



PERSONAL COMPUTER PRODUCTS

20030 SENAGO (MI) ITALY

Via Monterosa, 22 - Tel. (02) 99.89.407

STAMPA PROVVISORIA

MANUALE PER FLOPPY DISK DRIVE NEW

Leggere attentamente quanto contenuto
in questo manuale, eviterete spiacevoli
inconvenienti ed imparerete ad apprezzare
tutte le caratteristiche di questa interfaccia...

PREFAZIONE

Le grandi quantita' di dati che un computer apparentemente piccolo come lo Spectrum puo' elaborare, richiedono memorie di massa ad alta velocita' ed efficienza, per eliminare da un canto la necessita' di lunghe attese per l'INPUT e l'OUTPUT delle informazioni, e dall'altro per consentire l'accesso RANDOM non solo ai file, ma anche ad ogni singolo record.

Lo strumento tecnico piu' idoneo a raggiungere questi obiettivi e' il FLOPPY DISK DRIVE e la SANDY e' orgogliosa di presentare il sistema ad alte prestazioni FDD2.

Esso e' dotato di un D.O.S. specificamente studiato per lo Spectrum, del quale utilizza le opportune parole-chiave di comando.

Ne risulta una velocita' di ricerca e di caricamento dati che consente tempi decine di volte inferiori a quelli consentiti dal registratore, nonche' la possibilita' di svincolarsi dalla capacita' RAM della macchina tramite un velocissimo accesso RANDOM ai dati su disco.

L' INTERFACCIA

L' interfaccia e' racchiusa in un contenitore di alluminio, dal quale sporgono, anteriormente il connettore per il computer, posteriormente il BUS a cui e' possibile collegare altre interfacce, poiche' mantiene esattamente identiche le tracce della scheda madre dello Spectrum.

Sulla parte laterale sinistra, all'interno del foro, si trova la presa per lo spinotto d'alimentazione, che non deve essere collegato al computer ma all' interfaccia.

L' interfaccia FDD2 contiene in un' apposita ROM il sistema operativo dei dischi, ovvero un programma in linguaggio macchina che permette all' utente di utilizzare il drive e di operare sui dischi in modo estremamente semplice.

IL DRIVE

Il drive e' sostanzialmente un giradischi ad altissima precisione, che puo' sia leggere che scrivere dati su un supporto magnetico a forma di disco.

Differentemente da quanto siamo abituati con i dischi audio, che hanno un' unica pista a forma di spirale, sulla quale la puntina passa dalla zona piu' esterna verso il centro, i dischi dei dati vengono impressi su piste circolari concentriche chiamate tracce (in inglese TRACKS).

Il loro numero su di una facciata puo' essere di 40 (singola faccia), o di 80 tracce (doppia faccia).

Un disco con piu' tracce contiene proporzionalmente piu' dati e ha bisogno pero' di un drive apposito con una meccanica di maggior precisione e costo.

All' interfaccia FDD2 e' possibile collegare contemporaneamente ben tre unita' di drive.

QUESTO MANUALE...

* Il primo capitolo mostra la corretta installazione dell' interfaccia e del drive al tuo Spectrum.

Contiene inoltre tutti quei suggerimenti che l' esperienza consiglia per un lungo ed efficace utilizzo del sistema.

Leggi molto attentamente, perche' molti degli argomenti sono di importanza vitale per la macchina.

* Il secondo capitolo ti introduce nel nuovo ambiente operativo, insegnandoti ad utilizzare il drive e ad operare con un disco nuovo.

* Il terzo capitolo ti informa sui comandi del sistema operativo dei dischi, cioe' sul loro significato, sintassi, effetto. Per ogni comando viene riportato un esempio.

* Il quarto capitolo ti mostra come adattare i tuoi vecchi programmi in modo che possano girare con la velocita' impressionante del floppy disk drive.

Una libreria di programmi su disco significa una maggiore efficienza ed una minore perdita di tempo.

* L'accesso RANDOM ai file e' la nuova arma potente che il sistema FDD2 implementa sul tuo ZX Spectrum.

Come utilizzarla, come gestirla, come sfruttarne le incredibili risorse e' l'obiettivo del quinto capitolo .

QUESTO MANUALE...

* Il primo capitolo descrive la semplice installazione interfaccia e del drive al tuo Spectrum.

Contiene inoltre una guida per il sistema di gestione dei file.

Importante anche per la macchina.

Il secondo capitolo descrive il nuovo sistema di gestione dei file e la sua interfaccia con il sistema di gestione dei file.

Il terzo capitolo ti informa sui comandi del sistema di gestione dei file.

CAPITOLO 1

La confezione contiene :

- 1 - l' interfaccia,
- 2 - il floppy disk drive (da 3" o 5")
- 3 - il cavo piatto a 34 poli di collegamento
- 4 - un dischetto di sistema
- 5 - Un manuale d'uso

Il drive contiene una meccanica ad alta precisione per cui e' necessario evitare qualsiasi urto.

Per attivare il sistema, procedere nel modo seguente :

- 1 - assicurarsi che l'alimentazione del computer sia disinserita
- 2 - collegare con l'apposito cavo piatto l'interfaccia al floppy disk drive
- 3 - inserire il connettore a pettine dell'interfaccia nella apposita finestra posteriore del computer
- 4 - togliere il disco protettivo di cartone dal drive (5")
- 5 - inserire lo spinotto d'alimentazione nella apposita presa situata sul lato destro dell'interfaccia
- 6 - collegare alla rete la spina dell'alimentatore
- 7 - controllare che sullo schermo appaia la scritta " SANDY ...
.....
- 8 - controllare l'avvenuta accensione del diodo led posto anteriormente allo stesso

Adesso il sistema e' pronto per essere utilizzato : lo Spectrum e' in ambiente DOS cioe' risponde soltanto ai comandi del sistema operativo dischi.

