

# L'ADP Basic

51 nuovi comandi per le vostre periferiche

di Andrea de Prisco

*Continua la serie di articoli sull'implementazione di nuove istruzioni Basic sul vostro Commodore 64. Dopo aver visto le istruzioni più o meno standard per l'uso di una unità a dischetti, in questo numero vedremo comandi un po' più evoluti, sempre per la gestione del driver 1541. Potremo cambiare nome a un dischetto, visionare un programma senza occupare la memoria, curiosare tra tracce e settori di un floppy ecc. ecc.*

## Seconda parte

Divideremo questo articolo in due sezioni. Nella prima daremo una descrizione dei comandi implementati, loro uso e sintassi (tipo manuale di istruzioni) senza scendere nei dettagli organizzativi. Nella seconda sezione, spiegheremo come e perché tali comandi funzionano a livello di interprete Basic, riferendoci ai disassemblati listati in queste pagine. Al lettore la scelta su cosa leggere. È ovvio che per intenderci sulla seconda sezione è richiesta una seppur minima conoscenza di linguaggio macchina 6510, o 6502 se preferite.

### Altri 8 comandi

I comandi presentati questo mese sono in tutto 8, alcuni dei quali veramente inediti. Inizieremo dal comando EXE che carica e manda in esecuzione un programma da disco. La sua sintassi è:

EXE "NomeProgramma"  
dove "NomeProgramma" è il file da caricare e mandare in esecuzione. Equivale a un DLOAD (visto nel numero scorso) seguito a fine caricamento da un RUN + [RETURN]. Anche questo comando di caricamento è possibile darlo da Directory, semplicemente posizionandosi sul programma scelto (dopo il CAT) e digitando EXE seguito da qualche spazio per cancellare il tipo di file mostrato su video insieme al nome vero e proprio.

BLOAD "NomeProgramma"  
Questo comando carica da disco un programma (presumibilmente, ma non necessariamente) in linguaggio macchina, posizionandolo non all'inizio area Ram utente, ma nello stesso punto a partire dal quale era stato salvato. Corrisponde alla sequenza Basic LOAD "NomeProgramma", 8,1.  
BSAVE "NomeProgramma,Inizio,Fine"

Serve per salvare programmi L.M. o più in generale sezioni di memoria, diverse da quella occupata da un programma Basic. Gli indirizzi iniziale e finale vanno espressi in esadecimale, di 4 cifre, e (molto importante!) inseriti fra apici di seguito al Nome-

Programma. Esempio: vogliamo salvare su disco la Rom del Basic (indirizzi \$A000-\$BFFF):

BSAVE "RomBasic,A000,BFFF" [RETURN]

Il quarto comando di questo mese è RANGE e serve per conoscere, senza caricare nulla, dove sarà posizionato un programma usando il comando BLOAD. La sua sintassi è:

RANGE "NomeProgramma"  
e stampa su video in decimale l'indirizzo iniziale e finale. Se ad esempio, dopo aver salvato con BSAVE la Rom del Basic digitiamo:

RANGE "RomBasic" [RETURN]  
vedremo apparire su video i valori 40960 e 49151 (corrispondenti a \$A000 e \$BFFF).  
TRSE Traccia,Settore

È usato per leggere un settore di una determinata traccia di un dischetto. Dato che non tutte le tracce hanno uguale numero di settori, in figura 3 è riportata la corrispondenza Numero Traccia-Numero Settori.

APPEND "NomeProgramma"

Serve per saldare brutalmente due programmi Basic, uno in memoria e l'altro su disco. Brutalmente vuol dire che i numeri linea dei due programmi non hanno voce in capitolo: dopo l'ultima linea del primo programma è posizionata la prima linea del secondo. Si dia uno sguardo alla foto 3.

Per cambiare il nome a un dischetto senza riformattarlo, ossia senza perdite di file, è disponibile il comando:

DISKNAME "NuovoNome"

In una primitiva versione di questo comando era possibile cambiare anche l'identificatore. Poi qualcuno (grazie Fulvio, come farei senza di te, n.d.r.) mi ha ricordato che l'identificatore non sta solo nella Directory ma inciso al momento del formattamento nell'header di tutti i settori. Cambiarlo solo nella directory non ha effetto: per il driver, l'ID è quello inciso su tutto il dischetto. Cambiarlo in tutti i settori non è possibile da computer, ma solo da parte del sistema operativo del disco (leggi

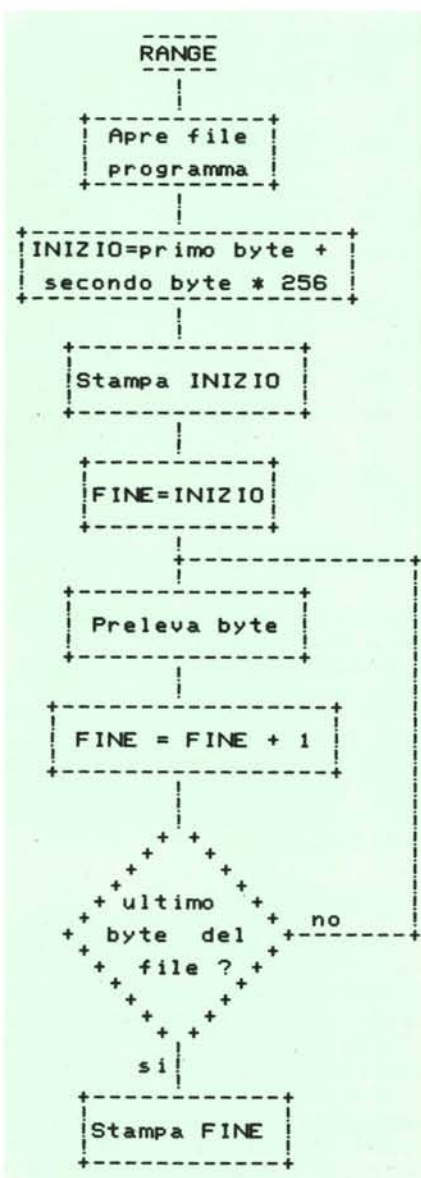


Figura 1 - Flow-chart del comando RANGE.

Numero Traccia	Settori	Totale
1 - 17	0 - 20	21
18 - 24	0 - 18	19
25 - 30	0 - 17	18
31 - 35	0 - 16	17
Totale Settori		664

```

READY.

RANGE "LOADER"
2049
3072

READY.
    
```

Foto 1 - Comando RANGE.

Figura 3 - Corrispondenza traccia-settori di un dischetto.

