



Il linguaggio macchina sullo Spectrum

Quinta parte

Dopo la parentesi del numero scorso, in questo numero di TuttoSpectrum riprendiamo l'argomento gestione dell'ingresso-uscita in linguaggio macchina. Nella puntata precedente abbiamo visto come sia semplice scrivere un carattere sullo schermo. È sufficiente caricare il codice del carattere nel registro A dello Z80 e poi utilizzare l'istruzione RST 10. Mediante l'invio in stampa di particolari sequenze di caratteri è possibile modificare a piacimento la posizione in cui appariranno sul video i caratteri, sortendo lo stesso effetto che in Basic si ottiene con la funzione AT.

Vi avevamo proposto come esempio un semplice programma in linguaggio macchina che produceva la stampa di una stringa (una sequenza di caratteri) sul video. Il nucleo del programma è costituito da un ciclo che ripete una RST 10 (ovvero una chiamata della routine di stampa) tante volte quanti sono i caratteri da stampare:

```

LOOP LD  A,B  parte più significativa
           del contatore
      OR  C    l'OR di B e C è 0 se
           la stringa è terminata
      DEC BC   decrementa il contatore
           di lunghezza
    
```

```

RET  Z      termina se la lunghezza
           è 0
LD   A,(DE) carica in A il carattere
           da stampare
INC  DE     si posiziona sul carattere
           successivo
RST  #10    esegue la stampa
JR   LOOP   ripete il ciclo
    
```

Questo segmento di programma è in realtà contenuto anche nella ROM dello Spectrum a partire dalla locazione esadecimale 203C. Ogni volta che in un programma è richiesta la stampa di una stringa non è necessario quindi riscrivere questo sottoprogramma, ma è sufficiente andare a richiamare quello contenuto nella ROM mediante l'istruzione

```
CALL #203C
```

Naturalmente occorre che all'atto della chiamata la coppia di registri DE contenga l'indirizzo della locazione contenente il primo carattere della stringa, e la coppia BC contenga un valore pari al numero di caratteri contenuti nella stringa.

Gli attributi dei caratteri

Prima di proseguire il nostro discorso sulla gestione dello schermo in linguaggio macchina sarà bene fare una

piccola digressione su come lo Spectrum gestisce la memoria video, ovvero quella zona della memoria in cui sono contenute le informazioni su cosa deve essere rappresentato sullo schermo del televisore.

Come ben sapete lo schermo dello ZX Spectrum è suddiviso in 768 (= 24 righe per 32 colonne) quadratini di 8 per 8 pixel. Il pixel (abbreviazione inglese per picture element) è la più piccola parte di schermo che può essere controllata, accesa o spenta, singolarmente. Complessivamente quindi lo schermo è costituito da $256 \times 192 = 49152$ pixel.

Non è possibile far assumere a ciascun pixel il colore che si desidera; esistono infatti delle limitazioni. Per ciascuna area di 8 per 8 pixel in cui è suddiviso lo schermo, è possibile specificare un unico colore di sfondo, che i pixel assumeranno quando sono «spenti», ed un unico colore per il primo piano, che i pixel assumeranno quando sono «accesi». Per analogia con un foglio da disegno, questi due colori vengono di solito indicati come «colore della carta» (PAPER) e «colore dell'inchiostro» (INK).

Tutto ciò viene rappresentato in memoria a partire dall'indirizzo 16384. Tale rappresentazione è suddivisa in due parti.

Nella prima esiste un bit di memoria per ogni pixel del video. Se tale bit vale uno il corrispondente pixel assume il colore dell'inchiostro, se vale zero il colore della carta. Poiché i pixel del video sono 49152, ed ogni byte contiene 8 bit, per una rappresentazione completa dello schermo occorrono $6144 (= 49152 \div 8)$ locazioni di memoria, quelle che vanno dalla 16384 alla 22527. Come sia organizzata la corrispondenza tra locazioni di memoria e pixel del video è un argomento piuttosto complesso e per il momento non ci riguarda, ci torneremo quando parleremo di grafica in alta risoluzione in uno dei prossimi numeri.

