



LOVATO ELECTRIC S.P.A.

24020 GORLE (BERGAMO) ITALIA
VIA DON E. MAZZA, 12
TEL. 035 4282111
TELEFAX (Nazionale): 035 4282200
TELEFAX (International): +39 035 4282400
Web www.LovatoElectric.com
E-mail info@LovatoElectric.com

1375 | GB 1113



SERIE RGK

**CONTROLLORE
GRUPPI ELETTROGENI**

**PROTOCOLLO DI COMUNICAZIONE
MODBUS®**



RGK SERIES

GENERATING SET CONTROLLER

MODBUS® COMMUNICATION PROTOCOL



PROTOCOLLO MODBUS®

I dispositivi di controllo dei gruppi elettrogeni della serie RGK700 supportano i protocolli di comunicazione Modbus RTU®, Modbus ASCII® e Modbus TCP® sulle porte seriali RS-232 e interfaccia ottica.

I dispositivi di controllo dei gruppi elettrogeni della serie RGK800 supportano i protocolli di comunicazione Modbus RTU®, Modbus ASCII® e Modbus TCP® sulle porte seriali RS-485, interfaccia ottica e sui moduli di espansione:

- EXM 10 11 RS 232
- EXM 10 12 RS485
- EXM 10 20 RS 485 + 2 relè
- EXM 10 10 USB
- EXM 10 13 Ethernet

Grazie a questa funzione e' possibile leggere lo stato degli apparecchi e controllare gli stessi tramite software di supervisione standard forniti da terze parti (SCADA) oppure tramite apparecchiature dotate di interfaccia Modbus® quali PLC e terminali intelligenti.

IMPOSTAZIONE DEI PARAMETRI

Per configurare il protocollo Modbus®, accedere al SETUP MENU e selezionare il menu M20.

Solo per l'RGK800 è possibile configurare 3 porte di comunicazione (n=1..3).

MENU M20 – COMUNICAZIONE SERIALE

PAR	Funzione	Range	Default
P20.n.01	Indirizzo	1 ..255	1
P20.n.02	Velocità RS-232 (baud)	1200 2400 4800 9600 19200 38400 57600 115200	9600 baud
P20.n.03	Formato dati	8 bit Nessuna 8 bit Dispari 8 bit Pari 7 bit Dispari 7 bit Pari	8 bit Nessuna
P20.n.04	Stop bit	1 2	1
P20.n.05	Protocollo	Modbus RTU Modbus ASCII Modbus TCP Propr. ASCII	Modbus RTU

Solo RGK800:

Per il modulo di espansione EXP 10 13 (Ethernet) esistono altri tre parametri.

PAR	Funzione	Range	Default
P20.n.06	Indirizzo IP	000.000.000.000 255.255.255.255	000.000.000.000
P20.n.07	Subnet MASK	000.000.000.000 255.255.255.255	000.000.000.000
P20.n.08	TCP-IP Port	0 - 9999	1001

MODBUS® PROTOCOL

The RGK700 series of generating set controller support the communication protocols Modbus RTU®, Modbus ASCII®, ModbusTCP® on the RS-232 and optical interface.

The RGK800 series of generating set controller support the communication protocols Modbus RTU®, Modbus ASCII®, ModbusTCP® on the RS-485, optical interface and the expansion modules.

- EXM 10 11 RS 232
- EXM 10 12 RS485
- EXM 10 20 RS 485 + 2 relè
- EXM 10 10 USB
- EXM 10 13 Ethernet

Using this function it is possible to read the device status and to control the units through third-party supervision software (SCADA) or through other intelligent devices supporting Modbus®, like PLCs.

PARAMETER SETTING

To configure the Modbus® protocol, enter SETUP MENU and choose the M20 menu:

Only for RGK800 it is possible to configure 3 different serial communication (n=1..3).

MENU M20 – SERIAL COMMUNICATION

PAR	Function	Range	Default
P20.n.01	Address	1 ..255	1
P20.n.02	RS-232 Baud Rate	1200 2400 4800 9600 19200 38400 57600 115200	9600 baud
P20.n.03	Data format	8 bit None 8 bit Odd 8 bit Even 7 bit Odd 7 bit Even	8 bit None
P20.n.04	Stop bit	1 2	1
P20.n.05	Protocol	Modbus RTU Modbus ASCII Modbus TCP Propr. ASCII	Modbus RTU

Only for RGK800:

For expansion module EXP 10 13 (Ethernet), there are other three parameters.

PAR	Function	Range	Default
P20.n.06	IP Address	000.000.000.000 255.255.255.255	000.000.000.000
P20.n.07	Subnet MASK	000.000.000.000 255.255.255.255	000.000.000.000
P20.n.08	TCP-IP Port	0 - 9999	1001

PROTOCOLLO MODBUS® RTU

Quando si utilizza il protocollo Modbus® RTU, la struttura del messaggio di comunicazione è così costituita:

T1	Indirizzo (8 bit)	Funzione (8 bit)	Dati (N x 8 bit)	CRC (16 bit)	T1
T2					T2
T3					T3

- Il campo Indirizzo contiene l'indirizzo dello strumento slave cui il messaggio viene inviato.
- Il campo Funzione contiene il codice della funzione che deve essere eseguita dallo slave.
- Il campo Dati contiene i dati inviati allo slave o quelli inviati dallo slave come risposta ad una domanda.
- Per la serie RGK la lunghezza massima consentita per il campo dati è di 80 registri da 16 bit (160 bytes)

• Il campo CRC consente sia al master che allo slave di verificare se ci sono errori di trasmissione. Questo consente, in caso di disturbo sulla linea di trasmissione, di ignorare il messaggio inviato per evitare problemi sia dal lato master che slave.

• La sequenza T1 T2 T3 corrisponde al tempo durante il quale non devono essere scambiati dati sul bus di comunicazione, per consentire agli strumenti collegati di riconoscere la fine di un messaggio e l'inizio del successivo. Questo tempo deve essere pari a 3.5 caratteri.

L'RGK misura il tempo trascorso tra la ricezione di un carattere e il successivo e se questo tempo supera quello necessario per trasmettere 3.5 caratteri, riferiti al baud rate impostato, il prossimo carattere viene considerato l'inizio di un nuovo messaggio.

FUNZIONI MODBUS®

Le funzioni disponibili sono:

03 = Read input register	Consente la lettura delle misure disponibili nel RGK
04 = Read input register	Consente la lettura delle misure disponibili nel RGK.
06 = Preset single register	Permette la scrittura dei parametri
07 = Read exception	Permette di leggere lo stato dell' apparecchio
10 = Preset multiple register	Permette la scrittura di più parametri
17 = Report slave ID	Permette di leggere informazioni relative all' apparecchio

Per esempio, se si vuole leggere dal RGK con indirizzo 01 il valore della potenza attiva L2 di rete che si trova alla locazione 36 (24 Hex), il messaggio da spedire è il seguente:

01	04	00	23	00	02	80	01
----	----	----	----	----	----	----	----

Dove:

01= indirizzo slave

04 = funzione di lettura locazione

00 23 = indirizzo della locazione diminuito di un'unità, contenete il valore della potenza attiva L2 rete

00 02 = numero di registri da leggere a partire dall'indirizzo 22

80 01 = checksum CRC

MODBUS® RTU PROTOCOL

If one selects the Modbus® RTU protocol, the communication message has the following structure:

T1	Address (8 bit)	Function (8 bit)	Data (N x 8 bit)	CRC (16 bit)	T1
T2					T2
T3					T3

- The Address field holds the serial address of the slave destination device.
- The Function field holds the code of the function that must be executed by the slave.
- The Data field contains data sent to the slave or data received from the slave in response to a query.
- For the RGK series, the maximum length for the data field is 80 16-bit registers (160 bytes)

• The CRC field allows the master and slave devices to check the message integrity. If a message has been corrupted by electrical noise or interference, the CRC field allows the devices to recognize the error and thereby to ignore the message.

• The T1 T2 T3 sequence corresponds to a time in which data must not be exchanged on the communication bus to allow the connected devices to recognize the end of one message and the beginning of another. This time must be at least 3.5 times the time required to send one character.

The RGK measures the time that elapses from the reception of one character and the following. If this time exceeds the time necessary to send 3.5 characters at the selected baudrate, then the next character will be considered as the first of a new message.

MODBUS® FUNCTIONS

The available functions are:

03 = Read input register	Allows to read the RGK measures.
04 = Read input register	Allows to read the RGK measures.
06 = Preset single register	Allows writing parameters
07 = Read exception	Allows to read the device status
10 = Preset multiple register	Allows writing several parameters
17 = Report slave ID	Allows to read information about the device.

For instance, to read the value of active power of line L2 of mains , which resides at location 36 (24 Hex), from the RGK with serial address 01, the message to send is the following:

01	04	00	23	00	02	80	01
----	----	----	----	----	----	----	----

Whereas:

01= slave address

04 = Modbus® function 'Read input register'

00 23 = Address of the required register (active power of L2 of mains) decreased by one

00 02 = Number of registers to be read beginning from address 22

80 01 = CRC Checksum

La risposta del DMG è la seguente:

01	04	04	00	01	8D	C0	CF	44
----	----	----	----	----	----	----	----	----

Dove:

01 = indirizzo del RGK (Slave 01)
04 = funzione richiesta dal Master
04 = numero di byte inviati dal RGK

00 01 8D C0 = valore esadecimale potenza attiva L2 di rete = 101824 = 1,01824 KW
CF 44 = checksum CRC

FUNZIONE 04: READ INPUT REGISTER

La funzione 04 permette di leggere una o più grandezze consecutive in memoria. L'indirizzo di ciascuna grandezza è indicato nella Tabella 2 riportate nelle ultime pagine del presente manuale. Come da standard Modbus®, l'indirizzo specificato nel messaggio va diminuito di 1 rispetto a quello effettivo riportato nella tabella.

Se l'indirizzo richiesto non è compreso nella tabella o il numero di registri richiesti è maggiore del numero consentito RGK ritorna un messaggio di errore (vedi tabella errori).

