



## i trucchi del CP/M

di Pierluigi Panunzi

Occupandoci del CP/M, dalla prima puntata ad oggi, si è parlato molte volte della struttura di tale sistema operativo, facendo particolare menzione a quel modulo chiamato BDOS.

A partire da questo numero ci dedicheremo dunque all'approfondimento delle nostre conoscenze sul BDOS.

### Il modulo BDOS

Il B.D.O.S. (Basic Disk Operating System, Sistema Operativo di Base su Disco), è da pensarsi come il cuore del CP/M: è sempre lì, pronto ad eseguire tutte le funzioni "primitive" che un buon Sistema Operativo deve saper effettuare.

Il fatto che i progettisti del CP/M, quelli della Digital Research, abbiano previsto tutte le primitive, consente al programmatore in linguaggio macchina di creare programmi universali.

Spieghiamo cosa significa questa affermazione: con "universali" intendiamo programmi che possono girare "ad occhi chiusi" su una qualsiasi macchina che adotta il CP/M, ottenendo in tal modo una compatibilità del 100%!!

Tutto questo si ottiene, come unica condizione, fidandosi ciecamente (e non vediamo perché no!) del BDOS ed utilizzando esclusivamente le sue innumerevoli funzioni.

Queste funzioni si possono dividere in due blocchi logici fondamentali: le routine di I/O relative a byte (le prime 13) e le routine di I/O dei file su disco (le restanti 26).

Mentre le prime, come detto, ragionano in termini di "byte" singolo, le ultime permettono la gestione dei file in base a record di 128 byte.

Tutte queste routine, che vediamo elencate nella tabella 1 e delle quali ci occuperemo una per volta, hanno un unico punto di ingresso comune (il che ne facilita enormemente la gestione) per poi diramarsi in base al proprio numero d'ordine progressivo.

Vediamo subito come si effettua una generica chiamata:

si pone nel registro C il valore del numero progressivo di cui sopra e si chiama la subroutine posta a partire dall'indirizzo

0005. Si fa dunque così, semplicemente:

LD C, numero funzione

CALL 0005H

In realtà come vedremo, dovremo aggiungere ulteriori parametri in altrettanti registri: ad esempio se la routine dovrà manipolare un certo byte, tale valore dovrà essere posto in E, se invece si tratterà di un valore a 16 bit (in genere un indirizzo) allora dovremo porlo nella coppia di registri DE.

Viceversa in uscita da una routine avremo il seguente comportamento: se la routine fornisce come risposta un byte (ad esempio un carattere ottenuto dalla tastiera, un codice di errore, ecc), lo ritroveremo nel registro A, mentre se si tratterà di un valore a 16 bit, lo ritroveremo in HL.

Inoltre avremo che A e L conterranno (per convenzione) sempre lo stesso valore, come pure B e H.

Attenzione però che tutti gli altri registri dovranno essere eventualmente salvati prima della chiamata alla routine, in quanto (per la legge di Murphy) sicuramente un certo registro a noi utilissimo verrà usato anche dalla routine, che lo rifornirà a sua volta in uscita "bello che alterato!"

Comunque, facendo riferimento alla tabella 1 e per i dettagli alla tabella 2, andiamo ora a vedere una per una le singole funzioni indicando, laddove necessario, un esempio di applicazione.

Inoltre, a mano a mano che introdurremo concetti nuovi, apriremo le dovute parentesi, come pure accadrà nel caso in cui intenderemo soffermarci su un particolare aspetto del CP/M, anche se già trattato in altre puntate.

### Routine n. 0: System Reset

In questo caso non vi sono parametri da inviare, o da ricevere: tale routine come dice il suo nome, permette di resettare tutto il sistema e di "tornare al CP/M". Ma vediamo in dettaglio.

In particolare effettua un nuovo caricamento del modulo CCP, ricostruisce la zona di memoria contenente i vettori indicanti quali blocchi del disco sono allocati e quali no; posiziona, come buffer di interscambio con il disco la zona di memoria lunga 128 byte a partire da 0080H (che prende il nome di "DMA Address"); predispone tutti i dischi del sistema nello stato di lettura e/o scrittura (R/W Status) ed infine passa "lo scettro" al CCP il quale visualizzerà sullo schermo il famoso prompt "A".

