MV Electrical network management

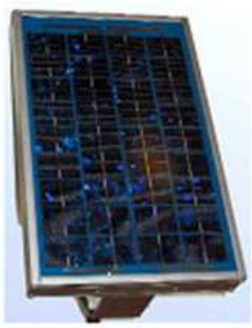
Easergy Range

# FLITE116 – G200

Wireless Communicating indicator

## Modbus communication

Appendix to the User's manual




Schneider Electric

# Contents

# General

## Functionality

**Monitoring**

- Network faults (di/dt or IMAX)
- Voltage losses/ returns
- Flite Low battery and communication failures

**Measurements**

- Phase currents(I_min, I_max, I_mean and I_inst)
- MV presence statistics

**Remote control**

- Flite parameters
- G200 parameters
- Long range communication
- Storage and alarm information

## Characteristics

| | |
|---|---|
| type of transmission | asynchronous serial |
| protocol | Standard Modbus slave or Modbus TCP |
| speed | 300, 600, 1200, 2400, 4800, 9600, 19200 baud |
| data format | 1 start bit, 8 data bits with no parity, 1 stop bit |
| electrical interface | RS232, GSM or GPRS |

## Specific Modbus TCP features for GPRS solar version

G200 GPRS can be used as any MODBUS TCP slave but an enhanced specification has been added to solve two major issues:

- Modbus TCP is not compliant with a solar power supply due to the fact that the connection is permanent.
- In many cases, GPRS access does not provide a static IP to the RTU. So the SCADA can not establish the connection.

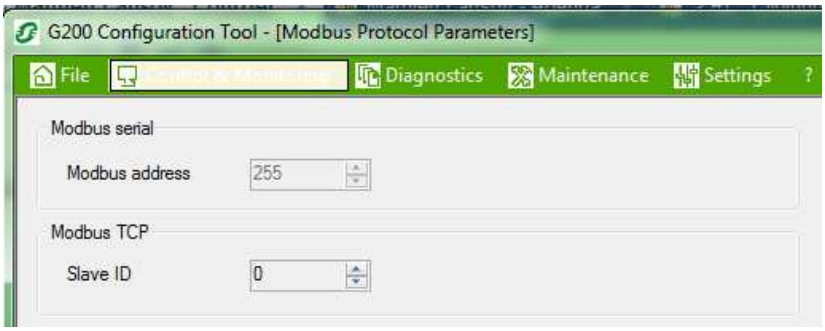
Consequently, a **non-permanent mode** has been developed. In this mode, Scada system and G200 can be both TCP client and server. It means that the G200 is establishing a connection on alarm. The SCADA system can also establish a connection on demand but in both case TCP link is never maintained.

For dynamic IP, each time the G200 is changing its IP address, an alarm can be activated and the G200 is establishing a connection to provide its new IP address to the SCADA..

This mode has been implemented in the L500 system.

## Protocol settings

**MODBUS SERIAL**




**Modbus Address:**

- Source address used by the remote SCADA to identify the G200 through the protocol.
- Adjustable from 1 to 255 in GSM and RS232 version.
- 1 is the default modbus address value in GSM and RS232 version. In GPRS version, the value is fixed to 255.

**MODBUS TCP**

**Slave ID:**

The slave Id is used when the G200 is working in **non-permanent mode** with **dynamic IP**. This mode is implemented in the L500 SCADA to identify the G200 on incoming TCP connection.

The Slave Id is read by the SCADA using the Identification frame (see below).

- The slave Id must be different for every G200 equipment and must be set in the L500 system as well.
- Value is from 0 to 65534.

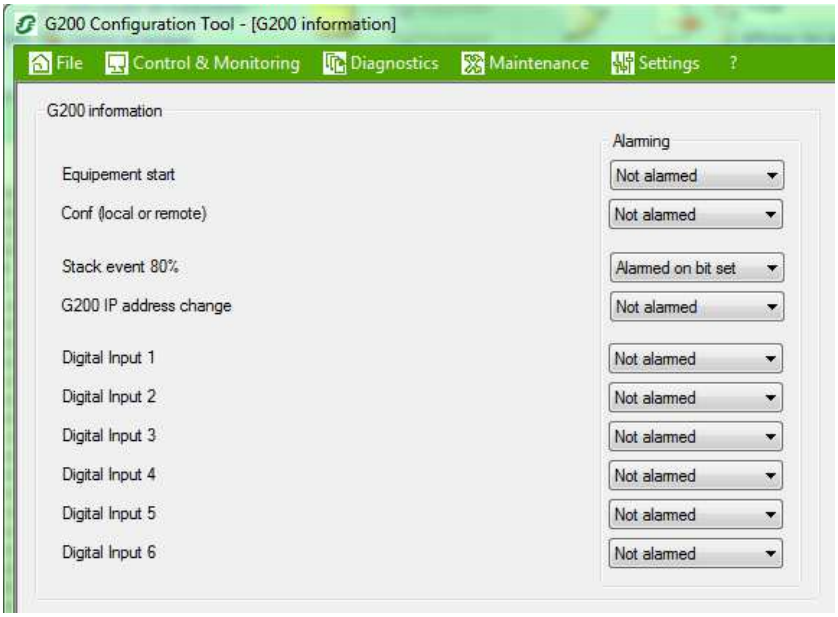
## GPRS version

### G200 INFORMATION

***G200 IP address change:*** **(GPRS version only):**

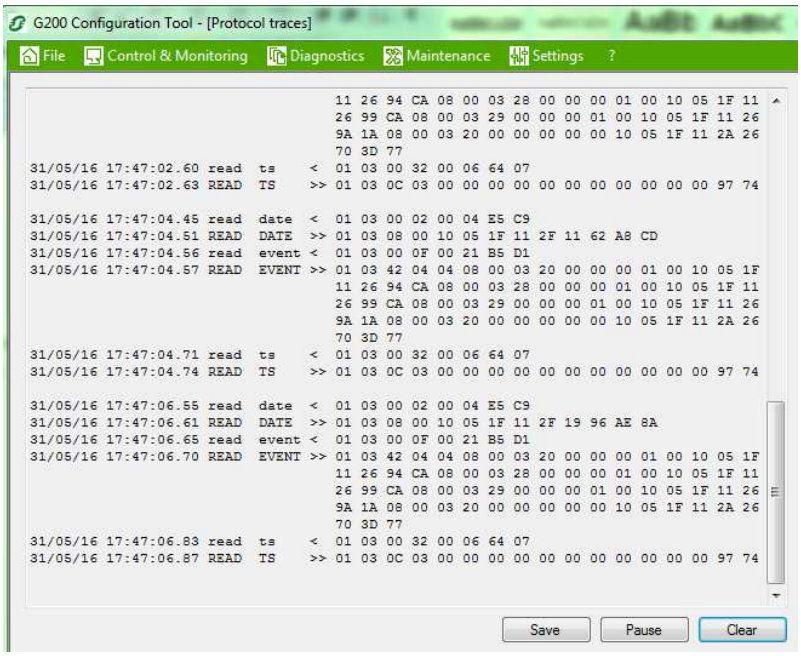
- Alarm on G200 IP address change
- Only *Alarmed* and *Not alarmed* can be selected.
- If *Alarmed* is selected, when G200 IP address changes, G200 calls the SCADA.
- This feature must be activated when:
  - G200 IP **address** is **dynamic**
  - And when G200 is connected to a **L500 SCADA**
  - With this feature, L500 is able to refresh G200 IP address

If this option is activated, it is not necessary to configure a call once a day. It will be done at modem reset.



## MODBUS analyser

G200 provides a protocol analyser (with a Modbus frame specific decoder). This feature is accessible from the *Protocol traces...* menu on the PC connected to the configuration plug.