Le informazioni relative al colore di ciascuna delle 24 per 32 aree in cui è suddiviso lo schermo richiedono un byte ciascuna. Il formato con cui vengono memorizzate tali informazioni è evidenziato in figura 1 assieme alla corrispondenza tra locazioni di memoria e aree dello schermo. Come vedete vengono considerati quattro elementi: il colore della carta, quello dell'inchiostro, la luminosità ed il lampeggio. Ricordiamo che quando un carattere lampeggia il colore dell'inchiostro e quello della carta si invertono tra loro periodicamente. Le 768 locazioni di memoria necessarie per contenere queste informazioni, che d'ora in poi chiameremo attributi, vanno dall'indirizzo 22528 all'indirizzo 23295.

Per conoscere il valore degli attributi

ti relativi ad una certa posizione del video si può usare la funzione Basic ATTR nel formato

ATTR (<numero riga>,
<numero colonna>)

dove <numero riga> è un intero tra 0 e 23 e <numero colonna> un intero tra 0 e 31. Il valore restituito è la rappresentazione in decimale del contenuto della locazione di memoria corrispondente. Quando lo Spectrum viene acceso il valore degli attributi è carta bianca, inchiostro nero, luminosità normale e niente lampeggio. Il valore restituito dalla funzione ATTR è 56 per tutto lo schermo.

Gli attributi corrispondenti alla parte superiore del video (quella dove vengono listati i programmi per intenderci) possono essere modificati in maniera permanente mediante le istruzioni Basic PAPER, INK, FLASH, BRIGHT. Esistono altri due comandi correlati più o meno direttamente alla modifica degli attributi, sono INVERSE e OVER. Per una descrizione dettagliata del loro funzionamento vi rimandiamo al capitolo 16 del manuale d'uso.

Nella parte inferiore dello schermo, quella dove normalmente appaiono i messaggi di sistema e le linee di Basic da editare, il colore di sfondo è sempre uguale al colore del contorno dello schermo (BORDER). Il comando Basic BORDER serve appunto a modificare il colore della cornice ed il colore della carta nella parte inferiore del video. Il Basic però non consente espressamente di variare a piacimento in questa zona il colore dell'inchiostro, così come non è possibile selezionare il modo alta luminosità né quello lampeggio. Il colore dell'inchiostro viene scelto automaticamente, bianco o nero, dal sistema in modo da contrastare con quello dello sfondo. Vedremo tra breve come sia possibile controllare in modo completo gli attributi della parte inferiore dello schermo agendo sulle variabili di sistema.

Modificare gli attributi usando il linguaggio macchina

La codifica illustrata in figura 1 relativa ai byte dell'area attributi è comune anche a cinque variabili di sistema, esse sono BORDCR, ATTR-P, ATTR-T, MASK-P, MASK-T, dove la T e la P finale stanno rispettivamente per temporaneo e permanente.

Le variabili ATTR-T e ATTR-P contengono i valori correnti degli attributi. Tali valori vengono ricopiati nell'area attributi della memoria ogni volta che viene eseguita un'operazione che riguarda il video. Le variabili MASK-T e MASK-P permettono inve-

ce di ricordare se è stato specificato il parametro di colore 8, che ha il significato di trasparente. Sul significato del colore «trasparente» potete consultare il manuale a pagina 83.

Nella maggior parte delle situazioni lo Spectrum utilizza i valori temporanei, ad esempio quando viene eseguita un'operazione RST #10 di scrittura di un carattere. Il caso più significativo di uso dei valori permanenti è la routine di cancellazione dello schermo CLS.

Modificare il valore contenuto in ATTR-P è equivalente ad utilizzare le istruzioni Basic PAPER, INK, BRIGHT, FLASH, come comandi, ovvero non all'interno di uno statement PRINT. Quando questi comandi vengono impartiti da Basic il loro effetto non è immediato. I nuovi valori permanenti per carta ed inchiostro vengono di volta in volta copiati in quelli temporanei ed ogni nuovo carattere che viene scritto sullo schermo appare con i nuovi attributi. I caratteri presenti già in precedenza rimangono tali e quali.