Richiesta Master:

Indirizzo slave	08h
Funzione	04h
MSB Indirizzo registro	00h
LSB Indirizzo registro	0Fh
MSB Numero registri	00h
LSB Numero registri	08h
LSB CRC	C1h
MSB CRC	56h

Nell'esempio vengono richiesti ,allo slave numero 8, 8 registri consecutivi a partire dall'indirizzo 10h.

Quindi vengono letti i registri dall' 10h al 17h.

Il comando termina sempre con il valore di checksum CRC.

Risposta Slave:

Indirizzo slave	08h
Funzione	04h
Numero di byte	10h
MSB Dato 10h	00h
LSB Dato 10h	00h
-----	---
MSB Dato 17h	00h
LSB Dato 17h	00h
LSB CRC	5Eh
MSB CRC	83h

La risposta è composta sempre dall'indirizzo dello slave, dalla funzione richiesta dal Master e dai dati dei registri richiesti. La risposta termina sempre con il valore di checksum CRC.

The DMG answer is the following:

01	04	04	00	01	FB	00	E9	74
----	----	----	----	----	----	----	----	----

Where:

01 = RGK address (Slave 01)
04 = Function requested by the master
04 = Number of bytes sent by the RGK

00 01 8D C0 = Hex value of the active power L2 of mains =101824 = 1,01824 KW
CF 44 = CRC checksum

FUNCTION 04: READ INPUT REGISTER

The Modbus® function 04 allows to read one or more consecutive registers from the slave memory. The address of each measure is given in the table 2 on the final pages of this manual.

As for Modbus® standard, the address in the query message must be decreased by one from the effective address reported in the table.

If the measure address is not included in the table or the number of requested registers exceeds the acceptable max number, the RGK will return an error code (see error table).

Master query:

Slave address	08h
Function	04h
MSB address	00h
LSB address	0Fh
MSB register number	00h
LSB register number	08h
LSB CRC	C1h
MSB CRC	56h

In the above example, slave 08 is requested for 8 consecutive registers beginning with address 10h.

Thus, registers from 10h to 17h will be returned. As usual, the message ends with the CRC checksum.

Slave response:

Slave address	08h
Function	04h
Byte number	10h
MSB register 10h	00h
LSB register 10h	00h
-----	---
MSB register 17h	00h
LSB register 17h	00h
LSB CRC	5Eh
MSB CRC	83h

The response is always composed of the slave address, the function code requested by the master and the contents of the requested registers. The answer ends with the CRC.

FUNZIONE 06: PRESET SINGLE REGISTER

Questa funzione permette di scrivere nei registri. Essa può essere utilizzata solo con i registri di indirizzo superiore a 1000 Hex. È possibile ad esempio impostare i parametri del setup. Qualora il valore impostato non rientri nel valore minimo e massimo della tabella il RGK risponderà con un messaggio di errore. Se viene richiesto un parametro ad un indirizzo inesistente verrà risposto con un messaggio di errore. L'indirizzo ed il range valido per i vari parametri può essere trovato nelle Tabelle 5, 6 e 7.

Richiesta Master:

Indirizzo slave	08h
Funzione	06h
MSB Indirizzo registro	2Fh
LSB Indirizzo registro	0Fh
MSB Dato	00h
LSB Dato	0Ah
LSB CRC	31h
MSB CRC	83h

Risposta Slave:

La risposta è un eco della domanda, cioè viene inviato al master l'indirizzo del dato da modificare e il nuovo valore del parametro.

FUNZIONE 07: READ EXCEPTION STATUS

Tale funzione permette di leggere lo stato in cui si trova il commutatore di linea.

Richiesta Master:

Indirizzo slave	08h
Funzione	07h
LSB CRC	47h
MSB CRC	B2h

La tabella seguente riporta il significato del byte inviato dal RGK come risposta:

BIT	SIGNIFICATO
0	Modo operativo OFF / Reset
1	Modo operativo MAN
2	Modo operativo AUT
3	Modo operativo TEST
4	In errore
5	Richiesta marcia motore
6	
7	Allarme globale attivato

FUNZIONE 17: REPORT SLAVE ID

Questa funzione permette di identificare il tipo di dispositivo.

Richiesta Master.

Indirizzo slave	08h
Funzione	11h
LSB CRC	C6h
MSB CRC	7Ch

FUNZIONE 06: PRESET SINGLE REGISTER

This function allows to write in the registers. It can be used only with registers with address higher than 1000 Hex. For instance, it is possible to change setup parameters. If the value is not in the correct range, the RGK will answer with an error message. In the same way, if the parameter address is not recognised, the RGK will send an error response.

The address and the valid range for each parameter are indicated in Tables 5, 6 and 7.

Master message:

Indirizzo slave	08h
Funzione	06h
MSB Indirizzo registro	2Fh
LSB Indirizzo registro	0Fh
MSB Dato	00h
LSB Dato	0Ah
LSB CRC	31h
MSB CRC	83h

Slave response:

The slave response is an echo to the query, that is the slave sends back to the master the address and the new value of the variable.

FUNZIONE 07: READ EXCEPTION STATUS

This function allows to read the status of the automatic transfer switch.

Master query:

Slave address	08h
Function	07h
LSB CRC	47h
MSB CRC	B2h

The following table gives the meaning of the status byte sent by the RGK as answer:

BIT	MEANING
0	Operative mode OFF / Reset
1	Operative mode MAN
2	Operative mode AUT
3	Operative mode TEST
4	Error on
5	Engine starting request
6	
7	Global alarm on

FUNZIONE 17: REPORT SLAVE ID

This function allows to identify the device type.

Master query.

Slave address	08h
Function	11h
LSB CRC	C6h
MSB CRC	7Ch

Risposta Slave:

Indirizzo slave	08h
Funzione	11h
Contatore bytes	08h
Dato 01 (Tipo) ❶	B4h
Dato 02 (Revisione software)	04h
Dato 03 (Revisione hardware)	00h
Dato 04 (Revisione parametri)	01h
Dato 05 (tipologia di prodotto) ❷	00h
Dato 06 (riservato)	00h
Dato 07 (riservato)	00h
Dato 08 (riservato)	00h
LSB CRC	...h
MSB CRC	...h

❶

170 - AAh = RGK700 AMF
 171 - ABh = RGK700 SA
 180 - B4h = RGK800 AMF
 181 - B5h = RGK800 SA

❷

0 - 00h = Serie RGK

ERRORI

Nel caso lo slave riceva un messaggio errato, segnala la condizione al master rispondendo con un messaggio composto dalla funzione richiesta in OR con 80 Hex, seguita da un codice di errore.

Nella seguente tabella vengono riportati i codici di errore inviati dallo slave al master:

TABELLA 1: CODICI ERRORE

COD	ERRORE
01	Funzione non valida
02	Indirizzo registro illegale
03	Valore del parametro fuori range
04	Impossibile effettuare operazione
06	Slave occupato, funzione momentaneamente non disponibile

Slave response:

Slave address	08h
Function	11h
Contatore bytes	08h
Data 01 (Type) ❶	B4h
Data 02 (Sw revision)	04h
Data 03 (Hardware revision)	00h
Data 04 (Parameter revision)	01h
Data 05 (type of device) ❷	00h
Data 06 (reserved)	00h
Data 07 (reserved)	01h
Data 08 (reserved)	01h
LSB CRC	...h
MSB CRC	...h

❶

170 - AAh = RGK700 AMF
 171 - ABh = RGK700 SA
 180 - B4h = RGK800 AMF
 181 - B5h = RGK800 SA

❷

0 - 00h = RGK series

ERRORS

In case the slave receives an incorrect message, it answers with a message composed by the queried function ORed with 80 Hex, followed by an error code byte.

In the following table are reported the error codes sent by the slave to the master:

TABLE 1: ERROR CODES

CODE	ERROR
01	Invalid function
02	Invalid address
03	Parameter out of range
04	Function execution impossible
06	Slave busy, function momentarily not available

FUNZIONE 16: PRESET MULTIPLE REGISTER

Questa funzione permette di modificare più parametri consecutivamente o parametri composti da più di 2 byte. L'indirizzo ed il range valido per i vari parametri possono essere trovati nella Tabella 8.

Richiesta Master:

Indirizzo slave	08h
Funzione	10h
MSB Indirizzo registro	20h
LSB Indirizzo registro	01h
MSB Numero registri	00h
LSB Numero registri	02h
MSB Dato	00h
LSB Dato	00h
MSB Dato	00h
LSB Dato	00h
LSB CRC	85h
MSB CRC	3Eh

Risposta Slave:

Indirizzo slave	08h
Funzione	10h
MSB Indirizzo registro	20h
LSB Indirizzo registro	01h
MSB Numero byte	00h
LSB Numero byte	02h
LSB CRC	1Bh
MSB CRC	51h

FUNZIONE 16: PRESET MULTIPLE REGISTER

This function allows to modify multiple parameters with a single message, or to preset a value longer than one register. The address and the valid range for each parameter are stated in Table 8.

Master message:

Slave address	08h
Function	10h
MSB register address	20h
LSB register address	01h
MSB register number	00h
LSB register number	02h
MSB data	00h
LSB data	00h
MSB data	00h
LSB data	00h
LSB CRC	85h
MSB CRC	3Eh

Slave response:

Slave address	08h
Function	10h
MSB register address	20h
LSB register address	01h
MSB byte number	00h
LSB byte number	02h
LSB CRC	1Bh
MSB CRC	51h

PROTOCOLLO MODBUS® ASCII

Il protocollo Modbus® ASCII viene utilizzato normalmente nelle applicazioni che richiedono di comunicare via modem.

Le funzioni e gli indirizzi disponibili sono gli stessi della versione RTU, ma i caratteri trasmessi sono in ASCII e la terminazione del messaggio non è effettuata a tempo ma con dei caratteri di ritorno a capo.

