

INTERFACCIA SERIALE ASCII – MODBUS

NOTE E DEFINIZIONI COMUNI AI DUE PROTOCOLLI

Questa relazione intende descrivere il funzionamento dell'interfaccia seriale degli strumenti DM50 (1/16DIN=48x48mm) e DM500 (1/8DIN=96x48mm). Verranno descritti i protocolli e le sequenze di caratteri necessarie al colloquio con lo strumento. Nella trattazione si chiamerà "terminale" l'apparecchio intelligente collegato allo strumento, che potrà essere, ad esempio, un personal computer, un pocket PC, un terminale, un PLC o una qualunque macchina in grado di gestire le comunicazioni via seriale.

*NOTA BENE: Da qui in avanti ci riferiremo al DM500 sottintendendo anche lo strumento DM50.
Eventuali differenze saranno di volta in volta espressamente specificate.
Si noti che le diciture relative ai nomi di famiglie, gruppi o parametri, e gli mnemonici di alcuni dei parametri stessi sono riportati per esteso così come appaiono sullo strumento DM500.
Per il DM50 tali diciture risultano ovviamente abbreviate, ma comunque facilmente identificabili.*

Le interfacce seriali disponibili sono due (mutualmente esclusive):

- lo standard V.24 (RS232C)
- lo standard RS485.

Per strumenti tipo DM50 l'interfaccia RS232 consente di realizzare un collegamento punto-punto mentre per strumenti tipo DM500 l'interfaccia RS232 consente di realizzare un collegamento a catena di più strumenti, in quanto il segnale proveniente dal terminale è bufferato e ritrasmesso su un canale di uscita (che potrà quindi essere collegato ad un altro strumento), mentre il segnale diretto verso il terminale si inserisce sulla linea eventualmente collegata ad altri strumenti (l'interfaccia è progettata in modo da evitare conflitti hardware di trasmissione). Sebbene lo standard RS232C e l'interfaccia stessa permettano la trasmissione full-duplex, la comunicazione prevede una risposta dello strumento solo dopo l'interpretazione di una richiesta da parte del terminale; è quindi evidente che la trasmissione sarà in pratica di tipo half-duplex. L'interfaccia RS232 sarà operativa con protocollo ASCII oppure MODBUS (mutualmente esclusivi) in accordo alla sigla di ordinazione dell'apparecchio.

L'interfaccia RS485 consente anch'essa il collegamento di più strumenti con il terminale, ma con il vantaggio di usare 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 minimo di commutazione tra ricezione e trasmissione degli strumenti è impostabile da 1 a 10 ms. L'interfaccia RS485 sarà operativa con protocollo ASCII oppure MODBUS (mutualmente esclusivi) in accordo alla sigla di ordinazione dell'apparecchio.

NOTE PER IL COLLEGAMENTO HARDWARE DEL DM50x AD UN PERSONAL COMPUTER

Si tenga presente che lo strumento suppone di comunicare con il terminale senza usare nessun tipo di handshake software o hardware: per cui sarà necessario cortocircuitare il segnale RTS con CTS ed il segnale DTR con DSR dell'interfaccia seriale del Personal Computer, per evitare che alcuni sistemi operativi blocchino la comunicazione in attesa dei segnali CTS o DSR, che normalmente arrivano in risposta dai dispositivi collegati.

Interfaccia Seriale ASCII

PARAMETRI DI CONFIGURAZIONE

Nella famiglia **ConFiG** si trova il gruppo di parametri che impostano e regolano la comunicazione seriale, indicato con '**rS CoM**'; i parametri disponibili sono:

Protoc	<i>OFF</i> : Interfaccia seriale disabilitata ASCII: Interfaccia seriale abilitata con protocollo di comunicazione software 'ASCII'
Addr	1 - 255: Numero di identificazione dello strumento
bAud	Velocità di trasmissione: 300, 600, 1200, 2400, 4800, 9600 baud
ModE	<i>LocAL</i> : 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 kEyLok del gruppo diSPL e ModE del gruppo rS CoM) <i>rEMotE</i> : La tastiera dello strumento permette di vedere i parametri, ma non ne permette la modifica (fatta eccezione per i parametri kEyLok del gruppo diSPL e ModE del gruppo rS CoM) mentre il terminale può leggere e modificare tutti i parametri.
dELAY	Ritardo alla risposta: 1, 2, 3, 4, 5, 6, 8, 10 ms

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 protocollo di comunicazione è lo standard RS232 o lo standard RS485 a seconda della scheda installata nello strumento.

RICHIESTA DI UN PARAMETRO DA PARTE DEL TERMINALE

La sequenza di caratteri da mandare allo strumento è la seguente:

- 1) carattere di start (ascii STX, 2 dec, 0x02 hex)
- 2) primo carattere identificativo strumento (tradotto in esadecimale)
- 3) secondo carattere identificativo strumento (tradotto in esadecimale)
- 4) comando di lettura (ascii 'R', 82 dec, hex. 0x52 hex)
- 5) primo carattere identificativo della locazione da leggere (tradotto in esadecimale)
- 6) secondo carattere identificativo della locazione da leggere (tradotto in esadecimale)
- 7) carattere di stop (ascii ETX, 3 dec, 0x03 hex)
- 8) carattere di controllo, calcolato come XOR di tutti i precedenti caratteri

Esempio: supponiamo che lo strumento abbia come indirizzo 123 dec, e che si voglia leggere il parametro 'SEt' dell'allarme 1.

L'indirizzo 123 dec corrisponde, in esadecimale, al valore 0x7B; inoltre dalla tabella locazioni si vede che il parametro 'SEt' dell'allarme 1 è accessibile alla locazione esadecimale 0x25.

