

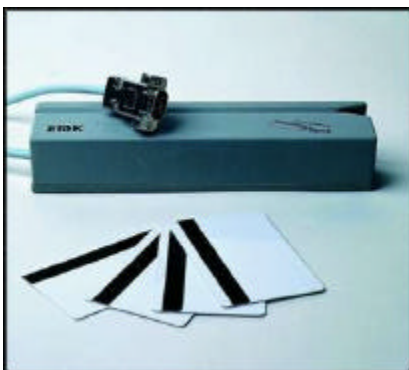
# MANUALE DI PRODOTTO



## PTR-RS3

---

Lettoce manuale a tre piste con interfaccia RS-232



Versione 1.2



## MANUALE di PRODOTTO

# PTR-RS3

**Versione 1.2**

## Tabella Revisioni

Rev.	Data	Fogli modificati
AA	16.04.97	Tutti
AB	09.07.97	1, 2, 4, 26 ... 42
AC	24.09.97	1, 2, 36
B	11.02.04	Tutti RESTYLING MANUALE

Le informazioni contenute in questo documento sono di esclusiva proprietà della società **Emk S.r.l.** eccetto dove diversamente specificato; tali informazioni non possono essere riprodotte né per intero né in parte senza l'esplicita autorizzazione scritta della stessa società.

Il documento può essere soggetto a variazioni senza che questo comporti alcun obbligo di preavviso da parte della **Emk**



# Sommario

MANUALE DI PRODOTTO .....	1
Tabella Revisioni .....	2
CAPITOLO 1 .....	5
Generalità.....	5
Introduzione .....	5
Caratteristiche di Prodotto .....	5
Documenti di riferimento.....	6
CAPITOLO 2 .....	7
Architettura .....	7
Introduzione .....	7
Configurazione hardware .....	8
CAPITOLO 3 .....	9
Funzionalità.....	9
Introduzione .....	9
Configurazioni all'accensione .....	9
Formati Tessera .....	10
<i>Dimensioni delle Tessere</i> .....	10
Funzioni di controllo.....	11
<i>Controlli sulle tessere standard</i> .....	11
<i>Controllo presenza della tessera</i> .....	12
Caratteristiche diagnostiche .....	12
<i>Autodiagnostica all'accensione</i> .....	12
CAPITOLO 4 .....	13
Visibilità Software .....	13
Introduzione .....	13
Protocollo di Comunicazione .....	13
Struttura dei messaggi .....	15
Formato dei caratteri.....	15
Formato dei comandi .....	16
Formato messaggi PTR-RS3 .....	17



Livello “Applicazione” .....	19
Programma di Test in ambiente Ms-Dos .....	19
Ambiente Windows (32 bit).....	26
Livello DDE .....	26
Livello DLL.....	29
Comandi per la Libreria d’Interfaccia PTR-RS.....	29
Status Codes .....	29
PTR_OpenChannel.....	31
PTR_CloseChannel.....	31
PTR_OpenSession .....	32
PTR_CloseSession.....	32
PTR_GetData .....	33
PTR_SendCommand.....	34
Wake_up .....	34
Programma Dimostrativo .....	35
Contenuto del dischetto a corredo .....	36
<b>CAPITOLO 5</b> .....	<b>37</b>
Connessioni .....	37
Introduzione .....	37
Cavo .....	37
Interfaccia Fisica.....	37
Connettori RS-232.....	38
Connettore interno .....	38
<b>CAPITOLO 6</b> .....	<b>39</b>
Caratteristiche fisiche.....	39
Dimensioni e pesi.....	39
Condizioni ambientali .....	39
Prestazioni .....	40
Affidabilità.....	41



## CAPITOLO 1

# Generalità

### Introduzione

Questo manuale contiene le informazioni tecniche necessarie all'uso del lettore PTR-RS3 prodotto dalla società Emk

### Caratteristiche di Prodotto

Il prodotto, denominato **PTR-RS3**, è un lettore di tessere magnetiche utilizzabili come chiavi di accesso per sistemi di sicurezza, per il riconoscimento utente, per la distribuzione automatica, per il controllo di produzione ecc.

Il **PTR-RS3** è collegabile ad un Personal Computer tramite linea seriale RS-232 dalla quale può essere direttamente alimentato.

La lettura della tessera è realizzata mediante scorrimento manuale monodirezionale; un selettore automatico interno permette la commutazione in lettura su una delle tre piste standard ISO.

Un apposito switch permette la rilevazione della presenza della tessera; un led e un buzzer (opzionale) permettono segnalazioni visive ed acustiche verso l'utente.

Il **PTR-RS3** è realizzato in versione da tavolo con gommini anti-scivolo (opzionalmente in versione da parete).



## **Documenti di riferimento**

- Standard Internazionali: ISO 7810, 7811, 7812, 7813
- Standard IBM 3604

## CAPITOLO 2

# Architettura

## Introduzione

Il dispositivo PTR-RS3 è un'unità da tavolo, tipicamente utilizzata come chiave di accesso ad un sistema informativo.

Abilita un singolo punto di passaggio a cui vengono associate le seguenti funzioni:

- Lettore magnetico di tessere capace di leggere monodirezionalmente una tessera preregistrata in traccia 1, 2 o 3 con caratteristiche fisiche rispondenti al documento ISO 7810 (standard internazionale). Poiché il **PTR-RS3** è collegato via linea RS-232 ad un controllore esterno (PC o altro), l'alimentazione può essere direttamente ricavata dai segnali di interfaccia
- Led di segnalazione di 'lettore pronto'; a lettura effettuata viene temporaneamente spento
- Buzzer per eventuali segnalazioni acustiche
- Switch di rilevamento della condizione 'badge presente'
- Interfaccia RS-232
- Il circuito di alimentazione utilizza i segnali RTS e DTR

## Configurazione hardware

La seguente figura descrive a blocchi le parti che compongono il dispositivo **PTR-RS3**:

- Box metallico verniciato oppure in alluminio anodizzato nero o grigio
- Piastra elettronica di base (governo) contenente le seguenti sezioni:
  - Unità Centrale (microprocessore) con relativa memoria di programma (FIRMWARE)
  - Interfaccia seriale RS-232
  - Circuiti per la decodifica magnetica
  - Circuiti di alimentazione locale
  - Ponticelli di configurazione
  - Led e Buzzer
- Testina di lettura dei dati su pista magnetica (traccia standard ISO 1, 2 o 3).
- Switch di segnalazione della presenza della tessera magnetica.
- Cavo con terminazione standard RS-232 per PC (9/25 vie).