INCONVENIENTI DI FUNZIONAMENTO E CONSIGLI PER UN USO CORRETTO

Se non appare la scritta sullo schermo :

- verificare il collegamento alla rete di alimentazione
- verificare il fusibile dell'alimentatore
- togliere l'alimentazione e riefettuare il collegamento tra interfaccia e Spectrum

Se dopo l'accensione del drive il motore gira in continuo :

- spegnere il drive e disinserire l'alimentazione (220v)
- controllare che la banda rossa del cavo sia verso l'alto ; se non lo fosse, estrarre il cavo a 34 poli e capovolgerlo

ECCO ALCUNI SUGGERIMENTI PER UN USO CORRETTO

- A) Nel caso il sistema si bloccasse premere il tasto di reset posto sul lato destro dell'interfaccia.
- B) Evitare che il drive venga a contatto con liquidi e proteggerlo dalla polvere.
- C) Per un corretto funzionamento, controllare che il pettine di collegamento tra drive e interfaccia sia bene inserito.
- D) In caso di collegamento di ulteriori interfacce, quella del disk drive deve essere sempre collegata per prima cioè direttamente al connettore del computer; una configurazione già collaudata e che contiene le interfacce più utilizzate, è la seguente :
 1. INTERFACCIA DISCO
 2. INTERFACCIA JOYSTICK
 3. INTERFACCIA PER STAMPANTE
- E) ESTRARRE SEMPRE IL DISCHETTO DAL DRIVE PRIMA DI SPEGNERLO. SPEGNERE IL DRIVE PRIMA DI STACCARE L'ALIMENTAZIONE GENERALE. QUALSIASI OPERAZIONE DI CONNESSIONE O SCONESSIONE DI QUESTA COME DI ALTRE INTERFACCE DEVE ESSERE FATTA A COMPUTER SPENTO

Adesso il sistema è pronto per essere utilizzato. Il Spectrum è un sistema di grande potenza e può essere utilizzato per molte applicazioni.

Se non appare la scritta sullo schermo...

Interfaccia e Spectrum

CAPITOLO 2

A questo punto ti trovi con il computer e il drive accesi, sul monitor e' presente il messaggio di copyright, il DOS ti chiede di inserire la password.

P A S S W O R D : " "

Infatti il nostro sistema e' fornito di questa protezione dei dischi i quali non potranno essere letti senza aver inserito la parola chiave esatta. Questa parola chiave viene impressa durante l'inizializzazione di un dischetto nuovo operazione che prende nome di FORMATTAZIONE. La formattazione e' necessaria per tutti i dischi vergini, e racchiude le seguenti operazioni:

- Memorizzazione della password
- Definizione, controllo delle tracce
- Memorizzazione del nome del disco

Per effettuare la formattazione devi usare il programma presente sul disco di sistema. Inserisci quindi quest'ultimo nel drive e introduci la sua password cioe' spc.

Attenzione che il DOS fa differenza tra spc o SPC o Spc e cosi' via.

Digita ENTER e il prompt A' apparira' sulla sinistra in basso, il che sta ad indicare che e' attivo il drive A (l'interfaccia ne puo' collegare fino a 3 A,B,C).

Digita il comando CAT e sul video appariranno i programmi presenti sul disco.

Lancia il programma di formattazione con :

RUN "format" CODE

Apparira' subito il menu' riportato qui sotto, la tua scelta sara' quindi relativa al tipo di drive: (40 o 80 tracce singola o doppia densita')

Riproduzione video MENU'

Una volta scelto il tipo di drive il DOS ti chiede su quale drive e' presente il disco da formattare. Digita il numero del drive se ne hai uno solo e da il via alla formattazione digitando F.

Una volta scelto il tipo di drive il DOS ti chiede su quale drive e' presente il disco da formattare. Digita A se ne hai uno solo e da il via alla formattazione digitando F.

A questo punto ti vengono chieste la password e il nome del disco (quest' ultimo non vincolante per la lettura). Inserisci una parola standard per i tuoi dischi.

IL DOS inizia la mappatura delle tracce controllandone l'integrita' e dando informazione sul video delle operazioni che sta eseguendo:

ESEMPIO segnalazioni video (reading- format etc)

In caso di errore la traccia interessata verra' segnalata, per poi proseguire con le altre. Se non vi sono stati errori il disco e' pronto per essere usato.

Altrimenti ripeti la formattazione in quanto, a volte, causa di errori possono essere dei granelli di polvere che verranno rimossi eseguendola piu' volte.

Ricordati che formattare un disco contenente dei programmi significa CANCELLARLI.

Per tornare al DOS digita

Q= quit del programma format

Per tornare al BASIC

RETURN+ENTER+ENTER

e il cursore K dello Spectrum apparira' in basso a sinistra.

Prima di concludere questo capitolo qualche consiglio per l'uso dei dischetti (solo 5"1/4)

- non piegarli
- non toccare con le dita la parte visibile dalla finestra di lettura per la testina
- non scrivere sull'etichetta di identificazione con matite o penne a sfera; usare penne a feltro
- non effettuare cancellature con gomme
- non posizionare il disco vicino a bobine (telefono, trasf. televisori ecc) anche per dischi da 3"
- Fare attenzione all' elettricita' statica che puoi accumulare su moquette (anche per dischi da 3")

La Directory

Per poter sapere cosa contiene un dischetto basta semplicemente conoscere la sua PASSWORD ed abilitare il comando CAT seguito da `enter` (prova ad usare questo comando sul disco di sistema).

Dopo aver fatto cio' sul video ti apparira' quanto segue:

```

Title: system
16 File(s)
0 Del. File(s)

A: format      <C> 10: dbdos      <C> 1
A: demo        <B> 30: SCREEN0   <C> 1
A: SCREEN1     <C> 27: SCREEN2   <C> 1
A: SCREEN3     <C> 27: SCREEN4   <C> 1
A: SCREEN5     <C> 27: SCREEN6   <C> 1
A: backup     <C> 10: backup    <B> 1
A: copy       <C> 14: copy     <B> 1
A: writer     <B> 27: writer c  <C> 4
449 Free.
    
```

Le prime tre righe della directory indicano che il titolo del disco e' "system", che vi sono 16 file salvati e zero deletati e la lettera che si trova tra i simboli di maggiore e minore ci segnala il "FILE TYPE".

Per i programmi scritti in linguaggio macchina la lettera sara' `C`, per quelli basic `B` e per i file dati sara' `D`.

Infine il numero che segue tale lettera indica la sua lunghezze in numero di settori (settore = 256 bytes).