Per cui i caratteri da mandare saranno:

ascii	dec.	hex
STX	2	02
'7'	55	37
'B'	66	42
'R'	82	52
'2'	50	32
'5'	53	35
ETX	3	03
	33	21

RISPOSTA DELLO STRUMENTO AD UNA RICHIESTA DI UN PARAMETRO

Supponendo, ad esempio, che il valore 'SEt 1' richiesto sia 8542 dec, lo strumento risponderà con la sequenza di caratteri:

- 1) carattere di start (ascii STX, 2 dec, 0x02 hex)
- 2) segno, che sarà il carattere '+' per valori positivi (come in questo caso) o '-' per valori negativi
- 3, 4, 5, 6, 7) cifre che compongono il valore desiderato, in questo caso '08542'
- 8) carattere di stop (ascii ETX, 3 dec, 0x03 hex)
- 9) carattere di controllo, calcolato come XOR di tutti i precedenti caratteri

I caratteri che lo strumento manderà saranno quindi:

ascii	dec.	hex
STX	2	02
'+'	43	2B
'0'	48	30
'8'	56	38
'5'	53	35
'4'	52	34
'2'	50	32
ETX	3	03
	17	11

Nota per DM50: Se lo strumento è un DM50, la prima cifra del valore resterà sempre a 0. Ad esempio, se il valore che lo strumento deve trasmettere è pari a 1845, lo strumento manterrà il formato a 5 cifre (per la compatibilità col DM500), ma con la prima forzata a 0. Quindi si avrà la sequenza di caratteri STX, '+', '0', '1', '8', '4', '5' ETX e Check Value calcolato come per il DM500, cioè XOR di tutti i caratteri precedenti (compresi il carattere forzato a '0').

RICHIESTA DI SETTAGGIO DI UN PARAMETRO DA PARTE DEL TERMINALE

La sequenza di caratteri da mandare allo strumento è la seguente:

- 1) carattere di start (ascii STX, 2 dec, 0x02 hex)
- 2) primo carattere identificativo strumento (tradotto in esadecimale)
- 3) secondo carattere identificativo strumento (tradotto in esadecimale)
- 4) comando di scrittura (ascii 'W', 87 dec, hex. 0x57 hex)
- 5) primo carattere identificativo della locazione da scrivere (tradotto in esadecimale)
- 6) secondo carattere identificativo della locazione da scrivere (tradotto in esadecimale)
- 7) carattere ascii '=' (61 dec, 0x3d hex)
- 8) segno, che sarà il carattere '+' per valori positivi o '-' per valori negativi
- 9, 10, 11, 12, 13) cifre che compongono il valore desiderato
- 14) carattere di stop (ascii ETX, 3 dec, 0x03 hex)
- 15) carattere di controllo, calcolato come XOR di tutti i precedenti caratteri.

Esempio: supponiamo che lo strumento abbia come indirizzo 14 dec, e che si voglia settare il parametro 'SEt Lo' dell'allarme 4 con il valore -12502.

L'indirizzo 14 dec corrisponde, in esadecimale, al valore 0x0E; inoltre dalla tabella 'locazioni' si vede che il parametro 'SEt Lo' dell'allarme 4 è accessibile alla locazione 0x53 hex (83 dec).

Per cui i caratteri da mandare saranno:

ascii	dec.	hex

STX	2	02
'0'	48	30
'E'	69	45
'W'	87	57
'8'	53	35
'B'	51	33
'='	61	3D
'-'	45	2D
'1'	49	31
'2'	50	32
'5'	53	35
'0'	48	30
'2'	52	32
ETX	3	03
	1	01

Nota per DM50: Se lo strumento è un DM50, la prima cifra del valore resterà sempre a 0. Ad esempio, se il valore che si vuole memorizzare è -1250, lo strumento manterrà il formato a 5 cifre (per la compatibilità col DM500), ma con la prima forzata a 0. Quindi si avrà la sequenza di caratteri STX, '+', '0', '1', '2', '5', '0' ETX e Check Value calcolato come per il DM500, cioè XOR di tutti i caratteri precedenti (compresi il carattere forzato a '0').

RISPOSTA DELLO STRUMENTO AD UNA RICHIESTA DI SETTAGGIO DI UN PARAMETRO

Lo strumento risponderà con la sequenza di caratteri:

- 1) carattere di start (ascii STX, 2 dec, 0x02 hex)
- 2) carattere 'E' (69 dec, 0x45 hex)
- 3, 4) carattere '0' (48 dec, 0x30 hex)
- 5) codice errore, che potrà essere '0', '1', '2', e così via, secondo la 'tabella errori' sotto riportata
- 6) carattere di stop (ascii ETX, 3 dec, 0x03 hex)
- 7) carattere di controllo, calcolato come XOR di tutti i precedenti caratteri

Se non ci sono impedimenti alla memorizzazione del parametro specificato con il valore desiderato (cioè se il parametro 'ModE' del gruppo 'rS CoM' è impostato su 'rEMotE', il parametro non è protetto dalla chiave strumento ed il valore è all'interno dei limiti consentiti) lo strumento risponderà con la sequenza di caratteri

ascii	dec.	hex

STX	2	02
'E'	69	45
'0'	48	30
'0'	48	30
'0'	48	30
ETX	3	03
	116	74

TABELLA ERRORI

Lo strumento può rispondere con codici di errore diversi da 0; più precisamente si potranno avere i seguenti errori:

'E000'	Nessun errore: scrittura eseguita correttamente
'E001'	Comando non riconosciuto
'E002'	Errore in scrittura: valore del parametro fuori dai limiti permessi
'E003'	Errore in scrittura: il parametro in questione è protetto in scrittura
'E004'	Errore in lettura: il parametro in questione è protetto in lettura

INTERPRETAZIONE DELLE RISPOSTE

Molti dei parametri dello strumento non sono numeri puri, per cui il numero che viene trasmesso dallo strumento in risposta all'interrogazione del terminale va interpretato secondo le sigle mnemoniche che gli danno un significato preciso.

I valori possibili dei parametri mnemonici ed i loro corrispondenti significati sono riportati nelle tabelle 1, 2, ..., 27.

