

L'ADP Basic:

P come Plotter

di Andrea de Prisco

Terzo appuntamento con l'ADP Basic, il tool di istruzioni per facilitare l'uso delle periferiche del Commodore 64. Dopo aver visto l'uso col drive 1541, questo mese giocheremo le nostre carte con il plotter 1520, sempre della Commodore. Vedremo come è possibile scrivere, disegnare, cambiare rapidamente colore penna, formato carattere, direttamente da tastiera o da programma, senza impazzire con dodicimila OPEN. Comandi semplici, come vedremo: Write per scrivere, Move per muovere la penna, Draw per tracciare linee.

Terza parte

20 comandi Plotter-oriented

D'ora in poi, chi ha un plotter 1520 potrà comunicare con questo tramite 20 nuove istruzioni Basic. Prima di spiegare dettagliatamente la loro implementazione a livello interprete, diamo la sintassi corretta per ognuna. Si suppone che chi legge sappia come normalmente funziona un plotter Commodore.

Il primo comando è PL e, a seconda di cosa precede, ha diversi significati:

1) PL ON: predispone il plotter come normale periferica di output. Corrisponde a un OPEN 4,6 seguito da CMD 4. Dopo

tale comando, qualsiasi PRINT sarà dirottata su periferica.

2) PL OFF: riabilita il video quale periferica di output. Si usa per disattivare lo stato PL ON.

3) PL RESET: resetta il plotter, provocando il self test delle quattro penne colorate. Altri comandi inviati durante il reset vengono ignorati.

4) PL SMALL: predispone la scrittura con lettere minuscole normalmente e maiuscole se precedute da SHIFT. Se dato dopo PL ON, forza un PL OFF automaticamente (il video torna periferica di output)

5) PL CAP: predispone la scrittura con lettere maiuscole normalmente e minuscole se precedute da SHIFT. Valgono le stesse considerazioni di PL SMALL.

La veste un po' strana di questo comando (un prefisso, PL, e 5 diversi suffissi) è stata data per uniformare il funzionamento con i comandi per stampante che vedremo sul prossimo numero. Avremo anche li PR ON, PR OFF, PR RESET, PR SMALL e PR CAP, con analogo significato. Ricordarsi di usare PL ON solo dopo aver scelto le varie opzioni di stampa (colore penna, dimensione carattere, rotazione), come vedremo più avanti.

Per stampare direttamente una o più linee, senza cambiare periferica di output, è disponibile il comando WRITE. La sua sintassi è identica a quella del comando PRINT del Basic Standard del Commodore 64. Ad esempio, potremmo scrivere:

```
WRITE "TOTALE = ";C1;"PUNTEGGIO = ";C2
```

o qualsiasi altra combinazione di stringhe e valori numerici.

Per cambiare il formato dei caratteri è disponibile il comando SIZE, seguito da 10, 20, 40 o 80 a seconda del numero di caratteri per linea da stampare. Un parametro specificato diverso da quelli sopra elencati genera il messaggio di errore ILLEGAL QUANTITY. Ad esempio, per passare al modo 80 colonne è sufficiente digitare:

