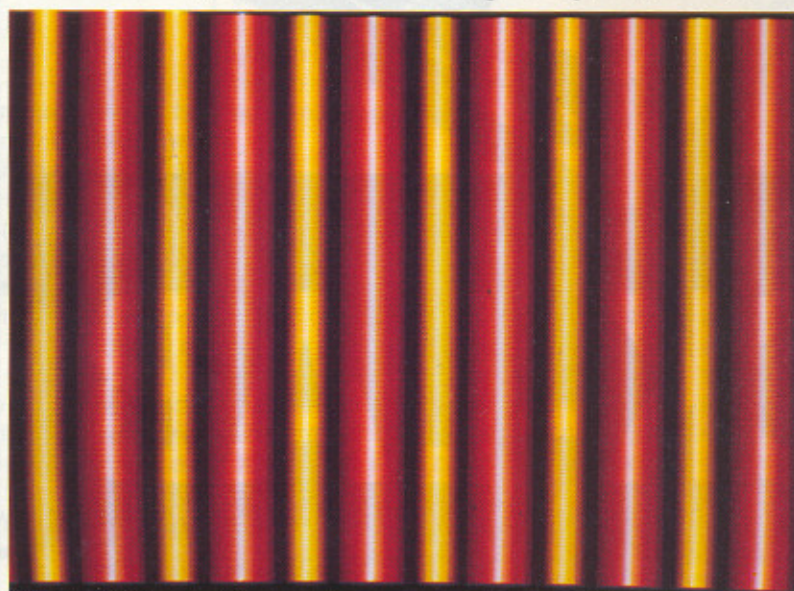


I DUE BASIC



del

PC 128

ULTIMA PUNTATA

Con questa puntata si conclude la serie di articoli dedicati ai due Basic del PC 128, articoli che ci hanno seguiti fin dalla nostra prima uscita in edicola.

Nella speranza che il nostro sforzo vi sia stato di una qualche utilità, vi diamo ora appuntamento per scoprire assieme altre meraviglie del PC 128.

SQR

TIPO: Funzione matematica

SINTASSI: SQR(n.)

COMMENTO:

Dove n. e' un numero o una variabile

numerica, positivo.

La funzione SPQ(), ritorna la radice quadrata dell'argomento.

Se n. e' un numero negativo, il computer segnala un errore di tipo "Illegal Function Call".

```
10 CLS
20 INPUT "Immettere un numero ";A
30 IF A<0 THEN PRINT "E' un numero negativo,
ripetilo!";GOTO 20
40 CLS
50 PRINT "La radice quadrata di";A;
"e' ";PRINT
60 PRINT SQR(A)
70 END
```

STEP

TIPO: Funzione

SINTASSI: STEP(n.)

COMMENTO:

La funzione STEP, se usata con i comandi grafici (BOX, BOXF, CIRCLE, CIRCLEF, LINE, PAINT PSET), permette di attuare uno spostamento relativo, calcolato sull'ultimo punto definito. Gli spostamenti possono essere sia negativi che positivi.

```
10 CLS
20 BOXF(100,100)-STEP(-50,50)
```

```
BOXF(200,150)-STEP(40,40)
30 END
```

STICK

TIPO: Funzione

SINTASSI: STICK(n.)

COMMENTO:

Dove n. e' un numero che puo' valere 0 o 1.

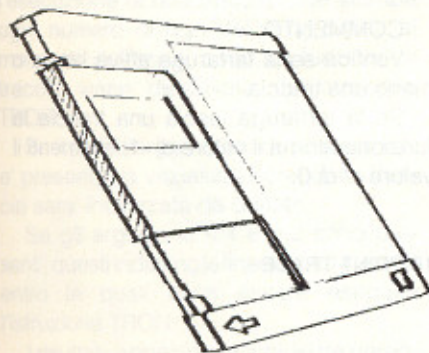
La funzione STICK() ritorna la direzione del joystick specificato in argomento.

