

**TERMOREGOLATORE PROGRAMMABILE
PROGRAMMABLE TEMP. CONTROLLER****TM9x****INTERFACCIA SERIALE MOD-BUS****INTRODUZIONE**

Questa relazione intende descrivere il funzionamento dell'interfaccia seriale degli strumenti TM9x. Verranno descritti il protocollo e le sequenze di caratteri necessarie al colloquio con lo strumento. Nella relazione si chiamerà "terminale" l'apparecchio intelligente collegato allo strumento, che potrà essere, ad esempio, un personal computer, un PLC o una qualunque macchina in grado di gestire le comunicazioni via seriale.

Le interfacce seriali disponibili sono due: lo standard V.24 (RS232C) e lo standard RS485.

Sebbene lo standard RS232C permetta la trasmissione full-duplex, la comunicazione prevede una risposta dello strumento dopo l'interpretazione di una richiesta del terminale; è quindi evidente che la trasmissione sarà in pratica di tipo half-duplex.

L'interfaccia RS485 consente il collegamento di più strumenti con il terminale, usando un semplice doppino per tale collegamento. Per questo tipo di interfaccia, però, il terminale dovrà gestire correttamente la commutazione trasmissione-ricezione, in quanto il canale RS485 così realizzato è, per definizione, di tipo half-duplex. Il ritardo tra la fine della richiesta e l'inizio della risposta è di circa 15 ms.

PARAMETRI DI CONFIGURAZIONE

I parametri a bordo regolatore che controllano le comunicazioni seriali sono:

Prt *OFF*: Interfaccia seriale disabilitata

Mdb: Interfaccia seriale abilitata con protocollo di comunicazione ModBus RTU

Quando questo parametro è impostato *Mdb*, compaiono anche i seguenti parametri

Ind 1 - 99: Numero di identificazione dello strumento

bdr Velocità di trasmissione: 3, 6, 12, 24, 48, 96 che significano rispettivamente 300, 600, 1200, 2400, 4800, 9600 baud

AnS *LOC*: La tastiera dello strumento è completamente operativa mentre il terminale può solo leggere dati e parametri, ma non può modificarli (fatta eccezione per i parametri **AnS** e **kEy**)

rEM: La tastiera dello strumento permette di vedere i parametri, ma non ne permette la modifica (fatta eccezione per i parametri **AnS** e **kEy**) mentre il terminale può leggere e modificare tutti i parametri.

PROTOCOLLO DI COMUNICAZIONE HARDWARE

La trasmissione è di tipo seriale asincrona half-duplex (lo strumento che sta ricevendo non trasmette, mentre se sta trasmettendo non riceve dati), con 1 bit di start, 8 bits per carattere, nessun bit di parità ed 1 bit di stop; la velocità della comunicazione può essere impostata cambiando un parametro dello strumento e può essere di 300, 600, 1200, 2400, 4800 o 9600 baud. Il canale di comunicazione è lo standard RS232 o lo standard RS485 a seconda della scheda installata nello strumento.

PROTOCOLLO DI COMUNICAZIONE SOFTWARE

E' stato preso come riferimento il protocollo MOD-BUS standard Modicon. Le funzioni MOD-BUS riconosciute sono la 3, la 4 e la 6 (N.B. la 3 e la 4 vengono interpretate come se fossero lo stesso comando, cioè *words reading*; la 6 è la funzione *single word writing*).

Ci sono due differenze tra il protocollo implementato sui TM9x ed il protocollo standard:

- 1) non è stata prevista la funzione di *broadcasting*, cioè non è previsto che un comando mandato all'indirizzo 0 venga riconosciuto da tutti gli strumenti in rete, indipendentemente dal parametro **Ind** impostato nei vari strumenti;
- 2) se il terminale manda un codice funzione diverso da quelli riconosciuti (3, 4 e 6), lo strumento manda un error reply con codice di errore 1.

Tutti i dati vengono trasmessi in modo binario.

La condizione di START è riconosciuta quando il ritardo tra due caratteri consecutivi trasmessi supera 1 T.U. (Time Unit = tempo necessario a trasmettere 1 carattere).

Una richiesta del terminale ha sempre il seguente formato:

- 1° byte: indirizzo strumento (per selezionare, fra molti, lo strumento desiderato)
- 2° byte: codice funzione (3, 4 o 6)
- 3° byte: byte più significativo della word da leggere o da scrivere
- 4° byte: byte meno significativo della word da leggere o da scrivere
- 5° byte: lettura: parte alta del numero di words da leggere; scrittura: parte alta del valore da scrivere
- 6° byte: lettura: parte bassa del numero di words da leggere; scrittura: parte bassa del valore da scrivere
- 7° byte: parte bassa del CRC-16
- 8° byte: parte alta del CRC-16.

Il calcolo del CRC-16 bit viene effettuato secondo le specifiche del MOD-BUS.

I valori dei parametri verranno sempre espressi sotto forma binaria a 16 bit con segno: ciò vuol dire che si potranno avere valori compresi tra -32768 e +32767. Bisognerà perciò porre attenzione al byte alto trasmesso, perché se il suo valore supera 127 (7Fhex), il valore finale del parametro sarà negativo, pari a $((\text{byte alto} * 256) + \text{byte basso}) - 65536$.