Perché vengano modificati gli attributi di tutta la parte superiore dello schermo è sufficiente che venga eseguita un'operazione di cancellazione dello schermo, che ha sempre come ef-

fetto collaterale quello di ricopiare gli attributi permanenti in tutta l'area attributi del video. Ciò accade per esempio quando viene eseguita l'istruzione CLS oppure viene battuto ENTER a vuoto.

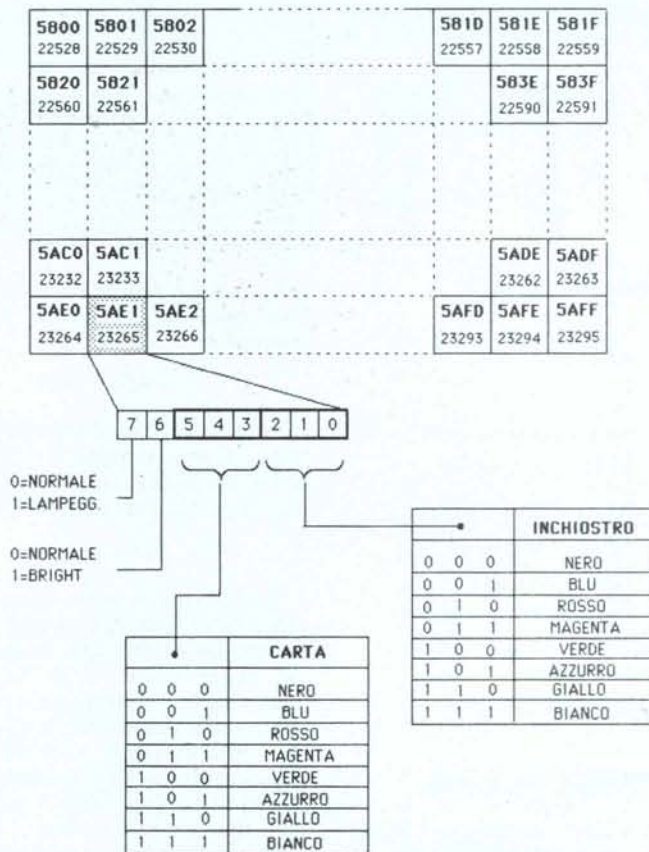
In linguaggio macchina accade la stessa cosa: caricando un valore in una variabile permanente non si ottengono risultati immediati. Occorre che il nuovo valore degli attributi venga copiato nell'area attributi della memoria. Ciò può essere eseguito espressamente mediante un apposito segmento di programma, oppure si può utilizzare la routine Assembler che esegue la cancellazione dello schermo, quella chiamata dall'istruzione CLS del Basic. Tale chiamata avviene mediante la

CALL #0D6B

Viceversa, modificare i valori nell'area attributi senza modificare corrispondentemente il valore contenuto in ATTR-P ha un effetto solamente temporaneo, in quanto alla prima occasione il sistema ristabilirà i valori precedenti.

La variabile BORDCR controlla gli attributi relativi alla parte inferiore dello schermo ed alla cornice. Il comportamento è analogo a quello di

Figura 1



ATTR-P, scrivendo gli opportuni valori è possibile ottenere qualsiasi combinazione di attributi si desidera, anche nella parte inferiore dello schermo.

Volendo cambiare il colore della cornice con effetto immediato, la sequenza di istruzioni da eseguire è:

```
LD A, <colore>
OUT (+FE),A
```

provvede a inviare alla ULA (l'integrato che nello Spectrum gestisce il video) la opportuna informazione.

Quando il Basic esegue una istruzione di stampa come PRINT, INPUT o PLOT utilizza i valori temporanei degli attributi. Di solito tali valori temporanei coincidono con quelli permanenti, a meno che non sia stato specificato all'interno della istruzione un parametro differente, come ad esempio nel segmento seguente:

```
10 PAPER 7: INK 0
20 CLS
30 PRINT «esempio»
40 PRINT PAPER 1; INK 2; «esempio».
```

Il valore di questi parametri è limitato alla istruzione di PRINT, INPUT, o PLOT che viene eseguita.