**Display:**

- The first column gives the time and date of the message in minutes, seconds and 100ths of seconds.

- The second column indicates the type of frame. Upper case characters are used for frames transmitted by the G200. This is confirmed by the double chevron '>>' in column 3. On the other hand, all the lower case characters pertain to frames received by the remote control station (confirmed by a single chevron '<' in column 3).

- The last column displays the frame in hexadecimal form. The "+" and ' * ' signs may precede the display of the frame:

  - The '+' sign indicates frames not intended for the equipment,
  - The ' * ' sign indicates an erroneous frame (incomplete frame, faulty construction,...).

# MODBUS data addresses and encoding

## General

**Addressing with RS232 and GSM version**

A MODBUS master can access 255 storage spaces of 64K words (255 MODBUS addresses).

- Addressing of G200 range is limited to 255 equipments
- To increase the addressing capability, the MODBUS master may use the test zone to identify more accurately the equipment.
- Mechanism of the data encoding: data are split to a non-permanent communication. But a permanent communication (direct connection) with a continuous scrolling of the slave is more simple, so a lot of things aren't useful :

  - report by exception

  - events

  - alarms

  - parameters

- test zone

**Addressing with GPRS version**

Scada and RTU can be both TCP client and server.

To identify a RTU, a scada may use its IP address (if the network provided fix IP address for the G200) or its Slave ID if the IP address is dynamic..

In the case there is no fix IP address for the equipment on the network, the scada must be listening on an opened port on which the RTU can open a TCP connection. To identify the RTU, the scada automatically sends an identification frame on connection. In case of IP address change, the equipment can be set to call the Scada so that it can call back afterwards.

### Reply messages

■ Upon receipt of a request recognised by the equipment (read or write), transmission of the data corresponding to the MODBUS specifications.

■ Upon receipt of a request not recognised by the equipment, transmission of an exception message (type 1, 2 or 3 only).

### Read zone

■ The number of words read may not exceed the size of the checked zone.

■ Some zones may only be accessed as a whole.

### Notes

■ The bit by bit write and read functions are not used in the G200 application.

■ Values followed by the letter "h" are in hexadecimal form (e.g. 0003h).

■ In the charts describing the data exchanged between the master and the G200, the hatched strips in the "authorised function" columns indicate the zones that are accessible as a whole.

### Terminology

■ TCD: remote control (digital output encoded in 2 bits)

■ TSS: single-state remote indication (digital input encoded in 1 bit)

■ TM: remote metering (analog input encoded in 16 bits)

## Identification / configuration zone

| | word address 0000h to 0001h | access mode | authorised function |
|---|---|---|---|
| Software version | 0000h | read | 3,4 |
| Status | 0001h | read | 3,4 |

■ **Bit 0 to 7 of status indicates the type of the equipment.** (read only)

= 104 decimal (68h) for G200 Modbus GSM v2.00
= 105 decimal (69h) for G200 Modbus GPRS v1.00

■ **Bit 15 of status indicates:**

■ 0 = No events loss

■ 1 = Loss of events

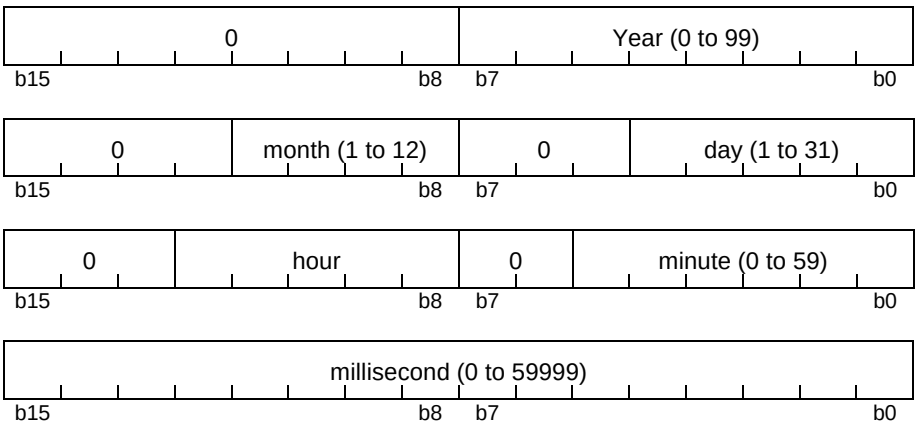
This bit is set when the modbus event file is full. The event ''event loss'' is then placed in the file. As long as this event is in the file, no other event can be memorised. This bit is reset when the file is half empty. This change of state doesn't generate an event.

## Time synchronisation zone

This zone contains the internal date and time of the equipment for time stamping of events.

The zone may only be read or written as a whole.

| binary date | word address 0002h to 0005h | access mode | authorised function |
|---|---|---|---|
| year | 0002h | read/write | 3,4,16 |
| month+day | 0003h | read/write | 3,4 |
| hours+minutes | 0004h | read/write | 3,4 |
| milliseconds | 0005h | read/write | 3,4 |



## Test zone

The test zone contains 9 words that can be read or written. It is recorded in saved RAM and is available to users to facilitate final adjustment tests or to record an identification of the equipment.

The content of the zone does not have any effect on the G200 functions.

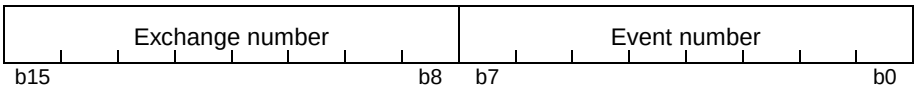
| Test zone | word address | access mode | authorised function |
|---|---|---|---|
| 9 words | 0006h to 000Eh | read/write | 1,2,3,4,5,6,16 |

## Event zone

This zone contains the time stamp events.

| Event zone | word address | access mode | authorised function |
|---|---|---|---|
| exchange word | 000Fh | read/write | 3,4,6,16 |
| event 1 | 0010h to 0017h | read | 3,4 |
| event 2 | 0018h to 001Fh | read | 3,4 |
| event 3 | 0020h to 0027h | read | 3,4 |
| event 4 | 0028h to 002Fh | read | 3,4 |

Exchange word format:




Only the exchange word may be written.
It is possible to read the exchange zone as a whole or the exchange word only.

The exchange word is used to manage a specific protocol to be sure not to lose events as a result of a MODBUS communication problem. The event table is numbered for that purpose.

The exchange word consists of 2 bytes:

- Most significant byte = exchange number which identifies each event frame. It is preset to zero when the G200 is switched on. When it reaches its maximum value (FFh), it automatically goes back to 0. The G200 numbers the exchanges and the master acknowledges the numbering.

- Least significant byte = number of valid events in the event zone (maximum 4).

**Encoding of events**

Each event is encoded with 4 words related to the event, followed by 4 words containing the event time-stamping data:

| | |
|---|---|
| ■ word 1: identification word | |
| 0800h /2048<br>binary event | 0400h/1024<br>16 bits analogical event |
| ■ word 2: event address | |
| □ 001Fh /31:<br>Event loss bit ( set only on appearance)<br>Digital event bit address<br>□ 0320h to 037Fh:<br>TSS 1 to 96 | Analogical event word address<br>□ 40h to 75h:<br>TM 1 to 54 |
| ■ word 3: event value MSW | |
| 0 | 0 |
| ■ word 4: event value LSW | |
| 0 = 1 to 0<br>change of state<br>1 = 0 to 1<br>change of state | 16 bits analogical event value |
| ■ words 5 to 8:<br>time-stamping with same format as date zone. | |

**Acknowledgement of events**

To inform the G200 that it has correctly received the frame it has read, the master must :

☐ write the number of the last exchange it has received in the "exchange number" byte

☐ reset the "number of events" byte of the exchange word to zero.