Interfaccia Seriale ModBus

PARAMETRI DI CONFIGURAZIONE

Nella famiglia **ConFIG** si trova il gruppo di parametri che impostano e regolano la comunicazione seriale, indicato con '**rS CoM**'; i parametri disponibili sono:

Protoc	<i>OFF</i> :	Interfaccia seriale disabilitata
	<i>Modbus</i> :	Interfaccia seriale abilitata con protocollo ModBus RTU
Addr	<i>1 - 255</i> :	Numero di identificazione dello strumento
bAud		Velocità di trasmissione: 300, 600, 1200, 2400, 4800, 9600 baud
ModE	<i>LocAL</i> :	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 kEyLok del gruppo diSPL e ModE del gruppo rS CoM)
	<i>rEMotE</i> :	La tastiera dello strumento permette di vedere i parametri, ma non ne permette la modifica (fatta eccezione per i parametri kEyLok del gruppo diSPL e ModE del gruppo rS CoM) mentre il terminale può leggere e modificare tutti i parametri.
dELAy		Ritardo alla risposta: 1, 2, 3, 4, 5, 6, 8, 10 ms

PROTOCOLLO DI COMUNICAZIONE HARDWARE

E' stato preso come riferimento il protocollo MODBUS standard Modicon in versione RTU (Remote Terminal Unit). 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 protocollo di comunicazione è lo standard RS232 o lo standard RS485 a seconda della scheda installata nello strumento.

PROTOCOLLO DI COMUNICAZIONE SOFTWARE

Con riferimento al protocollo MODBUS standard Modicon RTU le funzioni 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 DM50x 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 **Addr** impostato nei vari strumenti;
- 2) il numero di words che si possono richiedere con i comandi 3 e 4 è limitato a 1; cioè ogni operazione di lettura è relativa ad un solo parametro;
- 3) 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, es. 9600baud: 10bit/car / 9600bit/s = 1.04ms/car).

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 MODBUS.

I valori dei parametri verranno sempre espressi sotto forma binaria a 32 bit con segno: ciò vuol dire che si potranno avere valori compresi tra $-2.147.483.648$ (0x80000000) e $+2.147.483.647$ (0x7FFFFFFF). Bisognerà perciò porre attenzione al byte alto trasmesso, perché se il suo valore supera 127 (7FHex), il valore finale del parametro sarà negativo.

Funzione '3' o '4': words reading

La risposta dello strumento alla funzione di lettura consisterà in un frame di 9 bytes, 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 4N, dove N è il numero di words rich.)
- 4°, 5°, 6°, 7°: valore della word richiesta (dal byte più alto a quello più basso rispettivamente)
- 8°, 9°: 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 10 bytes:

- 1° byte: indirizzo strumento
- 2° byte: codice funzione
- 3°, 4° byte: indirizzo word da scrivere (byte alto e byte basso rispettivamente)
- 5°, 6°, 7°, 8°: valore scritto (dal byte più alto a quello più basso rispettivamente)
- 9°, 10° 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

ValoreErrore associato

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

Esempi

Esempio di lettura

Si supponga di voler leggere il parametro 'PrESEt' del contatore 1 da uno strumento che ha per indirizzo seriale '04'. Il parametro desiderato ha per indirizzo 1020Hex.

L'indirizzo di partenza delle letture è quindi 0x1020, il numero di words da leggere forzato a 1.

La sequenza di bytes da spedire allo strumento è:

```
id   com  < address > < words > < CRC >
0x04 0x03 0x10 0x20 0x00 0x01 0x81 0x55
```

La risposta dello strumento sarà:

```
id   com  bytes < msb  DATO   lsb > < CRC >
0x04 0x03 0x04 0x00 0x00 0x01 0xF4 0xAF 0x24
```

Quindi, per questo esempio, il PrESEt del contatore 1 è 0x000001F4Hex, cioè 500dec.

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 lo stesso parametro letto prima (cioè il PrESEt del contatore 1, locazione 0x1020) con il valore di 1000dec (cioè 0x000003E8Hex) sullo stesso strumento di prima (id = 4). La sequenza di bytes da spedire allo strumento è:

```
id   com  < address > < msb  DATO   lsb > < CRC >
0x04 0x06 0x10 0x20 0x00 0x00 0x03 0xe8 0xA4 0x11
```

La risposta dello strumento, nel caso dell'avvenuta scrittura, sarà identica alla richiesta fatta, cioè:

```
id   com  < address > < msb  DATO   lsb > < CRC >
0x04 0x06 0x10 0x20 0x00 0x00 0x03 0xe8 0xA4 0x11
```

Nel caso in cui, invece, i parametri di controllo scrittura (LocAL / rEMotE o kEyLok), oppure il valore del parametro stesso non rientri nel range permesso, lo strumento risponderà con un messaggio di errore di tipo "error reply".

Consigli di carattere generale

- in caso di problemi di risposta o di comportamento è spesso utile riprovare da una situazione certa di partenza e quindi caricare i parametri di default ecc. ecc.

- evitare nel modo più assoluto le situazioni di conflitto o di eventuale incertezza operativa. In linea di massima lo strumento gestisce autonomamente queste situazioni ma non sempre il comportamento (es. priorità e/o precedenze) segue una logica facilmente identificabile e tantomeno facilmente descrivibile.

- il progetto di riferimento che supporta questa famiglia di strumenti è costantemente sottoposto ad aggiunte, miglioramenti e revisioni. E' nostra primaria cura tenere regolarmente aggiornata la documentazione relativa che può essere oggetto di modifiche, anche importanti, senza preavviso.