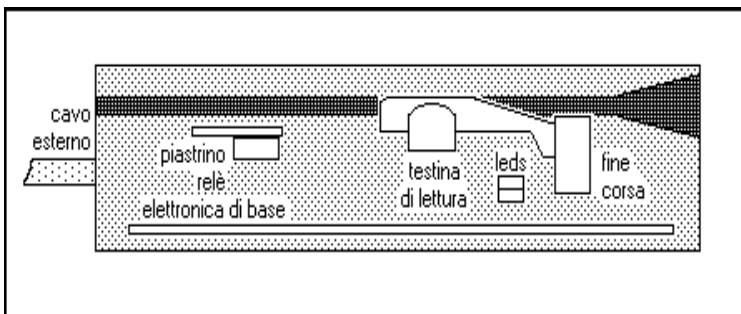


Fig. 2.1 - CONFIGURAZIONE HARDWARE

## CAPITOLO 3

# Funzionalità

## Introduzione

All'accensione del PTR-RS3 il led viene immediatamente spento (fase di inizializzazione) per evitare anche il minimo consumo non necessario; in questa fase il dispositivo rileva la posizione dei ponticelli e si configura alla velocità selezionata.

Al termine di questa brevissima fase il lettore entra nello stato 'idle' per permettere la carica dei condensatori di alimentazione.

Dopo qualche secondo il dispositivo esegue la fase di autodiagnosi ultimata la quale entra nello stato Stand-By (IDLE) in attesa di comandi da linea RS-232.

Alla ricezione di uno dei tre possibili comandi di lettura, il lettore accende il Led segnalando il suo stato di 'pronto a leggere' le tessere magnetiche.

In questa condizione, quando l'utente striscia la tessera, il PTR-RS3 esegue anzitutto i controlli sul formato fisico e quindi si comporta come descritto nel paragrafo "Funzioni di Controllo".

## Configurazioni all'accensione

Come già accennato in precedenza, il PTR-RS3 può essere configurato mediante il posizionamento di un apposito ponticello interno per operare alla velocità di linea di 1.200 oppure 4.800 baud.

La condizione di default per ciò che riguarda lo standard di lettura applicato è ISO-3554.

## Formati Tessera

Le tessere magnetiche che il dispositivo **PTR-RS3** è in grado di leggere sono codificate secondo gli Standard Internazionali ISO 7810, 7811, 7812, 7813 di cui sono riportati i dati salienti:

- Traccia 1: ISO 1 IATA
  - Densità: 210 bpi
  - Codice alfanumerico (6 bits + parità)
  - Totale caratteri: 79 (76 dati + 3 caratteri di controllo)
- Traccia 2: ISO 2 ABA
  - Densità: 75 bpi
  - Codice numerico (4 bits + parità)
  - Totale caratteri: 40 (37 dati + 3 caratteri di controllo)
- Traccia 3: ISO 3 THRIFT
  - Densità: 210 bpi
  - Codice numerico (4 bits + parità)
  - Totale caratteri: 107 (104 dati + 3 caratteri di controllo)

In alternativa allo standard ISO può essere applicato, mediante ricezione di un apposito comando in linea, lo standard di lettura IBM 3604.

## Dimensioni delle Tessere

Il formato fisico di queste tessere è quello definito dallo Standard Internazionale ISO 7810.



## Funzioni di controllo

Il **PTR-RS3** esegue dei controlli sul formato fisico dei dati letti da tessera magnetica in accordo con lo standard internazionale. Tali controlli riguardano:

- la rilevazione del carattere **START SENTINEL**
- il controllo della parità di ogni singolo carattere
- la rilevazione del carattere **STOP SENTINEL**
- il calcolo per il controllo del carattere **LRC finale**

In configurazioni speciali il **PTR-RS3** può eseguire dei controlli aggiuntivi sul formato della tessera; i prossimi paragrafi ne danno una spiegazione più dettagliata. Un altro controllo è effettuato sullo switch di "fine corsa" allo scopo di rilevare il momento dell'inserzione e dell'estrazione della tessera.

## Controlli sulle tessere standard

Le tessere standard non necessitano di ulteriori controlli oltre a quelli già descritti in precedenza.

A lettura avvenuta (corretta od errata), il led rosso viene spento ed il messaggio (dati tessera oppure errore) spedito in linea; la ricezione del carattere **ACK** oppure di un altro comando (**ACK** implicito) riaccende il led. Se il controllore esterno invia il carattere **NACK**, il **PTR-RS3** ripete la trasmissione dei dati dopo circa 0,5 sec. tenendo il led spento (il time-out da 0,5 sec. serve a salvare la carica dei condensatori di alimentazione).

### *Controllo presenza della tessera*

Il controllo sullo switch di fine-corsa viene effettuato contemporaneamente alla lettura dei dati magnetici: se lo switch di fine corsa non viene azionato, il PTR-RS3 non legge i dati tessera.

Questo evita di avere letture errate in seguito ad una non corretta inserzione del badge (per esempio facendo scorrere il badge non posizionato sul fondo del lettore).

Il rilevamento viene effettuato dallo switch posizionato in asse alla testina di lettura.

Al termine della lettura viene continuamente controllato il momento dell'estrazione tessera; in tal caso il PTR-RS3 aggiorna il relativo bit nel byte 'status'.

## **Caratteristiche diagnostiche**

Il lettore di tessere **PTR-RS3** ha un'architettura sufficientemente semplice da non richiedere una diagnostica complessa.