After acknowledgement, the G200 erases the events that have already been transmitted and replaces them by new ones when applicable.

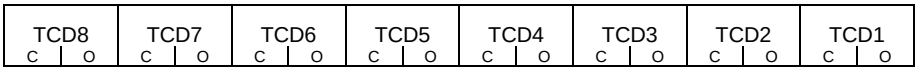
Note: until the exchange word written by the master becomes "X,0" (with X = number of the previous exchange that the master wishes to acknowledge), the exchange word in the table remains at "X, number of previous events".

If the number is equal to zero, the master is not required to acknowledge a message with no event.

## TCD / TSS zone

| TCD / TSS | word address | access mode | function authorised |
|---|---|---|---|
| TCD 1-8 | 0030h | write | 5,6 |
| CR | 0031h | read | 1,2,3,4,5,6 |
| TSS 1-16 | 0032h | read | 1,2,3,4 |
| TSS 17-32 | 0033h | read | 1,2,3,4 |
| TSS 33-48 | 0034h | read | 1,2,3,4 |
| TSS 49-64 | 0035h | read | 1,2,3,4 |
| TSS 65-80 | 0036h | read | 1,2,3,4 |
| TSS 81-96 | 0037h | read | 1,2,3,4 |

**Each TCD (Double Digital Output) word is encoded as follows:**




| TCD | Single remote indications | Word bit |
|---|---|---|
| 1 | Get DATA (wake-up FLITEs to have fresh data) | 30h 0-1 |

**A remote control** TCD is encoded in 2 bits:

- 01 = open order
- 10 = closing order

The TCDs are assigned as follows:

- TCD1..8 : control 1..8.

- Here, only 1 TCD is used

Writing a TCD word performs remote control orders. Only one remote control order at a time may be requested.

The control order zone (TCD) may be read with bit and word read function code. As it contains no information the data is 0.

**The CR code** (result code) gives information on the processing of the remote control order carried out by the G 200:

- bit 0: Remote control in progress.

- bit 1: Fault concerning the initial remote control order

- bit 2: Serious fault detected during internal check.

- bit 3: not used.

- bit 4: not used.

- bit 5: Failure to execute for an unknown reason.

The control center system may reset this code by writing a 0 to the relevant address.

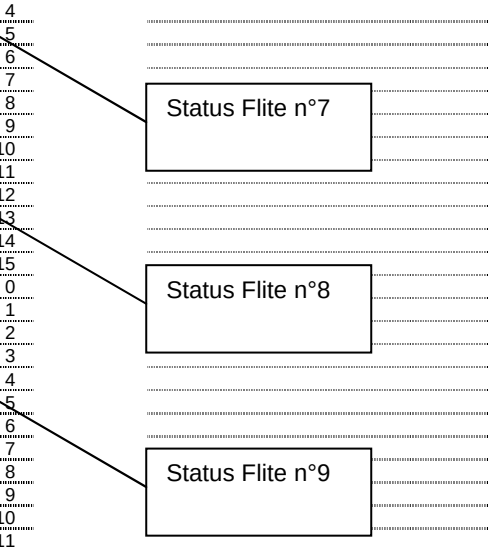
**Each TSS (Simple Digital Input) word is encoded as follows:**

| TSS16 | TSS15 | TSS14 | TSS13 | TSS12 | TSS11 | TSS10 | TSS9 | TSS8 | TSS7 | TSS6 | TSS5 | TSS4 | TSS3 | TSS2 | TSS1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| b15 | | | | | | | b8 | b7 | | | | | | | b0 |

| TSS | Single remote indications | Word bit |
|---|---|---|
| 1 | Equipment start | 32h 0 |
| 2 | Configuration | 32h 1 |
| 3 | Modbus event stack 80% | 32h 2 |
| 4 | Reserved | 32h 3 |
| 5 | Reserved | 32h 4 |
| 6 | Reserved | 32h 5 |
| 7 | Reserved | 32h 6 |
| 8 | Reserved | 32h 7 |
| 9 | DI 1 Digital input 1 | 32h 8 |
| 10 | DI 2 Digital input 2 | 32h 9 |
| 11 | DI 3 Digital input 3 | 32h 10 |
| 12 | DI 4 Digital input 4 | 32h 11 |
| 13 | DI 5 Digital input 5 | 32h 12 |
| 14 | DI 6 Digital input 6 | 32h 13 |
| 15 | Reserved | 32h 14 |
| 16 | Reserved | 32h 15 |
| 17 | Fault dI/dt - ind. 1 | 33h 0 |
| 18 | Fault Imax - ind. 1 | 33h 1 |
| 19 | Battery fault - ind. 1 | 33h 2 |
| 20 | Volt. Presence - ind. 1 | 33h 3 |
| 21 | Comm. Fault - ind. 1 | 33h 4 |
| 22 | Flite presence - ind. 1 | 33h 5 |
| 23 | Config in progress - ind. 1 | 33h 6 |
| 24 | Config fault - ind. 1 | 33h 7 |
| 25 | Fault dI/dt - ind. 2 | 33h 8 |
| 26 | Fault Imax - ind. 2 | 33h 9 |
| 27 | Battery fault - ind. 2 | 33h 10 |
| 28 | Volt. Presence - ind. 2 | 33h 11 |
| 29 | Comm. Fault - ind. 2 | 33h 12 |
| 30 | Flite presence - ind. 2 | 33h 13 |
| 31 | Config in progress - ind. 2 | 33h 14 |
| 32 | Config fault - ind. 2 | 33h 15 |
| 33 | Fault dI/dt - ind. 3 | 34h 0 |
| 34 | Fault Imax - ind. 3 | 34h 1 |
| 35 | Battery fault - ind. 3 | 34h 2 |
| 36 | Volt. Presence - ind. 3 | 34h 3 |
| 37 | Comm. Fault - ind. 3 | 34h 4 |
| 38 | Flite presence - ind. 3 | 34h 5 |
| 39 | Config in progress - ind. 3 | 34h 6 |
| 40 | Config fault - ind. 3 | 34h 7 |
| 41 | Fault dI/dt - ind. 4 | 34h 8 |
| 42 | Fault Imax - ind. 4 | 34h 9 |
| 43 | Battery fault - ind. 4 | 34h 10 |
| 44 | Volt. Presence - ind. 4 | 34h 11 |
| 45 | Comm. Fault - ind. 4 | 34h 12 |
| 46 | Flite presence - ind. 4 | 34h 13 |
| 47 | Config in progress - ind. 4 | 34h 14 |
| 48 | Config fault - ind. 4 | 34h 15 |