Tabella locazioni lettura/scrittura parametri dello strumento

Gruppo	Parametro	Ascii Lettura	Ascii Scrittura	MdBus	Significato
InPUT	SEnSr	R00	W00	0x1000	Tab. 1
	InPLo	R01	W01	0x1001	numerico
	dISLo	R02	W02	0x1002	numerico
	InPHi	R03	W03	0x1003	numerico
	dISHi	R04	W04	0x1004	numerico
dISPL	OvEr	R05	W05	0x1005	numerico
	UndEr	R06	W06	0x1006	numerico
	OFSEt	R07	W07	0x1007	numerico
	dECIM	R08	W08	0x1008	Tab. 2
	rOUnd	R09	W09	0x1009	Tab. 3
	Unit	R0A	W0A	0x100A	Tab. 4
	tMOUt	R54	W54	0x1054	Tab. 5
	StorE	R55	W55	0x1055	Tab. 6
HIdE	R56	W56	0x1056	Tab. 6	
PEAk	vALUE	R0B	W0B	0x100B	Tab. 7
	tIME	R0C	W0C	0x100C	numerico
AdCnv	t Cnv	R0D	W0D	0x100D	numerico
	n Avg	R0E	W0E	0x100E	numerico
	SCOSt	R0F	W0F	0x100F	numerico
	tIME	R10	W10	0x1010	Tab. 8
kEyLk	LEvEL	R11	W11	0x1011	Tab. 9
rEtrS	SOUrC	R12	W12	0x1012	Tab. 10
	SPEd	R13	W13	0x1013	Tab. 11
	AnLo	R14	W14	0x1014	numerico
	OULo	R15	W15	0x1015	numerico
	AnHi	R16	W16	0x1016	numerico
	OUIHi	R17	W17	0x1017	numerico
	rSCOM	PrOtC	R18	W18	0x1018
Addr		R19	W19	0x1019	numerico
bAUd		R1A	W1A	0x101A	Tab. 13
MOdE		R1B	W1B	0x101B	Tab. 14
USLin	EnABL	R57	W57	0x1057	Tab. 6
	In 0	R58	W58	0x1058	numerico
	OU 0	R59	W59	0x1059	numerico
	In 1	R5A	W5A	0x105A	numerico
	OU 1	R5B	W5B	0x105B	numerico
	In 2	R5C	W5C	0x105C	numerico
	OU 2	R5D	W5D	0x105D	numerico
	In 3	R5E	W5E	0x105E	numerico
	OU 3	R5F	W5F	0x105F	numerico
	In 4	R60	W60	0x1060	numerico
	OU 4	R61	W61	0x1061	numerico
	In 5	R62	W62	0x1062	numerico
	OU 5	R63	W63	0x1063	numerico
	In 6	R64	W64	0x1064	numerico
	OU 6	R65	W65	0x1065	numerico
	In 7	R66	W66	0x1066	numerico
	OU 7	R67	W67	0x1067	numerico
	In 8	R68	W68	0x1068	numerico
	OU 8	R69	W69	0x1069	numerico
	In 9	R6A	W6A	0x106A	numerico
	OU 9	R6B	W6B	0x106B	numerico
	In 10	R6C	W6C	0x106C	numerico
OU 10	R6D	W6D	0x106D	numerico	
In 11	R6E	W6E	0x106E	numerico	

	OU 11	R6F	W6F	0x106F	numerico
	In 12	R70	W70	0x1070	numerico
	OU 12	R71	W71	0x1071	numerico
	In 13	R72	W72	0x1072	numerico
	OU 13	R73	W73	0x1073	numerico
	In 14	R74	W74	0x1074	numerico
	OU 14	R75	W75	0x1075	numerico
	In 15	R76	W76	0x1076	numerico
	OU 15	R77	W77	0x1077	numerico
	In 16	R78	W78	0x1078	numerico
	OU 16	R79	W79	0x1079	numerico
	In 17	R7A	W7A	0x107A	numerico
	OU 17	R7B	W7B	0x107B	numerico
	In 18	R7C	W7C	0x107C	numerico
	OU 18	R7D	W7D	0x107D	numerico
	In 19	R7E	W7E	0x107E	numerico
	OU 19	R7F	W7F	0x107F	numerico
ALrM1	SOUrC	R1C	W1C	0x101C	Tab. 15
	tyPE	R1D	W1D	0x101D	Tab. 16
	Inhib	R1E	W1E	0x101E	Tab. 6
	FunCt	R1F	W1F	0x101F	Tab. 17
	rELE	R20	W20	0x1020	Tab. 18
	rESEt	R21	W21	0x1021	Tab. 19
	rEFEr	R22	W22	0x1022	Tab. 20
	OndLy	R23	W23	0x1023	numerico
	OFdLy	R24	W24	0x1024	numerico
	SEt	R25	W25	0x1025	numerico
	Hy Hi	R26	W26	0x1026	numerico
	Hy Lo	R27	W27	0x1027	numerico
	SEtHi	R28	W28	0x1028	numerico
	SEtLo	R29	W29	0x1029	numerico
ALrM2	SOUrC	R2A	W2A	0x102A	vedi ALrM1
	tyPE	R2B	W2B	0x102B	"
	Inhib	R2C	W2C	0x102C	"
	FunCt	R2D	W2D	0x102D	"
	rELE	R2E	W2E	0x102E	"
	rESEt	R2F	W2F	0x102F	"
	rEFEr	R30	W30	0x1030	"
	OndLy	R31	W31	0x1031	numerico
	OFdLy	R32	W32	0x1032	numerico
	SEt	R33	W33	0x1033	numerico
	Hy Hi	R34	W34	0x1034	numerico
	Hy Lo	R35	W35	0x1035	numerico
	SEtHi	R36	W36	0x1036	numerico
	SEtLo	R37	W37	0x1037	numerico
ALrM3	SOUrC	R38	W38	0x1038	vedi ALrM1
	tyPE	R39	W39	0x1039	"
	Inhib	R3A	W3A	0x103A	"
	FunCt	R3B	W3B	0x103B	"
	rELE	R3C	W3C	0x103C	"
	rESEt	R3D	W3D	0x103D	"
	rEFEr	R3E	W3E	0x103E	"
	OndLy	R3F	W3F	0x103F	numerico
	OFdLy	R40	W40	0x1040	numerico
	SEt	R41	W41	0x1041	numerico
	Hy Hi	R42	W42	0x1042	numerico
	Hy Lo	R43	W43	0x1043	numerico
	SEtHi	R44	W44	0x1044	numerico
	SEtLo	R45	W45	0x1045	numerico
ALrM4	SOUrC	R46	W46	0x1006	vedi ALrM1
	tyPE	R47	W47	0x1007	"
	Inhib	R48	W48	0x1008	"
	FunCt	R49	W49	0x1009	"
	rELE	R4A	W4A	0x100A	"

rESEt	R4B	W4B	0x100B	“
rEFEr	R4C	W4C	0x100C	“
OndLy	R4D	W4D	0x100D	numerico
OFdLy	R4E	W4E	0x100E	numerico
SEt	R4F	W4F	0x100F	numerico
Hy Hi	R50	W50	0x1000	numerico
Hy Lo	R51	W51	0x1001	numerico
SEtHi	R52	W52	0x1002	numerico
SEtLo	R53	W53	0x1003	numerico

