Contents

Rev	DESCRIPTION	Date	Sw
B	Formal revision	10/05/2016	➢ 1.00

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 + 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 */
```
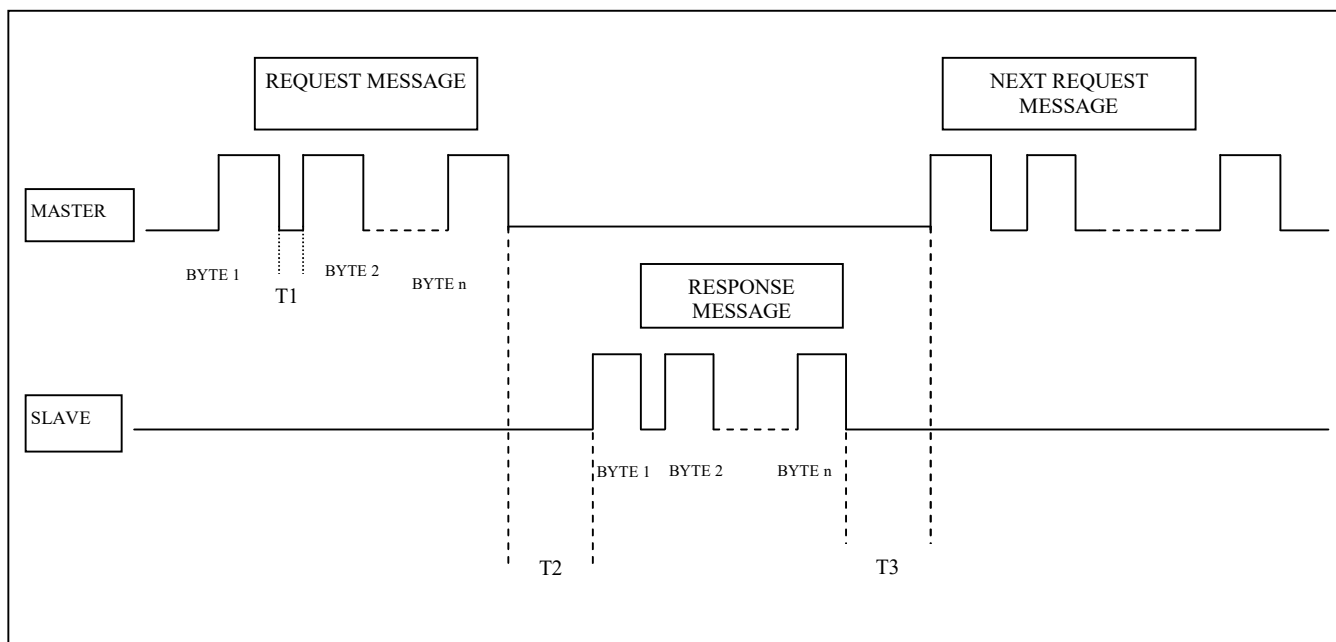
2.4 Error management

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.** If this time exceeds the max. time allowed, the message is not considered by device.	Typ. = 20 ms
T2	**Slave response time** 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.

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
First Counter	0x2000	Long	voltage	mV
	0x2002	Long	current	mA
	0x2004	Long	active power	W/100
	0x2006	WORD	sign of active power	(2)
	0x2007	Long	reactive power	Var/100
	0x2009	WORD	sign of reactive power	(2)
	0x200a	WORD	power factor	1/100
	0x200b	WORD	sector of power factor (cap or ind)	(1)
	0x200c	Long	average power	W/100
	0x200e	Long	peak maximum demand	W/100
	0x2010	Long	Elapsed time	min
	0x2012	Long	total positive active energy	kWh/100
	0x2014	Long	total positive reactive energy	kvarh/100
	0x2016	WORD	Frequency	Hz/10
Second Counter	0x2020	Long	voltage	mV
	0x2022	Long	current	mA
	0x2024	Long	active power	W/100
	0x2026	WORD	sign of active power	(2)
	0x2027	Long	reactive power	Var/100
	0x2029	WORD	sign of reactive power	(2)
	0x202a	WORD	power factor	1/100
	0x202b	WORD	sector of power factor (cap or ind)	(1)
	0x202c	Long	average power	W/100
	0x202e	Long	peak maximum demand	W/100
	0x2030	Long	Elapsed time	min
	0x2032	Long	total positive active energy	kWh/100
	0x2034	Long	total positive reactive energy	kvarh/100
	0x2036	WORD	Frequency	Hz/10
Third Counter	0x2040	Long	voltage	mV
	0x2042	Long	current	mA
	0x2044	Long	active power	W/100
	0x2046	WORD	sign of active power	(2)
	0x2047	Long	reactive power	Var/100
	0x2049	WORD	sign of reactive power	(2)
	0x204a	WORD	power factor	1/100
	0x204b	WORD	sector of power factor (cap or ind)	(1)
	0x204c	Long	average power	W/100
	0x204e	Long	peak maximum demand	W/100
	0x2050	Long	Elapsed time	min
	0x2052	Long	total positive active energy	kWh/100
	0x2054	Long	total positive reactive energy	kvarh/100
	0x2056	WORD	Frequency	Hz/10
	0x110	LONG	Power Threshold for starting counter 1	% *100 (2.50%=250)
	0x112	LONG	Power Threshold for starting counter 2	% *100 (2.50%=250)
	0x114	LONG	Power Threshold for starting counter 3	% *100 (2.50%=250)
	0xC8	WORD	Parameter reset	(4)
	0300	WORD	Device identifier	0x03

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

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

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

      0 : positive
      1 : negative

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

            WRITABLE ONLY

      0x01 : reset peak maximum demand counter 1
      0x02 : reset peak maximum demand counter 2
      0x04 : reset peak maximum demand counter 3
      0x08 : reset timer counter 1
      0x10 : reset timer counter 2
      0x20 : reset timer counter 3
```

4.2 Variables description

Energy

Positive energy
Format : long
Measurement unit : Hundreds of kWh/kvarh

Average power

Average power
This is the power calculated with the shifting average algorithm. It is updated every minute.
Format : long
Measurement unit : W/100
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 : long
Measurement unit : W/100

Operating time counter

Format : long
Measurement unit : min