CAPITOLO 4

Questo capitolo si occupa di come trasformare i tuoi programmi che utilizzano il registratore.

A questo punto e' utile ricordare la sintassi richiesta per abilitare un comando DOS da un programma in BASIC :

```
RANDOMIZE USR 15363 : REM : COMANDO DOS
```

Dopo la REM ci deve essere un solo unico comando DOS.

Non devi confondere il valore 15363 con 15300 perche' nel secondo caso si passerebbe al DOS con l'apparizione del cursore A' senza l'esecuzione del comando che segue la REM.

Adattando il tuo vecchio programma, puoi incontrare diverse situazioni :

a) PROBLEMA DELL' AUTOSTART

Come sai, caricando un programma (BASIC o Linguaggio Macchina) da disco con il comando DOS 'RUN', esso va direttamente in esecuzione.

Consideriamo, a questo punto, i due casi separatamente.

Programma BASIC

La differenza sostanziale e' che col DOS non possiamo far partire il programma da una linea qualsiasi.

Il DOS, infatti, manda in esecuzione il programma BASIC sempre dalla prima linea.

Quindi, nel caso che il nostro vecchio programma parta da una linea diversa, bastera' inserire all'inizio una linea GOTO all'effettivo numero della linea di partenza.

Esempio: se il programma partiva dalla linea 3000 e' sufficiente inserire la linea : 1 GOTO 3000

```
.....  
.....  
3000 REM inizio programma
```

In caso di stesura di un nuovo programma, devi ricordarti di mettere sempre nelle prime linee tutte quelle operazioni che servono a lanciare il programma :

- caricamento dati
- caricamento routine in linguaggio macchina
- inizializzazione delle variabili ecc.

Programma in L.M.

Qui la cosa e' piu' semplice in quanto il DOS nel comando SAVE ci permette di aggiungere rispetto al comando standard un terzo valore.

Questo rappresenta la locazione da cui iniziera' l'esecuzione del nostro programma in LM.

Esempio : SAVE "scroll" CODE 64000,1000,645000

In questo esempio salviamo il codice macchina denominato scroll la cui lunghezza e di 1000 bytes, a partire dall'

indirizzo 64000 e fissiamo il suo autostart dalla locazione 64500.

Ricordate che omettendo il valore della locazione di start e caricando ugualmente il programma con RUN si puo' avere il blocco del sistema.

Altri comandi che dovranno essere sicuramente modificati sono S A V E e L O A D.

Il DOS si comporta in generale come il sistema operativo dello Spectrum e i seguenti sono gli unici accorgimenti da ricordare:

1 - Operando col disco e' indispensabile il nome del programma da salvare o caricare.

2 - Non si possono salvare dati o programmi con lo stesso nome.

3 - Se esegui da BASIC un caricamento di dati o programmi che non sono presenti sul disco, il programma saltera' all'istruzione successiva senza nessuna segnalazione.

Se eseguito direttamente lo stesso comando dara' il messaggio "NO FILE (s)".

4 - Salvando delle zone di memoria con il CODE, il nome deve essere inserito direttamente nel comando e non tramite una variabile:

```
SAVE n$ code 30000,3450 e' ERRATO
```

Se fosse indispensabile, e' possibile aggirare il problema pokando direttamente i codici dei caratteri del nome, nella locazione di memoria dopo il comando SAVE preceduto da REM; questa locazione la si puo' trovare scandendo l'area BASIC fino ad incontrare il codice della REM; il primo carattere andra' inserito nella locazione REM +4 visto che dopo la REM abbiamo i ":" (duepunti), il SAVE e le virgolette.

ES:

```
locazione REM =34000
```

```
locazione d'inizio pokatura caratteri = 34004
```

5 - Ricordati che nella linea BASIC dove e' presente

il comando DOS ci possono essere altre istruzioni prima, ma NON dopo.

6 - Dopo un NEW bisogna inserire di nuovo la password.

Ecco in riassunto le operazioni da fare per implementare sul disco un programma registrato su nastro:

- Tornare al BASIC come spiegato al cap. 2
- Caricare da nastro il programma che ci interessa
- Fare i controlli descritti ed effettuare le opportune modifiche
- Tornare al DOS con RANDOMIZE USR 15300
- Salvare il programma con il comando SAVE
- In caso di codici macchina fare attenzione ai CLEAR e alla locazione di autostart.

CAPITOLO 5

RICERCA RANDOM

Prima di entrare nei particolari di questa nuova possibilita' del tuo Spectrum e' utile vedere in generale, cosa e' la ricerca random (casuale) e cosa puo' fare.

Iniziamo ricordando come e' organizzato il disco.

Esso e' diviso in settori che nel caso di un drive densita' normale a 40 tracce sono 390.

Ognuno di questi settori possono contenere 256 bytes, questo e' importante perche' sia che noi scriviamo in un settore 5 o 50 bytes il DOS ne considera sempre 256 e vedremo poi come sfruttarli bene.

Premesso questo, la ricerca random e' la possibilita' di scrivere e rileggere casualmente in uno dei 390 settori proprio come facciamo con le celle della memoria RAM.

E' evidente quindi la potenza del sistema FDD2, che permette di usare il disco come una memoria virtuale su cui memorizzare, durante il funzionamento di un programma, dati che invece dovrebbero essere altrimenti immagazzinati in RAM.

La velocita' d'accesso e' tale che il programma non ne risentira', e la memoria RAM dello spectrum rimane interamente disponibile per il programma e tutte quelle routine che serviranno ad esso.

Consideriamo ora l'organizzazione dei dati su disco.

La procedura e' quella di creare degli insiemi di piu' settori che d'ora in avanti chiameremo FILE.

Ogni settore, che come detto contiene 256 bytes, rappresenta un sotto insieme di dati che puo' avere una sua organizzazione interna.

I settori li chiameremo RECORD. Per cui ricapitolando:

Grande insieme di settori = 'FILE
Settori(256bytes) = 'RECORD

Questa struttura adeguatamente organizzata puo' operare su qualsiasi tipo di dato. Facciamo un esempio:

Vogliamo realizzare un archivio che dovra' contenere i titoli dei libri della nostra biblioteca.