In Assembler si può utilizzare un meccanismo perfettamente analogo per mezzo della RST 10. Occorre inviare in stampa speciali codici seguiti dalla determinazione del parametro temporaneo che si vuole impostare. I codici sono:

| SIGNIFICATO | DECIMALE | ESADEC. |
|-------------|----------|---------|
| INK | 16 | 10 |
| PAPER | 17 | 11 |
| FLASH | 18 | 12 |
| BRIGHT | 19 | 13 |
| INVERSE | 20 | 14 |
| OVER | 21 | 15 |

Per esempio la sequenza:

```
0E 02 LD A,#02 apre il canale "C"
0D 01 16 CALL #1601
0D 0E 0D CALL #0D0E cancella lo schermo
0E 02 LD A,#02 riapre il canale "E"
0D 01 16 CALL #1601
0E 11 LD A,#11 codice di PAPER
07 RST #10
0E 02 LD A,#02 codice di RED
07 RST #10
0E 1A LD A,#1A codice di INK
07 RST #10
0E 04 LD A,#04 codice di GREEN
07 RST #10
0E 40 LD A,#40 "MC"
07 RST #10
0E 40 LD A,#40 "C"
07 RST #10
0E RET
```

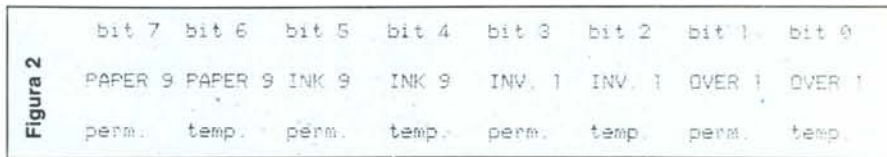
è equivalente a:

```
10 CLS
20 PRINT PAPER 2; INK 4; «MC»
```

Può essere utile cambiare i valori temporanei nei permanenti e viceversa. Per copiare i valori permanenti nel-

| Tabella 1 | VARIABILE | DECIMALE | ESADECIMALE |
|-----------|-----------------|----------|-------------|
| | ATTR-P | 23693 | 5C8D |
| | MASK-P | 23694 | 5C8E |
| | P-FLAG | 23697 | 5C91 |
| FLASH 0 | RES 7, (ATTR-P) | | |
| FLASH 1 | SET 7, (ATTR-P) | | |
| FLASH 8 | SET 7, (MASK-P) | | |
| BRIGHT 0 | RES 6, (ATTR-P) | | |
| BRIGHT 1 | SET 6, (ATTR-P) | | |
| BRIGHT 8 | SET 6, (MASK-P) | | |
| OVER 1 | SET 1, (P-FLAG) | | |
| OVER 0 | RES 1, (P-FLAG) | | |
| INVERSE 1 | SET 3, (P-FLAG) | | |
| INVERSE 0 | RES 3, (P-FLAG) | | |
| PAPER 8 | SET 3, (MASK-P) | | |
| | SET 4, (MASK-P) | | |
| | SET 5, (MASK-P) | | |
| PAPER 9 | SET 7, (P-FLAG) | | |
| INK 8 | SET 0, (MASK-P) | | |
| | SET 1, (MASK-P) | | |
| | SET 2, (MASK-P) | | |
| INK 9 | SET 5, (P-FLAG) | | |

-- N.B. Il formato corretto dell'istruzione assembler è'
SET n, (HL)
occorre quindi prima caricare in HL il valore contenuto tra parentesi, e poi eseguire l'istruzione indicata.
Esempio:
PAPER 9 = LD HL, P-FLAG
SET 7, (HL)



le variabili temporanee conviene usare una apposita routine di sistema. La chiamata è:

```
CALL #0D4D
```

Se viceversa i valori temporanei devono essere ricopiati nelle variabili permanenti, la relativa routine di sistema viene chiamata tramite la:

```
CALL #1CAD
```

Queste due routine, oltre a modificare i valori contenuti nelle variabili di sistema già citate, aggiornano il contenuto della variabile P-FLAG, che contiene i valori temporanei e permanenti relativi alle istruzioni Basic PAPER 9, INK 9, INVERSE e OVER. Per il significato di queste istruzioni vi rimandiamo al capitolo sedici del manuale.