Abbiamo parlato già parecchie volte del CCP (Console, Command Processor): ricordiamo qui che si tratta di un modulo di gestione nei comandi provenienti dalla

#### Funzioni del BDOS

Codice	Descrizione
0	Reset del sistema
1	Letture di un byte da Console
2	Scrittura di un byte sullo schermo
3	Letture di un byte dal dispositivo "Reader"
4	Scrittura di un byte verso il dispositivo "Puncher"
5	Scrittura di un byte verso il dispositivo "List"
6	Input/Output diretto da Console
7	Leggi l'IOBYTE
8	Scrivi l'IOBYTE
9	Invia alla Console una stringa terminante con "\$"
10	Leggi una stringa dalla Console
11	Test se c'è un tasto premuto
12	Leggi il numero di versione del CP/M
13	Reset del sistema a dischi
14	Selezione di un drive logico specificato
15	Apri un file per R/W
16	Chiudi un file dopo R/W
17	Cerca nella Directory la prima ricorrenza di un file
18	Cerca la prossima ricorrenza
19	Cancella un file
20	Leggi il "record" sequenzialmente
21	Scrivi il "record" sequenzialmente
22	Crea un nuovo file
23	Rinomina un file
24	Indica quali dischi sono attivi
25	Leggi il valore del disco di default
26	Scrivi il valore "DMA Address"
27	Leggi l'indirizzo di un "allocation vector"
28	Poni il disco in stato di Read Only
29	Indica i dischi in stato di Read Only
30	Poni il file in stato di "System" o di Read Only
31	Leggi l'indirizzo del DPB
32	Poni/leggi il valore dell'"User"
33	Leggi un "record" random
34	Scrivi un "record" random
35	Leggi l'ampiezza di un file
36	Scrivi il numero del record per il prossimo accesso
37	Resetta il drive specificato
40	Riempi il "record" random con zeri

Tabella 1

consolle, modulo che può anche essere ricoperto dal nostro programma, nel caso che avessimo bisogno di ulteriore spazio di memoria.

Ecco perché almeno al Reset del sistema deve essere ricaricato: se non fosse stato toccato, ancora meglio!

Si ha così la certezza di non incorrere in deleteri crash del sistema dovuti a byte alterati, garbage, ecc.

Infine questa routine di sistema è usata da tutti i programmi ad alto livello quando alla loro fine richiedono il ritorno al CP/M.

### Routine n. 1: Read Console Byte

Questa funzione prende il carattere proveniente dalla tastiera e lo depone nel registro A: se tale carattere ha un codice ASCII maggiore o uguale a 32 (20H) sarà visualizzato sullo schermo. Viceversa se sarà un carattere di controllo (da 00H a 1FH) si avrà una distinzione:

— il CARRIAGE RETURN (ODH, Control - M) e il LINE FEED (OAH, Control-J) porteranno il cursore a nuova linea e faranno eseguire il comando impostato.

— il BACK SPACE (08H, Control-H) arretrerà il cursore di una posizione.

— il TAB (09H, Control-I) farà avanzare il cursore fino alla prossima posizione multipla di 8. Tutti gli altri caratteri di controllo, verranno letti, ma non subiranno un'eco sul video; in particolare si ha che:

— il Control-S fermerà l'Output su video fino alla pressione di un tasto qualsiasi.

— il Control-P invierà verso la stampante l'Output, fino ad una nuova pressione del Control-P stesso.

Attenzione! La stampante deve essere "ON LINE" altrimenti il sistema si inchioda.

Inoltre, se non c'è alcun carattere in Input dalla tastiera, la routine ne aspetta uno fermando così l'elaborazione.

Riguardo a tale funzione c'è da dire che, per il fatto che pone sul video (salvo i casi visti) tutto quanto proviene da tastiera, non sarà utilizzabile tutte quelle volte in cui l'Output su video risulterebbe deleterio se non semplicemente indesiderato: ad esempio laddove alcuni caratteri di controllo di posizionamento del cursore provocherebbero effetti collaterali (cancellazioni accidentali, segni grafici, ecc.), oppure quando si richiede una risposta dell'operatore ad un certo Prompt.

