SPECTRUM SOFTWARE

Todos los programas existentes para los ordenadores Spectrum (incluyendo todos los títulos) son completamente compatibles con su nuevo ordenador ZX Spectrum +.

DORLING KINDERSLEY LTD.
en asociación con
SINCLAIR RESEARCH LTD.

ZX Spectrum+ Guía para el empleo

sinclair

ZX Spectrum+

Guía para el empleo



SPECTRUM SOFTWARE
Todos los programas existentes para los ordenadores Spectrum (incluyendo todos los títulos) son completamente compatibles con su nuevo ordenador ZX Spectrum +.

PRESENTANDO EL ZX SPECTRUM +

Sinclair Research encabeza los avances en la tecnología del microchip que ha puesto la computación al alcance de todos. Después del desarrollo del primer ordenador de escritorio económico del mundo, el ZX80, hemos combinado aún más la capacidad de computación con un mayor valor en sus sucesores – los ordenadores ZX81, ZX Spectrum y QL. Nuestra lema es el uso fácil tanto en el deseño del ordenador como en la manera de funcionar. Con el ordenador ZX Spectrum+, Sinclair Research ha avanzado un paso más. En él encontrará usted una máquina con todas las mejores características del Spectrum en una versión más avanzada que identifica este ordenador como el más poderoso y popular de todos y aún más fácil de usar. Esperamos que usted aprovechará todas la ventajas que ofrece su nuevo ordenador.

Clive Sinclair

INDICE

PRIMEROS PASOS 3

PROGRAMANDO SU ORDENADOR ZX SPECTRUM + 17

APRENDIENDO ACERCA DE SU ZX SPECTRUM + 41

NOCIONES BÁSICAS ACERCA DEL LENGUAJE BASIC DE SINCLAIR 49

por Neil Ardley
publicado por Dorling Kindersley Ltd
en asociación con
Sinclair Research Ltd.

COMO UTILIZAR ESTE LIBRO

Esta guía para su ZX Spectrum + contiene cuatro capítulos, cada uno de ellos identificado por un color. Para remitirse a uno de los capítulos bastará con dirigirse a la seccion del color correspondiente.

1 PRIMEROS PASOS

■ Como organizar su sistema ZX Spectrum + ■ Sintonizando la televisión ■ Como solucionar problemas ■ Cuáles son las capacidades de su ZX Spectrum +? ■ Como utilizar programas de ordenador comerciales ■ El procedimiento para cargar un programa ■ Preguntas y respuestas acerca de los problemas que puedan producirse al cargar un programa

2 PROGRAMANDO SU ORDENADOR ZX SPECTRUM +

■ El teclado – el panel de controles de su ordenador ■ Como operar las teclas ■ Una calculadora en su televisor ■ Colores, como emplearlos ■ Gráficas sencillas ■ Dibujando con su ordenador ■ Como diseñar sus propios patrones y dibujos ■ El procedimiento para crear carácteres ■ Dibujos animados ■ Un instrumento musical y efectos sonoros ■ Como registrar sus programas ■ Soluciones para los problemas que puedan producirse al registrar programas

3 APRENDIENDO ACERCA DE SU ZX SPECTRUM +

■ Cuáles son sus componentes? ■ Cómo funciona? ■ El procedimiento para conectar equipos accesorios ■ Informacion técnica

4 NOCIONES ACERCA DEL LENGUAJE BASIC DE SINCLAIR

■ Guía de referencia para el programador utilizando las palabras clave del BASIC de Sinclair ■ Su Spectrum se comunica con usted mediante reportes en la pantalla del televisor ■ Más allá de los confines del BASIC ■ La terminología de los ordenadores: un breve diccionario

PRIMEROS PASOS

El propósito de este capítulo es explicarle como comenzar a explorar el potencial de su ordenador ZX Spectrum +. Le indicaremos como armar y organizar su sistema para que pueda entrar en acción cuando usted se lo indique. Una vez completada esta etapa usted podrá elegir entre dos caminos. La primera posibilidad es introducir algunos programas en su ordenador Spectrum, los cuales demostrarán su operación, incluyendo las capacidades del color y sonido, o, alternativamente, usted puede preferir aprender a utilizar programas de ordenador comerciales, prontos para ser procesados, tales como juegos de ordenador ("computer games"). Cualquiera sea su elección, usted tendrá la oportunidad de disfrutar plenamente de las posibilidades ofrecidas por su nuevo ordenador.

1

COMO ORGANIZAR SU SISTEMA ZX SPECTRUM +

El procedimiento para organizar su sistema Spectrum es el siguiente: en primer lugar verifique que tenga disponibles todos los elementos enumerados en la lista de control a continuación. En segundo lugar siga las instrucciones detalladas en la página opuesta.

Conecte todos los componentes seguramente. Recuerde que si durante la operación de su Spectrum usted desconecta la electricidad o desenchufa el ordenador, se perderán todos los programas, información o resultados almacenados en la memoria del ordenador.

Después de haber terminado de trabajar con su ordenador, apaguelo y desenchufelo.

Lista de control: Tiene todos los elementos necesarios?

Al desembalar su ordenador usted encontrará:

1 su ordenador ZX Spectrum +

2 su fuente de energia eléctrica. Esta produce el suministro de 9 volt CD requeridos por el ordenador Spectrum.

3 cable de conexión a la antena de su televisor. Conecta su Spectrum con la televisión.

4 cable de la grabadora de cassette. Conecta su Spectrum con la grabadora de cassette.

5 la Targeta de Garantia. Complete la targeta y remitala como se indica en ella.

6 la Guia para el Empleo de su ordenador en cassette.

7 este Manual

Usted debera suministrar los elementos siguientes:

1 una televisión.

2 una grabadora de cassette.

3 un enchufe eléctrico para conectar el sistema a la red de electricidad.



Organizando su equipo: preguntas y respuestas

Es necesario que la televisión sea de colores?
No, no es necesario, aunque si su televisión es blanco y negro entonces usted no podra apreciar los colores producidos por su Spectrum.

Sirve caulquier tipo de televisión?
En principio su Spectrum le ofrecerá una imagen en cualquier televisión. Si este no es el caso, la causa del problema puede ser que el ordenador y la televisión tengan diferentes sistemas para generar las imagenes. Esto puede suceder si su televisión es muy antigua, o si la televisión y su Spectrum han sido adquiridas en paises diferentes. En caso de duda le aconsejamos que consulte a su vendedor de televisiones.

Es posible emplear un monitor en lugar de la televisión?
Si. es posible que el vendedor del ordenador también puede sumninistrarle un monitor, el cual le ofrecerá una mejor imagen para su Spectrum.

Puedo utilizar el accesorio ZX 16 RAM?
No. Este accesorio RAM (o "RAM pack") solamente puede ser utilizado con el ordenador Sinclair ZX81.

Conectando su ZX Spectrum + a la red eléctrica

Primero instale un enchufe eléctrico en el cable del suministro de electricidad de su ordenador. En algunos paises será necesario instalar un fusible de 3A en el enchufe eléctrico. Observe que su Spectrum no requiere una conexión a tierra, aun si el enchufe que usted instale sea de tres espigas. A continuación, complete las siguientes etapas, siguiendo la secuencia de las ilustraciones, para conectar su Spectrum a la red eléctrica y su televisión. Después de haber conectado debidamente el sistema, diríjase a la página siguiente, donde encontrará las instrucciones para sintonizar.



Inserte el pequeño enchufe del cable del suministro de electricidad en el enchufe marcado 9VDC de su Spectrum.



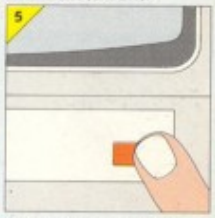
Enchufar el enchufe de la electricidad principal Recuerde que su ordenador Spectrum no tiene un interruptor de conectado/ desconectado (ON/OFF).



