Annex of Operation Manual

# GMA200-MT/-MW

## Modbus implementation




GfG GESELLSCHAFT FÜR GERÄTEBAU MBH ▌ KLÖNNESTRASSE 99 ▌ D-44143 DORTMUND ▌ TELEFON +49 / (0)2 31 / 5 64 00 –0 ▌ FAX +49 / (0)2 31 / 51 63 13
INFO@GFG-MBH.COM ▌ WWW.GASMESSUNG.DE ▌ AMTSGERICHT DORTMUND HR B 2742 ▌ ZERTIFIZIERT NACH DIN EN ISO 9001:2008 ▌ ATEX QM ZERTIFIZIERT

# Table of content



## 1. General Information

The GMA200-MT/-MW has three RS485-busses (1 x GMA-Bus and 2x TRM-busses). The TRM bus nodes operate in the master mode and are intended for the communication with connected transmitters and relay modules. The GMA bus node may optionally be configured for master or slave mode. In the master mode it is intended for the communication with connected relay modules. In the slave mode only data can be readout from the GMA bus mode and processed by an extern master device (e.g. central unit or gateway to the control center), i.e. in this operation mode no relay modules can be managed by the GMA on the GMA bus. The communication on all busses is carried out according to the Modbus protocol in the RTU mode.
This document describes how and which data can be read from the GMA200-MT/-MW and therefore concerns only the GMA-bus in the slave operating mode.

## 2. PC Connection / Interface

RS485 (Multipoint)

- Bus-connection of master and slave via a two-core screened cable (clamping connection); e.g. bus-line 1x2x0,22mm2 BUS-LD or Bus-cable Y(St)Y 2x2x0,8

## 3. Bus structure

The maximum bus length should not exceed a distance of 1200m (see RS485 specification). A maximum of 64 modules can be connected to the GMA-bus. The transmission rate depends on the line length and can be set to a maximum of 230400 Baud.
The real maximum of line length and transmission rate depends very much on the construction of the network. In particular, Star topologies should be avoided due to resultant long stub lines. Advantageous is the construction of Daisy Chains. The following table shows the maximum Baud rates in dependency of the line lengths. The following example is intended to be a rough guideline.

| Line length | max. Baud rate |
|---|---|
| Up to 500m | 230400 Baud |
| Up to 1000m | 115200 Baud |
| Up to 1200m | 57600 Baud |

The bus levels are defined by the master. The bus should be terminated on both sides with 120Ω termination resistor. The GMA200-MT/-MW contains terminal resistors for all three busses, which can be switched on/off by a slide switch. These switches are located near the terminals and can be seen after removing the terminal cover. They can be activated by using suitable tools (e.g. screwdriver or similar) without opening the housing of the GMA200. The switch for the GMA-bus is located directly to the left of terminal 11 on the wall mounting module. On the rail mounting module this switch is located behind the terminals 67 – 69 (see also instruction manual).

## 4. Parameter of communication

| | |
|---|---|
| Bus Address | 1 … 247 (in slave operating mode)<br>Comment: Specification of the address also defines the operating mode of the bus node. The address 0 activates the master operating mode. Any other address activates the slave operating mode. |
| Transmission Rate | 9600, 19200, 38400, 57600, 115200, 230400 Baud |
| Data Format | 1 start bit<br>8 data bits<br>1 parity bit (even parity) with 1 stop bit (8E1) |
| Transmission Mode | RTU (Remote Terminal Unit) according to [MoSL] |

The settings of bus address and baud rate can be done in the service menu of the GMA or via the configuration software GMA Config on a PC (see Instruction manual). The data format and the transmission mode are fixed and cannot be changed.

## 5. Implementation of protocol

In reference to the ISO/OSI model the Modbus protocol consists of application layer, data link layer and physical layer, as shown in the adjacent table. The odd four layers remain unused.

| Layer | ISO/OSI Model | Modbus Protocol | Reference |
|---|---|---|---|
| 7 | Application Layer | MODBUS Application Protocol | [MAPS] |
| 6 | Presentation Layer | empty | |
| 5 | Session Layer | empty | |
| 4 | Transport Layer | empty | |
| 3 | Network Layer | empty | |
| 2 | Data Link Layer | MODBUS Serial Line Protocol | [MoSL] |
| 1 | Physical Layer | EIA/TIA-485 | |

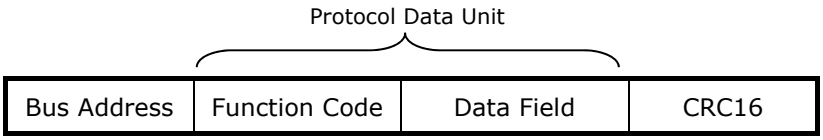
The subsequent protocol description exclusively refers to the application layer.

## 5.1 Modbus Function Codes

The GMA data can be read and written through the GMA-Bus. The data access is based on mapped register within the GMA, on which the Master can access to write and read data by using the following specialized Modbus function Codes [MAPS].