| TSS | Single remote indications | Word bit |
|---|---|---|
| 49 | Fault dI/dt - ind. 5 | 35h 0 |
| 50 | Fault Imax - ind. 5 | 35h 1 |
| 51 | Battery fault - ind. 5 | 35h 2 |
| 52 | Volt. Presence - ind. 5 | 35h 3 |
| 53 | Comm. Fault - ind. 5 | 35h 4 |
| 54 | Flite presence - ind. 5 | 35h 5 |
| 55 | Config in progress - ind. 5 | 35h 6 |
| 56 | Config fault - ind. 5 | 35h 7 |
| 57 | Fault dI/dt - ind. 6 | 35h 8 |
| 58 | Fault Imax - ind. 6 | 35h 9 |
| 59 | Battery fault - ind. 6 | 35h 10 |
| 60 | Volt. Presence - ind. 6 | 35h 11 |
| 61 | Comm. Fault - ind. 6 | 35h 12 |
| 62 | Flite presence - ind. 6 | 35h 13 |
| 63 | Config in progress - ind. 6 | 35h 14 |
| 64 | Config fault - ind. 6 | 35h 15 |
| 65 | Fault dI/dt - ind. 7 | 36h 0 |
| 66 | Fault Imax - ind. 7 | 36h 1 |
| 67 | Battery fault - ind. 7 | 36h 2 |
| 68 | Volt. Presence - ind. 7 | 36h 3 |
| 69 | Comm. Fault - ind. 7 | 36h 4 |
| 70 | Flite presence - ind. 7 | 36h 5 |
| 71 | Config in progress - ind. 7 | 36h 6 |
| 72 | Config fault - ind. 7 | 36h 7 |
| 73 | Fault dI/dt - ind. 8 | 36h 8 |
| 74 | Fault Imax - ind. 8 | 36h 9 |
| 75 | Battery fault - ind. 8 | 36h 10 |
| 76 | Volt. Presence - ind. 8 | 36h 11 |
| 77 | Comm. Fault - ind. 8 | 36h 12 |
| 78 | Flite presence - ind. 8 | 36h 13 |
| 79 | Config in progress - ind. 8 | 36h 14 |
| 80 | Config fault - ind. 8 | 36h 15 |
| 81 | Fault dI/dt - ind. 9 | 37h 0 |
| 82 | Fault Imax - ind. 9 | 37h 1 |
| 83 | Battery fault - ind. 9 | 37h 2 |
| 84 | Volt. Presence - ind. 9 | 37h 3 |
| 85 | Comm. Fault - ind. 9 | 37h 4 |
| 86 | Flite presence - ind. 9 | 37h 5 |
| 87 | Config in progress - ind. 9 | 37h 6 |
| 88 | Config fault - ind. 9 | 37h 7 |
| 89 | Reserved | 37h 8 |
| 90 | Reserved | 37h 9 |
| 91 | Reserved | 37h 10 |
| 92 | Reserved | 37h 11 |
| 93 | Reserved | 37h 12 |
| 94 | Reserved | 37h 13 |
| 95 | Reserved | 37h 14 |
| 96 | Reserved | 37h 15 |

| TSS | Single remote indications | Word bit |
|---|---|---|
| 97 | Reserved | 38h 0 |
| 98 | Reserved | 38h 1 |
| 99 | Reserved | 38h 2 |
| 100 | Reserved | 38h 3 |
| 101 | Reserved | 38h 4 |
| 102 | Reserved | 38h 5 |
| 103 | Reserved | 38h 6 |
| 104 | Reserved | 38h 7 |
| 105 | Reserved | 38h 8 |
| 106 | Reserved | 38h 9 |
| 107 | Reserved | 38h 10 |
| 108 | Reserved | 38h 11 |
| 109 | Reserved | 38h 12 |
| 110 | Reserved | 38h 13 |
| 111 | Reserved | 38h 14 |
| 112 | Reserved | 38h 15 |



## Remote metering zone

| Metering | Word address | | access mode | function authorised |
|---|---|---|---|---|
| | Hex. | decimal | | |
| I_mean - F1 | 0040h | 64 | read | 3,4 |
| I_min – F1 | 0041h | 65 | read | 3,4 |
| I_max – F1 | 0042h | 66 | read | 3,4 |
| Voltage pres. – F1 | 0043h | 67 | read | 3,4 |
| Comms count. - F1 | 0044h | 68 | read | 3,4 |
| I_inst – F1 | 0045h | 69 | read | 3,4 |
| I_mean – F2 | 0046h | 70 | read | 3,4 |
| I_min – F2 | 0047h | 71 | read | 3,4 |
| I_max – F2 | 0048h | 72 | read | 3,4 |
| Voltage pres. – F2 | 0049h | 73 | read | 3,4 |
| Comms count. – F2 | 004Ah | 74 | read | 3,4 |
| I_inst – F2 | 004Bh | 75 | read | 3,4 |
| I_mean – F3 | 004Ch | 76 | read | 3,4 |
| I_min – F3 | 004Dh | 77 | read | 3,4 |
| I_max – F3 | 004Eh | 78 | read | 3,4 |
| Voltage pres. – F3 | 004Fh | 79 | read | 3,4 |
| Comms count. – F3 | 0050h | 80 | read | 3,4 |
| I_inst – F3 | 0051h | 81 | read | 3,4 |
| I_mean – F4 | 0052h | 82 | read | 3,4 |
| I_min – F4 | 0053h | 83 | read | 3,4 |
| I_max – F4 | 0054h | 84 | read | 3,4 |
| Voltage pres. – F4 | 0055h | 85 | read | 3,4 |
| Comms count. – F4 | 0056h | 86 | read | 3,4 |
| I_inst – F4 | 0057h | 87 | read | 3,4 |
| I_mean – F5 | 0058h | 88 | read | 3,4 |
| I_min – F5 | 0059h | 89 | read | 3,4 |
| I_max – F5 | 005Ah | 90 | read | 3,4 |
| Voltage pres. – F5 | 005Bh | 91 | read | 3,4 |
| Comms count. – F5 | 005Ch | 92 | read | 3,4 |
| I_inst – F5 | 005Dh | 93 | read | 3,4 |
| I_mean – F6 | 005Eh | 94 | read | 3,4 |
| I_min – F6 | 005Fh | 95 | read | 3,4 |
| I_max – F6 | 0060h | 96 | read | 3,4 |
| Voltage pres. – F6 | 0061h | 97 | read | 3,4 |
| Comms count. – F6 | 0062h | 98 | read | 3,4 |
| I_inst – F6 | 0063h | 99 | read | 3,4 |
| I_mean – F7 | 0064h | 100 | read | 3,4 |
| I_min – F7 | 0065h | 101 | read | 3,4 |
| I_max – F7 | 0066h | 102 | read | 3,4 |
| Voltage pres. – F7 | 0067h | 103 | read | 3,4 |
| Comms count. – F7 | 0068h | 104 | read | 3,4 |
| I_inst – F7 | 0069h | 105 | read | 3,4 |
| I_mean – F8 | 006Ah | 106 | read | 3,4 |
| I_min – F8 | 006Bh | 107 | read | 3,4 |
| I_max – F8 | 006Ch | 108 | read | 3,4 |
| Voltage pres. – F8 | 006Dh | 109 | read | 3,4 |
| Comms count. – F8 | 006Eh | 110 | read | 3,4 |
| I_inst – F8 | 006Fh | 111 | read | 3,4 |
| I_mean – F9 | 0070h | 112 | read | 3,4 |
| I_min – F9 | 0071h | 113 | read | 3,4 |
| I_max – F9 | 0072h | 114 | read | 3,4 |
| Voltage pres. – F9 | 0073h | 115 | read | 3,4 |
| Comms count. – F9 | 0074h | 116 | read | 3,4 |
| I_inst – F9 | 0075h | 117 | read | 3,4 |
| Reserved | 0078h to 008Fh | 118 to 143 | read | 3,4 |

- Each TM (or Analog Input) value is a signed value encoded in 2's complement 16-bit word.
- 0x8000 stands for non valid value.