Insertar el cable de la televisión en el enchufe marcado TV de su Spectrum. Solamente uno de los enchufes en el cable de la televisión se ajustara a este punto de conexion.



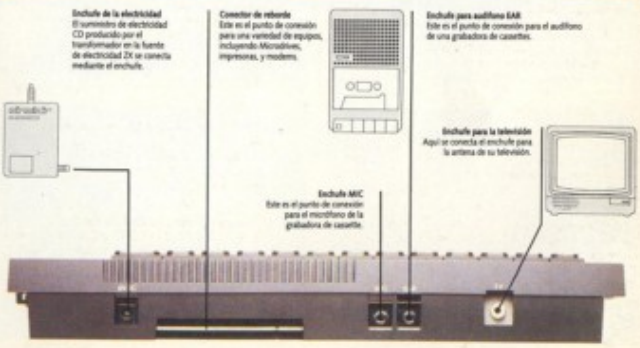
Retire el cable que conecta la antena de su televisión. Ahora conecte el cable de la televisión de su Spectrum, en el conector de la antena de su televisión.



Conecte su televisión y baje su volumen al mínimo.

Ahora todo esta pronto para proceder a la siguiente etapa: sintonizar la televisión para que reciba las señales de su Spectrum.

Los enchufes y conectores del Spectrum



SINTONIZANDO SU TV

Su Spectrum emite una señal de video de televisión colores en las frecuencias correspondientes al canal 36 de la banda de UHF. Por lo tanto, es necesario sintonizar su televisión en este canal para que proyecte la imagen generada por el ordenador.

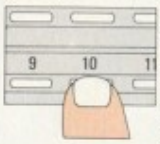
Después de haber conectado su ordenador a la televisión, y después de haberlo conectado, la siguiente etapa es sintonizar la televisión hasta que la pantalla de esta última muestre el mensaje de Spectrum, tal como en la ilustración. Si no puede recibir esta imagen en su televisión, o si los colores no estan bien, le sugerimos que lea las instrucciones a continuación.



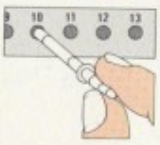
Tipos de Controles de sintonización



Sintonización variable
Un control de sintonización variable selecciona cualquier canal. Haga girar la perilla hasta obtener el mensaje.



Sintonización de botón
Seleccione uno de los botones de sintonización par ser utilizado como el botón de sintonización de su ordenador. Ajuste el botón hasta obtener el mensaje en la pantalla de la televisión. Si fuera posible elija un botón que no sea utilizado para sintonizar canales de televisión, de esta forma evitará tener que reajustarlo cada vez que desee utilizar su televisión para operar el ordenador.





Sintonización electrónica
Este metodo de sintonización es similar al de botón, excepto que el equipo se ajusta automáticamente en el canal seleccionado.

Como verificar los colores del Spectrum

Para verificar los colores del Spectrum basta con oprimir la tecla B y a continuación introducir un número del 1 al 6. A continuación se producen los siguientes eventos: (1) el mensaje desaparece de la pantalla; (2) aparece la palabra BORDER (borde) y el número correspondiente. Ahora oprima la tecla ENTER. El área del "borde" de la pantalla debe cambiar al color indicado en la tecla-número seleccionada. Las pantallas en las ilustraciones continuación muestran que cuando usted introduce en el ordenador (en el orden indicado) las instrucciones BORDER 4 – ENTER, BORDER 3 – ENTER, y finalmente BORDER 7, el color del borde cambia a blanco.

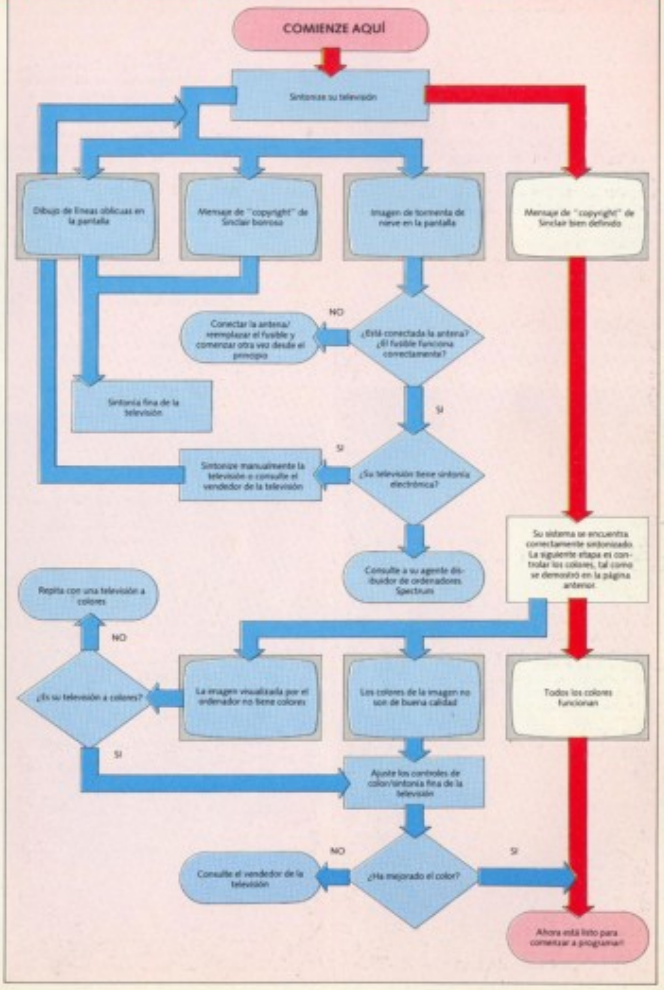
BORDER 4



BORDER 3



Soluciones para los problemas que pueden presentarse al organizar el equipo



CUALES SON LAS CAPACIDADES DE SU ZX SPECTRUM +?

Primero, algunos experimentos

Ahora que su Spectrum está conectado y la televisión ha sido sintonizada correctamente, haga la prueba de oprimir algunas de las teclas del ordenador. Como puede apreciar, inmediatamente aparecen palabras y letras en la pantalla de la televisión, y también algunos números.

Sin embargo, usted debe saber como programar el Spectrum para darle instrucciones sobre que hacer. Pero, no tenga miedo, pruebe las teclas sin temor.

Ahora oprima el botón de reajuste (reset) ubicado en el costado izquierdo de su ordenador, y se encontrará listo para comenzar a trabajar con su Spectrum.

Como introducir informacion o instrucciones en su ordenador

Para introducir cualquier palabra, letra o número en su ordenador: primero tome nota su posicion sobre la tecla y a continuacion utiliza la misma secuencia de teclas selectoras que se indica a continuacion:



Los detalles completos sobre como operar las teclas se er las páginas 20–21.

A continuación: como programar su Spectrum

Su Spectrum puede hacer muchas cosas. Pero para hacerla funcionar es necesario que usted le comunique una serie de instrucciones, que se denominan programa de ordenador.

Los siguientes son una colección de programas breves que le demostrarán algunas de los capacidades de su Spectrum, incluyendo colores, sonidos y gráficas. Todo lo que usted debe hacer es introducir el programa en el ordenador, exactament como se indica. Las imagenes en la pantalla de la televisión le indican el resultado de sus operaciones con el teclado. Si desea experimentar con los programas, puede hacerlo siguiendo las instrucciones en la página opuesta sobre **Como alterar un programa**.

Como introducir y ejecutar un programa

Cada serie de instrucciones es mostrada en una lista, denominada listado. Los listados de los programas contienen varias secciones, cada una de las cuales comienza con un número, por ejemplo **10**, **20**, etc. Cada una de estas secciones es denominada línea del programa.