In quest'ultimo caso la pressione di un tasto farebbe ripartire l'elaborazione, anche se per caso il tasto premuto è errato. Vedremo invece la funzione n. 9 (Read Console String), che permetterà all'operatore di usare tutti i caratteri di editing prima di inviare una certa stringa di comando, al limite anche di un solo carattere, dopo aver premuto il tasto RETURN.

Tabella 2

## Funzioni del BDOS in dettaglio

Valore di C	Nome	Input	Output
0	System Reset	—	—
1	Console Input	—	A = caratteri ASCII
2	Console Output	E = carattere ASCII	—

```

; istruzione DSK(n)
;
;
; scan del testo per input dell'argomento
6000 CB6920 CALL 2069H
6003 E60F AND 0FH
; selezione del disco desiderato
6005 5F LD E,A
6006 0E0E LD C,0EH
6008 CD0500 CALL BDOS
; calcolo dell'indirizzo
; del DPS (Disk Parameter Block)
600B 0E1F LD C,1FH
600D CD0500 CALL BDOS
6010 110500 LD DE,0005H
6013 19 ADD HL,DE
6014 5E LD E,(HL)
6015 23 INC HL
6016 56 LD D,(HL)
; DE ← numero massimo di blocchi
6017 13 INC DE
; calcolo dell'indirizzo del
; vettore di "allocazione"
601B 0E1B LD C,1BH
601A CD0500 CALL BDOS
601D DD210000 LD IX,0000H
6021 060B ALFA: LD B,0BH
6023 7E LD A,(HL)
6024 F5 PUSH AF
6025 F1 BETA: POP AF
6026 07 RLCA
; se C=1 il blocco e' occupato
6027 3B02 JR C,GAMMA
; se C=0 il blocco e' libero
; incremento del contatore
6029 DD23 INC IX
602B F5 GAMMA: PUSH AF
602C 1B DEC DE
602D 7A LD A,D
602E B3 OR E
602F 2B06 JR Z,DELTA
6031 10F2 DJNZ BETA
6033 23 INC HL
6034 F1 POP AF
6035 18EA JR ALFA
6037 F1 DELTA: POP AF
6038 DBE5 PUSH IX
603A E1 POP HL
; il valore del contatore viene
; passato al programma chiamante
603B C3CD29 JP 29CDH
END

```

Nella rubrica "I trucchi del CP/M" del mese di novembre (n. 35) è stata omessa la pubblicazione di questo listato. Ci scusiamo con i lettori.

## Routine n. 2: Write Console Byte

È questa la prima routine che necessita di ulteriore parametro all'atto della chiamata. È proprio il carattere che vogliamo inviare in Output su video o su stampante.

In particolare la chiamata sarà:

```
LD C,2
LD E,carattere
CALL 0005H
```

dove il carattere deve essere codificato in ASCII.

Come nel caso della routine precedente un carattere TAB in Input verrà visualizzato con otto spazi mentre in ogni caso la routine andrà a vedere se da tastiera sta per arrivare un carattere: se è un Control-S, l'Output verrà bloccato come abbiamo visto prima; se invece è Control-P, l'output verrà inviato verso la stampante.

Entrambi i procedimenti possono essere ripetuti più volte il primo per fermare e poi riprendere la visualizzazione, il secondo per stampare solo alcune parti di un certo testo, posto tra coppie di Control-P. Attenzione che alla fine il numero di Control-P sia pari, altrimenti non si ha il ritorno corretto al CP/M e tutti i caratteri digitati da tastiera verrebbero immediatamente scritti pure dalla stampante.

In questo caso riportiamo due esempi di utilizzazione della routine in esame, per la

stampa di una generica stringa terminante con un carattere nullo e per un altro tipo di routine analoga, ma più flessibile e "strana".