## Parameters zone

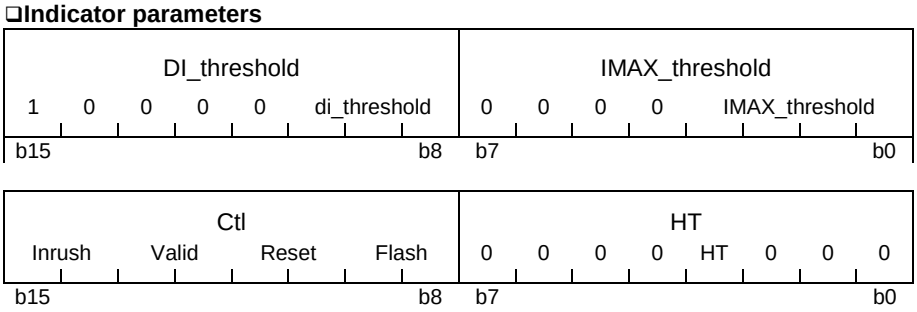
| Parameters | Word address | | access mode | function authorised |
|---|---|---|---|---|
| | Hex. | Decimal | | |
| Storage information | 0090h to 0091h | 144 to 145 | Read/write | 1,2,3,4,5,6 |
| Alarm information | 0092h to 0094h | 146 to 148 | Read/write | 1,2,3,4,5,6 |
| Primary Host phone number | 0095h to 0098h | 149 to 152 | Read/write | 1,2,3,4,5,6 |
| Standby host phone number (standby) | 0099h to 009Ch | 153 to 156 | Read/write | 1,2,3,4,5,6 |
| SMS service center phone number | 009Dh to 00A0h | 157 to 160 | Read/write | 1,2,3,4,5,6 |
| SMS user phone number | 00A1h to 00A4h | 161 to 164 | Read/write | 1,2,3,4,5,6 |
| Cyclic dial up period | 00A5h to 00A6h | 165 to 166 | Read/write | 1,2,3,4,5,6 |
| G200 IP address | 00A7h to 00A8h | 167 to 168 | Read | 1,2,3,4,5,6 |
| SCADA IP address | 00A9h to 00AA.h | 169 to 170 | Read/write | 1,2,3,4,5,6 |
| G200 local port | 00ABh | 171 | Read/write | 1,2,3,4,5,6 |
| SCADA remote port | 00ACh | 172 | Read/write | 1,2,3,4,5,6 |
| Reserved | 00ACh to 00AFh | 173 to 175 | | |
| Measurement period | 00B0h | 176 | Read/write | 1,2,3,4,5,6 |
| Current deadband (%) | 00B1h | 177 | Read/write | 1,2,3,4,5,6 |
| Minimum current variation | 00B2h | 178 | Read/write | 1,2,3,4,5,6 |
| Voltage deadband | 00B3h | 179 | Read/write | 1,2,3,4,5,6 |
| Communication fault counter threshold | 00B4h | 180 | Read/write | 1,2,3,4,5,6 |
| Config. Indicator F1 | 00B5h to 00B6h | 181 to 182 | Read/write | 1,2,3,4,5,6 |
| Config. Indicator F2 | 00B7h to 00B8h | 183 to 184 | Read/write | 1,2,3,4,5,6 |
| Config. Indicator F3 | 00B9h to 00BAh | 185 to 186 | Read/write | 1,2,3,4,5,6 |
| Config. Indicator F4 | 00BBh to 00BCh | 187 to 188 | Read/write | 1,2,3,4,5,6 |
| Config. Indicator F5 | 00BDh to 00BEh | 189 to 190 | Read/write | 1,2,3,4,5,6 |
| Config. Indicator F6 | 00BFh to 00C0h | 191 to 192 | Read/write | 1,2,3,4,5,6 |
| Config. Indicator F7 | 00C1h to 00C2h | 193 to 194 | Read/write | 1,2,3,4,5,6 |
| Config. Indicator F8 | 00C3h to 00C4h | 195 to 196 | Read/write | 1,2,3,4,5,6 |
| Config. Indicator F9 | 00C5h to 00C6h | 197 to 198 | Read/write | 1,2,3,4,5,6 |
| Reserved | 00C7h to CFh | 199 to 207 | Read/write | 1,2,3,4,5,6 |

# MODBUS data addresses and encoding

**WARNING**

**These two words must be written at the same time to avoid unnecessary communications with flites**

**❑Indicator parameters**




| | |
|---|---|
| DI_threshold | User-selected di/dt threshold value |
| IMAX_threshold | User-selected IMAX threshold value |
| Ctl | This control Word is used to configure following parameters : inrush time-out, fault confirmation per voltage absence, automatic voltage reset and flash time-out |
| HT | Electrical field threshold above which the MV voltage is detected. |

Recommended values are in bold:

| di_threshold | di/dt value (for 50Hz networks) |
|---|---|
| 1xxx0000 | 6 A / 30 ms |
| 1xxx0001 | 12 A / 30 ms |
| 1xxx0010 | 24 A / 30 ms |
| 1xxx0011 | 30 A / 30 ms |
| 1xxx0100 | 40 A / 30 ms |
| **1xxx0101** | **60 A / 30 ms** |
| 1xxx0110 | 80 A / 30 ms |
| 1xxx0111 | OFF |

| IMAX_Threshold | IMAX value |
|---|---|
| XXXX0000 | 800 A |
| XXXX0001 | 100 A |
| XXXX0010 | 150 A |
| XXXX0011 | 200 A |
| XXXX0100 | 250 A |
| XXXX0101 | 300 A |
| XXXX0110 | 400 A |
| **XXXX0111** | **500 A** |
| XXXX1000 | 600 A |

| Ctl | Description |
|---|---|
| 00XXXXXX | No Inrush |
| **01XXXXXX** | **Inrush : 3s** |
| 10XXXXXX | Inrush : 30 s |
| 11XXXXXX | Inrush : 60 s |
| XX00XXXX | No validation |
| **XX01XXXX** | **Validation** |
| XX10XXXX | Not used |
| XX11XXXX | Not used |
| XXXX00XX | No auto. Reset |
| **XXXX01XX** | **auto. reset 3s** |
| XXXX10XX | auto. reset 30s |
| XXXX11XX | auto. reset 60s |
| **XXXXXX00** | **2 h flash time** |
| XXXXXX01 | 4 h flash time |
| XXXXXX10 | 8 h flash time |
| XXXXXX11 | 16 h flash time |

| HT | voltage presence |
|---|---|
| **XXXX0XXX** | **A** |
| XXXX1XXX | B |

**Note:** for 60 Hz networks, dt becomes 25ms

❑ **An TM event is stored in modbus stack upon 1 bit:**
When the bit is set to <0>, a change is not added in the event stack. When set to <1>, it is added.

When modbus event stack overflows, no more event is stored in the stack until it reaches half emptiness.

| Storage | Single remote indications | Word | bit |
|---|---|---|---|
| 1 | I_average | 90h | 0 |
| 2 | I_min | 90h | 1 |
| 3 | I_max | 90h | 2 |
| 4 | Voltage Presence(kV/m) | 90h | 3 |
| 5 | Reserved | 90h | 4 |
| 6 | Reserved | 90h | 5 |
| 7 | Reserved | 90h | 6 |
| 8 | Reserved | 90h | 7 |
| 9 | Reserved | 90h | 8 |
| 10 | Reserved | 90h | 9 |
| 11 | Reserved | 90h | 10 |
| 12 | Reserved | 90h | 11 |
| 13 | Reserved | 90h | 12 |
| 14 | Reserved | 90h | 13 |
| 15 | Reserved | 90h | 14 |
| 16 | Reserved | 90h | 15 |
| 1 | Reserved | 91h | 0 |
| 2 | Reserved | 91h | 1 |
| 3 | Reserved | 91h | 2 |
| 4 | Reserved | 91h | 3 |
| 5 | Reserved | 91h | 4 |
| 6 | Reserved | 91h | 5 |
| 7 | Reserved | 91h | 6 |
| 8 | Reserved | 91h | 7 |
| 9 | Reserved | 91h | 8 |
| 10 | Reserved | 91h | 9 |
| 11 | Reserved | 91h | 10 |
| 12 | Reserved | 91h | 11 |
| 13 | Reserved | 91h | 12 |
| 14 | Reserved | 91h | 13 |
| 15 | Reserved | 91h | 14 |
| 16 | Reserved | 91h | 15 |

❑ **Alarm information:**

This enables the user to choose whether a change of state creates an alarm or not, for each type of information:

- 00 = Alarm is not used
- 01 = Alarmed on bit set
- 10 = Alarmed on bit clear
- 11 = Alarmed on bit set and clear

| Alarm | Single remote indications | Word | bit |
|---|---|---|---|
| 1 | di/dt Fault | 92h | 0-1 |
| 2 | IMAX Fault | 92h | 2-3 |
| 3 | Battery check | 92h | 4-5 |
| 4 | Voltage Presence | 92h | 6-7 |
| 5 | Comm. Fault | 92h | 8-9 |
| 6 | Reserved | 92h | 10-11 |
| 7 | Configuration in progress | 92h | 12-13 |
| 8 | Configuration fault | 92h | 14-15 |
| 9 | Equipment Start | 93h | 0-1 |
| 10 | Configuration | 93h | 2-3 |
| 11 | Modbus event stack 80% | 93h | 4-5 |
| 12 | Reserved | 93h | 6-7 |
| 13 | Alarm message set up | 93h | 8-9 |
| 14 | SMS message system enabled | 93h | 10-11 |
| 15 | Test alarm | 93h | 12-13 |
| 16 | Reserved | 93h | 14-15 |
| 1 | Digital Input 1 | 94h | 0-1 |
| 2 | Digital Input 2 | 94h | 2-3 |
| 3 | Digital Input 3 | 94h | 4-5 |
| 4 | Digital Input 4 | 94h | 6-7 |
| 5 | Digital Input 5 | 94h | 8-9 |
| 6 | Digital Input 6 | 94h | 10-11 |
| 7 | Reserved | 94h | 12-13 |
| 8 | Reserved | 94h | 14-15 |

Except for Alarm 13, 14,15,16 , when bits are set to ***00,*** the information is not alarmed. When set to ***01***, it is alarmed on bit set. When set to ***10***, it is alarmed on bit reset. When set to ***11***, it is alarmed on both bit set and bit reset.

For 13, 14 and 15 and 16, values are restricted to 00(not alarmed) and 11(alarmed) values.

■ **Test alarm**: A bit is used to test the alarm mechanism : if bits is written with "11" by the master MODBUS, an alarm signal will set off one minute later. The bit will then be set to "00" by the G200 if the alarm is acquitted.

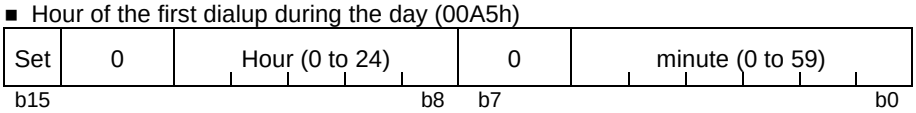
■ **Alarm message set up:** bits are used to set up the alarm mechanism: if bits are written with "11" by the master MODBUS, The alarm mechanism is set up. If bits are written with "00" by the master MODBUS, no alarm neither cyclic dialup will be do by the equipment

■ **Short message system enabled**: bits are used to set up the SMS mechanism : if bits are written with "11" by the master MODBUS, an alarm will send a SMS.

❑ **Cyclic dial up period.**

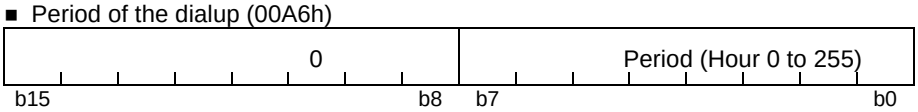
The equipment may periodically dial up. This function can be used to check that G200 is alive and to download measurements.

■ Hour of the first dialup during the day (00A5h)




Set = 1 : Cyclic dial up mechanism is ON (only if "Alarm message set up"=1)
= 0 : Cyclic dial up mechanism is OFF

■ Period of the dialup (00A6h)




Period (number of hours) between to dial up

Each time when data is written in this zone(A5-A6h) automatic call is Re-initialise.

❑ **Phone number (for GSM only)**

Phone number of the host computer system or SMS, used to send the alarms.
**15 figures maximum encoded.**
Only this figures are accepted : <0 to 9>, '+'=<A>
Zone initialised with <F..F> : Flair 200C doesn't send alarms.
i.e. :+330476606599 encoded value :

| Phone number | Word |
|---|---|
| FFFA | 95h |
| 3304 | 96h |
| 7660 | 97h |
| 6599 | 98h |

❑ **IP address (for GPRS only)**
For instance, 193.251.9.68 is converted as follows :

| IP address | Word |
|---|---|
| C1FB | A7h |
| 0944 | A8h |

❑ **IP port (for GPRS only)**

**Possible values:** 1 to 65535

For instance, 502 converted as follows :

| IP port | Word |
|---|---|
| 0x01F6 | ABh |

❑ **Load current deadband (I_mean, I_min, I_max)**
Load current variation (expressed in %) above which the measured current value is stored in the event stack.

Possible values: 0 to 100

❑ **Minimum current variation for deadband (I_mean, I_min, I_max)**
Minimum load current variation (expressed in A) above which the measured current value is stored in the event stack.

Possible values: 0 to 250

**Caution:** to be stored, a current measurement must overtake **both** the load current variation **and** the minimum current variation.

❑ **DeadBand voltage availability**
Minimum voltage presence percentage variation above which the voltage availability is stored in the event stack.

Possible values: 0 to 20

❑**Communication fault counter threshold**

Successive number of missing hourly measurements above which an alarm is sent.

Possible values: 1 to 4.

❑**Measurement period**

Period of time for current measurement recording

Possible values: 0002h= test (every 2 minutes)
003Ch= standard( every 1 hour)

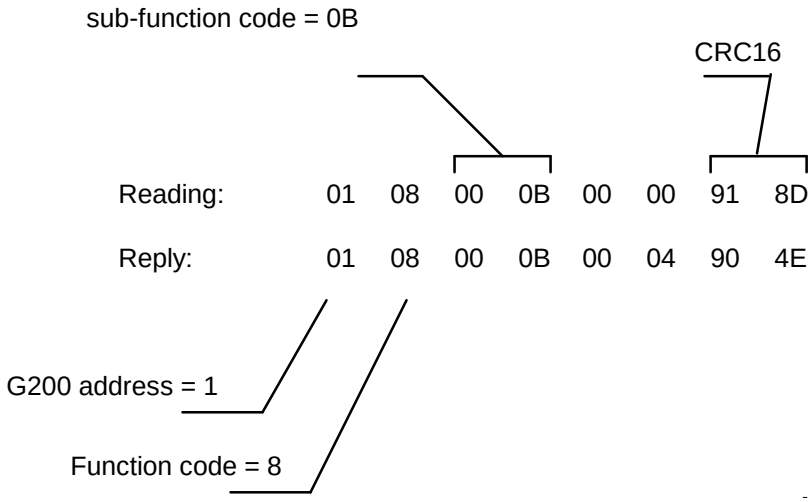
***Caution:*** *it is mandatory to use 1 hour measurement period for normal operation.*

## Diagnostic counter reading

The sub-function codes recognised by the G200 are:

- 0000h: return query data.
- 000Ah: clear counters and diagnostic register.
- 000Bh: reading of the number of frames exchanged.
- 000Ch: reading of the number frames received with CRC errors (CPT2).
- 000Dh: reading of the number of exception replies (CPT3).
- 000Eh: reading of the number of frames addressed to the station (CPT4).
- 000Fh: reading of broadcast requests received (CPT5).

The most significant bit of the sub-function code should be assigned with the sub-address of the G200 to be accessed.



## Report by exception without any modem

This function allows G200 to report an alarm to the master when :

- The link between G200 / Master is multipoint (permanent link, radio, optical fiber ...).
- The Master doesn't poll G200 all the time.

