

Librería estándar CPC Basic

ABS!

Función

Sintaxis: ABS!(<valor>)

Descripción: Devuelva el valor absoluto del <valor> dado.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(ABS!(-67.98), "_0.00")
END
```

AFTER

Orden

Sintaxis: AFTER <tiempo>,<etiqueta>,<buffer>

Descripción: Crea una interrupción que se ejecutará tras <tiempo> cincuentavos de segundo, llamando a la rutina identificada por <etiqueta>. Debe darse un espacio de 13 bytes señalado por <buffer> para el trabajo interno de la interrupción.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM a$, indicador#, r%, buffer_after#(12)
AFTER 250, rutina, @buffer_after#(0):CLS
PRINT "Adivine una letra en 5 segundos":PRINT
inicio
a$=INKEY$:IF indicador#=1 THEN
r%=REMAIN%(@buffer_after#(0)):END
IF A$<>CHR$(INT&(RND!()*26+97)) THEN GOTO inicio
PRINT a$+" es correcto. Usted gana."
SOUND 1,478:SOUND 1,358:r%=REMAIN%(@buffer_after#(0)):END
rutina
PRINT "demasiado tarde. Yo gano."
SOUND 1,2000:indicador#=1:RETURN
```

ASC%

Función

Sintaxis: ASC%(<texto>)

Descripción: Devuelve el código ASCII del primer carácter del <texto> especificado.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(ASC%("x"), "__0")
END
```

ATN!

Función

Sintaxis: ATN!(<valor>)

Descripción: Devuelve el resultado de buscar el arcotangente de <valor>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(ATN!(1), "0.000000000")
END
```

BIN\$

Función

Sintaxis: BIN\$(<valor>[,<dígitos>])

Descripción: Devuelve una cadena que representa el <valor> dado en código binario, con tantos dígitos como se indique en <dígitos>. Si no se indica este valor se la cadena tendrá sólo los dígitos imprescindibles para representar el <valor>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT BIN$(64, 8)
END
```

BORDER

Orden

Sintaxis: BORDER <color1>[,<color2>]

Descripción: Establece el borde de la pantalla a los colores especificados, parpadenado entre ellos. Si no se indica <color2> se considera el mismo que <color1>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

REM 729 combinaciones para el borde!
DIM a#,b#,t
SPEEDINK 5,5
FOR a#=0 TO 26
  FOR b#=0 TO 26
    BORDER a#,b#:CLS:LOCATE 14,13
    PRINT "border "+DEC$(a#,"_0")+"," "+DEC$(b#,"_0")
    FOR t=1 to 1000:NEXT
  NEXT
NEXT
NEXT
END
```

CALL

Orden

Sintaxis: CALL <dirección>

Descripción: Ejecuta la rutina en código máquina que se encuentra en la <dirección> indicada.

Ejemplo: CHAIN "CPCBasic.BAS"

```
CALL 0
END
```

CBYTE#

Función

Sintaxis: CBYTE#(<valor>)

Descripción: Convierte el <valor> dado a tipo byte. Se puede perder precisión en la conversión si el resultado es mayor que la capacidad de un byte.

Ejemplo: (Sin ejemplo)

CHR\$

Función

Sintaxis: CHR\$(<código>)

Descripción: Devuelve una cadena formada por el carácter cuyo código se especifica.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFINT A-Z
DIM x
FOR x=32 TO 255
  PRINT DEC$(x, "##0 ") + CHR$(x) + SPACE$(3)
NEXT
END
```

CINT%

Función

Sintaxis: CINT%(<valor>)

Descripción: Convierte el <valor> dado a tipo entero. Se puede perder precisión en la conversión si el resultado es mayor que la capacidad de un entero.

Ejemplo: CHAIN "CPCBasic.BAS"

DIM n
n=1.9999
PRINT DEC\$(CINT%(n), "0")
END

CLEARINPUT

Orden

Sintaxis: CLEARINPUT

Descripción: Vacía el buffer del teclado.

Ejemplo: CHAIN "CPCBasic.BAS"

DIM t
CLS
PRINT "Teclee letras ahora!":PRINT
FOR t=1 TO 6000
NEXT
CLEARINPUT
END

CLG

Orden

Sintaxis: CLG [<tinta>]

Descripción: Borra la pantalla gráfica con el color del papel de gráficos actual. Si se especifica <tinta> se establece previamente este color para el papel de gráficos.

Ejemplo: CHAIN "CPCBasic.BAS"

LOCATE 1,20
CLG 3
END

CLNG&

Función

Sintaxis: CLNG(<valor>)

Descripción: Convierte el <valor> dado a tipo long. Se puede perder precisión en la conversión si el resultado es mayor que la capacidad de un long.

Ejemplo: (Sin ejemplo)

CLOSEIN

Orden

Sintaxis: CLOSEIN

Descripción: Cierra el fichero de entrada que se encuentre abierto actualmente.

Ejemplo: (Sin ejemplo)

CLOSEOUT

Orden

Sintaxis: CLOSEOUT

Descripción: Cierra el fichero de salida que se encuentra abierto actualmente.

Ejemplo: (Sin ejemplo)

CLS

Orden

Sintaxis: CLS [<canal>]

Descripción: Borra la ventana de texto especificada por <canal> con el color del papel de texto actual. Si éste no se indica se borra la ventana de cauce 0.

Ejemplo: CHAIN "CPCBasic.BAS"

```
PAPER 3,2  
CLS 2  
END
```

COPYCHR\$

Función

Sintaxis: COPYCHR\$([<cauce>])

Descripción: Devuelve una cadena formada por el carácter que se encuentra en la posición del cursor de texto en el <cauce> indicado. Si no se especifica <cauce> se interpreta que es el 0.

Ejemplo: CHAIN "CPCBasic.BAS"

DIM a\$
CLS
PRINT "Extremo superior izquierdo"
LOCATE 1,1
a\$=COPYCHR\$(0)
LOCATE 1,20
PRINT a\$
END

COS!

Función

Sintaxis: COS!(<valor>)

Descripción: Devuelve el valor del coseno correspondiente a <valor>.

Ejemplo: CHAIN "CPCBasic.BAS"

DEG
PRINT DEC\$(COS!(45), "0.00000000")
END

CREAL!

Función

Sintaxis: CREAL!(<valor>)

Descripción: Convierte el <valor> dado a tipo real.

Ejemplo: CHAIN "CPCBasic.BAS"

DIM a
a=PI!()
PRINT DEC\$(CINT%(a), "0"):PRINT
PRINT DEC\$(CREAL!(a), "0.00000000")
END

CURSOR

Orden

Sintaxis: CURSOR [<sistema>][,<usuario>]

Descripción: Activa o desactiva el cursor de texto. Si los valores de <sistema> y <usuario> son 1 el cursor se visualiza, si alguno de dichos valores es 0 el cursor deja de visualizarse.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM a$
CURSOR 1
PRINT "pregunta?"
inicio
a$=INKEY$():IF a$="" THEN GOTO inicio
PRINT a$
CURSOR 0
END
```

DEC\$

Función

Sintaxis: DEC\$(<valor>,<plantilla>)

Descripción: Devuelve una cadena que representa <valor> según la plantilla dada, que puede contener los siguientes símbolos:

- # representa un dígito si es significativo o espacio en blanco si no lo es.
- _ representa un dígito si es significativo o nada si no lo es.
- 0 representa un dígito tanto si es significativo como si no lo es.
- + representa el signo de <valor>.
- representa el signo de <valor> si es negativo o nada si no lo es.

