## Contents

| Rev | DESCRIPTION | Date | Sw |
|---|---|---|---|
| B | Formal revision | 10/05/2016 | ➢ 3.03 |
|  |  |  |  |
|  |  |  |  |

# 1.0 ABSTRACT

### Physical level

The electrical communication line complies with the EIA-RS485 standard in half-duplex modality.
In this case, as only two wires are used, only one instrument at a time can engage the line; this means that there must be a master which polls the slave instruments so the demand and the request are alternated.

On the same line only 32 instruments can be attached (master included). In order to increase the number of the slave instrument, the necessary repeaters must be used.

The communication parameters are :

| | |
|---|---|
| Baud rate | : programmable (device dependant) |
| bit n. | : 8 |
| stop bit | : 1 |
| parity | : programmable (device dependant) |

### Data link level

The data are transmitted in a packet form (message) and are checked by a U_WORD (CRC).
See the description of the data packet in the next paragraphs for more details.

### Application level

The communication protocol used is MODBUS / JBUS compatible.
Up to 255 different instruments can be managed by the protocol.
There are no limitations to the number of possible retries done by the master.
A delay between the response from the slave and the next command could be necessary and it is specified for each device (timing).

## 2.0 DATA MESSAGE DESCRIPTION

The generic data message is composed as following :

|  |  |  |  |
|---|---|---|---|
| Device address | Functional code | Data | CRC word |

Two answers are possible :

Answer containing data

|  |  |  |  |
|---|---|---|---|
| Device address | Functional code | Data | CRC word |

Error answer

|  |  |  |  |
|---|---|---|---|
| Device address | Functional code<br>+ 0x80 | Error code | CRC word |

### 2.1 Parameters description

Device address : device identification number in the network.
It must be the same for the demand and the answer.
Format : 1 BYTE from 0 to 0xff
0 is for broadcast messages with no answer

Functional code : command code
Used functional code :
Format : 1 BYTE
0x03 : reading of consecutive words
0x10 : writing of consecutive words

Data : they can be
- the address of the required words (in the demand)
- the data (in the answer)

CRC word : it is the result of the calculation done on all the bytes in the message

## 2.2 Data format

The following types of format are used for the data values :

| | | |
|---|---|---|
| * U_WORD | : one WORD | - unsigned |
| * S_WORD | : one WORD | - signed |
| * UD_WORD | : two WORDS | - unsigned |
| * SD_WORD | : two WORDS | - signed |

If the required data is in a D_WORD format, 2 WORDS are transmitted and the MSW comes before the LSW

| MSB | LSB | MSB | LSB |
|---|---|---|---|
| Most Significant WORD | | Least Significant WORD | |

Example : 1000 = 0x 03 e8 or
0x 00 00 03 e8 (if UD_WORD)

| MSB | LSB | MSB | LSB |
|---|---|---|---|
| 0x00 | 0x00 | 0x03 | 0xe8 |

## 2.3 Description of CRC calculation

The following is an example of the CRC calculation in C language.

