

Una delle caratteristiche peculiari del sistema-pocket Sharp PC-1500 è senza dubbio la stampante plotter CE-150, la quale rappresenta forse il registratore grafico ad elevate prestazioni più economico che esista oggi in commercio.

Il modo più immediato per poter avvicinare a questo accessorio è il classico programma di tracciamento grafico di una funzione: a tale scopo questo mese abbiamo scelto due fra i migliori programmi grafici ricevuti dai lettori, al fine di illustrare le prestazioni del CE-150.

Forse il programma di Brandimarte si presta ad un impiego più generale, essendo in grado di disegnare fino a 6 funzioni contemporaneamente, mentre quello di Nannelli è più indirizzato verso la risoluzione del compito di matematica per l'esame di maturità scientifica.

Di quest'ultimo vogliamo comunque evidenziare l'astuto impiego dell'istruzione LLIST n nel corso del programma, per descrivere, a grafico effettuato, le equazioni delle funzioni disegnate.

Purtroppo, per esigenze tipografiche, gli esempi pubblicati in queste pagine sono stampati in bianco e nero: gli output dei due programmi sono comunque in 4 colori, i quali garantiscono un'ottima e rapida interpretazione dei grafici.

A proposito di leggibilità, vorremmo pregare tutti coloro i quali ci inviano programmi per il PC-1500, di allegare il listato nel formato CSIZE 2, onde evitare errori di lettura del testo stesso.

GRAFICI 1

di Marco Brandimarte - Torino

Si tratta del classico tracciamento di grafici di funzioni reali di una variabile reale che da sempre gira nelle facoltà di Ingegneria, ma in questa versione è portato ai massimi livelli grazie proprio alla presenza del plotter CE-150. Si ha inoltre la possibilità di parametrizzare la funzione così da confrontare vari andamenti della stessa, o addirittura di confrontare funzioni diverse.

Volendo ottenere dei grafici significativi, per prima cosa vengono calcolati il massimo ed il minimo tra i punti considerati della funzione, quindi viene determinato un fattore di scala per le ordinate (S) analogamente a quanto già fatto per le ascisse (R); ciò fa in modo che la curva occupi sempre tutto lo spazio disponibile. Questa di avere gli assi coordinati non isometrici è una delle possibili scelte: con essa, per esempio, tutte le rette passanti per l'origine saranno sempre inclinate di 45° nel grafico, ma ciò comporta che qualsiasi funzione risulterà sempre leggibile. Le successive valutazioni, fatte per valori diversi del parametro, manterranno invece il fattore di scala precedentemente calcolato in modo da poter giudicare eventuali amplificazioni o attenuazioni. È ovvio che per scopi particolari si può parametrizzare anche rispetto alle ascisse. Il confronto di funzioni diverse è possibile scrivendole in linee differenti e sfruttando una capacità del Basic della Sharp che prevede l'istruzione GOTO e

GOSUB <expr>, dove <expr> può essere un'espressione qualsiasi, ma anche semplicemente una variabile, come nel nostro caso (L). I programmi diventano così formalmente poco leggibili, ma senz'altro, una volta "debuggati", sono molto più potenti e permettono una certa interattività con l'utente.

Il funzionamento del programma è intrinsecamente molto semplice: durante la prima fase di input si richiede il numero di intervalli in cui la funzione verrà valutata e graficata, il quale è proporzionale alla precisione desiderata ed anche al tempo di esecuzione; comunque oltre 216 non si può andare perché questa è la massima definizione ottenibile dal plotterino: intorno a $40 \div 50$ è l'ideale nella maggior parte dei casi.

Quindi si richiede se la funzione è parametrizzata, nel qual caso il valore del parametro verrà richiesto ogni volta insieme al

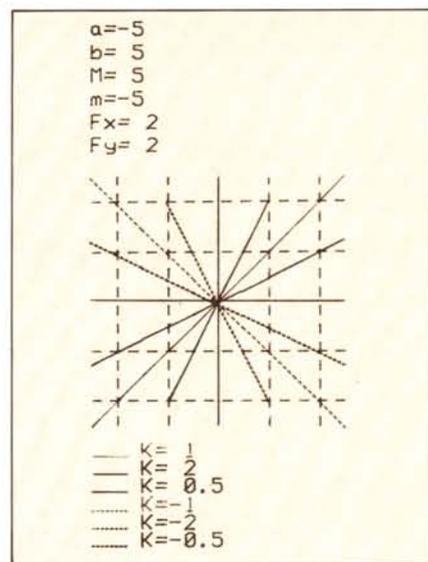


Figura 5 - "Grafici 1" - Coesistenza di più plot della stessa funzione in un unico grafico: $500 F = K \cdot X$ RETURN

