ExTox
Gasmess-Systeme GmbH

Instruction Manual
Modbus Converter Unigate-CL-RS

BA_MODBUS-Gateway_e_2011-12-16.doc, Stand: 16.12.2011

(Subject to technical changes)

1 Introduction

The Unigate-CL-RS Module serves as gateway between the ExTox RS232 Interface and the bus system of the customer (Modbus). Every second the ExTox Control Units ET-4D2 and ET-8D send a defined protocol via RS232 Interface. All data is buffered in the module and can be read out on Modbus side every time. It is possible to run several modules on the same bus. Therefore it is important to assign a clear address to each module.

2 Assignment of address

The address is adjusted at the module by means of the two turn able dip switches S6 (high nibble) and S7 (low nibble) in hexadecimal format.

Example: S6 is "B"
S7 is "3"

Modbus address is "B3" (hex) = "179" (decimal)

2.1 Adjustment of type of Interface

By means of the turn able dip switch S4 the type of Modbus Interface is adjusted. Three positions are possible:

0: RS232 1: RS422 2: RS485

2.2 Adjustment of Baud Rate:

Via dip switch S5 the Modbus baud rate is adjusted. Here ten positions are possible:

0: 300 Baud	1: 600 Baud
2: 1200 Baud	3: 2400 Baud
4: 4800 Baud	5: 9600 Baud
6: 19200 Baud	7: 38400 Baud
8: 57600 Baud	9: 115200 Baud

The position of the dip switch is only read out once on switching-on. In case for example a slave address should be changed, the module has to be switched off and on after adjusting the dip switch.

Factory sided setting is RS485 and 9600 Baud.

3 Modbus Protocol

3.1 General

Transfer mode is the RTU-Mode (Remote Terminal Unit). Modbus ASCII is not supported. Data format is "8N1" (8 Data bits, no parity, 1 Stop bit).

Modbus requests are only answered for the functional codes 0x03 and 0x04 (reading of words). On one request maximum 80 words (160 Bytes) can be read out. The Master sends a request to the slave. This request consists of 8 Bytes. The protocol is as follows:

1. Byte:	Slave address	
2. Byte:	Functional code (0x03 or 0x04)	
3. & 4. Byte:	Address first word to be read	(16 Bit Word)
5. & 6. Byte:	Number of words to be read	(16 Bit Word)
7. & 8. Byte:	CRC16 Check sum	(16 Bit Word)

The Slave answer in the following format:

1. Byte:	Slave address	
2. Byte:	Functional code (0x03 or 0x04)	
3. Byte:	Number of read Bytes	
from 4. Byte:	Data bytes	
then:	CRC16 Check sum	(16 Bit Word)

It has to be considered that it is only possible to read out words and no single bytes. But it is possible to address memory byte wise, that means to read out from any byte address.

3.2 Memory assignment of buffer (from ET4/ET8 Rev. 100728):

Remark: Independent from the connected control unit ET-4D2 or ET-8D the memory assignment is equivalent. If an ET-4D2 is used the measurement value "zero" will be send for channels 5 to 8. Also alarm status for channels 5 to 8 will be "No alarm" always.

Byte 0	Start sign (0x24)
Byte 1	Life sign (0x30 - 0x39) every second
Byte 2	Number of measuring point (0x30 – 0x36)
Byte 3	Status of system: 0x30=Air, 0x31=Rinse, 0x32=Measurement

Byte 4..9	Measured value Channel 1 in µA ASCII, leading zero=Blank
Byte 10..15	Measured value Channel 2 in µA ASCII, leading zero=Blank
Byte 16..21	Measured value Channel 3 in µA ASCII, leading zero=Blank
Byte 22..27	Measured value Channel 4 in µA ASCII, leading zero=Blank
Byte 28..33	Measured value Channel 5 in µA ASCII, leading zero=Blank
Byte 34..39	Measured value Channel 6 in µA ASCII, leading zero=Blank
Byte 40..45	Measured value Channel 7 in µA ASCII, leading zero=Blank
Byte 46..51	Measured value Channel 8 in µA ASCII, leading zero=Blank