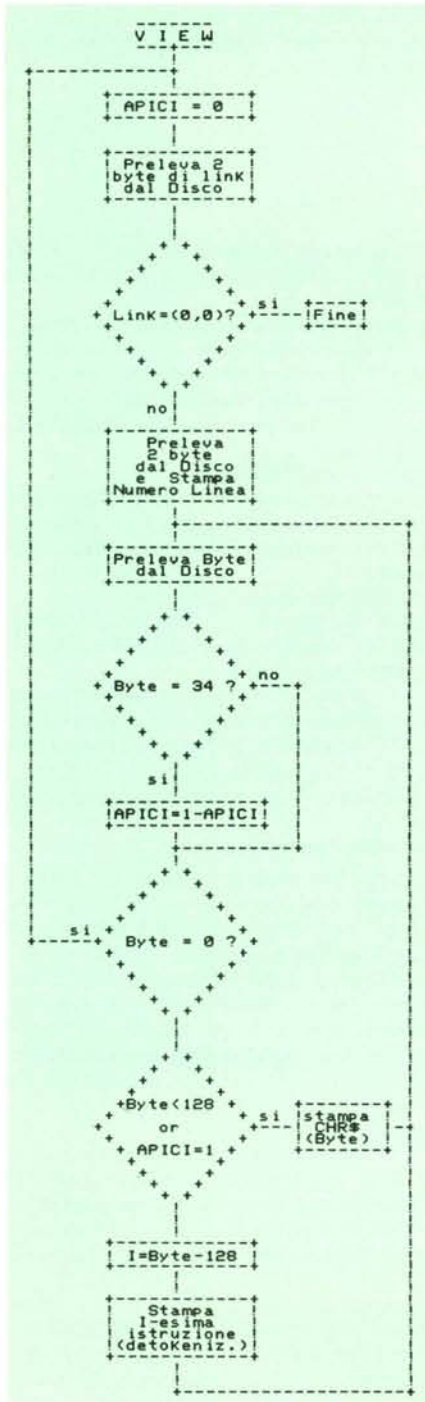


Figura 2 - Flow-chart del comando VIEW.

Indirizzo	Nome	Registri	Descrizione
%%FBA	Set logical, first and second address	A,X,Y	Setta il numero logico, l'indirizzo primario e l'indirizzo secondario di un file prima dell'apertura. In A si pone il n. file, in X il device e in Y l'ind. secondario.
%%FBD	Set file name information	A,X,Y	Setta le informazioni circa il nome di un file prima della sua apertura. Si indica in A la lunghezza del nome, in X e in Y l'indirizzo dove e' attivato.
%%FBC	Open logical file	---	Apri il file specificato con le routine %%FBA e %%FBD.
%%FCD	Close logical file	A	Chiude il file il cui numero e' specificato in A.
%%FCB	Open channel for input	X	Predispongono un file precedentemente aperto come canale di input. (trasferimento dati da periferica a CPU)
%%FCB	Open channel for output	X	Predispongono un file precedentemente aperto come canale di output. (trasferimento dati da CPU a periferica)
%%FCC	Close input and output channels	---	Resetta i canali di input e di output.
%%FCF	Input character from channel	A	Ogni chiamata a questa routine provoca un get per il file input precedentemente aperto. In A il codice Ascii del carattere letto.
%%FDE	Output character to channel	A	Scarica nel file output precedentemente aperto il carattere contenuto in A.
%%FE7	Close all file	---	Chiude tutti i file aperti. (resetta la File Table del 64)
%%FB7	Read I/O status word	A	Trasferisce in A il valore della variabile ST. Usata per conoscere se e' stato letto l'ultimo carattere del file (A=64).
%%FE1	Check stop key	PSW	Serve per controllare la pressione del tasto RUN/STOP. Pone il bit 2 della PSW a 1 se durante la chiamata di questa routine il tasto RUN/STOP e' premuto.

Figura 4 - Le routine del Kernal.

lavoraccio da cani), morale della favola: chi si accontenta gode.

L'ultimo comando di questo mese è: VIEW "NomeProgramma" e serve per visionare un programma, vedere il suo listato, senza caricarlo in memoria: direttamente dal disco sul video (o sui vostri teleschermi, se vi fa più comodo). Per bloccare lo scroll è possibile agire sul tasto Run/Stop.

**General Remarks**

Il listato 1 implementa l'istruzione EXE. Come facilmente prevedibile, il suo funzionamento è abbastanza semplice: si limita a inserire nel buffer di tastiera la sequenza "[CLR][R][Shift U][RETURN]" e di cedere il controllo al normale DLOAD (indirizzo %C4AC). Il primo carattere (corrispondente a un cuoricino reverse) serve per cancellare il video; il secondo unito al terzo sono l'abbreviazione del comando RUN, il quarto è il Return da tastiera. Appena

caricato il programma, il buffer di tastiera si svuota cancellando il video e dando RUN al programma.

Il listato 2 mostra l'implementazione del comando BLOAD: anch'esso molto semplice. Come DLOAD, non fa che settare alcuni byte prima di cedere il controllo al LOAD del Basic standard. Oltre a indicare come periferica il disco (device # 8), inserisce in %B9 il numero 1. %B9 è appunto l'indirizzo secondario del LOAD.

L'istruzione BSAVE è implementata dal listato 4; la subroutine del listato 3 è adoperata da BSAVE per convertire il codice ASCII di una cifra Hex (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F) nel suo effettivo valore, compreso cioè tra 0 e 15.

Fino all'indirizzo Hex %C667, il programma BSAVE non fa altro che convertire i parametri di inizio e fine passati insieme al nome del file, ponendoli nelle locazioni FB-FC e A8-A9. Naturalmente controlla la sintassi (numero cifre e presenza di virgole al punto giusto) eventualmente in-

```

C5B6 A9 04 LDA #04
C5B8 85 C6 STA #C6
C5BA A9 93 LDA #93
C5BC 8D 77 02 STA #0277
C5BF A9 52 LDA #52
C5C1 8D 78 02 STA #0278
C5C4 A9 05 LDA #05
C5C6 8D 79 02 STA #0279
C5C9 A9 0D LDA #0D
C5CB 8D 7A 02 STA #027A
C5CE 4C AC C4 JMP #C4AC

```

Listato 1 - Comando EXE.

```

C5D1 A9 00 LDA #00
C5D3 85 0A STA #0A
C5D5 20 04 E1 JSR #E104
C5D8 A9 08 LDA #08
C5DA 85 BA STA #BA
C5DC A9 01 LDA #01
C5DE 85 B9 STA #B9
C5E0 4C 6F E1 JMP #E16F

```

Listato 2 - Comando BLOAD.

```

C5E3 29 CF AND #CF
C5E5 85 AA STA #AA
C5E7 24 AA BIT #AA
C5E9 50 03 BVC #C5EE
C5EB 18 CLC
C5EC 69 09 ADC #09
C5EE 29 0F AND #0F
C5F0 60 RTS