En cada una de las líneas del programa, se encontrarán palabras completas o abreviaciones compuestas de dos o más letras, tales como **PRINT**, **LED**, **RND**, **PI**, **PAPER** y **GOTO**. Estas son denominadas palabras claves y no es necesario introducirlas letra por letra. Bastará utilizar las teclas correspondientes, cada una de las cuales tiene escrita una palabra clave (por ejemplo la palabra clave **PRINT** se encuentra sobre la tecla P). Para introducir una de estas instrucciones busque la palabra clave correspondiente en el teclado y siga las instrucciones en la sección sobre como introducir información al ordenador en el **capítulo 2**.

A medida que usted introduce la información correspondiente a una línea, mediante el teclado, esta vá apareciendo en la parte inferior de la pantalla de la televisión. Al terminar la línea del programa, oprima la tecla ENTER. La línea cambiará de posición, desplazándose a la parte superior de la pantalla. Repetir el mismo procedimiento para las demás líneas. Si comete un error, consulte la sección sobre **Como corregir errores** en la página siguiente.

Después de haber introducido todas las líneas del programa en su ordenador, oprima la tecla R. La palabra clave **RUN** aparece en la pantalla. Ahora oprima la tecla ENTER y su Spectrum entrará en acción, ejecutando el programa.

NAMES

```
10 BORDER 1: INK RND*7
20 PAPER RND*7
30 PRINT "ZX Spectrum +";
40 GO TO 10
```



El nombre ZX Spectrum + aparece en la pantalla en una variedad de colores. El ordenador se detiene y la pregunta mediante en mensaje en la parte inferior de la pantalla: **scroll?** (en inglés "scroll" significa enrollar). Para desplazar la imagen en la pantalla hacia arriba, bastará con oprimir cualquier tecla excepto: N, SPACE, BREAK o STOP. Si usted detiene el movimiento de la pantalla y oprime las teclas (en el orden indicado): primero BREAK, luego R (RUN) seguida de ENTER, los nombres aparecerán en un patrón de colores diferentes.

Otro experimento

En la línea 3 del programa sustituya "ZX Spectrum +" por su nombre entre paréntesis, por ejemplo

```
30 PRINT "JUAN";
```

No se olvide de incluir el punto y coma(;).

FIGURAS

```
10 LET a$=""
20 FOR x=1 TO 7
30 LET a$=a$+CHR$ (RND*14+129)
40 NEXT x
50 INK RND*7
60 BORDER RND*7
70 PRINT a$;
80 GO TO 70
```



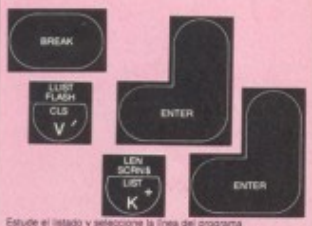
Este programa genera una serie de figuras de colores en la pantalla. Cuando la pantalla está llena, el proceso se detiene y aparece el mensaje "scroll?". Si desea continuar con el programa oprima cualquier tecla (excepto N, SPACE, BREAK o STOP) para desplazar la imagen hacia arriba. Para ver un nuevo tipo de dibujo con una combinación de colores diferente, cuando su ordenador le pregunte "scroll?" introduzca las siguientes instrucciones en el orden indicado: oprima la tecla N, luego BREAK, seguido por R (RUN) y finalmente ENTER.

Pruebe esto

En la línea **20** cambie el número 7 por otro número, para obtener un patrón diferente. Por ejemplo, pruebe introducir el número 8.

Como alterar un programa

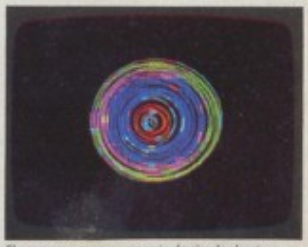
Espere hasta el final del programa o interrúmpalo mediante la instrucción BREAK. A continuación oprima V (CLS), seguida de ENTER, K (LIST), y finalmente de ENTER. Como resultado de estas instrucciones el listado del programa aparecerá en la pantalla.



Estude el listado y seleccione la línea del programa que desea modificar. Introduzca la línea de programa incorporando las nuevas instrucciones o información que desea incluir en por el teclado, incluyendo su número, y finalmente oprima ENTER. La nueva línea aparecerá en el listado. Oprima R (RUN) y ENTER, el nuevo programa entrará en operación.

CIRCULOS LUMINOSOS

```
10 BORDER 0: PAPER 0: CLS
20 CIRCLE INK RND*6: FLASH RND
;120+RND*8,80+RND*5,RND*60
30 BEEP 0.1,RND*60
40 IF RND>.9 THEN GO TO 60
50 GO TO 20
60 FOR y=-2 TO 4
70 FOR x=0 TO 6
80 BORDER x
90 BEEP .05,x+y
100 NEXT x
110 NEXT y
120 RUN
```



El programa genera una serie de círculos luminosos casi concéntricos, de una variedad de colores. De pronto, el borde cambia de colores, el ordenador produce un sonido y aparece un nuevo juego de círculos.

Pruebe esto

Después de listar el programa (LIST), introduzca la instrucción PAPER 7 y oprima la tecla ENTER. A continuación introduzca una nueva línea 20, pero sin las palabras claves **FLASH RND**. Como resultado los círculos no relumbrarán más.

MOSAICO LOCO

```
  5 BORDER 0: CLS
 10 LET h=16: LET v=11
 20 LET v=INT (RND*3-1): LET v=
INT (RND*3-1)
 30 INK RND*7
 40 FOR z=1 TO 20
 50 PRINT AT v,h;CHR$ 143
 60 LET h=h+v
 70 LET v=v+v
 80 IF h<0 THEN LET h=31
 90 IF h>31 THEN LET h=0
100 IF v<0 THEN LET v=21
110 IF v>21 THEN LET v=0
120 NEXT z
130 GO TO 20
```



Un cuadrado de colores se mueve en la pantalla, desarrollando un patrón de colores. Cada vez que el programa es reiniciado el patrón producido es diferente.

Otro experimento
Cambie el número 143 por 42, en la línea 50 del programa, y el cuadrado se transformará en estrellas! Pruebe otros números entre 33 y 142. Consulte la tabla de juego de carácteres en la página 51 para ver que sucederá.

Como reiniciar un programa

Algunos de etos programas, por ejemplo Barras y Estrellas, llegan al final y producen el reporte **0 OK** y el número de la última línea del programa. Esto sirve para indicarle que el ordenador ha ejecutado la totalidad del programa. Para reiniciarlo solamente se requiere oprimir R (RUN) y ENTER.

Otros programas continuan operando, por ejemplo el del Mosaico Loco, o se reinician automáticamente, como Amanecer.



Para detener estos programas debe oprimirse la tecla BREAK.

Oprima esta tecla hasta que el programa se interrumpa y aparezca el reporte BREAK. Para reiniciar el programa oprima R (RUN) y ENTER.

Como corregir errores

Si ha oprimido una tecla equivocada, o si no ha oprimido correctamente las teclas de cambio o EXTEND MODE no se preocupe! Simplemente oprima la tecla DELETE y la última palabra clave, signo, letra, o número desaparecerá. Para borrar más mantenga deprimada la tecla DELETE.



Si ha cometido un error después de haber oprimido ENTER, es posible que aparezca un signo de interrogación destellante. Este es el lugar donde usted ha cometido el error. Oprima DELETE para borrar la línea hasta el signo de interrogación, y a continuación complete la línea del programa correctamente. Finalmente oprima ENTER.

Es posible que usted introduzca en el ordenador una línea conteniendo errores. En este caso el ordenador se detendrá y producirá un reporte del error. El reporte (en la parte inferior de la pantalla) le indicará cual es la línea incorrecta. A continuación usted debe introducir nuevamente toda la línea correctamente y oprimir las siguientes teclas: ENTER,R(RUN) y ENTER. Ahora el programa debería ejecutarse sin más problemas.