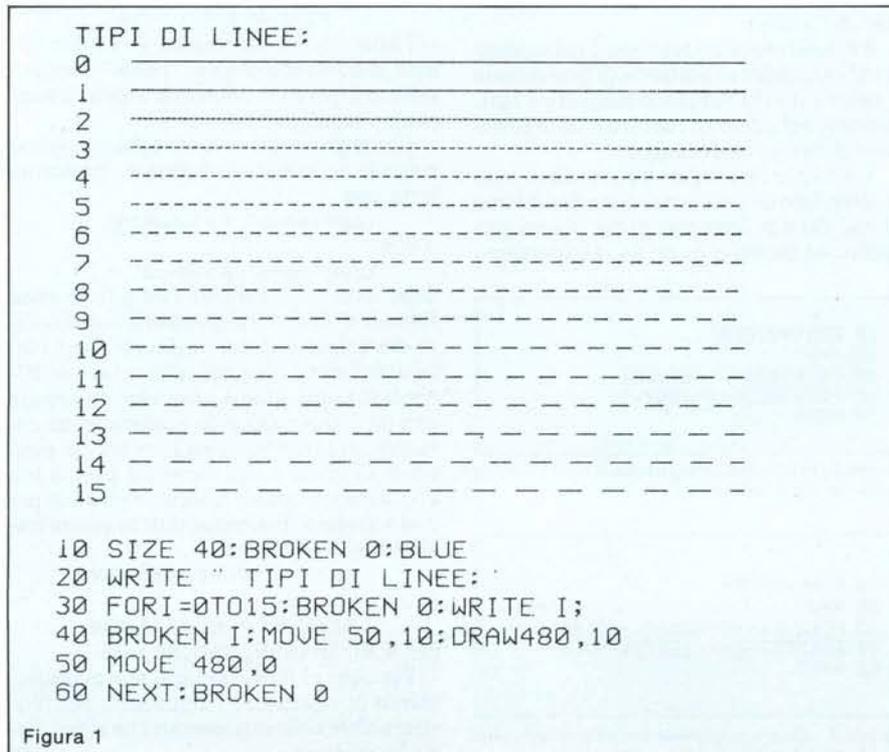
```
SIZE 80
```

Il comando ROTATE serve per attivare la scrittura di caratteri orizzontali (per tabulazioni lungo il senso di scorrimento della carta). Più precisamente:

ROTATE ON: ruota i caratteri di novanta gradi in senso orario

ROTATE OFF: seleziona il modo caratteri verticali (modo standard).

Per cambiare colore penna si può procedere in due distinti modi: scrivendo direttamente (in inglese) il colore o con il coman-

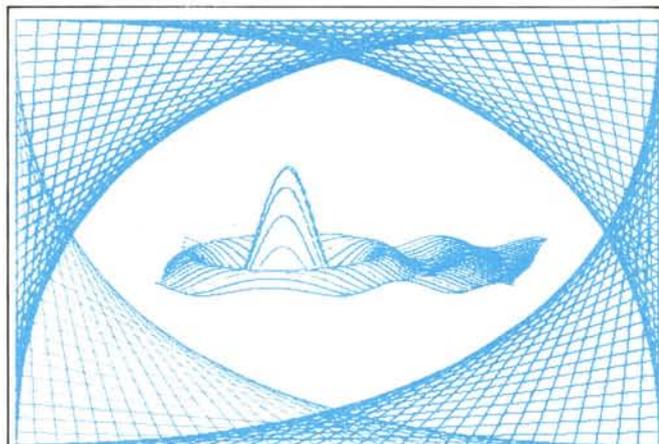




```

0 INPUT "QUANTI ";K:PRINT:FORT=1TOK:PRINTT:C=0
5 A=40:SIZE 40
7 MOVE0,-400:SET
10 FORAN=0T0360STEP30:X0=AN+20:Y0=SIN(AN/4/180*PI)*250:C=C+1
30 GOSUB1000
40 COLOR C
50 RELMOVE X1,Y1
60 RELDRAW X2,Y2
70 RELDRAW X3,Y3
80 RELDRAW X1,Y1
100 NEXT:RELMOVE 0,125
110 WRITE "MARIO BIANCHI"
120 WRITE "VIALE CERTOSA 255"
130 WRITE "10101 TORINO"
135 WRITE "TEL.: 011/7978787"
140 COLOR 0:SIZE 80
150 WRITE "STUDIO: BIANCHI & ROSSI SNC"
160 WRITE "VIA CERVI 88 - TORINO"
170 WRITE "TEL.: 011/5657585"
175 WRITE "ORARIO: 9-13 16-19.30"
180 NEXT
999 END
1000 X1=X0+A*COS(AL)
1005 AL=AN/180*PI
1010 Y1=Y0+A*SIN(AL)
1020 X2=X0+A*COS(AL+2.0944)
1030 Y2=Y0+A*SIN(AL+2.0944)
1040 X3=X0+A*COS(AL+4.1888)
1050 Y3=Y0+A*SIN(AL+4.1888)
1060 RETURN
    
```

Listato A - Questo programma permette di stampare simpatici biglietti da visita.



```

5 FORK=1T020:WRITE CHR$(17):NEXT
6 REM *****
7 REM *
8 REM *   000   000   0000  000   *
9 REM *   0   0   0   0   0   0   *
10 REM *   0   0   00000  000   0   *
11 REM *   0   0   0   0   0   0   *
12 REM *   000   0   0   0000  000   *
13 REM *
14 REM *
15 REM *   -----   *
16 REM *   (C) 1984 ADP-SOFTWARE   *
17 REM *   -----   *
18 REM *
19 REM *****
20 FORK=0T0479STEP16
25 MOVE 0,K/1.5
30 RED:DRAW K,320
40 BLACK:DRAW 479,320-K/1.5
50 BLUE:DRAW 479-K,0
60 GREEN:DRAW 0,K/1.5
70 NEXT
100 MOVE100,100:SET
110 DIMA%(320):T=15
120 FORY0=-9.5T09.5STEP.7:FL=1
130 FORX0=-9.5T019STEP.08
140 Q=SQR(X0*X0+Y0*Y0)
150 Y=INT((SIN(Q)/Q)*75+T*1.3)
160 X=INT((X0+9.5)*9.5+T*.92-14)
170 IFX>0ANDX<320THENIFY>A%(X)THENA%(X)=Y:GOSUB1000:GOTO190
180 RELMOVE X,Y
190 NEXT:T=T+1:NEXT:END
1000 IFFL=1THENFL=0:COLOR C:C=C+1:RELMOVE X,Y:RETURN
1010 RELDRAW X,Y:RETURN
    
```

