
# EM21

## COMMUNICATION PROTOCOL

Version 1 Revision 1

July 16th, 2008

# Index

## 1.1 Introduction

The RS485 serial interface supports the MODBUS/JBUS (RTU) protocol. In this document only the information necessary to read/write from/to EM21 has been reported (not all the parts of the protocol have been implemented).
For a complete description of the MODBUS protocol please refer to the "Modbus_Application_Protocol_V1_1a.pdf" document that is downloadable from the www.modbus.org web site.

## 1.2 MODBUS functions

These functions are available on EM21:

- Reading of n "Holding Registers" (code 03h)
- Reading of n "Input Register" (code 04h)
- Writing of one "Holding Registers" (code 06h)
- Diagnostic (code 08h with sub-function code 00h)
- Broadcast mode (writing instruction on address 00h)

**IMPORTANT:**

1) In this document the "Modbus address" field is indicated in two modes:
   1.1) "**Modicom address**": it is the "6-digit Modicom" representation with Modbus function code 04 (Read Input Registers). It is possible to read the same values with function code 03 (Read Holding Registers) replacing the first digit ("3") with the number "4".
   1.2) "**Physical address**": it is the "word address" value to be included in the communication frame.
2) The functions 03h and 04h have exactly the same effect and can be used indifferently.
3) The communication parameters are to be set according to the configuration of the instrument (refer to EM21 instruction manual)

### 1.2.1 Function 03h (Read Holding Registers)

This function is used to read the contents of a contiguous block of holding registers (word). The Request frame specifies the starting register address and the number of registers to be read. It is possible to read maximum 11 registers (words) with a single request, when not differently specified.
The register data in the response message are packed as two bytes per register (word), with the binary contents right justified within each byte. For each register, the first byte contains the high order bits (MSB) and the second contains the low order bits (LSB).

Request frame

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | |
| Function code | 1 byte | 03h | |
| Starting address | 2 bytes | 0000h to FFFFh | Byte order: MSB, LSB |
| Quantity of registers (**N** word) | 2 bytes | 1 to 10h (1 to 11) | Byte order: MSB, LSB |
| CRC | 2 bytes | | |

Response frame (correct action)

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | |
| Function code | 1 byte | 03h | |
| Quantity of requested bytes | 1 byte | **N** word * 2 | |
| Register value | **N***2 bytes | | Byte order: MSB, LSB |
| CRC | 2 bytes | | |

Response frame (incorrect action)

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | Possible exception :<br>01h: illegal function<br>02h: illegal data address<br>03h: illegal data value<br>04h: slave device failure |
| Function code | 1 byte | 83h | |
| Exception code | 1 byte | 01h, 02h, 03h, 04h (see note) | |
| CRC | 2 bytes | | |

### 1.2.2 Function 04h (Read Input Registers)

This function code is used to read the contents of a contiguous block of input registers (word). The Request frame specifies the starting register address and the number of registers to be read. It is possible to read maximum 11 register (word) with a single request, when not differently specified.
The register data in the response message are packed as two bytes per register (word), with the binary contents right justified within each byte. For each register, the first byte contains the high order bits (MSB) and the second contains the low order bits (LSB).

Request frame

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | |
| Function code | 1 byte | 04h | |
| Starting address | 2 bytes | 0000h to FFFFh | Byte order: MSB, LSB |
| Quantity of registers (**N** word) | 2 bytes | 1 to 10h (1 to 11) | Byte order: MSB, LSB |
| CRC | 2 bytes | | |

Response frame (correct action)

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | |
| Function code | 1 byte | 04h | |
| Quantity of requested bytes | 1 byte | **N** word * 2 | |
| Register value | **N***2 bytes | | Byte order: MSB, LSB |
| CRC | 2 bytes | | |

Response frame (incorrect action)

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | Possible exception :<br>01h: illegal function<br>02h: illegal data address<br>03h: illegal data value<br>04h: slave device failure |
| Function code | 1 byte | 84h | |
| Exception code | 1 byte | 01h, 02h, 03h, 04h | |
| CRC | 2 bytes | | |

### 1.2.3 Function 06h (Write Single Holding Register)

This function code is used to write a single holding register. The Request frame specifies the address of the register (word) to be written and its content.
The correct response is an echo of the request, returned after the register content has been written.