Byte 52	Alarm 1 / Channel 1:	0x30=no Alarm, 0x31=Alarm
Byte 53	Alarm 1 / Channel 2:	0x30=no Alarm, 0x31=Alarm
Byte 54	Alarm 1 / Channel 3:	0x30=no Alarm, 0x31=Alarm
Byte 55	Alarm 1 / Channel 4:	0x30=no Alarm, 0x31=Alarm
Byte 56	Alarm 1 / Channel 5:	0x30=no Alarm, 0x31=Alarm
Byte 57	Alarm 1 / Channel 6:	0x30=no Alarm, 0x31=Alarm
Byte 58	Alarm 1 / Channel 7:	0x30=no Alarm, 0x31=Alarm
Byte 59	Alarm 1 / Channel 8:	0x30=no Alarm, 0x31=Alarm

Byte 60	Alarm 2 / Channel 1:	0x30=no Alarm, 0x31=Alarm
Byte 61	Alarm 2 / Channel 2:	0x30=no Alarm, 0x31=Alarm
Byte 62	Alarm 2 / Channel 3:	0x30=no Alarm, 0x31=Alarm
Byte 63	Alarm 2 / Channel 4:	0x30=no Alarm, 0x31=Alarm
Byte 64	Alarm 2 / Channel 5:	0x30=no Alarm, 0x31=Alarm
Byte 65	Alarm 2 / Channel 6:	0x30=no Alarm, 0x31=Alarm
Byte 66	Alarm 2 / Channel 7:	0x30=no Alarm, 0x31=Alarm
Byte 67	Alarm 2 / Channel 8:	0x30=no Alarm, 0x31=Alarm

Byte 68	Alarm 3 / Channel 1:	0x30=no Alarm, 0x31=Alarm
Byte 69	Alarm 3 / Channel 2:	0x30=no Alarm, 0x31=Alarm
Byte 70	Alarm 3 / Channel 3:	0x30=no Alarm, 0x31=Alarm
Byte 71	Alarm 3 / Channel 4:	0x30=no Alarm, 0x31=Alarm
Byte 72	Alarm 3 / Channel 5:	0x30=no Alarm, 0x31=Alarm
Byte 73	Alarm 3 / Channel 6:	0x30=no Alarm, 0x31=Alarm
Byte 74	Alarm 3 / Channel 7:	0x30=no Alarm, 0x31=Alarm
Byte 75	Alarm 3 / Channel 8:	0x30=no Alarm, 0x31=Alarm

Byte 76	Fault / Channel 1:	0x30=no Fault, 0x31=Fault
Byte 77	Fault / Channel 2:	0x30=no Fault, 0x31=Fault
Byte 78	Fault / Channel 3:	0x30=no Fault, 0x31=Fault
Byte 79	Fault / Channel 4:	0x30=no Fault, 0x31=Fault
Byte 80	Fault / Channel 5:	0x30=no Fault, 0x31=Fault
Byte 81	Fault / Channel 6:	0x30=no Fault, 0x31=Fault
Byte 82	Fault / Channel 7:	0x30=no Fault, 0x31=Fault
Byte 83	Fault / Channel 8:	0x30=no Fault, 0x31=Fault
Byte 84	Fault Housing fan:	0x30=no Fault, 0x31=Fault
Byte 85	Fault Flow-through:	0x30=no Fault, 0x31=Fault
Byte 86	Fault Communication:	0x30=no Fault, 0x31=Fault
Byte 87	Fault int. current:	0x30=no Fault, 0x31=Fault
Byte 88	Fault int. Check sum:	0x30=no Fault, 0x31=Fault
Byte 89	End sign 1 (0x0D)	
Byte 90	End sign 2 (0x0A)	