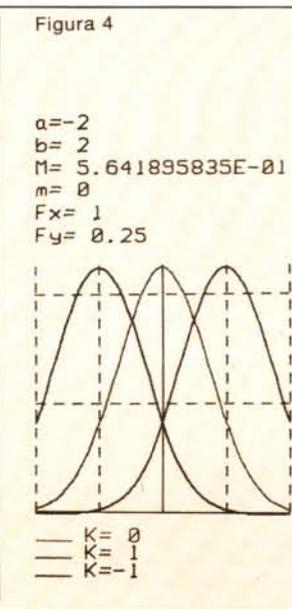
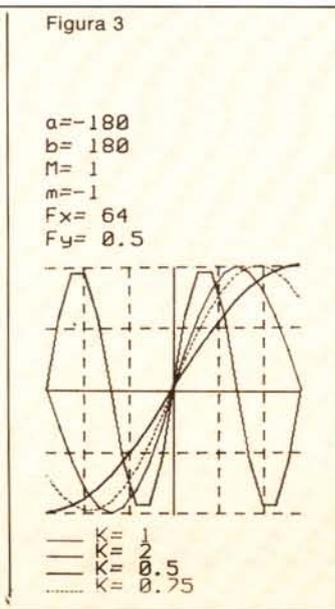
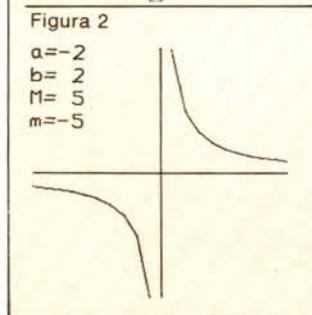
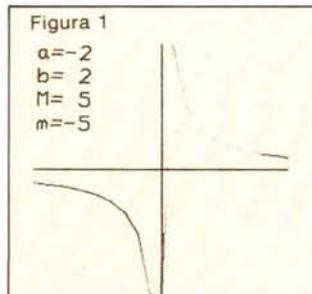


Figura 1 - "Grafici 1" - Output relativo alle seguenti linee: $500 \text{ IF } X=0$ RETURN, $505 F=1/X$: RETURN

Figura 2 - "Grafici 1" - È stato eliminato il segmento indesiderato. La funzione sarà: $500 \text{ IF } X=0 \text{ LET } J=1$: RETURN $505 F=1/X$: IF $P=0 \text{ LET } J=1$ 510 RETURN

Figura 3 - "Grafici 1" - Esempio di curva parametrizzata (numero di intervalli = 20): $500 F = \text{SIN}(K \cdot X)$: RETURN