Request frame

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | |
| Function code | 1 byte | 06h | |
| Starting address | 2 bytes | 0000h to FFFFh | Byte order: MSB, LSB |
| Register value | 2 bytes | 0000h to FFFFh | Byte order: MSB, LSB |
| CRC | 2 bytes | | |

Response frame (correct action)

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | |
| Function code | 1 byte | 06h | |
| Starting address | 2 bytes | 0000h to FFFFh | Byte order: MSB, LSB |
| Register value | 2 bytes | 0000h to FFFFh | Byte order: MSB, LSB |

| | | | |
|---|---|---|---|
| CRC | 2 bytes | | |

Response frame (incorrect action)

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | Possible exception :<br>01h: illegal function<br>02h: illegal data address<br>03h: illegal data value<br>04h: slave device failure |
| Function code | 1 byte | 86h | |
| Exception code | 1 byte | 01h, 02h, 03h, 04h | |
| CRC | 2 bytes | | |

### 1.2.4 Function 08h (Diagnostic with sub-function code 00h)

MODBUS function 08h provides a series of tests to check the communication system between a client (Master) device and a server (Slave), or to check various internal error conditions in a server.
EM24-DIN supports only 0000h sub-function code (Return Query Data). With this sub-function the data passed in the request data field is to be returned (looped back) in the response. The entire response message should be identical to the request.

Request frame

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | |
| Function code | 1 byte | 08h | |
| Sub-function | 2 bytes | 0000h | |
| Data (**N** word) | **N** *2 bytes | Data | Byte order: MSB, LSB |
| CRC | 2 bytes | | |

Response frame (correct action)

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7 (1 to 247) | |
| Function code | 1 byte | 08h | |
| Sub-function | 2 bytes | 0000h | |
| Data (**N** word) | **N** *2 bytes | Data | Byte order: MSB, LSB |
| CRC | 2 bytes | | |

Response frame (incorrect action)

| Description | Length | Value | Note |
|---|---|---|---|
| Physical address | 1 byte | 1 to F7h (1 to 247) | Possible exception :<br>01h: illegal function<br>02h: illegal data address<br>03h: illegal data value<br>04h: slave device failure |
| Function code | 1 byte | 88h | |
| Exception code | 1 byte | 01h, 02h, 03h, 04h | |
| CRC | 2 bytes | | |

### 1.2.5 Broadcast mode

In broadcast mode the master can send a request (command) to all the slaves. No response is returned to broadcast requests sent by the master. It is possible to send the broadcast message only with function code 06h using address 00h.

## 1.3 Application notes

### 1.3.1 RS485 general considerations

1. To avoid errors due to the signal reflections or line coupling, it is necessary to terminate the bus at the beginning and at the end (inserting a 120 ohm 1/2W 5% resistor between line B and A in the last instrument and in the Host interface).
2. The network termination is necessary even in case of point-to-point connection and/or of short distances.
3. For connections longer than 1000m or if in the network there are more than 160 instruments (with 1/5 unit load as used in EM21 interface), a signal repeater is necessary.
4. For bus connection it is suggested to use an AWG24 balanced pair cable and to add a third wire for GND connection. Connect GND to the shield if a shielded cable is used.
5. The GND is to be connected to ground only at the host side.
6. If an instrument does not answer within the “max answering time”, it is necessary to repeat the query. If the instrument does not answer after 2 or 3 consecutive queries, it is to be considered as not connected, faulty or reached with a wrong address. The same consideration is valid in case of CRC errors or incomplete response frames.

### 1.3.2 MODBUS timing

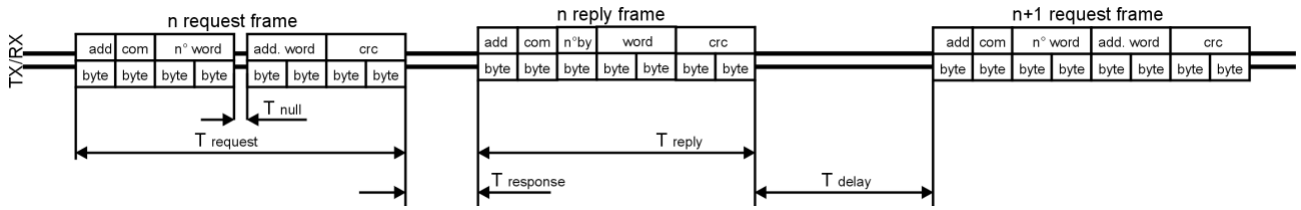



Fig. 1 : 2-wire timing diagram

| Timing characteristics of reading function: | msec |
|---|---|
| T response: Max answering time | 500ms |
| T response: Typical answering time | 40ms |
| T delay: Minimum time before a new query | 3,5char |
| T null: Max interruption time during the request frame | 2,5char |

