schiele / Entrelec		systron® PM Bus module MODBUS	

Manufacturer	Device Type	Device Name	Note
Schneider Electric (Conzzerv)	Electrical Meter	Power Max EM6400	
Stulz	Air Conditioning Units	MIB 7000	Requires long Communication Timeout.
Swegon	FCU Thermostat	TCO D	
Thies Clima	Weather station	WSC 11	
Thermokon Wallmodule	Wallmodule	WRF08 12T	
Trend	Electrical Meter	EM-MP0/400	
DIEHL/Hydrometer	M-Bus pulse converter	IZAR Port Pulse Mini	
Wilo	Pump	IF-Modul Modbus	
WAGO	PLC with RS485 interface	750-819 with 750-653	
YOSHI	Air Water System	AWS D3-E1 (1)	

(1) Requires writemultiple. This can be achieved by an xwmain.ini setting (see section "Force Function Code 16 (0x10) Write Multiple Registers for 16-bit Registers" on pg. 42).

Non-Supported Modbus RTU Slave Devices

Manufacturer	Device Type	Device Name	Note
ABB	Energy Meter	B23	Problem: The devices sometimes fail to respond to request. Connected datapoints often change from "normal" to "fault" state.
Frico	Pump	SIRe Advanced	Problem: The devices sometimes answer with an exception code and sometimes with correct values. Connected datapoints often change from "normal" to "fault" state.
IFM / EXPERT	MODBUS I/O Expansion module	TP3237 (RT EX9017)	Problem 1: The Modbus gateway requires one external 125Ohm resistor per 0…20mA analog input. Problem 2: The Init mode does not work => No configuration possible!
Intesis	Room controller	ME-AC_MBS-1	Problem: The devices sometimes answer with an exception code and sometimes with correct values. Connected datapoints often change from "normal" to "fault" state.
Merlin Gerin	Energy Meter	CM4000	Problem: 64-bit values are provided across four consecutive 16-bit registers, which the EAGLE controller does not yet support.
Mitsubishi	Generator	MGSB-MERP-STD-014	Problem: The devices sometimes fail to respond to request. Connected datapoints often change from "normal" to "fault" state.
Schneider	Energy Meter	A9MEM3255	Problem: 64-bit values are provided across four consecutive 16-bit registers, which the EAGLE controller does not yet support.
SIEMENS	Energy Meter	PAC2200 PAC 3100	Problem: 64-bit values are provided across four consecutive 16-bit registers, which the EAGLE controller does not yet support.
Sentron	Energy Meter	PAC 4200	Problem: 64-bit values are provided across four consecutive 16-bit registers, which the EAGLE controller does not yet support.

M-BUS

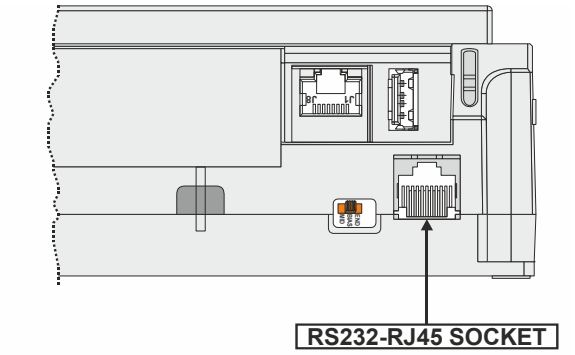


Fig. 50. Side view of EAGLE with RS232 port (M-Bus port)

The EAGLE controller supports M-Bus master functionality via its onboard RS232 / RJ45 socket, which is marked [1] in the above graphic.

In order to communicate with M-Bus meters, it is necessary to connect an M-Bus level converter like PW3, PW20 or PW60 (http://www.relay.de/) between the RS232 interface of the EAGLE controller and the M-Bus meters.

For applications with more than 60 M-Bus meters, the "IZAR center" has been used in field installations successfully.

References

The M-Bus (also called Meter-Bus) protocol is described in the European standard EN 1434-3.

Documents can be found at http://www.m-bus.com/

M-Bus Addressing

- Maximum number of M-Bus devices per EAGLE: 250 (beginning with controller firmware 3.03.05 and CARE software 10.03.00 or higher)
- Address Range: M-Bus slaves can have a primary address between 1 and 250.

NOTE: The address must be unique bus-wide

- Communication rates: 300, 2,400, and 9,600 Baud are supported, individually per M-Bus slave.
- Measurement Cycle: Individually per M-Bus slave, the measurement cycle can be configured from 1 to 604,800 sec (i.e., 1 second to 7 days).

If more that 60 M-Bus meters needs to be connected via one bus, an appropriate Levelconverter needs to be used. Up to CARE 10.04.00 only 60 M-Bus meters per bus have been tested by Honeywell.

M-Bus Cable Specifications and Wiring

Bus Length

- Max. M-Bus length: 350 meters from PW3 / PW20 / PW60, at baud rates of 9.6 kbaud or slower with shielded, twisted pair cable: J-Y-(St)-Y 2 x 2 x 0,8.
- The M-Bus can be extended to 1,000 meters, depending upon the baud rate, and provided that the following electrical limitations are observed:
- Bus voltage must at no point fall below 12 VDC
- Maximum cable capacitance of 180 nF

For bus length extension, M-Bus repeaters can be used, but have not been tested by Honeywell. Hence, it is the responsibility of the installing / commissioning personnel to ensure proper functioning.

Shielding

Shielding is especially recommended when the M-Bus cable is installed in areas with expected or actual electromagnetic noise. Avoiding such areas is to be preferred.

Use shielded, twisted pair cable J-Y-(St)-Y 2 x 2 x 0,8 and connect the shield to a noise-free earth ground – only once per M-Bus connection.

Calculation of M-Bus Cable Length

There is a tool available which allows the following calculations:

- the maximum cable length depending on the number of meters
- the maximum number of meters depending on the cable length

In order to execute this tool, double click on the icon below:

MBusNetworkCalculation.exe

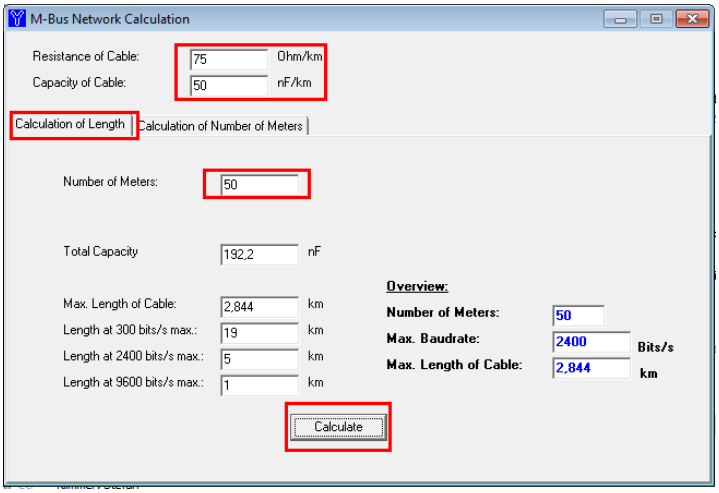


Fig. 51. Screenshot of M-Bus Network Calculation tool

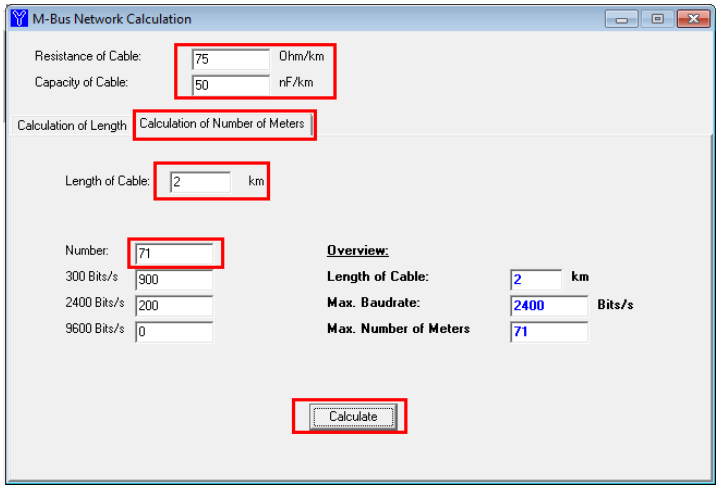


Fig. 52. Screenshots of M-Bus Network Calculation tool

Wiring Topology

M-Bus meters must be connected to the bus cable in parallel.

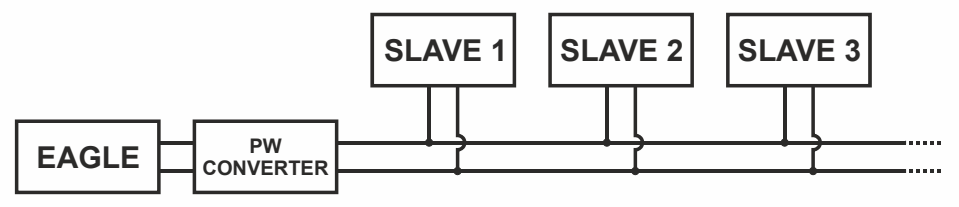


Fig. 53. Allowed M-Bus wiring topology

Cabling CARE to PW3/PW20/PW60

- Use a one-by-one RS232 cable Female / Female

9-Pin sub-D connector pin no. (female)	RS232 function	9-Pin sub-D connector pin no. (female)
1	DCD	1
2	RxD	2
3	TxD	3
4	DTR	4
5	GND	5
6	DSR	6
7	RTS	7

Table 6. RS232 on CARE PC to PW cable specifications

Cabling EAGLE to PW3/PW20/PW60

- Use the XW586 cable between the RS232 / RJ45 socket of the EAGLE and the PW adapters.
- The XW586 cable has a length of 1.8 m, and the pin-out listed in Table 7.
- In case a third-party cable is used instead of the XW586 cable, the third-party cable must have a max. length of 15 meters and a max. cable capacitance of 2,500 pF.

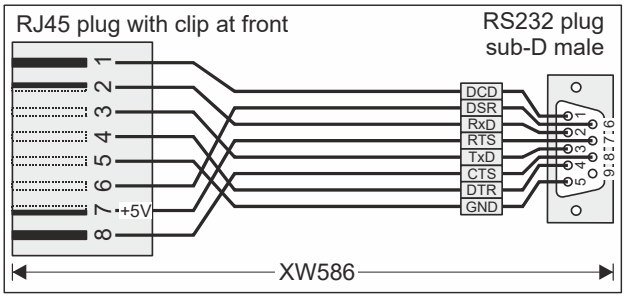


Fig. 54. XW586 power / communication cable details

RJ45 plug, pin no.	RS232 function	9-Pin sub-D connector pin no.
1	DCD	1
2	RxD	2
3	TxD	3
4	DTR	4
5	GND	5
6	DSR	6
7	RTS	7
8	CTS	8
--	Not used	9

Table 7. XW586 cable specifications

RJ45 plug, pin no.	RS232 function
1	
2	RxD
3	TxD
4	
5	GND
6	
7	
8	

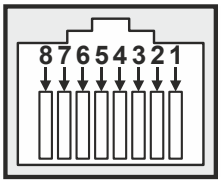


Table 8. EAGLE RS232 / RJ45 socket specifications

Cabling PW3/PW20/PW60 to M-Bus

- Use shielded, twisted pair cable J-Y-(St)-Y 2 x 2 x 0,8.
- Shielding is especially recommended when the M-Bus cable is installed in areas with expected or actual electro-magnetic noise. Avoiding such areas is to be preferred.
- Connect the shield to a noise-free earth ground – only once per M-Bus connection.
- Power the EAGLE controller and the PW M-Bus Adapter with separate transformers – see WARNING below.