Figura 4 - "Grafici 1" - Gaussiana con varianza unitaria parametrizzata sul valor medio. Numero di intervalli = 50: $600 F = 1/(\sqrt{\pi}) \cdot \text{EXP}(-X \cdot X/2)$: RETURN

```

10: "A":CLEAR :
TEXT :COLOR 0:
GOSUB 200:
INPUT "Estr. ;
nf.?" ;A:INPUT
"Estr. sup.?" ;
B
12: WAIT 0:PRINT "
PARAMETRICA?":
B$=INKEY$ :IF
B$=""GOTO 12
13: IF B$="S"GOSUB
210
15: INPUT "INTERVA
LLI?" ;H:D=(B-A
)/H:R=(B-A)/21
6:CLS
20: FOR I=0TO H:X=
A+I*D:P=X:
GOSUB L:IF F>M
LET M=F:GOTO 3
0
25: IF F<NLET N=F
30:NEXT I:TEXT :
LPRINT "a=" ;A:
LPRINT "b=" ;B:
LPRINT "m=" ;M:
LPRINT "n=" ;N:
B$=""
35: PRINT "GRIGLIA
?":B$=INKEY$ :
IF B$=""GOTO 3
5
40: IF B$<>"S"GOTO
80
45: IF (ABS A<ABS
B)AND A<>0LET
C=ABS A/2:GOTO
55
50:C=ABS B/2
55:FX=2^(INT ((
LOG C)/LOG 2))
60:IF (ABS N<ABS
M)AND N<>0LET
C=ABS N/2:GOTO
70
65:C=ABS M/2
70:FY=2^(INT ((
LOG C)/LOG 2))
75:LPRINT "Fx=" ;F
X:LPRINT "Fy="
;FY
80:GRAPH :S=(M-N)
/216:IF A>0OR
B<0LET X=0:
GOTO 90
85:X=-A/R
90:GLCURSOR (X,-M
/S):SORGN
100:LINE (-X,0)-(B
/R,0),0:LINE (
0,M/S)-(0,N/S)
105:U=1:IF B$<>"S"
GOTO 150
110:U=0
115:U=U-FX:IF U>=A
THEN LINE (U/R
,M/S)-(U/R,N/S
),7:GOTO 115
120:U=0
125:U=U+FX:IF U<=B
THEN LINE (U/R
,M/S)-(U/R,N/S
),7:GOTO 125
130:U=0
135:U=U-FY:IF U>=N
THEN LINE (A/R
,U/S)-(B/R,U/S
),7:GOTO 135
140:U=0
145:U=U+FY:IF U<=M
THEN LINE (A/R
,U/S)-(B/R,U/S
),7:GOTO 145
150:X=A:GOSUB L:P=
A:Q=F:FOR I=1
TO H
155:X=A+I*D:J=0:
GOSUB L:IF (F>
MOR F<NOR Q>M
OR Q<NOR J=1)
GOTO 170
160:LINE (P/R,Q/S)
-(X/R,F/S),G,U
170:P=X:Q=F:NEXT I
:IF W=0GOTO 18
0
172:GOSUB 220:
GOSUB 200:
GOSUB 210:U=U+
1:IF U=4LET U=
1:G=G+1
175:GOTO 150
177:GLCURSOR (0,N/
S)
180:TEXT :LF 4:
COLOR 0:END
200:BEEP 1,50,200:
INPUT "LINEA I
NIZIO?" ;L:IF L
=0GOTO 175
205:RETURN
210:INPUT "PARAMET
RO?" ;K:W=1:
RETURN
220:LINE (A/R,N/S-
(G*3+U)*15-10)
-(A/R+30,N/S-(
G*3+U)*15-10),
G,U:LPRINT "K
=" ;K:RETURN

```

Figura 6 - Listato del programma "Grafici 1"

numero di linea d'inizio della funzione stessa (o delle varie funzioni). Dopodiché viene richiesto se si desidera una grigliatura del grafico per facilitarne la lettura quantitativa; nel caso non la si volesse affatto, è sufficiente eliminare le linee dalla 35 alla 75 e dalla 105 alla 145 comprese. Questa viene scelta automaticamente dal programma in ragione di 2 elevato a potenze positive o negative in modo da avere sempre almeno due punti tarati per ogni regione del piano (cioè le righe della grigliatura verranno disegnate per esempio ogni 0.25 o 0.5 o 1 o 2 o 4 ecc.). Volendo infittirle, è sufficiente aumentare il divisore 2 nelle linee 45, 50, 60 e 65. Vengono poi stampati i relativi valori degli intervalli di taratura Fx e Fy e la griglia viene eseguita con linee tratteggiate.

Il metodo adottato può sembrare uno spreco di byte, ma è il più efficiente per ciò che riguarda l'esecuzione esatta di quanto richiesto, evitando di eseguire lunghi test per fare in modo sia che il pennino non tenti di tracciare qualcosa fuori del formato, sia che la grigliatura acquisti in generalità. Quindi comincia il plot vero e proprio.

Le uniche osservazioni da fare sulle righe 150-180 sono le seguenti:

150: prepara le variabili in modo che il pennino parta dall'esatta posizione in corrispondenza dell'estremo inferiore.

155: incrementa la X, calcola la funzione in X ed esegue la successiva 160 solo se gli estremi da plottare non sono fuori del grafico (caso di discontinuità o di parametri che amplificano la funzione) e cioè se F (punto attuale) e Q (punto precedente) non sono maggiori del massimo o minori del minimo. J=1 viene settata eventualmente nella subroutine di calcolo della funzione in caso di discontinuità per evitare che il pennino unisca con una bella linea colorata, ciò che non doveva unire. Si pensi ad esempio ad $F=1/X$ e si noti la differenza tra le figure 1 e 2.

160: traccia la linea con l'adatto colore e tratteggio.