Cualquier otro carácter se representará en el resultado tal cual.

No se deben especificar plantillas con más de 10 símbolos que representen dígitos.

Ejemplo: (Sin ejemplo)

DEG

Orden

Sintaxis: DEG

Descripción: Establece el modo para las funciones trigonométricas a grados sexagesimales.

Ejemplo: (Sin ejemplo)

DI

Orden

Sintaxis: DI

Descripción: Inhibe las interrupciones.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFINT A-Z
DIM X1,X2,X,Y,C$,buffer_every#(12)
CLS:TAG:EVERY 10,rutina,@buffer_every#(0)
inicio
X1=RND!()*320:X2=RND!()*200
Y=200+RND!()*200:C$=CHR$(RND!()*255)
FOR X=320-X1 TO 320+X2 STEP 4
DI
MOVE 320,0,1:MOVE X-2,Y:MOVE X,Y
PRINT " "+C$:FRAME
EI:NEXT:GOTO inicio
rutina
MOVE 320,0:DRAW X+8,Y-16,0:RETURN
END
```

DRAW

Orden

Sintaxis: DRAW <x>,<y>[,<tinta>][,<modo>]

Descripción: Dibuja una recta desde la posición actual del cursor gráfico hasta <x>,<y> con el color de tinta gráfica actual. Si se especifica <tinta> se establece previamente la tinta gráfica a dicho color. Si se especifica <modo> se establece previamente el modo de dibujo al especificado (0 para normal, 1 para XOR, 2 para AND y 3 para OR).

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFINT A-Z
DIM x,y,z
MODE 0:BORDER 0:PAPER 0:INK 0,0
inicio
x=RND!()*640:y=RND!()*400:z=RND!()*15
DRAW x,y,z
GOTO inicio
```

DRAWR

Orden

Sintaxis: DRAWR <x>,<y>[,<tinta>][,<modo>]

Descripción: Dibuja una recta desde la posición actual del cursor gráfico hasta esta misma posición incrementada en <x> e <y> unidades, con el color de tinta gráfica actual. Si se especifica <tinta> se establece previamente la tinta gráfica a dicho color. Si se especifica <modo> se establece previamente el modo de dibujo al especificado (0 para normal, 1 para XOR, 2 para AND y 3 para OR).

Ejemplo: CHAIN "CPCBasic.BAS"

DIM n#
CLS:PRINT "escalera"
MOVE 0,350:FOR n#=1 TO 8
DRAWR 50,0
DRAWR 0,-50
NEXT:MOVE 348,0:FILL 3
bucle:GOTO bucle

EI

Orden

Sintaxis: EI

Descripción: Activa las interrupciones.

Ejemplo: (Sin ejemplo)

ENT

Orden

Sintaxis: ENT <número>[,<escalones1>,<altura1>,<pausa1>][,<escalones2>,<altura2>,<pausa2>][,<escalones3>,<altura3>,<pausa3>][,<escalones4>,<altura4>,<pausa4>][,<escalones5>,<altura5>,<pausa5>]

Descripción: Define la envolvente de tono especificada por <número> (de 1 a 15) con los valores indicados por cada grupo de valores indicado por <escalones> (de 0 a 239), <altura> (de -128 a +127) y <pausa> (de 0 a 255), pudiendo definir hasta 5 grupos de estos valores.

Ejemplo: CHAIN "CPCBasic.BAS"

ENT 1,10,-50,10,10,50,10
SOUND 1,500,200,10,,1
END

ENV

Orden

Sintaxis: ENV <número>[,<escalones1>,<altura1>,<pausa1>][,<escalones2>,<altura2>,<pausa2>][,<escalones3>,<altura3>,<pausa3>][,<escalones4>,<altura4>,<pausa4>][,<escalones5>,<altura5>,<pausa5>]

Descripción: Define la envolvente de volumen especificada por <número> (de 1 a 15) con los valores indicados por cada grupo de valores indicado por <escalones> (de 0 a 239), <altura> (de -128 a +127) y <pausa> (de 0 a 255), pudiendo definir hasta 5 grupos de estos valores.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

ENV 1,15,-1,10,15,1,10
SOUND 1,200,300,15,1
END
```

EOF#

Función

Sintaxis: EOF#()

Descripción: Devuelve -1 ó 0 respectivamente según se haya alcanzado o no el final de fichero abierto para lectura.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM a$
OPENIN "ex1.bas"
WHILE NOT EOF#()
  a$=INPUT$(9)
  PRINT a$:PRINT
WEND
CLOSEIN
END
```

ERR#

Función

Sintaxis: ERR#()

Descripción: Devuelve 0 ó -1 respectivamente según haya sido correcta o incorrecta la última operación realizada en disco entre las siguientes: OPENIN, CLOSEIN, LOAD, SAVE, INPUT\$.

Ejemplo: (Sin ejemplo)

EVERY

Orden

Sintaxis: EVERY <tiempo>, <etiqueta>, <buffer>

Descripción: Crea una interrupción que se ejecutará cada <tiempo> cincuentavos de segundo, llamando a la rutina identificada por <etiqueta>. Debe darse un espacio de 13 bytes señalado por <buffer> para el trabajo interno de la interrupción.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM buffer_every#(12)
EVERY 50,rutina,@buffer_every#(0)
bucle:GOTO bucle
rutina
SOUND 1,20
RETURN
```

EXP!

Función

Sintaxis: EXP!(<valor>)

Descripción: Devuelve el valor de la constante e elevada a <valor>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(EXP!(6.876),"__0.000000")
END
```

FILL

Orden

Sintaxis: FILL <tinta>

Descripción: Rellena la zona delimitada de la pantalla desde la posición actual del cursor gráfico con el valor de <tinta> especificado.