NOTE: If, alternatively, only a single transformer is available, when connecting a laptop, PC, web browser, CL-Touch, or 3rd-party touch panel to the USB 2.0 Device Interface on the front of the EAGLE controller, use an optical isolator for the USB connection or substitute an M-Bus Mikro-Master USB (Relay GmbH, D-33106 Paderborn) for the PW M-Bus Adapter.

M-Bus Connection Procedure

1) Install the PW M-Bus Adapter on DIN rail. Insert a screwdriver into the slot in the DIN rail clamp on the underside of the PW and pry downward to loosen clamp until the unit snaps onto the rail.

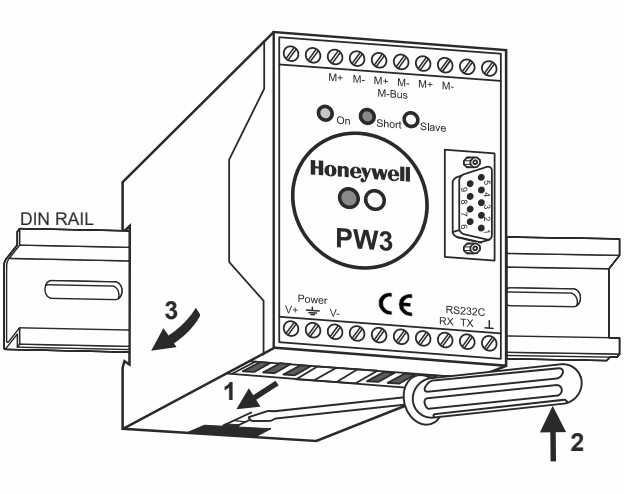


Fig. 55. Mounting of PW (PW3 shown here)

2) Connect the M-Bus devices to the PW M-Bus Adapter. All M+ and M- terminals are connected in parallel in the PW M-Bus Adapter.

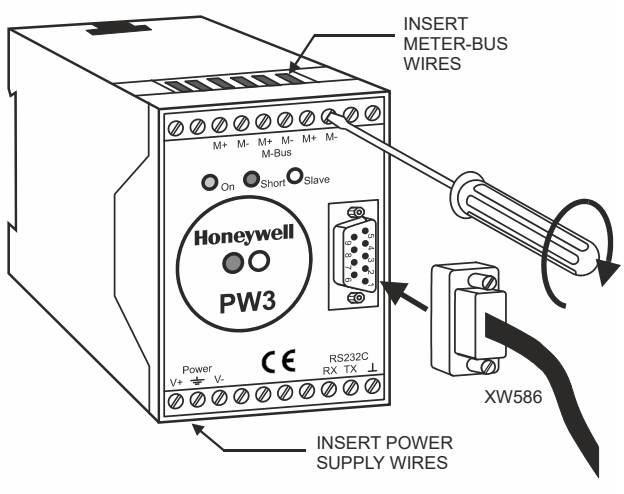


Fig. 56. PW M-Bus adapter connections

3) Connect the PW M-Bus Adapter to the RS232 / RJ45 socket of the EAGLE using the XW586 cable.

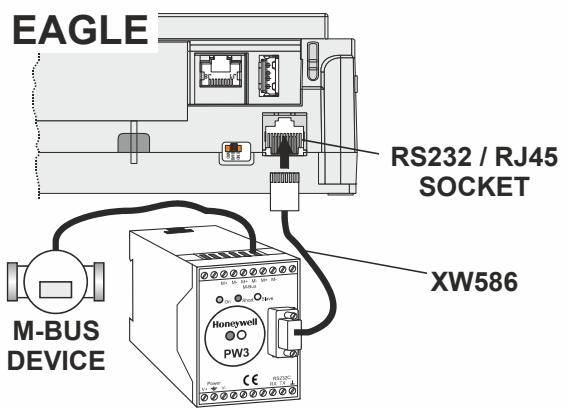


Fig. 57. Connecting the EAGLE to the PW M-Bus adapter

4) Connect 24 V power to the M-Bus Adapter.

⚠ WARNING

Risk of electric shock or equipment damage!

► It is prohibited to power the EAGLE controller with the same transformer used to power other controllers or devices (e.g., the PW M-Bus Adapter).

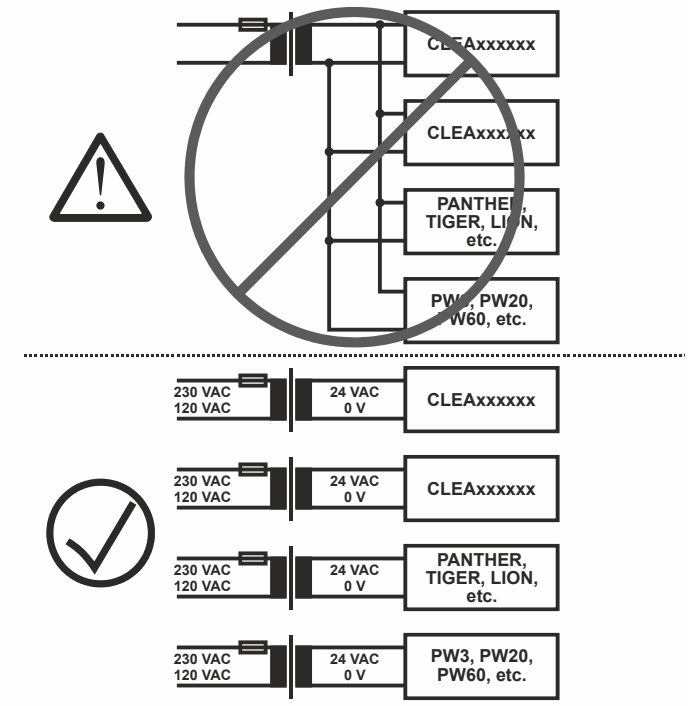


Fig. 58. Connecting power to the PW M-Bus adapter

M-Bus Communication Failure Indication

As soon as the EAGLE runs an application which includes M-Bus points, it will try to communicate with the M-Bus devices.

If an M-Bus slave does not respond to a message from the EAGLE controller, the EAGLE controller will repeat the message several times. If the M-Bus slave still does not respond, the EAGLE will stall the messaging to this M-Bus slave for 60 seconds. After these 60 seconds, the EAGLE controller will again address the M-Bus slave. This cycle will be repeated endlessly as long as there is no answer.

If the EAGLE does not receive any response for ≥ 5 minutes, the mapped datapoint will go to the communication failure condition.

M-Bus "Unreliable_Other" Indication

Each M-Bus device has a status byte with 8 bits. Bits 2, 3, and 4 are used to set all mapped data-records from this M-Bus device to the "Unreliable_Other" condition, which will result in the corresponding datapoint value being set to the last valid value. This means that the value from the M-Bus device is not shown in the datapoint; rather, an old value is shown.

Additionally, each data record has a function field which can indicate the data as "Value with Error." In this case, the corresponding datapoint (not all datapoints) is set to the "Unreliable_Other" condition.

Bit	Meaning	Result
0	app.-specific	ignored
1	app.-specific	ignored
2	low power	reliability flag set to Unreliable_Other
3	perm. error	reliability flag set to Unreliable_Other
4	temp. error	reliability flag set to Unreliable_Other
5	manuf.-spec.	ignored
6	manuf.-spec.	ignored
7	manuf.-spec.	ignored

Table 9. Bits used to set all mapped datapoints

Start-Up Sequence

1) Connect PW3/PW20/PW60 to the EAGLE RS232.
2) Connect one M-Bus meter to the PW3/PW20/PW60.
3) Download application into the EAGLE.
4) Verify live M-Bus values in XW-Online:
    - Live online values of the M-Bus data records of the M-Bus meter can be seen in the present value of the mapped BACnet datapoints.

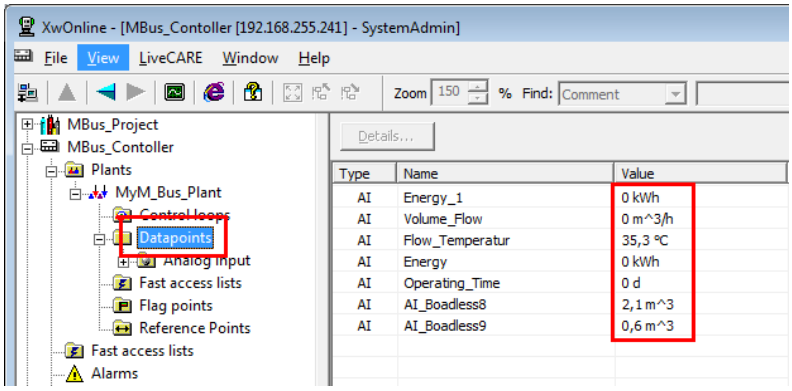


Fig. 59. Screenshot of datapoint list view in XW-Online

- The technical address "RS232-1//" in the point details dialog indicates that the BACnet datapoint is associated with an M-Bus data record on RS 232 port 1.

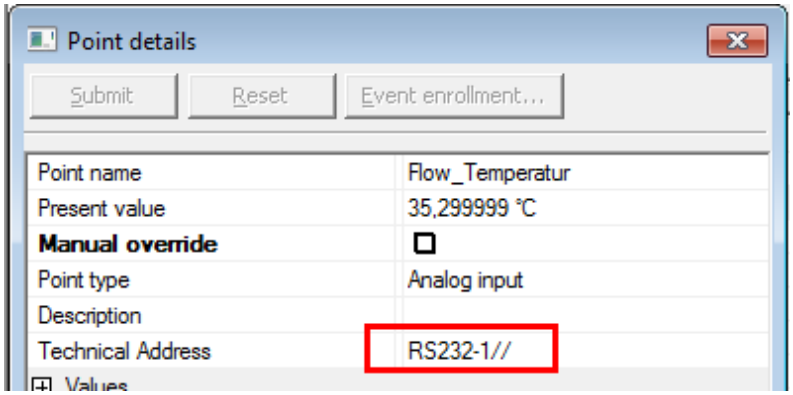
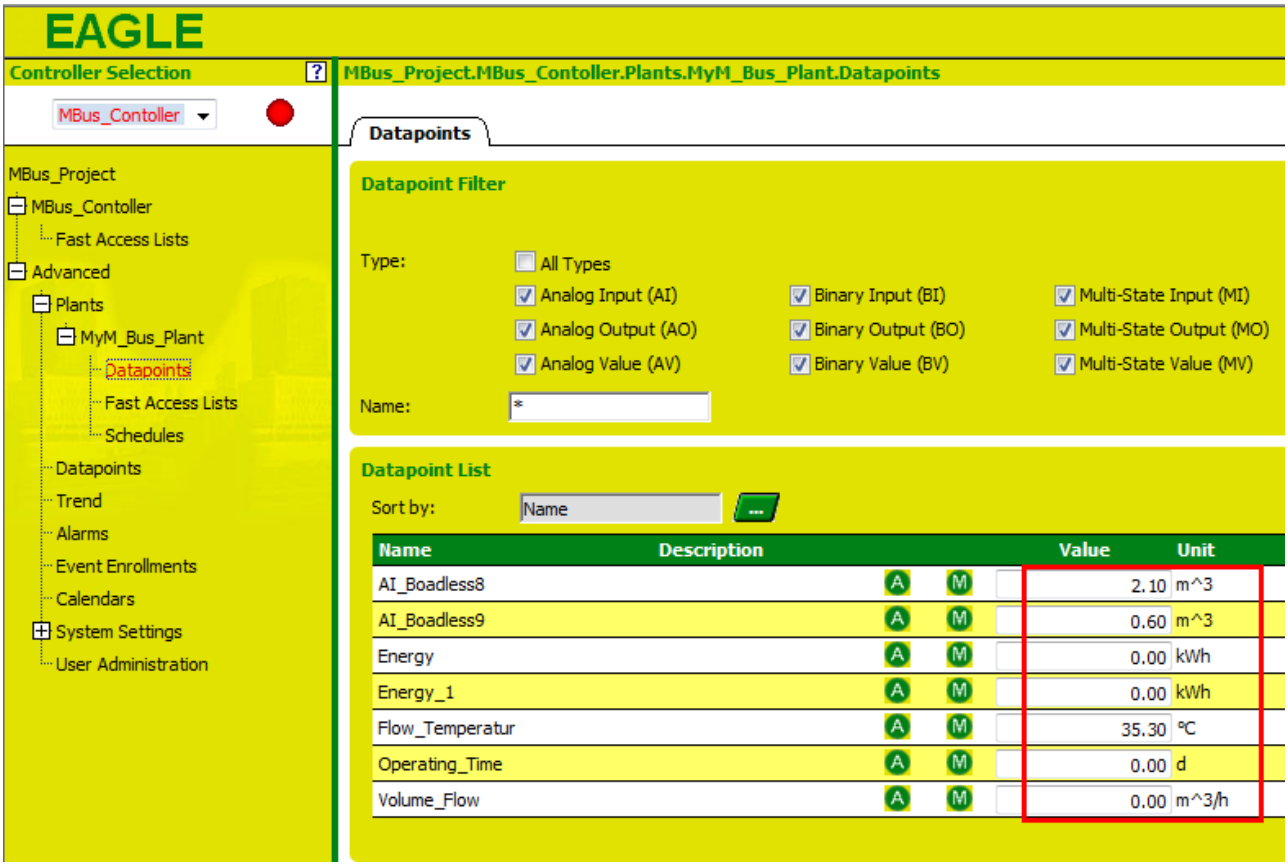