## **Autodiagnostica all'accensione**

Tutte le volte riceve l'alimentazione (fase di reset H/W), dopo un tempo di qualche secondo necessario alla carica dei condensatori di alimentazione, il **PTR-RS3** esegue la routine di autodiagnosi ultimata la quale entra nello stato stand-by (IDLE = basso consumo).

L'autodiagnostica all'accensione controlla il corretto funzionamento della Ram interna e della Eprom esterna; il risultato non corretto viene eventualmente segnalato sul led (blink del codice di errore).

## CAPITOLO 4

# Visibilità Software

## Introduzione

Questa sezione contiene la descrizione dettagliata delle modalità di comunicazione tra il lettore PTR-RS3 ed il suo controllore esterno; oltre al protocollo ed all'interfaccia comandi, viene fornita la descrizione dell'interfaccia software di più alto livello per ambienti Dos e Windows

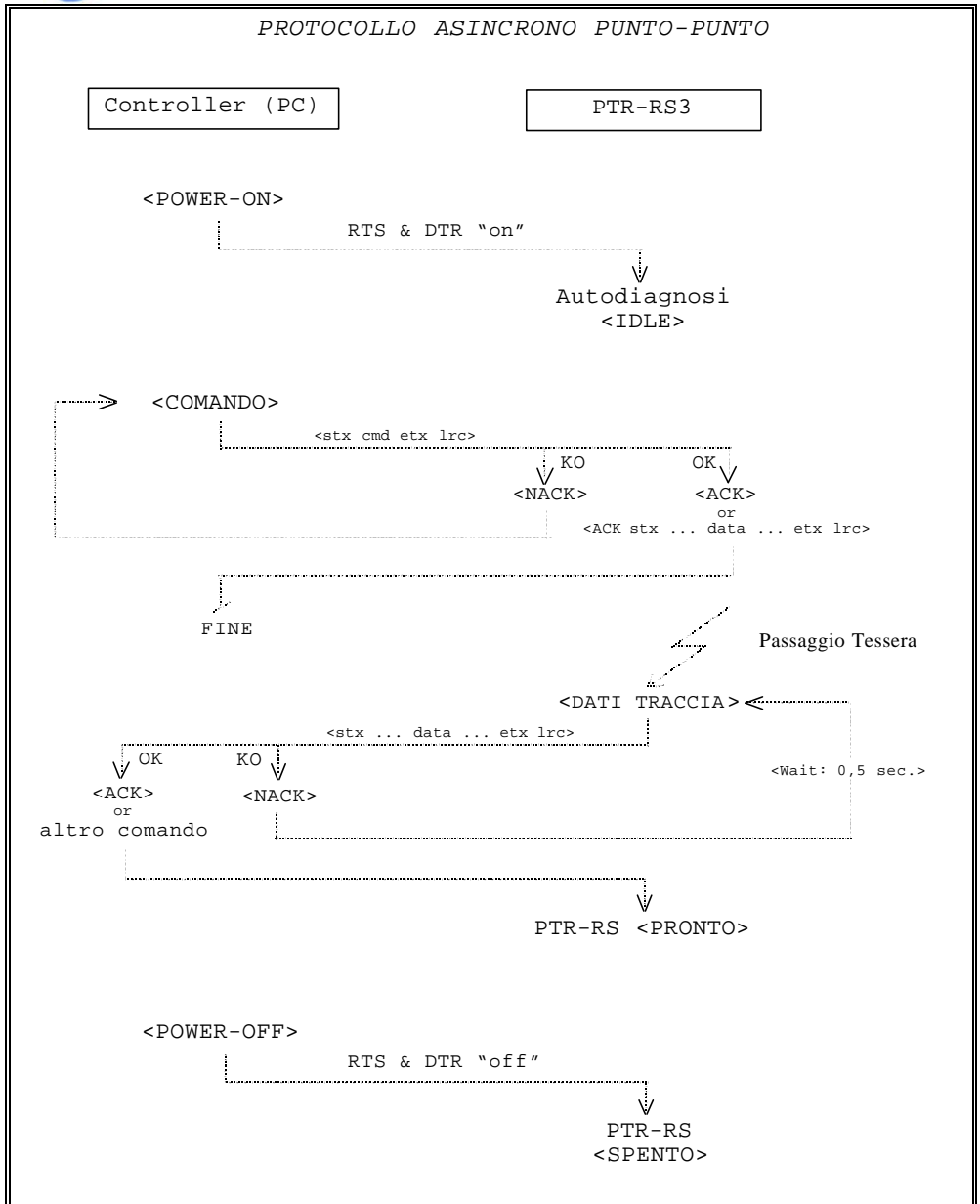
## Protocollo di Comunicazione

Per comunicare i dati al controllore esterno il PTR-RS3 utilizza la linea seriale di tipo RS-232.

Il protocollo di comunicazione ha una struttura molto semplice poiché la connessione, di tipo locale, è da considerarsi a basso tasso d'errore.

La figura seguente mostra le caratteristiche salienti di tale protocollo.

PROTOCOLLO ASINCRONO PUNTO-PUNTO





## Struttura dei messaggi

La struttura dei messaggi in seguito descritta è riferita ai frame scambiati tra PTR-RS3 ed il suo controllore esterno (PC o altro).



dove:

<STX> = 0x82

<ETX> = 0x03

<LRC> = parità longitudinale (STX ed ETX compresi) su 7 bit

### *Formato dei caratteri*

Il formato dei caratteri inviati in linea è il seguente:

- velocità 1.200 oppure 4.800 bps
- formato 7 bit
- parità pari
- stop bit 1



### Formato dei comandi

Di seguito viene dato l'elenco dei comandi ammessi per il lettore PTR-RS3 (livello "trasporto"):

Comandi	Hex	Descrizione
A	0x41	Set della modalità di lettura: <i>ISO 3554</i> (default)
B0	0x42 0x30	Set modalità '0' per il buzzer: <i>lett.OK= no beep; lett. KO=1beep @ 2Khz x 0,5 sec.</i>
B1	0x42 0x31	Set modalità '1' per il buzzer: <i>lett.OK=1beep @ 2Khz x 0,5 sec.; lett. KO=3 beep</i>
B2	0x42 0x32	Set modalità '2' per il buzzer: <i>lett.OK= no beep; lett. KO= no beep.</i>
D	0x44	Disabilita la lettura (reset bit 3 e bit 6 del byte status)
I	0x49	Set della modalità di lettura: <i>IBM 3604</i>
R	0x52	Richiesta della numero di Release e Data del Firmware
S	0x53	Richiesta di Stato
T1	0x54 0x31	Seleziona Traccia #1 (attivo per un solo badge)
T2	0x54 0x32	Seleziona Traccia #2 (attivo per un solo badge)
T3	0x54 0x33	Seleziona Traccia #3 (attivo per un solo badge)

Le risposte ai comandi elencati possono essere semplici (ACK, NACK) oppure a lunghezza variabile (per gli ultimi cinque comandi in tabella).