Se si seleziona il parametro P7.x.05 o P7.05 o come protocollo Modbus® ASCII, la struttura del messaggio di comunicazione sulla relativa porta di comunicazione è così costituita:

:	Indirizzo 2 chars	Funzione 2 chars	Dati (N chars)	LRC 2 chars	CR LF
---	----------------------	---------------------	-------------------	----------------	----------

- Il campo Indirizzo contiene l'indirizzo dello strumento slave cui il messaggio viene inviato.
- Il campo Funzione contiene il codice della funzione che deve essere eseguita dallo slave.
- Il campo Dati contiene i dati inviati allo slave o quelli inviati dallo slave come risposta ad una domanda. La massima lunghezza consentita è di (ved. Pag. 3) registri consecutivi.
- Il campo LRC consente sia al master che allo slave di verificare se ci sono errori di trasmissione. Questo consente, in caso di disturbo sulla linea di trasmissione, di ignorare il messaggio inviato per evitare problemi sia dal lato master che slave.
- Il messaggio termina sempre con i caratteri di controllo CRLF (0D 0A).

Esempio:

Per esempio, se si vuole leggere dal DMG con indirizzo 8 il valore della corrente di fase L3 equivalente che si trova alla locazione 12 (0C Hex), il messaggio da spedire è il seguente:

:	08	04	00	0B	00	02	E7	CRLF
---	----	----	----	----	----	----	----	------

Dove:

: = ASCII 3Ah = Delimitatore inizio messaggio
08 = indirizzo slave.
04 = funzione di lettura locazione.
00 0B = indirizzo della locazione diminuito di un'unità, contenente il valore della corrente di fase L3
00 02 = numero di registri da leggere a partire dall'indirizzo 04.
E7 = checksum LRC.
CRLF = ASCII 0Dh 0Ah = delimitatore fine messaggio

La risposta del DMG è la seguente:

:	08	04	04	00	00	A8	AE	9B	CR LF
---	----	----	----	----	----	----	----	----	----------

Dove:

: = ASCII 3Ah = Delimitatore inizio messaggio
08 = indirizzo del DMG (Slave 08).
04 = funzione richiesta dal Master.
04 = numero di byte inviati dallo slave.
00 00 A8 AE = valore esadecimale della corrente di fase L3 = 4.3182 A.
9B = checksum LRC.
CRLF = ASCII 0Dh 0Ah = delimitatore fine messaggio

MODBUS® ASCII PROTOCOL

The Modbus® ASCII protocol is normally used in application that require to communicate through a couple of modems.

The functions and addresses available are the same as for the RTU version, but the transmitted characters are in ASCII and the message end is delimited by Carriage return/ Line Feed instead of a transmission pause.

If one selects the parameter P7.x.05 or P7.05 as Modbus® ASCII protocol, the communication message on the correspondent communication port has the following structure:

:	Address (2 chars)	Function (2 chars)	Dates (N chars)	LRC (2 chars)	CR LF
---	----------------------	-----------------------	--------------------	---------------------	----------

- The Address field holds the serial address of the slave destination device.
- The Function field holds the code of the function that must be executed by the slave.
- The Data field contains data sent to the slave or data received from the slave in response to a query. The maximum allowable length is of (read pag. 3) consecutive registers.
- The LRC field allows the master and slave devices to check the message integrity. If a message has been corrupted by electrical noise or interference, the LRC field allows the devices to recognize the error and thereby ignore the message.
- The message terminates always with CRLF control character (0D 0A).

Example:

For instance, to read the value of the current phase L3, which resides at location 12 (0C Hex) from the slave with serial address 08, the message to send is the following:

:	08	04	00	0B	00	02	E7	CRLF
---	----	----	----	----	----	----	----	------

Whereas:

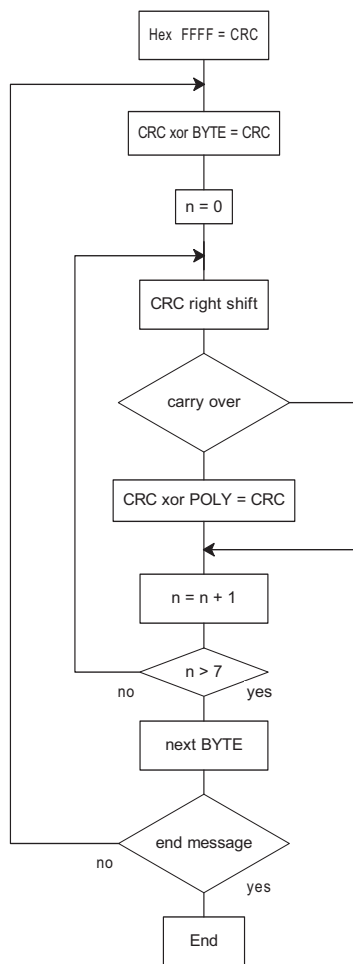
: = ASCII 3Ah message start delimiter
08 = slave address
04 = Modbus® function 'Read input register'
00 0B = Address of the required register (L3 current phase) decreased by one
00 02 = Number of registers to be read beginning from address 04
E7 = LRC Checksum
CRLF = ASCII 0Dh 0Ah = Message end delimiter

The DMG answer is the following:

:	08	04	04	00	00	A8	AE	9B	CR LF
---	----	----	----	----	----	----	----	----	----------

Whereas:

: = ASCII 3Ah message start delimiter
08 = DMG address (Slave 08)
04 = Function requested by the master
04 = Number of bytes sent by the multimeter
00 00 A8 AE = Hex value of the current phase of L3 (= 4.3182 A.)
9B = LRC checksum
CRLF = ASCII 0Dh 0Ah = Message end delimiter



Algoritmo di calcolo del CRC
CRC calculation algorithm

CALCOLO DEL CRC (CHECKSUM per RTU)

Esempio di calcolo:
Frame = 0207h

Inizializzazione CRC	1111	1111	1111	1111
Carica primo byte			0000	0010
Esegue xor con il primo Byte della frame	1111	1111	1111	1101
Esegue primo shift a dx	0111	1111	1111	1110 1
Carry=1, carica polinomio	1010	0000	0000	0001
Esegue xor con il polinomio	1101	1111	1111	1111
Esegue secondo shift dx	0110	1111	1111	1111 1
Carry=1, carica polinomio	1010	0000	0000	0001
Esegue xor con il polinomio	1100	1111	1111	1110
Esegue terzo shift	0110	0111	1111	1111 0
Esegue quarto shift	0011	0011	1111	1111 1
Carry=1, carica polinomio	1010	0000	0000	0001
Esegue xor con il polinomio	1001	0011	1111	1110
Esegue quinto shift dx	0100	1001	1111	1111 0
Esegue sesto shift dx	0010	0100	1111	1111 1
Carry=1, carica polinomio	1010	0000	0000	0001
Esegue xor con polinomio	1000	0100	1111	1110
Esegue settimo shift dx	0100	0010	0111	1111 0
Esegue ottavo shift dx	0010	0001	0011	1111 1
Carry=1, carica polinomio	1010	0000	0000	0001
Carica secondo byte della frame			0000	0111
Esegue xor con il Secondo byte della frame	1000	0001	0011	1001
Esegue primo shift dx	0100	0000	1001	1100 1
Carry=1, carica polinomio	1010	0000	0000	0001
Esegue xor con il polinomio	1110	0000	1001	1101
Esegue secondo shift dx	0111	0000	0100	1110 1
Carry=1, carica polinomio	1010	0000	0000	0001
Esegue xor con il polinomio	1101	0000	0100	1111
Esegue terzo shift dx	0110	1000	0010	0111 1
Carry=1, carica polinomio	1010	0000	0000	0001
Esegue xor con il polinomio	1100	1000	0010	0110
Esegue quarto shift dx	0110	0100	0001	0011 0
Esegue quinto shift dx	0010	0100	0000	1001 1
Carry=1, carica polinomio	1010	0000	0000	0001
Esegue xor con il polinomio	1001	0010	0000	1000
Esegue sesto shift dx	0100	1001	0000	0100 0
Esegue settimo shift dx	0010	0100	1000	0010 0
Esegue ottavo shift dx	0001	0010	0100	0001 0
Risultato CRC	0001	0010		
	0100	0001		
	12h	41h		

Nota: Il byte 41h viene spedito per primo (anche se e' il LSB), poi viene trasmesso 12h.

CALCOLO LRC (CHECKSUM per ASCII)

Esempio di calcolo:

Indirizzo	01	00000001
Funzione	04	00000100
Start address hi.	00	00000000
Start address lo.	00	00000000
Numero registri	08	00001000
	Somma	00001101
	Complemento a 1	11110010
	+ 1	00000001
	Complemento a 2	11110101

Risultato LRC F5

CRC CALCULATION (CHECKSUM for RTU)

Example of CRC calculation:
Frame = 0207h

CRC initialization	1111	1111	1111	1111
Load the first byte			0000	0010
Execute xor with the first Byte of the frame	1111	1111	1111	1101
Execute 1st right shift	0111	1111	1111	1110 1
Carry=1,load polynomial	1010	0000	0000	0001
Execute xor with the polynomial	1101	1111	1111	1111
Execute 2nd right shift	0110	1111	1111	1111 1
Carry=1,load polynomial	1010	0000	0000	0001
Execute xor with the polynomial	1100	1111	1111	1110
Execute 3rd right shift	0110	0111	1111	1111 0
Execute 4th right shift	0011	0011	1111	1111 1
Carry=1,load polynomial	1010	0000	0000	0001
Execute xor with the polynomial	1001	0011	1111	1110
Execute 5th right shift	0100	1001	1111	1111 0
Execute 6th right shift	0010	0100	1111	1111 1
Carry=1,load polynomial	1010	0000	0000	0001
Execute xor with the polynomial	1000	0100	1111	1110
Execute 7th right shift	0100	0010	0111	1111 0
Execute 8th right shift	0010	0001	0011	1111 1
Carry=1,load polynomial	1010	0000	0000	0001
Load the second byte of the frame			0000	0111
Execute xor with the Second byte of the frame	1000	0001	0011	1001
Execute 1st right shift	0100	0000	1001	1100 1
Carry=1,load polynomial	1010	0000	0000	0001
Execute xor with the polynomial	1110	0000	1001	1101
Execute 2nd right shift	0111	0000	0100	1110 1
Carry=1,load polynomial	1010	0000	0000	0001
Execute xor with the polynomial	1101	0000	0100	1111
Execute 3rd right shift	0110	1000	0010	0111 1
Carry=1,load polynomial	1010	0000	0000	0001
Execute xor with the polynomial	1100	1000	0010	0110
Execute 4th right shift	0110	0100	0001	0011 0
Execute 5th right shift	0010	0100	0000	1001 1
Carry=1,load polynomial	1010	0000	0000	0001
Execute xor with the polynomial	1001	0010	0000	1000
Execute 6th right shift	0100	1001	0000	0100 0
Execute 7th right shift	0010	0100	1000	0010 0
Execute 8th right shift	0001	0010	0100	0001 0
CRC Result	0001	0010		
	0100	0001		
	12h	41h		

Note: The byte 41h is sent first (even if it is the LSB), then 12h is sent.