do COLOR che accetta come parametro un intero. Riassumendo:

colore	com. diretto	com. parametrico
nero	BLACK	COLOR 0
blu	BLUE	COLOR 1
verde	GREEN	COLOR 2
rosso	RED	COLOR 3

indicando come parametro del comando

COLOR un numero maggiore di 3, sarà interpretato Modulo 4 (il resto della divisione con 4). Per intenderci: COLOR 4 sarà di nuovo nero, COLOR 5 blu, COLOR 6 verde e così via.

E per quanto riguarda il Text Mode abbiamo finito: possiamo ad illustrare le istruzioni grafiche.

MOVE <Ascissa>,<Ordinata>

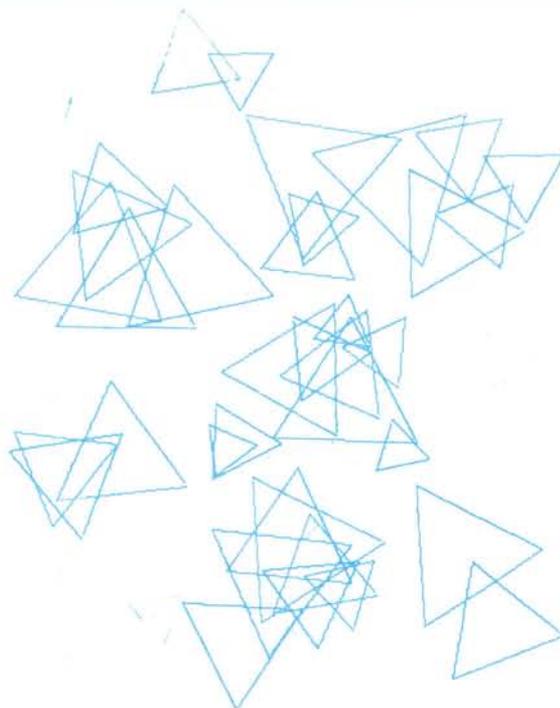
permette di spostare la testina di scrittura con la penna staccata dal foglio. Ascissa e Ordinata possono essere numeri, espressioni numeriche, variabili. Il plotter stesso ignora spostamenti oltre la cosiddetta Plottable-Area, come ben spiegato sul manuale di istruzione dello stesso a pag. 16.

DRAW <Ascissa>,<Ordinata>
muove la testina di scrittura mantenendo

```

10 MOVE0,-400:SET
11 REM *****
12 REM *
13 REM * TRIANGOLI RND *
14 REM *
15 REM *****
16 X0=30+RND(1)*400:Y0=RND(1)*319
20 AN=RND(1)*360:A=20+40*RND(1):C=C+1-(R
ND(1)).7)
30 GOSUB1000
40 COLOR C
50 RELMOVE X1,Y1
60 RELDRAW X2,Y2
70 RELDRAW X3,Y3
80 RELDRAW X1,Y1
90 RELMOVE 0,Y1
100 GETA$:IFA$=""THEN10
999 END
1000 AL=AN/180*PI
1005 X1=INT(X0+A*COS(AL))
1010 Y1=INT(Y0+A*SIN(AL))
1020 X2=INT(X0+A*COS(AL+2.0944))
1030 Y2=INT(Y0+A*SIN(AL+2.0944))
1040 X3=INT(X0+A*COS(AL+4.1888))
1050 Y3=INT(Y0+A*SIN(AL+4.1888))
1060 RETURN

```



appoggiata sul foglio la penna. In altre parole traccia linee dalla attuale posizione sino alla posizione specificata. Facciamo un esempio: disegniamo un bel quadrato al centro del foglio.

```

10 MOVE 120,0
20 DRAW 120,240
30 DRAW 360,240
40 DRAW 360,0
20 DRAW 120,0

