

Universal Gauge Emulator

UGE-207 User Manual / Installation Guide

LPG, Additive and Process tank application



Table of Contents

1	Preface.....	5
1.1	Document conventions.....	5
1.2	Preface.....	5
1.3	Legal aspects.....	5
1.4	EU Declaration of Conformity.....	5
1.5	Additional information.....	5
2	Introduction.....	6
2.1	What is the UGE-207?.....	6
2.1.1	Limitations	6
3	Connecting the UGE-207.....	7
3.1	Power supply.....	7
3.2	BPM bus.....	7
3.3	RS232/RS485 serial bus.....	8
3.4	Interconnect.....	8
4	Functionality.....	10
4.1	UGE-207 internals detailed.....	10
4.2	License model.....	11
4.3	UGE-207 Device and Emulated Gauge GPU addresses.....	12
4.4	GPU Gauge Emulation on the BPM host port.....	12
4.5	Modbus RTU Master on the RS field port.....	14
4.6	Command Line Interpreter.....	15
4.7	Message monitoring on the buses.....	15
4.8	Remote Management.....	16
4.9	Measuring BPM signal and noise.....	16
4.10	LED indicators.....	17
5	Configuration.....	19
5.1	Front Panel and Switches.....	19
5.2	TERMINAL mode.....	20
5.3	Configuring the UGE-207.....	22
5.3.1	Configuring in TERMINAL mode	22
5.3.2	Configuring via Device Address	23
5.4	Common UGE-207 Item list.....	25
5.5	Modbus Variable Related Item list.....	29
5.5.1	Modbus Variable Register Definition	30
5.5.2	Range Conversion Definition	32
5.5.3	Mobus Status Register Definition	33
5.5.4	Variable Invalid Status Definition	33
5.6	Gauge Related Item list.....	35

6 Commissioning and Examples.....	38
6.1 Entering RS port to TERMINAL Mode.....	38
6.2 Example: Emulation for four tanks with 4-20 mA level gauges.....	41
6.2.1 Pre conditions	41
6.2.2 Settings for UGE-207 Device	42
6.2.3 Settings for UGE-207 Data Points	42
6.2.4 Settings for UGE-207 Emulated Gauges	43
6.2.5 Overview of configuration and test	44
6.3 Example: Emulation for tank with 4-20 mA level, temperature, pressure.....	48
6.3.1 Pre conditions	48
6.3.2 Settings for UGE-207 Device	49
6.3.3 Settings for UGE-207 Data Points	49
6.3.4 Settings for UGE-207 Emulated Gauges	49
6.3.5 Overview of configuration and test	49
7 Diagnosing BPM fieldbus.....	53
8 Updating Firmware.....	56
8.1 On Windows.....	56
8.2 On Linux or MacOS.....	57
9 Appendix: Firmware release changelog.....	58

Index of Tables

Table 01: RS232/RS485 connection.....	8
Table 02: UGE-207 license model.....	11
Table 03: Supported Records by UGE-207-Connect Gauge Emulator.....	12
Table 04: Supported items for 'Z' record by UGE-207-Connect Gauge Emulator.....	12
Table 05: Item error messages.....	14
Table 06: LEDS.....	17
Table 07: BPM and RS LEDS.....	18
Table 08: Front panel.....	19
Table 09: DIP-switches.....	19
Table 10: Modbus Variable related Item list.....	29
Table 11: Modbus Variable Register Definition fields.....	30
Table 12: Variable Status Definition type.....	34
Table 13: Gauge related Item list.....	35

Table of Figures

Figure 01: Top connectors.....	7
Figure 02: Bottom connectors.....	8
Figure 03: Back plane PCB connector.....	9
Figure 04: UGE-207 with DIN rail back plane connector.....	9
Figure 05: UGE-207 DIN rail bus connector.....	9
Figure 06: UGE-207-Connect detailed functionality.....	11

Figure 07: Measuring BPM fieldbus signal strength.....17

Figure 08: Front plate.....19

Figure 09: Range conversion.....32

Figure 10: Commissioning of UGE-207.....38

Figure 11: Emulation for 4-20 mA level gauges.....41

Figure 12: Emulation for 4-20 mA level, temperature and pressure transmitters.....48

Figure 13: Measuring BPM fieldbus signal strength.....53

1 Preface

1.1 Document conventions

Warnings, Cautions and **Notes** are used throughout this installation guide to bring special matters to the immediate attention of the reader.

- A Warning concerns danger to the safety of the technician or user.
- A Caution draws the attention to an action which may damage the equipment.
- A Note points out a statement deserving more emphasis than the general text.

1.2 Preface

This installation guide is intended for technicians involved in the mechanical and electrical installation of the Exalon Delft UGE-207. The technician must have basic technical skills and knowledge of safety regulations.

1.3 Legal aspects

The information in this installation guide is the copyright property of Exalon Delft B.V., Netherlands. Exalon Delft B.V. disclaims any responsibility for personal injury or damage to equipment caused but not limited by:

- Deviation from any of the prescribed procedures.
- Execution of activities that are not prescribed.
- Neglect of the general safety precautions for handling tools, use of electricity and microwave radiation.

The contents, descriptions and specifications are subject to change without notice. Exalon Delft B.V. accepts no responsibility for any errors that may appear in this installation guide.

1.4 EU Declaration of Conformity

This Exalon Delft instrument is in conformity with all applicable EC Council Directives, including the EMC Directive 2014/30/EU. Refer to the EC Declaration of Conformity supplied with each instrument separately.

1.5 Additional information

Please do not hesitate to contact Exalon Delft or its representative if you require additional information.

2 Introduction

2.1 What is the UGE-207?

The UGE-207 is a multi purpose and versatile Gauge Emulator which enables seamlessly connection of (third party) instruments to the robust Honeywell-Enraf¹ BPM fieldbus. Integration of the instrument data into the Honeywell-Enraf CIU Tank Gateway is greatly simplified by emulating the Honeywell-Enraf GPU protocol.

The UGE-207-Connect version is specifically designed for LPG tank, additive tank and process tank applications. These applications are characterized by applying fit for purpose less accurate instruments and limited amount of data. The UGE-207-Connect application supports relevant tank data for radar level gauge, single spot average product and vapor temperature, vapor pressure and density transmitters.

The instrumentation data is retrieved by the Modbus RTU Master from one or multiple Modbus RTU slave devices. This also allows integration of 4-20 mA analog input Modbus RTU modules into the solution, enabling connection of instruments with a 4-20 mA output.

2.1.1 Limitations

The UGE-207-Connect version supports:

- One (1) BPM Host Port
- A maximum of six (6) Emulated Gauges on the BPM Field Port
- One (1) RS232/485 Modbus RTU Master Field Port
- A maximum of 20 Data Points mapped onto the Emulated Gauge variables

1 Honeywell-Enraf, Enraf, Honeywell are registered trademarks and trademarks of these organizations and companies. All other marks are property of their respective owners.

3 Connecting the UGE-207

The UGE-207 is a DIN-rail mountable product compatible with EN-IEC60715, mounting rail 35x7,5 mm. An UGE-207 can be interconnected by DIN rail bus connectors with a UCI-108 Universal Communication Interface allowing for the addition of a local service port.

The UGE-207 has three terminal blocks and one connector for connecting:

- Power supply (on top)
- BPM bus (on top)
- RS232/RS485 bus (on bottom)
- Interconnect by DIN rail bus (at the back side)

3.1 Power supply

Each UGE-207 has to be individually powered by means of the power terminals on top marked with 24V and GND, see Figure 01. The power terminal block provides double terminals for easy daisy-chaining power to the next module in a multi UGE-207 node. The power supply must comply with 24 Vdc +/- 5% and minimal 80 mA per module.

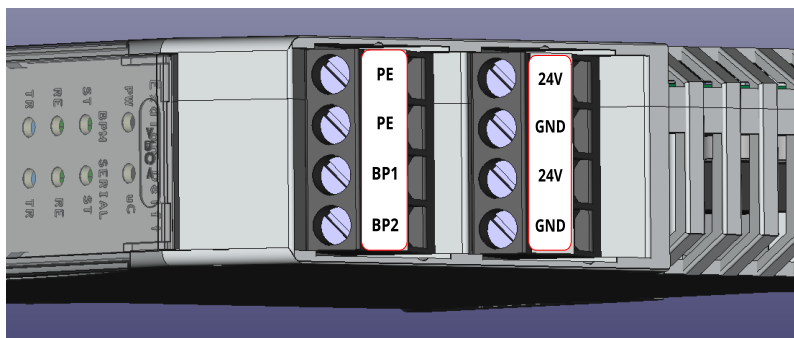


Figure 01: Top connectors

3.2 BPM bus

The BPM bus wires should be connected to BP1 & BP2, see Figure 01. The PE marked terminals are for Protective Earth.

The BPM cable shield has to be connected at one location to the Protective Earth.

3.3 RS232/RS485 serial bus

The RS serial bus should be connected to the terminals with labels A/RX, B/TX and GND terminals. See Figure 02 and Table 01 below.

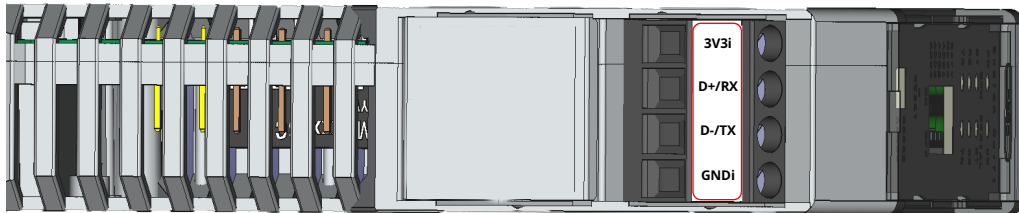


Figure 02: Bottom connectors

Table 01: RS232/RS485 connection

UGE-207 RS232/RS485	RS232 DTE DB9 male	RS485
3V3i		
D+/RX	TX (pin 3)	DATA+
D-/TX	RX (pin 2)	DATA-
GNDi	GND (pin 5)	GND

Note: RS232 TX is connected to the UGE-207 RX, RS232 RX is connected to UGE-207 TX.

The **3V3i** and **GNDi** is available as an isolated voltage to an optional RS485 external biasing and termination resistor network.

The RS switch position on the front side determines the function of the RS port, RS232 or RS485. For communication settings see Paragraph 5.2.

3.4 Interconnect

The UGE-207 has a PCB bus connector for optional use with an UCI-108 Universal Communication Interface. The UGE-207 is placed in one of the assembled DIN rail bus connectors mounted on the DIN rail forming the local communication bus.

Note: the optional DIN rail bus connector(s) has to be ordered separately.

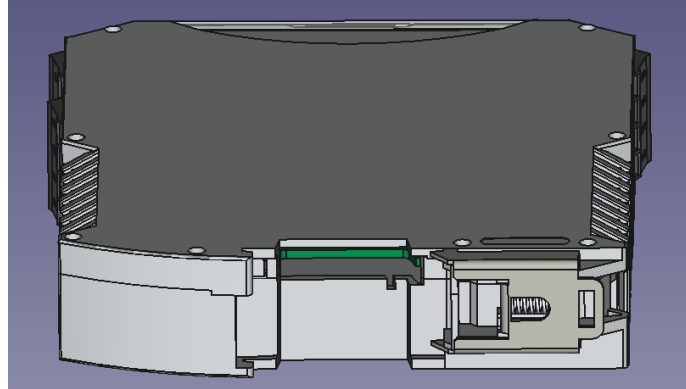


Figure 03: Back plane PCB connector

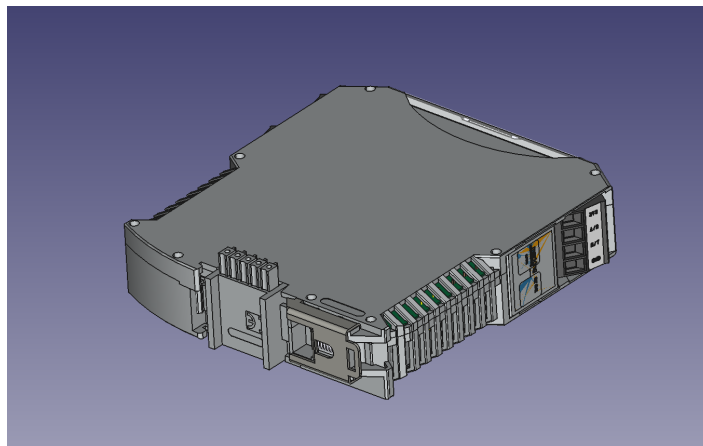


Figure 04: UGE-207 with DIN rail back plane connector

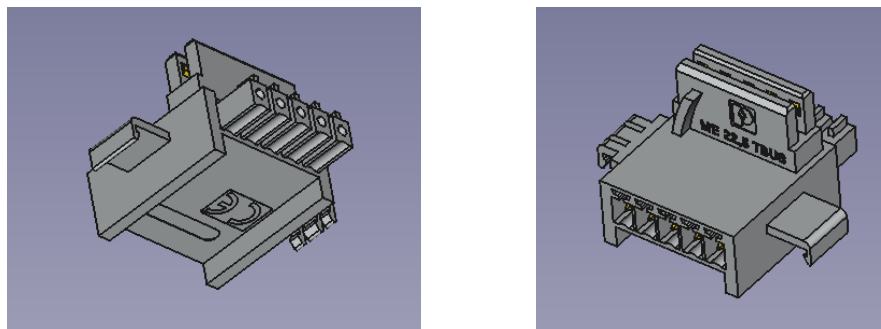


Figure 05: UGE-207 DIN rail bus connector

4 Functionality

The UGE-207-Connect version is a basic Gauge Emulator. It presents itself as a Honeywell-Enraf 973 SmartRadar on the BPM field bus to the host application for all relevant LPG, Additive and Process tank application data. The UGE-207-Connect collects the data from (third party) level-, temperature-, pressure and/or density instruments connected to its Modbus RTU field port.