LRC CALCULATION (CHECKSUM for ASCII)

Example of LRC calculation:

Address	01	00000001
Function	04	00000100
Start address hi.	00	00000000
Start address lo.	00	00000000
Number of registers	08	00001000
	Sum	00001101
	1. complement	11110010
	+ 1	00000001
	2. complement	11110101

LRC result F5

TABELLA 2:
MISURE FORNITE DAL PROTOCOLLO DI COM.
 (Utilizzabili con funzioni 03 e 04)

TABLE 2:
MEASURES SUPPLIED BY SERIAL
COMMUNICATION PROTOCOL
 (To be used with functions 03 and 04)

Indirizzo Address	WORDS	MISURA	MEASURE	UNITA' UNIT	FORMATO FORMAT	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
		MISURA MEDIA (AV)	AVERAGE MEASURE (AV)						
0002H	2	Tensione di fase L1 - Rete	L1 Phase Voltage - Mains	V/100	Unsigned long	•		•	
0004H	2	Tensione di fase L2 - Rete	L2 Phase Voltage - Mains	V/100	Unsigned long	•		•	
0006H	2	Tensione di fase L3 - Rete	L3 Phase Voltage - Mains	V/100	Unsigned long	•		•	
0008H	2	Tensione di fase L1 - Generatore	L1 Phase Voltage - Generator	V/100	Unsigned long	•	•	•	•
000AH	2	Tensione di fase L2 - Generatore	L2 Phase Voltage - Generator	V/100	Unsigned long	•	•	•	•
000CH	2	Tensione di fase L3 - Generatore	L3 Phase Voltage - Generator	V/100	Unsigned long	•	•	•	•
000EH	2	Corrente di fase L1	L1 Current	A/10000	Unsigned long	•	•	•	•
0010H	2	Corrente di fase L2	L2 Current	A/10000	Unsigned long	•	•	•	•
0012H	2	Corrente di fase L3	L3 Current	A/10000	Unsigned long	•	•	•	•
0014H	2	Corrente di Neutro	Neutral Current	A/10000	Unsigned long	•	•	•	•
0016H	2	Tensione L1-L2 - Rete	L1-L2 Voltage - Mains	V/100	Unsigned long	•		•	
0018H	2	Tensione L2-L3 - Rete	L2-L3 Voltage - Mains	V/100	Unsigned long	•		•	
001AH	2	Tensione L3-L1 - Rete	L3-L1 Voltage - Mains	V/100	Unsigned long	•		•	
001CH	2	Tensione L1-L2 - Generatore	L1-L2 Voltage - Generator	V/100	Unsigned long	•	•	•	•
001EH	2	Tensione L2-L3 - Generatore	L2-L3 Voltage - Generator	V/100	Unsigned long	•	•	•	•
0020H	2	Tensione L3-L1 - Generatore	L3-L1 Voltage - Generator	V/100	Unsigned long	•	•	•	•
0022H	2	Potenza Attiva L1 - Rete	L1 Active Power - Mains	W/100	Signed long	•		•	
0024H	2	Potenza Attiva L2 - Rete	L2 Active Power - Mains	W/100	Signed long	•		•	
0026H	2	Potenza Attiva L3 - Rete	L3 Active Power - Mains	W/100	Signed long	•		•	
0028H	2	Potenza Attiva L1 - Generatore	L1 Active Power - Generator	W/100	Signed long	•	•	•	•
002AH	2	Potenza Attiva L2 - Generatore	L2 Active Power - Generator	W/100	Signed long	•	•	•	•
002CH	2	Potenza Attiva L3 - Generatore	L3 Active Power - Generator	W/100	Signed long	•	•	•	•
002EH	2	Potenza Reattiva L1 - Rete	L1 Reactive Power - Mains	Var/100	Signed long	•		•	
0030H	2	Potenza Reattiva L2 - Rete	L2 Reactive Power - Mains	Var/100	Signed long	•		•	
0032H	2	Potenza Reattiva L3 - Rete	L3 Reactive Power - Mains	Var/100	Signed long	•		•	
0034H	2	Potenza Reattiva L1 - Generatore	L1 Reactive Power - Generator	Var/100	Signed long	•	•	•	•
0036H	2	Potenza Reattiva L2 - Generatore	L2 Reactive Power - Generator	Var/100	Signed long	•	•	•	•
0038H	2	Potenza Reattiva L3 - Generatore	L3 Reactive Power - Generator	Var/100	Signed long	•	•	•	•
003AH	2	Potenza Apparente L1 - Rete	L1 Apparent Power - Mains	VA/100	Unsigned long	•		•	
003CH	2	Potenza Apparente L2 - Rete	L2 Apparent Power - Mains	VA/100	Unsigned long	•		•	
003EH	2	Potenza Apparente L3 - Rete	L3 Apparent Power - Mains	VA/100	Unsigned long	•		•	
0040H	2	Potenza Apparente L1 - Generatore	L1 Apparent Power - Generator	VA/100	Unsigned long	•	•	•	•
0042H	2	Potenza Apparente L2 - Generatore	L2 Apparent Power - Generator	VA/100	Unsigned long	•	•	•	•
0044H	2	Potenza Apparente L3 - Generatore	L3 Apparent Power - Generator	VA/100	Unsigned long	•	•	•	•
0046H	2	Fattore Di Potenza L1 - Rete	L1 Power Factor - Mains	/10000	Signed long	•		•	
0048H	2	Fattore Di Potenza L2 - Rete	L2 Power Factor - Mains	/10000	Signed long	•		•	
004AH	2	Fattore Di Potenza L3 - Rete	L3 Power Factor - Mains	/10000	Signed long	•		•	
004CH	2	Fattore Di Potenza L1 - Generatore	L1 Power Factor - Generator	/10000	Signed long	•	•	•	•
004EH	2	Fattore Di Potenza L2 - Generatore	L2 Power Factor - Generator	/10000	Signed long	•	•	•	•
0050H	2	Fattore Di Potenza L3 - Generatore	L3 Power Factor - Generator	/10000	Signed long	•	•	•	•
0052H	2	Tensione LN equivalente- Rete	Eqv. Phase Voltage - Mains	V/100	Unsigned long	•		•	
0054H	2	Tensione LL equivalente- Rete	Eqv. Phase-To-Phase Voltage - Mains	V/100	Unsigned long	•		•	
0056H	2	Frequenza - Rete	Frequency - Mains	Hz/1000	Unsigned long	•		•	
0058H	2	Tensione LN equivalente - Generatore	Eqv. Phase Voltage - Generator	V/100	Unsigned long	•	•	•	•
005AH	2	Tensione LL equivalente - Generatore	Eqv. Phase-To-Phase Voltage - Generator	V/100	Unsigned long	•	•	•	•
005CH	2	Frequenza - Generatore	Frequency - Generator	Hz/1000	Unsigned long	•	•	•	•