```

Il comando HOME serve per riportare la penna nel punto di coordinate (0,0) detto di origine. Se vogliamo dichiarare un punto di origine diverso (un nuovo sistema di riferimento) è disponibile il comando SET. Per tracciare linee o spostare semplicemente la testina usando il nuovo sistema di riferimento si usano rispettivamente le istruzioni RELDRAW e RELMOVE. Facciamo un esempio, ridisegniamo il quadrato di cui sopra, facendo uso di un nuovo sistema di riferimento.

```

10 MOVE 240,0
20 SET
30 RELMOVE 120,120
40 RELDRAW 120,-120
50 RELDRAW -120,-120
60 RELDRAW -120,120
70 RELDRAW 120,120

```

alla linea 10 abbiamo spostato la penna nel punto centrale del foglio e lì abbiamo dichiarato la nuova origine (linea 20). Segue (linee 30-70) il disegno vero e proprio del quadrato, relativo al nuovo sistema di riferimento.

L'ultima istruzione di questo mese, il comando BROKEN, permette di usare diversi tipi di tratteggio per le linee. In figura 1 la distinta dei tipi e il relativo programma che l'ha generata.

Due istruzioni nascoste (ma non troppo)

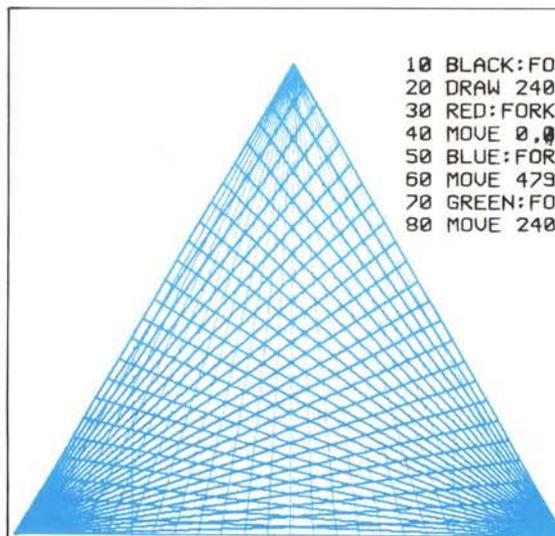
OFF e RESET dati senza alcun prefisso hanno un ben preciso significato. La prima disabilita l'ADP BASIC non rendendo più disponibili i nuovi comandi. Per rientrare in ambiente ADP è sufficiente digitare SYS 49152. RESET è più radicale: provoca un vero e proprio RESET di tutta la macchina, riinizializzando tutte le variabili interne con relativa perdita del programma BASIC mantenuto in memoria. In altre parole fate molta attenzione. Anche dopo il RESET, per riattivare l'ADP BASIC è sufficiente digitare SYS 49152.

General Remarks

I vari comandi presentati questo mese, fanno riferimento spesso e volentieri a tre subroutine listate nella pagina a fianco.

La prima, usata dal costrutto PL ON e dal comando WRITE, apre un file di output col plotter: corrisponde semplicemente a un OPEN 60,6,0. Per non interferire con altri file aperti dall'utente, l'ADP BASIC usa sempre numeri molto alti per i propri file.

La seconda e la terza subroutine sono usate praticamente da tutti gli altri comandi, essendo di uso più generale. Sono un OPEN e un CLOSE file specificato nell'ac-



```

10 BLACK:FORI=1TO25:WRITE"Q":NEXT
20 DRAW 240,416:DRAW 479,0:DRAW 0,0
30 RED:FORK=10TO230STEP10
40 MOVE 0,0:DRAW479-K,1.73*K:NEXT
50 BLUE:FORK=10TO230STEP10
60 MOVE 479,0:DRAWK,1.73*K:NEXT
70 GREEN:FORK=20TO460STEP20
80 MOVE 240,416:DRAWK,0:NEXT

```

accumulatore +80, indirizzo secondario accumulatore. Ad esempio, se l'accumulatore contiene 5, una chiamata alla seconda subroutine provocherà un:

```
OPEN 85,6,5:CMD 85
```

una chiamata alla terza:

```
CLOSE 85
```

Il listato 1 implementa l'istruzione PL. La prima operazione è un salto alla Char-Get Routine del sistema operativo. Questa, locata a partire dalla locazione \$0073 (si, in pagina zero per velocizzarla al massimo), non fa altro che prelevare il byte successivo della linea che si sta eseguendo. Chiamata con JSR\$0073, avanza di una posizione e preleva il byte; chiamata con

JSR\$0079 preleva il byte senza far avanzare il puntatore all'interno della linea.