The UGE-207-Connect version is equipped with an internal command line interpreter for commissioning and BPM communication monitoring purposes, operated from a PC terminal emulator. The command line interpreter shares the same RS serial port as used for the Modbus RTU Master during operation. Only one of these functions can be assigned to this RS port at the same time.

In addition, an UCI-108 Universal Communication Interface can be interconnected to form a single node. This (temporary) enables an additional RS port with command line interpreter for commissioning the UGE-207 and BPM communication monitoring and diagnose purposes.

4.1 UGE-207 internals detailed

This UGE-207-Connect version implements multiple Emulated Gauges in one single UGE-207 device. Each of the Emulated Gauges supports any of these 6 variables:

1. Product Level (innage)
2. Product Temperature
3. Vapour Temperature
4. Product Pressure (P1)
5. Vapour Pressure (P3)
6. Observed Density

The data for these Emulated Gauge variables is acquired by scanning the configured Modbus registers of connected Modbus RTU modules or transmitters. The Modbus registers can contain any numeric data representing process, 4-20 mA, ADC- or other values. The queried Modbus register values are characterized to a Data Point. The processed Data Points are stored in a realtime database and linked to the Emulated Gauges variables (see figure 06 'Figure 06: UGE-207-Connect detailed functionality').

The product level variable is provided with alarm limits and hysteresis, which are configurable per Emulated Gauge.

The number of supported Emulated Gauges and available characterized Data Points depends on the purchased UGE-207 license.

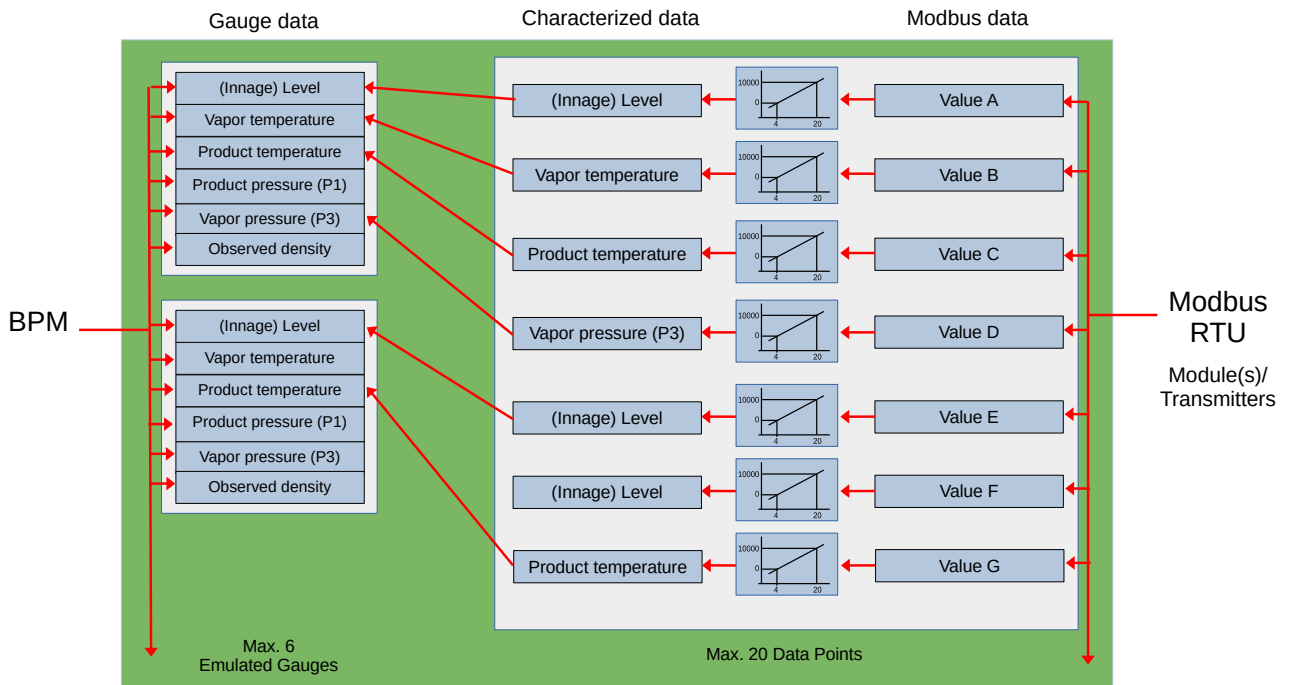


Figure 06: UGE-207-Connect detailed functionality

4.2 License model

The UGE-207 Gauge Emulator is available in three license variants: Base, Plus and Premium. Depending on the purchased UGE-207 license more Emulated Gauges and instrument Data Points become available. The license is determined at the time of order time and cannot be upgraded.

Table 02: UGE-207 license model

	UGE-207 license		
	Base	Plus	Premium
Max. number of configured Emulated Gauges	4	5	6
Max. number of configured Variables	4	12	20

4.3 UGE-207 Device and Emulated Gauge GPU addresses

The UGE-207-Connect supports up to six individual Emulated Gauge GPU addresses, depending on the license. The GPU address of each individual Emulated Gauge (nn) will respond to:

- nnBA, nnBB, nnBC, nnBD, nnBL for other record requests
- nn@X or nnBX for X-record requests
- nnBZ for Honeywell Gauge item requests

Due to optional support of the UCI-108 Modbus tunnel feature in the system the GPU address 00 cannot be used by one of the Emulated Gauges or in the BPM network.

To save GPU addresses the UGE-207 device is accessible for remote configuration via one of the Emulated Gauge GPU addresses (default set to 98), in-dependable wether the Emulated Gauges are enabled or disabled. The UGE-207 device will respond to:

- nnRX for X-record requests
- nnRZ for item requests

4.4 GPU Gauge Emulation on the BPM host port

The UGE-207-Connect Emulated Gauges responses to GPU messages compatible with the Honeywell-Enraf 973 radar gauge for the variables described in paragraph 4.1 'UGE-207 internals detailed'.

Table 03: Supported Records by UGE-207-Connect Gauge Emulator

Type	Name
A	Alarm Status
B	Alarm Status and Level
C	Alarm Status and Temperature
D	Alarm Status, Level and Temperature
L	Alarm Status and Signed Level
X	Identification
Z	Items

Table 04: Supported items for 'Z' record by UGE-207-Connect Gauge Emulator

Item	Description	Remark
------	-------------	--------

AG	Average vapour temperature	process data
AH	Alarm Hysteresis	configuration
AP	Average product temperature	process data
DI	Density dimension	configuration
DP	Decimal separator	configuration, fixed to '.'
DQ	Hydrostatic Density	process data
HA	High Alarm Level	configuration
HH	High High Alarm Level	configuration
LD	Level dimension	configuration
LA	Low Alarm Level	configuration
LL	Low Low Alarm Level	configuration
PN	Production number	configuration
PS	Protection status	configuration, fixed to '000'
O1	Pressure offset P1	configuration, fixed to '0.0'
O3	Pressure offset P3	configuration, fixed to '0.0'
P1	Pressure sensor P1	process data
P3	Pressure sensor P3	process data
PA	Available pressure transmitters	configuration
PI	Pressure dimension	configuration
QR	Radar Status request	process data
RI	Radar innage level	process data
QQ	Abbreviated Hydrostatic Density	process data
QF	Hydrostatic Status request	process data
SV	Software versions	configuration
TA	Transmission address	configuration
TD	Temperature dimension	configuration
TI	Tank Identifier	configuration
W1	Password 1	configuration, fixed
W2	Password 2	configuration, fixed

All Emulated Gauge items requests are read-only. See next table for possible item error reply messages.

Table 05: Item error messages

Reply	Description	Comment
!051	Unknown item	Item is not supported by Emulated Gauge
!055	No write access	Items to configure the UGE-207 device are performed via requests on one of the active individual Emulated Gauge addresses, see 5.6 Gauge Related Item list.

Note: Level related data is of type 'Innage'.

Note: Only records and items variables with coupled data points are available.

4.5 Modbus RTU Master on the RS field port

The Emulated Gauges variables are linked to Data Points. The data used for the Data Points is read from Modbus RTU slave devices. The following items in the UGE-207 device are available to configure and convert a Modbus register variable to a Data Point, where xx is the index of the Data Point ($xx=1..20$):

- **MVxx** Modbus Variable Register Definition.
Defines the Modbus slave address, Register type, Register number, Data type and Endian type to fetch the value which represents the instrument measurement.
- **MRxx** Modbus Variable Range Conversion Definition
Defines the conversion from the Modbus Variable value to a Data Point value by applying scale and offset and the unit.
- **MSxx** Modbus Status Register Definition
Optionally defines the Modbus slave address, Register type, Register number, Data type and Endian type to fetch the value which represents the status of the Modbus Variable.
- **MIxx** Modbus Variable Invalid Status Definition
Defines the conditions to indicate the invalid status of the Modbus Variable Register. The invalid status can be based on the register value defined by Modbus Variable Register or on the Modbus Status Register defined by Msxx.

Note: The Modbus Variable will also become invalid in case of 3 successive Modbus timeouts or Modbus error replies for fetching the Modbus Variable Register.

The condition is also stipulated that the instrumentation providing the Modbus Variables must already be correctly configured and commissioned on range and unit for the variable. Radar level gauges has to be commissioned for 'Innage' measurement.

4.6 Command Line Interpreter

The UGE-207 features a command-line interpreter for commissioning and communication monitoring purposes, operated from a PC terminal emulator. The command-line interpreter shares its RS232 communication port with the RS Field port of the Modbus RTU Master. Only one of these functions can be assigned to this RS port at the same time.

The function of the RS Field port is set by a DIP-switch and the RS switch on the front panel. TERMINAL for the command-line interpreter via the RS232 port or OPERATIONAL mode for the Modbus RTU Master via the RS232 or RS485 port.

Tip: reserve a spare DIN rail backplane position for a temporarily UCI-108 used for commissioning and service purposes. Make this UCI-108 node fixed as the Fourth node making it work in every UCI/UGE multi node.

4.7 Message monitoring on the buses

The command line interpreter (see Paragraph 5.2) also supports monitoring (listen mode) for both messages on the BPM host port or the Modbus RTU device replies on the RS field port. This will be initiated by the command: `msgmon`.

The logged message will be labeled with:

- GPU, FLX, MDB to indicate the protocol type of the message on the BPM host port.
- MBR to indicate the Modbus reply on the RS field port.
- [B], [B1], [B2], [B3], [B4], [R], [R1], [R2], [R3] or [R4] indicating on which port the message has been received. 'B' for BPM port and 'R' for RS port. The [G], [G1], [G2], [G3] or [G4] indicates the emulated gauge reply on the BPM port. The number after the letter indicates the UGE node address if applicable.

- Q and A indicating a Request or Answer message.

After the response data follows the timestamp (labeled with ts) and for BPM messages the BPM signal quality of the message (labeled with qa). The signal quality are two voltage values (in mV) measured.

Example:

```
MBR[R1]A 09 03 02 0F 8E DC 11    ts: 5897296
GPU[B1]Q 98RZSV    ts: 5897261    qa: 3038/3038
GPU[G1]A 98RZSVUGE-207 1.0.0 Premium CONNECT UGE_0001    ts: 5897328
MBR[R1]A 09 03 02 1F A9 91 CB    ts: 5897322
```

To ease importing the data in a spreadsheet the fields are separated using a TAB character².