Indirizzo Address	WORDS	MISURA	MEASURE	UNITA' UNIT	FORMATO FORMAT	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
005EH	2	Fattore Di Potenza equivalente - Rete	Eqv Power Factor - Mains	/10000	Signed long	•		•	
0060H	2	Vuoto	Free						
0062H	2	Vuoto	Free						
0064H	2	Fattore Di Potenza equivalente - Generatore	Eqv Power Factor - Generator	/10000	Signed long	•	•	•	•
0066H	2	Vuoto	Free						
0068H	2	Vuoto	Free						
006AH	2	Potenza Attiva equivalente - Rete	Eqv. Active Power - Mains	W/100	Signed long	•		•	
006CH	2	Potenza Reattiva equivalente - Rete	Eqv. Reactive Power - Mains	Var/100	Signed long	•		•	
006EH	2	Potenza Apparente equivalente - Rete	Eqv. Apparent Power - Mains	VA/100	Unsigned long	•		•	
0070H	2	Potenza Attiva equivalente - Generatore	Eqv. Active Power - Generator	W/100	Signed long	•	•	•	•
0072H	2	Potenza Reattiva equivalente - Generatore	Eqv. Reactive Power - Generator	Var/100	Signed long	•	•	•	•
0074H	2	Potenza Apparente equivalente - Generatore	Eqv. Apparent Power - Generator	VA/100	Unsigned long	•	•	•	•
0076H	2	Potenza Attiva % - Rete	% Active Power - Mains	%/100	Signed long	•		•	
0078H	2	Potenza Reattiva % - Rete	% Reactive Power - Mains	%/100	Signed long	•		•	
007AH	2	Potenza Apparente % - Rete	% Apparent Power - Mains	%/100	Unsigned long	•		•	
007EH	2	Potenza Reattiva % - Generatore	% Reactive Power - Generator	%/100	Signed long	•	•	•	•
0080H	2	Potenza Apparente % - Generatore	% Apparent Power - Generator	%/100	Unsigned long	•	•	•	•
0082H	2	Asimmetria Tensione Fase-Fase - Rete	Phase-Phase Voltage Asymmetry - Mains	%/100	Unsigned long	•		•	
0084H	2	Asimmetria Tensione Fase-Neutro - Rete	Phase-Neural Voltage Asymmetry- Mains	%/100	Unsigned long	•		•	
0086H	2	Asimmetria Corrente - Rete	Current Asymmetry - Mains	%/100	Unsigned long	•		•	
0088H	2	Asimmetria Tensione LL - Generatore	Phase-Phase Voltage Asymmetry - Generator	%/100	Unsigned long	•	•	•	•
008AH	2	Asimmetria Tensione LN - Generatore	Phase-Neural Voltage Asymmetry - Generator	%/100	Unsigned long	•	•	•	•
008CH	2	Asimmetria Corrente - Generatore	Current Asymmetry - Generator	%/100	Unsigned long	•	•	•	•
008EH	2	Velocità motore	Engine speed	Rpm/10	Unsigned long	•	•	•	•
0090H	2	Thd Tensione L1-2- Generatore	L1-2 Voltage Thd - Generator	%/100	Unsigned long	•	•	•	•
0092H	2	Thd Tensione L2-3- Generatore	L2-3 Voltage Thd - Generator	%/100	Unsigned long	•	•	•	•
0094H	2	Thd Tensione L3-1- Generatore	L3-1 Voltage Thd - Generator	%/100	Unsigned long	•	•	•	•
0096H	2	Thd Tensione L1- Generatore	L1 Voltage Thd - Generator	%/100	Unsigned long	•	•	•	•
0098H	2	Thd Tensione L2- Generatore	L2 Voltage Thd - Generator	%/100	Unsigned long	•	•	•	•
009AH	2	Thd Tensione L3- Generatore	L3 Voltage Thd - Generator	%/100	Unsigned long	•	•	•	•
009CH	2	Thd Corrente L1- Generatore	L1 Current Thd - Generator	%/100	Unsigned long	•	•	•	•
009EH	2	Thd Corrente L2- Generatore	L2 Current Thd - Generator	%/100	Unsigned long	•	•	•	•
00A0H	2	Thd Corrente L3- Generatore	L3 Current Thd - Generator	%/100	Unsigned long	•	•	•	•
00A2H	2	Thd Corrente N- Generatore	N Current Thd - Generator	%/100	Unsigned long	•	•	•	•
00A4H	2	CosPhi L1- Generatore	L1 CosPhi - Generator	/10000	Signed long	•	•	•	•
00A6H	2	CosPhi L2- Generatore	L2 CosPhi - Generator	/10000	Signed long	•	•	•	•
00A8H	2	CosPhi L3- Generatore	L3 CosPhi - Generator	/10000	Signed long	•	•	•	•
00AAH	2	Corrente di fase L1 - visualizzata -	L1 Current - view -	A/10000	Unsigned long	•	•	•	•
00ACH	2	Corrente di fase L2- visualizzata -	L2 Current - view -	A/10000	Unsigned long	•	•	•	•
00AEH	2	Corrente di fase L3 - visualizzata -	L3 Current - view -	A/10000	Unsigned long	•	•	•	•
00B0H	2	Corrente di Neutro - visualizzata -	Neutral Current - view -	A/10000	Unsigned long	•	•	•	•
00B2H	2	Velocità motore W o Pick-UP	Engine speed W or Pick-UP	Rpm/10	Unsigned long	•	•	•	•
1100H	2	Tempo mancante alla commutazione da GEN1 a GEN2	Time remaining before switchover (GEN1-GEN2)	h	Unsigned long	•	•	•	•
1102H	2	Tempo mancante alla commutazione da GEN2 a GEN1	Time remaining before switchover (GEN2-GEN1)	h	Unsigned long	•	•	•	•
1104H	2	Tempo mancante alla commutazione da GEN1 a GEN2	Time remaining before switchover (GEN1-GEN2)	min	Unsigned long	•	•	•	•
1106H	2	Tempo mancante alla commutazione da GEN2 a GEN1	Time remaining before switchover (GEN2-GEN1)	min	Unsigned long	•	•	•	•

Indirizzo Address	WORDS	MISURA	MEASURE	UNITA' UNIT	FORMATO FORMAT	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
		MISURA ISTANTANEA (IN)	INSTANTANEOUS MEASURE (IN)						
0200H	2	Tensione di fase L1 - Rete	L1 Phase Voltage - Mains	V/100	Unsigned long	•		•	
0202H	2	Tensione di fase L2 - Rete	L2 Phase Voltage - Mains	V/100	Unsigned long	•		•	
.....									
02B0H	2	Velocità motore W o Pick-UP	Engine speed W or Pick-UP	Rpm/10	Unsigned long	•	•	•	•
		MISURA MASSIMA (HI)	MAXIMUM MEASURE (HI)						
0400H	2	Tensione di fase L1 - Rete	L1 Phase Voltage - Mains	V/100	Unsigned long	•		•	
0402H	2	Tensione di fase L2 - Rete	L2 Phase Voltage - Mains	V/100	Unsigned long	•		•	
.....									
04B0H	2	Velocità motore W o Pick-UP	Engine speed W or Pick-UP	Rpm/10	Unsigned long	•	•	•	•
		MISURA MINIMA (LO)	MINIMUM MEASURE (LO)						
0600H	2	Tensione di fase L1 - Rete	L1 Phase Voltage - Mains	V/100	Unsigned long	•		•	
0602H	2	Tensione di fase L2 - Rete	L2 Phase Voltage - Mains	V/100	Unsigned long	•		•	
.....									
06B0H	2	Velocità motore W o Pick-UP	Engine speed W or Pick-UP	Rpm/10	Unsigned long	•	•	•	•

Indirizzo Address	WORDS	MISURA	MEASURE	UNITA' UNIT	FORMATO FORMAT	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
1D00H	2	Contatore CNT 1	Counter CNT 1	UM1	long	•	•	•	•
1D02H	2	Contatore CNT 2	Counter CNT 2	UM2	long	•	•	•	•
1D04H	2	Contatore CNT 3	Counter CNT 3	UM3	long	•	•	•	•
1D06H	2	Contatore CNT 4	Counter CNT 4	UM4	long	•	•	•	•
1D08H	2	Contatore CNT 5	Counter CNT 5	UM5	long	•	•		
1D0AH	2	Contatore CNT 6	Counter CNT 6	UM6	long	•	•		
1D0CH	2	Contatore CNT 7	Counter CNT 7	UM1	long	•	•		
1D0EH	2	Contatore CNT 8	Counter CNT 8	UM2	long	•	•		
0F50H	2	Ingresso analogico 1	Analog input 1	UM1	long	•	•		
0F52H	2	Ingresso analogico 2	Analog input 2	UM2	long	•	•		
0F54H	2	Ingresso analogico 3	Analog input 3	UM3	long	•	•		
0F56H	2	Ingresso analogico 4	Analog input 4	UM4	long	•	•		
0F58H	2	Ingresso analogico 5	Analog input 5	UM5	long	•	•		
0F5AH	2	Ingresso analogico 6	Analog input 6	UM6	long	•	•		
0F60H	2	Uscita analogico 1	Analog output 1	UM1	long	•	•		
0F62H	2	Uscita analogico 2	Analog output 2	UM2	long	•	•		
0F64H	2	Uscita analogico 3	Analog output 3	UM3	long	•	•		
0F66H	2	Uscita analogico 4	Analog output 4	UM4	long	•	•		
0F68H	2	Uscita analogico 5	Analog output 5	UM5	long	•	•		
0F6AH	2	Uscita analogico 6	Analog output 6	UM6	long	•	•		