I valori ritornati dalla funzione sono:

- 0 neutro
- 1 90o
- 2 45o
- 3 0o
- 4 -45o
- 5 -90o
- 6 -135o
- 7 180o
- 8 135o

Vediamo un piccolo programma dimostrativo:

```
10 CLS
```



```
20 X=19
30 Y=12
40 A=STICK(0)
50 IF STRIG(0)=-1 THEN END
60 LOCATE X,Y
70 PRINT " "
80 ON A GOSUB 160,180,210,230,260,280,
310,330
90 IF X<0 THEN X=0
```

```
100 IF X.39 THEN X=39
110 IF Y<0 THEN YY=0
120 IF Y>24 THEN Y=24
130 LOCATE X,Y
140 PRINT"X";
150 GOTO 40
160 Y=Y-1
170 RETURN
180 Y=Y+1
190 X=X+1
200 RETURN
210 X=X+1
220 RETURN
230 Y=Y+1
240 X=X+1
250 RETURN
260 Y=Y+1
270 RETURN
280 Y=Y+1
290 X=X-1
300 RETURN
310 X=X-1
320 RETURN
330 Y=Y-1
340 X=X-1
350 RETURN
```

STOP

TIPO: Istruzione

SINTASSI: STOP

COMMENTO:

Sospende temporaneamente l'esecuzione del programma, e fa apparire sullo schermo il messaggio "Break in ...".

E' possibile riprendere l'esecuzione del programma, per mezzo dell'istruzione CONT, oppure con GOTO.

STR\$

TIPO: Funzione

SINTASSI: STR\$(n.)

COMMENTO:

Dove n. e' un numero o una variabile numerica consentita dal Basic.

L'istruzione STR\$, converte il numero

in argomento in una stringa.

La funzione inversa di STR\$ e' VAL.

```
10 CLS
20 A$=STR$(1234567890)
30 PRINT A$
40 END
```



STRIG

TIPO: Funzione

SINTASSI: STRIG(n.)

COMMENTO:

Dove n. e' un numero o una variabile, il cui valore e' 0 oppure 1.

Ritorna lo stato del pulsante del joystick in argomento.

I valori di ritorno della funzione STRIG() sono:

- 1 pulsante premuto
- 0 pulsante non premuto.

IF STRIG(0)=-1 THEN END

STRING\$

TIPO: Funzione

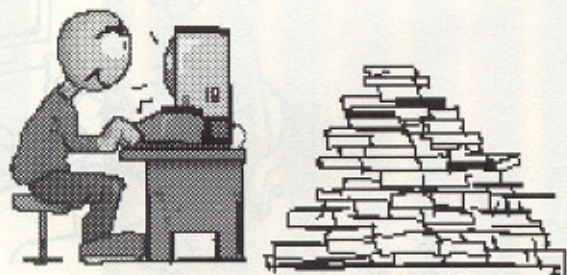
SINTASSI: STRING\$(n.,car.)

COMMENTO:

Dove n. e' un numero o una variabile permessa dal Basic; car. e' un carattere o il numero ASCII corrispondente.

La funzione fornisce una stringa di lunghezza n. i cui caratteri sono car., o hanno il codice ASCII di car.

```
10 CLS
20 FOR A=1 TO 10
30 FOR B=65 TO 75
40 PRINT STRING$(A,B)
```



```
50 NEXY B
60 NEXT A
70 END
```

SWAP

TIPO: Istruzione

SINTASSI: SWAPvar1,var2

COMMENTO:

Dove var1 e var2 sono due variabili o elementi di matrici.

Scambia il valore di due variabili dello stesso tipo.

E' possibile scambiare qualsiasi tipo di variabile (intero, precisione singola o doppia, stringa), ma le due variabili devono essere dello stesso tipo.

```
10 CLS
20 A$="10": B$="20": C$="X"
30 PRINT A$;C$;B$
40 SWAP A$,B$
50 PRINT A$;C$;B$
60 END
```

TAB

TIPO: Funzione

SINTASSI: TAB(n.)

COMMENTO:

Dove n. e' un numero o una variabile, consentito dal Basic.

La funzione TAB() permette di posizionare il cursore nella colonna indicata dall'argomento. Se tale colonna si trova a sinistra dell'attuale posizione del cursore, si avra' un ritorno di carrello. Se n. e' un valore nullo o negativo, la funzione non dara' alcun effetto.

Se si sta lavorando con caratteri in doppia larghezza, la funzione TAB() ne tiene conto.