170: chiude il ciclo FOR, trasferendo i punti di arrivo X e F nelle variabili di partenza del prossimo segmento P e Q. Terminato il ciclo, se la funzione era parametrizzata, si procede ad un nuovo input, altrimenti si va alla fine dove un Line Feed calcolato opportunamente fa uscire tutto il grafico dal plotter, pronto per lo "strap".

172: oltre ai richiami delle subroutine, si provvede al cambio di colore (variabile U) ed all'eventuale tratteggio del segmento plottato (variabile G) che entra in azione dopo aver usato tutti e tre i colori disponibili (il nero viene usato solo per gli assi) e viene ulteriormente spaziato ogni volta che si torna al blu.

175: il programma prosegue con un altro parametro.

177: nel caso di uscita da una funzione non parametrizzata, posiziona il cursore in modo da poter usufruire della successiva 180.

180: comune ad entrambe le uscite.

Quindi le seguenti 200 e 210 realizzano gli input di linea e di parametro. Quando si vuole uscire dal loop di tracciatura, che proseguirebbe all'infinito, si batte 0 come numero di linea; mentre la subroutine 220 provvede a stampare un pezzetto di linea con colore e tratteggio dell'ultimo plot e di seguito il valore del relativo parametro, curando che queste informazioni vengano incolonnate e giustamente spaziate.

La funzione può essere scritta a partire dalla linea 220 in poi, sapendo che la variabile indipendente da usare è X, il valore della funzione da restituire è in F, e l'eventuale parametro è in K.

Particolare cura va posta nella sua stesura nel caso che la $f(x)$ abbia delle discontinuità: se sono note, si fa un test di esclusione di quel punto, cioè se x_0 è il punto incriminato, possiamo scrivere $IF X=x_0$, RETURN. Se non sono note, si può scrivere (per esempio):

```
500 ON ERROR GOTO 510
```

```
505 F = f(x): RETURN
```

```
510 elimina errore: RETURN
```

Questi rimedi, però evitano solo che il programma si fermi per un syntax error; se vogliamo invece cautelarci anche da segmenti indesiderati nel grafico, allora useremo l'apposita variabile J o dei test aggiuntivi su X e P, per esempio, come mostrato nella figura 2.

Il grafico si sviluppa in senso orizzontale per usufruire di una leggibilità immediata, ma con qualche piccola modifica è possibile farlo sviluppare in lunghezza.

Non vi allarmate se, come nell'esempio delle sinusoidi di figura 3 dato $Fx = 64,64*3$ fa 192 e non 180. Infatti Fx e Fy valgono tra i punti tarati, quindi se gli estremi non hanno la retta tratteggiata, poco male: si sa già che valgono A o B ed M (massimo) o N (minimo).

La figura 3 mostra gli effetti della discretizzazione (sommità della sinusoide clippata) per $K = 2$.

GRAFICI 2

di Tommaso Nannelli - Firenze

Il programma può disegnare il grafico di una o due funzioni con o senza retta di asintoto, le equazioni delle quali devono essere inserite nelle linee 2, 6 e 30 nella forma $Y=F(X)$.