- **<ACK>** (0x06) risposta di accettazione
- **<NACK>** (0x95) richiesta di ripetizione

Nel caso Host trasmetta il carattere **<NACK>**, il **PTR-RS3** esegue la ripetizione del messaggio dopo un tempo di circa 0,5 sec.

### Formato messaggi PTR-RS3

Il formato dei messaggi inviati dal PTR-RS3 al suo controllore esterno è uno dei seguenti:

#### 1. Risposta al comando di *Richiesta del Numero di Release e Data*

Ack Stx R x.y gg / mm / aa [3] Etx lrc
--

dove:

x.y	numero di release (ad esempio: 1.0)
gg/mm/aa	data di emissione (ad esempio: 12/10/96)
[ 3 ]	campo opzionale presente solo per lettore a 3 piste

#### 2. Risposta al comando di *Richiesta di Stato*:

Ack Stx <sts> Etx lrc
-----------------------

dove:

<sts>

b6 = 0 = ultima lettura OK	b6 = 1 = ultima lettura KO
b5 = 0 = Ram test: KO	b5 = 1 = Ram test :OK
b4 = 0 = Rom test: KO	b4 = 1 = Rom test: OK
b3 = 0 = tessera non presente	b3 = 1 = tessera presente
b2, b1 = 00 modalità buzzer = 2	
b2, b1 = 01 modalità buzzer = 0 (default)	
b2, b1 = 10 modalità buzzer = 1	
b2, b1 = 11 non utilizzabile	
b0 = 0 = sel. piste #1 e #2	b0 = 1 = sel. pista #3



3. Risposta ai comandi di *Selezione alla lettura di una delle tre piste standard* (messaggio inviato a lettura avvenuta):

```
Stx <trk> <std> < ..... dati badge ..... > Etx lrc
```

dove:

<trk> n. di traccia selezionata (1, 2 oppure 3)  
<std> tipo di standard selezionato (A=ISO3554, I=IBM 3604)

La lunghezza della stringa <dati> è variabile a seconda della pista selezionata come segue:

- pista 1: dati alfanumerici (lunghezza massima 76)
- pista 2: dati numerici (lunghezza massima 37)
- pista 3: dati numerici (lunghezza massima 104)

Nel caso la lettura dia errore sul formato fisico, il messaggio inviato ad host è il seguente:

```
Stx ! Etx lrc
```

## Livello “Applicazione”

Il seguente paragrafo descrive le utilità disponibili e i programmi dimostrativi utili per sviluppare applicazioni in grado di gestire le funzionalità del lettore PTR-RS3.

### *Programma di Test in ambiente Ms-Dos*

E' un programma dimostrativo utilizzabile per verificare le funzionalità del lettore sotto il sistema operativo MS-DOS. Permette la selezione di ciascuna delle tre piste disponibili e dopo la lettura, visualizza i dati della pista desiderata.

```
// DEMO in linguaggio C++ (Borland) per il controllo del  
LETTORE-MAGNETICO a 3-PISTE Emk
```

```
#include <dos.h>  
#include <string.h>  
#include <stdlib.h>  
#include <conio.h>  
#include <alloc.h>  
#include <math.h>  
#include <stdio.h>  
#include <fcntl.h>  
#include <io.h>  
#include <bios.h>  
  
//***** da BIOSCOM  
#define COM1          0  
#define COM2          1  
#define COM3          2  
#define COM4          3  
#define DATA_READY  0x100  
#define OVER_RUN      0x200  
#define STX           2  
#define ETX           3  
#define ACK           6  
#define NACK          21
```



```
//***** Funzioni a basso livello
void get_message (void);
void send_message (char *str);

int status, pk;
char rx_msg[256];
int i, com_richiesta=COM1, portaDTR=0x3FC;
int baud_rate=0x80|0x2|0x18; // 1200,E,7,1
char *e="T1";

int xbioscom(int cmd, char abyte, int port)
{
int s;
switch (cmd)
{
case 1:
while ((pk=peek(0, 0x0417)&0x1)==0)
{
s=bioscom(3,0,com_richiesta);
if ((s&0x4000)==0x4000)
break;
}
s=inport((portaDTR&0xfff0)|0x8);
// svuota echo precedente se c'e`
outportb((portaDTR&0xfff0)|0x8,abyte);
break;

case 2:
while ((pk=peek(0, 0x0417)&0x1)==0)
{
s=bioscom(3,0,com_richiesta);
if ((s&0x100)==0x100)
break;
}
return (inport((portaDTR&0xfff0)|0x8));

case 3:
return (inport((portaDTR&0xfff0)|0xD));
}
}
```



```
void main(int argc, char *argv[])
{
    char argo[3];
    char buftx[10];
    char *baud="1200";

    for (i=1; i<argc; i++)
    {
        strcpy (argo, argv[i]);
       strupr(argo);
        switch (argo[0])
        { case '/':
            switch (argo[1])
            { case '1':
                com_richiesta = COM1;
                portaDTR=0x3fC;
                break;
            case '2':
                com_richiesta = COM2;
                portaDTR=0x2fC;
                break;
            case '3':
                com_richiesta = COM3;
                portaDTR=0x3EC;
                break;
            case '4':
                com_richiesta = COM4;
                portaDTR=0x2EC;
                break;
            case 'B':
                baud_rate=0xC0|0x2|0x18;
                // 4800,E,7,1 baud="4800";
                break;
            case '?':
            case 'H':
                printf (" opzioni:\n");
                printf ("          /n : n=porta di comunicazione
(default COM1)\n");
                printf ("          /B : SET BAUD-RATE
4800 (default 1200)\n");
```



```
        printf ("          t : pista di lettura(default
1)\n");
        exit(0);
        break;
        default:
        exit(1);          }

        break;
        case '1':
        e="T1";
        break;
        case '2':
        e="T2";
        break;
        case '3':
        e="T3";
        break;
    }
}
```

```
clrscr();
strcpy(rx_msg,"E M K: prova PTR seriale in traccia");
strcat(rx_msg,++e); e--;
strcat(rx_msg," baud-rate ");
strcat(rx_msg,baud);
printf("%s\n",rx_msg);

status = bioscom(0, baud_rate, com_richiesta);
outportb (portaDTR, 0xB); // DTR RTS on !
delay(3000);

printf("(SHIFT DI DESTRA per uscire)\n");
strcpy(bufx, "S");
send_message(bufx);
get_message();
i=rx_msg[0]&&8;
strcpy(rx_msg,"Tessera");
```



```
if (i)
    strcat(rx_msg,"presente");
else
    strcat(rx_msg,"assente");

while ((pk=peek(0, 0x0417)&0x1)==0)
{
    printf("%s\n",rx_msg);
    strcpy(bufTx, e);
    send_message(bufTx);
    get_message();
}
}
```