```

Listato 3

Subroutine usata da BSAVE.

```

C5F1 20 04 E1 JSR #E104
C5F4 A4 B7 LDY #B7
C5F6 88 DEY
C5F7 B1 BB LDA (#BB),Y
C5F9 20 E3 C5 JSR #C5E3
C5FC 85 A8 STA #A8
C5FE 88 DEY
C5FF B1 BB LDA (#BB),Y
C601 20 E3 C5 JSR #C5E3
C604 0A ASL
C605 0A ASL
C606 0A ASL
C607 0A ASL
C608 18 CLC
C609 65 A8 ADC #A8
C60B 85 A8 STA #A8
C60D 88 DEY
C60E B1 BB LDA (#BB),Y
C610 20 E3 C5 JSR #C5E3
C613 85 A9 STA #A9
C615 88 DEY
C616 B1 BB LDA (#BB),Y
C618 20 E3 C5 JSR #C5E3
C61B 0A ASL
C61C 0A ASL
C61D 0A ASL
C61E 0A ASL
C61F 18 CLC
C620 65 A9 ADC #A9
C622 85 A9 STA #A9
C624 88 DEY
C625 B1 BB LDA (#BB),Y
C627 C9 2C CMP #2C
C629 F0 05 BEQ #C630
C62B A2 0B LDX #0B
C62D 4C 8B E3 JMP #E38B
C630 88 DEY
C631 B1 BB LDA (#BB),Y
C633 20 E3 C5 JSR #C5E3
C636 85 FB STA #FB
C638 88 DEY
C639 B1 BB LDA (#BB),Y
C63B 20 E3 C5 JSR #C5E3
C63E 0A ASL
C63F 0A ASL

```

```

C640 0A ASL
C641 0A ASL
C642 18 CLC
C643 65 FB ADC #FB
C645 85 FB STA #FB
C647 88 DEY
C648 B1 BB LDA (#BB),Y
C64A 20 E3 C5 JSR #C5E3
C64D 85 FC STA #FC
C64F 88 DEY
C650 B1 BB LDA (#BB),Y
C652 20 E3 C5 JSR #C5E3
C655 0A ASL
C656 0A ASL
C657 0A ASL
C658 0A ASL
C659 18 CLC
C65A 65 FC ADC #FC
C65C 85 FC STA #FC
C65E 88 DEY
C65F B1 BB LDA (#BB),Y
C661 C9 2C CMP #2C
C663 F0 05 BEQ #C66A
C665 A2 0B LDX #0B
C667 4C 8B E3 JMP #E38B
C66A EA NOP
C66B 84 B7 STY #B7
C66D A9 41 LDA #41
C66F A2 0B LDX #0B
C671 A0 01 LDY #01
C673 20 BA FF JSR #FFBA
C676 20 C0 FF JSR #FFC0
C679 A2 41 LDX #41
C67B 20 C9 FF JSR #FFC9
C67E A5 FB LDA #FB
C680 20 D2 FF JSR #FFD2
C683 A5 FC LDA #FC
C685 20 D2 FF JSR #FFD2
C688 A0 00 LDY #00
C68A B1 FB LDA (#FB),Y
C68C 20 D2 FF JSR #FFD2
C68F E6 FB INC #FB
C691 00 02 BNE #C695
C693 E6 FC INC #FC
C695 A5 FB LDA #FB

```

```

C697 C5 A8 CMP #A8
C699 D0 EF BNE #C68A
C69B A5 FC LDA #FC
C69D C5 A9 CMP #A9
C69F D0 E9 BNE #C68A
C6A1 20 CC FF JSR #FFCC
C6A4 A9 41 LDA #41
C6A6 20 C3 FF JSR #FFC3
C6A9 4C 8C C4 JMP #C48C

```

Listato 4 - Comando BSAVE.

```

C6AC 20 D4 E1 JSR #E104
C6AF A9 43 LDA #43
C6B1 A2 08 LDX #08
C6B3 A0 00 LDY #00
C6B5 20 BA FF JSR #FFBA
C6B8 20 C0 FF JSR #FFC0
C6BB A2 43 LDX #43
C6BD 20 C6 FF JSR #FFC6
C6C0 20 CF FF JSR #FFCF
C6C3 85 FC STA #FC
C6C5 20 CF FF JSR #FFCF
C6C8 85 FD STA #FD
C6CA A6 FC LDX #FC
C6CC EA NOP
C6CD 20 CD BD JSR #BDCD
C6D0 A9 0D LDA #0D
C6D2 20 D2 FF JSR #FFD2
C6D5 E6 FC INC #FC
C6D7 00 02 BNE #C6D0B
C6D9 E6 FD INC #FD
C6DB 20 CF FF JSR #FFCF
C6DE 20 B7 FF JSR #FFB7
C6E1 F0 F2 BEQ #C6D5
C6E3 20 CC FF JSR #FFCC
C6E6 A9 43 LDA #43
C6E8 20 C3 FF JSR #FFC3
C6EB A6 FC LDX #FC
C6ED A5 FD LDA #FD
C6EF 20 CD BD JSR #BDCD
C6F2 A9 0D LDA #0D
C6F4 20 D2 FF JSR #FFD2
C6F7 4C 8C C4 JMP #C48C

```

Listato 5 - Comando RANGE.

## Senza l'ADP...

I comandi dell'ADP BASIC presentati su questo numero provvedono a salvare e a caricare da disco zone di memoria, a caricare programmi Basic e a mandarli in esecuzione, saldare insieme due programmi, cambiare il nome a un dischetto, visionare un programma che sta sul disco senza occupare la memoria, curiosare in un settore, e avere informazioni circa l'inizio e la fine di un programma sul dischetto. Senza ADP BASIC, realizzare le stesse funzioni è un po' un pasticcio. L'unico comando "facile" è il BLOAD: equivale a un LOAD "Programma", 8, 1. Per salvare di contro un programma in linguaggio macchina, posizionato in una zona diversa da quella generalmente usata dai programmi Basic, bisogna spostare i puntatori di inizio e fine programma (celle 43, 44, 45 e 46). Si fa coincidere il puntatore di inizio con il primo byte da salvare e il puntatore di fine con l'ultimo: segue un normalissimo SAVE.

Il comando EXE può facilmente essere simulato scrivendo il normale comando di caricamento LOAD "Programma", 8 e digitando al posto del [RETURN] la sequenza ":SHIFT + Run/Stop". Provare per credere.

Niente da fare per i comandi APPEND e VIEW, senza ADP BASIC non hanno molto senso: in ogni caso si tratta di scrivere opportune routine in linguaggio macchina (leggi: non ci si guadagna nulla).