Il primo esempio è il seguente:

```
OUTPUT LD A, (HL)
        OR A
        RET Z
        INC HL
        PUSH HL
        LD E,A
        LD C,2
        CALL 0005H
        POP HL
        JR OUTPUT
```

e la sua chiamata assume il seguente aspetto:

```
LD HL,MESSAG
CALL OUTPUT
```

dove MESSAG è l'indirizzo iniziale della stringa di caratteri ASCII da inviare e terminante con uno zero, ad esempio:

```
MESSAG DEFB "MCmicrocomputer", 0
```

La seconda routine invece è più simpatica in quanto permette di avere il messaggio subito "sotto" alla chiamata stessa ed evita di caricare HL con l'indirizzo della stringa: evita anche la presenza dell'indirizzo stesso, del quale ora non c'è più bisogno!

La routine è questa:

```
MOSTRA POP HL
```

```
LOOP LD A,(HL)
      INC HL
      OR A
      JR NZ,OLTRE
      JP (HL)
      PUSH HL
      LD E,A
      LD C,2
      CALL 0005H
      JR LOOP
```

in questo caso la chiamata sarà:

```
CALL MOSTRA
DEFB 'MCmicrocomputer', 0
```

Spendiamo, prima di concludere, due parole su questa routine: il POP HL che si trova all'inizio è molto importante.

In particolare, all'atto della chiamata alla subroutine, il Program Counter che viene salvato nello Stack, è (guarda caso!!) proprio l'indirizzo iniziale della stringa da visualizzare.

Quando, ad un certo punto, all'interno del LOOP troviamo che l'accumulatore è nullo, e il valore di HL (abilmente già incrementato) ci darà proprio l'indirizzo a cui saltare, come se la subroutine terminasse con lo "0" di fine stringa.

Attenzione! Una "RET" a questo punto darebbe dei risultati a dir poco disastrosi!

A risentirci la prossima puntata con altre funzioni del BDOS. **MC**



ELETRONICA VALDARNESE s.d.f. Via Marconi 9/A-Loc.Muraccio  
52025 MONTEVARCHI (AR) tel. 055/980242-982513 C/c postale N.10418523

### ESTRATTO DEL CATALOGO GENERALE

PER COMMODORE 64

GESTIONALI

CONT.GENERALE(D) 180.000

Fatturazione(D) 120.000

Magazzino(D) 120.000

Gestione negozi(D) 150.000

Cartella clinica(D) 150.000

Mailing list(D) 60.000

Agenda telef.(D) 60.000

ARREDOGRAPH 195.000

WORD/PROC.