POLIEDROS

```
  5 BORDER 1: PAPER 6: CLS
 10 INPUT n
 20 FOR r=20 TO 20 STEP 2
 25 LET x=128: LET y=87
 30 LET h1=r: LET v1=y
 35 PLOT x1,y1
 40 FOR a=0 TO 361 STEP 360/n
 50 LET h2=r*COS (a*PI/180)
 60 LET v2=y+r*SIN (a*PI/180)
 70 DRAW x2-h1,y2-v1
 80 LET h1=h2: LET v1=v2
 85 BEEP 0.02,r-20
 90 NEXT a
100 NEXT r
```



Al principio, este programa, produce una pantalla vacía. Oprima el número 6 seguido de ENTER. Se generará una figura de seis lados. Reinicie el programa y esta vez introduzca otro número para crear una figura diferente.

Pruebe esto
En la línea número 20, del programa cambie el número 2 por otro número. El ordenador aumenta su velocidad si el número es más alto, y los poliedros se separarán.

FRANJAS Y ESTRELLAS

```
 10 INK 2
 20 PAPER 7
 30 CLS
 40 FOR x=28 TO 148 STEP 20
 50 FOR y=0 TO 11
 60 PLOT 16,z+x: DRAU 216,0
 70 NEXT z
 80 NEXT x
 90 PLOT 16,28: DRAU 0,131
100 PLOT 232,28: DRAU 0,131
110 PAPER 1
120 INK 7
130 FOR x=2 TO 8 STEP 2
140 PRINT AT x,2;" * * * * * *"
150 PRINT AT x+1,2;" * * * * * *
"
160 NEXT x
170 PRINT AT x,2;"* * * * * * *"
```



Este programa genera en la pantalla la bandera de los Estados Unidos.

Un experimento
Pruebe cambiar los números de los colores de la bandera. El color de las franjas es indicado en la línea 10, el de las estrellas en la línea 120, y el del fondo de las estrellas en la línea 110.

Como iniciar un nuevo programa

Después de haber terminado un programa y si desea introducir otro totalmente nuevo. En este caso el procedimiento es esperar hasta que el programa corriente se haya completado, o, también, es posible interrumpirlo mediante el comando BREAK.

Es posible elegir entre dos procedimientos para borrar el programa antiguo de la memoria del ordenador. Un procedimiento es oprimir dos teclas: primero A (NEW) y luego ENTER. La pantalla de la televisión se apagará por unos instantes y luego aparecerá el mensaje de Spectrum.



Alternativamente, es posible utilizar un procedimiento más sencillo: basta con oprimir el botón de reajuste o reset. Este tiene el mismo efecto que si usted apagara su Spectrum y la volviera a encender.

AMANECER

```
 10 BORDER RND*6
 20 INK RND*7
 30 PAPER RND*6
 40 CLS
 50 LET z=RND*10+2
 60 FOR x=0 TO 174 STEP z
 70 PLOT 128,0
 80 DRAU -128,x
 90 BEEP .01,x/3
100 NEXT x
110 FOR x=-127 TO 127 STEP z
120 PLOT 128,0
130 DRAU x,175
140 BEEP .01,60
150 NEXT x
160 FOR x=174 TO 0 STEP -z
170 PLOT 128,0
180 DRAU 127,x
190 BEEP .01,x/3
200 NEXT x
210 PAUSE 200
220 GO TO 10
```



Este programa genera un amanecer de variados colores en la pantalla. Si la pantalla queda en blanco, no se asuste, espere un poco y tendrá un nuevo amanecer.

Experimento
Modifique el continido de la línea 20 del programa, cambiando el número 200 por otro, para así alterar el tiempo de duración de cada amanecer. 200 es equivalente a 4 segundos.

Cuál es la siguiente etapa?

Ahora usted puede elegir. Si desea conservar algunos de estos programas para volverlos a usar en el futuro, es posible almacenarlos, registrándolos en cintas de cassette. Para averiguar cual es el procedimiento, consulte la sección **Como almacenar sus programas** en la página 38.

Si desea continuar experimentando con su Spectrum, el Capítulo 2 **Programando su ordenador Spectrum** le ofrecerá toda la información necesaria. Hasta ahora usted ha probado los programas sin realmente comprender como es que funcionan. El Capítulo 2 le explicará varios de los aspectos de la programación de su Spectrum.

Por último, si desea probar algunas cintas conteniendo programas pre-grabados le sugerimos estudiar la sección **Como utilizar programas de ordenador comerciales.**

COMO UTILIZAR PROGRAMAS DE ORDENADOR COMERCIALES

Cuando introduce un programa en el Spectrum desde el teclado, usted produce una secuencia de señales código electrónicas, a medida que oprime cada tecla. Estas señales son dirigidas a la memoria del Spectrum, donde son almacenadas, para ser utilizades por el ordenador, después de que este ha recibido la instrucción de ejecutar el programa. Estas instrucciones permanecerán en la memoria del ordenador hasta que usted las borre (por ejemplo, introduciendo la instruccion NEW, presionando el botón de reajuste (reset) o apagando el ordenador.

Sin embargo, no siempre es necesario introducir el programa a través del teclado. Es posible comprar programas de ordenador pre-grabados comerciales. Estos son alimentados directamente al ordenador en forma automática. El empleo de programas pre-grabados, prontos para ser utilizados, le ahorrará el trabajo de tener que introducir las instrucciones para su ordenador a través del teclado, cada vez que desee utilizar su Spectrum. Otra ventaja es que podrá formar su propia biblioteca de programas listos para ser utilizados, en lugar de tener que escribirlos y alimentarlos al ordenador usted mismo. Los fabricantes de programas de ordenador producen programas de todas clases, los cuales son escritos por los mejores programadores. Existe disponible una amplia variedad de programas para el Spectrum. Le sugerimos que examine el catálogo de programas de Sinclair Spectrum, para estableuer cuales programas se encuentran disponibles. Esta amplia variedad le permitirá seleccionar el programa adecuado para sus necesidades.

El procedimiento para cargar un programa en su Spectrum

Las señales código en una cinta conteniendo un programa, consisten en una serie de notas agudas o graves, registrados a una velocidad de aproximadamente 1500 notas por segundo. Cuando usted pasa una cinta magnética conteniendo un programa en su grabadora, la grabadora produce la secuencia de notas que componen el programa. Al conectar la grabadora al ordenador Spectrum, las señales código en la cinta son transmitidas directamente a la memoria del Spectrum, donde son almacenadas. Este proceso se denomina **"cargar un programa"**.

Estas dos páginas le explicarán como conectar su grabadora de cassettes. Las paginas 14 y 15 le indicarán como utilizar la grabadora.

Preguntas acerca de los programas pregrabados

Qué es un "programa pregrabado" (en ingles "software")?
"Programas pregrabados" o en inglés "software" son instrucciones que se introducen en el ordenador para hacerlo operar. La expresión inglesa hardware, en cambio se refiere al ordenador, y otros equipos anciliares.

Por qué los programas pregrabados son producidos en cassettes?
Las cintas de cassette son fáciles de utilizar y no requieren ningún tipo de equipo especial. Solamente se necesita una grabadora de cassette para este tipo de programas.

Cómo suenan los programas pregrabados en cassettes?
Pase uno de sus cassettes en la grabadora sin conectar esta última al ordenador Spectrum. El sonido es inconfundible y es causado por las señales código que son transmitida al altavoz de la grabadora en lugar del ordenador. Las señales son transmitidas desde el cassette al Spectrum a una velocidad tan alta que que es imposible distinguir entre los sonidos individuales.

Existen otros tipos de programas pregrabados?
Si. Es posible obtener programas almacenados en cartuchos ROM en lugar de cintas de cassette. Estos cartuchos son instalados en un enchufe ubicado en la parte posterior de su Spectrum. Los programas almacenados en un cartucho ROM se cargan inmediatamente.

También es posible obtener programas pregrabados en cartuchos Microdrive. Estos accesorios contienen programas registrados magnéticamente como en las cintas de cassette. Es posible registrar varios programas en la misma cinta, y esta operación solo toma unos segundos, en lugar de minutos como sucede en el caso de los cassettes. Los cartuchos Microdrive son utilizados con la unidad Microdrive (ver la página 46).