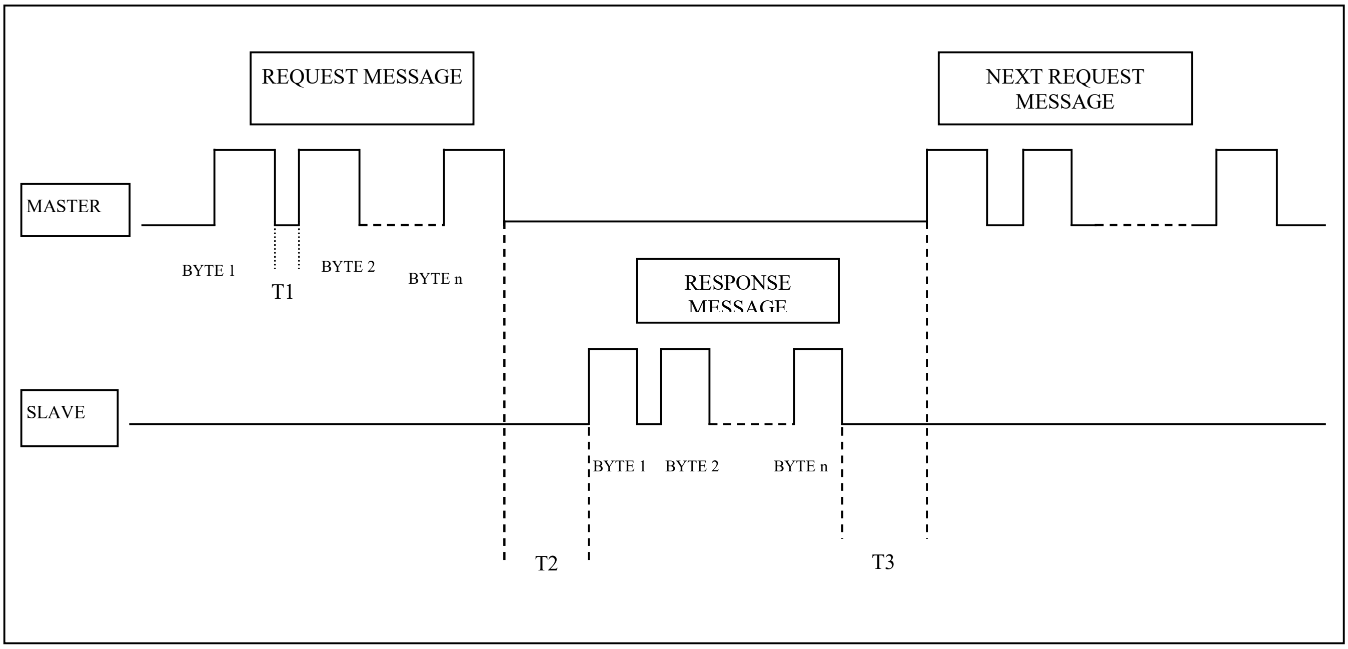
```
unsigned int calc_crc (char *ptbuf, unsigned int num)
/*     ****************************************************************
 *     Descrizione : calculates a data buffer CRC WORD
 *     Input       :      ptbuf = pointer to the first byte of the buffer
 *                   num   = number of bytes
 *     Output      : //
 *     Return      :
**     ***************************************************************/
 {
  unsigned int crc16;
  unsigned int temp;
  unsigned char c, flag;


  crc16 = 0xffff;                              /* init the CRC WORD */
  for (num; num>0; num--) {
        temp = (unsigned int) *ptbuf;          /* temp has the first byte */
        temp &= 0x00ff;                        /* mask the MSB */
         crc16 = crc16 ^ temp;                 /* crc16 XOR with temp */
        for (c=0; c<8; c++) {
              flag = crc16 & 0x01;             /* LSBit di crc16 is mantained */
              crc16 = crc16 >> 1;              /* Lsbit di crc16 is lost */
              if (flag != 0)
                   crc16 = crc16 ^ 0x0a001;    /* crc16 XOR with 0x0a001 */
        }
        ptbuf++;                               /* pointer to the next byte */
  }

  crc16 = (crc16 >> 8) | (crc16 << 8);         /* LSB is exchanged with MSB */

  return (crc16);

} /* calc_crc */
```

## 2.4 Error management

If the received message is incorrect (CRC16 is wrong) the polled slave doesn't answer. If the message is correct but there are errors (wrong functional code or data) it can't be accepted, so the slave answers with an error message.

The error codes are defined in the following part of the document.

## 2.5 Timing



| TIME | DESCRIPTION | Min & Max VALUES |
|---|---|---|
| T1 | **Time between characters.**<br>If this time exceeds the max. time allowed, the message is not considered by device. | Typ. = 20 ms |
| T2 | **Slave response time**<br>Minimum response delay to Master request. | Min = 20 ms |
| T3 | Time before a new message request from the Master can be issued | Min = 1 ms |

## 3.0 COMMANDS

### Code 0x03 : reading of one or more consecutive WORDS

Command format :

| BYTE | BYTE | MSB LSB | MSB LSB | |
|---|---|---|---|---|
| Device address | Funct. Code | First WORD address | WORDS number | CRC16 |

Answer format (containing data) :

| BYTE | BYTE | BYTE | MSB LSB | MSB LSB | |
|---|---|---|---|---|---|
| Device address | Funct. Code | BYTES number | WORD 1 ....... | WORD N. | CRC16 |

The BYTES number must always match the WORDS number (in the demand) * 2.

Answer format (the demand was wrong) :

| BYTE | BYTE | BYTE | |
|---|---|---|---|
| Device address | Funct. Code + 0x80 | Error code | CRC16 |

Error codes :
* 0x01 : incorrect functional code
* 0x02 : wrong first WORD address
* 0x03 : incorrect data

### Code 0x10 : writing of more consecutive WORDS

Command format :

| BYTE | BYTE | MSB LSB | MSB LSB | BYTE | MSB LSB | MSB LSB | |
|---|---|---|---|---|---|---|---|
| Device address | Funct. Code | First WORD address | WORDS number | BYTE numbers | Word Value | | CRC16 |

Answer format (containing data) :

| BYTE | BYTE | MSB LSB | MSB LSB | |
|---|---|---|---|---|
| Device address | Funct. Code | First WORD address | WORD N. | CRC16 |

The BYTES number must always match the WORDS number (in the demand) * 2.