Nota: no válido para model CPC464.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM n%,colorpluma#
MODE 0
FOR n%=1 TO 500
PRINT "O"
NEXT
inicio
colorpluma#=2+RND!()*13
FILL colorpluma#
GOTO inicio
```

FIX&

Función

Sintaxis: FIX&(<valor>)

Descripción: Devuelve la parte entera de <valor>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(FIX&(9.99999),"0")
END
```

FRAME

Orden

Sintaxis: FRAME

Descripción: Sincroniza la escritura de caracteres gráficos con el refresco de pantalla.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM x%,f#
iniciol
MODE 0
PRINT "sin FRAME"
inicio2
TAG
MOVE 0,200
FOR x%=0 TO 500 STEP 4
  IF f#=1 THEN FRAME
  MOVE x%,200
  PRINT " "+CHR$(143)
NEXT
IF f#=1 THEN f#=0:GOTO iniciol
CLS
TAGOFF
PRINT "con FRAME"
f#=1
GOTO inicio2
```

FRE%

Función

Sintaxis: FRE%()

Descripción: Devuelve la cantidad en bytes de memoria dinámica que queda libre para almacenamiento de cadenas, evaluación de expresiones, etc.

Ejemplo: CHAIN "CPCBasic.BAS"

PRINT DEC\$(FRE%(), "____0")
END

GRAPHICSPAPER

Orden

Sintaxis: GRAPHICSPAPER <color>

Descripción: Establece el color de papel gráfico actual a indicado en <color>.

Ejemplo: CHAIN "CPCBasic.BAS"

MODE 0
MASK 15
GRAPHICSPAPER 3
DRAW 640,0
END

GRAPHICSPEN

Orden

Sintaxis: GRAPHICSPEN <color>

Descripción: Establece el color de tinta gráfica actual a indicado en <color>.

Ejemplo: CHAIN "CPCBasic.BAS"

MODE 0
GRAPHICSPEN 15
MOVE 200,0
DRAW 200,400
MOVE 639,0
FILL 15
END

HEX\$

Función

Sintaxis: HEX\$(<valor>,<dígitos>)

Descripción: Devuelve una cadena que representa el <valor> dado en código hexadecimal, con tantos dígitos como se indique en <dígitos>. Si no se indica este valor se la cadena tendrá sólo los dígitos imprescindibles para representar el <valor>.

Ejemplo: CHAIN "CPCBasic.BAS"

```
PRINT HEX$(255,4)
END
```

INK

Orden

Sintaxis: INK <tinta>,<color1>[,<color2>]

Descripción: Asigna a la <tinta> indicada los colores <color1> y <color2>, parpadenando entre ellos. Si se omite <color2> se considera éste el mismo que <color1>.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM p#,i#,t
MODE 1:PAPER 0:PEN 1
FOR p#=0 TO 1
FOR i#=0 TO 26
INK p#,i#
LOCATE 16,12:PRINT "ink "+DEC$(p#,"0")+",""+DEC$(i#,"_0")
FOR t=1 TO 800:NEXT
NEXT
NEXT
INK 0,1:INK 1,24:CLS
END
```

INKEY\$

Función

Sintaxis: INKEY\$()

Descripción: Devuelve una cadena formada por el carácter que se haya pulsado. Si no se ha pulsado ninguno devuelve una cadena vacía.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM a$
CLS
PRINT "Elija Si o No (S/N):":PRINT
inicio
a$=INKEY$()
IF a$="" THEN GOTO inicio
IF a$="s" OR a$="S" THEN GOTO respuesta_si
IF a$="n" OR a$="N" THEN GOTO respuesta_no
GOTO inicio
respuesta_si:PRINT "Ha respondido SI":END
respuesta_no:PRINT "Ha respondido NO":END
```

INKEY%

Función

Sintaxis: INKEY%(<código>)

Descripción: Devuelve el estado de la tecla <código> según esté pulsada o no. Si no está pulsada devuelve -1, si lo está devuelve 0, si lo está conjuntamente con mayúsculas devuelve 32, si lo está conjuntamente con control devuelve 128 y si lo está conjuntamente con mayúsculas y control devuelve 160.

Ejemplo: CHAIN "CPCBasic.BAS"

```
bucle:IF INKEY%(55)<>32 THEN GOTO bucle
PRINT "Ha pulsado [SHIFT] y V"
CLEARINPUT
END
```

INP%

Función

Sintaxis: INP%(<dirección>)

Descripción: Da el valor actual de la puerta de entrada-salida <dirección>.

Ejemplo: CHAIN "CPCBasic.BAS"

```
PRINT DEC$(INP%(&FF77), "__0")
END
```

INPUT\$

Función

Sintaxis: INPUT\$([<cauce>])

Descripción: Devuelve una cadena de texto leída por teclado en la posición del cursor del <cauce> dado. Si <cauce> es 9 la cadena se lee del fichero de entrada abierto actualmente. Si no se especifica <cauce> se interpreta que es 0.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM a,b
MODE 1
inicio
PRINT "Escriba el primer factor: ":a=VAL(INPUT$()):PRINT
PRINT "Escriba el segundo factor: ":b=VAL(INPUT$()):PRINT
PRINT DEC$(a,"____0.00")+ " por "+DEC$(b,"____0.00")+ " es
"+DEC$(a*b,"____0.00"):PRINT
GOTO inicio
```

INSTR%

Función

Sintaxis: INSTR%(<cadena>,<subcadena>,[<inicio>])

Descripción: Devuelve la posición de <subcadena> dentro de <cadena>, comenzando a buscar desde la posición <inicio> de <cadena>. Si no se especifica <inicio> la búsqueda se hace desde el principio.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM n#,alfabeto$,a$,b$
CLS:FOR n#=1 TO 26
alfabeto$=alfabeto$+CHR$(n#+64)
NEXT
inicio
PRINT "Escriba una letra: ":a$=INPUT$():PRINT
b$=UPPER$(a$)
PRINT b$+" es la letra numero "
PRINT DEC$(INSTR%(alfabeto$,b$),"_0")
PRINT " del alfabeto":PRINT:PRINT
GOTO inicio
```

INT&

Función

Sintaxis: INT&(<valor>)

Descripción: Devuelve <valor> redondeado al número entero inferior igual o más próximo.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(INT&(-1.995), "-0")
END
```

JOY#

Función

Sintaxis: JOY#(<código>)

Descripción: Devuelve el estado del joystick correspondiente a <código> (0 para el primer joystick o 1 para el segundo). El valor devuelto es 1 si se está marcando hacia arriba, 2 hacia abajo, 4 hacia la izquierda, 8 hacia la derecha, 16 si se está pulsando el disparo 2 y 32 si está pulsando el disparo 1. Los valores pueden estar sumados si se realiza más de una acción simultáneamente.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

inicio
PRINT "Para detener el programa "
PRINT "accione el joystick":PRINT
IF JOY#(0)<>0 THEN END
GOTO inicio
```

KEY

Orden

Sintaxis: KEY <código>,<texto>

Descripción: Asigna a la tecla de <código> especificado (de 0 a 31 y de 128 a 159) el <texto> indicado.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

KEY 11,"border 13:paper 0:pen 1:ink 0,13:ink 1,0:mode
2:list"+CHR$(13)
END
```