In figura 2 è illustrata la disposizione dei vari flag nella variabile di sistema P-FLAG. In tabella 1 vi è una ta-

vola di conversione tra le istruzioni Basic FLASH, BRIGHT, OVER, INVERSE e le corrispondenti istruzioni Assembler che ottengono lo stesso effetto. In più sono riportate le codifiche corrispondenti a PAPER 8 e INK 8.

Abbiamo visto come sia possibile avere una gestione completa degli attributi di colore anche in linguaggio macchina. In almeno un caso, quando abbiamo parlato dell'istruzione BORDER, abbiamo visto come l'Assembler ci offra qualche possibilità in più rispetto al Basic. Tale aspetto sarà ancora più evidente quando nel prossimo numero ci occuperemo più specificamente della cancellazione dello schermo e dello scrolling del testo sul video. Vedremo infatti come sia possibile, chiamando apposite routine della ROM, cancellare solo una porzione di schermo, oppure fare scorrere solo un numero predeterminato di linee. **MC**

INTERFACCE PER APPLE

| | |
|---------------------------|---------|
| Controller Doppio Drive | 65.000 |
| 16K RAM | 83.000 |
| Language Card | 101.000 |
| 80 Colonne Soft/Switch | 118.000 |
| 8088 Card | 690.000 |
| Eprom Writer (16-64) | 110.000 |
| Prom Writer | 493.000 |
| Z/80 Card | 61.000 |
| RS-232 con cavo | 100.000 |
| Epson Printer e cavo | 88.000 |
| Grappler e cavo | 98.000 |
| Buffer 16K e cavo | 265.000 |
| Grappler + Buffer 16K | 397.000 |
| 128K RAM | 364.000 |
| AD-DA 12 Bit/16 canali | 504.000 |
| AD Card | 177.000 |
| DA Card | 298.000 |
| IEEE-488 | 264.000 |
| 6809 Card | 356.000 |
| Communication Card | 110.000 |
| Super Serial Card | 136.000 |
| Pal Color Card | 83.000 |
| RGB (8 colori) | 124.000 |
| RGB II (16 colori) | 194.000 |
| Stereo Music Card | 138.000 |
| Scheda Parlante | 87.000 |
| Wild Card | 87.000 |
| Scheda orologio | 124.000 |
| 6522 Card | 155.000 |
| Forth Card | 131.000 |
| I.C. Test Card | 307.000 |
| 80 Colonne + 64K IIE | 55.000 |
| 80 Colonne IIE | 26.000 |
| Adattatore x Drive IIC | 20.000 |
| Adattatore x Joystick IIC | 14.000 |

DRIVE PER APPLE

| | |
|----------------------|---------|
| Lisonic LS-39A | 168.000 |
| Chinon 051-All | 288.000 |
| Mitac Ad-8 | 268.000 |
| Chinon 360K + Contr. | 470.000 |

STAMPANTI APPLE - IBM

| | |
|---|-----------|
| — Star Gemini 10X (120 CPS./Dow Load) | 535.000 |
| — C.T.I. CPB-80 (130 CPS./2K Buffer/D. Load/Set IBM) | 550.000 |
| — Copal Sc-1200 L (120 CPS./N.L.Q./Set IBM) | 650.000 |
| — KDC-FT 5002 NEW (120 CPS./1K Buffer/Down Load/NQL/Set IBM) | 690.000 |
| — C.I.T. CPB-136 (130 CPS/2K Buffer/D. Load/Set IBM) | 860.000 |
| — Copal SC-5500I (180 CPS/132 Col./3K Buffer/D. Load/N.L.Q./Set IBM) | 1.150.000 |