4.8 Remote Management

Remote management for configuration or status monitoring of a UGE-207 Device is accomplished using proprietary GPU commands via one of the configured Emulated Gauge GPU addresses. See chapter 5.4, 5.5 and 5.6 for the detailed GPU command items.

4.9 Measuring BPM signal and noise

The BPM signal strength and noise indication feature can be used to determine the signal quality of the BPM fieldbus. The quality of the signal is effected by different factors like cable wire resistance, number of connected field instruments, location of connected field instruments, injected noise.

With this feature possible BPM communication issues can be isolated. It can be used to determine the location for placing a repeater to extend or improve the BPM fieldbus.

The UGE-207 BPM receiver circuit contains a Programmable Gain Amplifier (configured using item BG, see chapter 5.4 Common UGE-207 Item list). The signal strength of the received BPM messages before and after this PGA is measured. Different items are available to obtain information for analysis.

The example in 'Figure 07: Measuring BPM fieldbus signal strength', shows a setup to determine the BPM signal and noise on location B, e.g. the location of expected communication issues. One UCI-108 is temporary added at location A as command

² Note that some terminal programs replace a TAB character with spaces. For example TeraTerm does this but using its 'log to file' option the TAB characters remain preserved.

terminal in a multi node configuration and one UGE-207 in the field at location B. Different scenario's can be executed to locate the issue, see Chapter 7 Diagnosing BPM fieldbus.

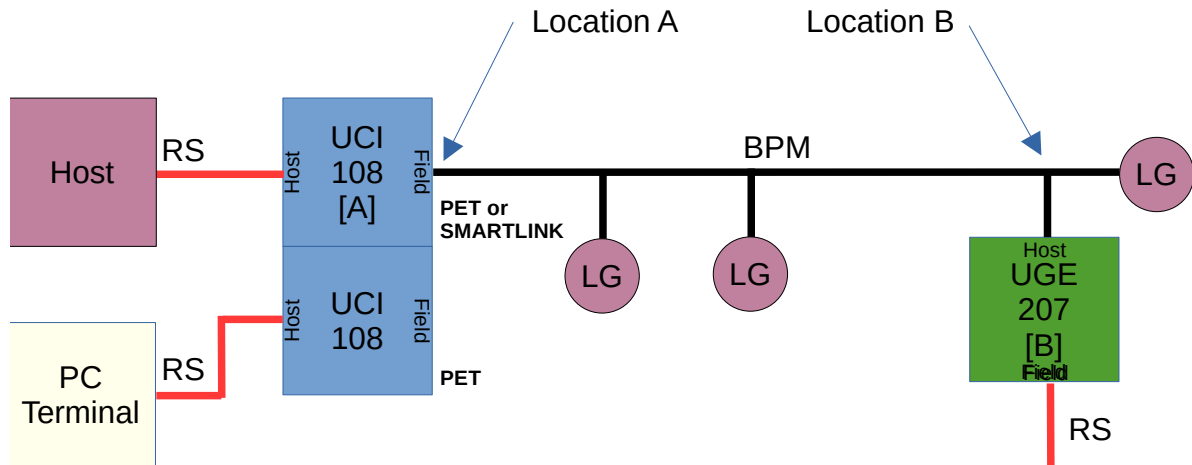


Figure 07: Measuring BPM fieldbus signal strength

4.10 LED indicators

The UGE-207 has 8 LEDs on its front.

Table 06: LEDS

Name	Description
PW	When the device is powered this LED will be on.
EXE	When the firmware is running this LED will blink.
BPM	These LEDS will flash when traffic is active on the BPM bus.
RS	These LEDS will flash when traffic is active on the serial bus.

EXE LED

During normal operation the EXE LED will blink with a rate of: 1S on / 1S off.

- When the UGE-207 is running in the bootloader the rate will be:
- When the bootloader detected an application in memory the LED will blink one short pulse.

- When the bootloader did **not** detect an application in memory the LED will blink two short pulses.

Table 07: BPM and RS LEDs

Name	Description
ERR	Will flash whenever an error occurs with the reception of a message on the bus.
RX	Host port: on unless receiving Field port: off unless receiving
TX	Host port: off unless sending Field port: on unless sending

When the RS port is set to RS232 and TERMINAL Mode the RS LEDs will flash when terminal input is received or transmitted.

5 Configuration

Configuring the UGE-207 is done using the switches on the outside of the enclosure and by item commands entered via a PC terminal program connected to the RS232 port.

5.1 Front Panel and Switches

Table 08: Front panel


 <p>UGE-207</p> <p>PW ● ● EXE</p> <p>ER ● RS ● ER</p> <p>RX ● RX</p> <p>TX ● TX</p> <p>TERM MISC AD-1 AD-2 AD-3 AD-4</p> <p>ON OFF</p> <p>● RESET</p> <p>RS232 RS485</p> <p>Exalon Delft</p>	<p>For the significance of the LEDs see Table 06: LEDS.</p> <p>For the DIP-switch positions see Table 09: DIP-switches.</p> <p>The Reset button will make the device restart.</p> <p>The RS switch needs to be set in the position corresponding the type connection in use: RS232 or RS485.</p>
---	--

Figure 08: Front plate

The hardware switches allow to configure:

Table 09: DIP-switches

Name	Label	Description
RS switch	RS232 RS485	Set the RS front port in 'RS232' or 'RS485' mode.

DIP switch 1	TERM	Set the UGE-207 in 'TERMINAL' mode (ON) or 'OPERATIONAL' mode (OFF).
DIP switch 2	MISC	Reserved
DIP switch 3..6		UGE-207 node address. When additional connecting an UCI-108 forming an UGE/UCI node using the DIN rail bus connector, use this switch to configure their individual node addresses.
	AD-1	Switch 3 ON: First Node
	AD-2	Switch 4 ON: Second Node
	AD-3	Switch 5 ON: Third Node
	AD-4	Switch 6 ON: Fourth Node

Note: detection of the DIP-switch positions and activation of related function is performed only after a power cycle or reset of the UGE-207.

5.2 TERMINAL mode

The UGE-207 features a command-line interpreter for commissioning and communication monitoring purposes, operated from a PC terminal emulator. The command-line interpreter shares its communication port with the Modbus RTU Master RS Field port.

A recommended alternative is to create a UGE-207/UCI-108 node by interconnecting both devices via DIN-rail backplane connectors. The UCI-108 enables a (temporary) additional RS port for the UGE-207 command line interpreter having the UGE-207 RS Field port available to connect to the Modbus RTU devices. After configuration the UCI-108 can be removed.

*Note: Set the UGE-207 DIP switch AD1 to ON to assign node address to 1 and the UCI-108 DIP switch AD4 to ON to assign node address to 4. Reset or power cycle both UGE and UCI.
Type at the UCI-108 command prompt the command 'cu 1' to connect to the UGE-207 command-line interpreter each time after reset or restart of UGE-207.*

In order to use the command-line interpreter the connected RS port (on the UGE-207 or UCI-108) to the PC terminal emulator has to be set to RS232 and assigned for TERMINAL mode. Remove the power, connect a PC terminal emulator to the RS port,

set the RS switch into it's RS232 position, set the 'TERM' DIP-switch into the 'ON' position and set power on.

The communication parameters for the PC terminal emulator are: 115200 baud, 8 data bits, 1 stop bit, no parity, no flow control.

The UGE-207 built-in command-line interpreter sends text lines ending with a 'newline character'.³

The command-line interpreter supports the following commands:

```

help      : print list of all available commands or items
item      : get/set configuration using Items
msgmon    : monitor messages (GPU, Flexconn, Modbus)
data      : show current data
mbr       : modbus read
mbw       : modbus write
stats     : show/reset statistics
config    : show/reset the current configuration
reset     : reset the UGE
cu        : change UGE/UCI (if multi node is enabled)

```

- The 'help' command prints above list.
- The 'item' command allows to read or write an Item value. See paragraph 5.4 'Common UGE-207 Item list', 5.6 'Gauge Related Item list' and 5.5 'Modbus Variable Related Item list'.
- The 'msgmon' command enables received BPM and Modbus messages to be printed on the terminal. Pressing a key will make this stop. 'Msgmon' shows the BPM and Modbus messages using a hex dump of the complete messages.
- The command 'data' shows the internal data of the Emulated Gauges as well as the Data Points. With this command the processing from Modbus Register inputs to the Emulated Gauges can be verified during commissioning.
 - data gauges : show the internal data for the emulated gauges
 - data datapoints : show the internal data of Modbus input registers and the corresponding Data Point.
- The 'mbr' and 'mbw' command enables the reading and writing of a Modbus register of a connected Modbus device on the RS port.
- The 'stats' command allows to read the current communication and/or system statistics.
 - stats cacheshow : show cache statistics

3 When using the 'Tera Term' application set the terminal configuration New-line options to: 'Receive: LF' and 'Transmit: CR' and uncheck 'Local Echo'.

- `stats commshow` : show communication statistics
- `stats commreset` : reset communication statistics
- `stats sysshow` : show system statistics
- The 'config' command allows to read the current configuration stored in non-volatile memory. The following command parameters are supported:
 - `config setup` : quick configuration by means of menu choices
 - `config show` : shows the items values for the UGE-207 device
 - `config gauges` : shows the items of the Emulated Gauges
 - `config datapoints` : shows the items of the Data Points
 - `config showall` : shows all items values
 - `config items` : generate items replicate of the current configuration
 - `config reset` : will reset the configuration back to the factory defaults

Some commands support the `help` parameter. For example `item help` prints a list of supported UGE-207 configuration items, `mbr help`, `mbw help`, `stats help` and `config help` shows an overview of available options.

To exit TERMINAL mode turn of the power, set the 'TERM' DIP-switch in the OFF position and if needed reconnect the front RS port.

5.3 Configuring the UGE-207

The UGE-207 can be configured by using Items, see paragraph 5.4, 5.5 and 5.6. This configuration can be accomplished in TERMINAL mode via the command 'item' or remotely via the BPM Host port using the UGE-207 GPU Device address with GPU messages requesting read/write **Items**.

5.3.1 Configuring in TERMINAL mode

The current UGE-207 configuration overview by description, item name and item value can be showed by the commands `config show`, `config gauges`, `config datapoints` or `config showall`. Command `config reset` resets the configuration back to the factory defaults.

Configuration via item command

Items are read by the command `item <Item>` where `<Item>` is one of the items from the Item list, see paragraph 5.4, 5.5 and 5.6. To get an overview of available Items type `item help` at the terminal prompt.

For example command `item UT` (the item for UGE-207 device tag) will respond with e.g. `UTUGE_0001`

Items can be written by the command `item <Item>=<Value>` where `<Item>` is one of the items from the Item list and `<Value>` one of the permitted possible values.

For example command `item UT=UGE_LBL` sets the UGE-207 device tag and will respond with `UM=UGE_LBL &`. The `&` is added to indicate a successful write.

Configuration via menu options

The UGE-207 has three group of items to be configured: the device items, the emulated gauge items and the Data Point items. To make this process easier and better structured the command `config setup` is introduced. Via the menu current values of the Items are shown in a list and can be selected by typing the menu option number and can be modified. After each modification the applicable Items are shown again.

This is the recommended method for configuration.

Save and restore the configuration

Once the UGE-207 configuration is finished this can be saved by the command `config items`. All configurable Items are printed to the terminal screen with the prefix `item`. This text can be selected, copied to the clipboard and saved to a plain text file.

The file content can be copied to the clipboard and pasted to the command prompt to restore the configuration.

5.3.2 Configuring via Device Address

Remote configuration via the UGE-207 Device address can be accomplished by GPU requests. The GPU read command messages consists of the elements:

- The two digit UGE-207s GPU Device addresses (default 98) of one of the Emulated Gauges.
- RZ, for the TOI/TOR for Items,
- The two letter Item indication. See paragraph 5.4, 5.5 and 5.6.

For example `98RZSV` results in a reply of `98RZSVUGE-207 1.0.0 Premium CONNECT UGE_0001`

A write command message consist of the same three elements as the read command plus the string `'=<value>'` with `<value>` the new value to set.

The UGE-207 returns with the write command appended with:

- a '&' to indicate a successful write
- a '!' in case of an invalid Item or value for the requested value. Any error message will be appended (in text) after the '!'.

For example writing the BPM gain `98RZBG=3` results in the reply `98RZBG=3&`

To prevent unwanted actions the command items need confirmation data in the form of 'YES'. For example:

```
98RZXR=YES
```

The item request messages can be generated by the Engauge GPU command line tool (Direct Communication).