Fig. 60. Screenshot of datapoint properties view in XW-Online

5) Verify live M-Bus values in the SVGA web-interface:

- Live online values of the M-Bus points can be seen from the physical M-Bus device through the present value of the mapped BACnet points.



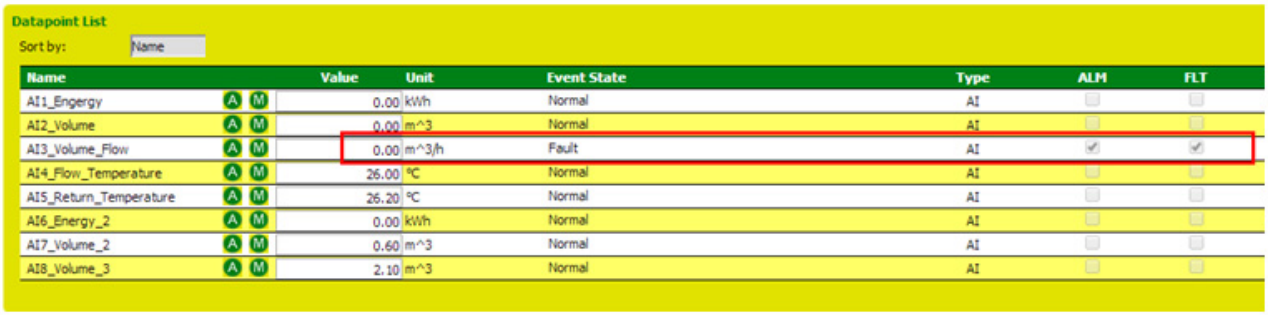


Fig. 61. Screenshots of datapoint list views in SVGA web pages

Checking and Verifying

M-Bus Statistics in the Standard SVGA Web Pages (800x600)

Under "Diagnostics", select "M-Bus"

Available information is

- Device Status
  In the event of communication issues, an "Info" text will be displayed with an error message. Here the Manufacturer documentation is required for the definition per bit.

NOTE: The update rate for the online/offline status Update rate Online / Offline depends on the scan setting for the meter in CARE. Default is 900 seconds (15 minutes).

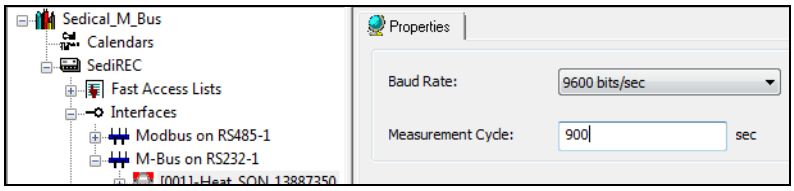


Fig. 62. Screenshot of M-Bus device properties view in CARE

- Communication status
- Last Access (successful read of data)

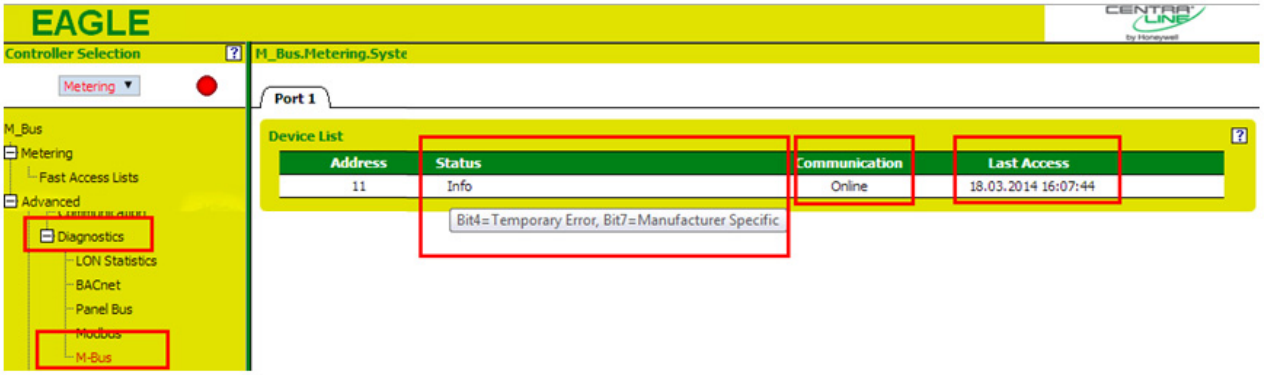


Fig. 63. Screenshot of M-Bus Diagnostics view in the SVGA web pages

Alarm LED and Alarm Reason

If one or more M-Bus meters are offline, the alarm LED on the controller is red.

The reason for Alarm LED is displayed in HMI of the EAGLE and in XW-Online. See Fig. 64 below.

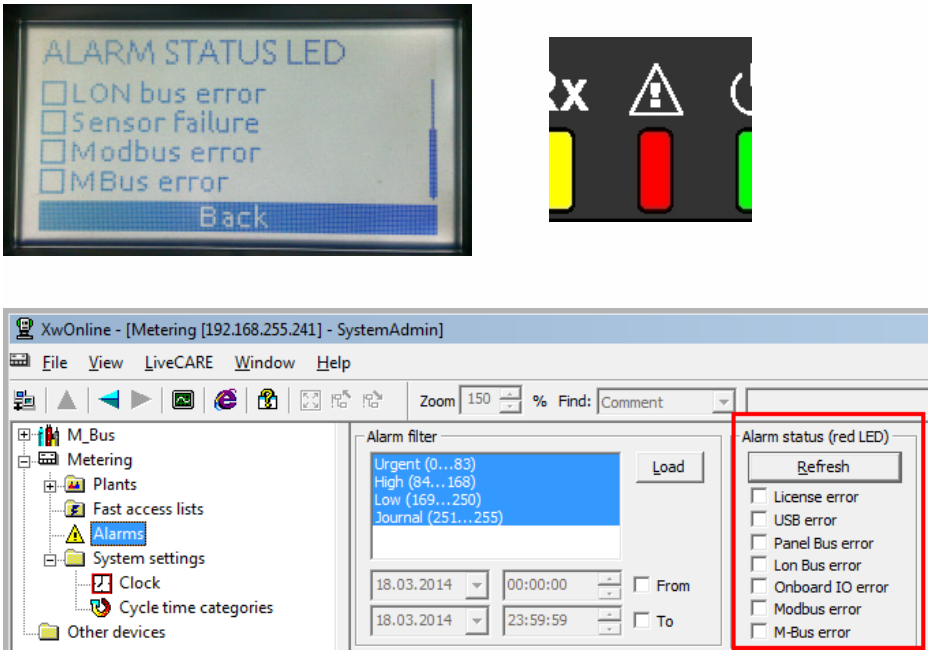


Fig. 64. Alarm LED and alarm reasons screens (EAGLE Alarm LED, Onboard HMI, XW-Online)

Debugging (using controller firmware 3.04.03 and CARE software 10.06 and higher)

1. Enable and upload M-Bus log-files from the controller

Starting with this controller firmware version, the M-Bus messages can be collected using CARE. Procedure:

a. Connect with the controller in the setup tab; after this, click on "Log Files – Configure"

b. Click in the "Select" column and the M-Bus.

c. Click the Start button; after this, the Status should be enabled.

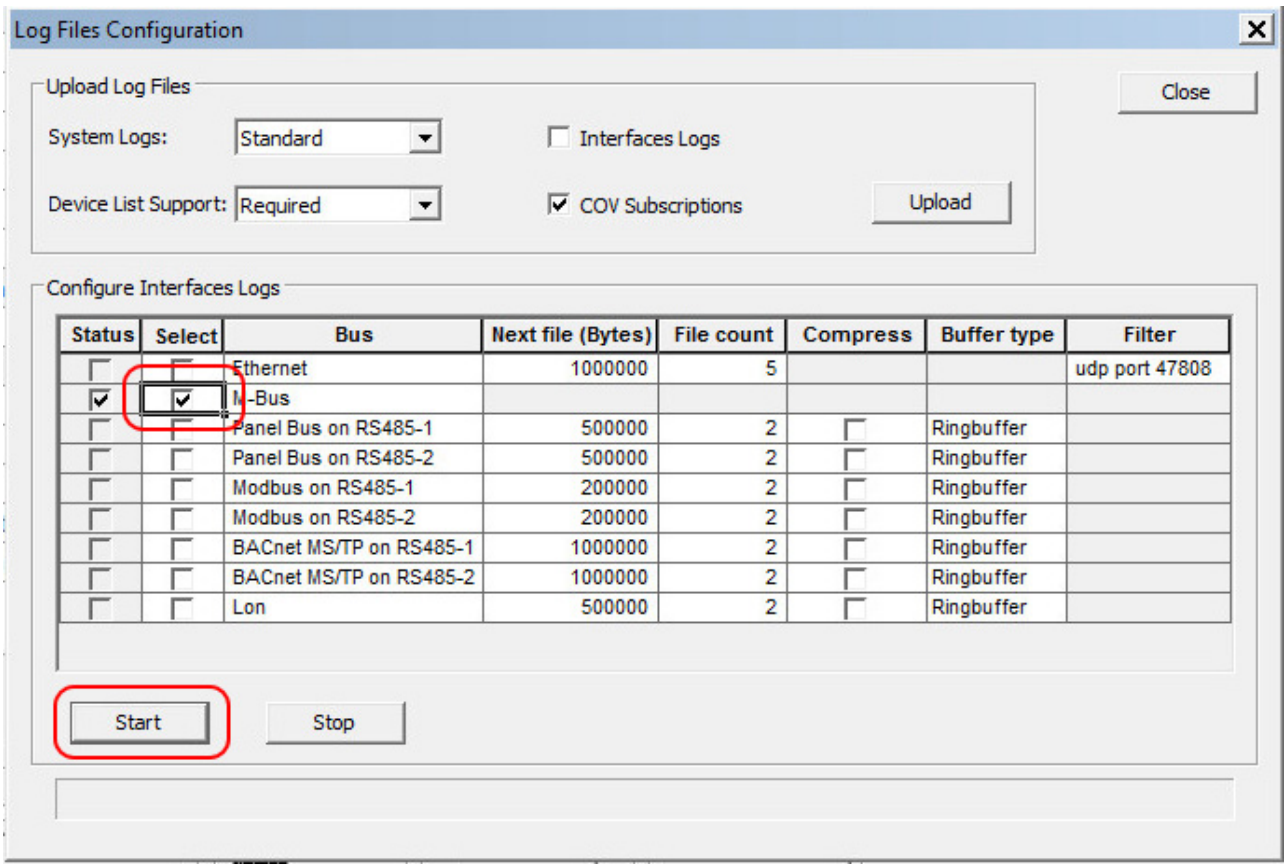


Fig. 65. Configure M-Bus logging with CARE

Please reduce the M-Bus scan time in the SVGA webpages. After the controller has read the data from the M-Bus device, the logs will be uploaded. For this, select the "Interfaces Logs" and click the upload button.