Dopo aver premuto DEF "space", sul display apparirà la richiesta "RETTA"? alla quale bisognerà rispondere con S o N; la seconda richiesta sarà "1 o 2 funzioni?" con la quale il PC-1500 chiede se si vuole disegnare solo la prima o tutte e due le funzioni impostate; in seguito si devono inserire i limiti degli assi X e Y entro i quali la funzione deve essere disegnata ed il fat-

```

1:ON ERROR GOTO 10
2:REM FUNZIONE
3:RETURN
6:REM FUNZIONE
7:RETURN
10:Y=YI-1
15:RETURN
20:X=XS:GOSUB 30:
RY=Y:X=XD:
GOSUB 30:RJ=Y
21:X=XS
22:IF RY<YI LET X=
X+INC:GOSUB 30
:RY=Y:GOTO 22
23:XG=X:X=XD
24:IF RJ>YS LET X=
X-INC:GOSUB 30
:RJ=Y:GOTO 24
25:COLOR 2:LINE (
XG*AN+ABS SS+2
,RY*AN*AU)-(X*
AN+ABS SS+2,RJ
*AN*AU):RETURN
30:REM RETTA
40:RETURN
69:" ":CLEAR :
RADIANT :GOTO 6
10
70:GRAPH
90:GLCURSOR (AN*
ABS XS,-AN*ABS
YS*AU):SORGN
95:COLOR 0
100:LINE (SS,0)-(S
D,0)
102:LINE (SD-7,-4)
-(SD,0)-(SD-7,
4)
104:FOR XG=ANTO SD
STEP AN
105:LINE (XG,-2)-(
XG,2)
106:NEXT XG:
GLCURSOR (SD-8
,-10):CSIZE 1:
LPRINT "X"
107:FOR XG=-ANTO S
SSTEP -AN:LINE
(XG,2)-(XG,-2)
:NEXT XG
108:IF AU<1 LET PU=
AU*10:GOTO 110
109:PU=AU
110:LINE (0,Q1*AU)
-(0,QS*AU)
112:LINE (-4,QS*AU
-7)-(0,QS*AU)-
(4,QS*AU-7)
113:FOR YG=AN*AUTO
QS*AU-ANSTEP A
N*PU:LINE (-2,
YG)-(-2,YG)
114:NEXT YG:
GLCURSOR (0,QS
*AU-10):LPRINT
"Y"
115:FOR YG=-AN*AU
TO Q1*AU+AN
STEP -AN*PU:
LINE (-2,YG)-(-
2,YG)
117:NEXT YG
120:GLCURSOR (SS,0
):SORGN
122:COLOR 3
125:X=XS:X=X-2*INC
130:FOR XG=-2 TO 21
5
135:IF FX=1GOSUB 6
:GOTO 145
140:GOSUB 1
145:Y1=Y*AN*AU:IF
Y1>YSOR Y<Y1
GOSUB 450:LET
X=X+INC:FL=1:
GOTO 180
147:IF FL=1LET O=X
G:Z=Y1:FL=0
150:X=X+INC
170:LINE (0,2)-(XG
,Y1)
175:O=XG:Z=Y1:F1=0
180:NEXT XG
210:IF NF=2LET FX=
1:NF=3:COLOR 1
:O=-3:Z=-3:
GOTO 125
213:IF RE$="S"
GOSUB 20
214:GLCURSOR (0,Q1
*AU)
215:TEXT
220:COLOR 3:CSIZE
1
225:LLIST 2:LF -4:
COLOR 1:LLIST
6:LF -4:COLOR
2:LLIST 30:LF
5
350:BEEP 3:END
450:IF F1=1RETURN
500:IF Y1>YS*AU
LINE (0,2)-(XG
,Q1*AU):F1=1:
RETURN
510:IF Y1<Y1*AU
LINE (0,2)-(XG
,Q1*AU):F1=1:
RETURN
610:INPUT "Retta?(
S/N)";RE$:Z=-3
:O=-3
614:INPUT "1 o 2 f
unzioni?";NF
620:INPUT "Lim.sin
asse X=";XS
630:INPUT "Lim.des
asse X=";XD
640:INPUT "Lim.inf
asse Y=";Y1
650:INPUT "Lim.sup
asse Y=";YS
655:INPUT "Fatt.sc
ala asse Y=";A
U
657:CLS
660:INC=(ABS XS+
ABS XD)/215:AN
=1/INC:SS=AN*X
S:SD=AN*XD:QS=
AN*YS:Q1=AN*Y1
:GOTO 70

```

Figura 8 - Listato del programma "Grafici 2"