Dunque, JSR\$0079 per sapere cosa segue PL; 5 casi: ON, OFF, RESET, SMALL, CAP (tutti e 5 tokenizzati, s'intende). Altrimenti stampa Syntax Error grazie al JMP\$C62B. I token dei 5 suffissi sono rispettivamente, in esadecimale, \$91, \$E0, \$DF, \$E1 e \$E2.

All'indirizzo \$CA94 inizia il corpo di PL ON: l'unica operazione compiuta è un salto alla subroutine 1, la specifica output del canale aperto e un salto brutale all'istruzione DATA per continuare la normale esecuzione dello statement successivo, se c'è.

PL OFF inizia a \$C9AF e semplicemen-

te stampa un [RETURN], \$0D in esadecimale e chiude canali e file aperti.

PL RESET spedisce un comando nullo al plotter dopo aver aperto un file con indirizzo secondario 7. Per fare questo (indirizzo \$C96B) dopo aver inserito in A il numero 7, salta alla subroutine 2. Segue un PRINT CHR\$(0) e un CLOSE tramite la subroutine 3.

PL SMALL, indirizzo \$C9CC, analogamente apre un file con indirizzo secondario 6 e spara al device il codice \$31 corrispondente al carattere "1".

PL CAP, indirizzo \$C9CB, fa lo stesso, spedendo il codice ASCII di "0".

Il listato 2 implementa l'istruzione HOME: per realizzarla, apre un file di indiriz-