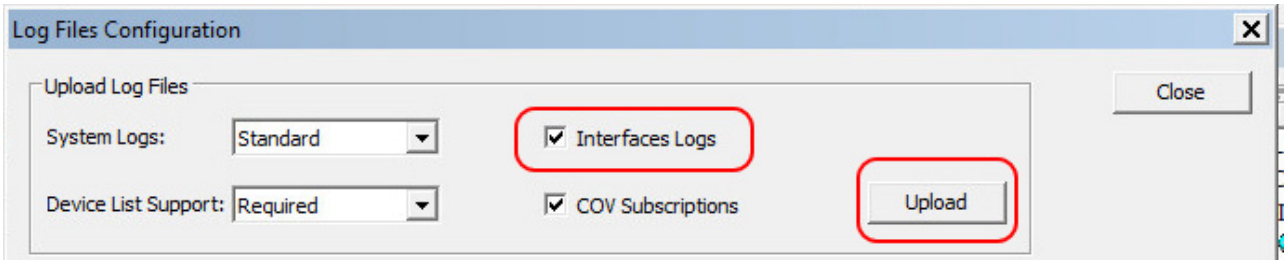


Fig. 66. Upload M-Bus logs

After the logs are uploaded, you will get a notification including the path:

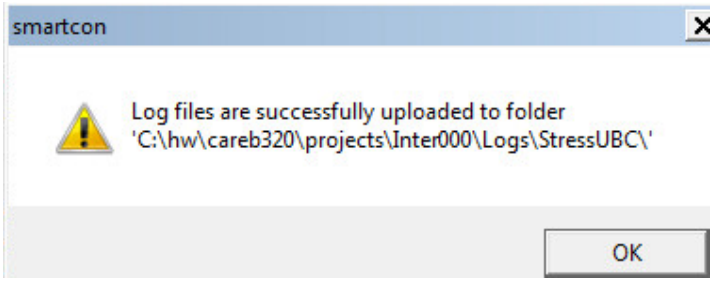


Fig. 67. Confirmation for uploaded M-Bus logs

2. Analyze the log data

File: sbytes001

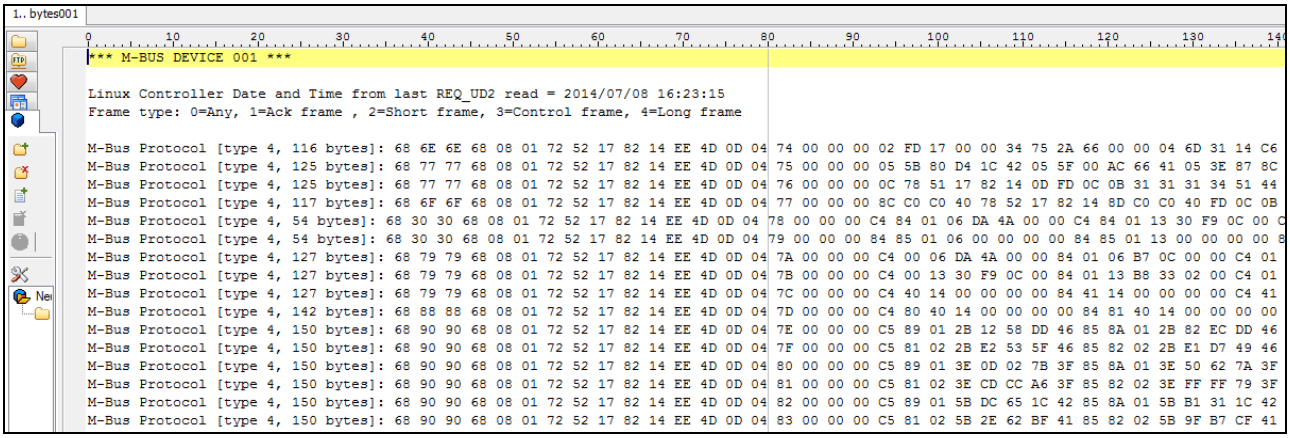


Fig. 68. sbytes001 file screenshot – opened in text editor

The bites file includes all the M-Bus frames as raw data as read via the M-Bus protocol

This always shows the data from the last read.

The update interval is equal to the M-Bus reading cycle defined in CARE.

File: datarec

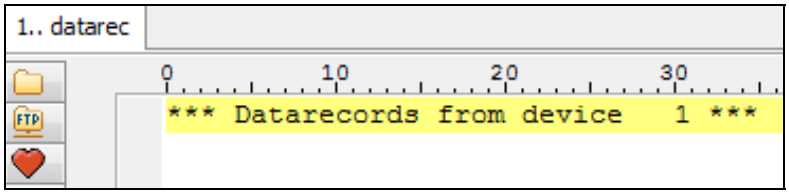


Fig. 69. datarec file screenshot – opened in text editor

This file contains information text, only.

File: info_app

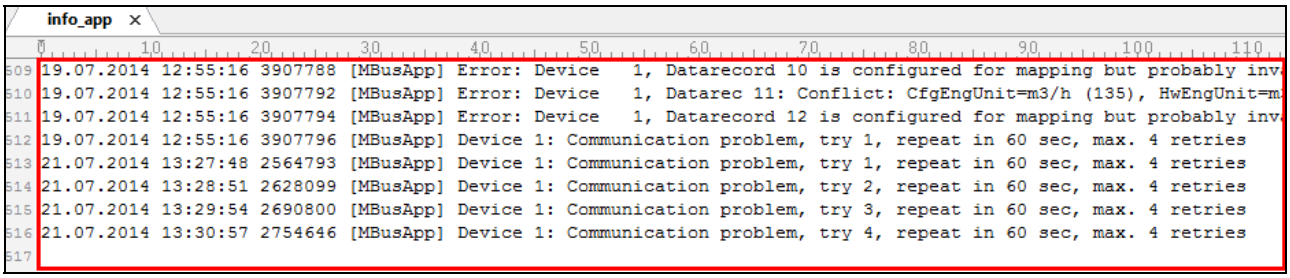


Fig. 70. info_app file screenshot – opened in text editor

Displays if there was an error during reading (communication).

NOTE: The Info_app file is updated after each new read. If the file size increases 500000 bytes, then the file is deleted and created again.

File: Info_hw

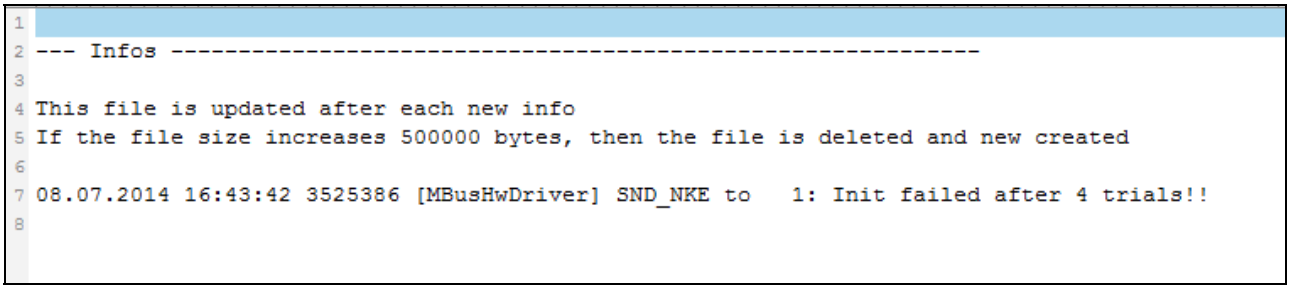


Fig. 71. info_hw file screenshot – opened in text editor

Date and time with error reason are given.

NOTE: If the file size increases 500000 bytes, then the file is deleted and created again.

File: protocol

```
[MBusHwDriver] SND_NKE to   1: Init  0/3
[MBusHwDriver] SND_NKE to   1: SuccessfullyAnswered = 1
[MBusHwDriver] SND_NKE to   1: Wait 100 msec
[MBusHwDriver] SND_NKE to   1: Init  1/3
[MBusHwDriver] SND_NKE to   1: SuccessfullyAnswered = 2
[MBusHwDriver] SND_UD  to   1: Reset 0/2
[MBusHwDriver] SND_UD2 to   1: Scan  0/2
[MBusHwDriver]: Device 1: run_MBus_hwDriver() returns error to MbusApplication()
[MBusHwDriver] SND_NKE to   1: Init  0/3
[MBusHwDriver] SND_NKE to   1: SuccessfullyAnswered = 0
[MBusHwDriver] SND_NKE to   1: Wait 200 msec
[MBusHwDriver] SND_NKE to   1: Init  1/3
[MBusHwDriver] SND_NKE to   1: SuccessfullyAnswered = 0
[MBusHwDriver] SND_NKE to   1: Wait 400 msec
[MBusHwDriver] SND_NKE to   1: Init  2/3
[MBusHwDriver] SND_NKE to   1: SuccessfullyAnswered = 0
[MBusHwDriver] SND_NKE to   1: Wait 600 msec
[MBusHwDriver] SND_NKE to   1: Init  3/3
[MBusHwDriver] SND_NKE to   1: SuccessfullyAnswered = 0
[MBusHwDriver] SND_NKE to   1: Init failed after 4 trials!!
```

Fig. 72. protocol file screenshot – opened in text editor

Creates an entry per each read request to the meter device indicating the meter's response.

Each successful read request is indicated by "1" or "2".

If a read request has failed, this info will be added to the file and is indicated with a "0".

List of responses

0 = Read request failed

1 = Meter is being initialized

2 = Feedback of meter.

3 = Meter is being initialized.

4 = Feedback of meter.

5 = Application of meter is being reset in order that the meter starts scanning at DR = "0".

6 = Raw data of meter is being received by the EAGLE.

NOTE: If the file size increases 500000 bytes, then the file is deleted and created again.