- **Read Input Registers 04 (0x04)**
- **Write Multiple Registers 16 (0x10)**
- **Read/Write Multiple Registers 23 (0x17)**

The registers represent 16-bit values with a 16-bit address range. The allocation of the transmission data to the register addresses will be described in section 5.3 "Register map". The following figure illustrates the universal structure of a Modbus frame, which consists of the bus address, the function code, the data field and the CRC16-modbus-checksum.




**Modbus Frame Structure**

## 5.2 Telegram Description

As illustrated in the upper figure, every telegram starts with the bus address and the function code and ends with the CRC16-modus-checksum. In between there is the application specific data field.
The CRC16-modbus-checksum is Modbus-specific and has the polynomial 0xA001 (see specifications according to [MoSL]). With the exception of the checksum all data will be transmitted in Big-Endian-Format (first MSB). The Checksum will be transmitted the Small-Endian-Format (first LSB).
Subsequently telegrams for the different general function codes and functionalities which are implemented in the GMA will be described.

### 5.2.1 Read Input Registers

The master sends the start address and the register number in the request for the readout of the register. The slave sends back the content of the requested register in the response.

| **Request** | Length | Content |
|---|---|---|
| Bus Address | 1 Byte | 1 … 247 |
| Function Code | 1 Byte | 0x04 |
| Start Address | 2 Bytes | 0x0000 … 0x06F2 |
| Number of Registers | 2 Bytes | 1 … 49 |
| Modbus Checksum | 2 Bytes | (0x0000 … 0xFFFF) |

| **Response** | Length | Content |
|---|---|---|
| Bus Address | 1 Byte | 1 … 247 |
| Function Code | 1 Byte | 0x04 |
| Number of Bytes | 1 Byte | 2 x N* |
| Register Content | N* x 2 Bytes | Daten |
| Modbus Checksum | 2 Bytes | (0x0000 … 0xFFFF) |

N* = Number of Register

### 5.2.2 Write Multiple Registers

The master sends the start address, the number and the content of the description of registers in the request for writing. The slave sends the start address and the number of registers as confirmation.

| **Request** | Lenght | Content |
|---|---|---|
| Bus Address | 1 Byte | 1 … 247 |
| Function Code | 1 Byte | 0x10 |
| Start Address | 2 Bytes | 0x1000 |
| Number of Registers | 2 Bytes | 1 |
| Number of Bytes | 1 Bytes | 2 |
| Register Content | 2 Bytes | Daten |
| Modbus Checksum | 2 Bytes | (0x0000 … 0xFFFF) |

| **Response** | Length | Content |
|---|---|---|
| Bus Address | 1 Byte | 1 … 247 |
| Function Code | 1 Byte | 0x10 |
| Start Address | 2 Bytes | 0x1000 |
| Number of Registers | 2 Bytes | 1 |
| Modbus Checksum | 2 Bytes | (0x0000 … 0xFFFF) |

### 5.2.3 Read / Write Multiple Registers

The Master sends the start address and the number of start address to be read, number and content of the register to be written in the request. The slave sends the content of the requested register in the response.

| **Request** | Length | Content | |
|---|---|---|---|
| Bus Address | 1 Byte | 1 … 247 | |
| Function Code | 1 Byte | 0x17 | |
| Start Address (R) | 2 Bytes | 0x0000 … 0x06F2 | (R) = Read |
| Number of Registers (R) | 2 Bytes | 1 … 49 | |
| Start Address (W) | 2 Bytes | 0x0000 | (W) = Write |
| Number of Registers (W) | 2 Bytes | 1 | |
| Number of Bytes (W) | 1 Bytes | 2 | |
| Register Content (W) | 2 Bytes | Datta | |
| Modbus Checksum | 2 Bytes | (0x0000 … 0xFFFF) | |

| **Response** | Length | Content | |
|---|---|---|---|
| Bus Address | 1 Byte | 1 … 247 | |
| Function Code | 1 Byte | 0x17 | |
| Number of Bytes | 1 Byte | 2 x N* | N* = Number of Registers |
| Register Content | N* x 2 Bytes | Data | |
| Modbus Checksum | 2 Bytes | (0x0000 … 0xFFFF) | |

### 5.2.4 Exception Code

If an error occurs during reading or writing, the Slave sends an exception code instead of the expected response and sets the bit with the highest value (MSB) in the function code.

| **Error-Response** | Länge | Inhalt |
|---|---|---|
| Bus Address | 1 Byte | 1 … 247 |
| Function Code | 1 Byte | 0x84 / 0x90 / 0x97 |
| Exception Code | 1 Byte | 0x01 ... 0x04 |
| Modbus Checksum | 2 Bytes | (0x0000 … 0xFFFF) |

In the implemented function codes the four following exception codes are possible.

| **Exception Code** | **Designation** | **Description of Error** |
|---|---|---|
| 0x01 | Illegal Function | Invalid Function Code |
| 0x02 | Illegal Data Address | Invalid Register Address |
| 0x03 | Illegal Data Value | Invalid number of register |
| 0x04 | Slave Device Failure | Error in reading the register |

## 5.3 Register Map

For the most part the registers are divided in two separate ranges. Only the read access of the one range is possible (Read Input Registers) while the other allows only the write access (Write Multiple Registers). Only one register (0x0000) is for both read and write access. Processes of reading and writing can also be combined (Read/Write Multiple Registers) in a communication cycle (Request ←→ Response).