KEYDEF

Orden

Sintaxis: KEYDEF <tecla>,<repetir>[,<normal>[,<mayúsculas>[,<control>]]]

Descripción: Define los códigos generados por la pulsación de una tecla. El valor <repetir> indica si la tecla se debe autorrepetir si se mantiene pulsada, y los valores <normal>, <mayúscula> y <control> indican el código generado por la tecla si se pulsa individualmente, junto a mayúsculas y junto a control respectivamente.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

KEY 159,"Esta es la tecla TAB"
KEYDEF 68,1,159
END
```

LEFT\$

Función

Sintaxis: LEFT\$(<cadena>,<longitud>)

Descripción: Devuelve una subcadena de <cadena> formada por los primeros <longitud> caracteres.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM a$,n#
CLS
a$="AMSTRAD"
FOR n#=1 TO 7
  PRINT LEFT$(a$,n#):PRINT
NEXT
END
```

LEN%

Función

Sintaxis: LEN%(<texto>)

Descripción: Devuelve la longitud en caracteres de <texto>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM a$
PRINT "Escriba una frase: ":a$=INPUT$:PRINT
PRINT "La frase tiene "
PRINT DEC$(LEN%(a$),"__0")+ " caracteres."
END
```

LOAD

Orden

Sintaxis: LOAD <fichero>,<dirección>

Descripción: Carga el contenido deL <fichero> en memoria a partir de la <dirección> indicada.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

MODE 0
LOAD "PANTALLA.BIN",&C000:REM Recuperar pantalla
bucle:GOTO bucle
```

LOCATE

Orden

Sintaxis: LOCATE <columna>,<fila>[,<cauce>]

Descripción: Situa el cursor de texto de la ventana especificada por <cauce> en la <columna> y <fila> indicadas. Si no se da <cauce> se realiza sobre la ventana de cauce 0.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM n#
MODE 1
FOR n#=1 TO 20
  LOCATE n#,n#
  PRINT CHR$(143)+"posicion "
  PRINT DEC$(n#,"_0")+","+"DEC$(n#,"_0")
NEXT
END
```

LOG!

Función

Sintaxis: LOG!(<valor>)

Descripción: Devuelve el logaritmo neperiano de <valor>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(LOG!(9999),"0.00000000")
END
```

LOG10!

Función

Sintaxis: LOG10!(<valor>)

Descripción: Devuelve el logaritmo decimal de <valor>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(LOG10!(9999),"0.00000000")
END
```

LOWER\$

Función

Sintaxis: LOWER\$(<texto>)

Descripción: Devuelve <texto> convertido a minúsculas.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM a$
a$="OBSERVE COMO SE CONVIERTEN LAS LETRAS A "
PRINT LOWER$(a$+"MINUSCULAS")
END
```

MASK

Orden

Sintaxis: MASK [<máscara>][,<primer punto>]

Descripción: Establece para el dibujo de líneas la <máscara> indicada, indicando si <primer punto> es 1 ó 0 si el primer punto de las líneas se pintará o no respectivamente.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM x#
MODE 0:INK 5,21:INK 8,16
inicio
MOVE -100*RND!(),400*RND!()
WHILE XPOS%(<)<640
  FOR x#=1 TO 8
    MASK 2^(8-x#)
    DRAWR 32,0,x#,1:MOVER -32,0
  NEXT
  MOVER 34,0
WEND:GOTO inicio
```

MAX!

Función

Sintaxis: MAX!(<valor1>,<valor2>)

Descripción: Devuelve el mayor valor entre <valor1> y <valor2>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM n
n=66
PRINT DEC$(MAX!(1,n), "_0")
END
```

MID\$

Función

Sintaxis: MID\$(<cadena>,<inicio>[,<longitud>])

Descripción: Devuelve una subcadena de <cadena> desde el carácter <inicio> de <longitud> caracteres. Si no se especifica <longitud> la subcadena será hasta el final de <cadena>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM a$,n#,T
a$="MASSACHUSSETTS"
PRINT "Deletreame "+a$:PRINT
PRINT "Vale ...":PRINT:PRINT
inicio
FOR n#=1 TO LEN%(a$)
  PRINT MID$(a$,n#,1)+CHR$(9)
  FOR t=1 TO 1400:NEXT:NEXT
PRINT:PRINT
PRINT "Ahora escriba usted otra palabra: ":a$=INPUT$:PRINT
GOTO inicio
```

MIN!

Función

Sintaxis: MIN!(<valor1>,<valor2>)

Descripción: Devuelve el menor valor entre <valor1> y <valor2>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(MIN!(3,2.999), "0.000")
END
```

MODE

Orden

Sintaxis: **MODE** <modo>

Descripción: Establece la pantalla al <modo> dado (0, 1 ó 2).

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM m#
inicio
m#=m#+1:IF m#>2 THEN m#=0
MODE m#
PRINT "Este es el modo "+DEC$(m#,"0"):PRINT
PRINT "Pulse una tecla."
bucle
IF INKEY$="" THEN GOTO bucle ELSE GOTO inicio
```

MOVE

Orden

Sintaxis: **MOVE** <x>,<y>[,<tinta>][,<modo>]]

Descripción: Sitúa el cursor gráfico en las coordenadas <x>,<y> con el color de tinta gráfica actual. Si se especifica <tinta> se establece previamente la tinta gráfica a dicho color. Si se especifica <modo> se establece previamente el modo de dibujo al especificado (0 para normal, 1 para XOR, 2 para AND y 3 para OR).

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFINT A-Z
DIM x,y
MODE 1:TAG
inicio
x=RND!()*800-100:y=RND!()*430
MOVE x,y
PRINT "Estoy aqui"
GOTO INICIO
```

MOVER

Orden

Sintaxis: **MOVER <x>,<y>[,<tinta>][,<modo>]**

Descripción: Desplaza el cursor gráfico <x> e <y> unidades desde la posición actual. Si se especifica <tinta> se establece previamente la tinta gráfica a dicho color. Si se especifica <modo> se establece previamente el modo de dibujo al especificado (0 para normal, 1 para XOR, 2 para AND y 3 para OR).

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM n#
MODE 1:TAG:MOVE 0,16
PRINT "primero se sube"
FOR n#=1 to 8
  MOVER -56,16
  PRINT "sube":NEXT:PRINT " y luego se baja"
FOR n#=1 TO 8
  MOVER -56,-16
  PRINT "baja":NEXT
END
```

ONSQ

Orden

Sintaxis: **ONSQ <canal>,<etiqueta>**

Descripción: Ejecuta la rutina identificada por <etiqueta> cuando la cola del <canal> especificado queda vacío.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM x#,s%
ENV 1,15,-1,1
ONSQ 1,rutina
MODE 0:ORIGIN 0,0,200,440,100,300
RESTORE datos
inicio
FOR x#=1 TO 13:FRAME:MOVE 330,200,x#
FILL x#:NEXT:GOTO inicio
rutina
READ s%:IF s%=0 THEN RESTORE datos:GOTO rutina
SOUND 1,s%,25,15,1
ONSQ 1,rutina:RETURN
datos:DATA 50%,60%,90%,100%,35%,200%,24%,500%,0%
```

OPENIN

Orden

Sintaxis: OPENIN <nombre>

Descripción: Abre en modo lectura el fichero <nombre>.