Cuál es la mejor grabadora de cassettes?
El Spectrum puede operar correctamente con una grabadora de cassettes portátil económica, perferiblemente conectada a la red eléctrica principal y no utilizando pilas. La grabadora debe tener su propio control de volumen, pero no es esencial que tenga un control de tono. También existen disponibles grabadoras de cassette especiales para funcionar con ordenadores. Estos equipos han sido diseñados para almacenar y cargar programas con más seguridad que los equipos ordinarios.

Los programas registrados en cinta requieren cuidados especiales?
Como cualquier otra forma de almacenamiento mediante impulsos magnéticos, los programas en el cassette son afectados por campos magnéticos de gran intensidad. Por lo tanto, es aconsejable mantener los cassettes lejos de cualquier fuente de electricidad o equipo utilizando energía eléctrica. Los cassettes también deben ser mantenidos limpios y protegidos del polvo.

Es posible utilizar cualquier tipo de programas pregrabados con su Spectrum?
No. Su ordenador solamente puede procesar programas pregrabados producidos especialmente para ordenadores ZX Spectrum +.

Conectando su grabadora de cassettes

Su ordenador viene con un cable especial para conectarlo a la grabadora de cassettes. Es el cable con los dos enchufes pequeños. Ubique la grabadora de cassettes junto a su Spectrum y enchufe el cable tal como se muestra. La grabadora de cassettes y el Spectrum pueden estar encendidos o apagados durante esta operación, aunque es una buena idea verificar que la grabadora no contenga ningún cassette al encender o apagar el ordenador para proteger los programas registrados en la cinta.

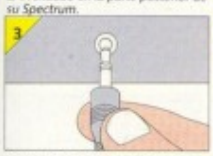
Haciendo las conexiones correctas



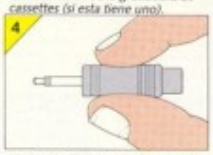
Introduzca cualquiera de los cuatro enchufes en el punto de conexion EAR situado en la parte posterior de su Spectrum.



Introduzca el otro enchufe del mismo color en el punto de conexion EAR de la grabadora de cassettes (si esta tiene uno).



Si su grabadora no cuenta con un punto de conexión EAR, conecte el enchufe en uno de los puntos de conexión para auriculares (si la grabadora tiene uno). También puede tratar de conectarlo al punto de conexión del altavoz externo.



Si el enchufe del cable de conexión de la grabadora no encaja en el punto de conexión de la grabadora, entonces necesitará un adaptador o un cable especial con los enchufes correctos (posiblemente podrá obtenerlo en alguna tienda de artículos eléctricos). El punto de conexión EAR del Spectrum requiere un enchufe macho de 3,5 mm y una señal de entrada de alrededor de 1 volt.

Recuerde que:

- Los enchufes del cable de la grabadora del Spectrum son identificados mediante un código de colores, para evitar errores al conectar dichos equipos entre si. Siempre utilice el mismo sistema al conectar la grabadora al ordenador: un color para los enchufes EAR y el otro para los enchufes MIC.
- Algunas grabadoras son afectadas por otros equipos eléctricos en sus cercanías, lo que puede distorsionar las señales transmitidas entre el ordenador y la grabadora, con el resultado de que no será posible cargar correctamente los programas. Si su grabadora parece no funcionar, pruebe alejarla de la televisión o el ordenador.

Enchufe para EAR y MIC
Al cargar el programa, es posible tener conectados ambos enchufes, tal como se muestra en la ilustración. Pero durante la operación de registro o "almacenamients" (SAVE) de programas (ver página 38) será necesario desconectar el cable EAR.



EL PROCEDIMIENTO PARA CARGAR UN PROGRAMA

Después de haber conectado la grabadora de cassettes a su Spectrum, todo se encuentra dispuesto para cargar y ejecutar un programa. Es posible utilizar una cinta conteniendo programas pre-grabados, listos para ser cargados en el ordenador, o una cinta grabada por usted y conteniendo programas introducidos al ordenador desde el teclado. El procedimiento en ambos casos es idéntico.

Encienda la grabadora. Verifique que el Spectrum también está debidamente enchufado y encendido (ON). Inserte el cassette en la grabadora. Si el ordenador se encuentra en el proceso de ejecutar otro programa espere hasta que sea completado o interrúmpalo, oprimiendo la tecla BREAK. A continuación introduzca en el ordenador la instrucción NEW u opere el botón de reajuste (reset) para limpiar la memoria del Spectrum, aunque esto no es necesario, porque cargar un nuevo programa tiene el efecto de borrar el contenido de la memoria.

Ahora siga las instrucciones numeradas. En caso de algún problema, diríjase a la página 16, y consulte la sección sobre **Soluciones para los problemas que pueden encontrarse al cargar programas pregrabados en el ordenador.**

1 Insertar el cassette y enrrollarlo hasta el principio de la cinta.

2 Ajuste los controles de volumen y tono de la grabadora en los niveles requeridos. Pruebe el nivel de volumen correspondiente a dos tercios del máximo, y si su grabadora cuenta con un control de tono, ajústelo al tono agudo máximo.

3 Oprima las teclas J y el mensaje LOAD debería aparecer en la pantalla. A continuación introduzca el nombre del programa desde el teclado del ordenador entre paréntesis, por ejemplo: **LOAD** "prog1"



4 Oprima la tecla ENTER. La pantalla se pondrá en blanco.

5 Eche a andar la grabadora. El borde de la pantalla cambiará de color, tomándose rojo o azul o una combinación de los dos colores. Esta señal le indica que su Spectrum está buscando el programa.



6 Pocos segundos después unas franjas azules y rojas comenzarán a desplazarse a lo largo del borde. Esta señal indica que el Spectrum ha comenzado a recibir una señal.

7 La palabra **Program** aparece en la pantalla, seguida por el nombre del programa, o **Bytes** seguidas de un nombre o letra. Esta señal le informa que el ordenador ha ubicado el programa que usted le indicara.



8 Las franjas rojas y azules reaparecen, mientras el ordenador espera para cargar el programa.

9 Ahora se visualiza en la pantalla un patrón de líneas amarillas y azules en el borde. Esta señal indica que su Spectrum se encuentra cargando el programa. Esta operación puede tomar varios minutos, según la extension del programa.



10 Las operaciones 7, 8 y 9 pueden ser repetidas varias veces si el programa está dividido en sectores.

11 El programa puede comenzar a operar inmediatamente, una vez cargado. Recuerde parar la grabadora.

12 Cuando el programa no se inicia automáticamente, una vez cargado, la pantalla se pondrá en blanco y aparecerá el reporte **0 OK, 0:1**. Pare la grabadora.



13 Oprima las teclas R (RUN) y ENTER. Ahora es posible ejecutar el programa.



Consejos utiles sobre como cargar programas

Los siguientes son algunos consejos útiles que le ayudarán a ahorrar tiempo al cargar un programa.

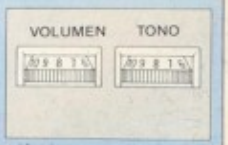
1 Para ubicar los programas rápidamente identifique todas las cintas claramente con rótulos adecuados. Si el cassette contiene más de un programa, anote en el rótulo los nombres de los programas en el orden en que fueron registrados. Escriba el nombre del programa exactamente como lo usará el ordenador.



2 Si su grabadora de cassettes tiene un contador de revoluciones, este se puede utilizar para ubicar la posición del programa requerido en la cinta con varios programas registrados. Enrolle la cinta y ponga el contador de revoluciones en cero. A continuación introduzca la instrucción LOAD seguida del nombre de cualquier programa (entre comillas) que NO se encuentre registrado en esta cinta. Eche a andar la grabadora y el Spectrum irá nombrando cada programa registrado en la cinta, sin cargarlo. Anote el número de revoluciones frente al nombre del programa. Utilizando esta lista usted podrá ubicar fácilmente el programa requerido en el futuro.