C950 20 CC FF JSR \$FFCC	C9C7 A9 07 LDA #\$07	CA30 A9 01 LDA #\$01	CAA0 C9 28 CMP ##28
C953 20 E7 FF JSR \$FFE7	C9C9 4C 7F C9 JMP \$C97F	CA3F 20 68 C9 JSR \$C968	CAA2 F0 09 BEQ \$CAAD
C956 A9 46 LDA #\$46	C9CC A9 06 LDA #\$06	CA42 A9 4A LDA #\$4A	CAA4 C9 50 CMP ##50
C958 A2 06 LDX ##06	C9CE 20 68 C9 JSR \$C968	CA44 D0 C6 BNE \$CA0C	CAA6 F0 05 BEQ \$CAAD
C95A A0 00 LDY ##00	C9D1 A9 31 LDA #\$31		CAA8 A2 0E LDX ##0E
C95C 20 BA FF JSR \$FFBA	C9D3 20 D2 FF JSR \$FFD2	<i>Listato 6 - Comando REL.</i>	CAA4 4C 8B E3 JMP \$E38B
C95F A9 00 LDA #\$00	C9D6 A9 06 LDA #\$06	CA46 20 9E AD JSR \$AD9E	CAAD A2 FE LDX ##FE
C961 20 BD FF JSR \$FFBD	C9D8 4C 7F C9 JMP \$C97F	CA49 20 DD BD JSR \$BD0D	CAAF E8 INX
C964 20 C0 FF JSR \$FFC0	C9DB A9 06 LDA #\$06	CA4C 20 1E AB JSR \$AB1E	CAB0 0A ASL
C967 60 RTS	C9DD 20 68 C9 JSR \$C968	CA4F 20 79 00 JSR \$0079	CAB1 90 FC BCC \$CAAF
<i>Subroutine 1 - OPEN 60,6,0.</i>	C9E0 A9 30 LDA #\$30	CA52 20 D2 FF JSR \$FFD2	CAB3 8A TXA
	C9E2 20 D2 FF JSR \$FFD2	CA55 20 73 00 JSR \$0073	CAB4 48 PHA
	C9E5 A9 06 LDA #\$06	CA58 20 9E AD JSR \$AD9E	CAB5 A9 03 LDA #\$03
	C9E7 4C 7F C9 JMP \$C97F	CA5B 20 DD BD JSR \$BD0D	CAB7 20 68 C9 JSR \$C968
<i>Listato 1 - Comando PL.</i>		CA5E 20 1E AB JSR \$AB1E	CABA 68 PLA
	C9ED A9 01 LDA #\$01	CA61 60 RTS	CABB AA TAX
	C9EF 20 68 C9 JSR \$C968	<i>Subroutine 4 - Spedisce al plotter i</i>	CABC A9 00 LDA #\$00
	C9F2 A9 48 LDA #\$48	<i>parametri delle istruzioni MOVE,</i>	CABE 20 CD BD JSR \$BD0D
	C9F4 20 D2 FF JSR \$FFD2	<i>DRAW, REMOVE e RELDRAW.</i>	CAC1 A9 03 LDA #\$03
	C9F7 A9 01 LDA #\$01		CAC3 4C 7F C9 JMP \$C97F
	C9F9 4C 7F C9 JMP \$C97F		<i>Listato 2 - Comando SIZE.</i>
<i>Listato 2 - Comando HOME.</i>		CAE2 A9 02 LDA #\$02	CAC6 20 79 00 JSR \$0079
	C9FC A9 01 LDA #\$01	CA64 20 68 C9 JSR \$C968	CAC9 C9 91 CMP ##91
	C9FE 20 68 C9 JSR \$C968	CA67 20 9E B7 JSR \$B79E	CACB F0 07 BEQ \$CAD4
	CA01 A9 49 LDA #\$49	CA6A 8A TXA	CACD C9 E3 CMF ##E0
	CA03 D0 EF BNE \$C9F4	CA6B 29 03 AND #\$03	CACF F0 03 BEQ \$CAD4
<i>Listato 3 - Comando SET.</i>		CA6D 09 30 ORA #\$30	CA01 4C 2B C6 JMP \$C22E
	CA05 A9 01 LDA #\$01	CA6F 20 D2 FF JSR \$FFD2	CA04 29 01 AND #\$01
	CA07 20 68 C9 JSR \$C968	CA72 A9 02 LDA #\$02	CA06 09 30 ORA #\$30
	CA0A A9 4D LDA #\$4D	CA74 4C 7F C9 JMP \$C97F	CA09 40 PHA
	CA0C 20 D2 FF JSR \$FFD2	<i>Listato 7 - Comando COLOR.</i>	CA03 A9 04 LDA #\$04
	CA0F 20 46 CA JSR \$CA46		CA06 20 68 C9 JSR \$C968
	CA12 A9 01 LDA #\$01		CA0E 66 PLA
	CA14 4C 7F C9 JMP \$C97F		CA0F 20 D2 FF JSR \$FFD2
<i>Listato 4 - Comando MOVE.</i>		CA80 20 D2 FF JSR \$FFD2	CAE2 A9 04 LDA #\$04
	CA17 A9 01 LDA #\$01	CA83 A9 02 LDA #\$02	CAE4 4C 7F C9 JMP \$C97F
	CA19 20 68 C9 JSR \$C968	CA85 4C 7F C9 JMP \$C97F	<i>Listato 13 - Comando ROTATE.</i>
	CA1C A9 44 LDA #\$44	<i>Listato 8 - Comando BLACK.</i>	CAE7 A9 05 LDA #\$05
	CA1E D0 EC BNE \$CA0C		CAE9 20 68 C9 JSR \$C968
<i>Listato 5 - Comando DRAW.</i>		CA89 A9 31 LDA #\$31	CAEC 20 9E AD JSR \$AD9E
	CA20 20 79 00 JSR \$0079	CA8A D0 ED BNE \$CA79	CAEF 20 DD BD JSR \$BD0D
	CA23 C9 E6 CMP ##E6	<i>Listato 9 - Comando BLUE.</i>	CAF2 20 1E AB JSR \$AB1E
	CA25 F0 07 BEQ \$CA2E		CAF5 A9 05 LDA #\$05
	CA27 C9 E7 CMP ##E7		CAF7 4C 7F C9 JMP \$C97F
	CA29 F0 0F BEQ \$CA3A		<i>Listato 14 - Comando BROKEN.</i>
	CA2B 4C 2B C6 JMP \$C62B		CAFA 20 50 C9 JSR \$C950
	CA2E 20 73 00 JSR \$0073		CAF0 A2 46 LDX ##46
	CA31 A9 01 LDA #\$01		CAFF 20 C9 FF JSR \$FFC9
	CA33 20 68 C9 JSR \$C968		CB02 A9 0D LDA #\$0D
	CA36 A9 52 LDA #\$52		CB04 20 A0 AA JSR \$AAA0
	CA38 D0 D2 BNE \$CA0C		CB07 20 CC FF JSR \$FFCC
	CA3A 20 73 00 JSR \$0073		CB0A A9 46 LDA #\$46
			CB0C 4C 7F C9 JMP \$C97F
			<i>Listato 15 - Comando WRITE.</i>

zo secondario 1 e spedisce una "H" (codice \$48) al plotter.

Il listato 3 implementa SET sfruttando, per risparmiare byte, parte dell'istruzione HOME. Apre lo stesso file e dopo aver posto in A il codice ASCII di "I" salta nel corpo di HOME che provvede a spedire A e a chiudere canale e file aperto.