### 5.3.1 Read Input Registers

Basically the transmission data, which can be read from the GMA, can be divided into four groups:

| Register Addresses | Transmission Data |
|---|---|
| 0x0000 … 0x0021 | GMA Identification Parameter |
| 0x0100 … 0x0121 | GMA Status as well as measured values and status information of the transmitter respectively of the Measuring Points |
| 0x0200 … 0x06F2 | Configuration parameters according to the measuring values of the transmitter respectively of the Measuring Points |

In section 5.5 "Chronological sequence of the measured value request" a description of which data should be read by the GMA can be found.

The subsequent table describes the allocation of transmission data to the register addresses and the structure of data in detail. By reading the register it should be ensured that the access to not specified register between the data segments in the register map should be denied. For this reason only a limited number of registers in dependency of the register start address can be read in a telegram.

**REGISTER MAP - READ INPUT REGISTERS 04 (0x04)**

| Register Address | max. request-able Register | Type of register (16 Bit) | Parameter | Explanation |
|---|---|---|---|---|
| 0x0000 | 34 | Unsigned | Signature | Signature for telegram identification<br>(will be decrement after reading) |
| 0x0001 | 33 | Unsigned | GMA Type | Coding:<br>*0: GMA200-MT16*<br>*1: GMA200-MT6*<br>*2: GMA200-MW16*<br>*3: GMA200-MW8*<br>*4: GMA200-MW4* |
| 0x0002 | 32 | Unsigned | FW Version (Main) | Firmware-Version of the Mainboard<br>*e.g. 192 = v1.92* |
| 0x0003 | 31 | Unsigned | FW Version (Disp) | Firmware-Version of the Displayboard<br>*e.g. 152 = v1.52* |
| 0x0004 | 30 | Unsigned | GMA Serial number | First ASCII-Character (UTF-8)<br>of the serial number with a maximum of 10 digits |
| … | … | … | … | *e.g: 49,50,49,49,49,57,50,55,0,0 = „12111927"*<br>*(here with Null-terminating)* |
| 0x000D | 21 | Unsigned | GMA Serial number | Last ASCII-Character (UTF-8)<br>of the serial number with maximum of 10 digits |
| 0x000E | 20 | Unsigned | GMA Designation | First ASCII-Character (UTF-8)<br>of the GMA-designation with a maximum of 20 digits |
| … | … | … | … | *e.g.: 72,65,85,83,32,51,0,0,0,0,0,0,0,0,0,0,0,0,0,0*<br>*= "HAUS 3"* |
| 0x0021 | 1 | Unsigned | GMA Designation | Last ASCII-Character (UTF-8)<br>of the GMA-designation with a maximum of 20 digits |

Continue on next page

| Register Address | max. request-able register | Type of register (16 Bit) | Parameter | Explanation | |
|---|---|---|---|---|---|
| 0x0100 | 34 | Unsigned | Signature | Signature for telegram identification (will be decrement after reading) | |
| 0x0101 | 33 | Unsigned | GMA-Status (#1) | Bit definition:<br>b0: Startup<br>b1: Error<br>b2: maintenance<br>b3: maintenance request<br>b4: Configuration mode } 1= active; 0=inactive<br>b5: Level on AL-Reset input 1<br>b6: Level on AL-Reset input 2 } 1= High, 0= Low | |
| 0x0102 | 32 | Unsigned | TRM-Status (#1) | Bit definition (1=active; 0=inactive):<br>b0: Activated<br>b1: Blocked<br>b2: Startup<br>b3: Error<br>b4: maintenance<br>b5: maintenance request<br>b6: Configuration mode<br>b7: Alarm 1<br>b8: Alarm 2<br>b9: Alarm 3<br>b10: Under-Range<br>b11: Over-Range | Measuring point 1 |
| 0x0103 | 31 | Signed | TRM Measuring value | Measuring value according to scaling factor, unit und type of gas<br>*Example:*<br>*- measuring value = 209*<br>*- scaling factor = -1*<br>*- unity = 2*<br>*- type of gas = 89*<br>*→ gas measuring value = 20,9 Vol.% $O_2$* | |
| … | … | … | … | … | Measuring points 2–15 |
| 0x0120 | 2 | Unsigned | TRM Status | *(see register address 0x0102)* | Measuring point 16 |
| 0x0121 | 1 | Signed | TRM Measuring value | *(see register address 0x0103)* | |
| 0x0200 | 49 | Unsigned | Signature | Signature for telegram identification (will be decrement after reading) | |
| 0x0201 | 48 | Signed | Scaling Factor | Of the TRM measuring value<br>–3: 0,001<br>–2: 0,01<br>–1: 0,1<br>0: 1 | Measuring point 1 |
| 0x0202 | 47 | Unsigned | Unit | Of the TRM measuring value<br>*Coding by number according GfG-tables of units and gas types (see section 6)* | |
| 0x0203 | 46 | Unsigned | Type of Gas | | |
| … | … | … | … | … | Measuring point 2–15 |
| 0x022E | 3 | Signed | Scaling Factor | *(see register address 0x0201)* | Measuring point 16 |
| 0x022F | 2 | Unsigned | Unit | *(see register address 0x0202)* | |
| 0x0230 | 1 | Unsigned | Type of Gas | *(see register address 0x0203)* | |