Quest'archivio dovra' ospitare 100 titoli e per ogni titolo contenere Autore, Casa ed., Argomenti trattati e tutte quelle informazioni che riteniamo utili.

Organizzeremo quindi un FILE di 100 RECORD (settori) che chiameremo "libri". Ogni record conterra' le informazioni prima citate. Se queste non raggiungono i 256 Bytes possiamo raggruppare piu' titoli in un record.

Teniamo a mente questo esempio perche' piu' tardi lo realizzeremo.

Entriamo adesso nei particolari della tecnica da usare col nostro drive per utilizzare la ricerca random, vedendo quali sono gli accorgimenti per una efficiente programmazione. I comandi DOS rispettivamente per leggere e scrivere sono:

```
PEEK "NOME DEL FILE" CODE INDIRIZZO BUFFER,N.SETTORE
```

```
POKE "NOME DEL FILE" CODE INDIRIZZO BUFFER,N.SETTORE
```

Ricordati di farli precedere da RANDOMIZE USR 15363:REM: se inseriti in programmi BASIC.

Analizziamoli:

NOME DEL FILE

Come premesso, una volta decisa la quantita' dei dati dobbiamo creare materialmente il file che li contenga. Per il nostro file "libri" dobbiamo salvare su disco una zona di memoria di 100 settori ,quindi $100 \times 256 = 25600$ byte con il nome "libri":

```
SAVE "libri" CODE 30000,25600
```

Ora sul nostro disco e' pronto il file "libri" per ricevere i nostri dati.

Come regola generale possiamo dire che per creare un file dobbiamo salvare una zona di memoria di uguale grandezza. Il piu' grande file realizzabile sara' quindi di 65535 bytes .

In caso di necessita' maggiore e' tuttavia possibile salvare due file con nome diverso cui accedere da programma.

INDIRIZZO BUFFER *vedi procedimento*

Il DOS della nostra interfaccia ha bisogno per la gestione casuale dei dati di un buffer ovvero di una zona di memoria di 256 bytes dove inserire i dati prima di passarli su disco, o dove caricarli in fase di lettura.

Il dato che inseriamo nel comando e' l'inizio di questa area di memoria. Teoricamente il buffer puo' essere in qualsiasi zona della memoria, ma dato che su questi 256 bytes dovremo effettuare delle operazioni, ci sara' una zona particolarmente pratica da usare.

NUMERO DI SETTORE

Corrisponde al settore usato per memorizzare il record o in fase di lettura, il settore contenente il record da leggere. Vediamo gli esempi:

POKE "LIBRI" CODE 30000,7
salva su disco nel settore 7 i 256 bytes contenuti in RAM
a partire dalla locazione 30000 .

PEEK "libri" CODE 30000,7
legge dal file "libri" il record n.7 e lo immagazzina in
RAM a Partire dalla locazione 30000.
Quindi i nostri dati saranno presenti da 30000 a 30256.

Chiariamo ora alcuni punti che sono importanti per creare
programmi funzionanti.

Iniziamo dalla funzione del buffer.

E' importante considerare che i dati che noi dobbiamo
immagazzinare sono trattati dal computer in modo diverso .

Infatti se dobbiamo immagazzinare dei caratteri, ogni byte
conterra' un codice ASCII ;quindi un record potra'contenere al
massimo 256 caratteri.

Usando numeri interi, il nostro record ne conterra'256 se sono
numeri che possono essere rappresentati con un byte (da 0 a 255)
oppure 128 se usiamo numeri da 2 bytes (32728 a 65535) o meno
ancora se usiamo numeri decimali che richiedono 5 byte per la
rappresentazione in virgola mobile.

Ne consegue che i dati che il computer e il disco si scambiano,
devono essere sottoposti in questo modo ad una codifica relativa
al tipo di dato.

IN pratica nei nostri primi programmi useremo una semplice
tecnica che lascia tutto il lavoro prima descritto nelle mani
del DOS e delle funzioni del BASIC Sinclair.

Il metodo consiste nel definire all'inizio e esclusivamente
all' INIZIO del programma (cio' e' fondamentale) una variabile
di stringa lunga 256 spazi.

Questa stringa verra' poi riempita con i dati di qualsiasi
tipo, numeri e non, su cui il nostro programma dovra' lavorare .

Se saranno numeri come ben sappiamo possiamo estrarre il valore
della stringa k\$="1234" con la funzion VAL:

VAL K\$=1234.

Una volta definita la stringa useremo come indirizzo del buffer
l'indirizzo del'inizio area variabili + 3.

Il tre dipende dal modo con cui lo Spectrum memorizza le
variabili di stringa.

Così facendo i dati da e verso il computer verranno sempre
inseriti all'interno della stringa e quindi manipolati con tutte
le funzioni residenti nel BASIC. Vediamo un esempio:

- Definisco la stringa J\$="256 spazi"
- L'indirizzo area variabili e':
ADR= peek 23627+256*peek 23628
- Inserisco nella stringa i dati da memorizzare

- Salvo la stringa J\$con:

```
RANDOMIZE USR 15363:REM:POKE CODE "Nome FILE" ADR+3,1
```

In questo modo ho salvato su disco nel primo record del file la stringa j\$ con i miei dati.

Viceversa in fase di lettura:

```
- RANDOMIZE USR 15363:REM: PEEK CODE" NOME FILE" ADR+3,1
```

carico la stringa j\$ con i 256 bytes del primo record (ricorda record =settore) che saranno disponibili come elementi della stringa.

E' importante ricordarsi di azzerare la stringa dopo averla salvata.

Per far questo NON si deve fare una assegnazione diretta del tipo : LET j\$=" 256 spazi", ma usare LET j\$(1 TO 256)= "256 spazi".

Questo vale per qualsiasi assegnazione dobbiamo fare alla stringa j\$.

Se infatti ridefiniamo direttamente la variabile usata come buffer, la sua posizione nell'area variabili mutera' e il nostro metodo non funzionera' piu'.

Se trattiamo invece dati numerici, dobbiamo operare in modo diverso.