Tabella locazioni lettura/scrittura variabili operative DM500

Variabile	Ascii Lettura	Ascii Scrittura	ModBus	Significato
Situazione relé	RFF	WFF	0x20FF	Tab. 21
Situazione Leds	RFE	WFE	0x20FE	Tab. 22
Situazione display unità	RFD	WFD	0x20FD	Tab. 23
Situazione display decine	RFC	WFC	0x20FC	Tab. 23
Situazione display centinaia	RFB	WFB	0x20FB	Tab. 23
Situazione display migliaia	RFA	WFA	0x20FA	Tab. 23
Situazione display decine di migliaia	RF9	WF9	0x20F9	Tab. 23
Situazione display unità di misura	RF8	WF8	0x20F8	Tab. 23
Variabile d'ingresso	RF7	read only	0x20F7	numerico
Variabile d'ingresso filtrata	RF6	read only	0x20F6	numerico
Flags situazione strumento	RF5	read only	0x20F5	Tab. 24
Stato di errore dello strumento	RF4	read only	0x20F4	Tab. 25
Situazione allarmi	RF3	read only	0x20F3	Tab. 26
Situazione tasti	RF2	read only	0x20F2	Tab. 27
Caricamento parametri di Default	write only	W80		

Tabella locazioni lettura/scrittura variabili operative DM50

Variabile	Ascii Lettura	Ascii Scrittura	ModBus	Significato
Situazione relé	RFF	WFF	0x20FF	Tab. 42
Situazione Leds	RFE	WFE	0x20FE	Tab. 43
Situazione display unità superiori	RFD	WFD	0x20FD	Tab. 44
Situazione display decine sup.	RFC	WFC	0x20FC	Tab. 44
Situazione display centinaia sup.	RFB	WFB	0x20FB	Tab. 44
Situazione display migliaia sup.	RFA	WFA	0x20FA	Tab. 44
Situazione display unità inferiori	RF9	WF9	0x20F9	Tab. 44
Situazione display decine inf.	RF8	WF8	0x20F8	Tab. 44
Situazione display centinaia inf.	REF	WEF	0x20EF	Tab. 44
Situazione display migliaia inf.	REE	WEE	0x20EE	Tab. 44
Variabile d'ingresso	RF7	read only	0x20F7	numerico
Variabile d'ingresso filtrata	RF6	read only	0x20F6	numerico
Flags situazione strumento	RF5	read only	0x20F5	Tab. 24
Stato di errore dello strumento	RF4	read only	0x20F4	Tab. 25
Situazione allarmi	RF3	read only	0x20F3	Tab. 26
Situazione tasti	RF2	read only	0x20F2	Tab. 27
Caricamento parametri di Default	write only	W80		

NOTE N.B. : NOTE VALIDE SOLO PER VERSIONE ASCII

Scrivendo la locazione 0xFF (situazione relé) si forzano i relé per un tempo di 2 secondi. Scaduto questo tempo lo strumento riassume il controllo dei relé.

Scrivendo le locazioni da 0xFE (situazione leds) a 0xF8 (Situazione display unità di misura) si forzano le indicazioni del display interessato per un tempo di 2 secondi. Scaduto questo tempo lo strumento riassume il controllo del display interessato.

E' quindi evidente che se si vuole mantenere il controllo delle risorse per un lungo periodo di tempo, il terminale dovrà mandare i comandi appropriati prima dello scadere del tempo di timeout (refresh di scrittura).

Per caricare i parametri di default mandare il comando "W80=+00001"; verranno aggiornati tutti i parametri ad eccezione del gruppo RSCOM, in cui verrà solamente forzato "MOdE" su "LOCAL".

Per riarmare un relé in blocco (a causa del reset manuale) basta forzare a 0 il bit di blocco del relé interessato (locazione RFF, bits 4, 5, 6, 7).

Tabella 1

Valore	Significato
0	Termoresistenza Pt 100
1	Termoresistenza Pt 100 decimale
2	Termocoppia tipo J
3	Termocoppia tipo J decimale
4	Termocoppia tipo K
5	Termocoppia tipo K decimale
6	Termocoppia tipo L
7	Termocoppia tipo L decimale
8	Termocoppia tipo N
9	Termocoppia tipo N decimale
10	Termocoppia tipo T
11	Termocoppia tipo T decimale
12	Termocoppia tipo R
13	Termocoppia tipo S
14	Termocoppia tipo B
15	Ingresso lineare 50 mV dc
16	Ingresso lineare 1 V dc
17	Ingresso lineare 10 V dc
18	Ingresso lineare 500 V dc
19	Ingresso lineare 20 mA dc
20	Ingresso lineare 100 V ac
21	Ingresso lineare 5 A ac

Tabella 2

0	Punto decimale spento
1	Posizione punto decimale 1999.9
2	Posizione punto decimale 199.99
3	Posizione punto decimale 19.999
4	Posizione punto decimale 1.9999

Tabella 3

0	Arrotondamento a 1
1	Arrotondamento a 2
2	Arrotondamento a 5
3	Arrotondamento a 10

Tabella 4

0	Unità di misura visualizzata: k
1	Unità di misura visualizzata: °C
2	Unità di misura visualizzata: °F
3	Unità di misura visualizzata: V
4	Unità di misura visualizzata: A
5	Unità di misura visualizzata: g
6	Unità di misura spenta

Tabella 5

0	Timeout tastiera 5 secondi
1	Timeout tastiera 10 secondi
2	Timeout tastiera 20 secondi
3	Timeout tastiera 30 secondi

Tabella 6

0	OFF: funzione esclusa
1	On: funzione attiva

Tabella 7

0	OFF (peak disabilitata e attivo Hold)
1	HiInf: picco massimo infinito
2	HitMr: picco massimo temporizzato
3	LoInf picco minimo infinito
4	LotMr picco minimo temporizzato

Tabella 8

Valore	Significato
0	Reinializza filtro: dopo 0,00 secondi fuori dalla finestra
1	Reinializza filtro: dopo 0.25 secondi fuori dalla finestra
2	Reinializza filtro: dopo 0.50 secondi fuori dalla finestra
3	Reinializza filtro: dopo 0.75 secondi fuori dalla finestra
4	Reinializza filtro: dopo 1.00 secondi fuori dalla finestra
5	Reinializza filtro: dopo 1.25 secondi fuori dalla finestra
6	Reinializza filtro: dopo 1.50 secondi fuori dalla finestra
7	Reinializza filtro: dopo 1.75 secondi fuori dalla finestra
8	Reinializza filtro: dopo 2.00 secondi fuori dalla finestra