INTERFACCE PER PC/XT IBM

| | |
|----------------------------|---------|
| Controller 4D. + Cavo | 196.000 |
| Printer Card | 112.000 |
| Color Graphic | 280.000 |
| Color Graphic + Printer | 370.000 |
| Monoc. Gaph. (Hercules II) | 320.000 |
| Multifunction 256K | 274.000 |
| Multifunction 384K | 364.000 |
| AD-DA (12 Bit - 16 Can.) | 499.000 |
| 512 RAM (Ø RAM) | 166.000 |
| RS-232 | 160.000 |
| Game I/O | 88.000 |
| Eprom Writer (16-256) | 430.000 |
| 8255 | 290.000 |
| Rete Locale I-NET | 980.000 |

DRIVE PER PC/XT IBM

| | |
|---------------------|---------|
| Chinon 502 | 299.000 |
| Chinon 502L | 347.000 |
| Matsushita 561 I MB | 480.000 |

INTERFACCE E DRIVE X AT

| | |
|--------------------------------------|-----------|
| AT Controller | 460.000 |
| AT Parallel/Serial Card | 280.000 |
| AT Multifunction 2,5 MB | 680.000 |
| AT H. Disk Controller + 2FDD | 1.380.000 |
| Hard Disk Teac 10 MB (senza contr.) | 1.320.000 |
| Hard Disk Seagate 20 MB (senza con.) | 1.690.000 |

AT COMPATIBILE



Versione Base: Main Board ØK espandibile ad 1 M.B., alimentatore 200 W, Cabinet in metallo, tastiera
L. 3.950.000

II E COMPATIBILE + PAD NUMERICO

128K RAM + 80 Colonne



Il E compatibile al 100% dotato di comodo Pad numerico. Viene fornito completo di scheda 80 colonne + 64K
L. 760.000

II E COMPAT. CON TASTIERA SEPARATA

128K RAM + 80 Colonne



Il E compatibile al 100%. Versatile realizzazione con tastiera separata, intelligente, Pad numerico, Tasti funzione definibili, Autorepeat e «Bip» disinseribile. Possibilità di inserimento di N. Due Drive Slim all'interno del Cabinet.
L. 920.000

PREZZI
IVA
ESCLUSA



LA CASA DEL
COMPUTER

IMPORTAZIONE DIRETTA

0587
212.312

VIA MISERICORDIA, 84 - 56025 PONTEDERA (PI)

**FRA TUTTI I COMPATIBILI
IL NOSTRO GIRA PIÙ VELOCE**

PC/XT TURBO

L. 1.680.000

Clock 6,67-4,77 MHz
Main Board Esp. 640K



N. 1 Drive DS/DD 360K controller,
Main Board ØK espandibile A 640K,
Alimentatore 130 W, Tastiera K5 S

PC/XT STANDARD (4,77 MHz)

L. 1.420.000

Configurazione come sopra ma con Main Board 128K espandibile a 256K

*** Per le interfacce video vedere listino***

| | |
|---------------------------------|------------|
| Monitor Philips Monocr. x IBM | L. 227.000 |
| Monitor Cabell MR Colori x IBM | L. 439.000 |
| Monitor Philips HR Colori x IBM | L. 690.000 |

ABBIAMO PRONTA CONSEGNA:

- Stampanti 120-130-180 CPS — Modem
- Accoppiatori acustici — Monitor
- Hard Disk 10-20 MB — STREAMER 20 MB
- Plotter — Digitizer

RICHIEDETEICI LO «SPECIALE PROMOZIONE»

DATAFLEX

PROFESSIONAL 5 1/4



CON BOX IN PLASTICA OMAGGIO!!! SCONTI PER QUANTITÀ

| SINGOLA F. - DOPPIA D. | DOPPIA F. DOPPIA D. |
|------------------------|---------------------|
| 200 Pezzi 2.230 | 200 Pezzi 2.870 |
| 100 Pezzi 2.350 | 100 Pezzi 3.150 |
| 30 Pezzi 2.550 | 30 Pezzi 3.400 |