Answer format (the demand was wrong) :

| BYTE | BYTE | BYTE | |
|---|---|---|---|
| Device address | Funct. Code + 0x80 | Error code | CRC16 |

Error codes :
* 0x01 : incorrect functional code
* 0x02 : wrong first WORD address
* 0x03 : incorrect data

# 4.0 VARIABLES

## 4.1 Data addresses

Both variables and groups of variables can be required.
All the variables with consecutive addresses can be required at one time.
The following is the table with the addresses and the meaning of the variables.

| Address | | Read/Write | Format | Description |
|---|---|---|---|---|
| HEX | DEC | | | |
| | | | | **Energy** |
| 0x325 | 805 | R | UD_WORD | 3-phase : Total positive active energy |
| 0x329 | 809 | R | UD_WORD | 3-phase : Total positive reactive energy |
| 0x32d | 813 | R/W[1] | UD_WORD | 3-phase : Partial positive active energy |
| 0x331 | 817 | R/W[1] | UD_WORD | 3-phase : Partial positive reactive energy |
| | | | | **Average power** |
| 0x350 | 848 | R | UD_WORD | 3-phase : average power |
| 0x354 | 852 | R/W[1] | UD_WORD | 3-phase : peak maximum demand |
| 0x358 | 856 | R/W[1] | UD_WORD | 3-phase : peak maximum demand 2° tariffs (where available) |
| | | | | |
| 0x348 | 840 | R | UD_WORD | Operating time counter (where available) |

Note 1: The only writable value is 0x0000000 in order to reset the stored value.
Different values won’t have effect.

The following table must be used to retrieve all information of the real time measurements.
The user can poll on both tables without any more operation, just change the Modbus address in the protocol data message.

| **Address** | Byte n. | **Description** | **Unit** |
|---|---|---|---|
| 0x1000 | UD_WORD | Phase 1 : phase voltage | mV |
| 0x1002 | UD_WORD | Phase 2 : phase voltage | mV |
| 0x1004 | UD_WORD | Phase 3 : phase voltage | mV |
| 0x1006 | UD_WORD | Phase 1 : current | mA |
| 0x1008 | UD_WORD | Phase 2 : current | mA |
| 0x100a | UD_WORD | Phase 3 : current | mA |
| 0x100c | UD_WORD | 0 | |
| 0x100e | UD_WORD | Chained voltage : L1-L2 | mV |
| 0x1010 | UD_WORD | Chained voltage : L2-L3 | mV |
| 0x1012 | UD_WORD | Chained voltage : L3-L1 | mV |
| 0x1014 | UD_WORD | 3-phase : active power | (1) |
| 0x1016 | UD_WORD | 3-phase : reactive power | (1) |
| 0x1018 | UD_WORD | 3-phase : apparent power | (1) |
| 0x101a | U_WORD | 3-phase : sign of active power | (2) |
| 0x101b | U_WORD | 3-phase : sign of reactive power | (2) |
| 0x101c | UD_WORD | 3-phase : total positive active energy | (3) |
| 0x101e | UD_WORD | 3-phase : total positive reactive energy | (3) |
| 0x1020 | UD_WORD | 0 | |
| 0x1022 | UD_WORD | Run hour meter | minutes |
| 0x1024 | U_WORD | 3-phase : power factor | 1/100 |
| 0x1025 | U_WORD | 3-phase : sector of power factor (cap or ind) | (4) |
| 0x1026 | U_WORD | Frequency | Hz/10 |
| 0x1027 | UD_WORD | 3-phase : average power | (1) |
| 0x1029 | UD_WORD | 3-phase : peak maximum demand | (1) |
| 0x102b | U_WORD | Time counter for average power | minutes |
| 0x102c | UD_WORD | Phase 1 : active power | (1) |
| 0x102e | UD_WORD | Phase 2 : active power | (1) |
| 0x1030 | UD_WORD | Phase 3 : active power | (1) |
| 0x1032 | U_WORD | Phase 1 : sign of active power | (2) |
| 0x1033 | U_WORD | Phase 2 : sign of active power | (2) |
| 0x1034 | U_WORD | Phase 3 : sign of active power | (2) |
| 0x1035 | UD_WORD | Phase 1 : reactive power | (1) |
| 0x1037 | UD_WORD | Phase 2 : reactive power | (1) |
| 0x1039 | UD_WORD | Phase 3 : reactive power | (1) |
| 0x103b | U_WORD | Phase 1 : sign of reactive power | (2) |
| 0x103c | U_WORD | Phase 2 : sign of reactive power | (2) |
| 0x103d | U_WORD | Phase 3 : sign of reactive power | (2) |
| 0x103e | UD_WORD | 3-phase : partial/second tariff positive active energy | (3) |
| 0x1040 | UD_WORD | 3-phase : partial/second tariff positive reactive energy | (3) |
| 0x1042 | UD_WORD | 3-phase : second tariff peak maximum demand | (1) |
| 0x1044 | U_WORD | 3-phase : power factor phase 1 | 1/100 |
| 0x1045 | U_WORD | 3-phase : power factor phase 2 | 1/100 |
| 0x1046 | U_WORD | 3-phase : power factor phase 3 | 1/100 |
| 0x1047 | U_WORD | 3-phase : sector of power factor phase 1 | (4) |
| 0x1048 | U_WORD | 3-phase : sector of power factor phase 2 | (4) |
| 0x1049 | U_WORD | 3-phase : sector of power factor phase 3 | (4) |