File: scan00x

```
*** M-BUS DEVICE 001 ***

Linux Controller Date and Time from last read = 2014/07/14 15:36:44

Primary Address=1, Baudrate=9600, ProductName (if known)=|Sontex Supercal 531|
Manufacturer=|SON|, Id=|14821752|, Version=13, Medium=|Heating outlet|, AccessNumber=193, Signature=0, Number of Datarecords=236
Status=0

*** DATA RECORDS ***

  DR            Description              Value          Factor      Unit   Function           Timestamp          RefValue RefUnit
 0.0            Error flags  |               0|            n.a.      n.a.    Instant.  2014-07-14T15:36:27            n.a.
 1.0      Actuality Duration ---       34748 ---          INVALID   minutes   Err Val  2014-07-14T15:36:27       34748.0000 min
 2.0 Time Point (time date)  12.07.2014 20:03:00            n.a.      n.a.    Instant.  2014-07-14T15:36:27            n.a.
 3.0                 Energy          25125.000       1000.000000        Wh    Instant.  2014-07-14T15:36:27   25125000.0000 Wh
 4.0                 Volume        1136520.000          0.001000       m^3    Instant.  2014-07-14T15:36:27      1136.5200 m3
 5.0      Unknown (VIF=0x79)  |               1|            n.a.      n.a.    Instant.  2014-07-14T15:36:27            n.a.
 6.0                 Volume              0.000          0.010000       m^3    Instant.  2014-07-14T15:36:27         0.0000 m3
 7.0      Unknown (VIF=0x79)  |               2|            n.a.      n.a.    Instant.  2014-07-14T15:36:27            n.a.
```

Fig. 73. scan00x file screenshot – opened in text editor

Includes the data last read from the meter

The number of the file corresponds to the primary address of the meter, e.g., scan001 = primary address 1 of meter

DR (Data Record) are the data that are read from the device. In CARE, the same numbers will be displayed for the data records.

NOTE: CARE starts counting from 1. For example, the DR in log file 3 for "Energy" will be listed in CARE as "004 – Energy".

All read values of the raw M-Bus data are converted to a reference value. In CARE, this value needs to be converted in relation to the selected engineering unit.

File: Schedule

```
--- M-Bus Communication ------------------------------------------------------

This file is updated after each reading of MBus messages

Cycle   = Measurement cycle time
Success = Number of successfully read of M-Bus messages since power up
Fault   = Number of faulty M-Bus messages since power up

| Addr | Last read try       | Cycle [sec] |    Success |       Fault |
|  001 | 14.07.2014 15:36:45 |         900 |      71786 |         156 |
```

Fig. 74. schedule file screenshot – opened in text editor

Lists the last execution time per M-Bus meter.

The "Cycle time" is the definition done in CARE.

Success = Number of successfully read M-Bus messages since power-up.

Fault = Number of faulty M-Bus messages since power-up.

File: Send00x

```
--- Primary Address 001 ------------------------------------------------

This file is updated every measurement cycle. It shows then measurement cycle and whether a value is send.
A line with only the timestamp means that a value is read but it has no significant change. (Shows measurement cycle).
If the file size increases 300000 bytes, then the file is deleted and new created.
Reason for Reliability UNRELIABLE_OTHER: Mapped DR with function = Err value
Reason for Reliability COMM_FAILURE: MBus reading error (primaryAddr, baudrate, ...)
  or mapping conflict (device record index, eng. units)

|Timestamp          |            DR 003|Rel|            DR 004|Rel|            DR 009|Rel|            DR 010|Rel|
|                   |            Energy|   |            Volume|   |  Flow temperature|   |Return temperature|   |
|14.07.2014 06:54:44|                  |   |       1125.285034|OK |         29.038818|OK |         13.951172|OK |
|14.07.2014 06:55:13|      24963.000000|OK |       1125.300049|OK |         28.440918|OK |         13.951660|OK |
|14.07.2014 06:55:41|                  |   |       1125.314941|OK |         27.918457|OK |         13.951172|OK |
|14.07.2014 06:56:09|                  |   |       1125.328979|OK |         28.866455|OK |         13.952148|OK |
|14.07.2014 06:56:37|                  |   |       1125.342041|OK |         26.890137|OK |         13.951172|OK |
|14.07.2014 06:57:06|      24964.000000|OK |       1125.354004|OK |         29.071289|OK |         13.947266|OK |
|14.07.2014 06:57:34|                  |   |       1125.369019|OK |         26.692383|OK |         13.952148|OK |
```

Fig. 75. Send00x file screenshot – opened in text editor

For each meter, a separate file set will be created depending on its primary address: 001 = prim address 1.

In the file set, only those points will be listed that have a datapoint mapping.

DR (Data Record) are the data that are read from the device. In CARE, the same numbers will be displayed for the data records.

NOTE: CARE starts counting from 1. E.g., the DR in log file 3 for "Energy" will be listed in CARE as "004 – Energy"

The value that is displayed in the list is the value sent to the datapoint. If there is a "I" instead of the value, that means that there was no (only a minor) change on the datapoint value. If there is no change at all for all Data Records, the line will be empty.

Tested M-Bus Meters

AQUAMETRO

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
Calec ST	Heat		X	X	-

Danfoss

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
Sonometer 1100 with M-Bus communication Module	Heat		A	A	-

NZR

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
DHZ 5/65	Energy		X	X	X

DIEHL / Hydrometer

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
IZAR Port Pulse Mini	M-Bus pulse converter		-	X	-

ELSTER INSTROMET

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
Absolute Encoder S1/D for RVG100	Energy (Gas)		X	X	-

Engelmann Sensor

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
SensoStar 2C	Heat		X	X	X

EMU Electronic

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
EMU Allrounder 3/75	Energy		X	X	X

Honeywell

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
		All OS-Numbers starting with	300	2400	9600
EW773 Series with M-Bus	Heat	EW7730K, EW7730M, EW7731K, EW7731M, any other equipped with module EWA3022071	A	A	-
EW447 / EW450	Heat	EW447M, EW450M	A	A	-
EW130 Series with M-Bus	Water	EW1300BM, EW1301BM	A	A	-
EW545 Series with M-Bus	Heat	When equipped with module EWA3022071	A	A	-
EW500 Series with M-Bus module	Heat	EW5001CM, any other equipped with module EWA500C-MBUS	-	X	-
EW110 Series with M-Bus module	Water	EW1100CM, EW1101CM, any other equipped with module EWA110Cxxx-MBUS	-	X	-
TMP-A	Water		X	X	X
LU2925M1269	Water		X	X	-

Itron

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
CF51, M2_ACW_13183194	Heat	1)	X	X	X
Mbus Cyble 2.0	Water		X	X	-

1) Display of changed values on its own HMI is slow. Seen with Return Temperature.

Landys+Gyr

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
T230	Heat		A	A	-

Relay

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
PadPuls M4L			A	A	-
PadPuls M1			A	A	A

SBC
Saia-Burgess Controls

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
EEM230-D-M	Energy		A	A	A
AWD3D5WM00C3A00	Energy		A	A	A
ALE3D5F	Energy		A	A	A

Schneider Electric

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
iEM3235	Energy		X	X	X

Sensus

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
PolluStat E	Heat		A	A	-
PolluTherm	Heat		A	A	-
PolluCom E	Heat		A	A	-
Residia MUK (MA 2207 / MS 8300)	Water		A	A	-
HRI-B1/08	Water		A	A	-

Sontex

Meter Device Type	Meter Category	Note	Baud Rates A = Auto detection		
			300	2400	9600
Supercal 531	Heat		X	X	X

Techem

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
Classic S	Heat		X	X	-
Classic SII	Heat		X	X	-
Ultra S3	Heat		X	X	-

Watts Industries

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
CAMICAL	Heat		-	X	-

NON-Supported M-Bus Meters

Diehl Metering

Meter Device Type	Meter Category	Notes	Baud Rates A = Auto detection		
			300	2400	9600
Scylar INT M		After power cycle, meter will not run anymore.			

LONWORKS

References

See also Excel 50/5000 LONWORKS Mechanisms (Product Literature No.: EN0B-0270GE51) for LONWORKS network segment load issues and optimization.

The EAGLE can be connected to LONWORKS networks.

Via its USB 2.0 Host interface (see Fig. 76 below), the EAGLE Controller can be connected to the IF-LON external Interface adapter and thus to LONWORKS networks.

The IF-LON is equipped with a free-topology transceiver (FTT10A) for communication (at a data transmission rate of 78 Kbaud) on LONWORKS networks (using the LonTalk protocol).

IF-LON Wiring

The IF-LON comes with a 0.8-meter-long, standard USB cable. The maximum length of wire from the EAGLE controller to the IF-LON is 0.8 m.

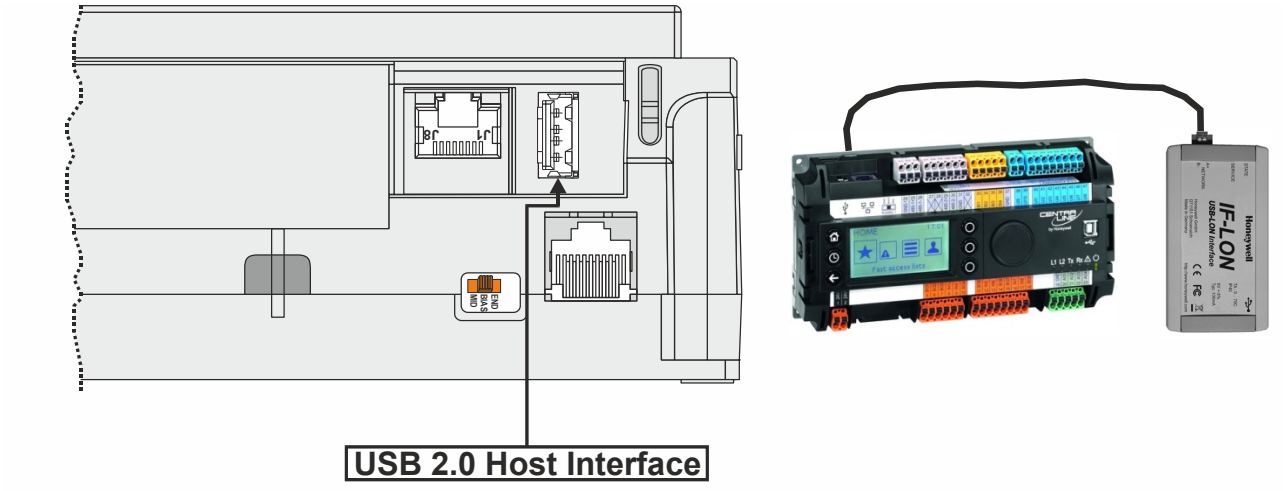


Fig. 76. EAGLE USB 2.0 Host interface and wiring to IF-LON

IF-LON Mounting

The IF-LON is designed for mounting onto panel rail.

For more details on mounting, please see IF-LON – Mounting Instructions (Product Literature No.: MU1B-0545GE51).

LonWorks Bus Cable Specifications and Wiring

The LONWORKS network is insensitive to polarity, eliminating the possibility of installation errors due to miswiring.

IMPORTANT

Do not bundle wires carrying field device signals or LONWORKS communications together with high-voltage power supply or relay cables. Specifically, maintain a min. separation of 3 inches (76 mm) between such cables. Local wiring codes may take precedence over this recommendation.

IMPORTANT

Try to avoid installing in areas of high electromagnetic noise (EMI).
The LONWORKS transceiver can be affected by electromagnetic fields generated by frequency converters. If possible, position frequency converters in a different cabinet, or allow a min. distance of 18 in. (50 cm) between frequency converters and their respective cabling, and Distributed I/O modules.

LONWORKS Cable Types

- Use wire with a minimum size of 20 AWG (0.5 mm2) and a maximum size of 14 AWG (2.5 mm2).
- Cable length depends on the chosen wiring topology, and are described in the following sections.

LonWorks Wiring Topology

Different network configurations - daisy-chain, loop, and star configurations, or any combination thereof - are possible, as long as the max. wire length requirements given below are met.