Il comando RANGE si può simulare aprendo un file di tipo programma (indirizzo secondario): i primi due byte letti sono l'indirizzo iniziale; per conoscere l'indirizzo finale non c'è altro modo che continuare a leggere fino al segnale di fine file (variabile ST = 64) contando a partire da inizio. Si dia uno sguardo anche al flow di figura 1.

Per i comandi TRSE e DISKNAME diamo qui a fianco i due listatini Basic che implementano tali funzioni. Il primo è stato già pubblicato sul n. 33, articolo i Segreti del Disco, stesso autore.

```

10 INPUT "TRACCA & SETTORE ";TR,SE
20 OPEN1,8,15
30 OPEN5,8,5,"#"
40 PRINT#1,"U1:";5;0;TR;SE
50 FOR I=0TO255:
60 GET#5,A$:IFA$=" "THENA$=CHR$(0)
70 A$=STR$(ASC(A$))
80 PRINTRIGHT$(" "+A$,4);:NEXT
90 CLOSE5:CLOSE1
READY.

```

```

100 DIMA$(255):OPEN1,8,15,"I"
110 OPEN3,8,3,"#"
120 PRINT#1,"U1:";3;0;18;0
130 FOR I=0TO255:GET#3,A$:IFA$=" "THENA$=CHR$(0)
140 A$(I)=A$:NEXT
150 CLOSE3:CLOSE1
170 A$="":INPUT "NUOVO NOME = ";A$
180 A$=LEFT$(A$+" ",16)
190 FOR I=1TOLEN(A$):A$(143+I)=MID$(A$,I,1):NEXT
240 OPEN1,8,15,"I"
250 OPEN3,8,3,"#"
260 PRINT#1,"B-P:";3;0
270 FOR I=0TO255:PRINT#3,A$(I):NEXT
280 PRINT#1,"B-P:";3;0
290 PRINT#1,"U2:";3;0;18;0
300 CLOSE3:PRINT#1,"I":CLOSE1:END
READY.

```

```

C6FA 20 EB B7 JSR #B7EB
C6FD 06 FC STX #FC
C6FF 20 3B C0 JSR #C03B
C702 A9 43 LDA #43
C704 A2 08 LDX #08
C706 A0 05 LDY #05
C708 20 BA FF JSR #FFBA
C70B A9 01 LDA #01
C70D A2 3A LDX #3A
C70F A0 A0 LDY #A0
C711 20 B0 FF JSR #FFB0
C714 20 C0 FF JSR #FFC0
C717 A2 48 LDX #48
C719 20 C9 FF JSR #FFC9
C71C A9 E0 LDA #E0
C71E A0 CF LDY #CF
C720 20 1E AB JSR #AB1E
C723 A6 14 LDX #14
C725 A9 00 LDA #00
C727 20 CD BD JSR #BDCD
C72A A9 2C LDA #2C
C72C 20 D2 FF JSR #FFD2
C72F A6 FC LDX #FC

```

Listato 6 - Comando TRSE.

```

C731 A9 00 LDA #00
C733 20 CD BD JSR #BDCD
C736 A9 00 LDA #00
C738 20 D2 FF JSR #FFD2
C73B 20 CC FF JSR #FFCC
C73E A2 43 LDX #43
C740 20 C6 FF JSR #FFC6
C743 A2 00 LDX #00
C745 06 AA STX #AA
C747 A9 20 LDA #20
C749 20 D2 FF JSR #FFD2
C74C 20 CF FF JSR #FFCF
C74F AA TAX
C750 A9 00 LDA #00
C752 20 C8 C4 JSR #C4C8
C755 A0 8D 02 LDA #028D
C758 D0 FB BNE #C755
C75A E6 AA INC #AA
C75C D0 E9 BNE #C747
C75E 20 CC FF JSR #FFCC
C761 A9 43 LDA #43
C763 20 C3 FF JSR #FFC3
C766 4C 8C C4 JMP #C48C

```

```

C769 20 D4 E1 JSR #E1D4
C76C A5 20 LDA #20
C76E 38 SEC
C76F E9 02 SBC #02
C771 AA TAX
C772 A5 2E LDA #2E
C774 E9 00 SBC #00
C776 A8 TAY
C777 A9 08 LDA #08
C779 85 BA STA #BA
C77B A9 00 LDA #00
C77D 20 D5 FF JSR #FFD5
C780 20 33 A5 JSR #A533
C783 38 SEC
C784 A5 20 LDA #20
C786 E9 02 SBC #02
C788 85 57 STA #57
C78A A5 2E LDA #2E
C78C E9 00 SBC #00
C78E 85 58 STA #58
C790 A0 00 LDY #00
C792 B1 57 LDA (#57),Y
C794 D0 1D BNE #C7B3
C796 C8 INY

```

```

C797 B1 57 LDA (#57),Y
C799 D0 18 BNE #C7B3
C79B A5 57 LDA #57
C79D 18 CLC
C79E 69 02 ADC #02
C7A0 85 2D STA #2D
C7A2 85 2F STA #2F
C7A4 85 31 STA #31
C7A6 A5 58 LDA #58
C7A8 69 00 ADC #00
C7AA 85 2E STA #2E
C7AC 85 30 STA #30
C7AE 85 32 STA #32
C7B0 4C 8C C4 JMP #C48C
C7B3 A0 00 LDY #00
C7B5 B1 57 LDA (#57),Y
C7B7 85 59 STA #59
C7B9 C8 INY
C7BB B1 57 LDA (#57),Y
C7BC 85 58 STA #58
C7BE A5 59 LDA #59
C7C0 85 57 STA #57
C7C2 18 CLC
C7C3 90 C8 BCC #C790

```

Listato 7 - Comando APPEND.

```

C7C5 20 D4 E1 JSR #E1D4
C7C8 A2 11 LDX #11
C7CA A9 A0 LDA #A0
C7CC 9D 40 03 STA #0340,X
C7CF CA DEX
C7D0 10 FA BPL #C7CC
C7D2 A4 B7 LDY #B7
C7D4 F0 0A BEQ #C7E0
C7D6 88 DEY
C7D7 B1 BB LDA (#BB),Y
C7D9 99 40 03 STA #0340,Y
C7DC C0 00 CPY #00
C7DE D0 F6 BNE #C7D6
C7E0 20 3B C0 JSR #C03B
C7E3 A2 48 LDX #48
C7E5 20 C9 FF JSR #FFC9
C7E8 A9 49 LDA #49
C7EA 20 D2 FF JSR #FFD2
C7ED 20 CC FF JSR #FFCC
C7F0 A9 43 LDA #43

```

Listato 8 - Comando DISKNAME.

```

C7F2 A2 08 LDX #08
C7F4 A0 03 LDY #03
C7F6 20 BA FF JSR #FFBA
C7F9 A9 01 LDA #01
C7FB A2 3A LDX #3A
C7FD A0 A0 LDY #A0
C7FF 20 B0 FF JSR #FFB0
C802 20 C0 FF JSR #FFC0
C805 A2 48 LDX #48
C807 20 C9 FF JSR #FFC9
C80A A9 F0 LDA #F0
C80C A0 CF LDY #CF
C80E 20 1E AB JSR #AB1E
C811 20 CC FF JSR #FFCC
C814 A2 43 LDX #43
C816 20 C6 FF JSR #FFC6
C819 A2 00 LDY #00
C81B 20 CF FF JSR #FFCF
C81E 9D 00 A0 STA #A000,X
C821 E8 INX

```

```

C822 D0 F7 BNE #C81B
C824 20 CC FF JSR #FFCC
C827 A2 11 LDX #11
C829 BD 40 03 LDA #0340,X
C82C 9D 90 A0 STA #A090,X
C82F CA DEX
C830 10 F7 BPL #C829
C832 A2 48 LDX #48
C834 20 C9 FF JSR #FFC9
C837 A9 E8 LDA #E8
C839 A0 CF LDY #CF
C83B 20 1E AB JSR #AB1E
C83E 20 CC FF JSR #FFCC
C841 A2 43 LDX #43
C843 20 C9 FF JSR #FFC9
C846 A5 01 LDA #01
C848 29 FE AND #FE
C84A 85 01 STA #01
C84C A2 00 LDX #00
C84E BD 00 A0 LDA #A000,X

```

```

C851 20 D2 FF JSR #FFD2
C854 E8 INX
C856 D0 F7 BNE #C84E
C857 20 CC FF JSR #FFCC
C85A A5 01 LDA #01
C85C 09 01 ORA #01
C85E 85 01 STA #01
C860 A9 32 LDA #32
C862 8D F1 CF STA #CFF1
C865 A2 48 LDX #48
C867 20 C9 FF JSR #FFC9
C86A A9 F0 LDA #F0
C86C A0 CF LDY #CF
C86E 20 1E AB JSR #AB1E
C871 20 CC FF JSR #FFCC
C874 A9 43 LDA #43
C876 20 C3 FF JSR #FFC3
C879 A9 31 LDA #31
C87B 8D F1 CF STA #CFF1
C87E 4C 00 C4 JMP #C400

```

```

C881 20 D4 E1 JSR #E1D4
C884 A9 00 LDA #00
C886 85 BA STA #BA
C888 A9 49 LDA #49
C88A 85 B8 STA #B8
C88C 20 C0 FF JSR #FFC0
C88F 20 41 C0 JSR #C041
C892 A2 48 LDX #48
C894 20 C6 FF JSR #FFC6
C897 20 CF FF JSR #FFCF
C89A 85 FC STA #FC
C89C 20 CF FF JSR #FFCF
C89F 18 CLC
C8A0 65 FC ADC #FC
C8A2 85 FC STA #FC
C8A4 20 CF FF JSR #FFCF
C8A7 20 B7 FF JSR #FFB7
C8AA C9 40 CMP #40
C8AC D0 F6 BNE #C8A4
C8AE 20 CC FF JSR #FFCC
C8B1 A5 FC LDA #FC
C8B3 C9 60 CMP #60
C8B5 F0 08 BEQ #C8BF
C8B7 20 E7 FF JSR #FFE7

```

Listato 9 - Comando VIEW.

```

C88A A2 04 LDX #04
C88C 4C 8B E3 JMP #E38B
C88F A2 49 LDX #49
C8C1 20 C6 FF JSR #FFC6
C8C4 20 CF FF JSR #FFCF
C8C7 20 CF FF JSR #FFCF
C8CA 20 CF FF JSR #FFCF
C8CD AA TAX
C8CE 20 CF FF JSR #FFCF
C8D1 E0 00 CPX #00
C8D3 D0 06 BNE #C8D3
C8D5 C9 00 CMP #00
C8D7 D0 02 BNE #C8D7
C8D9 F0 6A BEQ #C845
C8DB A9 00 LDA #00
C8DD 85 FC STA #FC
C8DF A9 00 LDA #00
C8E1 20 D2 FF JSR #FFD2
C8E4 20 E1 FF JSR #FFE1
C8E7 F0 5C BEQ #C845
C8E9 20 CF FF JSR #FFCF
C8EC AA TAX
C8ED 20 CF FF JSR #FFCF
C8F0 20 CD BD JSR #BDCD

```

```

C8F3 A9 20 LDA #20
C8F5 20 D2 FF JSR #FFD2
C8F8 F0 CD BEQ #C8CA
C8FD C9 22 CMP #22
C8FF D0 08 BNE #C909
C901 48 PHA
C902 A5 FC LDA #FC
C904 A9 01 EOR #01
C906 85 FC STA #FC
C908 68 PLA
C909 C9 00 CMP #00
C90B 30 05 BMI #C912
C90D 20 D2 FF JSR #FFD2
C910 D0 E6 BNE #C8F8
C912 A4 FC LDY #FC
C914 D0 F7 BNE #C90D
C916 48 PHA
C917 A9 00 LDA #00
C919 85 A8 STA #A8
C91B A9 CD LDA #CD
C91D 85 A9 STA #A9
C91F A0 00 LDY #00
C921 68 PLA

```

```

C922 29 7F AND #7F
C924 AA TAX
C925 F0 0C BEQ #C933
C927 C8 INY
C928 D0 02 BNE #C92C
C92A E6 A9 INC #A9
C92C B1 A8 LDA (#A8),Y
C92E 10 F7 BPL #C927
C930 CA DEX
C931 D0 F4 BNE #C927
C933 C8 INY
C934 D0 02 BNE #C93B
C936 E6 A9 INC #A9
C938 B1 A8 LDA (#A8),Y
C93A 48 PHA
C93B 29 7F AND #7F
C93D 20 D2 FF JSR #FFD2
C940 68 PLA
C941 10 F0 BPL #C933
C943 30 B3 BMI #C8F8
C945 20 CC FF JSR #FFCC
C948 A9 49 LDA #49
C94A 20 C3 FF JSR #FFC3
C94D 4C F8 A8 JMP #A8F8

```

```

100 FOR I=49152 TO 49785:READ I
110 POKE I, I: NEXT
120 FOR I=50176 TO 50813:READ I
130 POKE I, I: NEXT
140 FOR I=52480 TO 52785:READ I
150 POKE I, I: NEXT
151 FOR I=49302 TO 49317:READ I
161 POKE I, I: NEXT
171 FOR I=52796 TO 53247:READ I
181 POKE I, I: NEXT
191 FOR I=50614 TO 51535:READ I
201 POKE I, I: NEXT
290 SYS 49152: END
300
310 +
320 +   ***   ****   ****   ****   ***   ***   ***   ***   +
330 +   *   *   *   *   *   *   *   *   *   *   *   *   *   *   +
340 +   *****   *   ****   ****   ****   ****   *   *   *   +
350 +   *   *   *   *   *   *   *   *   *   *   *   *   *   *   +
360 +   *   *   ****   *   ****   *   *   ***   ***   ***   +
370 +
380 +
390 +
100 +   (C) 1984   ADP SOFTWARE
110 +
100 *****
1000 DATA 120,169,30,141,4,3,169,193,141,5,3,169,68,141,6,3,169,194,141,7,3,234
1010 DATA 234,169,249,141,8,3,169,192,141,9,3,208,48,32,204,255,32,231,255,169
1020 DATA 68,162,4,160,0,32,186,255,169,0,32,189,255,32,192,255,96,32,204,255
1030 DATA 32,231,255,169,72,162,8,160,15,32,186,255,169,0,32,189,255,32,192,255
1040 DATA 96,169,11,141,32,208,141,33,208,169,155,32,210,255,96,0,0,0,0,0,0
1050 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,255,195,18,196,37,196
1060 DATA 52,196,91,196,106,196,139,196,171,196,185,196,195,196,18,197,0,0,0,0
1070 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1080 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1090 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1100 DATA 48,4,201,204,16,4,56,76,231,167,56,233,204,10,168,169,167,72,169,233
1110 DATA 72,185,129,192,72,185,128,192,72,76,115,0,166,122,160,4,132,15,189,0
1120 DATA 2,16,7,201,255,240,73,232,208,244,201,32,240,66,133,8,201,34,240,97
1130 DATA 36,15,112,56,201,63,208,4,169,153,208,48,201,48,144,4,201,60,144,40
1140 DATA 132,113,160,0,169,204,133,254,169,1,133,253,132,11,136,134,122,202,200
1150 DATA 208,2,230,254,232,189,0,2,56,241,253,240,242,201,128,208,48,5,11,164
1160 DATA 113,232,200,153,251,1,185,251,1,240,60,56,233,58,240,4,201,73,208,2
1170 DATA 133,15,56,233,85,208,148,133,8,189,0,2,240,223,197,8,240,219,200,153
1180 DATA 251,1,232,208,240,166,122,230,11,198,253,200,208,2,230,254,177,253,16
1190 DATA 247,230,253,177,253,208,175,189,0,2,16,184,153,253,1,198,123,169,255
1200 DATA 133,122,96,144,6,240,4,201,171,208,247,32,107,169,32,19,166,32,121,0
1210 DATA 240,12,201,171,208,232,32,115,0,32,107,169,208,224,104,104,165,20,5
1220 DATA 2,208,6,169,255,133,20,133,21,160,1,132,15,177,95,240,67,32,44,168
1230 DATA 32,215,170,200,177,95,170,200,177,95,197,21,208,4,228,20,240,2,176,44
1240 DATA 132,73,32,205,189,169,32,164,73,41,127,32,71,171,201,34,208,6,165,15
1250 DATA 73,255,133,15,200,240,17,177,95,208,16,168,177,95,170,200,177,95,134
1260 DATA 95,133,96,208,181,76,134,227,108,6,3,16,215,201,255,240,211,36,15,48
1270 DATA 207,56,233,127,170,132,73,160,255,169,204,133,255,169,1,133,254,202
1280 DATA 240,11,200,208,2,230,255,177,254,16,247,48,242,200,208,2,230,255,177
1290 DATA 254,48,164,32,71,171,208,242
1999 REM *****
2000 DATA 32,59,192,162,72,32,201,255,169,73,32,210,255,32,231,255,76,248,168
2010 DATA 32,59,192,162,72,32,201,255,169,86,32,210,255,32,231,255,76,248,168
2020 DATA 32,59,192,162,72,32,201,255,169,83,32,210,255,208,67,32,59,192,162,72
2030 DATA 32,201,255,169,78,32,210,255,169,58,32,210,255,32,212,225,160,0,177
2040 DATA 187,32,210,255,200,196,183,208,246,32,204,255,76,248,168,32,59,192,162
2050 DATA 72,32,201,255,169,67,32,210,255,208,13,32,59,192,162,72,32,201,255,169
2060 DATA 82,32,210,255,169,58,32,210,255,32,212,225,160,0,177,187,32,210,255
2070 DATA 200,196,183,208,246,169,13,32,210,255,32,59,192,162,72,32,198,255,32
2080 DATA 207,255,32,210,255,32,183,255,201,64,208,243,32,59,192,76,248,168,169
2090 DATA 0,133,10,32,212,225,169,8,133,186,76,111,225,32,212,225,169,8,133,186
2100 DATA 76,89,225,169,1,208,230,133,98,134,99,162,144,56,32,73,166,32,223,189
2110 DATA 133,253,132,254,160,255,162,0,200,177,253,208,251,132,252,56,169,3,229
2120 DATA 252,168,240,9,169,32,157,60,3,232,136,208,249,177,253,157,60,3,232,200
2130 DATA 201,0,208,245,202,169,32,157,60,3,232,169,0,157,60,3,169,60,160,3,76
2140 DATA 30,171,169,65,162,8,160,0,32,186,255,169,1,162,96,160,163,32,189,255
2150 DATA 32,192,255,162,65,32,198,255,32,207,255,32,207,255,32,207,255,32,207
2160 DATA 255,169,13,32,210,255,32,225,255,240,104,32,183,255,208,99,32,207,255
2170 DATA 133,254,32,207,255,166,254,32,200,196,32,207,255,234,234,201,34,240
2180 DATA 14,201,66,208,243,32,210,255,32,207,255,208,248,240,199,162,0,157,195
2190 DATA 2,232,32,207,255,157,195,2,201,34,208,245,32,207,255,201,32,240,249
2200 DATA 141,192,2,32,207,255,141,193,2,169,32,141,194,2,234,169,0,157,195,2
2210 DATA 162,255,232,189,192,2,240,5,32,210,255,208,245,32,207,255,208,251,240
2220 DATA 136,32,204,255,169,65,32,195,255,76,248,168
2999 REM *****
3000 DATA 0,69,78,196,70,79,210,78,69,88,212,68,65,84,193,73,78,80,85,84,163,73

```

(continua a pag. 138)

Listato Basic: le linee pari sono uguali a quelle del numero scorso. Le dispari implementano le istruzioni di questo mese.

viando messaggi di errore al video. Dall'indirizzo C66B in poi, dopo aver aperto un file di tipo programma (indirizzo secondario 1, siamo nel caso di scrittura) e aver spedito al disco il puntatore di inizio, manda al driver tutti i byte da salvare. Termina chiudendo ogni canale e il file di tipo PRG, saltando infine all'istruzione FLASH per visualizzare un eventuale messaggio del disco.

Il comando RANGE (listato 5) apre un file di tipo programma (indirizzo secondario 0, lettura) col nome passato come parametro. Legge i primi due byte, interpretandoli come indirizzo iniziale, e stampa il loro valore (primo byte + secondo byte \* 256) su video. Continua a leggere byte, incrementando opportunamente un contatore, fino a quando ST diventa 64. Termina stampando l'indirizzo di fine e saltando all'istruzione FLASH per eventuali errori di disco.

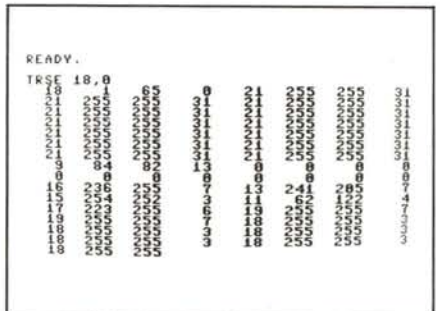


Foto 2 - Comando TRSE, è stata richiesta la traccia 18 settore 0.

Il listato 6 implementa il comando TRSE. Il salto alla subroutine \$B7EB serve per leggere i parametri passati. Al suo ritorno X conterrà il Settore, \$14 la Traccia. Per il resto fa esattamente quanto indicato (in Basic) nel riquadro a pag. 135.

Il comando APPEND pone in X e in Y l'inizio dell'area di caricamento (indirizzo di fine programma in memoria -2) e salta alla subroutine Kernel \$FFD5 (load from device). Il salto a \$A533 serve per concatenare linee Basic. Prima di terminare, scorre i link delle nuove linee per settare i vari puntatori di fine Basic, inizio variabili ecc.

Più in dettaglio vedremo le ultime due istruzioni, fiore all'occhiello di questa puntata. Il comando DISKNAME (listato 8) dopo aver letto il nuovo nome passato come parametro (JSR \$E1D4) prepara un buffer di 18 byte a partire dalla locazione \$340, inizialmente riempiendolo di "shift spazi". Successivamente pone il nuovo nome nel buffer (ciclo \$C7D6...\$C7DE). A questo punto apre il canale di comunicazione, dà un INIT al disco e apre un file dati di nome "\$" (allocazione di area Ram del disco per trasferimento dati).

Il comando spedito è "U1:3,0,18,0" (stivato nelle locazioni \$CFF0 e seguenti) e serve per trasferire nell'area Ram allocata il settore 0 della traccia 18 che contiene il nome del dischetto. Dalla Ram del disco il settore è trasferito nelle locazioni \$A000-

(segue da pag. 137)

```

3010 DATA78,80,85,212,68,73,205,82,69,65,196,76,69,212,71,79,84,207,82,85,206
3020 DATA73,198,82,69,83,84,79,82,197,71,79,83,85,194,82,69,84,85,82,206,82,69
3030 DATA205,83,84,79,208,79,206,87,65,73,212,76,79,65,196,83,65,86,197,86,69
3040 DATA82,73,70,217,68,69,198,80,79,75,197,80,82,73,78,84,163,80,82,73,78,212
3050 DATA67,79,78,212,76,73,83,212,67,76,210,67,77,136,83,83,211,79,80,69,206
3060 DATA67,76,79,83,197,71,69,212,78,69,215,84,65,66,168,84,207,70,206,83,80
3070 DATA67,169,84,72,69,206,78,79,212,83,34,69,208,171,173,170,175,222,65,78
3080 DATA196,79,210,190,189,180,83,71,206,73,78,212,65,66,211,65,83,210,70,82
3090 DATA197,80,79,211,83,81,210,82,78,196,76,79,199,69,88,208,67,79,211,83,73
3100 DATA206,84,65,206,65,84,206,80,69,69,203,76,69,206,83,84,82,164,86,65,204
3110 DATA65,83,195,67,72,82,164,76,69,70,84,164,82,73,71,72,64,164,77,73,68,164
3120 DATA71,207,73,78,73,212,96,68,65,84,197,69,82,65,84,197,70,77,65,212,67
3130 DATA79,80,217,82,69,78,65,77,197,70,76,65,83,200,68,76,79,65,196,68,63,65
3140 DATA86,197,68,86,69,210,67,65,212
1500 REM *****
3501 DATA191,197,208,197,240,197,171,198,249,198,104,199,196,199,128,200
3511 REM *****
3521 DATA69,82,197,69,76,79,65,196,66,83,65,86,197,82,65,78,71,197,84,82,83,197
3531 DATA65,80,80,69,78,196,68,73,83,75,78,65,77,197,86,73,69,215,0,0,0,0,0,0
3541 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3551 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3561 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3571 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3581 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3591 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3601 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3611 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3621 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3631 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3641 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3651 DATA0,0,0,0,0,0,0,0,0,0,85,49,58,53,44,48,44,0,66,45,80,58,51,44,48,0,85
3661 DATA49,58,51,44,48,44,49,56,44,48,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3671 REM *****
3681 DATA169,4,133,198,169,147,141,119,2,169,82,141,120,2,169,213,141,121,2,169
3691 DATA13,141,122,2,76,122,196,169,0,133,10,32,212,225,169,8,133,186,169,1
3701 DATA138,185,76,111,225,41,207,133,170,36,170,80,3,24,105,9,41,15,36,32,212
3711 DATA255,164,183,136,177,187,32,227,197,133,168,136,177,187,32,227,197,10
3721 DATA10,10,10,24,101,168,133,168,136,177,187,32,227,197,133,169,136,177,187
3731 DATA32,227,197,10,10,10,10,24,101,169,133,169,136,177,187,201,44,240,5,162
3741 DATA11,76,139,227,136,177,187,32,227,197,133,251,136,177,187,32,227,197
3751 DATA10,10,10,24,101,251,133,251,136,177,187,32,227,197,133,252,136,177
3761 DATA187,32,227,197,10,10,10,24,101,252,133,252,136,177,187,201,44,240
3771 DATA5,162,11,76,139,227,234,132,183,169,65,162,8,160,1,32,186,255,32,192
3781 DATA255,162,85,32,201,255,165,251,32,210,255,165,252,32,210,255,160,0,177
3791 DATA251,32,210,255,230,251,208,2,230,252,165,251,197,168,208,239,165,252
3801 DATA197,169,208,233,32,204,255,169,65,32,195,255,76,140,196,32,212,225,169
3811 DATA67,162,8,160,0,32,186,255,32,192,255,162,67,32,198,255,32,207,255,133
3821 DATA252,32,207,255,133,253,166,252,234,32,205,189,169,13,32,210,255,230
3831 DATA252,208,2,230,253,32,207,255,32,183,255,240,242,32,204,255,169,67,32
3841 DATA195,255,166,252,165,253,32,205,189,169,13,32,210,255,76,140,196,32,235
3851 DATA183,134,252,32,59,182,169,67,162,8,160,5,32,186,255,169,1,162,58,160
3861 DATA160,32,189,255,32,192,255,162,72,32,201,255,169,224,160,207,32,30,171
3871 DATA166,20,169,0,32,205,189,169,44,32,210,255,166,252,169,0,32,205,189,169
3881 DATA13,32,210,255,32,204,255,162,67,32,198,255,162,0,134,170,169,32,210
3891 DATA255,32,207,255,170,169,0,32,200,196,173,141,2,208,251,230,170,208,233
3901 DATA32,204,255,169,67,32,195,255,76,140,196,32,212,225,165,45,56,233,2,170
3911 DATA165,46,233,0,169,169,8,133,186,169,0,32,213,255,32,51,165,56,165,45
3921 DATA233,2,133,87,165,46,233,0,133,88,160,0,177,87,208,29,200,177,87,208
3931 DATA24,165,87,24,185,2,133,45,133,47,133,49,165,88,105,0,133,46,133,48,133
3941 DATA50,76,140,196,160,0,177,87,133,89,200,177,87,133,88,165,89,133,87,24
3951 DATA144,203,32,212,225,162,17,169,160,157,64,3,202,16,250,164,183,240,10
3961 DATA136,177,187,153,64,3,192,0,208,246,32,59,192,162,72,32,201,255,169,73
3971 DATA32,210,255,32,204,255,169,67,162,8,160,3,32,186,255,169,1,162,58,160
3981 DATA160,32,189,255,32,192,255,162,72,32,201,255,169,240,160,207,32,30,171
3991 DATA32,204,255,162,67,32,198,255,162,0,32,207,255,157,0,160,232,208,247
4001 DATA32,204,255,162,17,189,64,3,157,144,160,202,16,247,162,72,32,201,255
4011 DATA169,232,160,207,32,30,171,32,204,255,162,67,32,201,255,165,1,41,1,254
4021 DATA133,1,162,0,189,0,160,32,210,255,232,208,247,32,204,255,165,1,9,1,133
4031 DATA1,169,50,141,241,207,162,72,32,201,255,169,240,160,207,32,30,171,32
4041 DATA204,255,169,67,32,195,255,169,49,141,241,207,76,0,196,32,212,225,163
4051 DATA8,133,186,169,73,133,184,32,192,255,32,65,192,162,72,32,198,255,32,207
4061 DATA255,133,252,32,207,255,24,101,252,133,252,32,207,255,32,183,255,201
4071 DATA64,208,246,32,204,255,165,252,201,96,240,8,32,231,255,162,4,76,139,227
4081 DATA162,73,32,198,255,32,207,255,32,207,255,32,207,255,170,32,207,255,224
4091 DATA0,208,6,201,0,208,2,240,106,169,0,133,252,169,13,32,210,255,32,225,255
4101 DATA240,92,32,207,255,170,32,207,255,32,205,189,169,32,32,210,255,32,207
4111 DATA255,240,205,201,34,208,8,72,165,252,73,1,133,252,104,201,0,48,5,32,210
4121 DATA255,208,230,164,252,208,247,72,169,0,133,168,169,205,133,169,160,0,104
4131 DATA41,127,170,240,12,200,208,2,230,169,177,168,16,247,202,208,244,200,208
4141 DATA2,230,169,177,168,72,41,127,32,210,255,104,16,240,48,179,32,204,255
4151 DATA169,73,32,195,255,76,248,168
READY.

```

\$A0FF, nella Ram che doppia la Rom del Basic.

A partire da \$A090 è posto il nuovo nome. Dopo aver spedito il comando "B:P:3,0" per inizializzare il puntatore alla Ram del driver, inizia il trasferimento opposto, dal 64 all'area precedentemente allocata. Per fare questo è necessario sganziare la Rom del Basic per accedere in lettura alla Ram nascosta. Ciò avviene modificando il bit 0 del byte 1. Terminato questo trasferimento si "riaggancia" il Basic e si ricopia il tutto nel settore 0 della traccia 18. Si spedisce il comando "U2:3,0,18,0"

```

LIST
10 REM *****
20 REM QUESTO PROGRAMMA E' IN MEMORIA
30 REM *****
READY.
APPEND "PROVA"
SEARCHING FOR PROVA
LOADING
READY.
LIST
10 REM *****
20 REM QUESTO PROGRAMMA E' IN MEMORIA
30 REM *****
40 REM *****
50 REM QUESTO PROGRAMMA E' SU DISCO
60 REM *****
READY.

```

Foto 3 - Col comando APPEND è possibile saldare due programmi.

```

READY.
LIST
10 REM *****
20 REM QUESTO PROGRAMMA E' IN MEMORIA
30 REM *****
READY.
VIEW "PROVA"
10 REM *****
20 REM QUESTO PROGRAMMA E' SU DISCO
30 REM *****
READY.

```

Foto 4 - Il comando VIEW si usa per visionare un programma che sta su un dischetto.

ottenuto modificando in 2 l'uno di U1 stivato alla locazione \$CFF1. Dopo aver chiuso tutti i file, non resta che riconvertire in U1 l'U2 di cui sopra e saltare a \$C400 dove è implementata l'istruzione INIT.

Il comando VIEW (listato 9), dopo aver letto il nome del programma da listare, apre un file di tipo PRG e il canale di comunicazione col disco di tipo input per leggere un eventuale messaggio di errore. Ciò può avvenire se il programma non è presente sul disco. Subito dopo inizia la visualizzazione del programma prelevando byte dopo byte tutte le linee fino a quando non se ne incontra una (l'ultima del programma) con il link a (0,0). Come per la routine di LIST visita sul numero scorso per ogni byte maggiore di 127 incontrato al di fuori degli apici, è necessaria la dovuta detokenizzazione, la riconversione in caratteri del comando Basic tokenizzato. La tabella di tutte le istruzioni Basic (vecchie e nuove) è locata a partire dal byte \$CD00, nell'area occupata dall'ADP Basic. Terminata la visualizzazione sono chiusi i vari canali aperti, e lasciato il controllo all'istruzione DATA.