Indirizzo Address	WORDS	MISURA	MEASURE	UNITA' UNIT	FORMATO FORMAT	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
0F80H	2	Ore di lavoro totali	Engine working hours	h	Unsigned long	•	•	•	•
0F82H	2	Tempo di lavoro totali	Engine working time	s	Unsigned long	•	•	•	•
0F84H	2	Ore di lavoro parziali	Engine partial running hours	h	Unsigned long	•	•	•	•
0F86H	2	Tempo di lavoro parziali	Engine partial running time	s	Unsigned long	•	•	•	•
0F88H	2	Ore manutenzione 1	Maintenance time 1	h	Unsigned long	•	•	•	•
0F8AH	2	Ore manutenzione 2	Maintenance time 2	h	Unsigned long	•	•	•	•
0F8CH	2	Ore manutenzione 3	Maintenance time 3	h	Unsigned long	•	•	•	•
0F8EH	2	Ore noleggio	Rent time	h	Unsigned long	•	•	•	•
0F90H	2	Numero di avviamenti corretti	Good cranks	n	Unsigned long	•	•	•	•
0F92H	2	Numero totale avviamenti	Total cranks	n	Unsigned long	•	•	•	•
0F94H	2	Percentuale numero di avviamenti corretti	Rate good cranks	%/10	Unsigned long	•	•	•	•
0F96H	2	Numero totale chiusure teleruttore generatore	Generator contactor closing counter	n	Unsigned long	•	•	•	•
0F98H	2	Replica della funzione modbus 17	Modbus Function 17 clone		Unsigned long	•	•	•	•
0FA0H	2	Temperatura	Temperature	°C / °F	Unsigned long	•	•	•	•
0FA2H	2	Pressione	Pressure	Bar *10	Unsigned long	•	•	•	•
0FA4H	2	Carburante	Fuel	%	Unsigned long	•	•	•	•
0FA6H	2	Sensore ausiliario	Auxiliary sensor		Unsigned long	•	•	•	•
0FA8H	2	Tensione Batteria	Battery voltage	V/100	Unsigned long	•	•	•	•
0FAAH	2	Tensione ingress D+	D+ input voltage	V/100	Unsigned long	•	•	•	•
0FACH	2	Tensione ingress AC	AC input voltage	V/100	Unsigned long	•	•	•	•
1A20H	2	Energia Attiva Importata totale - Rete	Total imp. Active Energy - Mains	kWh / 10	Unsigned long	•	•		
1A22H	2	Energia Attiva Esportata totale - Rete	Total exported Active Energy - Mains	kWh / 10	Unsigned long	•	•		
1A24H	2	Energia Reattiva Importata totale - Rete	Total imp. Reactive Energy - Mains	kvarh / 10	Unsigned long	•	•		
1A26H	2	Energia Reattiva Esportata totale - Rete	Total exp. Reactive Energy - Mains	kvarh / 10	Unsigned long	•	•		
1A28H	2	Energia apparente totale- Rete	Total Apparent Energy - Mains	kVAh / 10	Unsigned long				
1A2AH	2	Energia Attiva Importata totale - Generatore	Total imp. Active Energy - Generator	kWh / 10	Unsigned long	•	•	•	•
1A2CH	2	Energia Attiva Esportata totale - Generatore	Total exported Active Energy - Generator	kWh / 10	Unsigned long	•	•	•	•
1A2EH	2	Energia Reattiva Importata totale - Generatore	Total imp. Reactive Energy - Generator	kvarh / 10	Unsigned long	•	•	•	•
1A30H	2	Energia Reattiva Esportata totale - Generatore	Total exp. Reactive Energy - Generator	kvarh / 10	Unsigned long	•	•	•	•
1A32H	2	Energia apparente totale- Generatore	Total Apparent Energy - Generator	kVAh / 10	Unsigned long	•	•	•	•
1B20H	2	Energia Attiva Importata parziale - Rete	Partial imp. Active Energy - Mains	kWh / 10	Unsigned long	•	•		
1B22H	2	Energia Attiva Esportata parziale - Rete	Partial exported Active Energy - Mains	kWh / 10	Unsigned long	•	•		
1B24H	2	Energia Reattiva Importata parziale - Rete	Partial imp. Reactive Energy - Mains	kvarh / 10	Unsigned long	•	•		
1B26H	2	Energia Reattiva Esportata parziale - Rete	Partial exp. Reactive Energy - Mains	kvarh / 10	Unsigned long	•	•		
1B28H	2	Energia apparente parziale - Rete	Partial Apparent Energy - Mains	kVAh / 10	Unsigned long				
1B2AH	2	Energia Attiva Importata parziale - Generatore	Partial imp. Active Energy - Generator	kWh / 10	Unsigned long	•	•	•	•
1B2CH	2	Energia Attiva Esportata parziale - Generatore	Partial exported Active Energy - Generator	kWh / 10	Unsigned long	•	•	•	•
1B2EH	2	Energia Reattiva Importata parziale - Generatore	Partial imp. Reactive Energy - Generator	kvarh / 10	Unsigned long	•	•	•	•
1B30H	2	Energia Reattiva Esportata parziale - Generatore	Partial exp. Reactive Energy - Generator	kvarh / 10	Unsigned long	•	•	•	•
1B32H	2	Energia apparente parziale - Generatore	Partial Apparent Energy - Generator	kVAh / 10	Unsigned long	•	•	•	•

Indirizzo Address	WORDS	MISURA	MEASURE	UNITA' UNIT	FORMATO FORMAT	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
2100H	1	OR di tutti gli ingressi	OR of all Inputs	1	Unsigned int	•	•	•	•
2101H	1	INP 1	INP 1	bool	Unsigned int	•	•	•	•
.....									
2110H	1	INP16	INP16	bool	Unsigned int	•	•	•	•
2140H	1	OR di tutti le uscite	OR of all Outputs	1	Unsigned int	•	•	•	•
2141H	1	OUT 1	OUT 1	bool	Unsigned int	•	•	•	•
.....									
2150H	1	OUT 16	OUT 16	bool	Unsigned int	•	•	•	•
2180H	1	OR di tutte le variabili remote	OR of all remote variables	1	Unsigned int	•	•	•	•
2181H	1	REM 1	REM 1	bool	Unsigned int	•	•	•	•
.....									
2190H	1	REM 16	REM 16	bool	Unsigned int	•	•	•	•
21C0H	1	OR di tutti i limiti	OR of all limits	1	Unsigned int	•	•	•	•
21C1H	1	LIM 1	LIM 1	bool	Unsigned int	•	•	•	•
.....									
21D0H	1	LIM 16	LIM 16	bool	Unsigned int	•	•	•	•

Indirizzo Address	WORDS	ALLARMI	ALARMS	UNITA' UNIT	FORMATO FORMAT	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
2200H	1	Allarmi A01-A16	Alarms A01-A16	1	•	•	•	•	•
2201H	1	Allarmi A17-A32	Alarms A17-A32	1	•	•	•	•	•
2202H	1	Allarmi A33-A48	Alarms A33-A48	1	•	•	•	•	•
2203H	1	Allarmi A49-A60-UA1-UA2-UA3-UA4	Alarms A49-A60-UA1-UA2-UA3-UA4	1	•	•	•	•	•
2204H	1	Allarmi UA5-UA6-UA7	Alarms UA5-UA6-UA7	1	•	•	•	•	•

Indirizzo Address	WORDS	STATI	STATUS	FORMATO FORMAT	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
2210H	1	Stato globale dispositivo (bit 0-bit15) ②	Device global status(bit 0-bit15) ②	1	•	•	•	•
2211H	1	Stato globale dispositivo (bit 16-bit31) ②	Device global status(bit 16-bit31) ②	1	•	•	•	•

① Esempio:

Il valore all'indirizzo 2100H è 0x05 (esadecimale), = 0x00000101 vuol dire che gli ingressi 1 e 3 sono attivi.

② Leggendo la word agli indirizzi 2210H vengono restituiti 32 Bit con significato com da tabella

Bit 0	Modalità OFF
Bit 1	Modalità MAN
Bit 2	Modalità AUT
Bit 3	Modalità TEST
Bit 4	Tensione di rete OK
Bit 5	Tensione di generatore OK
Bit 6	Motore avviato
Bit 7	Generatore pronto
Bit 8	Allarme globale
Bit 9	Avaria meccanica
Bit 10	Avaria elettrica
Bit 11	Allarmi inseriti
Bit 12	Test automatico in corso
Bit 13	Test automatico abilitato
Bit 14	Teleruttore rete chiuso
Bit 15	Teleruttore generatore chiuso
Bit 16	(non usato)
...
Bit 31	(non usato)

① Example:

The value at address 2100H is 0x05 (hexadecimal) = 0x00000101 means that the inputs 1 and 3 are active

② Reading the word at address 2210H will return 32 bits with the following meaning:

Bit 0	OFF mode
Bit 1	MAN mode
Bit 2	AUT mode
Bit 3	TEST mode
Bit 4	Mains voltage OK
Bit 5	Gen. voltage OK
Bit 6	Engine running
Bit 7	Generator ready
Bit 8	Global alarm
Bit 9	Mechanical fault
Bit 10	Electrical fault
Bit 11	Alarms enabled
Bit 12	Automatic test running
Bit 13	Automatic test enabled
Bit 14	Mains contactor closed
Bit 15	Generator contactor closed
Bit 16	(not used)
...
Bit 31	(not used)

**TABELLA 3:
COMANDI
(Utilizzabili con funzione 06)**

**TABLE 3:
COMMANDS
(To be used with function 06)**

Indirizzo Address	WORDS	STATI	STATUS	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
4F00 H	1	Imposta variabile remora REM1 ❶	Set remote variable REM1 ❶	•	•	•	•
4F01 H	1	Imposta variabile remora REM2	Set remote variable REM2	•	•	•	•
.....							
4F0FH	1	Imposta variabile remora REM16	Set remote variable REM16	•	•	•	•
2F00H		Cambio modalità operativa ❷	Operative mode change ❷				
2F0AH	1	Simulazione pressione tasti pannello frontale ❸	Front panel keystore simulation ❸	•	•	•	•
2F03H	1	Valore 01h: Salvataggio eeprom	Valore 01h: Eeprom save	•	•	•	•
		Valore 02h: Salvataggio Fram	Valore 02h: Fram save	Rev 8	Rev 8	Rev 6	Rev 6
		Valore 04h: Salvataggio eeprom e fram e reboot	Valore 04h: Eeprom, Fram save and reboot	•	•	•	•
		Valore 08h: Salvataggio eeprom e fram	Valore 08h: EEprom, Fram save	Rev 8	Rev 8	Rev 6	Rev 6
2F07H	1	Valore 00h: Reset apparecchio Valore 01h: Reset apparecchio con salvataggio in fram	Valore 00h: Reset device Valore 01h: Reset device and save Fram	•	•	•	•
2FF0H	1	Esecuzione comando menu comandi ❶	Command menu execution ❶	•	•	•	•
28FAH	1	Valore 01h: Salvataggio impostazione orologio datario	Valore 01H: Save real time clock setting	•	•	•	•

❶ Scrivendo il valore AAH all'indirizzo indicato viene impostata la variabile remota a 1, scrivendo BBH viene impostata a 0.