In this case configuration of G200 in the communication parameter menu is :

Modem: **Direct**

Alarm message enabled: **yes**

Then G200 can report an alarm by exception (modification of status, fault detection …)

- G200 transmits spontaneously an exception.

| Slave number | 00h | 00h | CRC16 |
|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 2 byte |

- The master then must read tables and events from the G200 which transmits spontaneously an exception.
- If the master doesn't reply by a reading of table, G200 has no transmits again the exception message after 1, 2, 5, 10, 10... minutes.
- G200 transmits this exception with a collision avoidance mechanism.

## Report by exception with GSM

When an indication configured as an alarm changes of state, G200 initiates an alarm cycle by dialling-up the main phone number after the ''dial-up delay time / first attempt''.

Two cases can occur:

*1 - The control center system doesn't answer:*

G200 dial-up again the ''main'' phone number after the ''dial-up delay time / second attempt'' and eventually try again after the ''dial-up delay time / third attempt''.

*2 - The control center system answer :*

The control center system send a broadcast message (Slave address = 0) and the function code = 0.

G200 sends back an exception message with its address, function code 0 with most significant bit set to 1 and the exception code = 0.

The control center system can then initiate a standard MODBUS Master/Slave communication.

## Report by exception with GPRS

When working in **non permanent mode** (G200 with solar panel power supply) and in case of alarm, the scada may be called by the G200.

The G200 initiates a TCP connection.

Two cases can occur:

*1 - The control center system isn't reachable:*

G200 again tries to make connection with the scada after the 'TCP/IP connect. delay / second attempt'' and eventually tries again after the 'TCP/IP connect. delay/ third attempt''.

*2 – Connection is successful:*
The control center system sends a broadcast message (Slave address = 0) with the function code = 17 (identification frame)

G200 answers this frame providing the control center with its own modbus address and its slave ID.

The control center system can then initiate a standard Modbus Master/Slave TCP communication.

```
41:12.42 identf      <  00 00 00 00 00 02 00 11
41:12.42 IDENTF      >> 00 00 00 00 00 06 FF 11 03 00 01 FF
41:13.81 write date  <  00 01 00 00 00 0F FF 10 00 02 00 04 08
                        00 07 04 0B 08 29 38 44
41:14.00 WRITE DATE  >> 00 01 00 00 00 06 FF 10 00 02 00 04
41:15.35 read  event <  00 02 00 00 00 06 FF 03 00 0F 00 21
41:15.35 READ  EVENT >> 00 02 00 00 00 45 FF 03 42 …
```

To avoid having the G200 not reachable, it can be configured to call at each change of IP address.

# MODBUS protocol (non GPRS version)

MODBUS is a master - slave protocol.

It is used to read or write one or more words (16 bits), as well as diagnostic counters.

Functions available:

■ **1**: read n output bits.

■ **2**: read n input bits.

■ **3**: read n output words.

■ **4**: read n input words.

■ **5**: write a bit.

■ **6**: write a word.

■ **8**: read diagnostic counters.

■ **16**: write several words.

Exchanges are carried out at the master's initiative and comprise a request from the master followed by the reply from the slave. The master's requests are addressed to a slave identified by its number in the first byte of the frame or else addressed to all the slaves (broadcast).

Broadcast commands are necessarily write commands. No reply is transmitted by the slaves.

### Structure of frames exchanged

All the frames exchanged (request and reply) have the same structure:

| Slave number | function code | data zone | check zone CRC16 |
|---|---|---|---|

Each message or frame contains 4 types of information:

■ slave number (1 byte): it specifies the receiving equipment (0 to FFh). If it is equal to zero, the request concerns all the slaves (broadcast) and there is no reply message.

■ function code (1 byte): it is used to select a command (read, write...) and check that the reply is correct.

■ data zone (n bytes): it contains the parameters linked to the function.

■ check zone (2 bytes): it is used to detect transmission errors.

**Please note that** words (2 bytes = 16 bits) are always written as high-order bits to low-order bits, with the exception of the CRC16 which is written as least significant bit, most significant bit.

### Synchronisation of exchanges

Any character that is received after a silence of more than 3 characters is considered as the beginning of a frame. A silence in the line equal to at least 3 characters should be respected between two frames.

Example: at 9600 baud, the time is equal to approximately 3 milliseconds.

### Checking of messages received by the slave

When the slave receives a frame, it checks the following, in order: CRC16, slave number, function code and function parameters.

■ If the CRC16 or the slave number are incorrect, the slave does not reply.

■ If the CRC16 and the slave number are correct, but the function code or parameters are not valid, the slave transmits an exception reply.

■ If the CRC16, slave number, function code and parameters are correct, the slave replies to the master's request.

### Exception reply transmitted by the slave

| Slave number | function code received with MSB set to 1 | Exception code<br>01 unknown function code<br>02 incorrect address<br>03 incorrect data | CRC16 |
|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 2 bytes |

## MODBUS TCP protocol (GPRS version only)

Modbus TCP protocol is based on the standard modbus protocol.

Same functions as standard modbus are available:

- **1**: read n output bits.
- **2**: read n input bits.
- **3**: read n output words.
- **4**: read n input words.
- **5**: write a bit.
- **6**: write a word.
- **8**: read diagnostic counters.
- **16**: write several words.

Plus:

- **17**: report slave ID. (only used with GPRS)

**Structure of frames exchanged**

Modbus TCP and its corresponding standard Modbus frame:

| Transaction identifier | Protocol identifier | Length | Unit identifier | function code | data zone |
|---|---|---|---|---|---|
| 2 bytes | 2 bytes | 2 bytes | 1 byte | 1 byte | |

Corresponding standard Modbus frame :

| Slave number | function code | data zone | check zone CRC16 |
|---|---|---|---|

- Transaction identifier: in the reply frame, the RTU sets the transaction identifier to the same value as the one in the request frame.
- Protocol identifier value is 0x0000.
- Length: it is the length of all the following data of the frame (including unit identifier and the function code)
- Unit identifier is the modbus address field of main menu of the communication module. Should be let to default value.

Except from the check zone that is suppressed in modbus TCP, the following of the field Length is treated the same way as in standard modbus.

In the following, the function codes will be described as used in standard modbus. To use them in modbus TCP, one only need to add transaction identifier, protocol identifier and length at the beginning of the frame and to cut the CRC at the end of it.

# Appendix

## Read N bits: functions n°1 and 2

Function 1: read output bits.
Function 2: read input bits.

**Request**

| Slave number | 1 or 2 | address of 1st bit (MSB+LSB) | number of bits | CRC16 |
|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 2 bytes | 2 bytes |

**Reply**

| Slave number | 1 or 2 | number of bytes read | 1st byte read | | last byte read | CRC16 |
|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 1 byte | 1 byte | N bytes | 1 byte | 2 bytes |

**Example**

Reading of 16 bits, bit address 300h of slave n°1
Request:`01 01 03 00 00 10 36 42`
Reply:`01 01 02 00 00 B9 FC`

## Read N words: functions n°3 and 4

The number of words to be read should be less than or equal to 125.

Function 3: read output words.
Function 4: read input words.