Note: some Items needs a restart of the UGE-207 via the GPU request `<nn>RZUS=YES` before they become active. Therefore it is best to perform this when configuration is completed.

5.4 Common UGE-207 Item list

Configuration of UGE-207 Device options that are common to all gauge emulation modes.

Name		Description
Common		
UI	Item name: Description: Returned values:	UGE-207 Identification information Returns the identification information and firmware version (read-only) A string formatted as: UGE-207 x.y.z <license> <mode> <tag> x.y.z indicating the firmware version number <license> indicating the license applicable <Base, Plus, Premium> <mode> indicating the current Gauge Emulation mode <CONNECT> <tag> indicating the eight (8) character UGE-207 Device tag
UT	Item name: Description: Possible values:	UGE-207 Device Tag Read or set the UGE-207 identification label of maximum eight (8) ASCII characters including spaces. More than 8 characters results in an error message. ASCII characters: 'space', '!' .. '~' (default 'UGE_0001')
BPM		
BB	Item name: Description: Possible values:	BPM Baudrate Read or set the baudrate of the BPM port 1200, 2400 or 4800 (default 1200) Note: after changing the new value will become in use only after the UGE-207 has been reset (item US or hardware reset).
BG	Item name: Description: Possible values:	BPM Gain Read or set the sensitivity of the BPM receiver 1..8 (default 8, maximum)
BD	Item name: Description:	BPM Turnaround delay Read or set the minimum time waited before sending a reply after it has received a request on a BPM Host port

	Possible values:	0..100 ms (default 5 ms)
RS		
RB	Item name: Description: Possible values:	RS port Baudrate Read or set the baudrate of the RS port 1200, 2400, 4800, 9600, 19200, 38400, 57600 or 115200 (default 38400) Note: after changing the new value will become in use only after the UGE-207 has been reset (item US or hardware reset).
RT	Item name: Description: Possible values:	RS port Time-out Read or set the time-out value of a RS Field port for the Modbus RTU Slave Device responses. Possible values: 10..2000 ms (default 250 ms)
RY	Item name: Description: Possible values:	RS port parity and stop bit settings Read or set the RS port parity and stop bits. Format <parity: N, E or O><stop bits: 1 or 2> N1, N2, E1, E2, O1, O2 (Default value: N2). Note: after changing the new value will become in use only after the UGE-207 has been reset (item US or hardware reset).
Commands Items		
UL	Item name: Description: Possible values: Example:	Terminal New-line characters The New-line characters used by the UCI-108 in terminal mode in order to print the lines one below the other in the PC terminal screen. CR, LF, CRLF (default CRLF) RZUL will return RZULCRLF RZUL=CR will set the New-line character to CR
US	Item name: Description: Returned values: Example:	Reset System The UCI-108 will restart immediately. No response will be given on this command. Requires the confirmation parameter. - RZUS=YES
ER	Item name: Description:	Configuration Reset The UCI-108 will clear the current configuration and will set all configuration the factory defaults. The UCI-108 will

	Returned values: Example:	reset after processing command. Requires a restart. Requires the confirmation parameter. - RZER=YES
XR	Item name: Description: Returned values: Example:	Reset Communication statistics All communication statistics including it's elapse time will be reset. Requires the confirmation parameter. - RZXZ=YES
Readonly Items		
BN	Item name: Description: Returned values:	Measure BPM Host Port Noise Level Start a BPM Noise level measurement on the BPM Host Port of a remote UGE-207 device and respond the results. See chapter 4.9 and 7 for more details. <adcOut1_noise:adcOut2_noise:adcOut1_Request:adcOut2_Request> in mV
EG	Item name: Description: Returned values:	Get Configuration Information Returns meta information about the stored configuration. It consists of the Configuration Id, Version Id and the Configuration Changed Counter. <key>:<version>:<ccc>
UD	Item name: Description: Returned values:	DIP switch position This item returns 6 digits indicating the positions of the 6 DIP switches. A '1' indicates the ON position, a '0' indicates the OFF position. A string formatted as for example: 001000
UE	Item name: Description: Return values:	Environmental conditions. Returns the power supply voltage (V) and internal temperature (degrees Celcius). A string formatted as: U=<current voltage>:<min. voltage>:<max voltage>;T=<current temperature>:<min. temperature>:<max. temperature>;
UH	Item name: Description:	Memory Allocation Information Returns the number of free items in the various memory pools.

	Returned values:	A string formatted as: xx:xx:xx:xx:xx
UJ	Item name: Description: Returned values:	System Statistics Historical Data. A string formatted as: RH=<total run hours>;RS=<number of resets>;WS=<number of watchdog resets>;U=<min. voltage>:<max. voltage>;T=<min. temperature>:<max. temperature>;
UP	Item name: Description: Returned values:	UGE-207 Processor information Returns the processor serial number 96-bit unique ID (hexadecimal)
UR	Item name: Description: Returned values:	Runtime information Returns the uptime A string formatted as RT=<days>:<hours>:<minutes>:<seconds>
SV	Item name: Description: Returned values:	Software Version Returns the same result as with item 'UI'. See item 'UI' <i>Note: used to same abbreviation to be compliant with this common known GPU item.</i>
XG	Item name: Description: Returned values:	Get Communication Statistics. Return message counts, error counts and elapsed time. ET=<dd:hh:mm:ss>;GPUB=<msg:err>; MODR=<msg:err>; GPUB: GPU messages on BPM port MODR: Modbus messages on RS port msg: message count err: error count
Modbus Field port Items		
MR	Item name: Description:	Modbus Register Read Reads a Modbus Register at specified Modbus RTU device connected to the RS Field port. Format: <nn>RZMR=<dev addr>:<reg type>:<reg addr>:<data type>:<endianess> See paragraph 5.5.1 'Modbus Variable Register Definition' for details of the parameters. Examples:

	Returned values:	<p>98RZMR=9:H:2:U16:BE 98RZMR=9:H:2:U32:BE 98RZMR=09 03 02 1F AF 11 C9& 98RZMR=09 03 04 1F AF 16 F3 5B E3& others: 98RZMR!Failed to create MR request 98RZMR!Timeout on Modbus request The terminal command 'item MR=...' is not supported, use instead command 'mbr'.</p>
MW	<p>Item name: Description:</p> <p>Returned values:</p>	<p>Modbus Register Write Writes a Modbus Register at specified Modbus RTU device connected to the RS Field port. Format: <nn>RZMW=<dev addr>:<reg addr>:0x<hex data> Use 0xYYYY to enter a U16 value and 0xYYYYYYYY for U32. Examples: 98RZMW=9:14:0x2710 98RZMW=9:13:0xD8F02710 98RZMW=09 10 00 0E 00 01 61 42& 98RZMW!Failed to create MR request 98RZMW!Timeout on Modbus request The terminal command 'item MW=...' is not supported, use instead command 'mbw'.</p>

5.5 Modbus Variable Related Item list

Up to 20 Data Points can be defined for use in the UGE-207-Connect application. Depending on the license the Emulated Gauge variables may be linked to 4, 12 or 20 Data Points.

Configuration of these Data Points is done by using items stated in the table below.

Table 10: Modbus Variable related Item list

MVxx	<p>Item name: Description:</p>	<p>Modbus Variable Register Definition See 5.5.1 Modbus Variable Register Definition</p>
MSxx	<p>Item name: Description:</p>	<p>Modbus Status Register Definition See 5.5.1 Modbus Variable Register Definition</p>
MIxx	<p>Item name:</p>	<p>Modbus Variable Invalid Status Definition</p>

	Description:	See 5.5.4 Variable Invalid Status Definition
MRxx	Item name: Description:	Modbus Variable Range Conversion Definition See 5.5.2 Range Conversion Definition

These items are named by a two character name and a two digit index xx, (with xx=1..20). For example:

- 99RZMV01 Requests the definition of the Modbus Variable Register number 1.
- 99RZMI04=D Sets the Invalid Status Definition of the Modbus Variable to 'Disable'.

Note: To link the Emulated Gauge variables to the Data Points use the GxLV items. Where Gx indicates the Emulated Gauge (1..6).

Note: only the Modbus Variables of linked Data Points in the Emulated Gauges are queried.

5.5.1 Modbus Variable Register Definition

The item MVxx is used to define the reading of the Modbus Variable Register value, where xx is the index of the variable (xx=1..20). The item MVxx has the following format:

<dev addr>:<reg type>:<reg addr>:<data type>:<endianess>

The table below for explanation of the above format.

Table 11: Modbus Variable Register Definition fields

Field	Specification								
dev addr	The decimal Modbus RTU slave address (dev addr=1..247)								
reg type	'H' for Holding Register (read data using Modbus function code 3) 'I' for Input Register (read data using Modbus function code 4)								
reg addr	The decimal start address of the modbus register to use, ranging from 0 to 65,535.								
data type	<table border="1" style="width: 100%;"> <thead> <tr> <th>Code</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>U16</td> <td>16-bits unsigned integer</td> </tr> <tr> <td>I16C1</td> <td>16-bits signed integer in ones complement format</td> </tr> <tr> <td>I16C2</td> <td>16-bits signed integer in twos complement format</td> </tr> </tbody> </table>	Code	Description	U16	16-bits unsigned integer	I16C1	16-bits signed integer in ones complement format	I16C2	16-bits signed integer in twos complement format
	Code	Description							
	U16	16-bits unsigned integer							
	I16C1	16-bits signed integer in ones complement format							
I16C2	16-bits signed integer in twos complement format								

	I16S	16-bits signed integer where the msb bit indicates the sign (sign-magnitude representation)		
	I16	alias for I16C2		
	U32	32-bits unsigned integer		
	I32C1	32-bits signed integer ones complement format		
	I32C2	32-bits signed integer twos complement format		
	I32S	32-bits signed integer where the msb bit indicates the sign (sign-magnitude representation)		
	I32	alias for I32C2		
	F32	32-bits float (IEEE 754)		
endianess	This is the byte order of the data.			
	Code	Description	Example 32 bits e.g. 0x33221100	Example 16 bits e.g. 0x1100
	BE	Big Endian (standard)	[3][2][1][0]	[1][0]
	LE	Little Endian	[0][1][2][3]	[0][1]
	BES	Big Endian Byte Swap	[2][3][0][1]	do not use
	LES	Little Endian Byte Swap	[1][0][3][2]	do not use

Note: The traditional modbus addressing format is five digits, where the first digit is a prefix for the memory area. '3' for Input registers (Modbus function code 4) and '4' for Holding registers (Modbus function 3).

Note: The 'reg addr' follows the Modbus PDU protocol definition and is addressed form 0..65535. In the Modbus data Model each data is addressed from 1..n.

Examples:

MV01=9:I:11:F32:BE

Defines the value at RTU Modbus slave address 9, Input register address 11 (Modbus data Model address 30012) with data type 32-bits float and Big Endian

MV01=20:H:7:U16:BE

Defines the value at RTU Modbus slave address 20, Holding register address 7 (Modbus data Model address 40008) with data type 16-bits unsigned integer and Big Endian

Note: with additional UCI-108 connected with DIN rail connector the Modbus replies can be monitored on the RS port by the command 'msgmon'

Note: the fetched Modbus register value is internally converted to a 32-bit float.

5.5.2 Range Conversion Definition

The defined Modbus Variable value has to be converted to be available as a Data Point. This is accomplished by defining its input range and required Data Point output range. The value of the Data Point has no unit of its own. The unit of the Data Points will be determined by the type of Emulated Gauge variable and is internally represented as Meters, Celcius, Pascal or Kg/m3. However the Emulated Gauge supports conversion to other units, in case required by the CIU host, by means of the configuration item GxGU, see 5.6 Gauge Related Item list.

The range conversion is done using the MRxx items. These items have the following format:

<lower sensor>:<upper sensor>:<lower reference>:<upper reference>

After reading a Modbus Variable Register an offset and a scale will be applied based on these 4 parameters. These offset and scale are calculated to have the lower sensor value being mapped to the low reference value and the upper sensor value to the upper reference value.