| | | | |
|---|---|---|---|
| 0x0c8 | BYTE | Reset - bit to bit defined | (5) |
| 0x100 | U_WORD | Current transformer ratio (KTA) | integer |
| 0x102 | U_WORD | Voltage transformer ratio (KTV) | *10<br>E.g. 1.0 => 10 |
| 0x300 | BYTE | Device identifier | 0x71 |

(1) ---------------------------------------------------------------------------------

```
W, var, VA / 100  if  KTA*KTI < 6000
W, var, VA        if  KTA*KTI >= 6000
```

(2) ---------------------------------------------------------------------------------

```
0 : positive
1 : negative
```

(3) ---------------------------------------------------------------------------------

| **Transformer ratio** | **Measurement unit** | **Display Format** | **Protocol Format** |
|---|---|---|---|
| 1 ≤ KTA*KTV < 10 | Wh(varh) * 10 | xxxxxx.yy k | xxxxxxyy |
| 10 ≤ KTA*KTV < 100 | Wh(varh) * 100 | xxxxxxx.y k | xxxxxxxy |
| 100 ≤ KTA*KTV < 1000 | kWh(kvarh) | xxxxxxxx k | xxxxxxxx |
| 1000 ≤ KTA*KTV < 10000 | kWh(kvarh) * 10 | xxxxxx.yy M | xxxxxxyy |
| 10000 ≤ KTA*KTV < 100000 | kWh(kvarh) * 100 | xxxxxxx.y M | xxxxxxxy |
| 100000 ≤ KTA*KTV | kWh(kvarh) * 1000 | xxxxxxxx M | xxxxxxxx |

(4) ---------------------------------------------------------------------------------

```
0 : PF = 0 or 1
1 : ind
2 : cap
```

(5) ---------------------------------------------------------------------------------

WRITE ONLY

```
0x01  : partial active energy
0x02  : partial reactive energy
0x08  : operating time counter reset (where available)
0x10  : peak maximum demand reset
```

## 4.2 Variables description

### Average power

Average power
**This is the power calculated with the shifting average algorithm. It is updated every minute.**
Format : UD_WORD
Measurement unit : W/100 or W due to the product respectively KTV*KTA<6000 or KTV*KTA>= 6000

Peak maximum demand
**This is the power obtained as the maximum of the average powers and it is updated at the end of average period.**
Format : UD_WORD
Measurement unit : W/100 or W due to the product respectively KTV*KTA<6000 or KTV*KTA>= 6000

### General

Current transformer ratio (KTA)

The current transformer ratio is the ratio between the rated primary value and the rated secondary value.
For example, if a CT primary/secondary ratio is 100/5, the value to be set in the device is 20 and this is also the value given on the remote line.
Format : U_WORD
Measurement unit : //

Voltage transformer ratio (KTV)
The voltage transformer ratio is the ratio between the rated primary value and the rated secondary value.
For example, if a VT primary/secondary is 380/100, the value to be set in the device is 3.8
For the TVs, the first decimal of the ratio is mantained and so the value given on the remote line is multiplied by 10, in this case 38.
Format : U_WORD
Measurement unit : //