**Request**

| Slave number | 3 or 4 | address of 1st word (MSB+LSB) | number of words (MSB+LSB) | CRC16 |
|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 2 bytes | 2 bytes |

**Reply**

| Slave number | 3 or 4 | number of bytes read | 1st word read (MSB+LSB) | | last word read (MSB+LSB) | CRC16 |
|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 1 byte | | 1 byte | 2 bytes |

**Example**

Reading of words 40h to 43h of slave n°1,
Request:`01 03 00 40 00 04 45 DD`
Reply:`01 03 08 00 00 80 00 80 00 80 00 C2 17`

# Appendix

## Write a bit: function n°5

**Request**

| Slave number | 5 | address of bit (MSB+LSB) | bit value | 0 | CRC16 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 1 byte | 1 byte | 2 bytes |

**Reply**

The reply is an echo of the request indicating that the slave has acknowledged the value contained in the request.

| Slave number | 5 | address of bit (MSB+LSB) | bit value | 0 | CRC16 |
|---|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 1 byte | 1 byte | 2 bytes |

**Example**

Writing of bit to 1, bit address 301h of slave n°1,
Request:`01 05 03 01 FF 00 D6 7E`
Reply:`01 05 03 01 FF 00 D6 7E`

## Write a word: function n°6

**Request**

| Slave number | 6 | address of word (MSB+LSB) | value of word (MSB+LSB) | CRC16 |
|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 2 bytes | 2 bytes |

**Reply**

The reply is an echo of the request indicating that the slave has acknowledged the value contained in the request.

| Slave number | 6 | address of word (MSB+LSB) | value of word (MSB+LSB) | CRC16 |
|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 2 bytes | 2 bytes |

**Example**

Writing of word 30h of slave n°1, at the value 0001h
Request:`01 06 00 30 00 01 48 05`
Reply:`01 06 00 30 00 01 48 05`

## Read diagnostic counters: function n°8

Each slave is assigned diagnostic counters. There are 5 counters in all per slave. The counters are 16-bit words. When they reach FFFFh, they go back to 0000h.
When a request is sent by the master, the most significant byte in the sub-function code is assigned by the G200 equipment offset to access and the data are at 0000h.
When the slave sends a reply, the data contain the value of the counter concerned.

**Request / reply**

| Slave number | 8 | sub-function code (MSB+LSB) | data (MSB+LSB) | CRC16 |
|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 2 bytes | 2 bytes |

| | sub-function code | data |
|---|---|---|
| the slave should send the echo of the request | xx00 | XXXX |
| resetting of diagnostic counters | xx0A | 0000 |
| **reading of total number:** | | |
| of frames received with no CRC errors (CPT1) | xx0B | XXXX |
| of frames received with CRC errors (CPT2) | xx0C | XXXX |
| of the number of exception replies (CPT3) | xx0D | XXXX |
| of frames addressed to the station (CPT4) (excluding broadcast) | xx0E | XXXX |
| of broadcast requests received and correctly executed (CPT5) | xx0F | XXXX |

Sub-function n°0 is used to test transmission. The slave sends back the echo of the data received.

**Examples**

Resetting of counters for slave n°1,
Request:`01 08 00 0A 00 00 C0 09`
Reply:`01 08 00 0A 00 00 C0 09`

Reading of broadcast requests received (CPT5) for slave n°1, offset 3 (300h in storage space)
Request:`01 08 03 0F 00 00 D0 4C`
Reply:`01 08 03 0F 00 05 10 4F`

# Appendix

## Write N consecutive words: function n°16

The number of words to be written is between 1 and 123 and the number of bytes is between 2 and 246.
Words are written in increasing order of addresses.

**Request**

| Slave number | 10h | address of 1st word to write | number of words to write | number of bytes to write | values of words to write | CRC16 |
|---|---|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 2 bytes | 1 byte | N bytes | 2 bytes |

**Reply**

| Slave number | 10h | address of 1st word written (MSB+LSB) | number of words written (MSB+LSB) | CRC16 |
|---|---|---|---|---|
| 1 byte | 1 byte | 2 bytes | 2 bytes | 2 bytes |

**Example**

Writing of words 0302h to 0305h of slave n°1, (addresses 02h to 05h) with the values 0060h, 0A10h, 0B33h, 1662h

```
Request:     01 10 03 02 00 04 08 00 60 0A 10 0B 33 16 62 96 B3
Reply:       01 10 03 02 00 04 60 4E

51:26.43 read  state <  00 16 00 00 00 06 FF 04 00 01 00 01
51:26.43 READ  STATE >> 00 16 00 00 00 05 FF 04 02 00 69
```

## Report Slave ID: function n°17

Only used with G200 GPRS and Modbus TCP

This function code is used in the case G200 doesn't have a fix IP address on the network. It makes it possible to identify the equipment calling.
G200 will always answer with Status to On (0xFF in last byte of frame)

**Request**

| Transaction identifier | Protocol identifier | Length | Unit identifier | 11h |
|---|---|---|---|---|
| 2 bytes | 2 bytes | 2 bytes | 1 byte | 1 byte |

**Reply**

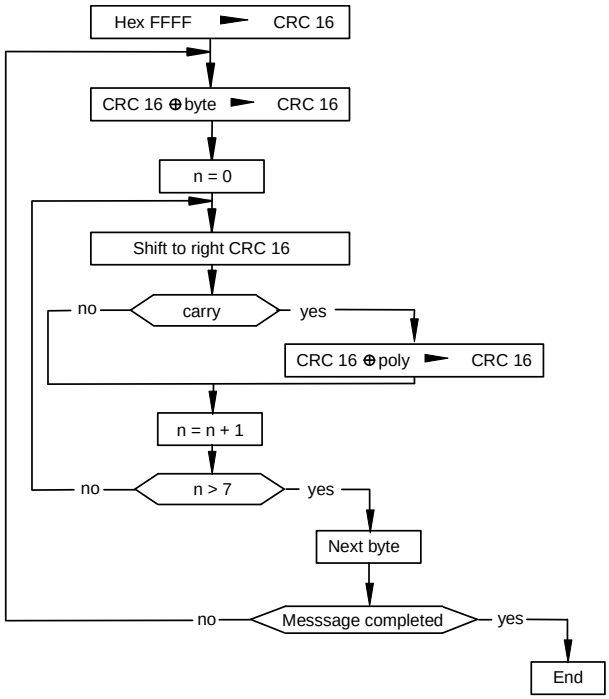
| Transaction identifier | Protocol identifier | Length | Unit identifier | 11h | 0x02 (Byte count) | Slave Id (MSB+LSB) | OxFF (Status à ON) |
|---|---|---|---|---|---|---|---|
| 2 bytes | 2 bytes | 2 bytes | 1 byte | 1 byte | 1 byte | 2 bytes | 1 byte |

**Example**

The request is addressed to all equipment connected (Unit identifier = 0x00) which corresponds to the only G200 that has created TCP connection. The G200 answers with modbus address to 255 and slave ID to 1.

```
41:12.42 identf      <  00 00 00 00 00 02 00 11
41:12.42 IDENTF      >> 00 00 00 00 00 06 FF 11 03 00 01 FF
```

## CRC 16 calculation algorithm




n = number of bits of data
poly= CRC16=1010 0000 0000 0001 calculation polynomial

## Write CRC 16 calculation in C language

Calculates and gives the CRC16 in the "buf" zone with length "len"■ *buf: pointer of buffer on which the calculations are performed.

■ len: length of buffer.

```
unsigned crc16(char *buf, int len)
    {
    #define POLY  0xA001
    char i;
    unsigned crc;

    for (crc = 0xFFFF; len != 0; len --)
        {
        crc ^= *buf ++;
        for (i = 0; i < 8; i ++)
            {
            if (crc & 0x0001)
                crc = (crc >> 1) ^ POLY;
            else
                crc >>= 1;
            }
        }
    return (crc);
    }
```

**Schneider Electric Industries SAS**

Schneider Electric Telecontrol
839 chemin des Batterses
Z.I. Ouest
01700 St Maurice de Beynost
Tel : +33 (0)4 78 55 13 13
Fax : +33 (0)4 78 55 50 00

http://www.schneider-electric.com
E-mail : telecontrol@schneider-electric.com



NT00142-EN-04 06/2016

Publication: Schneider Electric Telecontrol - Made in France
Production: Schneider Electric Telecontrol - Made in France
Impression: Schneider Electric Telecontrol - Made in France