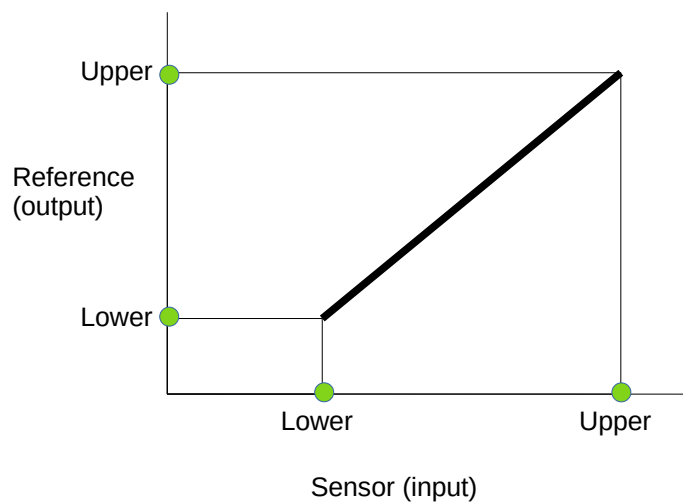


Figure 09: Range conversion

By default the range conversion values are zero for both lower values and 1 for both upper values which results in leaving the value as is.

Note: Data Points have a fixed unit and can be linked to any Emulated Gauge variable. The unit for linked Data Points will be determined by the internally used Emulated Gauge variables (Meters, Celcius, Pascal, Kg/m³).

Examples:

MR05=0:1000:0:1

Set the range conversion for Data Point 5 from millimeters to Meters.

MR06=4000:20000:-20:100

Set the range conversion for Data Point 6 from 4000 to 20000 micro amperes to -20 to 100 degrees Celcius.

MR07=0:16383:-20:100

Set the range conversion for Data Point 7 from ADC scale range to -20 to 100 degrees Celcius.

5.5.3 Modbus Status Register Definition

The item MSxx is used to define a Modbus Status Register to be used for the Variable Invalid Status Definition of the Modbus Variable, where xx is the index of the variable (xx=1..20). The item MSxx has the following format:

```
<dev addr>:<reg type>:<reg addr>:<data type>:<endianess>
```

See Table 11: Modbus Variable Register Definition fields for the explanation of the format. Commonly used <data type> for Status Register are U16 or U32.

Examples:

MS01=3:I:00013:U16:BE

Defines the status at RTU Modbus slave address 3, Input register address 00013 with data type 16-bits unsigned integer and Big Endian

5.5.4 Variable Invalid Status Definition

The status of a Modbus Variable Register value can be based on it's value or on the Modbus Status Register value. This is done using the MIxx items. These items have the following format:

```
<type>[:<value0>[:<value1>]]
```

Table 12: Variable Status Definition type

Indicator type	Meaning
D	Disabled. No extra processing is done for the status. Example: MI01=D
V	Error Value. The configured float value (<code>value0</code>) equal to which the process value indicates the status is 'invalid'. Example: MI01=V:382.0
L	Lower boundary. The configured float value (<code>value0</code>) below which the process value indicates the status is 'invalid'. Example: MI01=L:-32.0
H	Upper boundary. The configured float value (<code>value0</code>) above which the process value indicates the status is 'invalid'. Example: MI01=H:250.0
W	Error Value. The configured float values (<code>value0</code> & <code>value1</code>) form a window outside which the value indicates the status is 'invalid'. Example: MI01=W:-32.0:250.0
M	Error Mask. The configured U32 value (<code>value0</code>) is a mask representing the 'error' status bits. In case one or more of these corresponding bits in the Modbus Status Register are set the Modbus Variable Register value will be interpreted as invalid. Example: MI01=M:0x07 (stored as U32)
C	Complemented Error Mask. The configured U32 value (<code>value0</code>) is a mask representing the complemented 'error' status bits. In case one or more of the corresponding bits in the Modbus Status Register are cleared the Modbus Variable Register value will be interpreted as invalid. Example: MI01=C:0x03 (stored as U32)
S	Error Status. The configured U32 value (<code>value0</code>) representing the 'error' status bits. The error is considered active when the Status Register value equals the Error Status. Example: MI01=S:3009

Note: When a floating point number is specified always use a dot '.' as the decimal separator.

Note: M, C, S mask is always a 32 bit value. Status Register will be converted to 32 bits value before the and-ed operation with the defined mask.

Note: the fetched Modbus Variable Register or Modbus Status Register value is compared against the configured Invalid State value(s). This happens before Range Conversion is performed.

5.6 Gauge Related Item list

Up to six (6) gauges can be emulated by one UGE-207 device. Depending on the purchased license 4, 5 or 6 Emulated Gauges are available for operation.

Configuration of such an Emulated Gauge is done using items, see table below. These Emulated Gauge related items are prepended with the gauge id (G1, G2, . . . G6).

Examples:

99RZG1TI	Requests the item for the Tank Identifier for Emulated Gauge 1
99RZG4TI=SI_123	Sets the item for the Tank Identifier for Emulated Gauge 4

Table 13: Gauge related Item list

Name		Description
AL	Item name: Description: Possible values:	Level Alarm Level Alarm configuration. The level value (after range conversion) is compared to these values to set alarms. The unit is fixed and in meters. The format is: <hh>:<h>:<l>:<ll>:<hys> hh: float value indicating the High High level h: float value indicating the High level l: float value indicating the Low level ll: float value indicating the Low Low level hys: float value indicating the alarm hysteresis [0.0..1.0] default value 26.1:26:2:1.9:0.01
TA	Item name: Description: Possible values:	Tank Address. The GPU address of the Emulated Gauge. 1..99 (default value 98) <i>Note: address 00 is excluded for Modbus broadcast compatibility on BPM field bus</i>
GX	Item name: Description:	X-record Read or set the X-record string of maximal 8 ASCII characters including spaces. More than 8 characters will

	Possible values:	result in an error message. ASCII characters: 'space', '! .. '~' (default value '973 K10')
PN	Item name: Description: Possible values:	Production Number. The reported decimal production number. 0..99999 (default value 00000)
SV	Item name: Description: Possible values:	Software Version Read or set the Software Version string of maximal 38 ASCII characters. More than 38 characters will result in an error message. ASCII characters: 'space', '! .. '~' (default value '973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121')
TI	Item name: Description: Possible values:	Tank Identifier The Tank Identifier string Read or set the Tank Identifier label of maximum six (6) ASCII characters. ASCII character: 'space, '! .. '~' (default 'TI0098')
LV	Item name: Description: Possible values:	Link Data Point Value Link the Emulated Gauge variables to the converted Modbus Variables Data Points. It is a list of 6 indexes of which slots may be empty. Each slot can contain a value between 1..#licensed Modbus Variables. <ul style="list-style-type: none"> • First index: Product Level • Second index: Product Temperature • Third index: Vapour Temperature • Fourth index: Product Pressure (P1) • Fifth index: Vapour Pressure (P3) • Sixth index: Observed Density Default value LV=: : : : : meaning no variables linked and therefore this Emulated Gauge is disabled. Examples: All variables enabled: LV=1:2:3:4:5:6 Only level enabled: LV=1: : : : :
GU	Item name: Description:	Gauge Units The units used when sending responses from the

	Possible values:	<p>emulated gauge. Format <lu>:<tu>:<pu>:<du> <i>Note: the Emulated Gauge variables internally uses SI units for the linked Data Points : level in meters, temperature in °C, Pressure in Pa, Density in kg/m³.</i></p> <p>lu: length unit ('M', 'F' or 'I' for meters, feet, inches) tu: temperature unit ('C' or 'F' for Celcius or Fahrenheit) pu: pressure unit ('P', 'K' or 'S' for Pa, kPa or psi) du: density unit ('K' or 'L' for kg/m³ or lbs/ft³) (default value 'M:C:P:K')</p>
--	------------------	---

Note: When a floating point number is specified in item GxAL (with x=1..6) always use a dot '.' as the decimal separator.

6 Commissioning and Examples

This chapter describes examples in a step by step approach for the most commonly used UGE-207 Gauge Emulation setups.

The preferred way to commission the UGE-207 in the workshop is by creating a UGE-207/UCI-108 node by interconnecting both devices via DIN-rail backplane connectors. The UCI-108 enables a (temporary) additional RS port for the UGE-207 command line interpreter having the UGE-207 RS Field port available to connect to and test with the Modbus RTU devices. With an optional second UCI-108 the configured Emulated Gauges can be tested to by means of GPU requests. After configuration of the UGE-207 the UCI-108 can be removed.

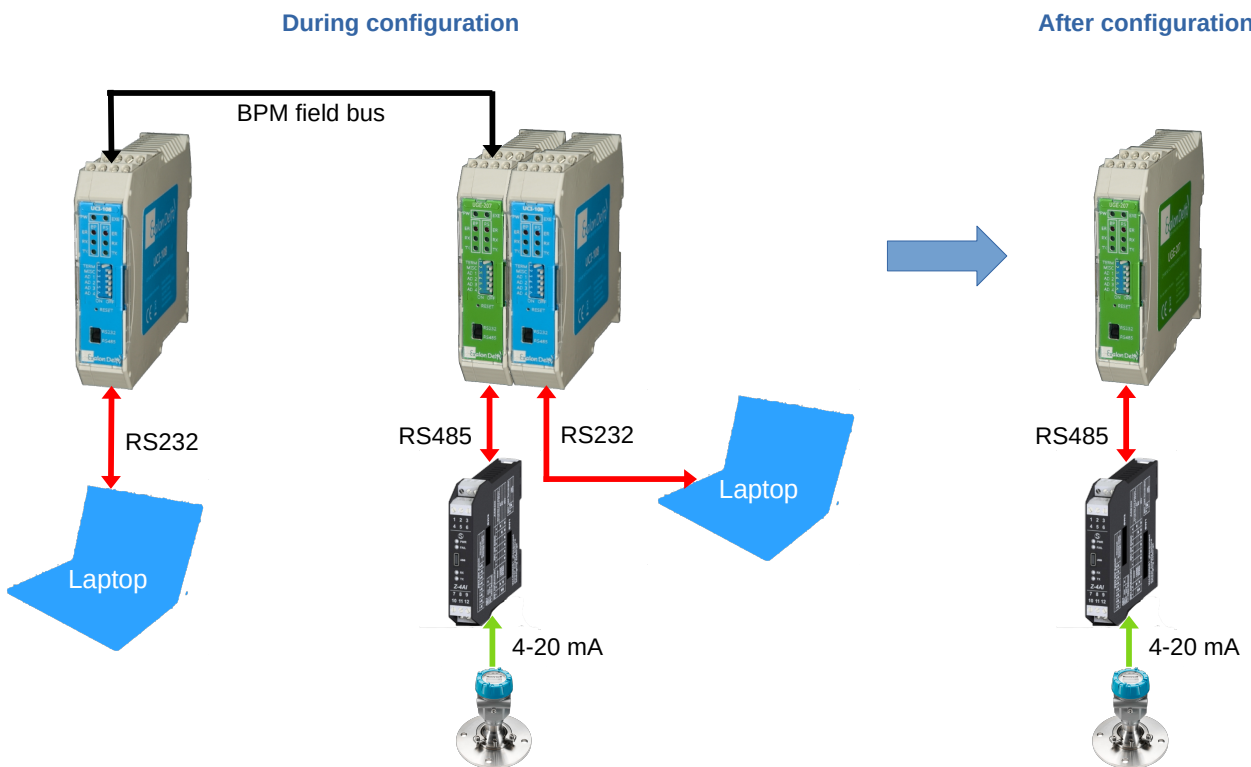


Figure 10: Commissioning of UGE-207

6.1 Entering RS port to TERMINAL Mode

Preparing the UGE-207/UCI-108 node:

- Connect the two DIN-rail bus connector together to form the UGE/UCI node backplane.

- Place the UGE-207 in the first and the UCI-108 in the second DIN-rail bus connector.
- Connect the 24VDC power supply to both the UGE-207 and UCI-108 terminals (**24VDC/GND**), see chapter 3.1 Power supply.
- Connect a PC terminal emulator via a RS232 cable to the RS serial port (**GNDiso/RX/TX**) of the UCI-108 and set the RS switch on the front panel in **RS232** position. See chapter 3.3 also RS232/RS485 serial bus.
- Connect the Modbus device(s) via RS485 to the RS serial port (**GNDiso/D+/D-**) of the UGE-207 and set the RS switch on the front panel in **RS485** position. See also chapter 3.3 RS232/RS485 serial bus.
- Set the UGE-207 DIP switch **AD1** in the **ON** position to assign node address 1.
- Set the UCI-108 DIP switch **AD4** in the **ON** position to assign node address 4.
- Enable the UCI-108 in Terminal mode by setting DIP switch **TERM** in the **ON** position, see chapter 5.1 Front Panel and Switches.
- Start a PC terminal emulation program connected to the UCI-108 serial port. Communication parameters are 115200 baud, 8 data bits, 1 stop bit, no parity, no flow control.
- Set the <CR><LF> correct in the terminal emulation to comply with the UCI-108 terminal output format ((default CRLF) in order to print the lines one below the other in the PC terminal emulation screen.