**//\*\*\*\*\* Corpo delle funzioni**

**/\* Get\_message** : attende un messaggio da linea RS-232 e lo copia nel buffer puntato da rx\_msg.

- Parametri facoltativi:
- nessuno \*/

```
void get_message (void)
{
    unsigned char datol, lrc = 0x00, i = 0x00, rxlrc=0;
    while ((pk=peek(0, 0x0417)&0x1)==0)
    {
        status = xbioscom(3, 0, com_richiesta);
        if (status & 1)
        {
            datol = xbioscom(2, 0, com_richiesta) & 0x7F;
            if (rxlrc)
            {
                rxlrc=0;
                if (lrc==datol)
                {
                    xbioscom(1, ACK, com_richiesta);
                    rx_msg[--i]=0;
                    break;
                }
            }
        }
    }
}
```



```
//          else
//          bioscom(1, NACK, com_richiesta);
    }
    rx_msg [i++] = datol;
    lrc ^= datol;
    if (datol == STX)
    {
        lrc = datol;
        i = 0;
    }
    if (datol == ETX) rxlrc=1;
}
}
```

/\* **Send\_message** : invia il comando in linea

- Parametri necessari:
- la stringa contenente il comando da scrivere nel buffer

(\*str)

- Parametri facoltativi:
- nessuno \*/

```
void send_message (char *str)
{
int status, datol, lrc;

    if (strlen (str) > 254)
        str [255] = 0x00;

    while ((pk=peek(0, 0x0417)&0x1)==0)
    {
        lrc = 0x01;
        xbioscom(1, STX, com_richiesta);
```



```
for (int i = 0; i < strlen (str); i++)
    {
        xbioscom(1, str[i] & 0x7f, com_richiesta); // Send String
        lrc ^= str[i] & 0x7f;
    }
xbioscom(1, ETX, com_richiesta);
xbioscom(1, lrc, com_richiesta);
break;
}
while ((pk=peek(0, 0x0417)&0x1)==0)
{
    lrc=inportb((portaDTR&0xfff0)|0xD);
    if ( (lrc&1)==1)
    {
        datol = inportb((portaDTR&0xfff0)|0x8);
        if (datol==ACK)
            break;
    }
}
}
```



### **Ambiente Windows (32 bit)**

In questo ambiente sono disponibili due diversi livelli di interfaccia descritti in seguito:

- DDE (un server DDE è disponibile per interfacciare un client DDE)
- DLL (una libreria di funzioni a più alto livello consente un rapido sviluppo di applicazioni in Visual Basic o simili)

Utilizzando una delle suddette interfacce è possibile controllare anche la versione hardware PTR-RS (lettore a singola testina con velocità d'interfaccia automaticamente selezionabile fino a 9.600 bps); l'interfaccia comandi di tale lettore viene filtrata dal programma PTRserver.

#### Livello DDE

Questo è il livello utilizzato dal programma PTRserver (sviluppato a 32-bit sotto Win-95 compatibile Win-NT) per comunicare con i programmi applicativi esterni; i **riferimenti DDE** sono i seguenti:

- Applicazione (SERVICE): **PTRSERVER**
- TOPIC: **PTR**
- ITEM: **MSG**

### **INTERFACCIA COMANDI**

L'interfaccia comandi tra il PTRserver ed i relativi Client (DLL incluso) è basata sulle seguenti regole:

- Formato del Messaggio:

Header (2)	Lunghezza (2)	[ dati ]
------------	---------------	----------

Ogni messaggio deve essere inviato al DDE-interface-layer dopo la conversione di ogni byte in due caratteri ASCII (per esempio, per inviare il byte 0x6D, si deve trasmettere 0x36 seguito da 0x44).

<Header> contiene 0x6D (“m”) ad evidenziare che i messaggi provengono da un lettore magnetico



- <Lunghezza> contiene la lunghezza del campo <dati> (compresa tra 0x00 ÷ 0xFF)
- <dati> contiene il messaggio opzionale inviato o ricevuto dal PTRserver

Il formato di Risposta dei messaggi del tipo 6D (badge magnetici) è il seguente:

Header (2)	Lungh. (2)	Status (2)	[dati]	eco-comando
------------	------------	------------	--------	-------------

- <Status> può contenere 0x00 (OK) o un valore diverso (specifico per ogni comando)
- <eco-comando> contiene l'eco del primo byte del campo <dati> ricevuto in precedenza

La seguente tabella contiene tutti i comandi disponibili.