❷ La seguente tabella indica i valori da scrivere all'indirizzo 2F00H per ottenere le corrispondenti funzioni

VALORE	Funzione
0	Passaggio a modalità OFF
1	Passaggio a modalità MAN
2	Passaggio a modalità AUT
3	Passaggio a modalità TEST

❸ La seguente tabella indica le posizioni dei bit da scrivere all'indirizzo 2F0AH per ottenere le corrispondenti funzioni

BIT	SIGNIFICATO
0	Tasto Su
1	Modalità MAN
2	Tasto destra
3	START
4	Modalità TEST
5	Modalità OFF
6	Modalità AUT
7	STOP
8	MAINS
9	Tasto Giù
10	Tasto enter
11	Tasto sinistra
12	GEN

❶ Writing AAH to the indicated address the remote variable will be set to 1, writing BBH the remote variable will be set to 0

❷ The following table shows the values to be written to address 2F00H to achieve the correspondent function

VALUE	Function
0	Switch to OFF mode
1	Switch to MAN mode
2	Switch to AUT mode
3	Switch to TEST mode

❸ The following table shows the bit position of the value to be written to address 2F0AH to achieve the correspondent function

BIT	MEANING
0	Key up
1	MAN mode
2	Key right
3	START
4	TEST mode
5	OFF mode
6	AUT mode
7	STOP mode
8	MAINS
9	Key down
10	Key enter
11	Key left
12	GEN

④ Scrivendo il valore da 0 a 24 all'indirizzo indicato viene eseguito la corrispondente funzione

	SIGNIFICATO (AMF)	SIGNIFICATO (SA)
0	C01 Reset intervallo manutenzione 1	C01 Reset intervallo manutenzione 1
1	C02 Reset intervallo manutenzione 2	C02 Reset intervallo manutenzione 2
2	C03 Reset intervallo manutenzione 3	C03 Reset intervallo manutenzione 3
3	C04 Reset contaore motore par	C04 Reset contaore motore par
4	C05 Reset contatore parziale energia rete	C06 Reset contatore parziale energia gen.
5	C06 Reset contatore parziale energia gen.	C07 Reset contatori generici CNTx
6	C07 Reset contatori generici CNTx	C08 Reset Limiti
7	C08 Reset Limiti	C09 Reset HI-LO
8	C09 Reset HI-LO	C10 Reset contaore motore tot
9	C10 Reset contaore motore tot	C11 Impostazione contaore motore
10	C11 Impostazione contaore motore	C12 Reset contatore avviamenti
11	C12 Reset contatore avviamenti	C13 Reset contatori chiusure
12	C13 Reset contatori chiusure	C15 Reset contatore totale energia generatore.
13	C14 Reset contatore totale energia rete	C16 Ricarica ore di noleggio
14	C15 Reset contatore totale energia generatore.	C17 Reset lista eventi
15	C16 Ricarica ore di noleggio	C18 Ripristino parametri a default
16	C17 Reset lista eventi	C19 Salva parametri nella memoria backup
17	C18 Ripristino parametri a default	C20 Ricarica parametri dalla memoria backup
18	C19 Salva parametri nella memoria backup	C21 Spurgo elettrovalvola
19	C20 Ricarica parametri dalla memoria backup	C22 Forced I/O
20	C21 Spurgo elettrovalvola	C23 Regolazione offset sensori resistivi
21	C22 FORCE IO	C24 Azzeramento Programma PLC
22	C23 Regolazione offset sensori resistivi	C25 Sleep mode
23	C24 Azzeramento Programma PLC	
24	C25 Sleep mode	

④ Writing value between 0 and 24 to the indicated address, the correspondent command will be executed

	MEANING (AMF)	SIGNIFICATO (SA)
0	C01 Reset maintenance interval 1	C01 Reset maintenance interval 1
1	C02 Reset maintenance interval 2	C02 Reset maintenance interval 2
2	C03 Reset maintenance interval 3	C03 Reset maintenance interval 3
3	C04 Reset engine partial hour counter	C04 Reset engine partial hour counter
4	C05 Reset mains partial counter.	C06 Reset generator partial counter.
5	C06 Reset generator partial counter.	C07 Reset generic counters CNTx
6	C07 Reset generic counters CNTx	C08 Reset High/ low
7	C08 Reset High/ low	C09 Reset engine total hour counter
8	C09 Reset engine total hour counter	C10 Engine hour counter settings
9	C10 Engine hour counter settings	C11 Reset no. starts counter
10	C11 Reset no. starts counter	C12 Reset starting counter
11	C12 Reset starting counter	C13 Reset closing counters
12	C13 Reset closing counters	C015 Reset generator total counter.
13	C14 Reset mains total counter.	C16 Reload rent hours
14	C015 Reset generator total counter.	C17 Reset events list
15	C16 Reload rent hours	C18 Reset default parameters
16	C17 Reset events list	C19 Save parameters in backup memory
17	C18 Reset default parameters	C20 Reload parameters from backup memory
18	C19 Save parameters in backup memory	C21 Fuel purge
19	C20 Reload parameters from backup memory	C22 Forced I/O
20	C21 Fuel purge	C23 Resistive sensors offset regulation
21	C22 Forced I/O	C24 Reset PLC program
22	C23 Resistive sensors offset regulation	C25 Sleep mode
23	C24 Reset PLC program	
24	C25 Sleep mode	

**TABELLA 6:
OROLOGIO DATARIO**

(Utilizzabili con funzioni 04 e 06)

Per rendere effettivi i cambiamenti, memorizzare le impostazioni utilizzando l'apposito comando descritto nella tabella 3.

**TABLE 6:
REAL TIME CLOCK**

(To be used with functions 04 and 06)

To make effective the changes, store them using the dedicated command described in table 3.

Indirizzo Address	WORD S	FUNZIONE	FUNCTION	RANGE	RGK 800 AMF	RGK 800 SA	RGK 700 AMF	RGK 700 SA
28F0H	1	Anno	Year	2000..2099	•	•	•	•
28F1H	1	Mese	Month	1-12	•	•	•	•
28F2H	1	Giorno	Day	1-31	•	•	•	•
28F3H	1	Ora	Hours	0-23	•	•	•	•
28F4H	1	Minuti	Minutes	0-59	•	•	•	•
28F5H	1	Secondi	Seconds	0-59	•	•	•	•

LETTURA LISTA EVENTI

Per leggere gli eventi bisogna svolgere la seguente procedura:

1. Eseguire la lettura di 1 registro con la funzione 4 all'indirizzo 5030H, il byte più significativo (msb) indica quanti eventi sono memorizzati (valore compreso tra 0 a 250), il byte meno significativo viene incrementato ogni volta che un evento viene salvato (valore compreso tra 0 a 250). Una volta memorizzati 250 eventi l'msb resterà a 250 mentre l'lsb tornerà a zero e poi continuerà ad incrementare.
2. Impostare l'indice dell'evento che si vuole leggere (minore del numero massimo di eventi memorizzati), per fare questo bisogna eseguire la funzione 6 all'indirizzo 5030H, specificando quale evento leggere.
3. Eseguire una lettura di 43 registri (con un'unica funzione 4) all'indirizzo 5032H
4. Il valore tornato è una stringa di 86 caratteri ASCII, che riportano la stessa descrizione dell'evento visibile sul display dell'RGK. L'indice dell'evento che si vuole leggere viene incrementato in automatico dopo la lettura del registro 5032H, al fine di velocizzare il download degli eventi
5. Se si vuole leggere l'evento successivo eseguire il punto 3, se si vuole leggere un qualsiasi altro evento eseguire il passo 2.

Vedere esempio

EVENT LOG READING

To read the events must do the following:

1. Perform the read of 1 register by using the function 4 at address 5030H, the most significant byte (msb) indicates how many events are stored (value between 0 to 250), the least significant byte (lsb) is incremented each time an event is saved (value between 0 to 250). Once stored the 250 events the msb will remain at 250 while the lsb will back to zero and after will continue to increase.
2. Set the index of the event that you want to read (less than the maximum number of events stored), to do this you perform the function 6 at 5030H, specifying which event read.
3. Perform a read of 43 registers (with a single function 4) at address 5032H
4. The value returned is a string of 86 ASCII characters, showing the same event description RGK visible on the display. The index of the event to be read is incremented automatically after a reading of the register 5032H, in order to speed up the download of events
5. If you want to read the next event performing step 4, if you want to read any other event do step 3.

See the example

ESEMPIO / EXAMPLE

Passo 1 : Lettura eventi memorizzati.

Step 1 : Reading events stored.

MASTER	Funzione / Function	= 4 (04H)								
	Indirizzo / Address	= 5030H (5030H – 0001H =502FH)								
	Nr. registri / Nr. registers	= 1 (01H)								
		<table border="1"> <tr> <td>01</td><td>04</td><td>50</td><td>2F</td><td>00</td><td>01</td><td>11</td><td>03</td> </tr> </table>	01	04	50	2F	00	01	11	03
01	04	50	2F	00	01	11	03			
RGK	Funzione / Function	= 4								
	Nr. byte / Nr. bytes.	= 1 (01H)								
	MSB	= 250 (FAH)								
	LSB	= 2 (02H)								
		<table border="1"> <tr> <td>01</td><td>04</td><td>02</td><td>FA</td><td>02</td><td>7A</td><td>51</td> </tr> </table>	01	04	02	FA	02	7A	51	
01	04	02	FA	02	7A	51				

Passo 2 : Impostare l'indice dell'evento da leggere.

Step 2 : Set the index of the event to read.

MASTER	Funzione / Function	= 6(06H)								
	Indirizzo / Address	= 5030H (5030H – 0001H =502FH)								
	Valore / Value	= 1 (01H)								
		<table border="1"> <tr> <td>01</td><td>06</td><td>50</td><td>2F</td><td>00</td><td>01</td><td>68</td><td>C3</td> </tr> </table>	01	06	50	2F	00	01	68	C3
01	06	50	2F	00	01	68	C3			
RGK	Funzione / Function	= 6								
	Indirizzo / Address	= 5030H (5030H – 0001H =502FH)								
	Valore / Value	= 1 (01H)								
		<table border="1"> <tr> <td>01</td><td>06</td><td>50</td><td>2F</td><td>00</td><td>01</td><td>68</td><td>C3</td> </tr> </table>	01	06	50	2F	00	01	68	C3
01	06	50	2F	00	01	68	C3			

Passo 3 : Leggere l'evento.

Step 3 : Read the event.