I separatori, virgola e punto e virgola, vengono usati come per l'istruzione PRINT normale.

```
10 CLS
20 PRINT "Ciao"; TAB(15);"Pippo"
30 PRINT "Ciao", TAB(15)"Pippo"
40 END
```

TAN

TIPO: Funzione matematica

SINTASSI: TAN(n.)

COMMENTO:

Dove n. e' un angolo espresso in radianti. Per convertire i gradi in radianti, moltiplicare n. per P/180, dove P=3,141593.

Calcola la tangente trigonometrica di n.

```
10 CLS
20 SCREEN1,6,6
30 LINE(0,100)-(319,100),2
40 LINE(0,50)-(319,50),4
50 LINE(0,150)-(319,150),4
60 FOR I=0 TO 319
70 AN=I/30
80 Y=100-50*TAN(AN)
90 IF Y<0 OR Y>199 THEN 110
100 PSET(I,Y),1
110 NEXT I
120 END
```

TRACE

TIPO: Funzione

SINTASSI: TRACE

COMMENTO:

Verifica se la tartaruga attiva lascia o meno una traccia.

Se la tartaruga lascia una traccia la funzione ritorna il valore di -1, altrimenti il valore e' di 0.

```
10 PRINT TRACE
```

TRACE

TIPO: Istruzione

SINTASSI: TRACEn.

COMMENTO:

Dove n. e' un numero che vale 0 oppure 1.
Permette di attivare (n.=1), oppure

disattivare (n.=0), la traccia lasciata dalla tartaruga attiva durante i suoi spostamenti.

TRACE 1

TROFF

TIPO: Istruzione

SINTASSI: TROFF

COMMENTO:

Annulla l'effetto dell'istruzione TRON, e pertanto chiude la traccia e, se aperto, il file di traccia.

TRON

TIPO: Istruzione

SINTASSI: TRONdescrittore file,n.1-n.2

COMMENTO:

Esegue la traccia dell'esecuzione delle specifiche di programma.

Utile come aiuto per la rivelazione degli errori, il comando TRON consente l'esecuzione di una traccia che stampa ogni numero di riga del programma al momento dell'esecuzione. La funzione di traccia viene disattivata dall'istruzione TROFF.

Se il primo argomento, descrittore file, e' presente, la visualizzazione della traccia sara' indirizzata da questo.

Se gli argomenti n.1 e n.2 sono presenti, questi indicano le linee di programma entro le quali deve essere eseguita l'istruzione TRON.

I risultati, appaiono racchiusi tra parentesi quadre.

```
10 CLS
110 TRON
120 IF A>0 THEN IF A>100 THEN 150 ELSE
160 ELSE 170
150 PRINT "A100":END
160 PRINT "0<A<100":END
170 PRINT "A<=0"
180 TROFF
```

190 END

TUNE

TIPO: Istruzione

SINTASSI: TUNE

COMMENTO:

Permette la regolazione della penna ottica; nel Basic 1.

Una volta attivata l'istruzione, ponendo la penna ottica a contatto dello schermo, su questo apparira' una linea verticale: se questa non coincide esattamente con la posizione della penna, la si deve spostare per mezzo dei tasti cursore.



TURTLE

TIPO: Istruzione

SINTASSI: TURTLEn.,cl.,ri.,str.

COMMENTO:

Dove n. e' un numero o una variabile numerica, ed indica il numero della tartaruga attiva; cl. e ri., rispettivamente colonna e riga, determinano le coordinate della posizione della tartaruga sullo scher-

mo, e sono compresi tra -32768 e 32768. Questi due parametri non sono obbligatori, e nel caso vengano omessi, la posizione della tartaruga non viene cambiata, oppure, se la tartaruga e' appena stata attivata, questa sara' visualizzata nel centro dello schermo.

Il quarto parametro, str., e' una stringa, o una variabile stringa, di caratteri che permette di definire la forma della tartaruga. All'interno, la stringa contiene delle coppie di valori che hanno la funzione di tracciare la forma della tartaruga stessa.

Le coppie di caratteri, contenuti dal quarto parametro, possono essere solo di quattro tipi: RaDn; RaUn; LaDn; LaUn Dove: a indica un angolo di rotazione; n indica l'entita' dello spostamento. Ambedue questi parametri, devono essere compresi in un campo di valori che va da 0 a 255. Per il parametro a, il valore di 256

corrisponde a 360 gradi.