For the optional second UCI-108 connected to the BPM bus to the UGE-207:

- Enable the UCI-108 in Terminal mode by setting DIP switch **TERM** in the **ON** position, see chapter 5.1 Front Panel and Switches.
- Start a PC terminal emulation program connected to the UCI-108 serial port. Communication parameters are 115200 baud, 8 data bits, 1 stop bit, no parity, no flow control.
- Set the <CR><LF> correct in the terminal emulation to comply with the UCI-108 terminal output format ((default CRLF) in order to print the lines one below the other in the PC terminal emulation screen.
- After configuration of UGE-207 GPU requests can be entered at the command prompt by 'gpu <gpu request>'.

To enter the RS port in TERMINAL Mode:

- Power on the UGE-207/UCI-108 node and wait for the prompt 'UCI-108>' in the terminal emulator (or press <enter> to get the command prompt).
- Type 'cu 1' at the command prompt to connect to the UGE-207 command-line interpreter responding with prompt 'UCI-108 [1]>'.
- Type 'help' to get an overview of available commands, see below:

```
UCI-108 [1]>help
*****
```

```

* Command Line Interpreter for the UGE-207
*
* To run a command type the command its arguments and press 'Enter'
* Some commands support the 'help' parameter
*
* Available commands:
* help      : print list of all available commands or items
* item      : get/set configuration using Items
* msgmon    : monitor messages (GPU & Flexconn)
* data      : show current data
* mbr       : modbus read
* mbw       : modbus write
* stats     : show/reset statistics
* config    : show/setup/reset the current configuration
* reset     : reset the UGE-207
* cu        : change UCI (if multi node is enabled)
*****
UCI-108 [1]>

```

After commissioning activities leave the TERMINAL Mode:

- Power off the UGE-207/UCI-108 node
- Set the **TERM** DIP-switch of the UCI-108 in the **OFF** position and set the RS switch into the RS232 or RS485 position so it matches the RS serial port for other use.
- Set the UCI-108 DIP switch **AD4** in the **OFF** position.
- Disconnect the PC terminal emulator cable.
- Remove the UCI-108 from the DIN rail bus connector.

6.2 Example: Emulation for four tanks with 4-20 mA level gauges

This example describes the UGE-207 configuration to emulate level gauges for tanks equipped with lower accuracy and less expensive 4-20 mA level gauges to integrate into the Honeywell-Enraf CIU. See figure below for the system overview.

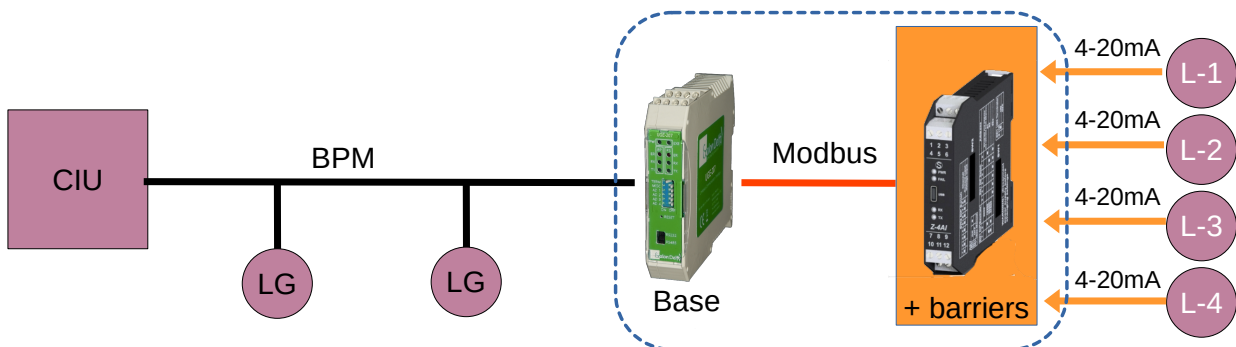


Figure 11: Emulation for 4-20 mA level gauges

Note: Barriers between Level Gauges are not shown. UGE-207 has to be placed in safe area or assembled in Ex d box if place in Zone 1 on the tank.

6.2.1 Pre conditions

- The 4-20 mA output range of the level gauges L-1 to L4 are set to:
 - 4 mA (4000 μ A) equals to 0 (zero) meter
 - 20 mA (20.000 μ A) equals to max. level value, e.g. 30 meter used in the example.
- The 4-20 mA gauges are via the barriers connected to a Modbus RTU multi channel Analog Input module. In this example the eight channel Z-8AI of the brand Seneca s.r.l. is used.
- The Z-8AI is connected to the UGE-207-Connect with a Base license to support four (4) emulated gauges with a maximum of four (4) Data Points to assign.
- The Z-8AI is configured by DIP switch settings to operate on Modbus device address 9 at 9600 baud 8N1, terminator enabled, analog inputs type to current.
- Use the Seneca 'Easy Setup' software tool to configure the Z-8AI 4-20 mA inputs for Electrical and Normalized Scaling factors. Set to the following values:
 - Electrical Start Scale: -20000
 - Electrical End Scale: 20000

- Normalized Start Scale: -20000
- Normalized End Scale: 20000
- The Z-8AI input channels used are IN1 to IN4 at respectively Modbus Register Address 40003 to 40006 (equals Register Address 2 to 5 in Modbus PDU protocol format, used during configuration).
- The UGE-207 is in the default configuration state. Type 'config reset' to return to the default state.

6.2.2 Settings for UGE-207 Device

- Type 'config setup' to start the quick menu, see screen below:

UGE-207 Main menu:

1. Common UGE configuration
2. Emulated Gauges
3. Modbus Variables

Enter choice:

- Choose option 1 'Common UGE configuration', the text will appear:

Setup Common UGE-207 items:

- | | | |
|------------------------------|------------|--|
| 1. UGE-207 Tag | = UGE_0001 | (8 ASCII characters) |
| 2. BPM port baudrate | = 1200 | [1200 2400 4800] |
| 3. BPM receiver gain | = 8 | [1..8] |
| 4. BPM port turnaround delay | = 5 | (ms) [0..100] |
| 5. RS port baudrate | = 38400 | [1200 2400 4800 9600 19200 38400 57600 115200] |
| 6. RS port timeout | = 250 | (ms) [10..2000] |
| 7. RS port parity | = N2 | [O E N] [1 2] |

Enter choice:

- Change the applicable UGE-207 Device settings, e.g. the 'UGE-207 Device tag' (option 1) and the 'RS port baudrate' (option 5) Set the RS baudrate to **9600**. Keep the 'RS port parity' in the default value **N2**.
- Press <enter> to leave the submenu and return to the Main menu.

6.2.3 Settings for UGE-207 Data Points

- Choose option 3 'Modbus Variables' to configure the Modbus Analog Input Registers and the corresponding Data Points. The next text will appear:

Choose Modbus Variable index [1..4] Enter choice:

- Choose Variable index 1 to configure the first gauge input connected to IN1 of the Z-8AI module. The next text will appear:

Modbus Variable 1 menu:

- | | |
|------------------------------|------------------|
| ...1. Read Value definition | = 247:I:0:U16:BE |
| 2. Range Conversion | = 0:1:0:1 |
| 3. Read Status definition | = 247:I:0:U16:BE |
| 4. Invalid Status definition | = D |

...Enter choice:

- Choose option 1 'Read Value definition':

Modbus Variable 18 Value definition:

- | | |
|---------------------|---------|
| 1. Device Address | : 247 |
| 2. Register Type | : Input |
| 3. Register Address | : 0 |
| 4. Data Type | : U16 |
| 5. Endianess | : BE |

...Enter choice:

- Select the options 1, 2, 3 and 4 and modify the settings respectively to the values **9** (Modbus Device Address), **H** (Holding Register), **2** (Register Address) and **I16** (Data Type). After entering the values the menu will show:

```
Modbus Variable 1 Value definition:
1. Device Address      : 9
2. Register Type      : Holding
3. Register Address   : 2
4. Data Type          : I16C2
5. Endianness         : BE
```

Enter choice:

- Press <enter> to leave the submenu and choose option 2 'Range Conversion'.

```
Modbus Variable 1 Range Conversion definition:
1. Lower Sensor value : 0
2. Upper Sensor value : 1
3. Lower Reference value : 0
4. Upper Reference value : 1
```

- Select the options 1, 2, 3 and 4 and modify the settings respectively to the values **0** (in μA), **20000** (in μA), **0** (in meter) and **30** (in meter) resulting in:

```
Modbus Variable 1 Range Conversion definition:
1. Lower Sensor value : 0
2. Upper Sensor value : 20000
3. Lower Reference value : 0
4. Upper Reference value : 30
```

Enter choice:

- Press twice <enter> to leave the submenu to return to:

Choose Modbus Variable index [1..4] Enter choice:

- Repeat above steps for the other three Modbus Variables associated with the level gauges on IN2 to IN4 of the Z-8AI. Use the following Z-8AI Register Addresses for the different Variable indexes:
 - Variable Index 2 at option 1, 3 'Register Address': **3**
 - Variable Index 3 at option 1, 3 'Register Address': **4**
 - Variable Index 4 at option 1, 3 'Register Address': **5**

- Press <enter> to leave the submenus and to return to the main menu:

```
UGE-207 Main menu:
1. Common UGE configuration
2. Emulated Gauges
3. Modbus Variables
Enter choice:
```

6.2.4 Settings for UGE-207 Emulated Gauges

- Select option 2 'Emulated Gauges':

Choose Gauge index [1..4] Enter choice:

- Select 1 to configure the first gauge, see terminal text below:

```
Setup Gauge items:
1. GPU address of emulated gauge = 98 0..99
2. Reported X-record string = 973 K10 max 8 ASCII characters
3. Reported Software Version string = 973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121 max 38 ASCII characters
4. Tank Identifier = TI0098 6 ASCII characters
5. Production Number = 00000 0..999999
6. The Variables to use = ::::: Max 6 Variable indices
7. The units of the emulated values = M:C:P:K 4 units
8. Level Alarm Values HH:H:L:LL:hys = 1000:1000:0:0:0.00999999978 5 float values
```

Enter choice:

- Select option 1 'GPU address of ...' to change the default GPU address of 98 to the desired value. In the example GPU address **21** is used.

- Select option 4 'Tank Identifier' to change the default tag 'TI0098' to the desired tag string. In the example tag **TI0021** is used.
- Select option 6 'The Variables to use' to link the Data Point number 1 of the first level gauge to the level variable of the first Emulated Gauge:

```
Sensor Data Points menu:
 1. Level           : -
 2. Product Temperature : -
 3. Vapour Temperature : -
 4. Pressure P1      : -
 5. Pressure P3      : -
 6. Density          : -
```

Enter choice:

- Select option 1 'Level' change the default value '-' into **1**.
- Press <enter> to return to the submenu 'Setup Gauge items:'
- Select option 8 'Level Alarm Values ...':

```
Gauge Level Alarm menu:
 1. High High Alarm : 1000
 2. High Alarm      : 1000
 3. Low Alarm       : 0
 4. Low Low Alarm   : 0
 5. Alarm Hysteresis : 0.01
```

Enter choice:

- Change the alarm values to the desired values, in this example
 - Option 1 'High High Alarm' into **28**,
 - Option 2 'High Alarm' into **27**
 - Option 3 'Low Alarm' into **2**
 - Option 4 'Low Low Alarm' into **0** meters.
 - Option 5 'Alarm Hysteresis' keep default value **0.01**
- Press <enter> twice to return to submenu to return to the submenu to configure the next Emulated Gauge.
- Repeat above steps for all Emulated Gauges but with:
 - GPU address of ...' to change the default GPU address of 98 to respectively **22, 23 and 24**.
 - 'Level' change the default value '-' into **respectively 2, 3 and 4**.
- Press <enter> twice to return to the command prompt.
- Restart the UGE-207 with the command 'reset' to operate with the new settings.