Il comando MOVE (listato 4) usa la subroutine 4. Questa non fa altro che spedire al plotter valori posti dopo il comando (le coordinate). Più precisamente, apre un file di indirizzo secondario 1, spedisce una "M" (codice \$4D), salta alla subroutine 4, chiude canale e file aperto.

Il comando DRAW (listato 5), analogamente, apre lo stesso file, spedisce una "D" e salta nel corpo di MOVE per inviare al plotter le coordinate.

Il listato 6 implementa il comando REL, che come visto può essere seguito solo da DRAW o da MOVE. Come per il comando PL, la prima operazione preleva il byte dopo REL. Confronta questo con i token di MOVE e DRAW (\$E6 e \$E7) e stampa SYNTAX ERROR se trova qualcosa di diverso. Il JSR\$0073 di \$CA2E serve per posizionare il puntatore al byte della linea eseguita, in corrispondenza del primo parametro. Apre un file di indirizzo secondario 1 e a seconda della presenza di MOVE o DRAW dopo REL, spedisce al plotter rispettivamente una "R" o una "J" prima di cedere il controllo a MOVE che invierà i parametri.

Il listato 7 implementa l'istruzione COLOR: apre un file di indirizzo secondario 2

Senza l'ADP...

Il Plotter 1520 della Commodore è, fortunatamente, una periferica abbastanza intelligente. Nel senso che dispone di un proprio linguaggio per eseguire comandi: basta spedire al device opportune stringhe tramite specifici canali di comunicazione. Un po' succedeva anche col drive 1541: i comandi (senza ADP, s'intende) vengono spediti tramite stringhe. Col plotter, essendo questi molti e i canali di comunicazione ben 8 (nel disco era uno solo), programmare l'uso di questa periferica diventa un'operazione assai seccante, quasi impossibile senza avere sottomano continuamente le pagine del manuale di istruzioni.

Ad esempio, per tracciare una linea, dalle coordinate (0,0) alle coordinate (100,100) è necessario un:

```
OPEN 1,6,1:PRINT#1,"D",100,100
per usare la penna verde:
OPEN 2,6,2:PRINT#2,2
per passare al modo 80 colonne:
OPEN 3,6,3:PRINT#3,0
```

e così via per le varie altre funzioni. Giudicate voi se non è più bello digitare semplicemente:

```
DRAW 100,100
GREEN
SIZE 80
```