Tabella 9

0	OFF: protezione parametri esclusa
1	Lo: protezione parametri parziale
2	Hi: protezione parametri totale

Tabella 10

0	InPUt
1	FILtr
2	Set 1
3	Set 2
4	Set 3
5	Set 4

Tabella 11

0	Lo: bassa velocità di risposta, alta risoluzione
1	Hi: alta velocità di risposta, bassa risoluzione

Tabella 12

0	OFF: interfaccia seriale disabilitata
1	ASCII: interfaccia seriale abilitata protocollo Ascii

Tabella 13

0	Velocità trasmissione 300 baud
1	Velocità trasmissione 600 baud
2	Velocità trasmissione 1200 baud
3	Velocità trasmissione 2400 baud
4	Velocità trasmissione 4800 baud
5	Velocità trasmissione 9600 baud

Tabella 14

0	LOCAL: funzionamento locale
1	rEMOt: funzionamento remoto

Le tabelle che seguono, relative all'allarme 1, valgono anche per l'allarme 2 e (se presenti) per gli allarmi 3 e 4

Tabella 15

0	OFF: allarme disabilitato
1	InPUt: ingresso allarme diretto
2	FILtr: ingresso allarme filtrato
3	dSPLy: ingresso allarme visualizzazione

Tabella 16

0	ALrHi: allarme di massima
1	ALrLo: allarme di minima

Tabella 17

0	POS: funzione logica positiva
1	nEg: funzione logica negativa

Tabella 18

0	dir: uscita fisica diretta
1	rEv: uscita fisica invertita

Tabella 19

0	AUtOM: reset automatico
1	MAAnUA: reset manuale

Tabella 20

0	AbSLt: set point assoluto
1	ALr 1: set point relativo allarme 1
2	ALr 2: set point relativo allarme 2
3	ALr 3: set point relativo allarme 3
4	ALr 4: set point relativo allarme 4

Tabella 21

Il dato (ricevuto o trasmesso) va interpretato come insieme di bit:

bit 0	Rele 1: 0 -> Off, 1 -> On
bit 1	Rele 2: 0 -> Off, 1 -> On
bit 2	Rele 3: 0 -> Off, 1 -> On
bit 3	Rele 4: 0 -> Off, 1 -> On
bit 4	Blocco rele 1: 0 = escluso, 1 = attivo
bit 5	Blocco rele 2: 0 = escluso, 1 = attivo
bit 6	Blocco rele 3: 0 = escluso, 1 = attivo
bit 7	Blocco rele 4: 0 = escluso, 1 = attivo

NB: I bit di blocco relé indicano che lo stato del relé non è cambiato a causa o dell'inibizione o a causa dell'eventuale tempo di eccitazione/diseccitazione ancora in corso.

Per sapere se lo stato del relé è quello reale, bisogna confrontare i bit 0, 1, 2, 3 con i corrispondenti bit della locazione RF3 (situazione allarmi).

Esempio: se lo stato del relé è ON, ma il corrispondente allarme è OFF, significa che il relé è in attesa di un reset manuale, o sta attendendo che scada il tempo di diseccitazione.

Tabella 22

Il dato (ricevuto o trasmesso) va interpretato come insieme di bit:

bit 0	Led 'H':	0 -> On	1 -> Off
bit 1	Led 'LK':	0 -> On	1 -> Off
bit 2	Led '2':	0 -> On	1 -> Off
bit 3	Led '1':	0 -> On	1 -> Off
bit 4	Led '3':	0 -> On	1 -> Off
bit 5	Led 'L':	0 -> On	1 -> Off
bit 6	Led 'PK':	0 -> On	1 -> Off
bit 7	Led '4':	0 -> On	1 -> Off

Tabella 23

Valore	Caratteri visualizzati (e visualizzabili)						
0	'0'	14	'E'	28	'i'	42	't'
1	'1'	15	'F'	29	'J'	43	't.'
2	'2'	16	'0.'	30	'J.'	44	'u'
3	'3'	17	'1.'	31	'k'	45	'v'
4	'4'	18	'S.'	32	'k.'	46	'y'
5	'5'	19	'9.'	33	'L'	47	'.'
6	'6'	20	'C.'	34	'L.'	48	'-.'
7	'7'	21	'd.'	35	'M'	49	'-1'
8	'8'	22	'F.'	36	'n'	50	'-1.'
9	'9'	23	'c'	37	'n.'	51	'.'
10	'A'	24	'H'	38	'o'	52	'8.'
11	'b'	25	'h'	39	'P'		
12	'C'	26	'I'	40	'P.'		
13	'd'	27	'l.'	41	'r'		

Tabella 24

Il dato ricevuto va interpretato come insieme di bit:

bit 0	valori calcolati superiori al massimo visualizzabile
bit 1	valori calcolati inferiori al minimo visualizzabile
bit 2	overrange logico
bit 3	underrange logico
bit 4	overrange fisico (valore in ingresso superiore alle capacità di conversione dell'ADC)
bit 5	underrange fisico (valore in ingresso inferiore alle capacità di conversione dell'ADC)
bit 6	misura visualizzata bloccata (Holding)
bit 7	presenza allarmi 3 e 4: 0 -> presenti allarmi 1 e 2 1 -> presenti allarmi 1, 2, 3, 4.

Tabella 25

0	Strumento correttamente funzionante
1	Malfunzionamento ADC
2	Errore lettura 3° filo input Pt100
3	Errore lettura compensazione giunto freddo
4	Malfunzionamento Eeprom interna
5	Errore di taratura
6 ÷ 12	Errore di ricezione via seriale

Tabella 26

Il dato ricevuto va interpretato come insieme di bit:

bit 0	Stato allarme 1:	0 -> a riposo	1 -> attivo
bit 1	Stato allarme 2:	0 -> a riposo	1 -> attivo
bit 2	Stato allarme 3:	0 -> a riposo	1 -> attivo
bit 3	Stato allarme 4:	0 -> a riposo	1 -> attivo
bit 4	Stato inibizione allarme 1:	0 = esclusa	1 = attiva
bit 5	Stato inibizione allarme 2:	0 = esclusa	1 = attiva
bit 6	Stato inibizione allarme 3:	0 = esclusa	1 = attiva
bit 7	Stato inibizione allarme 4:	0 = esclusa	1 = attiva

Vedi NB della tabella 21.