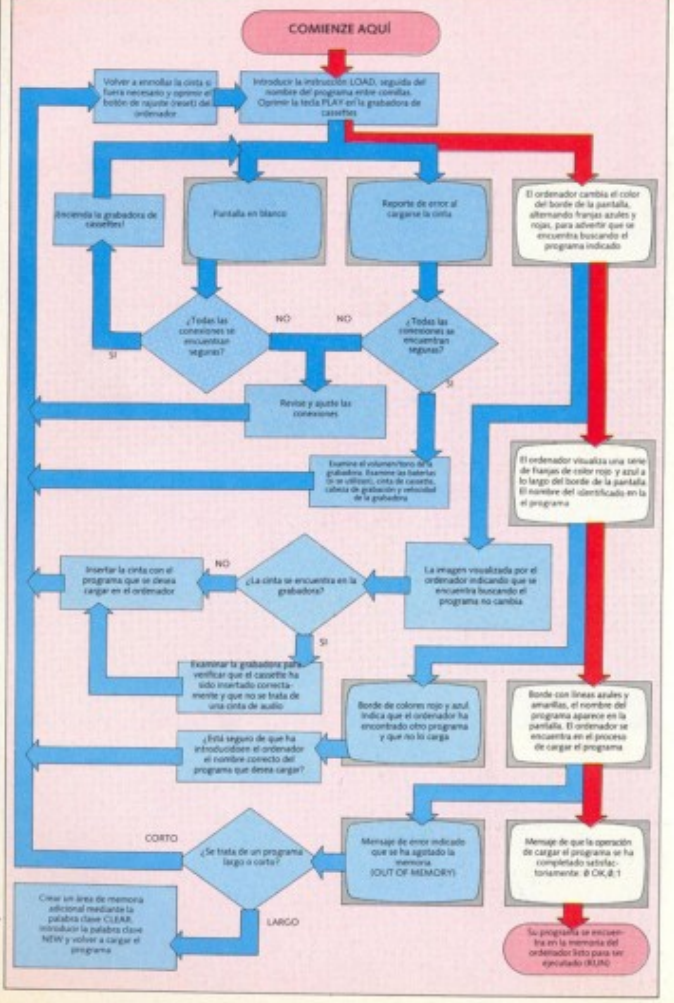


3 Si la cinta se encuentra justo antes del principio del programa requerido, o si no conoce el nombre del programa, entonces basta con introducir la instrucción LOAD"" en lugar de LOAD seguido del nombre del programa entre comillas. Cuide que no exista un espacio entre las dos comillas. Su Spectrum cargará el primer programa que encuentre en la cinta. Si el programa que aparece en la pantalla no es el que usted desea, bastará con oprimir la tecla BREAK, avanzar la cinta y volver a intentar.



4 Pruebe diferentes niveles de sonido y tono para determinar cuales son los que ayudan a su Spectrum a cargar los programas. Ajuste los controles de la grabadora a estos niveles antes de comenzar a cargar el programa.

Soluciones para los problemas que pueden encontrarse al cargar programas prearabados en el ordenador



PROGRAMANDO SU ORDENADOR ZX SPECTRUM +

Este capítulo es una introducción a la programación para el ordenador ZX Spectrum +. Su propósito es explicarle como dar instrucciones al ordenador utilizando el teclado. Mediante estos controles usted podrá poner a trabajar el ordenador. Se incluyen varios programas breves que le demostrarán las características especiales ofrecidas por el Spectrum. En el futuro podrá incorporar estas características en sus propios programas.

2

EL TECLADO – EL PANEL DE CONTROLES DE SU ORDENADOR

El ordenador ZX Spectrum + tiene su propio lenguaje, el lenguaje de ordenadores llamado BASIC. Para hacer que el ordenador obedezca sus instrucciones, usted debe redactar un programa en BASIC utilizando con tal propósito las teclas en el teclado de su Spectrum. Además, el teclado le permite controlar el ordenador mientras éste ejecuta el programa.

El ordenador Spectrum utiliza un dialecto o versión particular del idioma BASIC, que es sencillo pero muy poderoso. Además, el Spectrum tiene una característica muy importante que facilita la tarea de redactar un programa: el sistema de introducción de palabras clave mediante un tecla única.

Teclas y palabras clave

Las palabras clave son palabras con un significado especialmente definido en BASIC, cada una de ellas es una instruccion especifica para el ordenador, como, por ejemplo, PRINT o INPUT. En la mayoría de los ordenadores es necesario introducir las palabras clave letra por letra, como si fuera una máquina de escribir, y es necesario verificar que no se cometen errores. Pero en el Spectrum basta con oprimir la tecla indicada para obtener una palabra clave completa.

El BASIC de Sinclair tiene más de 80 palabras clave, las cuales son introducidas mediante un total de 36 teclas (26 teclas- letra y 10 teclas-número). Muchas de las teclas tienen la capacidad de producir varias palabras clave que serán reconocidas por el ordenador. La mayoría de las teclas le suministran una palabra clave determinada o una letra, número, signo o aún una forma (carácteres gráficos), todos los cuales pueden ser utilizados en sus programas.

Seleccionando palabras clave y símbolos

En el teclado del Spectrum se encuentran dos teclas que utilizará con mucha frecuencia, estas son: EXTEND MODE y SYMBOL SHIFT. El propósito de estas teclas es ayudar a seleccionar las palabras clave y signos que usted desea hacer aparecer en la pantalla, operandolas en combinación con las demás teclas. En primer lugar, le sugerimos que estudie la distribución del teclado. Las siguientes dos páginas le demostraran exactamente como seleccionar cualquiera de las instrucciones que aparecen indicadas en el teclado del ordenador. Después de haber aprendido todo esto, usted se encontrará pronto para escribir sus propios programas.



GRAPH
Esta tecla es utilizada para seleccionar formas o carácteres gráficos en las teclas 1 a 8. Si usted oprime esta tecla y a continuación oprime alguna de las teclas numéro (con o sin la tecla CAPS SHIFT) aparecerá un carácter gráfico en la pantalla. Si desea retornar el ordenador a su modo de operación normal bastará con volver a oprimir la tecla GRAPH.

TRUE VIDEO e INV VIDEO
Estas teclas insertan códigos en las líneas del programa para generar colores normales o invertidos.

TECLAS-COLORES
Estas seis teclas introducen palabras clave que controlan la visualizacion de los colores en la pantalla.

TECLAS-NÚMERO
Generan números, y tambien pueden introducir códigos de control para los colores en el programa. Ver la página 33. Las palabras clave situadas inmediatamente encima de los topes de las teclas 4 a 8 (exceptuando la tecla 8) son utilizadas solamente con las Microdrives ZX.

NEW
Esta tecla limpia el area de BASIC de la memoria, borrando cualquier programa que se encuentre almacenado en ella.

DELETE
Tecla es utilizada cuando se ha cometido un error, por ejemplo si se ha oprimido una tecla equivocada, para borrar una palabra clave, letra, número o signo – ver la pagina 10 por mas detalles.

EDIT
Tecla utilizada para cambiar el contenido de una línea en el programa sin que sea necesario volverlo a escribir totalmente. Ver la pagina 21.

EXTEND MODE
Esta tecla selecciona la palabra clave superior situada sobre el tope de cada tecla. Cuando se oprime esta tecla seguida de SYMBOL SHIFT y otra tecla, se seleccionara la palabra clave indicada inmediatamente encima del tope – ver las páginas 20–21.

CAPS SHIFT
Al oprimir esta tecla conjuntamente con una tecla correspondiente a una letra se producirá una mayúscula. Si desea generar una serie de letras mayúsculas utilize la tecla CAPS LOCK.

CAPS LOCK
Esta tecla produce letras mayúsculas. Para cambiar de mayúsculas a minúsculas bastará con volverla a oprimir.