Durante la stesura del programma l' area variabili si sposta ogni volta che si aggiunge una linea ; per questo motivo conviene aggiornare ADR ogni volta che si legge o si scrive:

```
LET ADR=PEEK(23627+256*23628)+3:RANDOMIZE USR 15363:REM:POKE  
"NOME FILE"ADR,N. SETTORE
```

ORGANIZZAZIONE DEI DATI

Riprendiamo ora l' esempio del'archivio dei libri.Si era stabilito che ogni record doveva contenere:

TITOLO DEL LIBRO

AUTORE

CASA EDITRICE

PRINCIPALI ARGOMENTI TRATTATI

Ognuna di queste specifiche prendono il nome di campi del record.

Il nostro archivio cosi' strutturato avra' quindi 4 campi.Definito questo dobbiamo stabilire quanti caratteri assegnare per ogni campo:

ad esempio	-TITOLO	40	CARATTERI
	-AUTORE	40	" "
	-CASA EDITRICE	70	" "
	-ARGONENTI TRATTATI	100	" "
	TOTALE	250	CARATTERI

Definiamo a questo punto l'ordine dei campi all'interno della stringa j\$ che presumiamo già definita nella linea 1 del nostro programma.

- J\$(1 TO 40) = TITOLO DEL LIBRO
- J\$(41 TO 80) = AUTORE
- J\$(81 TO 150) = CASA EDITRICE
- J\$(152 TO 250) = ARGOMENTI TRATTATI

Abbiamo così strutturato il nostro record. Ora che abbiamo chiaro anche il concetto di campo, vediamo come azzerare la nostra stringa nel caso in cui debba essere usata la funzione VAL.

Tutto è molto semplice: dobbiamo solo inserire all'inizio di ogni campo uno zero, poiché la funzione VAL non tratta le stringhe vuote.

Dunque se i nostri dati fossero numeri su cui eseguire operazioni, la nostra stringa così definita andrebbe azzerata con una maschera simile alla seguente:

```
LET J$(1 TO 256)="0.....0.....0.....0.....256CAR"  
                cam.1   cam.2       cam.3       cam.4
```

Così facendo *la stringa è azzerata*, in caso di uso di VAL su di un campo vuoto il risultato sarà zero, *la posizione di j\$ nell'area variabili non è cambiata.

Un'ultima cosa prima di lasciarvi al listato del programma che realizza l'archivio libri.

È necessario in un programma di gestione di un file memorizzare alcuni parametri circa il file stesso, parametri che devono essere letti e aggiornati ogni volta che il file è usato.

Questi dati possono rappresentare:

- risultati di operazioni sui record
- spazio rimasto
- numero di rekord scritti (forse il più importante)

Quest'ultimo ci evita infatti di scrivere su record già utilizzati oppure di perdere tempo leggendo in fase di ricerca record vuoti.

Questi parametri saranno evidentemente memorizzati sul file stesso in un record fisso, in genere il primo o l'ultimo.

All'inizializzazione di un nuovo file conviene azzerare con l'opportuna maschera il record dei parametri.

Infatti appena creato su disco, il nostro file può contenere valori importanti della memoria che potrebbero causare errori.

In ogni caso è meglio controllare se il record sia accidentalmente vuoto all'inizializzazione del file (vedere il listato del prog).

Va precisato inoltre che nel caso i parametri siano molti, (totali, valori parziali ecc) e visto che la stringa buffer va azzerata ogni volta, dobbiamo preparare una seconda stringa che verra' continuamente aggiornata ma non salvata.

Cio' avverra' solo alla fine delle operazioni scambiando il contenuto con la variabile buffer, esempio:

-stringa buffer j\$

-stringa dei parametri p\$

-alla fine delle operazioni sul file eseguo:

LET j\$(1 to 256) =p\$ (1 to 256) e salvo con il soliti comandi

Un ultima cosa prima di iniziare a introdurre il listato del programma ricordatevi di creare il file "libri" nel seguente modo:

SAVE "libri" CODE 30000,25600

B U O N L A V O R O !!!

B A C K U P

Questo programma e' sicuramente il piu' importante tra i vari programmi di utilita' che ti abbiamo fornito. Infatti esso ti permette di effettuare la copia "completa" di un qualsiasi disco, a patto naturalmente che tu sia a conoscenza della sua password.

Assumiamo per default che la password del dischetto sia sandy e di voler quindi duplicare quest'ultimo per avere una copia di sicurezza; dovrai procedere in questo modo:

inserisci il disco di sistema e digita:

RUN "backup" seguito da (enter)

ti apparira' quanto segue:

* BACKUP 1.0 *

~ 1984 SANDY COMPUTER PROD

PASSWORD FOR SOURCE DISK:

digita "sandy" seguito da (enter), ed a questo punto il programma ti guidera':

INSERT SOURCE DISK THEN PRESS Y

Y

INSERT DESTINATION DISK

THEN PRESS Y

Y

.....

Da questo punto in avanti il programma continuera' a chiederti di inserire prima il disco originale e dopo la copia (alternativamente) e come conferma di battere "Y".

Quando l'operazione di backup sara' terminata vedrai ricomparire il carattere di prompt del sistema operativo; nel caso in cui si riscontrasse qualche errore durante questa operazione ti verra' segnalato, e dovrai quindi ripetere tutta l'operazione.

Si e' costretti ad effettuare l'operazione di backup in piu' fasi poiche' l'area di RAM disponibile per i dati sullo Spectrum e di soli 24 Kbytes.

C O P Y

Questo programma ti permette di duplicare un singolo file alla volta, caricandolo da un disk drive (sorgente) e riscrivendolo su di un altro disk drive (destinazione). Inoltre, come potrai vedere, avrai a disposizione alcuni comandi del sistema operativo che ti permetteranno, ad esempio, di avere il CAtalogo del disco senza pero' uscire dal programma stesso.

Infatti noterai anche che il carattere di prompt del sistema operativo, una volta caricato il programma, non sara' piu' il simbolo " ", ma bensì "*" (A*).