R Indica rotazione a destra.

L Indica rotazione a sinistra.

D Indica spostamento con traccia.

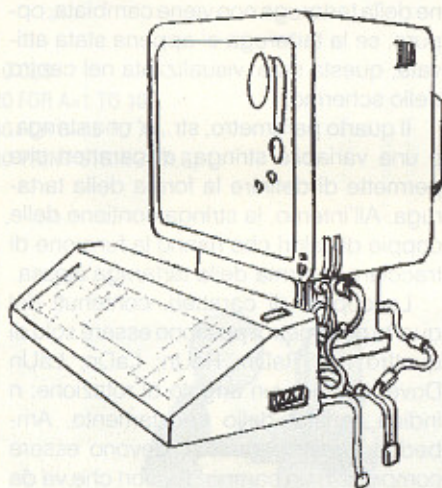
U Indica spostamento senza traccia.

L'istruzione TURTLE, permette di attivare e posizionare una tartaruga attiva.

```
10 CLS
20 TURTLE1,180,120,"L0D30L90D30L83D26"
30 SHOW1
40 END
```

Visualizza una tartaruga a forma trian-

golare, posta nelle coordinate 180, 120.



UNLOAD

TIPO: Istruzione

SINTASSI: UNLOADn.drive

COMMENTO:

Chiude tutti i file aperti sul dischetto contenuto dal drive specificato in n.drive, e ricopia tutte le informazioni utili sul dischetto.

Il drive di default e' il drive 0, oppure l'ultimo definito per mezzo dell'istruzione DEVICE.

UNLOAD1 chiude tutti i file aperti sul dischetto inserito nel drive 1.

USR

TIPO: Funzione

SINTASSI: USRn.(dati)

COMMENTO:

Dove n. e' un numero compreso tra 0 e 9 e corrisponde alla cifra fornita con la specifica DEFUSR per il programma desiderato. Se si omette n., si assume USR0.

La funzione USR viene usata per richiamare il sottoprogramma in linguaggio

macchina definito dall'utente.

Si fa presente che l'uso approfondito di questa, come di altre funzioni, non puo' far parte di questa trattazione, pertanto se ne consiglia l'uso solo ai programmatori esperti.

VAL

TIPO: Funzione

SINTASSI: VAL(str.)

COMMENTO:

Ritorna il valore dei primi numeri della stringa str. Questo a condizione che nessun segno (esclusi il - e il +), o carattere, preceda il numero.

```
10 CLS
20 PRINT VAL("0234X4")
30 PRINT VAL("-7,4ZE")
40 PRINT VAL("A123")
50 PRINT VAL("901")
60 PRINT VAL("-3.45P3")
70 PRINT VAL("+3.45P3")
80 END
```



VARPTR

TIPO: Funzione

SINTASSI: VARPTR(var.)

COMMENTO:

Dove var e' il nome di una variabile: numerica; stringa o dell'elemento di schiera del programma.

Il risultato della funzione e' un numero compreso tra -32768 e 32768. Quando il risultato e' un numero negativo, e' necessario aggiungere 65536 per ottenere l'indirizzo reale.

I valori sopra descritti, si riferiscono a un indirizzo, della memoria Basic, relativo al primo byte del dato identificato con var.

Solitamente VARPTR e' usato per ottenere l'indirizzo di una variabile o di una array da passare a sotto programmi in linguaggio macchina.

```
10 CLS
20 A=12345:AS="CIAO PIPPO":B=4567
30 PRINT VARPTR(A)
40 PRINT VARPTR(AS)
50 PRINT VARPTR(B)
60 END
```

VERIFY

TIPO: Istruzioni

SINTASSI: VERIFY ON
VERIFY OFF

COMMENTO:

Per mezzo del comando VERIFY ON il computer controlla tutti i dati contenuti su un dischetto. Tramite VERIFY OFF, invece l'opzione viene disattivata.

WAIT

TIPO: Istruzione

SINTASSI: WAIT indirizzo, maschera1, maschera2

COMMENTO:

Permette di sospendere l'esecuzione di un programma fino a quando non appare una precisa configurazione di bit all'indirizzo in argomento.