Regarding #1: See "Explanation of bit definition" at the end of this section

Continue on next page

| Register Address | max. request-able register | Type of register (16 Bit) | Parameter | Explanation | |
|---|---|---|---|---|---|
| 0x0300 | 49 | Unsigned | Signature | Signature for telegram identification (will be decrement after reading) | |
| 0x0301 | 48 | Signed | MBB | Start and end of the measuring range according to the scaling factor | 0x0301 0x0302 0x0303 |
| 0x0302 | 47 | Signed | MBE | | |
| 0x0303 | 46 | Unsigned | Directions of Alarm | Direction of the alarms AL3,AL2,AL1 with the bit definition b2, b1, b0. Value=1: exceeding Alarm Value=0: lower Alarm | |
| … | … | … | … | … | … |
| 0x032E | 3 | Signed | MBB | *(see register adress 0x0301)* | 0x032E 0x032F 0x0330 |
| 0x032F | 2 | Signed | MBE | *(see register adress 0x0302)* | |
| 0x0330 | 1 | Unsigned | Directions of Alarm | *(see register adress 0x0303)* | |
| 0x0400 | 49 | Unsigned | Signature | Signature for telegram identification (will be decrement after reading) | |
| 0x0401 | 48 | Signed | AL1 | Alarm treshold 1, 2, 3 according to the scaling factor. (Value=0: Alarm disabled) | 0x0401 0x0402 0x0403 |
| 0x0402 | 47 | Signed | AL2 | | |
| 0x0403 | 46 | Signed | AL3 | | |
| … | … | … | … | … | … |
| 0x042E | 3 | Signed | AL1 | *(see register adress 0x0401)* | 0x042E 0x042F 0x0430 |
| 0x042F | 2 | Signed | AL2 | *(see register adress 0x0402)* | |
| 0x0430 | 1 | Signed | AL3 | *(see register adress 0x0403)* | |
| 0x0500 | 19 | Unsigned | Signature | Signature for telegram identification (will be decrement after reading) | |
| 0x0501 | 18 | Unsigned | Designation of Measuring Point Text 1 | First ASCII-Character (UTF-8) of text 1 with a maximum of 10 digits | 0x0501 … 0x050A 0x050B … 0x0512 |
| … | … | … | … | *e.g. 82,97,117,109,32,49,0,0,0,0 = "Raum 1"* | |
| 0x050A | 9 | Unsigned | Designation of Measuring Point Text 1 | Last ASCII-Character (UTF-8) of text 1 with a maximum of 10 digits | |
| 0x050B | 8 | Unsigned | Designation of Measuring Point Text 2 | First ASCII-Character (UTF-8) of text 2 with a maximum of 8 digits | |
| … | … | … | … | *e.g.: 71,97,115,108,97,98,111,114 = "Gaslabor"* | |
| 0x0512 | 1 | Unsigned | Designation of Measuring Point Text 2 | Last ASCII-Character (UTF-8) of text 2 with a maximum of 8 digits | |
| 0x0520 | 19 | Unsigned | Signature | Signature for telegram identification (will be decrement after reading) | |
| 0x0521 | 18 | Unsigned | Designation of Measuring Point Text 1 | *(see register adress 0x0501)* | Measuring point 2 |
| … | | | | | |
| 0x0532 | 1 | Unsigned | Designation of Measuring Point Text 2 | *(see register adress 0x0512)* | |
| … | … | … | … | … | Measuring point 3–15 |
| 0x06E0 | 19 | Unsigned | Signature | Signature for telegram identification (will be decrement after reading) | |
| 0x06E1 | 18 | Unsigned | Designation of Measuring Point Text 1 | *(see register adress 0x0501)* | Measuring point 16 |
| … | … | … | … | … | |
| 0x06F2 | 1 | Unsigned | Designation of Measuring Point Text 2 | *(see register adress 0x0512)* | |

**Explanation of bit definition**

| GMA-Status – Bit definition | | |
|---|---|---|
| b0 | Startup | This bit is set, when after a reset the GMA is in the startup phase. During this period self-tests and preparations for the measuring operation will be done. Thus there are no valid measuring values from the Measuring Points for evaluation available. This bit will be reset as soon as the startup phase ends. |
| b1 | Error | This bit is set, when an active error on the GMA is existent (e.g. supply voltage outside the valid range). During this period measuring values of the Measuring Points should not be evaluated, because there is no guarantee for their validity. This bit will be reset as soon as the error(s) is (are) no longer active. |
| b2 | Maintenance | This bit is set, when maintenance is carried out on the GMA (e.g. Test of the relay outputs) and will be reset as soon as the maintenance is done. |
| b3 | Maintenence Request | This bit is set, when there is an active maintenance request on the GMA (e.g. excess of the inspection date) and will be reset as soon as no maintenance request is active. |
| b4 | Configuration Operation Mode | This bit is set, when the GMA switches into the configuration operation for the change of configuration parameters. During this period measuring values of the Measuring Points should not be evaluated, because there is no guarantee for their validity. This bit will be reset as soon as the configuration mode is quit. |
| b5 | Level on AL-Reset Input 1 | This bits indicate the level on the alarmreset inputs<br>1= High-level: Extern switch open<br>0= Low-Level: Extern switch closed |
| b6 | Level on AL-Reset Input 2 | |