All of the following specifications apply to a single network segment. Multiple segments may be combined using repeaters in order to increase the allowed number of nodes and distance.

Doubly-Terminated Daisy Chain

The recommended configuration is a daisy chain with double terminations (see Fig. 77). This layout allows for the max. length of the LONWORKS bus, and its simple structure presents the least number of possible problems, particularly when adding on to an existing bus.

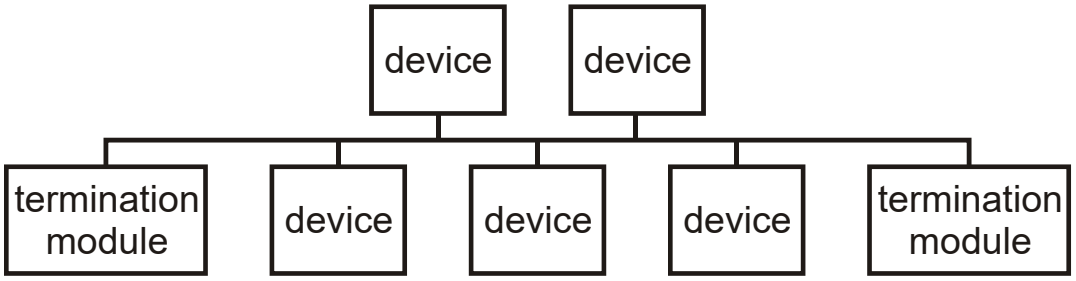


Fig. 77. Recommended configuration of doubly-terminated busses

Table 10. Cable types and bus length of doubly-terminated busses

cable type	max. bus length for segments with FTT-10 or FTT-10A transceivers, only
Belden 85102	8,900 ft (2,700 m)
Belden 8471	8,900 ft (2,700 m)
Level IV, 22AWG (9D220150)	4,600 ft (1,400 m)
Plenum rated Level IV, 22AWG (9H2201504)	4,600 ft (1,400 m)
JY (St) Y 2x2x0.8, twisted pair	3,000 ft (900 m)
TIA568A Category 5 24AWG, twisted pair	3,000 ft (900 m)

NOTE: When possible, use Honeywell AK3781, AK3782, AK3791, or AK3792 cable (US part numbers).

Free-Topology Wiring

Free topology requires only one termination and allows a variety of bus configurations, see Fig. 78 below:

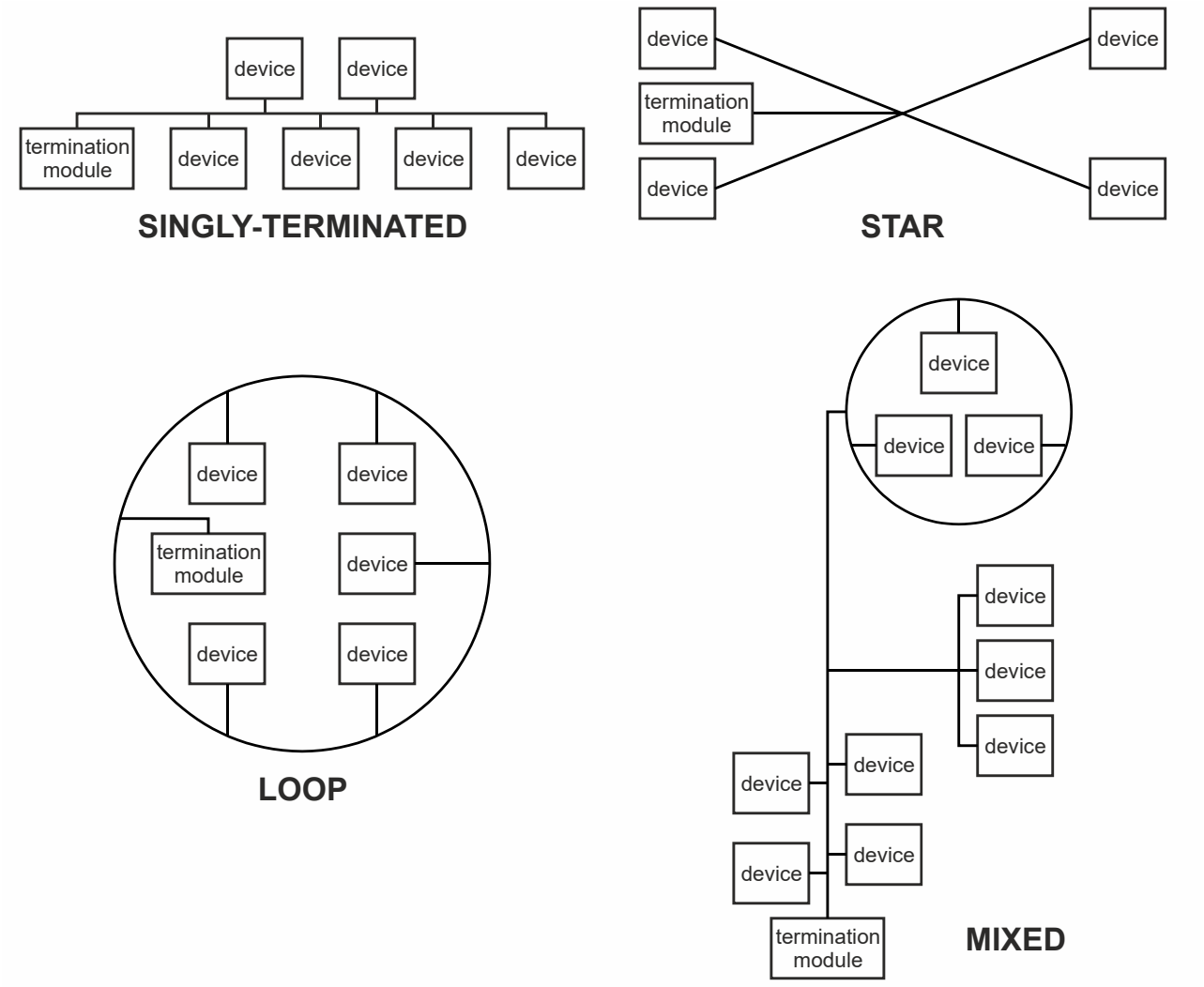


Fig. 78. Examples of free topology bus layouts

Distance Rules

The free topology transmission (FTT) specification includes two requirements which must be met for proper system operation: The distance from each transceiver to all other transceivers and to the termination (including the LPT-10 termination, if used) must not exceed the *max. node-to-node distance*. If multiple paths exists, the *max. total wire length* is the total amount of wire used (see Table 11).

Table 11. Cable types and bus length of free topology (singly-terminated) busses

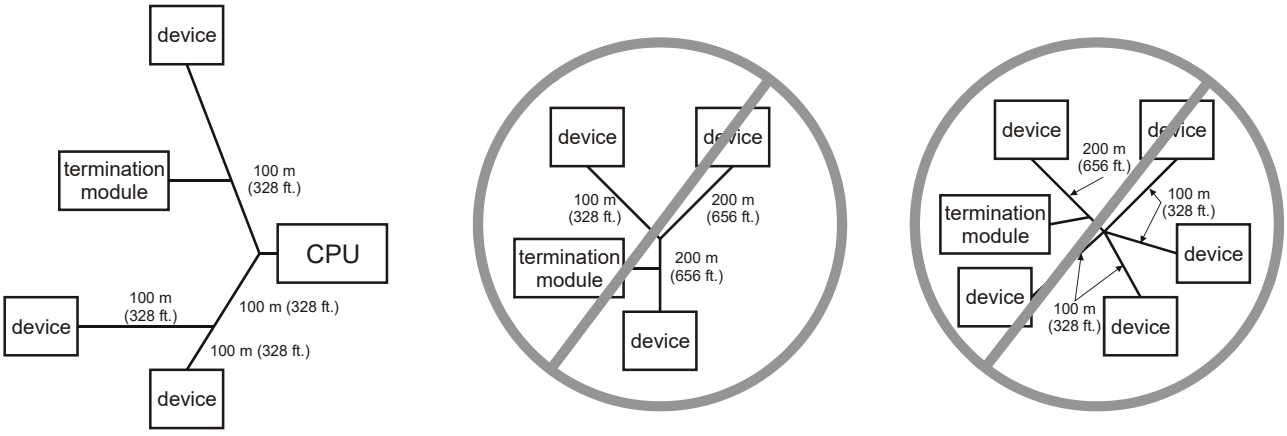
cable type	max. node-to-node distance	max. total wire length
Belden 85102	1,650 ft (500 m)	1,650 ft (500 m)
Belden 8471	1,300 ft (400 m)	1,650 ft (500 m)
Level IV, 22AWG (9D220150)	1,300 ft (400 m)	1,650 ft (500 m)
Plenum rated Level IV, 22AWG (9H2201504)	1,300 ft (400 m)	1,650 ft (500 m)
JY (St) Y 2x2x0.8, twisted pair	1,050 ft (320 m)	1,650 ft (500 m)
TIA568A Category 5 24AWG, twisted pair	825 ft (250 m)	1,500 ft (450 m)

NOTE: When possible, use Honeywell AK3781, AK3782, AK3791, or AK3792 cable (US part numbers).

IMPORTANT

Do not use different wire types or gauges on the same segment of the LONWORKS bus. The step change in line impedance characteristics would cause unpredictable reflections on the bus.

Examples of allowed and non-allowed free topology layouts for cable JY (St) Y 2x2x0.8 are shown below:



ALLOWED:
node-to-node = 200 m (656 ft.)
total wire length = 400 m (1312 ft.)

NOT ALLOWED:
node-to-node = 400 m (1312 ft.)
total wire length = 500 m (1640 ft.)

NOT ALLOWED:
node-to-node = 200 m (656 ft.)
total wire length = 600 m (1968 ft.)

Fig. 79. Example of allowed/non-allowed free topology layouts (max. node-to-node distance: 320 m, max. wire length: 500 m)

NOTE: In the event that the limit on the total wire length is exceeded, then FTT physical layer repeaters (FTT 10A) can be added to interconnect segments and increase the overall length by an amount equal to the original specification for that cable type and bus type for each repeater used. For example, adding repeaters for a doubly-terminated bus using JY (St) Y 2x2x0.8 cable increases the max. length 900 m (3,000 ft) for each repeater.

Termination

Depending upon the chosen network configuration – see previous sections - one or two terminations may be required.

The following LONWORKS termination module is available: **XAL-Term2**

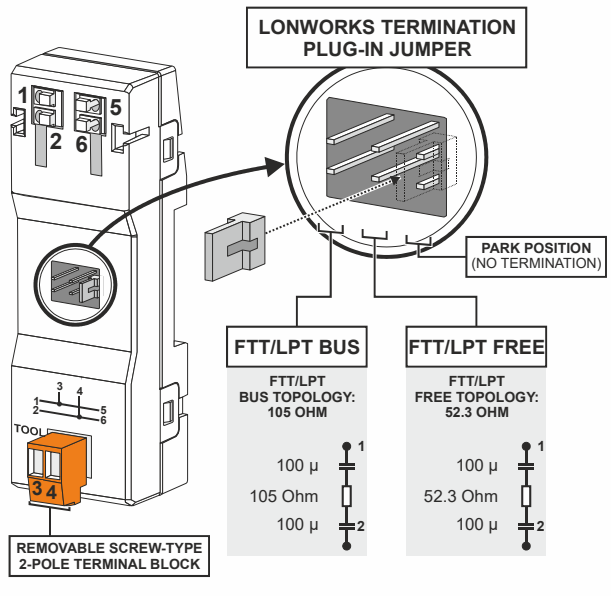


Fig. 80. LONWORKS connection and termination module

Advantages compared to other LONWORKS terminations include:

- Removable terminal to connect your tool, e.g., CARE 4.0, Excelon, LonMaker for Windows, etc. to the LONWORKS network without disturbing LONWORKS communication.
- Easy mechanical mounting – just clip it onto a DIN rail.
- Easy termination configuration via jumper (two possible settings: free topology wiring or daisy chain wiring) which is accessible from outside the housing – no re-wiring if the termination is changed.
- Easy wiring – not necessary to check polarity or color coding.
- The device has input / output terminals for the LONWORKS connection as well as a removable plug for the LONWORKS tool.