Ejemplo: CHAIN "CPCBasic.BAS"

```
REM abrir un fichero del disco para entrada
DIM a,a$
OPENIN "datos":a=VAL!(INPUT$(9)):a$=INPUT$(9)
CLOSEIN:PRINT "Los dos valores son:":PRINT
PRINT:PRINT DEC$(a,"_____0.00000")+CHR$(9)+a$
END
```

OPENOUT

Orden

Sintaxis: OPENOUT <nombre>

Descripción: Abre en modo escritura el fichero <nombre>.

Ejemplo: CHAIN "CPCBasic.BAS"

```
REM abrir un fichero del disco para salida
DIM a,a$
PRINT "Escriba un numero: ":a=VAL!(INPUT$()):PRINT
PRINT "Escriba una palabra: ":a$=INPUT$():PRINT
OPENOUT "datos":PRINT DEC$(a,"_____0.00000"),9:PRINT
,9:PRINT a$,9
CLOSEOUT:PRINT "Datos grabados en disco"
END
```

ORIGIN

Orden

Sintaxis: ORIGIN <X>,<y>[,<x1>[,<x2>[,<y1>[,<y2>]]]]

Descripción: Establece el origen de coordenadas gráficas en la posición absoluta <x>, <y>. Si se especifican los parámetros <x1>, <x2>, <y1>, <y2> se establecen con ellos los límites izquierdo, derecho, superior e inferior respectivamente de la ventana gráfica.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM x%
MODE 1:BORDER 13:TAG
ORIGIN 0,0,100,540,300,100
GRAPHICSPAPER 3:CLG
repetir
FOR x%=550 TO -340 STEP -10
MOVE x%,206
PRINT "Esta es una ventana grafica "
FRAME:NEXT:GOTO repetir
```

OUT

Orden

Sintaxis: OUT <puerto>,<valor>

Descripción: Envía el <valor> indicado al <puerto> de entrada-salida especificado.

Ejemplo: (Sin ejemplo)

PAPER

Orden

Sintaxis: PAPER <tinta>[,<cauce>]

Descripción: Estable la <tinta> especificada como color de papel para el <cauce> indicado. Si no se especifica <cauce> se supone el cauce 0.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM p#,t
MODE 0:PEN 0:INK 0,13
FOR p#=1 TO 15
  PAPER p#:CLS
  LOCATE 7,12:PRINT "paper "+DEC$(p#,"_0")
  FOR t=1 TO 1000:NEXT
NEXT
END
```

PEEK%

Función

Sintaxis: PEEK%(<dirección>)

Descripción: Devuelve el valor de la posición de memoria <dirección>.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM n&,p%
MODE 1
WINDOW 1,40,1,3:WINDOW 1,40,4,25,1
LOCATE 13,1:PRINT "MEMORIA":PRINT
PRINT:PRINT "Direccion"
LOCATE 20,3:PRINT "Contenido"
FOR n&=0 TO 65535
  p%=PEEK%(n&)
  PRINT DEC$(n&,"###0")+ " (&"+HEX$(n&,4)+") ",1
  PRINT SPACE$(6)+DEC$(p%,"#0")+ " (&"+HEX$(p%,2)+") ",1:PRINT
  "",1
NEXT
END
```

PEN

Orden

Sintaxis: PEN <tinta>[,<modo>][,<cauce>]

Descripción: Estable la <tinta> especificada como color de texto para el <cauce> indicado. Si no se especifica <cauce> se supone el cauce 0. Si se indica <modo> se establece como transparente u opaco según sea 1 ó 0 respectivamente.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM p#,t
MODE 0:PEN 0:INK 0,13
inicio
FOR p#=1 TO 15
  PEN p#:PRINT SPACE$(47)+"pen "+DEC$(p#,"_0"):PRINT
  FOR t=1 TO 1000:NEXT
NEXT
GOTO inicio
```

PI!

Función

Sintaxis: PI!()

Descripción: Devuelve el valor de la constante PI.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(PI!()), "0.00000000"
END
```

PLOT

Orden

Sintaxis: PLOT <x>,<y>[,<tinta>][,<modo>]

Descripción: Dibuja un punto en la posición <x>,<y> con el color de tinta gráfica actual. Si se especifica <tinta> se establece previamente la tinta gráfica a dicho color. Si se especifica <modo> se establece previamente el modo de dibujo al especificado (0 para normal, 1 para XOR, 2 para AND y 3 para OR).

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM X%,t&
MODE 1:BORDER 0:PAPER 0:PEN 1
INK 0,0:INK 1,26:INK 2,13,26:DEG
FOR x%=1 TO 360:ORIGIN 320,200
    DRAW 50*COS!(x%),50*SIN!(x%),1
    PLOT 100*COS!(x%),25*SIN!(x%):NEXT
ORIGIN 0,0:t&=TIME&()+700:WHILE TIME&()<t&
PLOT RND!()*640,RND!()*400:WEND
PLOT RND!()*640,RND!()*400,2
bucle:GOTO bucle
```

PLOTR

Orden

Sintaxis: PLOTR <x>,<y>[,<tinta>][,<modo>]

Descripción: Dibuja un punto en la posición actual del cursor gráfico incrementada en <x> e <y> unidades, con el color de tinta gráfica actual. Si se especifica <tinta> se establece previamente la tinta gráfica a dicho color. Si se especifica <modo> se establece previamente el modo de dibujo al especificado (0 para normal, 1 para XOR, 2 para AND y 3 para OR).