| TRM-Status - Bitdefinition | | |
|---|---|---|
| b0 | Activated | Measuring Point is active. The measuring values of <u>inactive</u> Measuring Points have a default value (0) and are not allowed to be evaluated. |
| b1 | Blocked | Measuring Point is blocked. I.e. independent from the measuring signal of the Measuring Point alarms and errors won't be triggered. For this Measuring Point the need for maintenance will be signalized. |
| b2 | Startup | Measuring Point is in the startup phase after a reset. As soon as this phase is done the bit will be reset. |
| b3 | Error | At least one error is active. Measuring value is invalid. |
| b4 | Maintenance | Maintenance on this Measuring Point. |
| b5 | Maintenance Request | At least one maintenance request is active. |
| b6 | Configuration Operation Mode | The GMA is in the configuration operation. Because the parameters can change during this operation, the validity of the measuring values cannot be guaranteed. These measuring values <u>should not</u> be evaluated. |
| b7 | Alarm 1 | Alarm 1 triggered |
| b8 | Alarm 2 | Alarm 2 triggered |
| b9 | Alarm 3 | Alarm 3 triggered |
| b10 | Under Range | The measuring value is clearly under the measuring range |
| b11 | Over Range | The measuring value is clearly over the measuring range |

### 5.3.2 Write Multiple Registers

Only two registers can be written to either set the telegram signature or to quit active alarms of the Measuring Points.

| REGISTER-MAP - WRITE MULTIPLE REGISTERS 16 (0x10) | | | | |
|---|---|---|---|---|
| **Register Address** | **Number of transmit-able Register** | **Type of register (16 Bit)** | **Parameter** | **Explanation** |
| 0x0000 | 1 | Unsigned | Signatur | Signature for telegram identification<br>(will be decrement after reading) |
| 0x1000 | 1 | Unsigned | Alarm Acknowledgement (AL-Reset) | Acknowledgement of active alarms of Measuring Points. A Measuring Point is allocated to every bit. On every set bit active alarms will be acknowledge according to the Measuring Point (MP) → Bit0:15 = MP1:16<br>example:<br>0x0001: Alarm acknowledgement of measuring point 1<br>0x0010: Alarm acknowledgement of measuring point 5<br>0xFFFF: Alarm acknowledgement of measuring point 1-16 |

## 5.4 Explicit Telegram Identification

For the explicit telegram identification a signature as register content in conjunction with the function code for reading and writing of registers **(Read/Write Multiple Register 0x17)** can be transmitted. Therefore the Master sends a signature with the request for writing the signature register and requires simultaneously the same signature along with further register values from the GMA, which will be send in a response to the Master thereafter. The signature, which therefore is transmitted by the Master, can be counter value, for example, which gets incremented before every request. By the comparison of the sent and the received signature by the GMA, the Master is able to recognize, if the received telegram is the response to the sent request.

All data segments start with the signature register, which can be read along with the attached data register of the single segments. In doing so the same signature gets accessed in the signature registers of the different segments. This signature can just be described by the only register with the address 0x0000 (see section 5.3 "Register map").

After reading the signature, it will automatically decremented by the GMA. If the master accesses the signature and data segments for reading repeatedly **(Read Input Registers 0x04)**, then due to automatic decrementation of signature the master is able to recognize that the responses from the GMA are ongoing created and sent telegrams.

The above described telegram identification presumes the reading of entire data segments with signature from the particular start address.

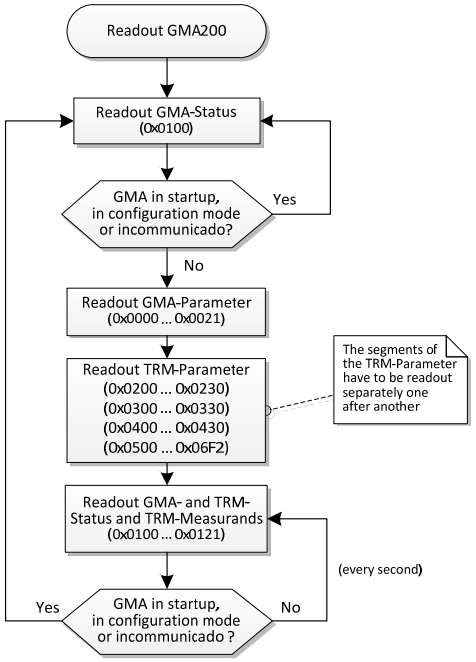
## 5.5 Chronological sequence of the measured value request

The configurations parameter of GMA and TRM must be read after the system start (Startup) or after configuration changes, because otherwise these data won't change. The parameters should also be read after long interruptions of communication since the executed changes which were made during the interruption could not be recognized. Every of four parameter signals should therefore be individually successively read.