PANEL BUS

The EAGLE Controller features two RS485 interfaces to which Panel Bus modules can be connected: RS485-1 (consisting of push-in terminals 24 [GND-1], 25, and 26) and/or RS485-2 (consisting of push-in terminals 29, 30, and 31 [GND-2]).

NOTE: GND-2 is internally connected with 24V-0 (terminal 1) and system GND (terminals 19+37)

Overview of Panel Bus I/O Modules

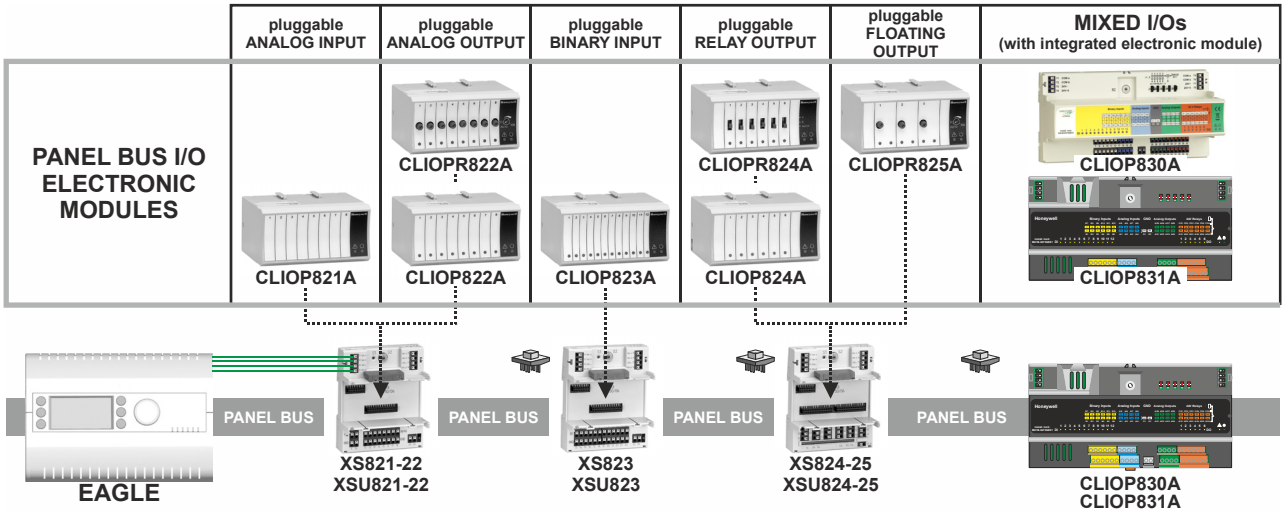


Fig. 81. Overview of Panel Bus I/O Modules

Panel Bus Addressing and Number of Modules

1) During engineering, the HEX address of the Panel I/O modules is defined.
2) It is essential that the HEX switch be set to the address assigned by the engineering tool.
3) Per module type (e.g., XF821A, XF822A, etc.) the HEX switch addressing needs to be unique.

NOTE: The EAGLE controller automatically commissions all Panel Bus I/O modules.

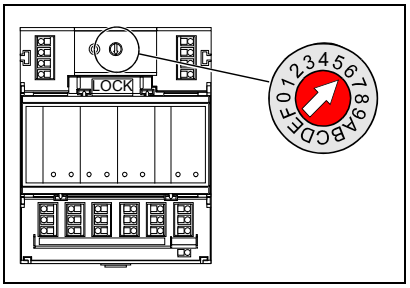


Fig. 82. Address Hex switch of a Panel Bus module

4) Max. number of Panel Bus I/O modules per RS485 interface: 64
   Max. number of Panel Bus I/O modules of a given module type: 16
5) Max. number of Panel Bus I/O modules per EAGLE: 128 (64 for models CLEA2014BxB)
   Max. number of Panel Bus I/O modules of a given module type: 32 (16 for models CLEA2014BxB)
6) Max. number of hardware I/O points including NVs per EAGLE: 600 (52 for models CLEA202014BxB)

Power Supply Cable Specifications

When checking the length of the power supply cable, the connection cables to all Panel Bus I/O Modules must be taken into account.

- Max. length: 3 m (from transformer to final module)
- Cross section: min. 0.75 mm2 (AWG 18)

Panel Bus on RS485-1 (isolated interface)

Cabling Topology, Wiring, and Termination

The Panel Bus is polarity-insensitive (except for loop topology, in which case it is polarity-sensitive).

Any Type of Cabling and Topology (including star and loop topology)

- Max. Panel Bus length: 40 meters
- No additional end termination permitted.

Twisted-Pair or Telephone Cable and Daisy Chain Topology

- Max. Panel Bus length: 1200 meters (9.6 – 76.8 kbps) or 1000 meters (115.2 kbps).
- Mandatory: The EAGLE must be positioned at one end of the Panel Bus, and an end termination (120 Ω) at the other end. Further, the three-position slide switch must be set to "END."

Example 1: Connecting RS485-1 to Panel Bus (single transformer)

NOTE: When connecting RS485-1 of the EAGLE with Panel Bus I/O modules, it is recommended that the slide switch be set to "END."

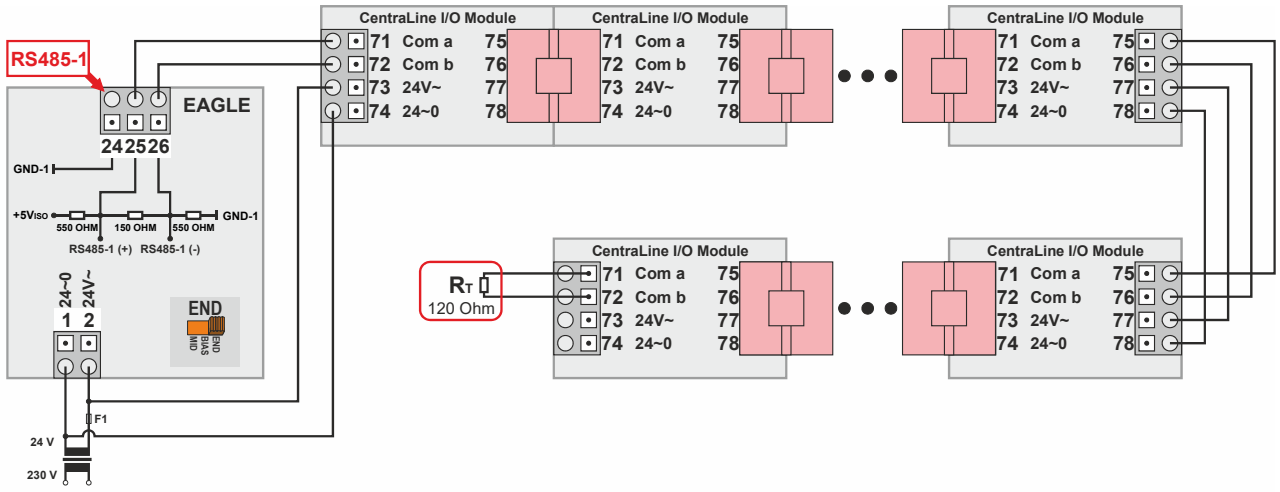


Fig. 83. Connecting RS485-1 of the EAGLE with Panel Bus I/O Modules (single transformer)

Example 2: Connecting RS485-1 to Panel Bus (two transformers)

NOTE: If two transformers are used, it is required that the 24V-0 terminal of the transformer powering the I/O modules be connected to terminal 24 of the EAGLE.

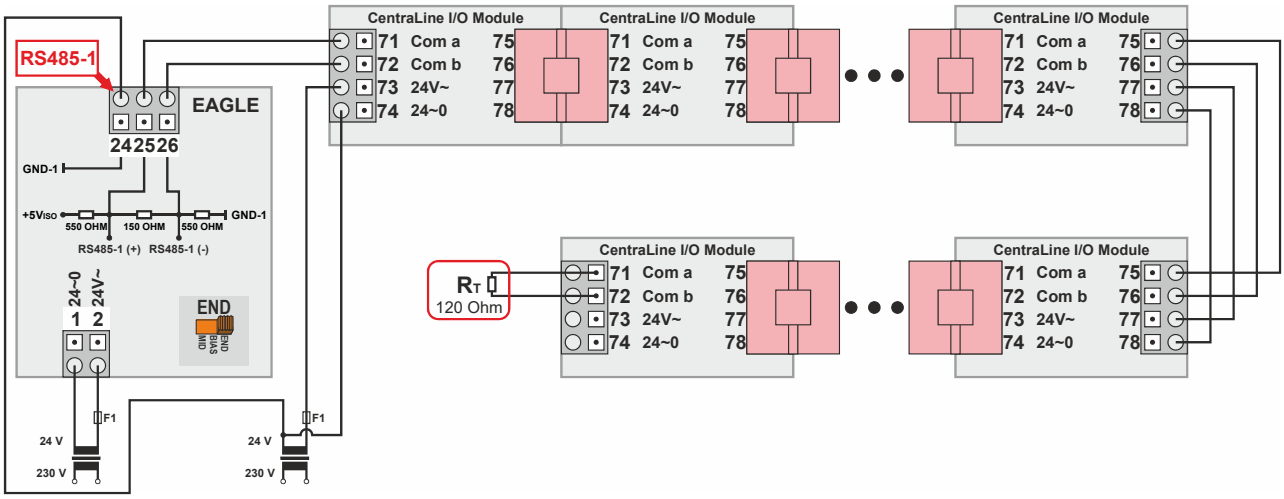


Fig. 84. Connecting RS485-1 of the EAGLE with Panel Bus I/O Modules (two transformers)

Panel Bus on RS485-2 (non-isolated interface)

NOTE: GND-2 is internally connected with 24V-0 (terminal 1) and system GND (terminals 19+37)

Panel Bus Cabling Topology, Wiring, and Termination

The Panel Bus is polarity-insensitive (except for loop topology, in which case it is polarity-sensitive).

Any type of cabling and topology (including star and loop topology)

- Max. Panel Bus length: 40 meters
- No additional end termination permitted.

Twisted-pair or telephone cable and daisy chain topology

- Max. Panel Bus length: 1200 meters (9.6 – 76.8 kbps) or 1000 meters (115.2 kbps).
- Mandatory: The EAGLE must be positioned at one end of the Panel Bus, and an end termination (120 Ω) at the other end.