Per caricare il programma:

```
RUN "copy"          seguito da (enter)
```

Ti apparira':

```
          * COPY 1.0 *  
1984 SANDY COMPUTER  PROD
```

```
COMMANDS ARE:
```

```
CAT  
NEW "newfile","oldfile"  
ERASE "filename"  
MOVE  
COPY "newfile","oldfile"  
RETURN
```

```
A*
```

Esempio n.1

Se vuoi copiare un file di nome "pippo" che si trova sul disk drive A, sul drive B bastera' digitare:

```
COPY "b:pluto","a:pippo"
```

Esempio n.2

Se invece vuoi copiare tutti i files contenuti sul disk drive A sul disk drive b, la sintassi e' questa:

```
COPY "b:*","a:*"
```

N.B.=Ricordati di specificare sempre il tipo file!

```
COPY "b:pluto","pippo"CODE
```

SOMMARIO DEI COMANDI D.O.S.
DEL NUOVO SISTEMA OPERATIVO

erase deleta un file su disco

move riorganizza le locazioni del file
 su disco dopo una erase

format f per formattare un nuovo disco
 v per verificare la qualita' delle tracce

load per caricare un programma da disco

new per cambiare il nome di un file su disco
 es.: new "newname", "oldname"

randomize
USR 15300 per ritornare nel D.O.S. dal BASIC

run carica un programma e ne da lo start
 es.: run "name"

save scrive un programma su disco

usr cambia la password del disco

cat illustra il catalogo del disco

*"A:" seleziona il drive A
*"B:" seleziona il drive B
*"C:" seleziona il drive C

peek per leggere un record di un file

poke per scrivere in un record di un file

merge per combinare due o più programmi in
 memoria, presenti sul disco

return da D.O.S. per ritornare al Basic Sinclair

SIRIO

IL COMANDO LOAD

Questo comando dato mentre ci si trova nel sistema operativo ti permettera' di caricare i programmi precedentemente salvati su disco, senza pero' dare l'autostart.

Se non avverranno errori durante il caricamento, al termine di questa operazione tornera' automaticamente al BASIC dello Spectrum ed il programma sara' pronto per essere listato.

La sintassi e' molto simile agli altri comandi:

- LOAD "nome file" seguito da (enter)

in questo caso verra' caricato il programma "nome file";

- LOAD "nome file" CODE seguito da (enter)

in questo caso, invece, verra' sempre caricato il programma "nome file" scritto pero' in codice macchina;

- LOAD "nome file"CODE 30000 seguito da (enter)

infine, in quest'ultimo caso, verra' caricato a partire dall'indirizzo RAM 30000.

N.B.= Trovandoti in BASIC per ritornare in ambiente DOS digita
RANDOMIZE USR 15300 seguito da (enter).

IL COMANDO MERGE

Questo comando viene usato per combinare due o piu' files di programmi in memoria, ottenendo cosi' un unico grande programma.
Per esempio:

- MERGE "old" per inserire il programma "old" con quello presente in memoria RAM

- MERGE "b:new" per inserire il programma "new" con quello presente in memoria RAM, che si trova sul disk drive B.

IL COMANDO DI START

Questo comando e' molto semplice da usarsi, quindi passiamo subito a vederne la sintassi:

RUN "nome del file" per i programmi BASIC

RUN "nome del file" CODE per i programmi in L.M.

Nel primo caso verra' caricato da disco il programma ed in modo automatico verra' dato lo start; cosi' anche nel secondo caso, ma per un programma scritto in codice macchina.

Infatti se non specificheremo il tipo del file verra' segnalato dal sistema:

"NO FILE (s)"

IL COMANDO RANDOMIZE

Quando ti trovi in BASIC Sinclair, usa questo comando per tornare al sistema operativo:

RANDOMIZE USR 15300 seguito da (enter).

IL COMANDO NEW

Anche questo comando e' molto semplice e non essendoci particolarita' da rilevare, possiamo passare subito a vederne la sintassi:

NEW "nuovo nome", "vecchio nome" seguito da (enter)

NEW "nuovo nome", "vecchio nome" CODE seguito da (enter)

Come avrai sicuramente capito, questa istruzione serve a cambiare il nome di un file gia' esistente su disco con un nuovo nome; cosi' nell'esempio precedente il file che si chiamava "vecchio nome", ora si chiamera' "nuovo nome". Ricordati di specificare il tipo di file.

IL COMANDO SAVE

Una volta passato al BASIC Sinclair e dopo aver scritto il tuo programma ti trovi ora nella condizione di voler salvare questo ultimo su disco; se però dai questo comando mentre ti trovi in ambiente basic verra' abilitata l'uscita del registratore, quindi per far si che cio' non avvenga devi operare nel modo seguente:

- RANDOMIZE USR 15300 seguito da (enter)

ed in questo modo ti apparira' il carattere di prompt A per confermarti che ti trovi in ambiente DOS;

- SAVE "nome file" seguito da (enter)
 per un programma
 basic

nel caso però il tuo programma sia scritto in codice macchina dovrai definire tre parametri extra; vediamo quali:

- SAVE "nome file" CODE address, length, (autorun address)

dove:

- . "address" e' l'indirizzo di inizio
 scrittura del programma
- . "length" e' la lunghezza in bytes
- . "autorun address" e' l'indirizzo di
 autostart (opzionale).

Nel caso in cui si debbano registrare delle variabili o delle variabili stringa, bisogna invece seguire questo formato:

- SAVE "nome file" DATA a()
- SAVE "nome file" DATA a\$()

IL COMANDO PEEK

Questo comando permette la lettura di un file in modo "random", ma puo' essere letto un solo record alla volta ed inoltre viene richiesto un buffer di RAM per ogni accesso. Va ricordato che questo tipo di accesso opera solo su file di tipo CODE.

Per esempio:

- PEEK "nome file" CODE 30000,1

legge il primo settore del file "nome file" e lo trasferisce in RAM dall'indirizzo 30000 per 256 bytes;

- PEEK "b:nome file" CODE 30000,10

legge il decimo settore del file "nome file" che si trova sul disk drive B e lo trasferisce in RAM a partire dall'indirizzo 30000 per 256 bytes.

IL COMANDO POKE

Questo comando serve per la scrittura di un file random e, come per l'istruzione PEEK puo' essere scritto un solo settore (record) alla volta e deve essere definito un buffer di memoria RAM. La sintassi e' identica a PEEK:

- POKE "nome file" CODE 30000,10

salvera' nel decimo settore del file "nome file" il contenuto della memoria RAM a partire dall'indirizzo 30000 per 256 bytes.