6.2.5 Overview of configuration and test

- Type after the reset of the UGE-207 at the command prompt "cu 1" to switch to the UGE-207 command line interpreter.
- Type 'config showall' to show all see the current configuration settings and check whether these conform to the list below:

```
UCI-108 [1]>config showall
```

```
=====|=====|=====
Description | Item | Active setting
=====|=====|=====
```

```

UGE-207 identification      | SV | UGE-207 1.1.0 Base CONNECT UGE_0001
UGE-207 device tag        | UT | UGE_0001
DIP switch positions (1..6) | UD | 001000
Multi-node address         |    | 1
Connection type RS port, BPM port |    | FIELD, HOST
EEProm meta data          | EG | 3210:0002:3
Processor serial number    | UP | 100004801951393035363334
=====
Terminal EOL Type         | UL | CRLF
BPM port baudrate         | BB | 1200
BPM receiver gain         | BG | 8
BPM port turnaround delay (ms) | BD | 5
RS port baudrate          | RB | 9600
RS port timeout (ms)      | RT | 250
RS port parameters        | RY | N2
=====
Transmission Address      | G1TA | 21
X-record                   | G1GX | 973 K10
Software version           | G1SV | 973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
Tank Identifier            | G1TI | TI0021
Product Number             | G1PN | 00000
Level Alarm                | G1AL | 28:27:0:0:0.0.00999999978
Linked Variables           | G1LV | 1:::::
Gauge Units                | G1GU | M:C:P:K
=====
Transmission Address      | G2TA | 22
X-record                   | G2GX | 973 K10
Software version           | G2SV | 973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
Tank Identifier            | G2TI | TI0022
Product Number             | G2PN | 00000
Level Alarm                | G2AL | 28:27:0:0:0.0.00999999978
Linked Variables           | G2LV | 2:::::
Gauge Units                | G2GU | M:C:P:K
=====
Transmission Address      | G3TA | 23
X-record                   | G3GX | 973 K10
Software version           | G3SV | 973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
Tank Identifier            | G3TI | TI0023
Product Number             | G3PN | 00000
Level Alarm                | G3AL | 28:27:0:0:0.0.00999999978
Linked Variables           | G3LV | 3:::::
Gauge Units                | G3GU | M:C:P:K
=====
Transmission Address      | G4TA | 24
X-record                   | G4GX | 973 K10
Software version           | G4SV | 973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
Tank Identifier            | G4TI | TI0024
Product Number             | G4PN | 00000
Level Alarm                | G4AL | 28:27:0:0:0.0.00999999978
Linked Variables           | G4LV | 4:::::
Gauge Units                | G4GU | M:C:P:K
=====
Modbus Value def.         | MV01 | 9:H:2:I16C2:BE
Range Conversion           | MR01 | 0:20000:0:30
Modbus Status def.        | MS01 | 247:I:0:U16:BE
Invalid Status def.       | MI01 | D
Modbus Value def.         | MV02 | 9:H:3:I16C2:BE
Range Conversion           | MR02 | 0:20000:0:30
Modbus Status def.        | MS02 | 247:I:0:U16:BE
Invalid Status def.       | MI02 | D
Modbus Value def.         | MV03 | 9:H:4:I16C2:BE
Range Conversion           | MR03 | 0:20000:0:30
Modbus Status def.        | MS03 | 247:I:0:U16:BE
Invalid Status def.       | MI03 | D
Modbus Value def.         | MV04 | 9:H:5:I16C2:BE
Range Conversion           | MR04 | 0:20000:0:30
Modbus Status def.        | MS04 | 247:I:0:U16:BE
Invalid Status def.       | MI04 | D
=====
License usage: 4 gauges out of 4 and 4 variables out of 4
UCI-108 [1]>

```

- Use command 'data datapoints' to show the operational internal values for the Modbus Analog Input Registers and the conversion to the corresponding Datapoints.

```
UCI-108 [1]>data datapoints
```

```
DP01: 0x00001F1E, 11.9490004 0x00
DP02: 0x00001714, 8.86200047 0x00
DP03: 0x000016FE, 8.82900047 0x00
DP04: 0x000027CB, 15.2805004 0x00
```

- Use command 'data gauges' to show the operational internal used values for the Emulated Gauges.

```
UCI-108 [1]>data gauges
```

```
Gauge[1]:
  level   : 11.9490004 0x00, alarm: 0x00

Gauge[2]:
  level   : 8.86200047 0x00, alarm: 0x00

Gauge[3]:
  level   : 8.82900047 0x00, alarm: 0x00

Gauge[4]:
  level   : 15.2805004 0x00, alarm: 0x00
```

- Retrieve the item list by the following command 'config items',

```
UCI-108 [1]>config items
```

```
item UL=CRLF
item BB=1200
item BG=8
item BD=5
item RB=9600
item RT=250
item G1TA=21
item G1GX=973 K10
item G1SV=973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
item G1TI=TI0021
item G1PN=00000
item G1AL=28:27:0:0:0.0.00999999978
item G1LV=1:::::
item G1GU=M:C:P:K
item G2TA=22
item G2GX=973 K10
item G2SV=973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
item G2TI=TI0022
item G2PN=00000
item G2AL=28:27:0:0:0.0.00999999978
item G2LV=2:::::
item G3GU=M:C:P:K
item G3TA=23
item G3GX=973 K10
item G3SV=973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
item G3TI=TI0023
item G3PN=00000
item G3AL=28:27:0:0:0.0.00999999978
item G3LV=3:::::
item G4GU=M:C:P:K
item G4TA=24
item G4GX=973 K10
item G4SV=973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
item G4TI=TI0024
item G4PN=00000
item G4AL=28:27:0:0:0.0.00999999978
item G4LV=4:::::
item G4GU=M:C:P:K
item MV01=9:H:2:I16C2:BE
```

```
item MR01=0:20000:0:30
item MS01=247:I:0:U16:BE
item MI01=D
item MV02=9:H:3:I16C2:BE
item MR02=0:20000:0:30
item MS02=247:I:0:U16:BE
item MI02=D
item MV03=9:H:4:I16C2:BE
item MR03=0:20000:0:30
item MS03=247:I:0:U16:BE
item MI03=D
item MV04=9:H:5:I16C2:BE
item MR04=0:20000:0:30
item MS04=247:I:0:U16:BE
item MI04=D
```

- Select, copy and save this list in a text file on your laptop. This can be reused later to configure the UGE-207 by an alternative method by using the command 'item'. Copy the text strings in bold to the command prompt and press <enter> for configuration of this item.

6.3 Example: Emulation for tank with 4-20 mA level, temperature, pressure

This example describes the UGE-207 configuration to emulate one level gauge for one tank equipped with 4-20 mA level gauge, average product spot temperature, vapour temperature and P3 vapor pressure transmitters to integrate into the Honeywell-Enraf CIU. See figure below for the system overview.

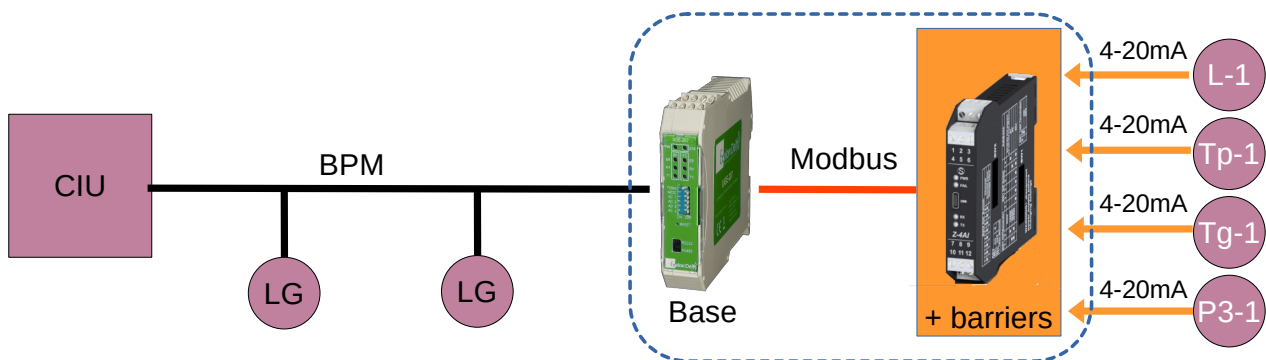


Figure 12: Emulation for 4-20 mA level, temperature and pressure transmitters

Note: Barriers between Level Gauges are not shown. UGE-207 has to be placed in safe area or assembled in Ex d box if place in Zone 1 on the tank.

6.3.1 Pre conditions

- The 4-20 mA output range of the instruments L-1, Tp-1, Tg-1 and P3-1 are set to:
 - L-1 on IN1:
 - 4 mA (4000 μ A) equals to 0 (zero) meter
 - 20 mA (20.000 μ A) equals to max. level value, e.g. 30 meter used in the example.
 - Tp-1 on IN2:
 - 4 mA (4000 μ A) equals to the minimal temperature, e.g. -40 $^{\circ}$ C used in the example.
 - 20 mA (20.000 μ A) equals to max. temperature value, e.g. -60 $^{\circ}$ C used in the example.
 - Tg-1 on IN3:
 - 4 mA (4000 μ A) equals to the minimal temperature, e.g. -40 $^{\circ}$ C used in the example.

- 20 mA (20.000 μ A) equals to max. temperature value, e.g. -60 °C used in the example.
- P3-1 on IN4:
 - 4 mA (4000 μ A) equals to 0 (zero) Pascal
 - 20 mA (20.000 μ A) equals to max. pressure value, e.g. 3000000 (3000 kPa) used in the example.
- See paragraph 6.2.1 for the other instructions on the preconditions.

6.3.2 Settings for UGE-207 Device

- See paragraph 6.2.2 for the instructions on the Device settings via the quick menu options.
- An alternative way is to configure by items. Type the following text for each item at the command prompt to configure the UGE-207 Device:

```
item RB=9600
```

6.3.3 Settings for UGE-207 Data Points

- See paragraph 6.2.3 for general instructions on the Data Point settings via the quick menu options.
- An alternative way is to configure by items. Type the following text for each item at the command prompt to configure the UGE-207 Data Points:

```
item MV01=9:H:2:I16C2:BE
item MR01=0:20000:0:30
item MV02=9:H:3:I16C2:BE
item MR02=0:20000:-40:60
item MV03=9:H:4:I16C2:BE
item MR03=0:20000:-40:60
item MV04=9:H:5:I16C2:BE
item MR04=0:20000:0:3000000
```

6.3.4 Settings for UGE-207 Emulated Gauges

- See paragraph 6.2.4 for general instructions on the Emulated Gauge settings via the quick menu options.
- An alternative way is to configure by items. Type the following text for each item at the command prompt to configure the UGE-207 Emulated Gauges:

```
item G1TA=21
item G1TI=TI0021
item G1AL=28:27:0:0:0.0.01
item G1LV=1:2:3::4:
```

6.3.5 Overview of configuration and test

- Restart the UGE-207 with the command 'reset' to operate with the new settings.
- Type after the reset of the UGE-207 at the command prompt 'cu 1' to switch to the UGE-207 command line interpreter.
- Type 'config showall' to show all see the current configuration settings and check whether these conform to the list below:

```
UCI-108 [1]>config showall
```

```
=====|=====|=====
Description | Item | Active setting
=====|=====|=====
UGE-207 identification | SV | UGE-207 1.1.0 Base CONNECT UGE_0001
UGE-207 device tag | UT | UGE_0001
DIP switch positions (1..6) | UD | 001000
Multi-node address | | 1
Connection type RS port, BPM port | | FIELD, HOST
EEprom meta data | EG | 3210:0002:3
Processor serial number | UP | 100004801951393035363334
=====|=====|=====
Terminal EOL Type | UL | CRLF
BPM port baudrate | BB | 1200
BPM receiver gain | BG | 8
BPM port turnaround delay (ms) | BD | 5
RS port baudrate | RB | 9600
RS port timeout (ms) | RT | 250
RS port parameters | RY | N2
=====|=====|=====
Transmission Address | G1TA | 21
X-record | G1GX | 973 K10
Software version | G1SV | 973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
Tank Identifier | G1TI | TI0021
Product Number | G1PN | 00000
Level Alarm | G1AL | 28:27:0:0:0.00999999978
Linked Variables | G1LV | 1:2:3::4:
Gauge Units | G1GU | M:C:P:K
=====|=====|=====
Transmission Address | G2TA | 98
X-record | G2GX | 973 K10
Software version | G2SV | 973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
Tank Identifier | G2TI | TI0098
Product Number | G2PN | 00000
Level Alarm | G2AL | 1000:1000:0:0:0.00999999978
Linked Variables | G2LV | ::::
Gauge Units | G2GU | M:C:P:K
=====|=====|=====
Transmission Address | G3TA | 98
X-record | G3GX | 973 K10
Software version | G3SV | 973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
Tank Identifier | G3TI | TI0098
Product Number | G3PN | 00000
Level Alarm | G3AL | 1000:1000:0:0:0.00999999978
Linked Variables | G3LV | ::::
Gauge Units | G3GU | M:C:P:K
=====|=====|=====
Transmission Address | G4TA | 98
X-record | G4GX | 973 K10
Software version | G4SV | 973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
Tank Identifier | G4TI | TI0098
Product Number | G4PN | 00000
Level Alarm | G4AL | 1000:1000:0:0:0.00999999978
Linked Variables | G4LV | ::::
Gauge Units | G4GU | M:C:P:K
=====|=====|=====
Modbus Value def. | MV01 | 9:H:2:I16C2:BE
Range Conversion | MR01 | 0:20000:0:30
Modbus Status def. | MS01 | 247:I:0:U16:BE
Invalid Status def. | MI01 | D
Modbus Value def. | MV02 | 9:H:3:I16C2:BE
```

```

Range Conversion          | MR02 | 0:20000:-40:60
Modbus Status def.      | MS02 | 247:I:0:U16:BE
Invalid Status def.     | MI02 | D
Modbus Value def.       | MV03 | 9:H:4:I16C2:BE
Range Conversion        | MR03 | 0:20000:-40:60
Modbus Status def.     | MS03 | 247:I:0:U16:BE
Invalid Status def.     | MI03 | D
Modbus Value def.       | MV04 | 9:H:5:I16C2:BE
Range Conversion        | MR04 | 0:20000:0:3000000
Modbus Status def.     | MS04 | 247:I:0:U16:BE
Invalid Status def.     | MI04 | D
=====|=====|=====
License usage: 1 gauges out of 4 and 4 variables out of 4
UCI-108 [1]>