Subsequently the GMA and TRM status information as well as the TRM measuring values will be cyclical read together. The cycle time therefore should not underrun one second as this period meets with the cycle time for the measurement within the GMA. If during the cyclic status the GMA switches into the configuration mode, is not available or is in startup caused by a reset, the status and measurements of the transmitter don't need to be read, because the correctness of the measurements can't be ensured due to configuration changes.

In this case the status of the GMA should be read out until the GMA quits the configuration operation or the startup phase respectively the GMA is available again.

The illustration displays an example for the chronological sequence of reading the data from the GMA.




| **Example for data request** | Bus-Address (**#1**) | Function Code | Register Address (R) | Number of Register (R) | Register Address (W) | Number of Register (W) | Number of Bytes (W) | Register Content (W) (**#2**) | Modbus-Checksum |
|---|---|---|---|---|---|---|---|---|---|
| GMA identification parameter | 0x01 | 0x17 | 0x0000 | 0x0022 | 0x0000 | 0x0001 | 0x02 | 0x0000 | CRC16 |
| Status and measure values | 0x01 | 0x17 | 0x0100 | 0x0022 | 0x0000 | 0x0001 | 0x02 | 0x0001 | CRC16 |
| Measure value parameter | 0x01 | 0x17 | 0x0200 | 0x0031 | 0x0000 | 0x0001 | 0x02 | 0x0002 | CRC16 |
| Measuring ranges | 0x01 | 0x17 | 0x0300 | 0x0031 | 0x0000 | 0x0001 | 0x02 | 0x0003 | CRC16 |
| Alarm tresholds | 0x01 | 0x17 | 0x0400 | 0x0031 | 0x0000 | 0x0001 | 0x02 | 0x0004 | CRC16 |
| MP designation (MP1) | 0x01 | 0x17 | 0x0500 | 0x0013 | 0x0000 | 0x0001 | 0x02 | 0x0005 | CRC16 |
| MP designation (MP2) | 0x01 | 0x17 | 0x0520 | 0x0013 | 0x0000 | 0x0001 | 0x02 | 0x0006 | CRC16 |
| MP designation (MP3-15) | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| MP designation (MP16) | 0x01 | 0x17 | 0x06E0 | 0x0013 | 0x0000 | 0x0001 | 0x02 | 0x0014 | CRC16 |

Comment (**#1**): Bus Address of GMA (variable); (**#2**): This Signature (variable) should always change

# 6. Tables of GfG codes

## 6.1 GfG table of units

| No. | Code | Designation | No. | Code | Designation |
|---|---|---|---|---|---|
| **1** | ppm | Part per Million | **14** | A | Ampere |
| **2** | Vol% | Percent by volume | **15** | Ohm | Ohm |
| **3** | %LEL | Lower explosion level | **16** | Dig | Digit |
| **4** | ppb | Part per Billion | **23** | Grd | Degree |
| **5** | ug | Microgram | **24** | °F | Degree Fahrenheit |
| **6** | mg | Milligram | **25** | g | Gram |
| **7** | % | Percent | **26** | kg | Kilogramme |
| **8** | %% | Per mill | **27** | Pa | Pascal |
| **9** | m/s | Meter per second | **28** | kPa | Kilopascal |
| **10** | °C | Degree Celsius | **29** | Bar | Bar |
| **11** | mV | Millivolt | **30** | psi | Pound per square inch |
| **12** | V | Volt | **36** | mg/l | Milligramme per Liter |
| **13** | mA | Milliampere | **37** | slpm | Standard Liter per Minute |