MASTER	Funzione / Function	= 4 (04H)																																																																																																			
	Indirizzo / Address	= 5032H (5032H – 0001H =5031H)																																																																																																			
	Nr. registri / Nr. registers	= 43 (2BH)																																																																																																			
		<table border="1"> <tr> <td>01</td><td>04</td><td>50</td><td>31</td><td>00</td><td>2B</td><td>F0</td><td>DA</td> </tr> </table>	01	04	50	31	00	2B	F0	DA																																																																																											
01	04	50	31	00	2B	F0	DA																																																																																														
RGK	Funzione / Function	= 4 (04H)																																																																																																			
	Indirizzo / Address	= 5030H (5030H – 0001H =502FH)																																																																																																			
	Nr. byte / Nr. bytes	= 86 (56H)																																																																																																			
	Stringa / String	= 2012/07/18;09:34:52;E1100,CAMBIO MODALITÀ IN: MODALITÀ OFF																																																																																																			
		<table border="1"> <tr> <td>01</td><td>04</td><td>56</td><td>32</td><td>30</td><td>31</td><td>32</td><td>2F</td><td>30</td><td>37</td><td>31</td><td>42</td><td>3B</td><td>30</td><td>39</td><td>3A</td><td>33</td><td>34</td><td>3A</td> </tr> <tr> <td>35</td><td>32</td><td>3B</td><td>45</td><td>31</td><td>31</td><td>30</td><td>30</td><td>2C</td><td>43</td><td>41</td><td>4D</td><td>42</td><td>49</td><td>4F</td><td>20</td><td>4D</td><td>4F</td><td>44</td><td>41</td> </tr> <tr> <td>4C</td><td>49</td><td>54</td><td>C1</td><td>20</td><td>49</td><td>4E</td><td>3A</td><td>20</td><td>4D</td><td>4F</td><td>44</td><td>41</td><td>4C</td><td>49</td><td>54</td><td>C1</td><td>20</td><td>4F</td><td>46</td> </tr> <tr> <td>46</td><td>20</td><td>20</td><td>20</td><td>20</td><td>20</td><td>20</td><td>20</td><td>20</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td> </tr> <tr> <td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>00</td><td>E5</td><td>78</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </table>	01	04	56	32	30	31	32	2F	30	37	31	42	3B	30	39	3A	33	34	3A	35	32	3B	45	31	31	30	30	2C	43	41	4D	42	49	4F	20	4D	4F	44	41	4C	49	54	C1	20	49	4E	3A	20	4D	4F	44	41	4C	49	54	C1	20	4F	46	46	20	20	20	20	20	20	20	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	E5	78									
01	04	56	32	30	31	32	2F	30	37	31	42	3B	30	39	3A	33	34	3A																																																																																			
35	32	3B	45	31	31	30	30	2C	43	41	4D	42	49	4F	20	4D	4F	44	41																																																																																		
4C	49	54	C1	20	49	4E	3A	20	4D	4F	44	41	4C	49	54	C1	20	4F	46																																																																																		
46	20	20	20	20	20	20	20	20	00	00	00	00	00	00	00	00	00	00	00																																																																																		
00	00	00	00	00	00	00	00	00	E5	78																																																																																											

IMPOSTAZIONE PARAMETRI

Tramite il protocollo Modbus® e' possibile accedere ai parametri dei menu.

Per interpretare correttamente la corrispondenza fra valore numerico e funzione selezionata e/o unita' di misura, fare riferimento al manuale operativo del RGK.

PROCEDURA PER LA LETTURA DEI PARAMETRI

1. Scrivere il valore del menu che si vuole leggere tramite la funzione 6 all'indirizzo 5000H ❶.
2. Scrivere il valore del sottomenu (se esiste) che si vuole leggere tramite la funzione 6 all'indirizzo 5001H ❶.
3. Scrivere il valore del parametro che si vuole leggere tramite la funzione 6 all'indirizzo 5002H ❶.
4. Eseguire la funzione 4 all'indirizzo 5004H, di un numero di registri appropriato alla lunghezza del parametro (vedi tabella).
5. Se si vuole leggere il parametro successivo, (all'interno dello stesso menu/sottomenu) ripetere il passo 4, altrimenti eseguire il passo 1.

PROCEDURA PER LA SCRITTURA DEI PARAMETRI

1. Scrivere il valore del menu che si vuole modificare tramite la funzione 6 all'indirizzo 5000H ❶.
2. Scrivere il valore del sottomenu (se esiste) che si vuole modificare tramite la funzione 6 all'indirizzo 5001H ❶.
3. Scrivere il valore parametro che si vuole modificare tramite la funzione 6 all'indirizzo 5002H ❶.
4. Eseguire la funzione 16 all'indirizzo 5004H, di un numero di registri appropriato alla lunghezza del parametro.
5. Se si vuole scrivere il parametro successivo, all'interno dello stesso menu/sottomenu ripetere il passo 4, altrimenti eseguire il passo 1, se non bisogna scrivere ulteriori parametri eseguire il passo 6.
6. Per rendere effettivo un cambiamento nel menu di setup e' necessario memorizzare i valori in EEPROM, utilizzando l'apposito comando descritto nella tabella 3.(scrivere il valore 4 con la funzione 6 all' indirizzo 2F03H)

TIPO DI PARAMETRO	NUMERO REGISTRI
Testo lunghezza 6 caratteri (es. M25.01.06)	3 registri (6 byte)
Testo lunghezza 16 caratteri (es. M25.01.05)	8 registri (16 byte)
Testo lunghezza 20 caratteri (es. M01.10)	10 registri (20 byte)
Valore numerico < 32768 (es M01.07)	1 registri (2 byte)
Valore numerico > 32768 (es M02.08)	2 registri (4 byte)
Indirizzo IP (es. M20.0x.06 M20.0x.07)	2 registri (4 byte)

❶ E' possibile leggere il valore del menu, sottomenu e parametro memorizzati agli indirizzi 5000H,5001H e 5002H utilizzando la funzione 4

Vedere esempio

PARAMETER SETTING

Using the Modbus® protocol it is possible to access the menu parameters.

To correctly understand the correspondence between the numeric value and the selected function and/or the unit of measure, please see the RGK operating manual.

PROCEDURE FOR THE READING OF PARAMETERS

1. Write the value of the menu that you want to read by using the function 6 at address 5000H ❶.
2. Write the value of the submenu (if it is present) that you want to read by using the function 6 at address 5001H ❶.
3. Write the value of the parameter that you want to read by using the function 6 at address 5002H ❶.
4. Perform the function 4 at the address 5004H, with a number of registers appropriate to the length of the parameter (see table).
5. If you want to read the next parameter (in the same menu/submenu) repeat step 4, otherwise perform step 1.

PROCEDURE FOR THE WRITING OF PARAMETERS

1. Write the value of the menu that you want to change by using the function 6 at address 5000H ❶.
2. Write the value of the submenu (if it is present) that you want to change by using the function 6 at address 5001H ❶.
3. Write the value of the parameter that you want to change by using the function 6 at address 5001H ❶.
4. Perform the function 16 at address 5004H, with a number of registers appropriate to the length of the parameter
5. If you want to write the next parameter, in the same menu / submenu repeat step 4, otherwise perform step 1, if you do not have to write additional parameters go to step 6.
6. To make effective the changes made to setup parameters it is necessary to store the values in EEPROM, using the dedicated command described in table 3.(write value 4 by using function 6 at address 2F03H)

TYPE OF PARAMETER	NUMBER OF REGISTER
Text length 6 characters (ex. M25.01.06)	3 registers (6 byte)
Text length 16 characters (ex. M25.01.05)	8 registers (16 byte)
Text length 20 characters (ex. M01.10)	10 registers (20 byte)
Numeric value < 32768 (ex M01.07)	1 registers (2 byte)
Numeric value > 32768 (ex M02.08)	2 registers (4 byte)
IP address (ex. M20.0x.06 M20.0x.07)	2 registers (4 byte)

❶ It's possible to read the menu, submenus, and parameter stored at the addresses 5000H, 5001H and 5002H by using the function 4

See the example

ESEMPIO / EXAMPLE

Impostare a 230 il valore del parametro M04.02.01
 Set to 230 the value of parameter M04.02.01

Passo 1 :Impostazione menu 04.

Step 1 :Set menu 04.

MASTER Funzione / Function = 6
 Indirizzo / Address = 5000H (5000H – 0001H =4FFFH)
 Valore / Value = 4 (04H)

01	06	4F	FF	00	04	AE	ED
----	----	----	----	----	----	----	----

RGK Funzione / Function = 6
 Indirizzo / Address = 5000H (5000H – 0001H =4FFFH)
 Valore / Value = 4 (04H)

01	06	4F	FF	00	04	AE	ED
----	----	----	----	----	----	----	----

Passo 2 :Impostazione sottomenu 02.

Step 2 :Set submenu 02.

MASTER Funzione / Function = 6
 Indirizzo / Address = 5001H (5001H – 0001H =5000H)
 Valore / Value = 2 (02H)

01	06	50	00	00	02	19	0B
----	----	----	----	----	----	----	----

RGK Funzione / Function = 6
 Indirizzo / Address = 5001H (5001H – 0001H =5000H)
 Valore / Value = 2 (02H)

01	06	50	00	00	02	19	0B
----	----	----	----	----	----	----	----

Passo 3 :Impostazione parametro 01.

Step 3 :Set parameter 01.

MASTER Funzione / Function = 6
 Indirizzo / Address = 5002H (5002H – 0001H =5001H)
 Valore / Value = 1 (01H)

01	06	50	01	00	01	08	CA
----	----	----	----	----	----	----	----

RGK Funzione / Function = 6
 Indirizzo / Address = 5002H (5002H – 0001H =5001H)
 Valore / Value = 2 (02H)

01	06	50	01	00	01	08	CA
----	----	----	----	----	----	----	----

Passo 3 :Impostazione valore 230.

Step 3 :Set value 230.

MASTER Funzione / Function = 16 (10H)
 Indirizzo / Address = 5004H (5004H – 0001H =5003H)
 Nr. registri / Nr. register = 2 (02H)
 Nr. byte / Nr. bytes = 4 (04H)
 Valore / Value = 230 (000000E6H)

01	10	50	03	00	02	04	00	00	00	E6	CE	33
----	----	----	----	----	----	----	----	----	----	----	----	----

RGK Funzione / Function = 16 (10H)
 Indirizzo / Address = 5004H (5004H – 0001H =5003H)
 Valore / Value = 2 (02H)

01	10	50	03	00	02	A0	C8
----	----	----	----	----	----	----	----

Passo 6 : Salvataggio e riavvio.

Step 6 : Save and reboot.

MASTER Funzione / Function = 6 (06H)
 Indirizzo / Address = 2F03H (2F03H – 0001H =2F02H)
 Valore / Value = 4 (04H)

01	6	2F	02	00	04	21	1D
----	---	----	----	----	----	----	----

RGK Nessuna risposta/No answer