Ejemplo: CHAIN "CPCBasic.BAS"

```
REM dibujo de rectas con las teclas del cursor
BORDER 0:GRAPHICSPEN 1
inicio
MODE 1:PLOT 320,200
bucle
IF INKEY%(0)=0 THEN PLOTR 0,1
IF INKEY%(1)=0 THEN PLOTR 1,0
IF INKEY%(2)=0 THEN PLOTR 0,-1
IF INKEY%(8)=0 THEN PLOTR -1,0
IF INKEY%(9)=0 THEN GOTO inicio:REM copy = borrar
GOTO bucle
```

POKE

Orden

Sintaxis: POKE <dirección>,<valor>

Descripción: Introduce en la <dirección> de memoria dada el <valor> indicado.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM m&
FOR m&=49152 to 65535
  POKE m&,100
NEXT
END
```

POS#

Función

Sintaxis: POS#(<cauce>)

Descripción: Devuelve la posición horizontal del cursor de texto en el <cauce> especificado.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM n#
MODE 1:BORDER 0:LOCATE 8,2
PRINT "Utilice las teclas de cursor a la izquierda y cursor a
la derecha"
WINDOW 1,40,12,12:CURSOR 1,1
FOR n#=1 TO 19:PRINT CHR$(9):NEXT
inicio
IF INKEY%(1)<>-1 THEN PRINT CHR$(9)
IF INKEY%(8)<>-1 THEN PRINT CHR$(8)
LOCATE 1,24,1
PRINT "Posicion horizontal del ",1
PRINT "cursor de texto = ",1
PRINT DEC$(POS#(0),"#0"),1:GOTO inicio
```

PRINT

Orden

Sintaxis: PRINT <cadena>[,<cauce>]

Descripción: Escribe la <cadena> dada en la posición actual del cursor de texto o cursor gráfico, dependiendo de si se ha activado el modo escritura gráfica. Si se especifica <cauce> entre 0 y 7 se escribe sobre la ventana correspondiente a dicho cauce. Si se especifica <cauce> 8 la escritura será en la impresora y si para <cauce> se da el valor 9 la escritura será en fichero. Si no se da valor para <cauce> se tomará valor 0. Si no se da valor para <cadena> se realizará un retorno de carro y avance de página en pantalla, impresora o fichero según el <cauce> indicado.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM a$,b$
a$="corta"
b$="esta cadena es mas larga"
PRINT a$+b$:PRINT
PRINT a$+CHR$(9)+b$:PRINT
PRINT
PRINT b$+b$:PRINT
PRINT b$+CHR$(9)+b$:PRINT
END
```

RAD

Orden

Sintaxis: RAD

Descripción: Establece el cálculo de las funciones trigonométricas a radianes.

Ejemplo: (Sin ejemplo)

RANDOMIZE

Orden

Sintaxis: RANDOMIZE <valor>

Descripción: Inicializa la secuencia de números aleatorios con la semilla dada por el <valor> indicado.

Ejemplo: CHAIN "CPCBasic.BAS"

RANDOMIZE 123.456
PRINT DEC\$(RND!(), "0.00000")
END

RELEASE

Orden

Sintaxis: RELEASE <canal>

Descripción: Libera el canal de sonido especificados por <canal> según los valores 1, 2 ó 4. Se puede especificar varios canales sumando sus valores.

Ejemplo: CHAIN "CPCBasic.BAS"

SOUND 65,1000,100
PRINT "Pulse R para liberar el sonido"
bucle:IF INKEY%(50)=-1 THEN GOTO bucle
RELEASE 1
END

REMAIN%

Función

Sintaxis: **REMAIN%(<buffer>)**

Descripción: Devuelve el número de cincuentavos de segundo que restan para realizar la interrupción identificada por <buffer>. Esta interrupción queda desactivada.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM buffer_after1#(12),buffer_after2#(12)
AFTER 500,rutina1,@buffer_after1#(0)
AFTER 100,rutina2,@buffer_after2#(0)
bucle:PRINT "Programa funcionando":PRINT:GOTO bucle
rutina1
REM esta subrutina no sera invocada porque lo impide la ultima
linea
RETURN
rutina2
PRINT:PRINT "El temporizador 1 va a ser "
PRINT "inhibido por REMAIN.":PRINT
PRINT "Unidades de tiempo que quedaban: "
PRINT DEC$(REMAIN%(@buffer_after1#(0)),"____0")
END
```

RIGHT\$

Función

Sintaxis: **RIGHT\$(<cadena>,<longitud>)**

Descripción: Devuelve una subcadena de <cadena> formada por los últimos <longitud> caracteres.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM a$,n#
MODE 0:a$="Ordenador CPC6128"
FOR n#=1 TO 17:LOCATE 21-n#,n#
  PRINT RIGHT$(a$,n#):PRINT
NEXT
END
```

RND!

Función

Sintaxis: RND!(<código>)

Descripción: Devuelve un número aleatorio entre 0 y menor que 1. Si <código> es 0 devuelve el último número generado y si no se especifica <código> o es distinto de cero devuelve un nuevo número aleatorio.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFBYTE A-Z
DIM x,n
RANDOMIZE TIME&()
FOR x=1 TO -1 STEP -1
  PRINT "parametro de rnd = "+DEC$(x,"-0"):PRINT
  FOR n=1 TO 6
    PRINT DEC$(RND!(x),"0.000000000"):PRINT
  NEXT
NEXT
END
```

ROUND!

Función

Sintaxis: ROUND!(<valor>,<decimales>)

Descripción: Devuelve <valor> redondeado al número de <decimales> indicado. Si no se indica <decimales> se interpreta como 0.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM n#
FOR n#=4 TO -4 STEP -1
  PRINT DEC$(ROUND(1234.5678,n#),"###0.0000")+SPACE$(3)
  PRINT "con <decimales> = "+DEC$(n#,"-0"):PRINT
NEXT
END
```

SAVE

Orden

Sintaxis: SAVE <fichero>,<inicio>,<longitud>[,<ejecución>]

Descripción: Graba en el <fichero> el contenido de la memoria desde <inicio>, tantos bytes como indique <longitud> y opcionalmente si el contenido es un programa ejecutable se puede indicar con <ejecución> la dirección de arranque.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

MODE 0
LOCATE 7,12:PRINT "PANTALLA"
SAVE "PANTALLA.BIN",&C000,&4000:REM Guardar pantalla
END
```

SGN#

Función

Sintaxis: SGN#(<valor>)

Descripción: Devuelve 1 ó -1 dependiendo si el signo de <valor> es positivo o negativo. Devuelve 0 si <valor> es también 0.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DEFINT A-Z
DIM n
FOR n=200 TO -200 STEP -20
  PRINT "SGN da "
  PRINT DEC$(SGN#(n),"-0")+ " cuando el numero es "+DEC$(n,"-
__0"):PRINT
NEXT
END
```

SIN!

Función

Sintaxis: SIN!(<valor>)

Descripción: Devuelve el valor del seno correspondiente a <valor>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM n,y
CLS:DEG:ORIGIN 0,200
FOR n=0 TO 720
  y=SIN!(n)
  PLOT n*640/720,198*y:NEXT
bucle:GOTO bucle
```

SOUND

Orden

Sintaxis: **SOUND <canal>,<frecuencia>[,<duración>],[,<volumen>],[,<envolvente volumen>],[,<envolvente tono>],[,<ruido>]]]]]**

Descripción: Sitúa en la cola del <canal> el sonido dado por la <frecuencia>, <duración> (20 si no se especifica) y <volumen> (12 si no se especifica), utilizando la <envolvente volumen> y <envolvente tono> indicados (ninguna si no se especifica) y con un nivel de ruido dado por el valor de <ruido> (ninguno si no se especifica).