BEEP
Esta tecla produce la palabra clave que controla el sintetizador de sonido del Spectrum.

BARRA ESPACIADORA
Produce un espacio como en una máquina de escribir.

CONTROLES DE LOS CURSORES
Son las teclas que controlan el movimiento del cursor. Cada tecla lo hace desplazarse en la dirección de la flecha. Estas teclas son utilizadas frecuentemente en los programas, para controlar el movimiento de figuras en la pantalla. También se las utiliza para editar programas.

BREAK
Esta tecla detiene el programa que está siendo ejecutado. No borra el programa de la memoria del ordenador.

ENTER
Oprima esta tecla para introducir una línea de programa en la memoria del ordenador. Esta tecla se utiliza además para alimentar información al computador durante un programa.

SYMBOL SHIFT
Mantenga oprimida esta tecla y oprima una tecla-letra o tecla-número para seleccionar la palabra clave o signo indicado en el tope de dicha tecla. Utilizada después de EXTEND MODE, esta tecla seleccionará el símbolo o palabra clave indicada inmediatamente encima del tope de la tecla. Ver las páginas 20–21.

COMO OPERAR LAS TECLAS

Es posible obtener hasta seis palabras claves, letras, números o signos de la mayoría de las teclas de su ZX Spectrum +. Sin embargo, seleccionar un carácter o palabra clave en el teclado no es una operacion difficil, una vez que usted se familiarize con las características especiales de su ordenador. El resultado obtenido al oprimir una tecla dependerá del "modo" en el cual el ordenador se encuentra en ese momento. Cada modo determina que la tecla produzca una información diferente como, por ejemplo, una letra, una palabra clave o un carácter gráfico. La ventaja de este sistema es que el Spectrum le ayudará a seleccionar el modo apropiado, para introducir en el orden correcto, instrucciones e información en el ordenador.

Modo de palabra clave

Conecte y reajuste su Spectrum para que aparezca el mensaje inicial. Ahora oprima la tecla ENTER. El ordenador hará aparecer una letra K a destellos en el rincón de abaja a la izquierda. Este cuadro luminoso es el *cursor*. El cursor le señala el punto donde algo aparecerá en la pantalla, y la letra K le advierte que el ordenador se encuentra en el modo de *palabra clave*. Oprima cualquier tecla-letra y la palabra clave situada en el tope de la sección elevada de la tecla aparecerá en la pantalla. Por ejemplo, oprima la letra Q, y la palabra clave PLOT aparece en la pantalla. Oprima la tecla DELETE para borrar esa palabra clave y pruebe las demás teclas. Las teclas-número le darán números, pero tan pronto como usted oprime una tecla-letra, aparecerá la palabra clave superior en la sección elevada de la tecla.

Utilize DELETE otra vez para volver a hacer aparecer el cursor K. A continuación oprima cualquiera de las teclas SYMBOL SHIFT, manténgala deprimida y oprima cualquier otra tecla-letra. Ahora apareca la palabra clave o signo ubicado justo encima de la letra, en la sección elevada de la tecla. En el caso de las teclas-número aparecerá el signo a la derecha en la sección elevada.

Como seleccionar una palabra clave, símbolo o carácter

A continuación demonstraremos como seleccionar cualquiera de las palabras claves, símbolos o carácteres indicados en las teclas-letra o teclas-número. Cuando necesite seleccionar una de las funciones indicades en las teclas, examine primero la posición de la tecla deseada. Después, determine cual es el modo del teclado en ese momento. Si fuera necesario cambie el modo del teclado para obtener el resultado deseado. A continuación se incluye un ejemplo para demostrar cual es el procedimiento requerido para cambiar el modo del teclado.



Tecla-letra

Modo Palabra Clave (K)	
Tecla sola	BORDER
SYMBOL SHIFT + tecla	*
EXTENDED (E) MODE	
EXTEND MODE seguido de tecla sola	BIN
EXTEND MODE luego SYMBOL SHIFT y tecla	BRIGHT
Modo Letra (L)	
Tecla sola	b
CAPS SHIFT y tecla	B
SYMBOL SHIFT y tecla	*
Modo Mayúsculas (C)	
CAPS LOCK seguido por tecla sola	B
CAPS LOCK seguido por SYMBOL SHIFT y otra tecla	*
Modo Gráfico (G)	
GRAPH entonces teclas-letra entre A-U solamente	efectos gráficos seleccionados por el usuario



Tecla-número

Modo Palabra Clave (K)	
Tecla sola	3
SYMBOL SHIFT entonces tecla	#
EXTENDED (E) MODE	
EXTEND MODE seguido de tecla sola	magenta
EXTEND MODE luego SYMBOL SHIFT y tecla	LINE
Modo Letra (L)	
Tecla sola	3
SYMBOL SHIFT luego tecla	#
Modo Mayúsculas (C)	
CAPS LOCK entonces tecla sola	3
CAPS LOCK luego SYMBOL SHIFT y tecla	#
Modo Gráfico (G)	
GRAPH seguido de tecla sola	▀
GRAPH luego CAPS SHIFT y tecla	▄

Modo de letras y mayúsculas

Después de haber producido una palabra clave o signo operando en el modo Palabra Clave, el ordenador automáticamente cambia el modo del cursor al Modo L. El Modo L es también conocido como el Modo Letra (letter mode). Cuando en el Modo L, si usted oprime una tecla-letra, inmediatamente se visualiza dicha letra en minúscula. Igualmente, si oprime una tecla-número aparecerá el número correspondiente. Si desea obtener letras mayúsculas, bastara con oprimir la tecla CAPS SHIFT (cambio de mayúsculas) seguida de la tecla con la letra en cuestion.

Si desea escribir solamente con letras mayúsculas, oprima la tecla CAPS LOCK. El cursor cambia a la letra C. Su Spectrum se encuentra ahora en el Modo de Mayúsculas (capitals mode): cada vez que usted oprime una tecla-letra el ordenador imprimirá una letra mayúscula en la pantalla. El procedimiento para retornar al Modo Letra (L) es simplemente oprimir nuevamente la tecla CAPS LOCK.

EXTENDED MODE

El siguiente modo es denominado Extended Mode y es introducido presionando la tecla EXTENDED MODE. Como resultado el cursor cambia a la letra E. Después de haber cambiado a este modo, oprima cualquier tecla-letra y podrá apreciar que el ordenador presenta en la pantalla la palabra clave superior, del par de palabras clave situadas encima de la sección elevada de la tecla. Por ejemplo: ponga el ordenador en el Extended Mode y luego oprima la tecla B, el resultado en la pantalla será la palabra clave **BIN**. Para obtener la palabra clave inferior, o el signo encima de la sección elevada, el procedimiento es: (1) oprimir cualquiera de las dos teclas SYMBOL SHIFT, (2) mantener la tecla en la posición deprimida, (3) y oprimir la tecla-letra correspondiente. Por ejemplo, en el caso de la tecla-letra B, siguiendo este procedimiento se obtiene la palabra clave **BRIGHT**. Después de haber sido oprimida una tecla (la tecla EXTEND MODE) mientras el ordenador se encuentra en Extend Mode, este automáticamente retornará al Modo Letra o al Modo Mayúsculas.

Modo Gráfico

El quinto modo es el Modo Gráfico (Graphics Mode), y es introducido mediante la tecla GRAPH. A continuación el cursor cambia a la letra G. Oprima cualquiera de las teclas-número, del 1 al 8, y podrá ver como el carácter gráfico correspondiente aparece en la pantalla. A continuación oprima la tecla CAPS SHIFT y cualquier tecla-número, del 1 al 8. Los símbolos gráficos aparecen nuevamente, pero esta vez los colores blanco y negro han sido invertidos. Para abandonar el Modo Gráfico se oprime otra vez la tecla GRAPH, desde que el ordenador no abandonarlo automaticamente.