Tabella 27

Il dato ricevuto va interpretato come insieme di bit:

bit 0	Tasto ENTER
bit 1	Tasto INC
bit 2	Tasto DEC/CIFRA

Esempio di programma "C"

```

/*
21 giu 2006, esempio.c - rev. 21 giu 06

Testato il 21-06-06 con Pentium 100 Mhz,
O. S. MS-DOS Ver 6.20

Scritto per BORLAND C++ Version 1.00

Lo strumento deve avere settati i parametri seguenti
con i valori specificati:
'kEyLoK' (gruppo 'diSPL', famiglia 'ConFiG') -> 'OFF'
'Protoc' (gruppo 'rS CoM', famiglia 'ConFiG') -> 'ASCII'
'Addr' (gruppo 'rS CoM', famiglia 'ConFiG') -> '000001'
'bAud' (gruppo 'rS CoM', famiglia 'ConFiG') -> '9600'
'ModE' (gruppo 'rS CoM', famiglia 'ConFiG') -> 'rEMotE'

A titolo di esempio, verrà scritto il parametro 'SEt'
del gruppo 'ALrM 1', famiglia 'ALArM'
con il valore +184542.
Successivamente verrà letto il parametro 'SEt Hi'
del gruppo 'ALrM 1', famiglia 'ALArM'

La porta seriale COM1 verrà settata con:
velocità di comunicazione di 9600 baud,
8 bits per carattere, 1 bit di stop, nessuna parità.
*/

#include <stdio.h>
#include <bios.h>

void main()
{
    int caratteri_rx;
    unsigned char risposta[10];

    bioscom(0x00, 0xE3, 0x00); // inizializza porta seriale

    // trasmissione 1° comando:
    scrive parametro 'SEt' (comando W4B)
    del gruppo 'ALrM 1', famiglia 'ALArM' con il
    valore +184542

    bioscom(1, 2, 0);
    bioscom(1, '0', 0);
    bioscom(1, '1', 0);
    bioscom(1, 'W', 0);
    bioscom(1, '4', 0);
    bioscom(1, 'B', 0);
    bioscom(1, '=', 0);
    bioscom(1, '+', 0);
    bioscom(1, '1', 0);
    bioscom(1, '8', 0);
    bioscom(1, '4', 0);
    bioscom(1, '5', 0);
    bioscom(1, '4', 0);
    bioscom(1, '2', 0);
    bioscom(1, 3, 0);
    bioscom(1, 2 ^ '0' ^ '1' ^ 'W' ^ '4' ^ 'B' ^ '=' ^ '+' ^ '1' ^ '8'
        ^ '4' ^ '5' ^ '4' ^ '2' ^ 3, 0);

    for (caratteri_rx = 0; caratteri_rx < 7; caratteri_rx++)
        risposta[caratteri_rx] = bioscom(2, 0, 0);

    // stampa l'esito dell'operazione
    printf("Esito del comando W4B=+184542: ");
    if (caratteri_rx == 7) {
        for (caratteri_rx=1; caratteri_rx <=4; caratteri_rx++)
            printf("%c", risposta[caratteri_rx]);
    } else
        printf("Errore nella ricezione seriale");
    printf("\n");

    delay(1000); // ritardo 1 s

    // pulisce il buffer di ricezione
    bioscom(2, 0, 0);

    // trasmissione 2° comando: legge parametro 'SEt Hi'
    (comando R4E)
    del gruppo 'ALrM 1', famiglia 'ALArM'

    bioscom(1, 2, 0);
    bioscom(1, '0', 0);
    bioscom(1, '1', 0);
    bioscom(1, 'R', 0);
    bioscom(1, '4', 0);
    bioscom(1, 'E', 0);
    bioscom(1, 3, 0);
    bioscom(1, 2 ^ '0' ^ '1' ^ 'R' ^ '4' ^ 'E' ^ 3, 0);

    for (caratteri_rx = 0; caratteri_rx < 10; caratteri_rx++)
        risposta[caratteri_rx] = bioscom(2, 0, 0);

    printf("\nValore letto: %c%c%c%c%c%c%c\n",
        risposta[1],
        risposta[2],
        risposta[3],
        risposta[4],
        risposta[5],
        risposta[6],
        risposta[7] );
}