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM z%
FOR z%=0 TO 4095
    SOUND 1,z%,1,12
NEXT
END
```

SPACE\$

Función

Sintaxis: **SPACE\$(<longitud>)**

Descripción: Devuelve una cadena formada por tantos espacios como se indique en <longitud>.

Ejemplo: CHAIN "CPCBasic.BAS"

```
MODE 1
PRINT "Pongamos 5 espacios entre uno"
PRINT SPACE$(5)
PRINT "y otro"
END
```

SPEEDINK

Orden

Sintaxis: **SPEEDINK <valor1>,<valor2>**

Descripción: Especifica con <valor1> y <valor2> el tiempo de permanencia de cada color asignado a las tintas o al borde de la pantalla.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM i#,t
BORDER 7,18
FOR i#=30 TO 1 STEP -1
    SPEEDINK i#,i#
    FOR t=1 TO 1400:NEXT
NEXT
END
```

SPEEDKEY

Orden

Sintaxis: SPEEDKEY <inicio>,<repetición>

Descripción: Establece con <inicio> los cincantavos de segundo que tardará una tecla en comenzar a repetirse y con <repetición> los cincantavos de segundo entre cada repetición.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM k#,a$
CLS:FOR k#=7 TO 1 STEP -2
PRINT "Escriba su nombre y pulse [RETURN]":PRINT
SPEEDKEY k#,k#
a$=INPUT$():PRINT:NEXT
PRINT "Es un nombre muy interesante"
END
```

SPEEDWRITE

Orden

Sintaxis: SPEEDWRITE <velocidad>

Descripción: Establece la velocidad de grabación de archivos en disco a normal o rápida dependiendo de que el valor de <velocidad> sea 0 ó 1 respectivamente.

Ejemplo: (Sin ejemplo)

SQ#

Función

Sintaxis: SQ#(<canal>)

Descripción: Devuelve el estado del <canal> de sonido especificado. El valor debe interpretarse en binario del siguiente modo: bits 2, 1 y 0 es el número de espacios libres en la cola, bits 5, 4 y 3 es el estado de sincronización de la primera nota en cola, bit 6 indica si la primera nota en cola está retenida, bit 7 indica si el canal está activo.

Ejemplo: CHAIN "CPCBasic.BAS"

```
SOUND 65,100,100
PRINT DEC$(SQ%(1),"__0")
END
```

SQR!

Función

Sintaxis: SQR!(<valor>)

Descripción: Devuelve el resultado de la raíz cuadrada correspondiente a <valor>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"  
  
PRINT DEC$(SQR!(9), "0")  
END
```

STRING\$

Función

Sintaxis: STRING\$(<longitud>,<texto>)

Descripción: Devuelve una cadena formada por <longitud> caracteres repetidos del primer carácter de <texto>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"  
  
PRINT STRING$(40, "*")  
END
```

SYMBOL

Orden

Sintaxis: **SYMBOL** <carácter>[,<fila1>[,<fila2>[,<fila3>[,<fila4>[,<fila5>[,<fila6>[,<fila7>[,<fila8>]]]]]]]]]

Descripción: Establece la forma del <carácter> especificado según los valores <fila1> a <fila8>. Si alguno de ellos no se especifica se considera un valor 0 para esa fila.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFINT A-Z
DIM
fila1,fila2,fila3,fila4,fila5,fila6,fila7,fila8,buffer_simbolos
#(1208)
MODE 1:SYMBOLAFTER 105,@buffer_simbolos#(0)
fila1=255:REM 11111111 en binario
fila2=129:REM 10000001 en binario
fila3=189:REM 10111101 en binario
fila4=153:REM 10011001 en binario
fila5=153:REM 10011001 en binario
fila6=189:REM 10111101 en binario
fila7=129:REM 10000001 en binario
fila8=255:REM 11111111 en binario
PRINT "La linea siguiente redefine la letra i (105). Teclee
varias veces esa letra y luego liste el programa."
SYMBOL 105,fila1,fila2,fila3,fila4,fila5,fila6,fila7,fila8
END
```

SYMBOLAFTER

Orden

Sintaxis: **SYMBOLAFTER** <carácter>,<buffer>

Descripción: Establece como primer carácter redefinible el indicado en <carácter>. Debe especificarse la dirección de un buffer de 8 x n° caracteres redefinibles.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM buffer_simbolos#(1128)
CLS
SYMBOLAFTER 115,@buffer_simbolos#(0)
PRINT "La linea siguiente redefine la letra s "
SYMBOL 115,0,56,64,64,48,8,8,112
PRINT "a s":PRINT
PRINT "Cancele esta definicion de s":PRINT
PRINT "tecleando SYMBOL AFTER 240"
END
```

TAG

Orden

Sintaxis: TAG [<cauce>]

Descripción: Activa la escritura gráfica en la ventana de texto indicada con <cauce>. Si no se especifica <cauce> se toma éste como 0.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFINT A-Z
DIM a$,x,y,f,b
PRINT "Escriba su nombre: ":a$=INPUT$:CLS
PRINT "Se mueve usted mucho, "+a$
TAG
inicio
x=LEN%(a$)*17:y=50+RND!()*300:MOVE -x,y
FOR f=-x TO 640 STEP RND!()*7+3
  MOVE f,y:PRINT " "+a$:FRAME:NEXT
FOR b=640 TO -x STEP -RND!()*7+3
  MOVE b,y:PRINT a$+" ":FRAME:NEXT
GOTO inicio
```

TAGOFF

Orden

Sintaxis: TAGOFF [<cauce>]

Descripción: Desactiva la escritura gráfica en la ventana de texto indicada con <cauce>. Si no se especifica <cauce> se toma éste como 0.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFINT A-Z
dim periodo,x
MODE 2:TAG:REM texto ligado al cursor grafico
periodo=1984:FOR x=1 TO 640 STEP 70
MOVE x,400:DRAWR 0,-350
periodo=periodo+1:PRINT DEC$(periodo,"0000"):NEXT
TAGOFF:REM texto desconectado de cursor grafico
LOCATE 34,25:PRINT "Datos anuales"
bucle:GOTO bucle
```

TAN!

Función

Sintaxis: TAN!(<valor>)

Descripción: Devuelve el valor de la tangente correspondiente a <valor>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

PRINT DEC$(TAN!(45),"0.00000000")
END
```

TEST#

Función

Sintaxis: TEST#(<x>,<y>)

Descripción: Devuelve el color de la tinta del pixel de coordenadas <x> e <y>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

CLS
PRINT "Estamos utilizando la pluma numero "
PRINT DEC$(TEST#(10,386),"_0"):PRINT
PRINT "Cambie la pluma y el modo "
PRINT "... y ejecute otra vez el programa."
END
```

TESTR#

Función

Sintaxis: TESTR#(<x>,<y>)

Descripción: Devuelve el color de la tinta del pixel de la posición actual del cursor gráfico incrementada en <x> e <y>.

Ejemplo:

```
CHAIN "CPCBasic.BAS"

DIM x#,n#
MODE 0:FOR x#=1 TO 15:LOCATE 1,x#
PEN x#:PRINT STRING$(10,CHR$(143)):NEXT
MOVE 200,400:PEN 1
FOR n#=1 TO 23:LOCATE 12,n#
PRINT "pen "+DEC$(TESTR#(0,-16),"_0"):PRINT:NEXT
END
```

TIME&

Función

Sintaxis: TIME&

Descripción: Devuelve el número de trescientavos de segundos transcurridos desde que se reinició el sistema por última vez.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFBYTE A-Z
DIM hora,minuto,segundo,tiempo,referencia&
CLS:REM reloj
PRINT "Hora: ":hora=VAL!(INPUT$()):PRINT
PRINT "Minuto: ":minuto=VAL!(INPUT$()):PRINT
PRINT "Segundo: ":segundo=VAL!(INPUT$()):PRINT
iniciol
CLS:referencia&=INT&(TIME&()/300)
inicio2
WHILE hora<13
  WHILE minuto<60
    WHILE tiempo<60
      tiempo=(INT&(TIME&()/300)-referencia&)+segundo
      LOCATE 1,1
      PRINT DEC$(hora,"00 ")+DEC$(minuto,"00
")+DEC$(tiempo,"00"):PRINT
    WEND
    tiempo=0:segundo=0:minuto=minuto+1
    GOTO iniciol
  WEND
  minuto=0:hora=hora+1
WEND
hora=1
GOTO inicio2
```

UNT%

Función

Sintaxis: UNT%(<valor>)

Descripción: Devuelve el <valor> entero sin signo de 16 bits convertido a entero con signo.

Ejemplo: CHAIN "CPCBasic.BAS"

```
PRINT DEC$(UNT%(&FF66),"-____0")
END
```

UPPER\$

Función

Sintaxis: UPPER\$(<texto>)

Descripción: Devuelve <texto> convertido a mayúsculas.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM a$
CLS:a$="mire como crecen!"
PRINT UPPER$(a$)
END
```

VAL!

Función

Sintaxis: VAL!(<texto>)

Descripción: Devuelve el valor que representa <texto>, que puede especificarse en formato decimal, binario (prefijo &X) o hexadecimal (prefijo & ó &H).

Ejemplo: CHAIN "CPCBasic.BAS"

```
DEFBYTE A-Z
DIM a$,n,x
CLS:PRINT "Ya he aprendido la tabla de multiplicar":PRINT
inicio
PRINT:PRINT "Pulse una tecla del 1 al 9:":PRINT
bucle:a$=INKEY$:IF a$="" THEN GOTO bucle
n=VAL!(a$):IF n<1 OR n>9 THEN GOTO bucle
FOR x=1 TO 10
  PRINT DEC$(n,"0")+x+DEC$(x,"_0")+ "="+DEC$(n*x,"_0"):PRINT
NEXT:GOTO inicio
```

VPOS#

Función

Sintaxis: VPOS#(<cauce>)

Descripción: Devuelve la posición vertical del cursor de texto en el <cauce> especificado.

Ejemplo: CHAIN "CPCBasic.BAS"

```
MODE 1:BORDER 0
PRINT "Utilice las teclas subir/bajar cursor"
WINDOW 39,39,1,25:CURSOR 1,1
LOCATE 1,13
inicio
IF INKEY%(0)<>-1 THEN PRINT CHR$(11)
IF INKEY%(2)<>-1 THEN PRINT CHR$(10)
LOCATE 1,23,1
PRINT "Posicion vertical del",1:PRINT "",1
PRINT "cursor de texto = ",1
PRINT DEC$(VPOS#(0),"_0 "),1:GOTO inicio
```

WAIT

Orden

Sintaxis: WAIT <puerto>,<máscara>[,<inversion>]

Descripción: Espera hasta que por la dirección de entrada-salida <puerto> especificada se reciba un valor N tal que N XOR <inversión> AND <máscara> sea distinto de cero. Si no se especifica <inversión> no se realiza la operación XOR.

Ejemplo: CHAIN "CPCBasic.BAS"

```
WAIT &FF34,20,25
END
```

WINDOW

Orden

Sintaxis: WINDOW <x1>,<x2>,<y1>,<y2>[,<cauce>]

Descripción: Establece la ventana de <cauce> indicado desde las columnas de pantalla <x1> a <x2> y desde las filas <y1> a <y2>. Si no se especifica <cauce> se interpreta como que éste es el 0.

Ejemplo: CHAIN "CPCBasic.BAS"

```
MODE 0:BORDER 0:REM carta de ajuste
INK 0,0:INK 1,25:INK 2,23:INK 3,21
INK 4,17:INK 5,6:INK 6,2:INK 7,26
PAPER 0:CLS
PAPER 1:WINDOW 2,4,1,18:CLS
PAPER 2:WINDOW 5,7,1,18:CLS
PAPER 3:WINDOW 8,10,1,18:CLS
PAPER 4:WINDOW 11,13,1,18:CLS
PAPER 5:WINDOW 14,16,1,18:CLS
PAPER 6:WINDOW 17,19,1,18:CLS
PAPER 7:WINDOW 2,19,19,25:CLS
bucle:GOTO bucle
```

WINDOWSWAP

Orden

Sintaxis: WINDOWSWAP <cauce1>,<cauce2>

Descripción: Intercambia las características de las ventanas identificadas por <cauce1> y <cauce2>.

Ejemplo: CHAIN "CPCBasic.BAS"

```
DIM t
MODE 1:INK 1,24:INK 2,9:INK 3,6
WINDOW 21,40,13,25:PAPER 3
WINDOW 1,20,1,12,1:PAPER 2,1
CLS:PRINT " Ventana numero 0"
CLS 1:PRINT " Ventana numero 1",1
inicio
LOCATE 1,6
PRINT " Ventana roja (0)"+SPACE$(2)
LOCATE 1,6,1
PRINT " Ventana verde (1)",1
FOR t=1 TO 2000:NEXT
WINDOWSWAP 0,1:GOTO inicio
```

XPOS%

Función

Sintaxis: XPOS%()

Descripción: Devuelve la coordenada X del cursor gráfico.

Ejemplo: CHAIN "CPCBasic.BAS"

```
MODE 1:DRAW 320,200
PRINT "Posicion horizontal del cursor grafico = "
PRINT DEC$(XPOS%(),"__0")
END
```

YPOS%

Función

Sintaxis: YPOS%()

Descripción: Devuelve la coordenada Y del cursor gráfico.

Ejemplo: CHAIN "CPCBasic.BAS"

```
MODE 1:DRAW 320,200
PRINT "Posicion horizontal del cursor grafico = "
PRINT DEC$(YPOS%(),"__0")
END
```