Easy script(D/N) 70.000

Vizawrite (D) 75.000

Word Pro.III (D) 75.000

UTILITY/VARIE

Compilatore DTL(N)\* 40.000

Austro Compiler(D) 70.000

Pet Speed (D) 70.000

PASCAL OXFORD(D)\* 150.000

Assembler (N) 35.000

" (D)\* 60.000

" (C) 70.000

KPM Pascal (D) 80.000

Supermoon (N) 30.000

" (C) 60.000

Turbo tape (N) 28.000

Fast copy (D) 50.000

Unguard (D) 120.000

The Clone (D) 80.000

Disk Doctor (D) 50.000

80 colonne (D/N) 40.000

64 Diagnostics (C,N) 40.000

Master (D) 110.000

Tool (D) 70.000

The Manager (D)\* 120.000

Calc Result Easy(C) 95.000

" Exp(C+D) 160.000

Extended basic (C) 75.000

Compactor (D) 30.000

Scmpactor (D) 30.000

Superbase (B) 120.000

Basic Hedge (C) 95.000

Toto 13 (D/N) 60.000

TURBO DISK (D) 60.000

FAST FORMATTER 40.000

ISAM 64 75.000

Character Editor 28.000

Sprite Editor 28.000

Protector 250.000

Chiavi protezione 50.000

GRAFICA/MUSICA

Ultrabasic (D) 70.000

Pictograph (C) 69.000

Magic paint (D) 70.000

Koala paint (D) 95.000

Panorama (D)\* 65.000

Synthy (D)\* 70.000

SAM RECITER (D) 80.000

MUSICALC 1/2/3 (D) 120.000

PER VIC 20

Mailing list(D) 60.000

Magazzino (D) 95.000

Fatturazione(D) 95.000

Monitor (N) 28.000

40 colonne (N) 22.000

HARDWARE

Cavo centronics 38.000

Int. 64-Centronics 95.000

Pet/IEEE-Centr. 120.000

Buffer 8K Centr. 220.000

Int. ET 121-221 250.000

Monitor Verde 12" 179.000

" Arancio 189.000

Stamp. Tally MT80 690.000

Espan. 16K VIC 118.000

" 32K VIC 145.000

Joystick 22.000

Dischi scat. 10 39.500

Nastri C10-20-30x10 12.000

Vic Eprom progr. 180.000

Vic Mot.Bo.(45lots) 59.000

GIOCHI

RICHIEDERE CATALOGO PARTICOLARE

MANUALI IN ITALIANO

Pet Speed 15.000

Easy Script 20.000

Simon Basic 20.000

Master 25.000

Tool 15.000

Superbase 25.000

Vizawrite 15.000

Colossus (scacchi) 3.000

The Clone 10.000

Unguard 10.000

Statistica 12.000

Multiplan (HELP) 12.000

KPM Pascal 10.000

Pictograf 5.000

Word Pro.III 7.000

GUIDA AL CBM 64 25.000

\*\*Nuova edizione,riveduta e ampliata. L'unico con il Commodore Approved.

SISTEMA OPERATIVO 64 38.000

\*\*Questa edizione viene fornita con un programma Disassembler, Assembler, Monitor (N)

I SEGRETI DEL 1541 28.000

Tutto ciò che e' necessario sapere sul disco.Sistema Operativo disassemblato, la Pagina Zero, le routines, i Relatives approfonditi, il SORT, gli OVERLAY, Tecniche di protezione e sprotezione, Monitor per disco, Disassemblatore DOS e disco.

PERIFERICHE COMMODORE 25.000

\*\*Questo manuale e' stato scritto per insegnare a comprendere ed usare TUTTE le periferiche dei prodotti COMMODORE. 430 pagine. Files relatives su 1541. Lettura e scrittura dati e funzionamento HARDWARE. tavole BAM e DIRECTORY. Utilizzo delle porte IEEE-488,IEEE seriale, RS-232. Spiegazioni di quasi tutte le stampanti: 1515, 1525, 1526, MPS801, MPS802, 3022, 4022. Numerosi programmi fra cui: RECUPERO FILES, CROSS REFERENCE e addirittura un DATA BASE.

GUIDA AL PERSONAL VIC/20 25.000

\*\*Il piu' completo manuale che vi SVELA come e' costruito e come funziona questo computer. Collegamenti

elettrici, mappe di memoria, il linguaggio macchina. Tutta la grafica gestibile ed il suono.

CORSO DI GRAFICA 24.000

Come utilizzare la grafica anche senza essere programmatori. Come scrivere i giochi e come dare il movimento alle immagini. 12 lezioni, 4 programmi di base oltre 40 pagine di tavole.

ACCOPIATORE ACUSTICO

E' la grande novita' che permette di collegarsi a qualsiasi banca dati, scambiarsi programmi e notizie fra utenti. Disponibile prestissimo un centro di collegamento e scambio presso EVM. CHIEDERE OPUSCOLO GRATUITO sull' accoppiatore, modem e banche dati.

Nome \_\_\_\_\_

Cognome \_\_\_\_\_

Via \_\_\_\_\_

C.A.P. \_\_\_\_\_ Città' \_\_\_\_\_

INVIATEMI:

CATALOGO

CATALOGO GIOCHI

\_\_\_\_\_

LEGENDA/CONDIZIONI

D=Disco / N=Nastro / C=Cartridge / \*\*Con manuale in inglese

I prezzi, tranne che per i manuali, sono al netto di IVA. Per spedizioni in contrassegno, calcolare L. 5.500 per spese postali e varie. Con pagamento anticipato SPEDIZIONE GRATUITA. SCONTI PER I SIG. RIVENDITORI. CATALOGO GRATUITO A RICHIESTA. INSERIMENTO GRATUITO IN LISTA DI AGGIORNAMENTO. TUTTI I MESI NOVITA' SENZA IMPEGNO.