NOTE: The Panel Bus must not extend beyond a single building or building floor

Example 3: Connecting RS485-2 to Panel Bus (single transformer)

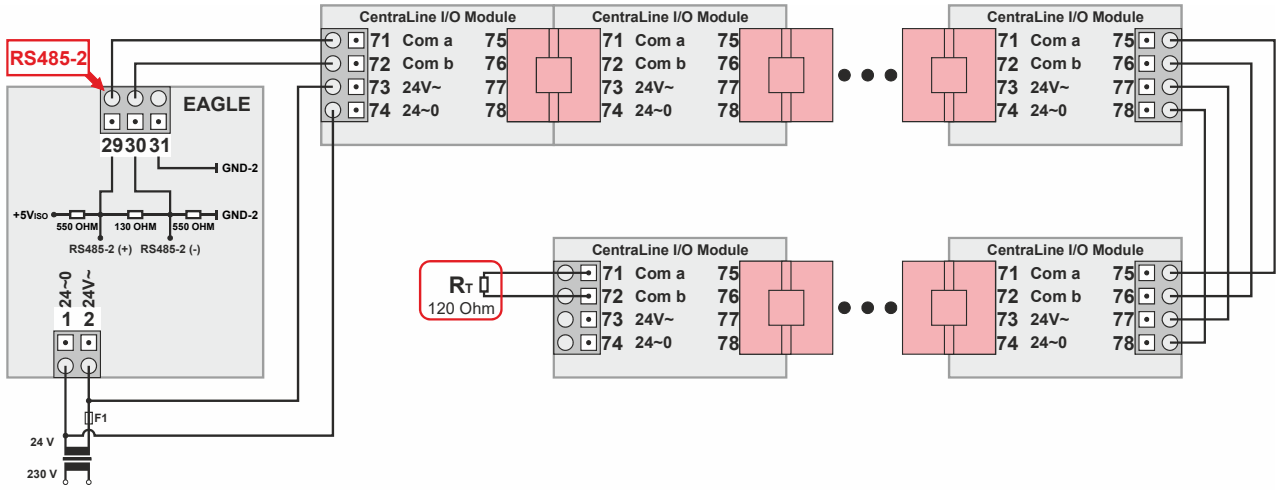


Fig. 85. Connecting RS485-2 of the EAGLE with Panel Bus I/O Modules (single transformer)

Example 4: Connecting RS485-2 to Panel Bus (two transformers)

NOTE: If two transformers are used, it is required that the 24V-0 terminal of the transformer powering the I/O modules be connected to terminal 31 of the EAGLE.

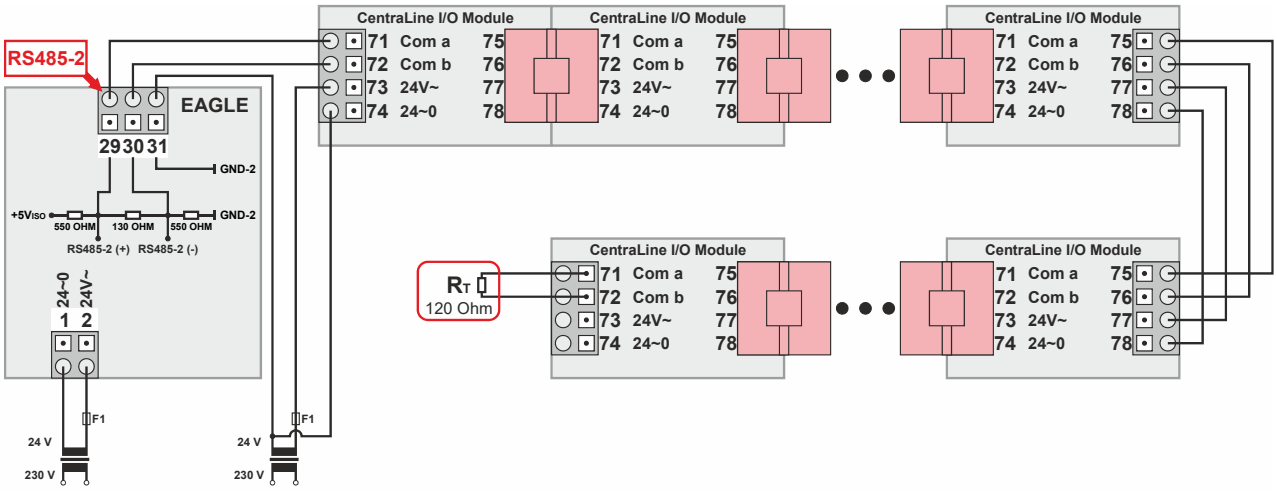


Fig. 86. Connecting RS485-2 of the EAGLE with Panel Bus I/O Modules (two transformers)

Debugging the Panel Bus

There is no tool to check for physical wiring issues. Please check the wiring before carrying out any software analysis by using the Diagnostic web page of the Panel Bus:

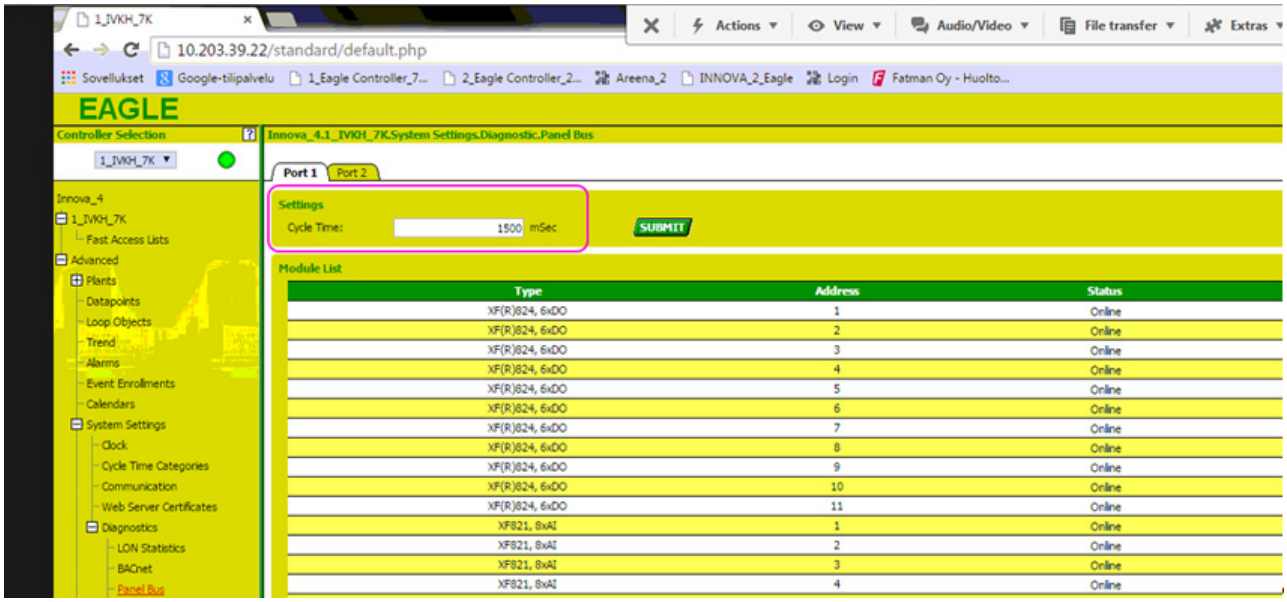


Fig. 87. Diagnostics web page for Panel Bus

On large Panel Bus systems (i.e., systems containing more than 16 devices), it is required that you check the Cycle Time Categories. If they are jumping, it is required that you set the Panel Bus cycle time. Both times should have a similar value.

Example: If the control loop cycle time is between 5 and 30 seconds (average 10 seconds), is it not required to scan the Panel Bus modules every 750 msec. In this case, the Panel Bus scan time can be reduced to 3 seconds. The control loop cycle time will then decrease. Please make sure the RACL cycle time is shorter than the Panel Bus scan time.

Panel Bus Cycle Time

For inputs and outputs, there are two mechanisms to write values:

- Standard cycle is sending/receiving only changed values.
- Heartbeat cycle is updating all values of a module.

Standard Execution (controller firmware 3.04.02 and higher)

- The standard cycle is executed within the base cycle time of 250 msec.
- This applies for up to 16 modules.
- If 16 < # modules ≤ 32, the cycle time is multiplied by 2 = 500 msec.
- If 32 < # modules ≤ 48, the cycle time is multiplied by 3 = 750 msec.
- If 48 < # modules ≤ 64, the cycle time is multiplied by 4 = 1000 msec.
- CLIOP830 modules each count as 1 module.

Standard Execution (controller firmware 3.04.01 and lower)

- The I/O cycle time configurable in the web interface is introduced.
- This needs be set after each CARE download, CARE 10.06 or higher will support this setting.
- This applies for maximal 16 modules (default = 250 msec = Min, Max = 10 seconds).
- If 16 < # modules ≤ 32, the cycle time (default = 500 msec = Min, Max = 10 seconds).
- If 32 < # modules ≤ 48, the cycle time (default = 750 msec = Min, Max = 10 seconds).
- If 48 < # modules ≤ 64, the cycle time (default = 1000 msec = Min, Max = 10 seconds).
- CLIOP830 modules each count as 4 modules.

Adjustable Panel Bus Scan Cycle Time (CARE software 10.06.00 and higher)

With CARE software 10.06.00 and higher, it is possible to adjust the Panel Bus scan cycle time not only in the SVGA web pages or using XW-Online, but also now with CARE. The shortest possible scan cycle time is 60 msec. See screenshot below.

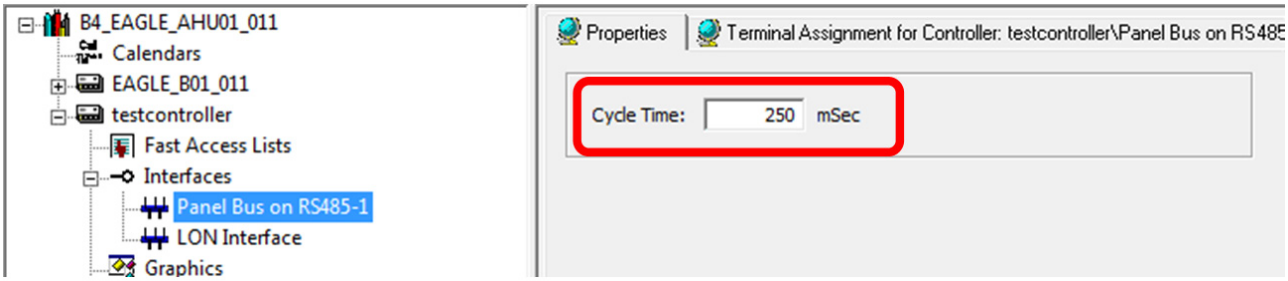


Fig. 88. Adjusting the Panel Bus cycle time in CARE

"Send on Delta" for Analog Inputs (controller firmware 3.04.02. and higher)

Beginning with this controller firmware, the following fixed SendOnDelta values are introduced in order to reduce the amount of value updates sent by the Panel Bus. This will avoid excessively high cycle times for high point count applications (> 500 I/Os), and it will avoid fault alarms for analog inputs.

- 0.5% for AI types: 0…10V / 0...20mA
  0…10V / 0…20mA with pull-up
  2...10V / 4...20mA

- 0.1K for AI types: NTC20K, NTC 10K
  PT1000_1, PT1000_2
  PT3000
  NI1000_TK500009
  BALCO

Updating Firmware of Panel Bus I/O Modules (controller firmware 3.04.03 and higher)

Beginning with controller firmware 3.04.03, the firmware of all Panel Bus modules is included in the controller firmware. As soon as the controller detects the presence of Panel Bus modules, it will check the currency of the modules' firmware and update their firmware, as necessary.

Manufactured for and on behalf of the Connected Building Division of Honeywell Products and Solutions SARL, Z.A. La Pièce, 16, 1180 Rolle, Switzerland by its Authorized Representative:

CentraLine
Honeywell GmbH
Böblinger Strasse 17
71101 Schönaich, Germany
Phone +49 (0) 7031 637 845
Fax +49 (0) 7031 637 740
info@centraline.com
www.centraline.com

Subject to change without notice
EN2Z-1002GE51 R1218

CENTRA LINE
by Honeywell