# 2 TABLES

## 2.1 Data format representation In Carlo Gavazzi instruments

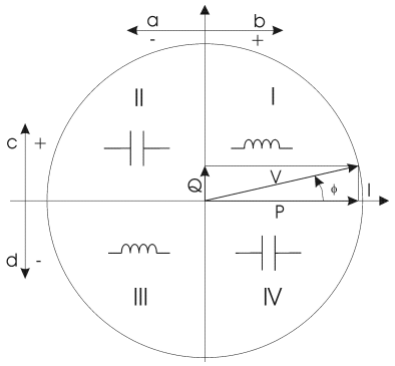
The variables are represented by integers or floating numbers, with 2's complement notation in case of "signed" format, using the following:

| Format | IEC data type | Description | Bits | Range |
|---|---|---|---|---|
| INT16 | INT | Integer | 16 | -32768 .. 32767 |
| UINT16 | UINT | Unsigned integer | 16 | 0 .. 65535 |
| INT32 | DINT | Double integer | 32 | $-2^{31}$ .. $2^{31}$ |
| UINT32 | UDINT | Unsigned double int | 32 | 0 .. $2^{32}$-1 |
| UINT64 | ULINT | Unsigned long integer | 64 | 0 .. $2^{64}$-1 |
| IEEE754 SP | | Single-precision floating-point | 32 | $-(1+[1-2^{-23}])x2^{127}$ .. $2^{128}$ |

For all the formats the byte order (inside the single word) is MSB->LSB. In INT32, UINT32 and UINT64 formats, the word order is LSW-> MSW.

### 2.1.1 Geometric representation

According to the signs of the power factor , the active power P and the reactive power Q, it is possible to obtain a geometric representation of the power vector, as indicated in the drawing below, according to EN 60253-23:




a = Exported active power
b = Imported active power
c = Imported reactive power
d = Exported reactive power

Fig. 2 : Geometric Representation

## 2.2 Maximum and minimum electrical values in EM21

The maximum electrical input values are reported in the following table. If the input is above the maximum value the display shows "----".

Table 2.1-1

| | AV5 input option | | AV6 input option | |
|---|---|---|---|---|
| | **Max value** | **Min value** | **Max value** | **Min value** |
| VL-N | 485V | 0 | 150V | 0 |
| VL-L | 840V | 0 | 260V | 0 |
| A | 6,5A | 0 | 6,5A | 0 |
| VT | 6000 | 1.0 | 6000 | 1.0 |
| CT | 60000 | 1.0 | 60000 | 1.0 |

The overflow indication “----“ is displayed when the MSB value of the relevant variable is 7FFFh.

## 2.3 Instantaneous variables and meters

**MODBUS**: read only mode with functions code 03 and 04 Table 2.3-1

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 300001 | 0000h | 2 | V L1-N | INT32 | Value weight: Volt*10 |
| 300003 | 0002h | 2 | V L2-N | INT32 | |
| 300005 | 0004h | 2 | V L3-N | INT32 | |
| 300007 | 0006h | 2 | V L1-L2 | INT32 | |
| 300009 | 0008h | 2 | V L2-L3 | INT32 | |
| 300011 | 000Ah | 2 | V L3-L1 | INT32 | |
| 300013 | 000Ch | 2 | A L1 | INT32 | Value weight: Ampere*1000 |
| 300015 | 000Eh | 2 | A L2 | INT32 | |
| 300017 | 0010h | 2 | A L3 | INT32 | |
| 300019 | 0012h | 2 | W L1 | INT32 | Value weight: Watt*10 |
| 300021 | 0014h | 2 | W L2 | INT32 | |
| 300023 | 0016h | 2 | W L3 | INT32 | |
| 300025 | 0018h | 2 | VA L1 | INT32 | Value weight: VA*10 |
| 300027 | 001Ah | 2 | VA L2 | INT32 | |
| 300029 | 001Ch | 2 | VA L3 | INT32 | |
| 300031 | 001Eh | 2 | VAR L1 | INT32 | Value weight: var*10 |
| 300033 | 0020h | 2 | VAR L2 | INT32 | |
| 300035 | 0022h | 2 | VAR L3 | INT32 | |
| 300037 | 0024h | 2 | V L-N Σ | INT32 | Value weight: Volt*10 |
| 300039 | 0026h | 2 | V L-L Σ | INT32 | |
| 300041 | 0028h | 2 | W Σ | INT32 | Value weight: Watt*10 |
| 300043 | 002Ah | 2 | VA Σ | INT32 | Value weight: VA*10 |
| 300045 | 002Ch | 2 | VAR Σ | INT32 | Value weight: var*10 |
| 300047 | 002Eh | 1 | PF L1 | INT16 | Negative values correspond to lead(C), positive value correspond to lag(L) Value weight: PF*1000 |
| 300048 | 002Fh | 1 | PF L2 | INT16 | |
| 300049 | 0030h | 1 | PF L3 | INT16 | |
| 300050 | 0031h | 1 | PF Σ | INT16 | |
| 300051 | 0032h | 1 | Phase sequence | INT16 | The value –1 corresponds to L1-L3-L2 sequence, the value 0 corresponds to L1-L2-L3 sequence. The phase sequence value is meaningful only in a 3-phase system |
| 300052 | 0033h | 1 | Hz | INT16 | Value weight: Hz |
| 300053 | 0034h | 2 | KWh(+) TOT | INT32 | Value weight: kWh*10 |
| 300055 | 0036h | 2 | Kvarh(+) TOT | INT32 | Value weight: kvarh*10 |