```

- Use command 'data datapoints' to show the operational internal values for the Modbus Analog Input Registers and the conversion to the corresponding Datapoints.

```
UCI-108 [1]>data datapoints
```

```

DP01: 0x000027E5, 15.3195 0x00
DP02: 0x00001715, -10.4549999 0x00
DP03: 0x00000F93, -20.0650005 0x00
DP04: 0x00001BCE, 1067700 0x00

```

- Use command 'data gauges' to show the operational internal used values for the Emulated Gauges.

```
UCI-108 [1]>data gauges
```

```

Gauge[1]:
  level   : 15.3195 0x00, alarm: 0x00
  prod. T : -10.4549999 0x00
  vap. T  : -20.0699997 0x00
  pressure: 1067700 0x00

```

- Retrieve the item list by the following command 'config items',

```
UCI-108 [1]>config items
```

```

item UL=CRLF
item BB=1200
item BG=8
item BD=5
item RB=9600
item RT=250
item G1TA=21
item G1GX=973 K10
item G1SV=973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
item G1TI=TI0021
item G1PN=00000
item G1AL=28:27:0:0:0.0.00999999978
item G1LV=1:2:3::4:
item G1GU=M:C:P:K
item G2TA=98
item G2GX=973 K10
item G2SV=973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
item G2TI=TI0098
item G2PN=00000
item G2AL=1000:1000:0:0:0.00999999978
item G2LV=::::
item G3GU=M:C:P:K
item G3TA=98
item G3GX=973 K10
item G3SV=973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
item G3TI=TI0098
item G3PN=00000
item G3AL=1000:1000:0:0:0.00999999978
item G3LV=::::

```

```
item G4GU=M:C:P:K
item G4TA=98
item G4GX=973 K10
item G4SV=973 NVR1.0-- XPUK1.0-- APUB1.0 HCUA121
item G4TI=TI0098
item G4PN=00000
item G4AL=1000:1000:0:0:0.00999999978
item G4LV=::::
item G4GU=M:C:P:K
item MV01=9:H:2:I16C2:BE
item MR01=0:20000:0:30
item MS01=247:I:0:U16:BE
item MI01=D
item MV02=9:H:3:I16C2:BE
item MR02=0:20000:-40:60
item MS02=247:I:0:U16:BE
item MI02=D
item MV03=9:H:4:I16C2:BE
item MR03=0:20000:-40:60
item MS03=247:I:0:U16:BE
item MI03=D
item MV04=9:H:5:I16C2:BE
item MR04=0:20000:0:3000000
item MS04=247:I:0:U16:BE
item MI04=D
```

- Select, copy and save this list in a text file on your laptop. This can be reused later to configure the UGE-207 by an alternative method by using the command 'item'. Copy the text strings in bold to the command prompt and press <enter> for configuration of this item.

7 Diagnosing BPM fieldbus

The example in the figure below shows a setup to determine the BPM signal strength and noise level on location B. On this location B communication issues are expected due to signal attenuation or noise injection on the BPM fieldbus cable.

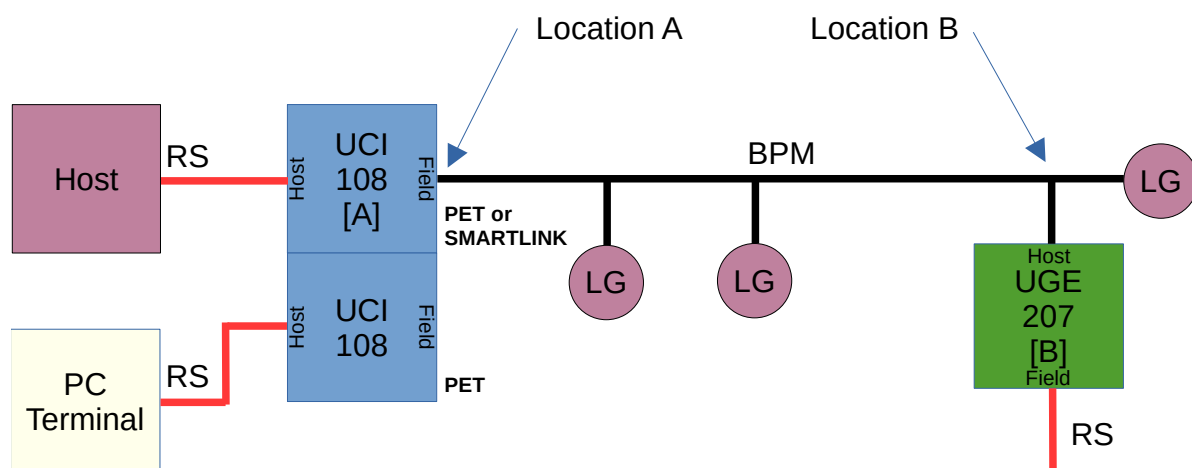


Figure 13: Measuring BPM fieldbus signal strength

Configure the setup as follows:

- UCI-108[A] is configured for use in multi UCI node making use of the DIN rail bus connector.
 - Set the node address to 1 by setting DIP switch 3 to 'ADDR 1'.
 - Configure UCI-108[A] in mode=SMARTLINK or mode=PET.
 - Configure UCI-108[A] as described in the previous chapter.
- UCI-108 is temporarily added at location A as command terminal in the multi UCI node configuration.
 - Set the node address to 4 by setting the DIP switch to 'ADDR 4'.
 - Set the UCI-108 in terminal mode by setting DIP switch 1 to 'TERM'. It will communicate to the PC Terminal program at 115k2 baud data format 8N1.
 - Configure UCI-108 in mode=PET.
- UCI-108[B] is temporarily added on location B in the field.
 - Set the UCI-108[B] in terminal mode by setting DIP switch 1 to 'TERM'. It will communicate to the PC Terminal program at 115k2 baud data format 8N1.

- Configure UCI-108[B] in mode=MODEM.
- Assign UCI-108[B] with GPU address e.g. 90 (to be used in the example)

Tip: reserve a spare DIN rail backplane position for a temporarily UCI-108 used for commissioning and service purposes. Make this UCI-108 node fixed as the Fourth node making it work in every UCI-108 multi node.

Different scenario's can be executed to locate the possible issues related to the BMP signal strength:

- On location A:
 - Switch prompt to UCI-108[A] terminal via the UCI-108 by the command `cu 1`. The prompt will show `UCI-108 [1]>`.
 - Issue the command `item LW` at the UCI-108[A] command prompt. This item shows the signal strength of the field instrument with the weakest received reply message. It will respond with `LW<address:uci:port:adcOut1:adcOut2>` detailed below:
 - `address` is the GPU address of the 'weakest' field instrument connected to the multi UCI node.
 - `uci` is the uci node address in the multi UCI node to which the 'weakest' field instrument is connected.
 - `port` is the port type to which the 'weakest' field instrument is connected. Is always `B` for BPM.
 - `adcOut1` the ADC value in mV before the Programmable Gain Amplifier (PGA). Recommended practical value has to be above 75 mV.
 - `adcOut2` the ADC value in mV before after the PGA. Optimum value would be around 2000 mV. In case higher mV values the PGA has to be lowered. First by querying the current value with the command `gpu 90RZBG` and then lower the value by one with the command `gpu 90RZBG=<new value>`.
- On location A:
 - Issue the command `item BL` at the UCI-108[A] command prompt. This item shows the noise level at location A for an idle BPM field bus. It will respond with `<adcOut1_noise:adcOut2_noise>` detailed below:
 - `adcOut1_noise` the ADC value in mV before the Programmable Gain Amplifier (PGA). Recommended practical value is below 50 mV.
 - `adcOut2_noise` the ADC value in mV before after the PGA.
- On location A:
 - Issue the command `gpu 90RZBN` at the UCI-108[A] command prompt. The UCI-108[B] is remotely addressed for item BN, measure BPM Noise level. It will response with `<adcOut1_noise:adcOut2_noise>` detailed below:

- `adcOut1_noise` the ADC value in mV before the Programmable Gain Amplifier (PGA). Recommended practical value is below 50 mV.
- `adcOut2_noise` the ADC value in mV before after the PGA.
- This item may be used for UCI-108s in the modes MODEM, PET or MODBUSTUNNEL with BPM port connected as host port to the BPM field bus.
- On location B:
 - Issue the command `msgmon` in the command line interpreter. The signal strength of received requests from location A to the field instruments (and their responses by the field instruments) are shown. The BPM fieldbus message are tagged with two measurements values, see Paragraph 4.7 for the format. These values can be used for further analysis.

8 Updating Firmware

Updating firmware can be done by booting into the UGE-207 bootloader and downloading the new firmware using the YModem protocol.

8.1 On Windows

Connect an YModem capable terminal program to the UGE-207 s RS port in RS232 mode. The communication parameters are 115200 baud, 8 data bits, 1 stop bit, no parity, no flow control.

This procedure has been tested with TeraTerm version 4.85 (SVN #5727). Newer versions might not work due to bugs in TeraTerms YModem implementation. ⁴

To boot the UGE-207 in it's bootloader set the **TERM** DIP-switch in the **On** position and press the **Reset Button** for more then 5 seconds.

The terminal emulator will then show a menu looking like:

```
***** Exalon Delft - Tank Gauging Solutions *****  
  
Bootloader 0.5. [248025001951393035363334]  
MSG: Console Activated.  
  
Current Application Image Status: present and valid.  
Released version 2.2, length: 145576, [C49A16C8]  
  
--- Firmware Maintenance Menu ---  
1: Download image to the internal Flash  
2: Execute the application
```

Type **1** and press **Enter** and the terminal will show:

```
Waiting for the firmware file to be sent ...  
(send 'a' to abort)
```

Then launch the YModem transfer (on TeraTerm use **File|Transfer|YModem|Send** and select the firmware file to flash.

4 Exalon Delft can supply you with a copy of this software. Another solution is using a macro to launch the transfer.

When done the result will be printed ("Programming completed." or an error message). The menu will reappear and the new firmware version should be shown.

8.2 On Linux or MacOS

Also on Linux and MacOS any YModem capable serial terminal program can be used.

There are also various command line utilities that can do the job. For example one could use **picocom** in combination with the **sb** tool (from the **lrzsz** package) the following way:

```
picocom -b 115200 -r -l -s "sz --ymodem -k -vv ./uci-108-  
application-2.2.0.bin" --imap lfcrLf /dev/ttyUSB0
```

Take care to use the correct serial port device.

To boot the UCI-108 in it's bootloader set the **TERM** DIP-switch in the **On** position and press the **Reset Button** for more then 5 seconds. The terminal emulator will then show a menu looking like described in the previous paragraph.

Type **1** and press **Enter** and the terminal will show:

```
Waiting for the firmware file to be sent ...  
(send 'a' to abort)
```

To start a file transfer press **Ctrl-A Ctrl-S** and press 'enter'-key to select the file already specified in the command arguments.

9 Appendix: Firmware release changelog

Release	Date	Changes
1.1.0	2024-11-27	Added: support for Product Pressure (P1)
1.0.3	2024-11-08	Fixed: issue showing 'ghost' characters on command prompt after assigning the Linked Values to an Emulated Gauge. Fixed: issue showing message '[ERROR] Failed to add request to IpRequestQueue'.
1.0.2	2024-08-28	Fixed: issue counting active license variables.
1.0.0	2024-07-26	Initial application release, operates with bootloader V0.5 and higher.