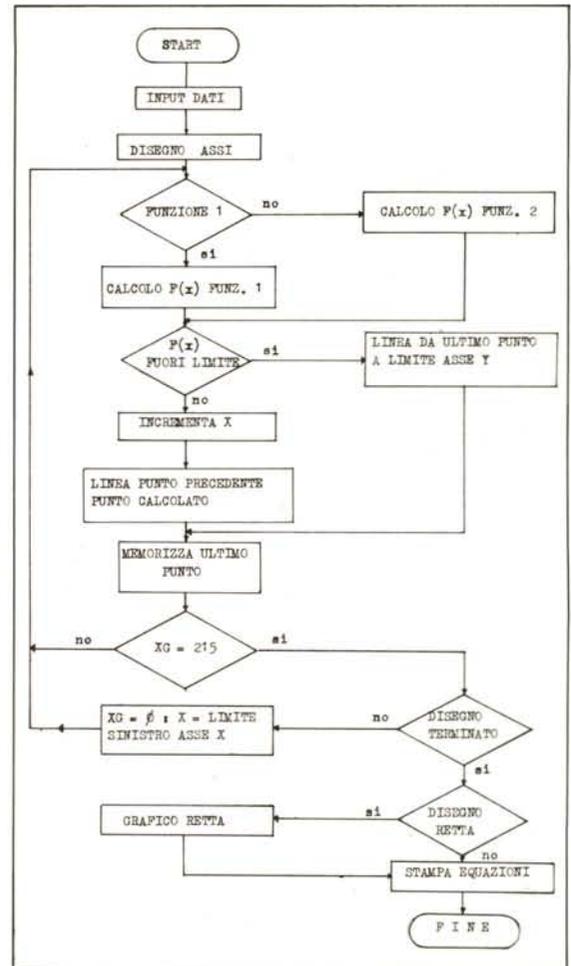


Figura 9 - Flow-chart programma Grafici 2.

tore di scala che provvede ad allungare o accorciare l'unità di misura dell'asse Y rispetto a quella dell'asse X.

Dopo aver disegnato il grafico delle funzioni richieste, il calcolatore provvede a listare le linee di programma nelle quali sono state inserite le equazioni delle curve da disegnare. Per accelerare i tempi di ela-

borazione, le subroutine che contengono le funzioni sono state poste all'inizio del programma, mentre quelle di inizializzazione e di input si trovano alla fine.

La parte centrale è costituita dal loop che va da 130 a 180, mentre altre routine (70-117) e (210-350) provvedono rispettivamente a disegnare gli assi ed a listare le

linee di programma che contengono le equazioni, in diversi colori. La prima funzione sarà disegnata in rosso, la seconda in blu e la retta in verde. Infine la subroutine che va dalle linee 450 a 510 provvede a non far uscire il grafico dai limiti dati, utile per le funzioni che hanno asintoti verticali ed obliqui.

MC

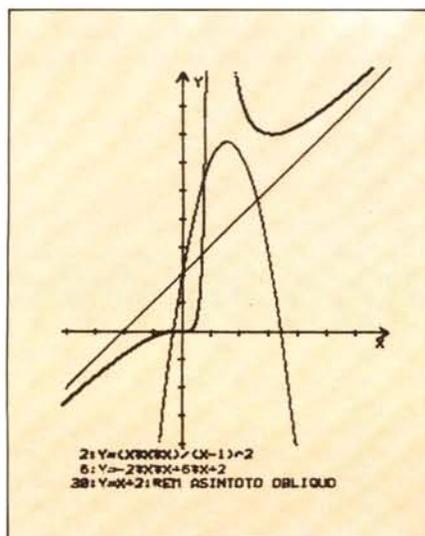


Figura 7 - "Grafici 2" - Esempio di output

...ANCORA SUL CE-150

Per quanto riguarda le istruzioni GLCURSOR e LINE, abbiamo notato che un incremento di 1 unità degli argomenti corrisponde ad uno spostamento di circa 0,2 mm del cursore del CE-150; dato che l'insieme dei valori che possono assumere tali argomenti (X,Y) è compreso nell'intervallo ± 2047 , è facilmente deducibile che il PC-1500 sarebbe in grado di effettuare disegni su fogli di circa 41 cm per lato! Riuscendo ad interpretare le funzioni delle 30 linee sul port di I/O della periferica, non dovrebbe essere impossibile realizzare un'opportuna interfaccia per plotter standard già esistenti sul mercato, oppure, per gli esperti, addirittura progettare tutto il sistema scrivente comprese penne e step-motor.

Aldilà di queste considerazioni tecniche, vorremmo segnalare un'ulteriore istruzione non citata dal fatidico quanto inutile ma-

nale del PC-1500. Si tratta di un'estensione dell'impiego dell'istruzione LINE:

LINE-(X,Y)

Posta in questo modo, l'istruzione provoca il tracciamento di un segmento dal punto in cui si trova attualmente il cursore, fino al punto individuato dalle coordinate (X,Y). Si intuisce quindi quanto questo potrebbe risultare utile nel disegno di grafici di funzione. Memorizzando infatti in X0 e Y0 le coordinate del primo punto di tabulazione, si potrà utilizzare la seguente routine per il tracciamento del grafico:

```

5000 GLCURSOR (X0, Y0)
5010 incrementa X
5020 calcola Y(X)
5030 LINE -(X, Y)
5040 IF condizione = VERA THEN
"OUT"
5050 GOTO 5010
"OUT"

```

F.M.