COMANDO	FORMATO	Formato-Risposta RS3	Formato-Risposta RS
Selez. ISO-3554	6D 01 41	6D 02 00 41	6D 02 00 41
Selez. IBM-3604	6D 01 49	6D 02 00 49	6D 02 A1 49 (error)
DISABILITA il lettore	6D 01 44	6D 02 00 44	6D 02 00 44 (power-off)
LEGGI STATUS	6D 01 53	6D 03 00 xx 53 xx = status byte	6D 03 00 xx 53 xx = 00, 01 or 02
SELEZIONA il Canale di Lettura	6D 02 54 xx xx = n. pista (01÷03)	6D 02 00 54	6D 02 00 54 (power-on)
INVIA REL. F/W	6D 01 52	6D ll 00 xx ... xx [yy] 52 ll = lungh. (0D or 0E) xx = 11 car. Messaggio Release (ASCII) [yy] = opz. (33 = RS3)	6D 05 00 xx.yy 52 xx.yy = Rel. (ASCII)
Modalità BUZZER	6D 02 42 xx xx = 00, 01 or 02	6D 02 00 42	6D 02 A1 42 (error)



Quando un messaggio non può essere gestito in modo appropriato dal PTRserver, la risposta diventa:

- 6D 02 A1 <eco-cmd> comando o parametri non validi  
6D 02 A2 <eco-cmd> è accaduto un errore di Time-Out durante la trasmissione del messaggio al PTR-RS3
- 6D 02 A3 <eco-cmd> è accaduto un errore di Retry durante la trasmissione del messaggio al PTR-RS3

I seguenti **messaggi non sollecitati** possono essere inviati dal dispositivo PTR (RS3 o RS):

- ad ogni lettura correttamente effettuata:

6D xx 00 0y 4z nn ..... nn 00
xx = lunghezza y = numero pista (1 or 2 or 3) z = standard di lettura (1 o 9: il PTR-RS può solo inviare il valore 1) nn = dati pista (ASCII: 20 ÷ 5F)

- ad ogni errore di lettura (valido solamente per PTR-RS3):

6D 02 B1 00
tale messaggio non può essere generato dal dispositivo PTR-RS

- ogni volta viene rilevata la condizione “badge-rimosso”(valida solamente per PTR-RS):

6D 03 00 1B 00
tale messaggio non può essere generato dal dispositivo PTR-RS3



### Livello DLL

La libreria PTR.dll è un insieme di comandi in grado di gestire le funzionalità del PTR-RS3 (o PTR-RS). Utilizzando questo interfaccia ad alto livello, uno sviluppatore può chiamare i comandi dall'applicazione scritta sia in linguaggio C che in Visual Basic o simili sotto Windows-95 o Windows-NT.

Le *funzioni DLL* contenute nella libreria in oggetto, utilizzano l'interfaccia DDE menzionata precedentemente.

### *Comandi per la Libreria d'Interfaccia PTR-RS*

- **PTR\_OpenChannel**                    Apre un canale verso il server DDE
- **PTR\_CloseChannel**                Chiude il canale aperto
- **PTR\_OpenSession**                Apre una sessione verso una traccia
- **PTR\_CloseSession**               Chiude una sessione
- **PTR\_GetData**                      Ritorna i dati disponibili
- **PTR\_SendCommand**                Invia un comando al PTR-RS3 (RS)

### *Status Codes*

I seguenti codici di stato vengono ritornati quando si riscontra un qualsiasi problema (altrimenti il codice di ritorno vale 0x00). Ad esempio quando un canale DDE non è ancora aperto oppure quando avviene un errore di sistema (Windows).

<b>Code</b>	<b>Significato</b>	<b>rif. DDE</b>
- 501	DDE error (manca la connessione al DDE)	-
- 503	Manca Open session (prima del comando in corso)	-
- 1010	Il canale DDE è occupato (in attesa di risposte)	-
- 311	Comando o parametro non valido	A1
- 312	Errore di Time-out in trasmissione	A2
- 313	Errore di Retry	A3
- 2	Non ci sono dati disponibili	-



I seguenti paragrafi usano le seguenti convenzioni tipografiche:

Esempio di convenzione	Descrizione
<b>PTR_OpenChannel</b>	Le parole in grassetto con la prima lettera maiuscola, indicano parole riservate per uno specifico linguaggio oppure nomi di comando.
<pre>for (i=0; i &lt; (INT16) Sess1.ResetLen; i++) { printf (" %02X", Sess1.ResetChar [i] ); }</pre>	Questo font viene usato per indicare il codice operativo.
<b>setup</b>	Parole in grassetto indicano di essere digitate
<i>[expressionlist]</i>	Sintatticamente, parole tra parentesi quadre indicano campi opzionali
<pre>Readout.text = "This should ↪ all be on one line in ↪ the code windows."</pre>	Il carattere di continuazione di linea (↪) indica che il codice che segue sulla successiva linea dovrebbe essere digitato in un'unica riga all'interno della "Code window".
CONSTANT.TXT	Parole composte da sole lettere maiuscole indicano nomi di file.
ENTER	Parole in maiuscoletto sono usate per indicare tasti o sequenze di tasti come ENTER o CTRL+R.
ALT+F1	Un segno (+) tra due nomi di tasto indica una combinazione di tasti. Per esempio, ALT+F1 significa di mantenere premuto il tasto ALT mentre si preme anche il tasto F1.



### ***PTR\_OpenChannel***

Tale comando richiede la connessione con il programma PTRserver (DDE\_server).

<b>Formato</b>	<i>C</i>	<code>__int16 PTR_OpenChannel (void)</code>
	<i>Visual Basic per Windows</i>	<code>Declare Function PTR_OpenChannel Lib "PTR.dll" () As Integer</code>

**Status codes** Vedi il paragrafo Status Codes per la lista dettagliata dei possibili codici di ritorno.

### ***PTR\_CloseChannel***

Tale comando termina la connessione con il programma PTRserver (DDE\_server).