Funzione '3' o '4': words reading

La risposta dello strumento alla funzione di lettura consisterà in un frame di $(5 + 2N)$ bytes, dove N è il numero di words richieste ed il cui significato è:

- 1° byte: indirizzo strumento
- 2° byte: codice funzione (3 o 4)
- 3° byte: numero di byte per i dati trasmessi (pari a $2N$, dove N è il numero di words richieste)
- 4°, 5°: valore della prima word richiesta (byte alto e basso rispettivamente)
- 6°, 7°: valore della seconda word richiesta
- ...
- $(2+2N)^\circ, (3+2N)^\circ$: valore della N-esima word richiesta
- $(4+2N)^\circ, (5+2N)^\circ$: CRC-16 (byte basso e alto rispettivamente)

Funzione 6: single word writing

La risposta dello strumento alla funzione di scrittura consisterà in un frame di 8 bytes:

- 1° byte: indirizzo strumento
- 2° byte: codice funzione
- 3°, 4° byte: indirizzo word da scrivere (byte alto e byte basso rispettivamente)
- 5°, 6° byte: valore scritto (byte alto e byte basso rispettivamente)
- 7°, 8° byte: CRC-16 (byte basso e byte alto rispettivamente)

Errore in risposta - error reply

Se il comando non può andare a buon fine, viene ritornato al terminale un frame di 5 bytes che specificano che tipo di errore si è verificato:

- 1° byte: indirizzo strumento
- 2° byte: codice funzione con il bit più significativo settato (codice funzione + 80hex)
- 3° byte: codice errore
- 4°, 5° byte: CRC-16 (byte basso e byte alto rispettivamente)

CODICI ERRORI

Valore Errore associato

- 1 Codice comando non riconosciuto
- 2 Indirizzo illegale
- 3 Valore illegale
- 9 Numero di dati richiesti illegale
- 10 Dato protetto in scrittura

LOCAZIONI LETTURA / SCRITTURA

Indirizzo		Letture	Scrittura
0x0001	Offset *	Si	Si
0x0002	Chiave tastiera (0=OFF , 1=Lo , 2=Hi)	Si	Si
0x0100	Banda proporzionale riscaldamento *	Si	Si
0x0101	Tempo derivativo risc. (o isteresi superiore) *	Si	Si
0x0102	Tempo integrale risc. (o isteresi inferiore) *	Si	Si
0x0103	Tempo di ciclo risc. *	Si	Si
0x0300	Set point main *	Si	Si
0x1000	Variabile di processo *	Si	No

* = per limiti vedi manuale programmazione

Si noti che la scrittura è possibile solo se in accordo con i parametri di blocco tastiera "kEy" e di modo seriale "AnS". In particolare se "AnS" è impostato su "rEM" allora i parametri sono impostabili via seriale, ma sono bloccati da tastiera. Al contrario, in modalità "LoC", si possono modificare da tastiera, ma non via seriale.

Esempio di lettura

Si supponga di voler leggere i 4 parametri relativi alla regolazione di riscaldamento :

- banda prop [100%],
- tempo der [10'],
- tempo int [4'],
- tempo di ciclo [10"])

dello strumento che ha per indirizzo 04.

L'indirizzo di partenza delle letture è 0x0100, le words da leggere sono 4.

la sequenza di bytes da spedire allo strumento è:

```
id      com  < address > < words > < CRC  >
0x04  0x03  0x01  0x00  0x00  0x04  0x45  0xA0
```

La risposta dello strumento sarà:

```
id      com  bytes < P.B.  > < t der > < t int > < t ciclo > < CRC  >
0x04  0x03  0x08  0x00  0x64  0x00  0x0A  0x00  0x04  0x00  0x0A  0xF8  0x1A
```

Quindi, per questo esempio, la banda proporzionale è di 100 punti (0x0064), il tempo derivativo è di 10 punti (0x000A), il tempo integrale è di 4 punti (0x0004), il tempo di ciclo è di 10 punti (0x000A).

Esempio di scrittura

L'unico comando di scrittura è il comando "0x06", che è il comando di scrittura per una singola word. Non è prevista la scrittura di più parametri con un solo comando.

Si supponga di voler scrivere il set point del main con il valore 150 sullo stesso strumento di prima (id = 4). La sequenza di bytes da spedire allo strumento è:

```
id      com  < address > < dato  > < CRC  >
0x04  0x06  0x03  0x00  0x00  0x96  0x09  0xB5
```

La risposta dello strumento sarà:

```
id      com  < address > < dato  > < CRC  >
0x04  0x06  0x03  0x00  0x00  0x96  0x09  0xB5
```



THERMOSYSTEMS s.r.l. Via delle Industrie, 8 - 24040 Forno San Giovanni (BG) - ITALY

phone: (+39) 0363 350159 fax: (+39) 0363 350362 web: www.thermosystems.it e-mail: info@thermosystems.it