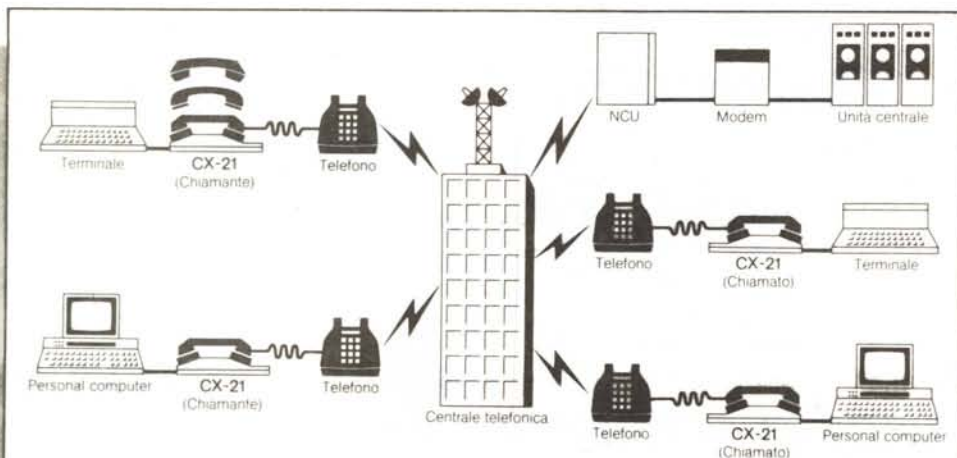
- I dischetti dataflex sono prodotti da uno dei più grossi fabbricanti americani che garantisce l'altissima qualità ed affidabilità.
- Uno speciale ed esclusivo strato «MULTICOT» protegge la superficie dall'usura del contatto con le testine garantendo minimo ben 10.000.000 di passaggi!!!
- La sicurezza dei Vs. dati è assicurata dall'ineccepibile supporto magnetico di primissima qualità.

**DATO L'INSTABILE MERCATO DEI CAMBI PREGASI TELEFONARE PER CONFERMA PREZZI E DISPONIBILITÀ
— RICHIEDETEICI IL CATALOGO — SCONTI AI SIG. RIVENDITORI**

EPSON • SEGI
Garanzia delle
Quattro
Stagioni
12 mesi di garanzia sempre su tutti i prodotti EPSON

DAI

un telefono anche al tuo computer



EPSON CX-21

Accoppiatore acustico



L'accoppiatore acustico CX-21 della Epson è uno strumento conveniente e affidabile dalle grandi prestazioni che dà la nuova dimensione delle telecomunicazioni al vostro computer.

Il CX-21 funziona a batterie ricaricabili ed è particolarmente adatto ai personal computer portatili, come il PX-8 e HX-20, per esempio.

Per accedere alle informazioni di una banca dati, o più semplicemente per comunicare con un altro computer basta inserire la cornetta e comporre il numero telefonico corrispondente.

Leggero, compatto, elegante ed affidabile il CX-21 è progettato per adattarsi a tutti gli apparecchi telefonici e consentire lo scambio rapido e sicuro di programmi e dati.

Caratteristiche

- Realizzato con speciali circuiti integrati CMOS
- Funzionamento a batterie NiCd ricaricabili
- Funzionamento full-duplex o half-duplex
- Velocità di trasmissione fino a 300 bit-sec.
- Interfaccia RS 232C
- Dimensioni 297x95x42 mm
- Peso 700 gr.

la voce del tuo personal



EPSON-SEGI S.p.A.
 20124 Milano - Via Timavo, 12 - Tel. 02-6709136-7-8-9-0
 40121 Bologna - Via Pietramellara, 65 A/B - Tel. 051-273686
 35128 Padova - Via Pellizzo, 23/9 int. 4/U - Tel. 049-8070870
 00199 Roma - Via Asmara, 58 - Tel. 06-8395786

studio martinetti