# TA

TRIUMPH-ADLER

- Modelli: PC -P3 - P4
- Minifloppy: da 320 Kbytes a 785 Kbytes
- Hard disk: da 5 Mbytes
- Video a fosf. verdi: 24 × 80 car.
- Stampanti:  
ad aghi DRH80-DRH136-DRS250;  
a margherita TRD170S
- Macchine da scrivere interfacciabili
- Linguaggi: BASIC - PASCAL  
FORTRAN IV - COBOL
- Prezzi da L. 1.050.000  
a L. 9.600.000



## BIBLIOTECA PROGRAMMI ALPHATRONIC

**Contabilità generale:** Partitari - Situazione contabile - Registri IVA - Denunce e allegati annuali IVA.

**Contabilità semplificata:** Registri IVA - Riepiloghi periodici - Situazione contabile - Elenco clienti e fornitori.

**Paghe e stipendi:** Cedolino - Quadrature - Elaborazioni mensili - Servizi annuali.

**Amministrazione condominiale:** Ripartizione - Acconti - Spese - Fornitori.

**Magazzino e fatturazione:** Gestione preventivi; conferma d'ordine, bolle in automatico con fatturazione (immediata, differita, accompagnatoria) in codice alfanumerico - Listino - Giornale - Inventari valorizzati: prezzi di acquisto - Inventari valorizzati: prezzi di vendita - tratte e ricevute bancarie - Statistica di vendita.

**Legge 373:** Calcolo e progettazione delle dispersioni termiche di un edificio.

**Programmi di utilità:** Cross-reference - Dump memoria/disco - Routine in assembler - Auto-index.

**Gestione Periti Infortunistica:** gestione tempario ed emissione automatica perizie, stampa inviti, stampa parcelle e fatture, situazione pagamenti con assicurazioni, ricerca su archivio storico.

**Gestione Aziende Vitivinicole:** gestione prodotti iniziali e finali, emissione bolle e fatture - vuoti e cauzioni - aggiornamenti e stampa registri.

**Mercati Ortofrutticoli:** Gestione partite, ordini, fatture, clienti, fornitori, stampe riepilogative di fine giornata, trasmissione dati per studi commerciali.

**Ottici e Lenti Corneali:** Gestione archivio clienti con prima e successive prescrizioni. Gestione magazzini: Lenti corneali - Ottica - Materiali - Sbozzi e semilavorati. Statistiche.

**Ingegneria in regime sismico - Data Base - Text editor - Mailing list - Alberghi**

**Case di spedizionieri e trasporti - Controlli numerici - Laboratori analisi**

**Collegamento HP-3000 come terminale intelligente - Gestione assicurazioni - Medicaldata**

**Word-processing (utilizzabile con Triumph-Adler SE 1010 o SE 1005) - Ingegneria Civile/2**

**Disponibili schede di tipo:** Parallela - Grafica ad alta risoluzione - Seriale - Orologio - Eprom Programmer.

RIVENDITORI AUTORIZZATI TRIUMPH-ADLER:

FIRENZE  
VIAREGGIO (LU)  
L'AQUILA  
ROMA  
ROMA  
ROMA  
MONTEROTONDO (RM)  
NAPOLI

SOFTLAB s.r.l.  
LOGICA COMPUTER s.a.s.  
S.E.A.D. s.r.l.  
EMMEPI COMPUTER s.n.c.  
PRIMAPRINT s.r.l.  
SIPED s.n.c.  
LINEA UFFICIO s.n.c.  
C.I.E.D. s.r.l.

V.le Europa 101 - Tel. 055/683859  
Via Fratti 165 - Tel. 0584/42704 - 32353  
Via dei Giardini 16/a - Tel. 0862/315754  
Via Accademia dei Virtuosi 7 - Tel. 06/5410273  
Via Ulisse Dini 10 - Tel. 06/5578338  
Via Amelia 32 - Tel. 06/787013  
Via S. Martino 36 - Tel. 06/9001082  
Via Broggia 11 - 80136 NAPOLI  
Tel. 081/8282545 - 8273625