<b>Formato</b>	<i>C</i>	<code>__int16 PTR_CloseChannel (void)</code>
	<i>Visual Basic per Windows</i>	<code>Declare Function PTR_CloseChannel Lib "PTR.dll" () As Integer</code>

**Status codes** Vedi il paragrafo Status Codes per la lista dettagliata dei possibili codici di ritorno.



### *PTR\_OpenSession*

Questo comando apre una sessione di lavoro con il formato carta desiderato (abilitando il PTR-RS3 o alimentando il PTR-RS).

**Formato**      *C*      `__int16 PTR_OpenSession (unsigned  
↪ int TrackNum, char *TrackType)`

*Visual Basic  
per Windows*      `Declare Function PTR_OpenSession Lib  
↪ "PTR.dll" (ByVal TrackNum As  
↪ Integer, ByVal TrackType As  
↪ String) As Integer`

**Parametri In.** La seguente tabella mostra la lista di parametri che possono essere dati a questo comando

<b>Parametri</b>	<b>Definizione</b>
TrackNum	Numero della traccia desiderata (da <b>1</b> a <b>3</b> ).
TrackType pointer	puntatore alla stringa contenente uno dei seguenti due tipi di standard: "ISO" oppure "IBM"

**Status codes**      Vedi il paragrafo Status Codes per la lista dettagliata dei possibili codici di ritorno.

### *PTR\_CloseSession*

Questo comando chiude una sessione di lavoro (disabilitando il PTR-RS3 o disalimentando il PTR-RS).

**Formato**      *C*      `__int16 PTR_CloseSession (void)`

*Visual Basic  
per Windows*      `Declare Function PTR_CloseSession Lib  
↪ "PTR.dll" () As Integer`

**Status codes**      Vedi il paragrafo Status Codes per la lista dettagliata dei possibili codici di ritorno.



### *PTR\_GetData*

Tale comando legge i messaggi non sollecitati provenienti dal lettore esterno e bufferizzati dal programma PTRserver.

<b>Formato</b>	<b>C</b>	<code>__int16 PTR_GetData (char *Data)</code>
	<b>Visual Basic</b>	<code>Declare Function PTR_GetData Lib</code>
	<b>per Windows</b>	<code>→"PTR.dll"(ByVal Data As String)</code>
		<code>→As Integer</code>

**Parametro In** La seguente tabella mostra la lista di parametri che possono essere dati a questo comando

<b>Parametro</b>	<b>Definizione</b>
Data pointer	puntatore alla stringa che conterrà i dati ricevuti

**Param. Out** Il comando in oggetto aggiorna i seguenti parametri:

<b>Parametro</b>	<b>Definizione</b>
Data	buffer dei dati ricevuti dal server (vedi "messaggi non sollecitati")

**Status codes** Vedi il paragrafo Status Codes per la lista dettagliata dei possibili codici di ritorno.



### *PTR\_SendCommand*

Questo comando invia uno dei comandi DDE disponibili al lettore esterno per mezzo del programma PTRserver.

<b>Formato</b>	<i>C</i>	<code>int16 PTR_SendCommand (char *Cmd, ↳char *Resp)</code>
	<i>Visual Basic per Windows</i>	<code>Declare Function PTR_SendCommand Lib ↳"PTR.dll" (ByVal Cmd As String, ↳ByVal Resp As String) As Integer</code>

**Parametri In** La seguente tabella mostra la lista di parametri che possono essere dati a questo comando

<b>Parametro</b>	<b>Definizione</b>
Cmd pointer	puntatore alla stringa che contiene il messaggio in uscita
Resp pointer	puntatore alla stringa che conterrà il messaggio di ritorno

**Param. Out** Il comando in oggetto aggiorna i seguenti parametri:

<b>Parametro</b>	<b>Definizione</b>
Resp	messaggio di ritorno ricevuto lettore esterno

**Status codes** Vedi il paragrafo Status Codes per la lista dettagliata dei possibili codici di ritorno.

### *Wake\_up*

Questo comando può essere usato per fermare un ciclo di attesa dal server dopo l'invio di un comando.

<b>Formato</b>	<i>C</i>	<code>void WakeUp (void)</code>
	<i>Visual Basic per Windows</i>	<code>Declare Function WakeUp Lib ↳"PTR.dll" ()</code>



### *Programma Dimostrativo*

Il programma **PtrRsDemo.exe** viene fornito a corredo quale esempio in aiuto agli sviluppatori; il linguaggio sorgente utilizzato è MSVC++ 5.0.

Dopo che il programma **PTRserver** è stato installato, anche il **PtrRsDemo** può essere lanciato; dopo il lancio visualizzerà le seguenti due sezioni:

- Parametri  
per selezionare:
  - 1) la traccia desiderata
  - 2) lo standard di lettura desiderato
  
- Dati  
per richiedere i dati dal buffer di lettura

Dopo la lettura del buffer, i dati relativi alla traccia selezionata vengono visualizzati.



Contenuto del dischetto a corredo

Di seguito la descrizione dei file forniti sul disco di supporto che comprende la Libreria d'Interfaccia PTR-RS ed il path di default dell'installazione guidata.

**Struttura di default dei file in ambiente Windows-32-bit**

<b>Path</b>	<b>File</b>	<b>Descrizione</b>
\\Programs\\Emk\\PTR\\server	PTRserver.exe	Server di comunicazione
\\Windows\\System	PTR.dll MSVCRT.dll MFC42.dll	PTR dynamic link library (amb.32 bit) dynamic link library per programma Demo dynamic link library per programma Demo
\\Programs\\Emk\\PTR\\demo	readme.txt PtrRsDemo.exe PtrRsDemo.cpp PtrRsApp.cpp PtrRsDemo.rc StdAfx.cpp PtrRsDemo.h PtrRsDemoDlg.h Resource.h StdAfx.h	Istruzioni per il programma Demo Eseguibile Demo Sorgente programma Main del Demo Risorse del Demo Tipi precompilati Prototipi del Main Prototipi del Dialogo Include per RC Prototipi dei precompilati

## CAPITOLO 5

# Connessioni

## Introduzione

In questo capitolo vengono descritte le connessioni disponibili sul dispositivo PTR-RS3.