## 6.2 GfG table of gas types respectively measured quantities

| No. | Additivity Formula | Gas (Substance) | No. | Additivity Formula | Gas (Substance) |
|---|---|---|---|---|---|
| 1 | $C_3H_6O$ | Acetone | 51 | $C_6H_{14}$ | n-Hexane |
| 2 | $C_2H_3N$ | Acetonitrile | 52 | $C_6H_{14}$ | i-Hexane |
| 3 | $C_2H_2$ | Acetylene | 53 | $C_6H_{12}O$ | Hexanone |
| 4 | $C_3H_3N$ | Acrylnitrile | 54 | $C_6H_{12}O_2$ | Isobutyl acetate |
| 5 | $C_3H_9N$ | Aminopropane | 55 | $CO_2$ | Carbon dioxid |
| 6 | $NH_3$ | Ammonia | 56 | CO | Carbon monoxid |
| 7 | $C_5H_{12}O$ | Amyl alkohol | 57 | $H_2+CH_4+N_2+CO+....$ | Coke oven gas |
| 8 | Benzine | Benzin 60/95 | 58 | $N_2+O_2+CO_2+...$ | Air |
| 9 | Benzine | Benzin 80/110 | 59 | $CH_4$ | Methane |
| 10 | Gasoline | Benzin 100/140 | 60 | $CH_4O$ | Methanol |
| 11 | $C_6H_6$ | Benzene | 61 | $C_3H_6O_2$ | Methyl acetate |
| 12 | Cmb.gas | Combustible Gases and Vapors | 62 | $CH_3OH$ | Methyl alcohol |
| 13 | $CBrF_3$ | Bromtrifluormethane | 63 | $C_6H_{12}O$ | Butyl methyl ketone |
| 14 | $C_4H_6$ | 1,3-Butadiene | 64 | $CH_3CL$ | Methyl chloride |
| 15 | $C_4H_{10}$ | n-Butane | 65 | $CH_2CL_2$ | Methylene chloride |
| 16 | $C_4H_{10}$ | i-Butane | 66 | $C_6H_{12}O$ | MIBK Methyl isobutyl ketone |
| 17 | $C_4H_{10}O$ | Butanol (n) | 67 | $C_4H_8O$ | Ethyl methyl keton |
| 18 | $C_4H_8O$ | MEK (Methyl ethyl ketone) | 68 | $C_3H_8O_2$ | Methyl glycol |
| 19 | $C_6H_{12}O_2$ | Butyl acetat (n) | 69 | $C_5H_8O_2$ | Methyl methacrylate |
| 20 | $C_6H_{12}O_2$ | Butyl acetat (i) | 70 | $C_4H_{10}O$ | Methylpropanol |
| 21 | $C_4H_{10}O$ | Butyl alcohol (n) | 71 | $CBrCLF_2$ | Bromchlorodifluoromethan |
| 22 | $C_4H_8$ | Butene | 72 | $C_9H_{20}$ | n-Nonane |
| 23 | $CL_2$ | Chlorine | 73 | $C_8H_{18}$ | Octane (i) |
| 24 | $CH_3CL$ | Chlormethane | 74 | $C_8H_{18}$ | Octane (n) |
| 25 | HCL | Hydrogen Chloride | 75 | $C_5H_{12}$ | Pentane (i) |
| 26 | HCN | Hydrogen Cyanide | 76 | $C_5H_{12}$ | Pentane (n) |
| 27 | $C_6H_{12}$ | Cyclohexane | 77 | $C_5H_{10}O$ | Pentanone |
| 28 | $C_5H_{10}$ | Cyclopentane | 78 | $C_5H_{10}$ | Pentene |
| 29 | $C_3H_6$ | Cyclopropane | 79 | $C_7H_{14}O_2$ | Pentyl acetate |
| 30 | R12 | R12 - Dichlorodifluormethan | 80 | $C_2CL_4$ | PER (Perchlorethylene) |
| 31 | $C_2H_4CL_2$ | Dichlorethan | 81 | $C_3H_8$ | Propane |
| 32 | R21 | R21 - Dichlorodifluoromethane | 82 | $C_3H_8O$ | Propanol (i) |
| 33 | $CH_2CL_2$ | Dichloromethane | 83 | $C_5H_{10}O_2$ | Propyl acetate (i) |
| 34 | $C_3H_6CL_2$ | Dichloropropane | 84 | $C_5H_{10}O_2$ | Propyl acetate (n) |
| 35 | $C_4H_{11}N$ | Diethylamine | 85 | $C_3H_8O$ | Propyl alcohol (n) |
| 36 | $C_2H_6O$ | Dimethyl Ether | 86 | $C_3H_8O$ | Propyl alcohol (i) |
| 37 | $C_3H_5CLO$ | Epichlorhydrin | 87 | $C_3H_6$ | Propene |
| 38 | Nat.gas | Natural Gas | 88 | $C_3H_6CL_2$ | Propylene dichloride |
| 39 | $C_2H_6$ | Ethane | 89 | $O_2$ | Oxygen |
| 40 | $C_2H_6O$ | Ethanol | 90 | $SO_2$ | Sulphur dioxide |
| 41 | $C_4H_8O_2$ | Ethyl acetate | 91 | $SF_6$ | Sulphur hexafluoride |
| 42 | $C_2H_6O$ | Ethyl alcohol | 92 | $H_2S$ | Hydrogen sulfide |
| 43 | $C_2H_4$ | Ethylene | 93 | $H_2+CH_4+N_2+CO+...$ | Town gas |
| 44 | $C_2H_4O$ | Ethylene oxide | 94 | $NO_2$ | Nitrogen dioxide |
| 45 | Gemisch | FAM-Benzin 65/95 | 95 | NO | Nitrogen monoxide |
| 46 | Gemisch | Flugbenzin 40/180 | 96 | $C_8H_8$ | Styrene |
| 47 | $CH_2O$ | Formaldehyde | 97 | $C_2H_2CL_4$ | Tetrachloroethene |
| 48 | $CHCLF_2$ | R22 - Chlordifluoromethane | 98 | $C_7H_8$ | Toluol |
| 49 | He | Helium | 99 | $C_2H_3CL_3$ | Trichloroethane |
| 50 | $C_7H_{16}$ | n-Heptane | 100 | $C_2HCL_3$ | TRI - Trichloroethylene |