... della serie: "e chi s'è visto, s'è visto!".

```
212 FOR I=49318T049355:READ I1:POKE I, I1:NEXT
222 FOR I=51536T051982:READ I1:POKE I, I1:NEXT
232 FOR I=52826T052904:READ I1:POKE I, I1:NEXT
242 FOR I=49248T049279:READ I1:POKE I, I1:NEXT
300 ++++++
310 +
320 + *** **** *****
330 + * * * * *
340 + *****
350 + * * * * *
360 + * * **** *
370 +
380 +
390 + (C) 1984 ADP SOFTWARE
400 +
410 +
500 ++++++
4202 DATA25,252,96,192,42,198,42,198,137,201,236,201,251,201,4,202,22,202,31
4212 DATA202,97,202,118,202,135,202,139,202,143,202,147,202,197,202,230,202,249
4222 DATA202
4252 REM *****
4302 DATA32,204,255,32,231,255,169,70,162,6,160,0,32,186,255,169,0,32,189,255
4312 DATA32,192,255,96,168,9,80,72,162,6,32,186,255,169,0,32,189,255,32,192,255
4322 DATA104,170,32,201,255,96,32,204,255,9,80,32,195,255,76,240,168,32,121,0
4332 DATA201,145,240,19,201,224,240,26,201,223,240,36,201,225,240,47,201,226
4342 DATA240,58,76,43,198,32,80,201,162,70,32,201,255,76,248,168,169,13,32,210
4352 DATA255,32,204,255,32,231,255,76,248,168,169,7,32,104,201,169,0,32,210,255
4362 DATA169,7,76,127,201,169,6,32,104,201,169,49,32,210,255,169,6,76,127,201
4372 DATA169,6,32,104,201,169,48,32,210,255,169,6,76,127,201,234,234,234,169
4382 DATA1,32,104,201,169,72,32,210,255,169,1,76,127,201,169,1,32,104,201,169
4392 DATA73,208,239,169,1,32,104,201,169,77,32,210,255,32,76,127,201,169,1,76,127
4402 DATA201,169,1,32,104,201,169,68,208,236,32,121,0,201,230,240,7,201,231,240
4412 DATA15,76,43,198,32,115,0,169,1,32,104,201,169,82,208,210,32,115,0,169,1
4422 DATA32,104,201,169,74,208,198,32,158,173,32,221,189,32,30,171,32,121,0,32
4432 DATA210,255,32,115,0,32,158,173,32,221,189,32,30,171,96,169,2,32,104,201
4442 DATA32,158,183,138,41,3,9,48,32,210,255,169,2,76,127,201,169,48,72,169,2
4452 DATA32,104,201,184,32,210,255,169,2,76,127,201,169,49,208,237,169,50,208
4462 DATA233,169,51,208,229,32,158,183,138,201,10,240,17,201,20,240,13,201,40
4472 DATA240,9,201,80,240,5,162,14,76,139,227,162,254,232,10,144,252,138,72,169
4482 DATA3,32,104,201,184,170,169,0,32,205,189,169,3,76,127,201,32,121,0,201
4492 DATA145,240,7,201,224,240,3,76,43,198,41,1,9,48,72,169,4,32,104,201,184
4502 DATA32,210,255,169,4,76,127,201,169,5,32,104,201,32,158,173,32,221,189,32
4512 DATA30,171,169,5,76,127,201,32,80,201,162,70,32,201,255,169,13,32,160,170
4522 DATA32,204,255,169,70,76,127,201
4552 REM *****
4602 DATA62,69,83,69,212,79,70,198,83,77,65,76,204,67,65,208,80,204,72,79,77
4612 DATA197,83,69,212,77,79,86,197,68,82,65,215,82,69,204,67,79,76,79,210,66
4622 DATA76,65,67,203,66,76,85,197,71,82,69,69,206,82,69,196,83,73,90,197,82
4632 DATA79,84,65,84,197,66,82,79,75,69,206,87,82,73,84,197
4652 REM *****
4702 DATA96,169,167,141,7,3,141,9,3,169,165,141,5,3,169,124,141,4,3,169,26,141
4712 DATA6,3,169,228,141,8,3,76,248,168
```

Listato Basic - Linee da aggiungere a quelle dei numeri scorsi per implementare i comandi presentati questo mese.

e spedisce al plotter un carattere compreso tra "0" e "3". Per fare questo, si avvale della routine del sistema operativo locata a partire dall'indirizzo \$B79E: preleva dalla linea che si sta eseguendo il parametro (valutando eventualmente un'espressione) e lo pone in X. Segue una opportuna mascherazione per risalire al codice ASCII del parametro (riportato all'intervallo 0...3 grazie a AND \$03). Dopo aver inviato il carattere così ottenuto al plotter, chiude canale e file aperto.

I listati 8, 9, 10 e 11 implementano rispettivamente i comandi BLACK, BLUE, GREEN, RED spedendo al plotter i caratteri "0", "1", "2" o "3" sempre dopo aver aperto un file di indirizzo secondario 2.

Il comando SIZE, implementato dal listato 12, per prima cosa controlla la regolarità del parametro passato. Se questo è diverso da 10, 20, 40 o 80 un messaggio ILLEGAL QUANTITY ERROR invaderà il vostro schermo. Dal byte \$CAAD in poi, una semplice routine trasforma 80 in 0, 40 in 1, 20 in 2 o 10 in 3: ciò perché tale è il parametro da inviare al plotter per cambiare formato di caratteri. Segue l'invio

vero e proprio, questa volta sfruttando la routine del sistema operativo \$BDCD che spedisce un numero contenuto in A (parte alta) e in X (parte bassa).

ROTATE (listato 13) grazie a JSR\$0079 controlla se è seguito da ON o OFF e a seconda dei due casi, dopo aver aperto un file di indirizzo secondario 4, invia al plotter un "1" o uno "0".

Il comando BROKEN, dopo aver aperto un file di indirizzo secondario 5, esegue tre routine del sistema operativo. Con la prima valuta l'espressione che segue il comando BROKEN (anche un numero è un'espressione). La seconda converte tale valore in una sequenza di codici ASCII (es. il numero 123 nella stringa "123"). La terza la invia alla periferica.

L'ultimo listato L.M. implementa il comando WRITE che, come visto, permette un output diretto su plotter. Le prime tre linee servono per definire il plotter come periferica di output. JSR\$AAA0 è un salto all'interpretazione dell'istruzione PRINT del Basic standard del Commodore 64. Segue la chiusura del canale e file aperti per l'uso.