```

Esempio di programma "QBASIC"

```

REM 21 giu 2006, comunica.bas - rev. 21 giu 06

REM Scritto per MSDOS QBASIC Version 1.1 di MSDOS.
REM Lo strumento deve avere settati i parametri seguenti
    con i valori specificati:
REM 'kEyLoK' (gruppo 'diSPL', famiglia 'ConFiG') -> 'OFF'
REM 'Protoc' (gruppo 'rS CoM', famiglia 'ConFiG') -> 'ASCII'
REM 'Addr' (gruppo 'rS CoM', famiglia 'ConFiG') -> '000001'
REM 'bAud' (gruppo 'rS CoM', famiglia 'ConFiG') -> '9600'
REM 'ModE' (gruppo 'rS CoM', famiglia 'ConFiG') -> 'rEMotE'

REM A titolo di esempio, verrà scritto il parametro 'SEt'
REM del gruppo 'ALrM 1', famiglia 'ALArM'
REM con il valore +184542.
REM Successivamente verrà letto il parametro 'SEt Hi'
REM del gruppo 'ALrM 1', famiglia 'ALArM'

REM La porta seriale COM1 verrà settata con una velocità di
    comunicazione di 9600 baud, 8 bits per carattere,
    1 bit di stop, nessuna parità.

10 DIM risposta(10)
20 CLS

100 OPEN "com1: 9600,n,8,1,cd,cs,ds,rs"
    FOR RANDOM AS #1

REM trasmissione 1ø comando:
    scrive parametro 'SEt' (comando W4B)
    del gruppo 'ALrM 1', famiglia 'ALArM'
    con il valore +184542

200 PRINT #1, CHR$(2);
205 PRINT #1, "0";
210 PRINT #1, "1";
215 PRINT #1, "W";
220 PRINT #1, "4";
225 PRINT #1, "B";
230 PRINT #1, "=";
235 PRINT #1, "+";
240 PRINT #1, "1";
245 PRINT #1, "8";
250 PRINT #1, "4";
255 PRINT #1, "5";
260 PRINT #1, "4";
262 PRINT #1, "2";
265 PRINT #1, CHR$(3);
270 PRINT #1, CHR$(2 XOR ASC("0") XOR ASC("1") XOR
    ASC("W") XOR ASC("4") XOR ASC("B") XOR ASC("=")
    XOR ASC("+") XOR ASC("1") XOR ASC("8") XOR
    ASC("4") XOR ASC("5") XOR ASC("4") XOR ASC("2")
    XOR 3);

300 FOR caratteri = 1 TO 7
310 risposta(caratteri) = ASC(INPUT$(1, 1))
320 NEXT caratteri

350 REM Stampa l'esito
355 PRINT "Esito comando W4B=+184542: ";
360 FOR caratteri = 2 TO 5
365 PRINT CHR$(risposta(caratteri));
370 NEXT caratteri

400 REM Pausa di 1 secondo
410 SLEEP (1)

420 PRINT
430 PRINT

REM trasmissione 2ø comando:
    legge parametro 'SEt Hi' (comando R4E)
    del gruppo 'ALrM 1', famiglia 'ALArM'

500 PRINT #1, CHR$(2);
510 PRINT #1, "0";
520 PRINT #1, "1";
530 PRINT #1, "R";
540 PRINT #1, "4";
550 PRINT #1, "E";
560 PRINT #1, CHR$(3);
570 PRINT #1, CHR$(2 XOR ASC("0") XOR
    ASC("1") XOR ASC("R") XOR ASC("4")
    XOR ASC("E") XOR 3);

600 FOR caratteri = 1 TO 10
610 risposta(caratteri) = ASC(INPUT$(1, 1))
620 NEXT caratteri

630 REM Stampa l'esito
635 PRINT "Esito comando R4E: ";
640 FOR caratteri = 2 TO 8
650 PRINT CHR$(risposta(caratteri));
660 NEXT caratteri

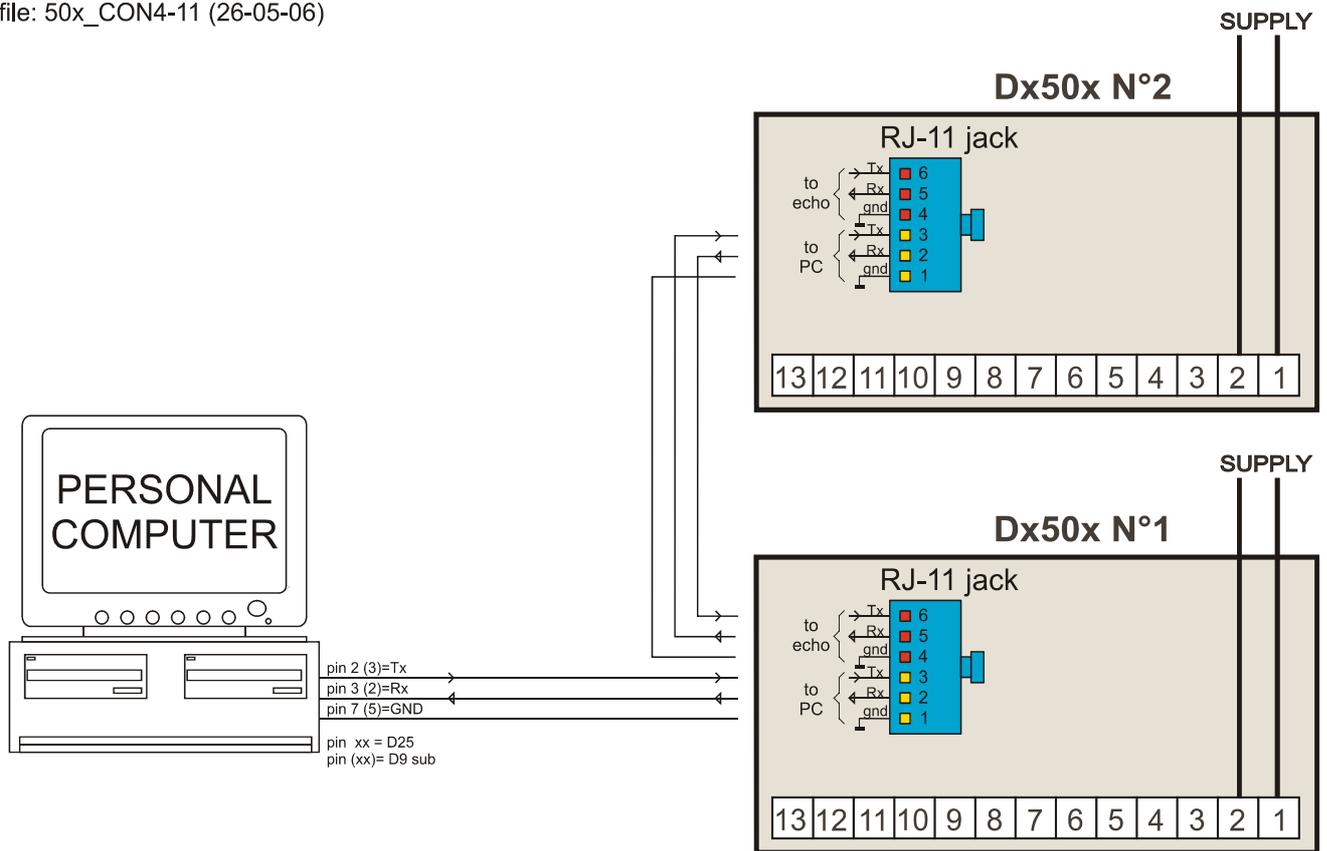
700 CLOSE (1)

800 END

```

RS 232 INTERFACE CONNECTION and MULTI DROP EXAMPLE

file: 50x_CON4-11 (26-05-06)



RS 485 INTERFACE CONNECTION and MULTI DROP EXAMPLE