IL COMANDO ERASE

Questo comando ti permettera' di cancellare un file che non desideri sia piu' presente sul tuo dischetto; bisognera' pero' specificare il tipo di file; vediamo un esempio:

- ERASE "nome file" per programmi BASIC
- ERASE "nome file" CODE per programmi L.M.
- ERASE "nome file" DATA per files di dati

N.B.= Una volta cancellato un file disco se vuoi recuperare lo spazio che quest'ultimo occupava devi utilizzare l'istruzione MOVE.

IL COMANDO MOVE

Una volta che e' stato cancellato un file dal dischetto rimane un cosiddetto "buco" che non sarebbe piu' possibile utilizzare se non avessimo a disposizione questa istruzione; infatti essa "muove" tutti i files contenuti nel dischetto, compattandoli.

In questo modo l'area occupata precedentemente dal file che e' stato deletato viene occupata anch'essa e lo spazio cosi' recuperato.

Come si deve effettuare questa operazione! Vediamola.
Dopo aver cancellato un file dal dischetto digitate:

- MOVE seguito da (enter)

in questo modo dopo un certo tempo (dipendente dal numero di files contenuti sul dischetto) ti riapparira' il carattere di prompt A del sistema operativo.

N.B.= Questa operazione richiede che vi sia a disposizione un minimo di 4 Kbytes di memoria RAM per operare. Se cosi' non fosse ti apparira' il messaggio:

* ERROR *

COME CAMBIARE LA PASSWORD

Tutti starete pensando che allora questa password non e' una vera protezione, visto che e' possibile cambiarla.

Tranquillizzatevi, perche' per poter modificare la password al disco bisogna prima conoscerne la vecchia e la procedura per poter effettuare tale operazione e' la seguente:

- . assicuratevi di aver gia' battuto la password attuale del disco e di trovarvi in ambiente DOS;
- . abilitate il comando USER e battete (enter);
- . ora il computer vi chiederà di digitare la nuova password che volete attribuire al disco; scrivetelo e battete (enter).

Supponendo che abbiate digitato "pippo", la nuova parola chiave del vostro dischetto sarà proprio pippo.

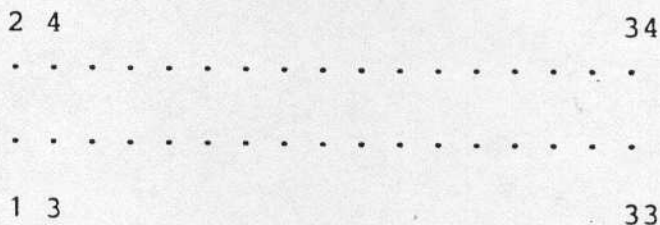
N.B.= Cercate di usare una password standard per i vostri dischetti, poiche' se dimenticaste il nome non potreste piu' accedere al disco in questione.

CONNESSIONI DEL CONNETTORE
SULL' INTERFACCIA

Le connessioni dei pin (vedi il diagramma sotto riportato) sono esattamente quelle dei bus standard SASI, ovvero SHUGART compatibile.

Vogliamo meterti a conoscenza delle connessioni nel caso in cui tu abbia acquistato solo l'interfaccia e voglia utilizzare un disk drive che già possiedi.