La lettura dei dati e il modo di usarli e' il seguente:

-Viene fatto un'operazione XOR tra il contenuto dell'indirizzo e la maschera 2.

-Viene fatta un'operazione AND tra il contenuto dell'indirizzo e la maschera 1.

Il programma prosegue se queste espressioni danno un risultato nullo.

Dopo questa semplice e semplicistica spiegazione, e' opportuno specificare che anche questa istruzione fa parte di quel campo di comandi utilizzabile solo dagli esperti, e pertanto esula un po' dagli intenti della nostra trattazione.

WINDOW

TIPO: Istruzione

SINTASSI: WINDOWA (cl.1,ri.1)-(cl.2,ri2)

COMMENTO:

Dove cl.1 e ri.1, sono dei numeri o delle variabili numeriche consentite dal Basic, che rappresentano le coordinate di un punto, corrispondente al vertice superiore sinistro di un rettangolo. L'angolo inferiore destro, invece, viene identificato dalla copia di valori contenuta in cl.2 e ri.2.

Il valore di cl. e' definito in un campo che va da 0 a 319 (oppure 639), mentre il

valore di ri. e' compreso tra 0 e 199.

Per mezzo dell'istruzione WINDOW, si puo' definire una finestra di visualizzazione grafica.

Una volta specificata l'ampiezza della finestra, le istruzioni di tracciamento grafico, saranno visualizzabili solo al suo interno.

10 CLS

20 WINDOW(100,100)-(200,180)

30 CIRCLEF(130,130),49,2

40 END

WRITE#

TIPO: Istruzione

SINTASSI: WRITE#n.canale,elenco dati

COMMENTO:

Dove n.canale e' il numero con il quale e' stato aperto il file per l'emissione;

elenco dati, e' una lista di espressioni stringa e/o numeriche separate da virgole.

L'istruzione WRITE# permette di scrivere i dati in un file sequenziale.

Nel caso di file ad accesso diretto, WRITE# mette i dati nel buffer di registrazione. Il trasferimento poi, viene fatto per mezzo della specifica PUT.

La differenza tra WRITE# e PRINT#, e' che WRITE# inserisce le virgole tra gli elementi quando vengono scritti e delimita le stringhe con apici.

WRITE#1,Prova1,PROVE

ZOOM

TIPO: Funzione

SINTASSI: ZOOM

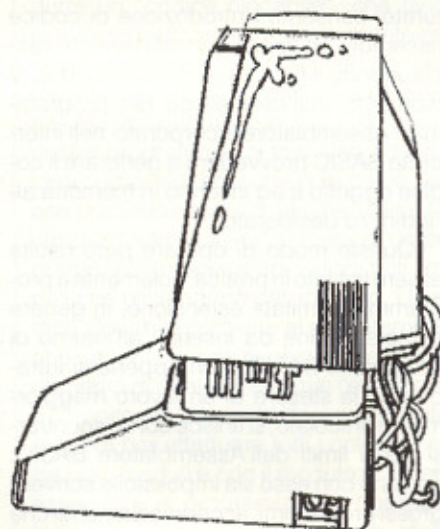
COMMENTO:

Ritorna la dimensione della tartaruga attiva. Tale risultato e' un numero compreso tra 0 e 255.

Per default, la dimensione di una tarta-

ruga definita per mezzo della specifica TURTLE, e' 16.

PRINT ZOOM



ZOOM

TIPO: Istruzione

SINTASSI: ZOOM TO n.

COMMENTO:

Fissa o modifica le dimensioni della tartaruga attiva.

Se TO e' presente la dimensione della tartaruga e' determinata in modo assoluto, se invece TO e' omissa, la dimensione della tartaruga viene incrementata del valore in argomento.

Il valore di n. e quindi la dimensione, puo' variare da 0 a 255. Seguendo il valore del secondo parametro di SHOW, la dimensione della tartaruga viene modificata immediatamente (se questo valore e' uguale a 1, ossia corrispondente a tartaruga libera), oppure solo dopo FWD o TURTLE.

TURTLE1: ZOOM TO 25 Fissa la dimensione della tartaruga 1 a 25.

ZOOM 30 Aumenta di 30 la dimensione della tartaruga attiva.