

Continuano a pervenire in redazione numerosi programmi per il pocket della Sharp, di interesse sempre crescente a causa delle continue scoperte effettuate da noi e dai lettori nell'ambito delle possibilità operative del PC-1500. La Sharp, nel frattempo, ha immesso recentemente sul mercato la nuova PC-1500A, la quale si differenzia dalla prima per una capacità RAM da 8K standard e per alcune innovazioni sul Basic.

I programmi di questo mese ci giungono da Ferrara e da Parma e consistono l'uno in una sorta di "computer-art" con il 1500, e l'altro in routine utili da impiegare per una gestione più ampia del display.

3) ottenere ingrandimenti o riduzioni in scala della figura; i fattori scala possono essere diversi per l'asse X e per l'asse Y, permettendo così un cambio delle proporzioni delle figure.

È inoltre possibile tracciare linee o posizionare la penna relativamente alla sua posizione precedente e non in riferimento al punto d'origine.

Un programma che utilizzi queste subroutine deve iniziare con l'istruzione GOSUB "I" che serve ad inizializzare alcune variabili e per stabilire l'origine al centro del rotolo di carta, cosa che permette di disegnare più facilmente rispetto ad un si-

GRAPH.PACK.		
5005: REM PACKAGE GRAFICO	5060: IF SK > 10R TK > 1GOSUB "S"	5140: X2=(O*G)+(P*H): Y2=(-O*H)+(P*G)
5009: REM INIZIALIZZAZIONE	5070: IF R < 0GOSUB "R"	5150: O=X2+J: P=Y2+K
5010: "I": GRAPH: GLCURSOR (108, 0): SORGN	5080: LINE (O, P)-(M, N)	5160: X2=(M*G)+(N*H): Y2=(-M*H)+(N*G)
5020: S=1: T=1: RETURN	5090: O=X1: P=Y1: RETURN	5170: M=X2+J: N=Y2+K: RETURN
5029: REM POSIZIONA	5099: REM SCALA	5179: REM P. RELATI VO
5030: "P": O=X: P=Y: RETURN	5100: "S": O=O*S: M=M*S: P=P*T: N=N*T	5180: "PR": O=O+X: P=P+Y: RETURN
5030: REM TRACCIA	5110: RETURN	5189: REM T RELATI VO
5040: "T": M=X: N=Y: X1=X: Y1=Y	5119: REM RUOTA	5190: "TR": X=X+O: Y=Y+P
5050: O=O+Q: P=P+W: M=M+Q: N=N+W	5120: "R": O=O-J: M=M-J: P=P-K: N=N-K	5200: GOSUB "T"
	5130: D=(R*PI)/180: H=SIN D: G= COS D	5210: RETURN

Figura 1 - Listato del programma Graph Pack.

GRAPH PACK

Fabrizio Fesani - Parma

Il programma rappresenta un primo tentativo di sfruttare in maniera più completa le elevate capacità grafiche del plotter CE-150. In realtà si tratta di un insieme di subroutine che permettono di creare sul plotter figure anche abbastanza complesse in modo più semplice che non utilizzando una serie di istruzioni LINE. La vera utilità di queste subroutine sta però nel fatto che, una volta disegnata la figura voluta, esse permettono di:

1) spostare l'intera figura in ogni posizione desiderata sia lungo l'asse X che lungo l'asse Y semplicemente modificando le coordinate del punto d'origine;

2) ruotare la figura sia intorno all'origine sia intorno ad ogni altro punto (di cui basta indicare le coordinate) di un numero prefissato di gradi;

stema con l'origine spostata tutta a sinistra.

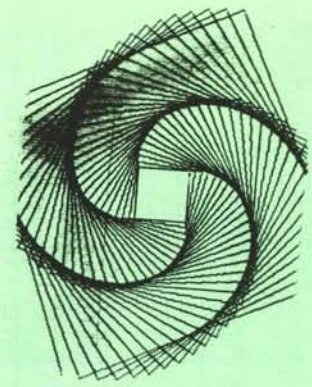
In seguito si deve stabilire il punto da cui la penna inizierà a disegnare, indicandone le coordinate mediante le variabili X e Y, e richiamando la subroutine "P". Per disegnare è sufficiente indicare sempre tramite le variabili X e Y del punto finale della linea da tracciare, e tramite l'istruzione GOSUB "T" andare alla subroutine di tracciamento.

Esempio:

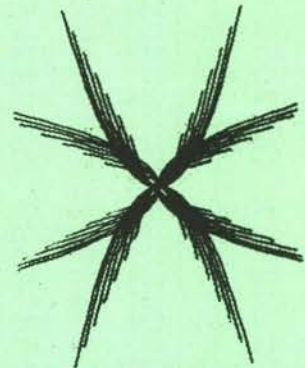
```
10 GOSUB "I"
20 X= 0: Y= 0: GOSUB "P"
30 X= 40: Y= 0: GOSUB "T"
40 X= 40: Y= 40: GOSUB "T"
50 X= 0: Y= 40: GOSUB "T"
60 X= 0: Y= 0: GOSUB "T"
```

In questo modo è possibile disegnare un quadrato con il primo spigolo nell'origine.

Lo spostamento della figura si ottiene mediante traslazione degli assi. Per far ciò



```
10: CLEAR : GOSUB "I"
20: FOR Z=0 TO 200 STEP 5
30: R=Z
40: S=Z/70+.5: T=S
50: X=40: Y=40: GOSUB "P"
51: X=40: Y=-40: GOSUB "T"
52: X=-40: Y=-40: GOSUB "T"
53: X=-40: Y=40: GOSUB "T"
54: X=40: Y=40: GOSUB "T"
70: NEXT Z: END
```



```
80: CLEAR : GOSUB "I"
85: FOR A=10 TO 46 STEP 4: S=A/10: T=S
90: FOR Z=0 TO 270 STEP 90: R=Z
96: IF B>3LET B=0
97: COLOR B
100: FOR I=-20 TO 20
110: X=I
120: Y=X*X/10
125: IF X=-20GOSUB "P"
128: GOSUB "T"
130: NEXT I: B=B+1: NEXT Z
140: NEXT A: END
```

Figura 2 e 3 - Esempi del programma Graph Pack. I printout sono a colori.

È sufficiente attribuire il valore desiderato alle variabili Q (asse X) e W (asse Y). Analogamente per cambiare la scala si deve assegnare il valore desiderato alle variabili S (asse X) e T (asse Y) che rappresentano il coefficiente di ingrandimento o riduzione.

Responsabile della rotazione è la variabile R; attribuendole un valore da 1 a 360 si ottiene la rotazione della figura intorno all'origine del numero di gradi specificato. Se si desidera la rotazione della figura intorno ad un punto qualsiasi diverso dall'origine, se ne devono indicare le coordinate delle variabili J e K.

È inoltre possibile posizionare la penna e tracciare le linee con valori delle coordinate espresse in modo relativo e non assoluto. I valori delle X e delle Y, cioè, indicano lo spostamento dalla posizione precedente e non si riferiscono all'origine. Per il posizionamento relativo si dovrà attribuire il valore desiderato alle variabili X e Y e richiamare a subroutine "PR". Similmente per il tracciamento relativo si richiamerà la subroutine "TR" e le variabili X e Y rappresenteranno le coordinate del punto d'arrivo della linea rispetto al punto di partenza. Con l'uso combinato delle diverse possibilità offerte da queste routine si ottengono effetti molto gradevoli e, a volte, risultati decisamente imprevedibili.

Negli esempi pubblicati si vede quello che si può ottenere dalla rotazione di una figura anche semplice abbinata ad un ingrandimento in scala. Nella figura 3 grazie alla nidificazione di più cicli FOR-NEXT si mostra come si possa disegnare una qualsiasi curva ed ottenere effetti grafici particolari. Va tenuto però presente che, per il corretto funzionamento della routine di rotazione, il PC-1500 deve essere posto nel modo RADIAN.

Come già accennato, le linee 5010 e 5020 provvedono all'inizializzazione; le variabili S e T sono poste uguali a uno in quanto rappresentano il coefficiente di scala. Alla linea 5030 troviamo la routine di posizionamento; le linee 5040 e 5050 calcolano M e N, punti di arrivo della linea da tracciare, tenendo conto anche di eventuali traslazioni degli assi. Alle linee 5060 e 5070 si trovano le condizioni grazie alle quali è possibile richiamare le rispettive subroutine se è stato inserito un fattore di scala o di riduzione. L'uso delle variabili X1 e Y1 nella linea 5090 serve ad impedire che ogni segmento tracciato riparta dall'origine. Nella linea 5100 si trova la routine di riduzione in scala mentre le linee dalla 5120 alla 5170 ricalcolano le coordinate dei singoli punti in caso di rotazione della figura. Infine alla linea 5180 troviamo la subroutine per il posizionamento relativo della penna ed alla linea 5190 la subroutine per il tracciamento relativo.

Essendo etichettati tutti i salti, è possibile cambiare facilmente i numeri di linea della routine a seconda delle esigenze del programma principale.

Routine e note sul display

Davide Gessi - Ferrara

La mappa di memoria del display può essere divisa in due parti: la prima inizia dalla locazione &7000 e termina a &704D, mentre la seconda comincia a &7100 e finisce a &714D. Nella prima sono memorizzate le varie configurazioni delle colonne 0-38 e 78-116, mentre nella seconda le colonne 39-77 e 117-155.

Ogni colonna è divisa in due parti di 4 e 3 punti rispettivamente; i primi quattro vengono memorizzati in un nibble di un byte il cui indirizzo è un numero pari, mentre i restanti tre, con uno spreco di un bit, in un nibble del byte (di indirizzo dispari) successivo. Rimane da scegliere così se il nibble deve essere quello "low" (meno significativo) o "high" (più significativo): le prime 78 colonne (0-77) del display sono memorizzate in un nibble "low", mentre le altre in un nibble "high".

Un esempio servirà a chiarire meglio le cose: si vuole che nella colonna 15 si accendano alcuni punti in modo che il pattern assuma valore 57 (&39). La colonna è memorizzata nella prima parte della memoria: in un nibble di un byte pari bisogna quindi memorizzare 9 (pattern dei primi 4 punti) ed in un nibble del byte successivo 3 (condizione dei restanti 3 punti). I nibble sono ambedue "low" perchè la colonna è

situata nella parte sinistra del display.

Ora bisognerà determinare gli indirizzi: dato che per ogni colonna occorrono due semibyte, e dato che vi sono 15 colonne prima di quella cercata, l'indirizzo sarà:

$$87000 + 80F \times 2 = 8711E (= 28702)$$

Quindi ora è sufficiente digitare POKE 28702,9,3.

In realtà in esecuzione diretta ciò sarebbe fattibile, ma inutile, perchè dopo il POKE appare il prompt e perciò scompare ogni configurazione dal display; comunque quanto detto in precedenza può essere verificato con il semplice programmino:

```
10 POKE 28702,9,3: PRINT
```

Un'operazione di tale genere non ha molta utilità, esistendo già l'istruzione adatta, tuttavia la conoscenza della mappa di memoria del display sarà utile per la realizzazione delle routine in linguaggio macchina di seguito descritto.

Le due routine in figura 4 e 5 permettono lo scroll fine orizzontale con wrap-around del display, nei due sensi. Le quattro locazioni &4000-&4003 sono utilizzate per salvare i dati relativi agli estremi del visualizzatore. Sono state in tale posizione perchè non sono, almeno apparentemente, utilizzate, neanche per la memoria RESERVE. È stata inoltre preferita questa soluzione, in luogo della utilizzazione dello stack, per evitare in casi particolari la saturazione dello stesso.

Queste due routine sono state scritte per

```
4004 - A5 70 4C LD A,704C
      - AE 40 00 LD 4000,A
      - A5 70 4D LD A,704D
      - AE 40 01 LD 4001,A
      - A5 71 4C LD A,714C
      - AE 40 02 LD 4002,A
      - A5 71 4D LD A,714D
      - AE 40 03 LD 4003,A
      - 68 70 LD D,70
      - BE 40 41 CALL 4041
      - 68 71 LD D,71
      - BE 40 41 CALL 4041
      - A5 40 00 LD A,4000
      - AE 71 00 LD 7100,A
      - A5 40 01 LD A,4001
      - AE 71 01 LD 7101,A
      - A5 40 02 LD A,4002
      - F1 RDA
      - AE 70 00 LD 7000,A
      - 45 40 03 LD A,4003
      - F1 RDA
      - AE 70 01 LD 7001,A
      - 9A RET
4041 - 6A 4B LD E,4B
      - FD 28 LD HL,DE
      - 44 INC HL
      - 44 INC HL
      - 25 LD A,(DE)
      - 43 ADEC HL
      - 8B 04 BNZED 04
      - 9A RET
```

Figura 4 - Routine di scroll destro.

```
4004 - A5 71 00 LD A,7100
      - AE 40 00 LD 4000,A
      - A5 71 01 LD A,7101
      - A4 40 01 LD 4001,A
      - A5 70 00 LD A,7000
      - F1 RDA
      - AE 40 02 LD 4002,A
      - A5 70 01 LD A,7001
      - F1 RDA
      - AE 40 03 LD 4003,A
      - 68 70 LD D,70
      - BE 40 41 CALL 4041
      - 68 71 LD D,71
      - BE 40 41 CALL 4041
      - A5 40 00 LD A,4000
      - AE 70 4C LD 704C,A
      - A5 40 01 LD A,4001
      - AE 70 4D LD 704D,A
      - A5 40 02 LD A,4002
      - AE 71 4C LD 714C,A
      - A5 40 03 LD A,4003
      - AE 71 4D LD 714D,A
      - 9A RET
4041 - 6A 02 LD E,02
      - FD 28 LD HL,DE
      - 46 DEC HL
      - 46 DEC HL
      - 65 INC A,(DE)
      - 41 AINC HL
      - 6E 4E CP E,4E
      - 99 06 BJNZ 06
      - 9A RET
```

Figura 5 - Routine di scroll sinistro.

4650 - 48 70 LD H,70	465D - 05 LD A,(HL)
- BE 46 5B CALL 465B	- BD FF XOR FF
- 48 72 LD H,71	- 43 ADEC HL
- BE 46 5B CALL 465B	- 4E FF CP L,FF
- 9A RET	- 99 08 BJNZ 08
465B - 4A 4D LD L,4D	- 9A RET

Figura 6 - Routine di Inverse display.

```

10:*A*WAIT 0:PRINT *SCROLL > *;;RESTORE 200:GOTO 100
20:*B*WAIT 0:PRINT *SCROLL < *;;RESTORE 300:GOTO 100
30:*C*WAIT 0:PRINT *INVERSE *;;RESTORE 400
100:INPUT *START: *;A:D=A
110:CLS :PRINT *START: *;A;
120:READ E:F=A-E
130:READ B:IF B<0THEN C=F-B:POKE D,C/256:POKE D+1,C-256*
    PEEK (D):D=D+1:GOTO 160
140:IF B>255THEN 170
150:POKE D,B
160:D=D+1:GOTO 130
170:D=D-1:BEEP 2:WAIT :PRINT *END: *;D:END
200:DATA 16388:REM Data routine scroll =>
210:DATA 165,112,76,174,64,0,165,112,77,174
220:DATA 64,1,165,113,76,174,64,2,165,113
230:DATA 77,174,64,3,104,112,190,-16449,104
240:DATA 113,190,-16449,165,64,0,174,113,0
250:DATA 165,64,1,174,113,1,165,64,2,241
260:DATA 174,112,0,165,64,3,241,174,112,1
270:DATA 154,106,75,253,40,68,68,37,67,136
280:DATA 4,154,256
300:DATA 16388:REM Data routine scroll <=
310:DATA 165,113,0,174,64,0,165,113,1,174
320:DATA 64,1,165,112,0,241,174,64,2,165
330:DATA 112,1,241,174,64,3,104,112,190,-16449
340:DATA 104,113,190,-16449,165,64,0,174
350:DATA 112,76,165,64,1,174,112,77,165,64
360:DATA 2,174,113,76,165,64,3,174,113,77
370:DATA 154,106,2,253,40,70,70,101,65,110
380:DATA 78,153,6,154,256
400:DATA 18000:REM Data routine Inverse
410:DATA 72,112,190,-18011,72,113,190,-18011
420:DATA 154,74,77,5,189,255,67,78,255,153
430:DATA 8,154,256

```

Figura 7 - Programma di caricamento e rilocazione.

essere chiamate da un programma Basic, e non prevedono quindi il salvataggio nello stack dei registri utilizzati (A.F.DE.HL).


Volendo, è possibile eliminare il wrap-around inserendo un'istruzione AND subito dopo ogni RDA incontrata. Per la routine SCROLL destro occorre AND F0, mentre per l'altra AND 0F; bisognerà inoltre sostituire le istruzioni CALL 4041 con CALL 4045. Per chiamare le routine CALL &4004.

La routine in figura 6 inverte lo stato di tutti i punti del display, ed è utile per rappresentare messaggi in INVERSE-VIDEO.

Il programma BASIC in figura 7 è in grado di caricare e rilocare le routine descritte in qualsiasi punto della memoria. Alla partenza, il programma chiede la prima locazione alla quale dovrà essere caricata la routine; al termine del caricamento, verrà mostrato l'indirizzo dell'ultima locazione occupata.

Questo programma può essere utilizzato per caricare qualsiasi routine, con queste regole:

1 - Il primo DATA deve contenere la locazione di partenza dell'area nella quale la routine è stata scritta, e l'ultimo deve contenere 256 o un numero superiore, per segnalare la fine delle istruzioni.

2 - Le istruzioni vanno inserite normalmente nei DATA, tranne quando vi è un indirizzo a due byte che deve essere rilocato: in tal caso bisogna sostituire ai due byte il numero decimale che risulterebbe dalla combinazione di questi due, cambiato di segno, in modo che la macchina possa distinguerlo dagli altri. 

La mappa di memoria (II parte)

Continuiamo su questo numero la pubblicazione della mappa di memoria del PC-1500, iniziata sul numero scorso. In particolare prenderemo in esame gli indirizzi di partenza di alcune routine utili del sistema operativo.

- 8000 - 9FFF 8K ROM per le espansioni e periferiche
- A000 - BFFF 8K ROM del printer-plotter
- C000 - FFFF ROM 16K del Sistema Operativo (routine)
- D0D2 Comparazione tra valori numerici
- D0F9 Comparazione fra stringhe
- D2EA Ricerca del numero di linea di programma
- D461 Ricerca l'indirizzo di una variabile
- D925 Concatenazione stringhe
- D9B1 CHR\$
- D9CF STR\$
- D9D7 VAL
- D9DD ASC se C=&60, LEN se C=&64
- D9F3 RIGHTS, MIDS, LEFTS
- E243 Scansione della tastiera (attende un carattere)
- E33F Auto Power Off
- E42C Scansione tastiera
- E8CA Visualizza il contenuto del display buffer
- ED00 Visualizza n caratteri sul display dalla posizione attuale del cursore
- ED3B Visualizza n caratteri sul display dalla posizione cursore = 0
- ED4D Visualizza un carattere sul display e incrementa il cursore

- ED57 Visualizza un carattere sul display senza incrementare il cursore
- ED95 Conversione HEX-DEC
- EDEF Visualizza una colonna grafica
- EFB6 X-Y
- EFBA X+Y
- F01A X*Y
- F084 X/Y
- F0E9 SQR X
- F161 LN X
- F165 LOG X
- F1CB EXP X
- F1D4 10 X
- F391 COS X
- F39E TAN X
- F3A2 SIN X
- F492 ACS X
- F496 ATN X
- F49A ASN X
- F531 DEG X
- F564 DMS X
- F597 ABS X
- F59D SGN X
- F5BE INT X
- F89C X Y
- FF00-FFF6 Vettori per salti e chiamate
- FFF8-FFF9 Indirizzo di partenza della routine MI
- FFFA-FFFB Indirizzo di partenza del Timer
- FFFC-FFFD Indirizzo di partenza della routine NMI
- FFFE-FFFF Indirizzo di partenza della routine RESET

L'ADVANCE 86A /START PUO' AIUTARLA A RADDOPPIARE LE SUE VENDITE DI PC

PERCHÈ È L'UNICO VERO PC 16 BIT A SOLO 1.400.000*

CON MICROPROCESSORE 8086 E CON

- 128 O 256KB DI MEMORIA UTENTE CON CONTROLLO DI PARITA'

- 40KB DI MEMORIA ROM

- MEMORIA GRAFICA A 16 COLORI

- ALTOPARLANTE INCORPORATO

- HARD E SOFTWARE 100% IBM/PC COMPATIBILE

- DIAGNOSTICA, BASIC E SISTEMA OPERATIVO PER CASSETTA CONTENUTO NELLA ROM

- IL BASIC COMPRESO

- UN SET DI 256 CARATTERI IN ROM

- VISUALIZZAZIONE TV, RGB E MONITOR COMP/SYNC A COLORI O MONOCROMATICO

- COMPLETA GESTIONE DEL VIDEO

- QUATTRO PAGINE DI VIDEO

- TESTO 80x25 O 40x25

- RISOLUZIONE GRAFICA 300x200 O 640x200

- PORTA PER CASSETTA, PER PENNA OTTICA, PER PENNA GIOCHI (JOYSTICK) E CON INTERFACCIA CENTRONICS

IN PIU' L'ADVANCE 86A /START PUO' UTILIZZARE STAMPANTI DI QUALUNQUE TIPO PURCHE' UTILIZZINO INTERFACCIA CENTRONICS; E' ESPANDIBILE DALL'UTENTE NEL MODELLO CON 2 FLOPPY DA 320KB CIASCUNO O NEL MODELLO CON HARD DISK DA 10MB; PUO' ESSERE CORREDATO DI COPROCESSORE ARITMETICO 8087; UTILIZZA CUSTOM CHIPS ED ELETTRONICA DELLA FERRANTI INSTRUMENTS E DRIVES DELLA SHUGART ASSOCIATES.

QUANDO DI UN PC SI PUO' DIRE TUTTO QUESTO NON E' NECESSARIO AGGIUNGERE ALTRO.



DISTRIBUTORE ESCLUSIVO

CONDOR INFORMATICS ITALIA SRL
VIA GRANCINI 8
20145 MILANO
TEL. 02/43.45.62-49.87.549-49.87.713

Chiunque desideri avere informazioni su un'eventuale concessione di vendita può telefonare o restituire questo tagliando.

NOME _____

SOCIETA' _____

INDIRIZZO _____

CITTA' _____

TEL. _____

MC

*IN FUNZIONE DEL CAMBIO DEL DOLLARO