## 2.4 Firmware version and revision code

**MODBUS**: read only mode with functions code 03 and 04 limited to a word at a time Table 2.4-1

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 300771 | 0302h | 1 | Version code | UINT 16 | Value=0: Version “A” |
| 300772 | 0303h | 1 | Revision code | UINT 16 | Value=0: Revision “0” |

## 2.5 Programming lock status

**MODBUS**: read only mode with functions code 03 and 04 limited to a word at a time Table 2.5-1

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 300773 | 0304h | 1 | Programming lock (trimmer position in the rear of the display) | UINT 16 | Value=1: programming locked Value=0: programming unlocked |

## 2.6 Carlo Gavazzi Controls identification code

**MODBUS**: read only mode with functions code 03 and 04 limited to a word at a time Table 2.6-1

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 300012 | 000Bh | 1 | Carlo Gavazzi Controls identification code | UINT 16 | Value=57: EM21 AV5 input model<br>Value=58: EM21 AV6 input model |

.

## 2.7 Programming parameter tables

### 2.7.1 Password configuration menu

**MODBUS**: read and write mode Table 2.7-1

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 304097 | 1000h | 1 | PASSWORD | UINT 16 | Minimum valid value: 0d<br>Maximum valid value: 999d |

### 2.7.2 “Application” menu

**MODBUS**: read and write mode Table 2.7-2

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 304098 | 1001h | 1 | Type of application | UINT 16 | Value=0: “A” application<br>Value=1: “B” application<br>Value=2: “C” application |

### 2.7.3 System configuration menu

**MODBUS**: read and write mode Table 2.7-3

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 304099 | 1002h | 1 | Measuring system | UINT 16 | Value=0: “3Pn”<br>Value=1: “3P1”<br>Value=2: “2P”<br>Value=3: “1P”<br>Value=4: “3P” |

### 2.7.4 PT and CT configuration menu

**MODBUS**: read and write mode Table 2.7-4

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 304100 | 1003h | 2 | Current transformer ratio | UINT 32 | Value min = 10 (CT=1,0)<br>Value max = 600000 (CT=60000.0) |
| 304102 | 1005h | 2 | Voltage transformer ratio | UINT 32 | Value min = 10 (VT=1,0)<br>Value max = 60000 (VT=6000.0) |

### 2.7.5 Pulse output configuration menu

**MODBUS**: read read and write mode

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 304104 | 1007h | 1 | Kwh per pulse | UINT 16 | Value min = 1 (0,01Kwh)<br>Value max = 999 (9,99KWh) |

### 2.7.6 Serial port configuration menu

**MODBUS**: read and write mode Table 2.7-5

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 304105 | 1008h | 1 | RS485 instrument address | UINT 16 | Value min = 1<br>Value max = 247 |

Note: The number of stop bits is fixed to “1” and the parity control is fixed to “none”.

### 2.7.7 Reset commands

**MODBUS**: write only mode Table 2.7-12

| Modicom address | Physical address | Length (words) | VARIABLE ENG. UNIT | Data Format | Notes |
|---|---|---|---|---|---|
| 312289 | 3000h | 1 | Reset of all meters (KwhTOT and KvarhTOT) | UINT 16 | Value=1: Command is executed<br><br>All other values produce no effects |

### 2.7.8 Note

A default value is not automatically assignes to the parameters when an out-of-range or invalid value is written.
If an invalid value is assigned to any parameter, the unit could not work properly.

# 3 REVISIONS

This is the first release of the EM21 communication protocol

### 3.1 Modifications from Version 1 Revision 0

The frequency value doesn’t has to multiplied by 10 (see par. 1.3, tab 2.3.1)