Continue on next page

| Nr. | Summen-Formel | Gas-Bezeichnung | Nr. | Summen-Formel | Gas-Bezeichnung |
|---|---|---|---|---|---|
| **101** | $CHF_3$ | R23 - Trifluormethan | **141** | Blend | Kerosene (180/220) |
| **102** | $C_4H_6O_2$ | Vinyl acetate | **142** | $CH_5N$ | Methylamine |
| **103** | $C_2H_3CL$ | Vinyl chloride | **143** | $SiCL_4$ | Silicon tetrachloride |
| **104** | $H_2$ | Hydrogen | **144** | $N_2$ | Nitrogen |
| **105** | $H_2$+CO+... | Watergas | **145** | $C_2H_3F_3$ | R143a - Trifluoroethane |
| **106** | $C_8H_{10}$ | Xylene | **146** | Blend | Diesel fuel |
| **107** | $O_3$ | Ozone | **147** | Blend ($C_2HF_5$ + $C_2H_3F_3$ + $C_2H_2F_4$) | R404a (Refrigerant blend R125+R143a+R134a) |
| **108** | $COCL_2$ | Phosgene | | | |
| **109** | $PH_3$ | Phosphine | **148** | $Br_2$ | Bromine |
| **110** | $SiH_4$ | Silane | **149** | VOC | VOC |
| **111** | $AsH_3$ | Arsine | **150** | PID | PID (Synonym for PID-Sensor) |
| **112** | $CLO_2$ | Chlorine dioxide | | | |
| **113** | $B_2H_6$ | Diborane | **151** | Blend ($C_2HF_5$ + $C_2H_3F_3$) | R507 (Refrigerant blend R125+R143a) |
| **114** | $C_2HCL_2F_3$ | R123 – Dichloro trifluoroethane | | | |
| **115** | $C_4H_{10}O$ | Diethyl ether | **152** | $C_3H_6O_2$ | ETF - Ethylformiat |
| **116** | $N_2O$ | Nitrous oxide | **153** | Ar | Argon |
| **117** | $C_2H_4O_2$ | Acetic acid | **154** | $C_2CL_3F_3$ | R113 -Trichlorotrifluoroethane |
| **118** | $F_2$ | Fluorine | **155** | $C_2H_3F_4$ | Refrigerant HFO-1234yf (2,3,3,3-Tetrafluorprop-1-en) |
| **119** | HF | Hydrogen fluoride | | | |
| **120** | $GeH_4$ | Hydrogen germane | **156** | Blend ($CH_2F_2$ + $C_2HF_5$ + $C_2H_2F_4$) | R407c (Refrigerant blend R32+R125+R134a) |
| **121** | $N_2H_4$ | Hydrazine | | | |
| **122** | $C_6H_6O$ | Phenol | **157** | Blend ($CH_2F_2$ + $C_2HF_5$) | R410a (Refrigerant blend R32+R125) |
| **123** | $C_3H_6O$ | Propylene oxide | | | |
| **124** | $C_2H_2F_4$ | R134a - Tetrafluoroethane | **158** | $NF_3$ | Nitrogen trifluoride |
| **125** | $C_4H_8S$ | THT - Tetrahydrothiophene | **159** | pH | pH |
| **126** | TAL | ToxAlert | **160** | Redox | Redox |
| **127** | $C_4H_5F_5$ | R365 - Pentafluorbutane | **161** | $C_4H_{10}S$ | TBM (tert-Butyl mercaptan) |
| **128** | $C_5H_{10}O_3$ | Ethyl lactate | **162** | HBr | Hydrogen bromide |
| **129** | $NH_4$+ | Ammonium | | | |
| **130** | $CCL_3F$ | R11 - Trichlorofluoromethane | | | |
| **131** | $C_3H_3F_5$ | R245fa - Pentafluoropropane | | | |
| **132** | $C_3H_4$ | Propyne | | | |
| **133** | $CS_2$ | Carbon disulfide | **225** | Signal | Signal (universal) |
| **134** | $BCL_3$ | Boron trichloride | **226** | Q | Volume flow |
| **135** | $BF_3$ | Boron trifluoride | **227** | P | Pressure |
| **136** | $CH_3Br$ | Bromomethane | **228** | m | Mass |
| **137** | $C_4H_{10}O$ | 2-Butanol | **229** | Wdir | Wind direction |
| **138** | $CH_4$+$CO_2$ | Landfill gas | **253** | T | Temperature |
| **139** | $C_2H_4F_2$ | R152a - Difluoroethane | **254** | Vair | Wind velocity |
| **140** | $C_4H_8O_2$ | 1,4-Dioxane | **255** | rH | Relative humidity |

## 7. References


[MAPS] MODBUS Application Protocol Specification V1.1;
*http://www.modbus.org*

[MoSL] MODBUS over Serial Line – Specification & Implementation Guide V1.02;
*http://www.modbus.org*





GfG Gesellschaft für Gerätebau mbH
Klönnestraße 99, D-44143 Dortmund
Phone: +49 / (0)231/56400 0
Fax: +49 / (0)231/516313
E-Mail: info@gasmessung.de
Internet: www.gfg.biz