0 v-- 1	2 --
0 v-- 3	4 --
0 v-- 5	6 -- disk drive d
0 v-- 7	8 -- index
0 v-- 9	10 -- disk drive a
0 v-- 11	12 -- disk drive b
0 v-- 13	14 -- disk drive c
0 v-- 15	16 -- head load (motor on)
0 v-- 17	18 -- direction
0 v-- 19	20 -- step
0 v-- 21	22 -- write data
0 v-- 23	24 -- write gate
0 v-- 25	26 -- track 00
0 v-- 27	28 -- write protected
0 v-- 29	30 -- read data
0 v-- 31	32 -- side 1
0 v-- 33	34 --



```
10 CLEAR : BORDER 0: PAPER 0: INK 7: CLS
15 REM _inizializzazione
16 REM _variabili
20 LET js=""
```

```
22 DIM s$(4,15): LET s$(1)="Titolo": LET s$(2)="Autore": LET s$(3)="Casa edit.
": LET s$(4)="Arg.Princ."
25 LET ca=1: LET kc=0
26 INPUT "vuoi azzerare il file?y/n"jz$
28 IF z$="y" THEN LET adr=(PEEK 23627+256*PEEK 23628)+3: RANDOMIZE USR 15363:
REM : POKE "libri"CODE adr,100
31 REM _lettura rec.parametri
32 LET adr=(PEEK 23627+256*PEEK 23628)+3: RANDOMIZE USR 15363: REM :PEEK "libr
i"CODE adr,100
34 REM _controlla se il file e$ _vuoto CHR$ 0 o CHR$ 32
35 IF j$(1)=CHR$ 0 OR j$(1)=" " THEN LET kc=1: GO TO 50
40 LET kc=VAL j$(1 TO 4)
50 CLS : PRINT AT 2,8;"_O P Z I O N I"
55 PRINT AT 6,8;"1) CREAZIONE FILE"
60 PRINT AT 8,8;"2) AGGIORNAMENTO
INSERIMENTO RECORD"
65 PRINT AT 11,8;"3) RICERCA del RECORD"
100 IF INKEY$="1" THEN GO TO 2000
105 IF INKEY$="2" THEN GO TO 4000
110 IF INKEY$(">"3" THEN GO TO 100
1000 CLS : PRINT AT 2,8;"_RICERCA"
1005 PRINT AT 8,8;"1)per Titolo"
1006 PRINT AT 10,8;"2)per Autore"
1007 PRINT AT 12,8;"3)per Casa ED."
1008 PRINT AT 14,8;"4)per Argomenti"
1009 PRINT AT 16,8;"5)Menu'": PAUSE 0
1010 LET t$=INKEY$
1013 IF t$="5" THEN GO TO 50
1015 IF t$="1" OR t$="2" OR t$="3" OR t$="4" THEN GO TO 1040
1020 GO TO 1010
1040 LET ca=1*(t$="1")+41*(t$="2")+81*(t$="3")+151*(t$="4")
1043 LET t=VAL t$
1045 INPUT "ricerca per "j(s$(t));" "j LINE t$
1049 GO SUB 8000
1090 PRINT AT 19,0;"Altra ricerca?y/n"
1100 LET z$=INKEY$
1105 IF z$="y" THEN GO TO 1000
1110 IF z$="n" THEN GO TO 50
1120 GO TO 1100
2000 LET kc=0: CLS : PRINT AT 2,8;"_INSERIMENTO RECORD"
2010 INPUT "Quanti rec. devi inserire?"irec
2020 FOR i=1 TO rec
2030 GO SUB 9998: CLS : INPUT "Titolo"j$(1 TO 40)
2035 INPUT "Autore"j$(41 TO 80)
2040 INPUT "Casa editrice"j$(81 TO 150)
2045 INPUT "Principali Arg. trattati"j$(151 TO 250)
2050 LET adr=(PEEK 23627+256*PEEK 23628)+3: RANDOMIZE USR 15363: REM : POKE "lib
ri"CODE adr,i
2054 REM _aggiorno rec. parametri
2055 LET kc=kc+1
2060 GO SUB 9998: LET j$(1 TO 4)=STR$ kc
2070 LET adr=(PEEK 23627+256*PEEK 23628)+3: RANDOMIZE USR 15363: REM : POKE "lib
ri"CODE adr,100
2080 CLS : PRINT AT 10,8;"_record inserito": PAUSE 50
2090 CLS : NEXT i
2100 GO TO 50
4000 CLS : PRINT AT 2,8;"_AGGIORNAMENTO FILE"
4003 PRINT #0;AT 0,0;"devi inserire un nuovo record y/n"i
4004 LET z$=INKEY$
4005 IF z$="y" THEN GO TO 5000
4006 IF z$="n" THEN GO TO 4010
```

```

4007 GO TO 4004
4010 PRINT AT 4,8;"Codici campi"
4015 PRINT AT 6,8;"1=> Titolo"
4020 PRINT AT 8,8;"2=> Autore"
4025 PRINT AT 10,8;"3=> Casa Editrice"
4030 PRINT AT 12,8;"4=> Argomenti principali"
4040 INPUT " Titolo del rec. da modificare "t$
4045 LET ca=1
4050 GO SUB 9998: GO SUB 8500
4055 INPUT "codice campo da modificare "i$cc
4060 INPUT (s$(cc));c$
4065 IF cc=1 THEN LET j$(1 TO 40)=c$
4070 IF cc=2 THEN LET j$(41 TO 80)=c$
4075 IF cc=3 THEN LET j$(81 TO 150)=c$
4080 IF cc=4 THEN LET j$(151 TO 250)=c$
4085 LET adr=(PEEK 23627+256*PEEK 23628)+3: RANDOMIZE USR 15363: REM : POKE "11b
r1"CODE adr,mod
4090 PRINT AT 21,0;"campo modificato "i$cc: PAUSE 80: PRINT AT 21,0;"
t=altro tit. n=altra mod. m=menu"
4095 LET z$=INKEY$
4100 IF z$="t" THEN GO TO 4000
4110 IF z$="n" THEN GO TO 4055
4115 IF z$="m" THEN GO TO 50
4120 GO TO 4095
5000 GO SUB 9998: CLS : INPUT "Titolo"j$(1 TO 40)
5010 INPUT "Casa editrice"j$(81 TO 150)
5045 INPUT "Pincipali Arg. trattati"j$(151 TO 250)
5050 INPUT "Autore"j$(41 TO 80)
5060 LET adr=(PEEK 23627+256*PEEK 23628)+3: RANDOMIZE USR 15363: REM : POKE "11b
r1"CODE adr,kc
5100 LET kc=kc+1: GO SUB 9998: LET j$(1 TO 4)=STR$ kc
5110 LET adr=(PEEK 23627+256*PEEK 23628)+3: RANDOMIZE USR 15363: REM : POKE "11b
r1"CODE adr,100
5200 GO TO 4090
6000 REM _STAMPA RECORD
6005 CLS : PRINT "RECORD numero "i$mod: PRINT
6010 PRINT PAPER 1; BRIGHT 1; s$(1); PAPER 0; BRIGHT 0; j$(1 TO 32)'
6015 PRINT ' PAPER 1; BRIGHT 1; s$(2); PAPER 0; BRIGHT 0; j$(41 TO 72)'
6020 PRINT ' PAPER 1; BRIGHT 1; s$(3); PAPER 0; BRIGHT 0; j$(81 TO 112)'
6025 PRINT ' PAPER 1; BRIGHT 1; s$(4); PAPER 0; BRIGHT 0; j$(151 TO 250)'
6030 PRINT AT 19,0; FLASH 1;"SPACE"; FLASH 0;" interrompe ricerca per:
j$(t)
6031 PRINT AT 2*(ca=1)+5*(ca=41)+8*(ca=81)+11*(ca=151),0; FLASH 1; s$(1*(ca=1)+2*(
ca=41)+3*(ca=81)+4*(ca=151))
6034 PRINT AT 21,0; FLASH 1;"ENTER"; FLASH 0;" per continuare"
6035 LET z$=INKEY$
6040 IF CODE INKEY$=32 THEN GO TO 1090
6045 IF z$="" THEN GO TO 6035
6100 RETURN
8000 REM _routine di ricerca
8010 FOR i=1 TO kc
8020 LET adr=(PEEK 23627+256*PEEK 23628)+3: RANDOMIZE USR 15363: REM :PEEK "11br
i"CODE adr,i
8030 IF t$=j$(ca TO ca+LEN t$-1) THEN LET mod=i: GO SUB 6000
8040 NEXT i: CLS : BEEP 0.1,16: PRINT AT 10,8;"_Fine ricerca": PAUSE 80: GO TO 5
0
8500 FOR i=1 TO kc
8510 LET adr=(PEEK 23627+256*PEEK 23628)+3: RANDOMIZE USR 15363: REM :PEEK "11br
i"CODE adr,i
8530 IF t$=j$(ca TO ca+LEN t$-1) THEN LET mod=i: RETURN
8540 NEXT i: CLS : BEEP 0.1,16: PRINT AT 10,8;"_Fine ricerca": PAUSE 80: GO TO 5
0
9998 LET j$(1 TO 256)="
9999 RETURN

```