## Cavo

Il PTR-RS3 viene fornito con un cavetto standard di lunghezza 2 metri; questo cavo termina con connettore a vaschetta a 9 vie (D-type) la cui topografia segnali viene fornita in seguito.

A corredo viene inoltre fornito un adattatore da 9 a 25 vie per collegare direttamente il PTR-RS3 ad un connettore DTE a 25 contatti.

## Interfaccia Fisica

L'interfaccia fisica tra il PTR-RS3 ed il suo controllore esterno (PC ecc.) è seriale di tipo asincrono; il PTR-RS3 trasmette i dati usando il segnale di trasmissione TxD\_PTR e riceve i comandi sulla linea RxD\_PTR.

La condizione "mark" del segnale di trasmissione viene derivata dal segnale di ricezione mediante una resistenza (pull-down).

I livelli del segnale TxD\_PTR, misurati con il PTR-RS3 connesso ad una interfaccia RS-232 standard (desk-top PC) sono i seguenti:

Livello	Min	Max
SPACE	+3V	+5V
MARK	-3V	-12V

Il PTR-RS3 viene alimentato quando entrambi i segnali DTR e RTS raggiungono la condizione "space" (livello RS-232 positivo).

## Connettori RS-232

La topografia dei segnali di un connettore DTE a 25 vie (lato PC) è la seguente:

01	02	03	04	05	06	07	08	09	10	11	12	13
sch	TxD	RxD	RTS			GND						
	14	15	16	17	18	19	20	21	22	23	24	25
							DTR					

La topografia dei segnali di un connettore DTE a 9 vie con contatti femmina (lato PC) è la seguente:

05	04	03	02	01
GND	DTR	TxD	RxD	
	09	08	07	06
		RTS		

## Connettore interno

Questo connettore è montato direttamente sulla piastra interna; la calza del cavo viene collegata alla struttura metallica con una vite disponibile nei pressi del connettore stesso.

Il connettore interno è del tipo AMP-MODU-II a 6 contatti .

La topografia dei suoi segnali è la seguente:

J2

1	2	3	4	5	6
DTR	RTS	TxD	RxD	VCC	GND



## CAPITOLO 6

# Caratteristiche fisiche

### Dimensioni e pesi

Le qualità meccaniche del dispositivo PTR-RS3 ne garantiscono la robustezza e lo rendono facilmente installabile.

La struttura metallica viene solitamente verniciata di colore grigio.

Le dimensioni fisiche del PTR-RS3 sono le seguenti:

**197 x 43 x 36 mm.**

Il peso è di circa **500** grammi.

### Condizioni ambientali

Il modulo PTR-RS3 è ideato per funzionare alle seguenti condizioni:

	Operativa	Immagazzinamento
TEMPERATURA	0 °C / +50°C	-30°C / +70°C
UMIDITA' (senza cond.)	5 % / 95 % RH	5 % / 95 % RH
ALTITUDINE	Liv.mare/4.000mt.	Liv.mare/4.000mt.



## Prestazioni

Per calcolare il tempo totale necessario per dare una risposta all'utente dopo una lettura del badge, devono essere prese in considerazione le seguenti voci:

- **Tempo di scorrimento:** può variare tra 300 msec. e 2 secondi.  
(l'azione manuale può variare tra 10 e 100 cm./sec)
- **Tempo di trasmissione:** questo tempo dipende dalla lunghezza del messaggio e dalla velocità di trasmissione (un carattere @ 4.800 bps dura 2 msec.)
- **Tempo di elaborazione:** 10 msec. (tempo presunto di elaborazione da parte del PC-Host)

Dalle suddette considerazioni ne consegue che il PTR-RS3 è in grado di dare la risposta all'utente in un tempo compreso tra i seguenti valori:

- tempo **minimo: 350 msec.**  
ottenuto alla massima velocità di scorrimento con pochi caratteri e con velocità di trasmissione pari a 4.800 bps
- tempo **massimo: 4 sec.**  
ottenuto alla minima velocità di scorrimento con molti caratteri e con velocità di trasmissione pari a 1.200 bps



## **Affidabilità**

In considerazione delle caratteristiche del PTR-RS3, i guasti meccanici vanno presi in esame separatamente rispetto a quelli elettrici.

L'**MTBF** calcolato della sola **electronica** è: **60.000 ore**

Il valore dell'**MTBF** delle rimanenti parti meccaniche, viene espresso in termini di cicli operativi come segue:

- **testine magnetiche:** **1.000.000 operazioni**
- **switch presenza badge:** **1.000.000 operazioni**



Sede Legale C.so M.d'Azeglio, 76 – 10126 - TORINO  
C.F. – P.Iva: 07169270019  
Capitale sociale Euro 71.136 i.v.  
R.E.A.. di Torino N 852530  
C.M. estero TO 086651

Sede Centrale Via Cuneo n°14  
10090 - S.BERNARDO D'IVREA (TO)  
Tel. 0125 - 631887  
Fax 0125 - 631935  
E-Mail: [ufficio.commerciale@emk.it](mailto:ufficio.commerciale@emk.it)

## REPORT FORM

A: **EMK srl** (Supporto Tecnico)

Fax: +39 125 631935

Il Vostro supporto è per noi essenziale. Vi preghiamo di utilizzare questo prospetto per:

- segnalare qualsiasi problema dovesTe incontrare sul prodotto o sulla documentazione ad esso riferita
- suggerire variazioni o migliorie per le versioni successive

Il nostro grazie per la Vostra collaborazione.

### **Vostrì riferimenti:**

Nome \_\_\_\_\_

Tit. \_\_\_\_\_

Società \_\_\_\_\_

Tel. \_\_\_\_\_

Indirizzo \_\_\_\_\_

Fax \_\_\_\_\_

– \_\_\_\_\_

Nazione \_\_\_\_\_

– \_\_\_\_\_

### **Dettagli:**

Prodotto \_\_\_\_\_

Versione \_\_\_\_\_

Numero di pagine \_\_\_\_\_

Data \_\_\_\_\_

Si prega di descrivere qui di seguito il problema o gli eventuali suggerimenti: