



Seconda parte

# Grafica 640x200: Drawto & HardCopy

■ Secondo appuntamento con la «misteriosa» grafica 640x200 dell'ancor più «misterioso» Commodore 128. Questo mese è la volta di due utility grafiche praticamente indispensabili, la prima per tracciare linee tra due punti, la seconda per ottenere una copia su carta dei vostri output grafici di altissima risoluzione ■

## Funzione DRAWTO

Il nome di questa prima funzione indica il fatto che non si tratta di tracciamento di linea tra due punti indicati, ma più semplicemente tra l'ultimo punto tracciato precedentemente e il punto indicato. Come dire che per tracciare una linea tra il punto di coordinate (20,30) e il punto (200,100) occorre prima effettuare un plot del primo e successivamente dare un comando di DRAWTO fino a (200,100).

Questo apparente appesantimento di gestione viene immediatamente riappagato quando si tratta di disegnare (e capita molto spesso) linee sequenziali ovvero quando abbiamo che l'ultimo punto della linea A è il primo punto della linea B e così via. Ad esempio quando dobbiamo tracciare un rettangolo, un cerchio o una funzione qualsiasi per approssimazione di punti non troppo vicini l'un l'altro. Operativamente parlando, dunque, per disegnare un rettangolo di vertici (10,10) (10,100) (100,100) e (100,10) con una normale funzione di tracciamento linee dovremmo scrivere:

```
DRAW 10, 10, 10, 100
DRAW 10, 100, 100, 100
DRAW 100, 100, 100, 10
DRAW 100, 10, 10, 10
```

in cui, notiamo, la seconda coordinata di ogni draw è uguale alla prima del successivo comando. Con una funzione Drawto scriveremmo:

```
PLOT 10, 10
DRAWNT0 100, 100
```

```
DRAWNT0 10, 100
DRAWTO 100, 10
DRAWTO 10, 10
```

## L'implementazione sul 128

A dire il vero, la scelta circa il DRAWTO invece che una semplice DRAW sul Commodore 128 è stata forzata anche dal fatto che tale macchina, quando si chiama una SYS non accetta più di tre parametri di 8 bit l'uno da assegnare rispettivamente ai registri A, X, Y. Come abbiamo già fatto lo scorso mese per il comando PLOT, la SYS relativa al DRAWTO sarà seguita dal valore di ascissa, suddivisa in parte bassa e parte alta, e dal valore di ordinata.

Il listato 1 contiene, sottoforma di linee DATA, i due programmi in linguaggio macchina di questo mese. Va fatto notare che per eseguire un DRAWTO, c'è bisogno (ovviamente) dei programmi pubblicati il mese scorso che servivano per accedere alla pagina grafica, per ripulirla, per tracciare punti e per tornare in modo testo... quindi prima di far partire il listato 1 bisogna avere in memoria le suddette funzioni. Fatto questo, per tracciare una linea dal punto (A, B) al punto (C, D) scriveremo:

```
SYS 4928, A and 255, A / 256, B
SYS 5200, C and 255, C / 256, D
```

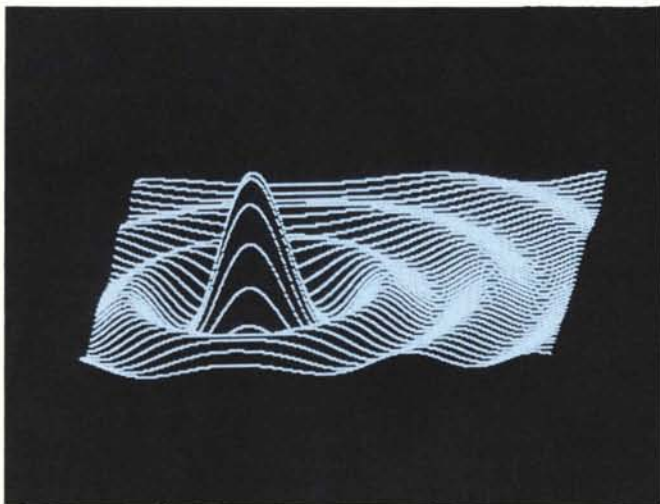
dove la prima SYS è il PLOT del punto iniziale e la seconda il DRAWTO sino al punto finale. Ricordiamo

che l'espressione «X and 255» restituisce la parte bassa della X (gli 8 bit meno significativi), mentre «X/256» restituisce la parte alta (gli 8 bit più significativi).

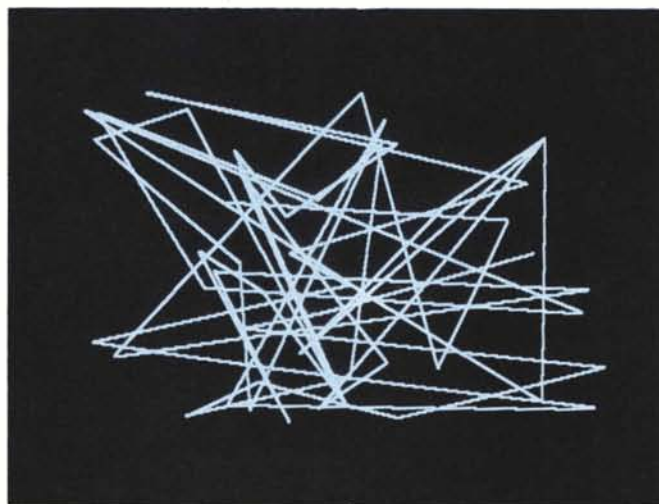
Nel corpo della funzione nessun controllo è effettuato circa la validità del punto di arrivo, ovvero se tracciamo una linea da (0,0) a (100,250), quindi con punto di arrivo fuori campo, apparirà sul video grafico solo la porzione di retta visibile (e con la giusta angolazione). Dato che i tre parametri della SYS sono ad otto bit e per la Y, che normalmente varia tra 0 e 199, è utilizzato un solo parametro, se tentiamo di tracciare una linea la cui Y eccede il limite di 255, il 128 segnalerà un errore ILLEGAL QUANTITY (è la SYS stessa a provocarlo, la routine non c'entra niente).

L'algoritmo di tracciamento linea è il ben noto Bresenham ed ha la particolarità di non richiedere né moltiplicazioni né divisioni né qualsiasi calcolo in virgola mobile qualunque linea si tracci, quindi è particolarmente indicato per essere trascritto in un linguaggio macchina come quello del 6502 che offre ben pochi strumenti di calcolo. Praticamente tale algoritmo traccia il primo punto, si calcola il segno dell'andamento verso il punto di arrivo e poi incrementa la x o la y a seconda di alcuni contatori interni, tracciando i punti via via calcolati fino al punto finale. Di per se stesso è molto veloce anche se a causa della strana architettura del 128 (memoria video separata e indirizzabile indirettamente tramite





1



3

alcuni registri anch'essi indirizzabili indirettamente) non sembra essere tale. Fatto funzionare in modo FAST riesce ad essere abbastanza accettabile.

Tutta la routine di DRAWTO è mostrata nel listato 2: i lettori più capaci potranno eventualmente provare a migliorarla, non senza comunicarci i loro risultati, scrivendo a questa rubrica (vedi riquadro a pagina 171).

Infine, i listati 5 e 6 sono un piccolo esempio di applicazioni "lineeeree", la prima solo orizzontali e verticali, la seconda con tutte le angolazioni. Date ovviamente il run dopo aver caricato la parte di codice del numero scorso e quella di listato 1. Per fermare l'esecuzione dei due demo basta premere qualsiasi tasto, se trattasi della freccetta a sinistra otterremo un hard copy (stampante collegata, accesa e rispondente ai requisiti indicati qui di seguito).

### Hard Copy 640 x 200

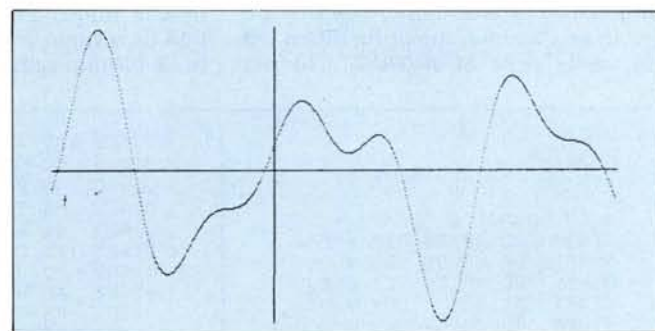
Prima di passare alla routine di HardCopy mostrata nei listati 3 e 4 avvertiamo i lettori che tale programma non gira sulla stampante MPS 803 (e famiglia: 801, 1515, 1525 ecc.) dato che queste hanno una risoluzione di 480 punti per riga e la grafica del 128 ne mette a disposizione 640. Per i nostri esperimenti abbiamo usato una stampante STAR NL-10 dotata di interfaccia Commodore la quale riesce a stampare fino a 1920 punti per linea. Noi l'abbiamo usata in modo 1280 punti in modo da ottenere una stampa quanto più proporzionata possibile al rapporto altezza/larghezza dell'immagine sul monitor.

Tale stampante, per accedere al modo grafico necessita di una serie di caratteri di controllo che settano la spaziatura verticale (per far combaciare le linee di stampa), ed indicano il tipo di risoluzione e il numero di pixel grafici

Figura 1 - Output del programma di listato 7.

Figura 3 - Output del programma di listato 6.

Figura 2  
HardCopy del  
programma Math  
Pack 128  
aggiornato.▶



```

10 FORI=5200T05646
20 READ I
30 POKE I, I: NEXT
40 REM *****
50 REM * PROGRAMMI DRAWTO E HCOPI PER C 128 IN *
60 REM * GRAFICA 640 X 200. MCMICROCOMPUTER 61 *
70 REM * ----- *
80 REM * (C) 1987 ADP-SOFTWARE *
90 REM * ----- *
95 REM *****
100 DATA 141, 11, 11, 142, 12, 11, 140, 13, 11, 162, 2, 189, 0, 11, 157
110 DATA 8, 11, 202, 16, 247, 169, 255, 141, 20, 11, 141, 21, 11, 56, 173
120 DATA 8, 11, 237, 11, 11, 141, 14, 11, 173, 9, 11, 237, 12, 11, 141, 15
130 DATA 11, 176, 29, 169, 1, 141, 20, 11, 169, 0, 141, 21, 11, 56, 173
140 DATA 11, 11, 237, 8, 11, 141, 14, 11, 173, 12, 11, 237, 9, 11, 141, 15
150 DATA 11, 169, 255, 141, 16, 11, 56, 173, 10, 11, 237, 13, 11, 141, 17
160 DATA 11, 176, 15, 169, 1, 141, 16, 11, 56, 173, 13, 11, 237, 10, 11
170 DATA 141, 17, 11, 162, 0, 142, 18, 11, 142, 19, 11, 173, 14, 11, 208
180 DATA 13, 173, 15, 11, 208, 8, 162, 255, 142, 18, 11, 142, 19, 11, 173
190 DATA 8, 11, 141, 0, 11, 174, 9, 11, 142, 1, 11, 172, 10, 11, 140, 2, 11
200 DATA 32, 76, 19, 173, 8, 11, 205, 11, 11, 208, 17, 173, 9, 11, 205, 12
210 DATA 11, 208, 9, 173, 10, 11, 205, 13, 11, 208, 1, 96, 173, 19, 11, 48
220 DATA 40, 24, 173, 8, 11, 109, 20, 11, 141, 8, 11, 173, 9, 11, 109, 21
230 DATA 11, 141, 9, 11, 56, 173, 18, 11, 237, 17, 11, 141, 18, 11, 173
240 DATA 19, 11, 233, 0, 141, 19, 11, 24, 144, 165, 24, 173, 10, 11, 109
250 DATA 16, 11, 141, 10, 11, 24, 173, 18, 11, 109, 14, 11, 141, 18, 11
260 DATA 173, 19, 11, 109, 15, 11, 141, 19, 11, 24, 144, 133, 234, 234
270 DATA 234, 234, 234, 234, 234, 234, 234, 234, 234, 234, 234, 162, 18
280 DATA 169, 0, 32, 204, 205, 162, 19, 32, 204, 205, 162, 25, 162, 0, 134
290 DATA 253, 169, 12, 133, 254, 32, 216, 205, 162, 0, 129, 253, 230, 253
300 DATA 208, 2, 230, 254, 165, 253, 201, 128, 208, 237, 165, 254, 201
310 DATA 14, 208, 231, 32, 160, 21, 136, 208, 217, 96, 234, 234, 234, 2
320 DATA 128, 76, 27, 8, 65, 27, 138, 72, 152, 72, 169, 4, 162, 4, 160, 0
330 DATA 32, 186, 255, 152, 32, 189, 255, 32, 192, 255, 162, 4, 32, 201
340 DATA 255, 162, 7, 189, 152, 21, 32, 210, 255, 202, 208, 247, 162, 0
350 DATA 160, 8, 30, 0, 12, 38, 250, 30, 80, 12, 38, 250, 30, 160, 12, 38
360 DATA 250, 30, 240, 12, 38, 250, 30, 64, 13, 38, 250, 30, 144, 13, 38
370 DATA 250, 30, 224, 13, 38, 250, 30, 48, 14, 38, 250, 165, 250, 32, 210
380 DATA 255, 136, 208, 208, 232, 224, 80, 208, 201, 169, 13, 32, 210
390 DATA 255, 32, 204, 255, 169, 4, 32, 195, 255, 104, 168, 104, 170, 96

```

Listato 1



# 128 da zero

per linea. Per semplicità, inoltre, è stato utilizzato il modo grafico 8 pixel che permette una più veloce trasformazione dal modo di memorizzare i pixel da parte del 128 al modo di stamparli da parte della NL-10.

In definitiva i codici inviati (ve li indichiamo in modo da poter facilmente adattare la routine ad altre stampanti, fateci sapere...) sono un ESC+«A»+CHRS(6) per la spaziatura e un ESC+«L»+CHRS(128)+CHRS(2) per indicare le specifiche grafiche.

I listati 3 e 4, come detto, eseguono l'hard copy: commentiamoli brevemente. La prima parte trasferisce l'immagine in memoria, un venticinquesimo per volta (pari a una linea di 8 pixel) in una zona di memoria libera del 128, nelle celle \$C00-\$E80. Ciò per

semplificare il più possibile la faccenda. Ad ogni iterazione è invocata la routine di stampa all'indirizzo 15A0. Questa, dopo aver salvato nello stack i registri x e y, apre un file di stampa. Per fare ciò utilizza 4 routine del sistema operativo atte allo scopo (si utilizzano per qualsiasi tipo di file). La prima, locata all'indirizzo \$FFBA permette di settare numero file, numero periferica e indirizzo secondario, nel nostro caso la tripla 4,4,0. La seconda routine, \$FFBD, si usa per indicare il nome del file, mettendo nel registro A la lunghezza del nome e dei registri x e y, come sempre spezzato in parte alta e parte bassa, la locazione in memoria di tale nome. Dal momento che noi non dovevamo indicare alcun nome di file è bastato settare pari a zero il con-

tenuto dell'accumulatore. La routine di \$FFCO esegue la open vera e propria riferendosi ai valori precedentemente settati con le due routine. Infine, con \$FFC9, dopo aver indicato nel registro x a quale file ci riferiamo, effettuiamo una sorta di CMD 4 con la quale stabiliamo che da questo momento in poi i caratteri inviati devono arrivare alla stampante.

Per inviare caratteri si usa la routine \$FFD2, caricando nell'accumulatore il codice ASCII di questi. Il ciclo che segue le operazioni di open serve per inviare i 7 codici di controllo di cui sopra. E si sono stati mappati a partire dall'indirizzo \$1598 (in testa alla routine) ed è lì che si dovrà mettere le mani per modificare tale sequenza. Importante: i caratteri sono messi in memoria al contrario per semplificare il loop. In altre parole il primo ESC è all'indirizzo \$159F l'ultimo carattere di controllo a \$1599.

Segue la vera e propria trasformazione di formato e relativo invio di dati da stampare. Terminata la linea si manda un return (\$0D) per far avanzare la carta e si chiude il canale e il file con la stampante. Tutto ciò 25 volte come da listato 3. RTS.

## Listati 7 e 8

Il listato 7 è un ulteriore esempio: traccia la funzione grafica mostrata in figura 1. Si tratta della ben nota  $Z = \sin(Q)/Q$  dove Q è la distanza dall'origine degli assi, misurata sul piano x-y. Terminato il disegno (data la complessità dei calcoli e soprattutto il numero delle iterazioni necessarie

### Listato 2

```

MONITOR
PC SR AC XR YR SP
;
B000 00 00 00 00 FB
01450 8D 0B 0B STA $0B0B
01453 8E 0C 0B STX $0B0C
01456 8C 0D 0B STY $0B0D
01459 A2 02 LDX #$02
0145B BD 00 0B LDA $0B00,X
0145E 9D 08 0B STA $0B08,X
01461 CA DEX
01462 10 F7 BPL $145B
01464 A9 FF LDA #$FF
01466 8D 14 0B STA $0B14
01469 8D 15 0B STA $0B15
0146C 38 SEC
0146D AD 08 0B LDA $0B08
01470 ED 0B 0B SBC $0B0B
01473 8D 0E 0B STA $0B0E
01476 AD 09 0B LDA $0B09
01479 ED 0C 0B SBC $0B0C
0147C 8D 0F 0B STA $0B0F
0147F B0 1D BCS $149E
01481 A9 01 LDA #$01
01483 8D 14 0B STA $0B14
01486 A9 00 LDA #$00
01488 8D 15 0B STA $0B15
0148B 38 SEC
0148C AD 0B 0B LDA $0B0B
0148F ED 08 0B SBC $0B08
01492 8D 0E 0B STA $0B0E
01495 AD 0C 0B LDA $0B0C
01498 ED 09 0B SBC $0B09
0149B 8D 0F 0B STA $0B0F
0149E A9 FF LDA #$FF
014A0 8D 10 0B STA $0B10
014A3 38 SEC
014A4 AD 0A 0B LDA $0B0A
014A7 ED 0D 0B SBC $0B0D
014AA 8D 11 0B STA $0B11
014AD B0 0F BCS $14BE
014AF A9 01 LDA #$01
014B1 8D 10 0B STA $0B10
014B4 38 SEC
014B5 AD 0D 0B LDA $0B0D
014B8 ED 0A 0B SBC $0B0A
014BB 8D 11 0B STA $0B11
014BE A2 00 LDX #$00
014C0 8E 12 0B STX $0B12
014C3 8E 13 0B STX $0B13
014C6 AD 0E 0B LDA $0B0E
014C9 D0 0D BNE $14D8

```

```

014CB AD 0F 0B LDA $0B0F
014CE D0 08 BNE $14D8
014D0 A2 FF LDX #$FF
014D2 8E 12 0B STX $0B12
014D5 8E 13 0B STX $0B13
014D8 AD 08 0B LDA $0B08
014DB 8D 00 0B STA $0B00
014DE AE 09 0B LDX $0B09
014E1 8E 01 0B STX $0B01
014E4 AC 0A 0B LDY $0B0A
014E7 8C 02 0B STY $0B02
014EA 20 4C 13 JSR $134C
014ED AD 08 0B LDA $0B08
014F0 CD 0B 0B CMP $0B0B
014F3 D0 11 BNE $1506
014F5 AD 09 0B LDA $0B09
014F8 CD 0C 0B CMP $0B0C
014FB D0 09 BNE $1506
014FD AD 0A 0B LDA $0B0A
01500 CD 0D 0B CMP $0B0D
01503 D0 01 BNE $1506
01505 60 RTS
01506 AD 13 0B LDA $0B13
01509 30 28 BMI $1533
0150B 18 CLC
0150C AD 08 0B LDA $0B08
0150F 6D 14 0B ADC $0B14
01512 8D 08 0B STA $0B08
01515 AD 09 0B LDA $0B09
01518 6D 15 0B ADC $0B15
0151B 8D 09 0B STA $0B09
0151E 38 SEC
0151F AD 12 0B LDA $0B12
01522 ED 11 0B SBC $0B11
01525 8D 12 0B STA $0B12
01528 AD 13 0B LDA $0B13
0152B E9 00 SBC #$00
0152D 8D 13 0B STA $0B13
01530 18 CLC
01531 90 A5 BCC $14D8
01533 18 CLC
01534 AD 0A 0B LDA $0B0A
01537 6D 10 0B ADC $0B10
0153A 8D 0A 0B STA $0B0A
0153D 18 CLC
0153E AD 12 0B LDA $0B12
01541 6D 0E 0B ADC $0B0E
01544 8D 12 0B STA $0B12
01547 AD 13 0B LDA $0B13
0154A 6D 0F 0B ADC $0B0F
0154D 8D 13 0B STA $0B13
01550 18 CLC
01551 90 85 BCC $14D8

```

### Listato 3

```

01560 A2 12 LDX #$12
01562 A9 00 LDA #$00
01564 20 CC CD JSR $CDCC
01567 A2 13 LDX #$13
01569 20 CC CD JSR $CDCC
0156C A0 19 LDY #$19
0156E A2 00 LDX #$00
01570 86 FD STX $FD
01572 A9 0C LDA #$0C
01574 85 FE STA $FE
01576 20 D8 CD JSR $CDD8
01579 A2 00 LDX #$00
0157B 81 FD STA ($FD,X)
0157D E6 FD INC $FD
0157F D0 02 BNE $1583
01581 E6 FE INC $FE
01583 A5 FD LDA $FD
01585 C9 80 CMP #$80
01587 D0 ED BNE $1576
01589 A5 FE LDA $FE
0158B C9 0E CMP #$0E
0158D D0 E7 BNE $1576
0158F 20 A0 15 JSR $15A0
01592 88 DEY
01593 D0 D9 BNE $156E
01595 60 RTS

```



## Listato 4

```

015A0 8A TXA
015A1 48 PHA
015A2 98 TYA
015A3 48 PHA
015A4 A9 04 LDA #04
015A6 A2 04 LDX #04
015A8 A0 00 LDY #00
015AA 20 BA FF JSR $FFBA
015AD 98 TYA
015AE 20 BD FF JSR $FFBD
015B1 20 C0 FF JSR $FFC0
015B4 A2 04 LDX #04
015B6 20 C9 FF JSR $FFC9
015B9 A2 07 LDX #07
015BB BD 98 15 LDA $1598.X
015BE 20 D2 FF JSR $FFD2
015C1 CA DEX
015C2 D0 F7 BNE $15BB
015C4 A2 00 LDX #00
015C6 A0 08 LDY #08
015C8 1E 00 0C ASL $0C00.X
015CB 26 FA ROL $FA
015CD 1E 50 0C ASL $0C50.X
015D0 26 FA ROL $FA
015D2 1E A0 0C ASL $0CA0.X
015D5 26 FA ROL $FA
015D7 1E F0 0C ASL $0CF0.X
015DA 26 FA ROL $FA
015DC 1E 40 0D ASL $0D40.X
015DF 26 FA ROL $FA
015E1 1E 90 0D ASL $0D90.X
015E4 26 FA ROL $FA
015E6 1E E0 0D ASL $0DE0.X
015E9 26 FA ROL $FA
015EB 1E 30 0E ASL $0E30.X
015EE 26 FA ROL $FA
015F0 A5 FA LDA $FA
015F2 20 D2 FF JSR $FFD2
015F5 88 DEY
015F6 D0 D0 BNE $15C8
015F8 E8 INX
015F9 E0 50 CPX #50
015FB D0 C9 BNE $15C6
015FD A9 0D LDA #0D
015FF 20 D2 FF JSR $FFD2
01602 20 CC FF JSR $FFCC
01605 A9 04 LDA #04
01607 20 C3 FF JSR $FFC3
0160A 68 PLA
0160B A8 TAY
0160C 68 PLA
0160D AA TAX
0160E 60 RTS

```

## Listato 7

```

1 FAST:SYS4864:SYS4886
2 REM *****
3 REM * TRACCIAMENTO FUNZIONE *
4 REM * TRIDIMENSIONALE *
5 REM * Z = SIN(Q)/Q *
6 REM * Q=SQR(X12+Y12) *
7 REM *****
10 DIM A%(640):T=15
20 FOR Y0=-9.5TO19STEP.7
30 FOR X0=-9.5TO19STEP.04
40 Q=SQR(X0*X0+Y0*Y0)
50 Y=INT(SIN(Q)/Q*75*T*1.3)
60 X=INT((X0+9.5)*19.5)+T*.92-14
70 IFX<0ORX>=640THEN 80
75 IF Y>A%(X) THENA%(X)=Y:Y=199-Y:GOSUB120
80 NEXT:T=T+2:NEXT
90 GETA$:IFA$=""THEN90
100 IFA$="-"THEN SYS5472:GOTO90
110 SYS4871:END
120 SYS4928,XAND255,X/256,Y
130 RETURN

```

## Listato 5

```

1 FAST
2 REM *****
3 REM *
4 REM * PROGRAMMA LINEE 1 *
5 REM *
6 REM *****
10 SYS4864:SYS4886
15 X=100:Y=100
20 SYS4928,X AND 255,X/255,Y
30 Y=199*RND(1):GOSUB100
40 X=639*RND(1):GOSUB100
70 GETA$:IFA$=""THEN30
75 IFA$="-"THENSYS5472
80 SYS4871:END
100 SYS5200,X AND 255,X/256,Y
110 RETURN

```

trascorrono alcune decine di minuti) premendo la classica freccettina a sinistra otteniamo l'hard copy su carta.

Infine, il listato 8 mostra le linee da aggiungere al programma math pack 128 presentato il mese scorso, per ottenere l'hard copy e per un tracciamento degli assi cartesiani più rapido. Come al solito, la copia su carta avviene premendo la freccettina quando è presente su video la funzione disegnata. Ovviamente è necessario aggiungere anche le linee data del listato 1 adoperando numeri linea diversi, ad esempio dal 1800 in poi, per non calpestare altre linee del vecchio listato. In figura 2 è presente un output di math pack trasferito su carta.

### Espansioni future

Non è proprio nello stile di MC fare anticipazioni, specialmente quando la notizia non implica una promessa da parte del sottoscritto, ma devo ricono-

## Listato 6

```

1 FAST
2 REM *****
3 REM *
4 REM * PROGRAMMA LINEE 2 *
5 REM *
6 REM *****
10 SYS4864:SYS4886
20 X=100:Y=100:GOSUB120
30 Y=199*RND(1)
40 X=639*RND(1):GOSUB100
70 GETA$:IFA$=""THEN30
75 IFA$="-"THENSYS5472
80 SYS4871:END
100 SYS5200,X AND 255,X/256,Y
110 RETURN
120 SYS4928,X AND 255,X/256,Y
130 RETURN

```

scere, che programmare a colpi di SYS è davvero penoso. Per la verità quanto presentato in questi due articoli è un embrione di ADP BASIC per il 128 che da molto tempo sto maturando dentro di me. Anticipazione per anticipazione vorrei aggiungere che il lavoro ancora non è iniziato quindi non conterei troppo su brevi scadenze: del resto fare una espansione Basic su una macchina così *difficile* come il 128 sicuramente non è facile, specialmente considerato che ancora non se ne sono viste sotto nessuna specie. Questo nuovo ADP BASIC, come intuibile, riguarderà istruzioni grafiche per l'altissima risoluzione, versante per il quale la Commodore davvero non s'è sforzata affatto. Avremo le solite istruzioni per disegnare punti, linee, cerchi, riempimenti di aree, più altre meno diffuse come il clip di aree di schermo o il modo grafico X-OR di punti e linee. Questo per citare qualcosa. Arrivederci al prossimo numero. **MC**

### Inviare i vostri contributi

Chiunque voglia contribuire alla rubrica «128 da zero» proponendo argomentazioni, trucchetti o semplicemente notizie riguardanti il Commodore 128 non deve far altro che inviare il proprio contributo in redazione, possibilmente indicando sulla busta «128 da zero». I lavori più interessanti saranno ospitati su queste pagine ovviamente ricompensando gli autori a seconda sia delle dimensioni che dell'importanza degli elaborati stessi. Non dimenticate (a tal proposito) di indicare sempre chiaro il vostro nome, indirizzo, codice fiscale. Anche il vostro numero di telefono potrebbe esserci utile per potervi eventualmente contattare prima della pubblicazione per chiarimenti. Buon lavoro.

## Listato 8

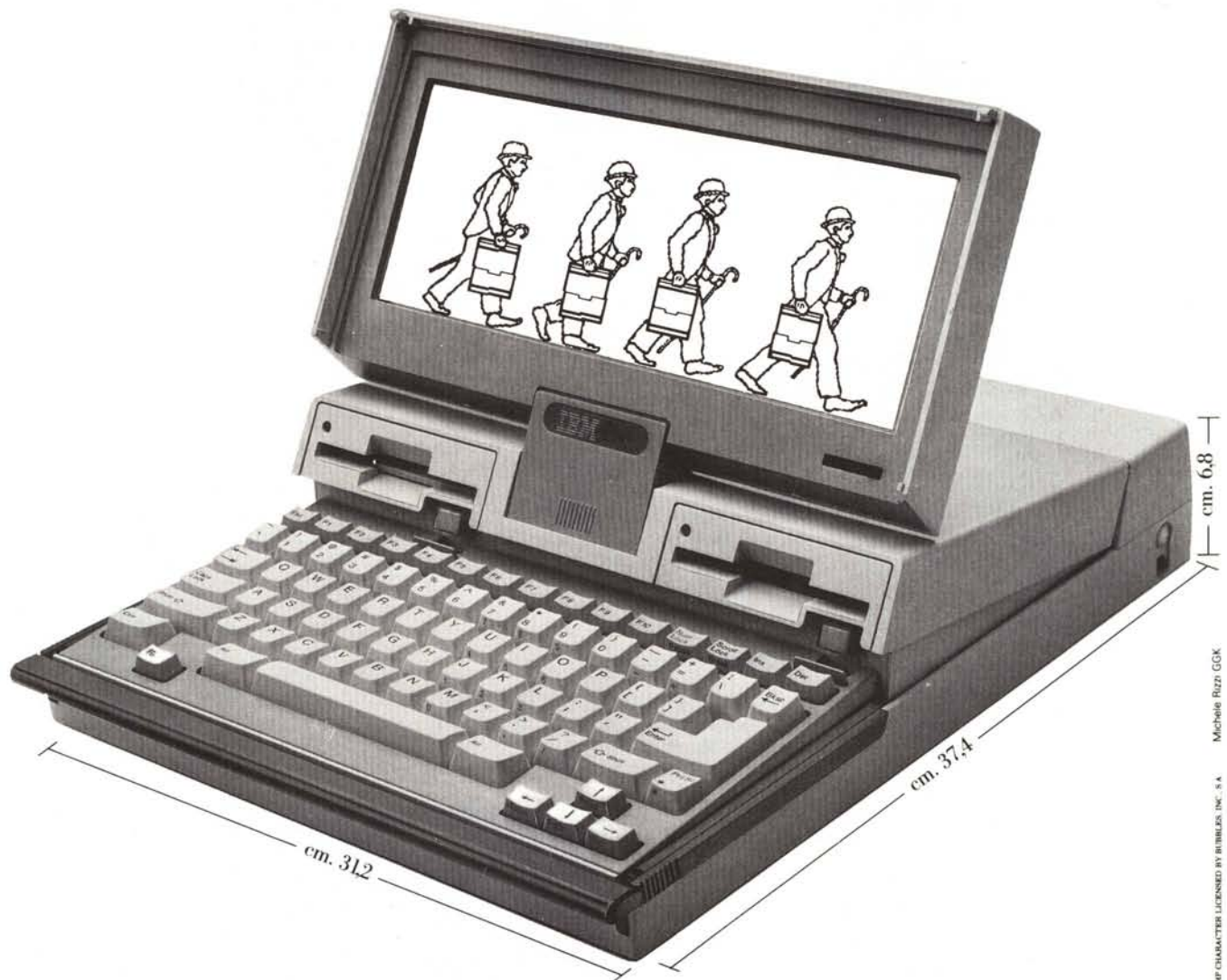
```

10 REM *****
20 REM * LINEE DA AGGIUNGERE AL *
30 REM * PROGRAMMA MATH PACK 128 *
40 REM * PER IL TRACCIAMENTO ASSI *
50 REM * E L'HARD COPY DEL VIDEO *
60 REM *****
730 Y=-Y1*KY:SYS4928,0,0,Y
740 SYS 5200,127,2,Y
760 X=-X1*KX:SYS4928,X AND 255,X/256,0
770 SYS 5200,X AND 255,X/256,199
875 IFZ$="-"THENSYS5472:GOTO870
1605 FORK=5200TO5646:READKK:POKEK, KK:NEXT
1800 REM *****
1810 REM * AGGIUNGERE ANCHE LE LINEE DATA *
1820 REM * DEL LISTATO 1, RENUMERANDOLE A *
1830 REM * A PARTIRE DA 1800. VEDI TESTO *
1840 REM *****

```



# Personal Computer IBM Ventiquattrore



## Piccolo così.

Portatile, leggero, potente, economico, il Personal Computer IBM Ventiquattrore fa parte della grande famiglia dei Personal Computer IBM. Progettato per non stare mai fermo e seguirti dovunque tu vada, il Ventiquattrore IBM è contenuto nelle dimensioni, più piccolo di una valigetta, e pesa poco più di cinque chilogrammi. **Le sue prestazioni**, come quelle di tutti i componenti della famiglia dei Personal Computer IBM, **sono grandi**, perché è la tecnologia IBM a renderlo ricco di novità.

Ha **512 Kb di memoria** e utilizza 2 minidischi ad alta densità di nuova concezione, che contengono ben 720 Kb di informazioni ciascuno. Dispone di uno schermo di tipo grafico a cristalli liquidi e può essere anche collegato, in ufficio o a casa, a un video a colori. Puoi utilizzarlo dappertutto perché, oltre all'alimentazione a rete, possiede anche una batteria ricaricabile. E poiché il personal che porti con te è un po' come il tuo biglietto da visita, il Ventiquattrore IBM è stato disegnato

da Richard Sapper per essere anche bello da vedere. Non meraviglia, quindi, che abbia vinto il 19° Premio SMAU per il design industriale. Se ancora non l'hai fatto, vai a vederlo e provarlo in uno degli **oltre 400 punti vendita e assistenza dei Concessionari IBM Personal Computer (\*)** in tutta Italia (gli indirizzi sono sulle pagine Gialle). Ti stupirai.

\*Puoi anche rivolgerti al tuo rappresentante IBM.