Editando programas

Cuando usted escriba sus propios programas para su Spectrum, deseará corregir los errores, ya sea en comandos o en líneas del programa, o efectuar alteraciones. Usted puede hacerlo facilmente editando el programa.

Como corregir un error

Cuando se introduce en el ordenador una línea en BASIC con errores o un comando equivocado, el Spectrum mostrará en la pantalla un ? destellante delante del del error. Para corregir el error, oprima la tecla de control de cursor de la derecha o izquierda para mover el cursor a la derecha del error. A continuación borre el error oprimiendo la tecla DELETE, o agregue los elementos necesarios. Finalmente oprima la tecla ENTER.

Por ejemplo, suponga que desea que su ordenador multiplique 7 por 8 pero se olvida de oprimir la tecla SYMBOL SHIFT para obtener el símbolo *. Como consecuencia, usted habria introducido.

PRINT 7b8

Su spectrum no puede obedecer esta instrucción, por lo tanto cuando usted oprime la tecla ENTER, el ordenador mostrará un signo de interrogación destellante delante de la b, el lugar donde se encuentra el error. Todo lo que usted tiene que hacer es: mover el cursor justo a la derecha del error; oprimir la tecla DELETE para borrar la b; oprimir la tecla SYMBOL SHIFT y la tecla B para obtener *; y oprimir la tecla ENTER para que el ordenador obedezca el comando correcto. No es ecesario mover el cursor hasta el final de la línea.



Como editar una línea de programa

Al escribir un programa, usted construye una secuencia de líneas numeradas dando instrucciones, la cual se denomina listado. Si después de haber escrito su programa, efectúa el listado oprimiendo las teclas K(LIST) y ENTER, verá el signo > próximo a una de las líneas del programa. Si no aparece, debe oprimir una de las teclas de control del movimiento vertical del cursor. Al oprimirse la tecla EDIT, el ordenador copiará la línea correspondiente en la parte inferior de la pantalla y se puede modificarla como antes con las teclas del cursor y DELETE. Oprima la tecta ENTER para colocar la nueva línea dentro del programa. Si desea editar otra línea, mueva el símbolo > mediante las teclas de control de movimiento vertical del cursor, hasta la línea que desea modificar y oprima la tecla EDIT. Si esta operación toma demasiado tiempo, introduzca LIST seguido del número de la línea y oprima EDIT. En cada caso, la línea requerida aparecerá al fondo de la pantalla y se le puede cambiar.

Para borrar una línea completa del programa, introduzca el número de la línea y oprima la tecla ENTER. Si introduce un programa que contiene un error, un informe de error aparecerá en la pantalla. Ver la página 74 por más detalles sobre estos informes.

UNA CALCULADORA EN SU TELEVISION

Su ZX SPECTRUM + puede hacer cálculos muy rápidamente y con gran precisión. Todo lo que necesita son algunos números y signos, tales como + y −, que le indiquen cuales son las operaciones que debe efectuar con los números.

Primero, introduzca la siguiente instrucción (el signo de + se encuentra en la tecla-letra K):

```
PRINT 6+2
```

Esta instrucción es un "comando". Cuando usted oprime la tecla ENTER, el comando desaparece de la pantalla y es sustituido por el resultado. En este ejemplo el resultado es el número 8.

Su Spectrum utiliza cinco signos conocidos como operadores artiméticos para los cálculos. El panel en la parte inferior de esta página explica la función de cada uno de ellos. El procedimiento para todos ellos es similar, siempre mediante la instrucción PRINT.

El Spectrum también puede visualizar en la pantalla los cálculos y su resultado. Introduzca el siguiente comando en su ordenador:

```
PRINT "6+2=";6+2
```

El ordenador responderá, visualizando:

```
6+2=8
```

Ha sucedido lo siguiente: el comando PRINT determina que todo lo que se encuentra entre ambas comillas ("") se visualize en la pantalla, en nuestro ejemplo: 6+2=.

El punto y coma le indica al ordenador que el resultado debe ser ubicado inmediatamente después del signo de igual (=).

Los signos utilizados por el Spectrum para sus cálculos

El ordenador emplea los siguientes operadores artméticos para efectuar operaciones matemáticas. Observe que el ordenador no utiliza los símbolos × o ÷.

Símbolo	Tecla	Función	Ejemplo
+	K	Suma dos números	8+2=10
−	J	Resta dos números	8−2=6
*	B	Multiplica dos números	8*2=16
/	V	Divide dos números	8/2= 4
↑	H	Eleva un número al cuadrado	8↑2=64

Su primer programa

Después de haber ejecutado un comando, su Spectrum lo olvida. Si usted desea que su ordenador repita los cálculos, entonces le sugerimos escribir un programa. Introduzca las instrucciones en el ordenador y luego oprima la tecla ENTER.

```
10 PRINT 6+2
```

Esta vez sus instrucciones no son obedecidas inmediatamente. En cambio, el ordenador visualiza la instrucción en la pantalla. A continuación oprima R (RUN) y la tecla ENTER. Inmediatamente aparece el resultado de la operación: 8.

La totalidad de la instruccion es ahora un programa de ordenador. Al poner un número al principio, su Spectrum almacena la instrucción en su memoria pero no la ejecuta hasta que se lo indique. Cuando introduce un programa oprimiendo R (RUN) seguido de ENTER, su Spectrum ejecutará la instrucción. La instrucción se transforma en una sentencia en lugar de un comando y forma una línea numerada en un programa. Las sentencias en un programa son siempre ejecutadas de acuerdo al número de línea. Estos normalmente van de diez en diez.

Ahora podemos poner a trabajar el Spectrum. Introduzca el programa a continuación. Recuerde oprimir la tecla ENTER después de haber introducido cada línea en el ordenador y al terminar oprima la tecla R (RUN) y ENTER. Una vez ejecutado el programa, el resultado debería ser el siguiente:

TABLA DE NÚMEROS

```
10 LET n=1
20 PRINT n;" ";
30 LET n=n+1
40 GO TO 20
```



La tabla visualiza todos los números entre 1 y 203. Ahora oprima cualquier tecla excepto N, la barra espaciadora, STOP o BREAK. A continuación aparece un juego completo de otros números.

Este programa utiliza una "variable". En este ejemplo la variable se denomina n. Es posible utilizar cualquier letra o palabra; en nuestro ejemplo n es la abreviación por el término número. Se ha adjudicado un valor a esta variable que cambia a medida que se ejecuta el programa. En la línea de programa 10, se ha incluido la palabra clave LET para adjudicar el valor 1 a la variable n. La línea de programa 20 visualiza el valor seguido de un espacio. Seguidamente, en la linear 30 volvemos a utilizar la palabra clave LET, esta vez para incrementar el valor de la variable en 1. Como resultado de esta instrucción, el valor de la variable ahora es 2 (1+1). La línea 40 contiene la palabra clave GOTO (en inglés: vé a) para dirigir al ordenador nuevamente a la línea 20, que ahora visualiza el número 2.

Este ciclo se repite una y otra vez hasta que la pantalla de su televisión está llena de números.

Como enseñarle a su ordenador a pedir números

Interrumpa el programa oprimiendo la tecla BREAK. Ahora introduzca una nueva línea.

```
10 INPUT n
```

Esta nueva línea de programa sustituye la línea 10 anterior. Cuando se ejecuta el programa, el ordenador esperará a que usted introduzca un número (INPUT en inglés significa introducir). Introduzca cualquier número desde el teclado de su ordenador y oprima la tecla ENTER. Ahora, los números de la tabla visualizada en la pantalla de la televisión, comienzan a partir del número que usted introdujera en el ordenador. Esto se debe a que INPUT n determina que el valor de la variable n sea idéntico al número que usted introdujera.

Programa para una tabla de multiplicar

Oprima el botón de resta para borrar de la memoria del ordenador el programa anterior. Introduzca en el ordenador el siguiente programa que hace que su Spectrum multiplique. Introduzca cualquier número mediante el teclado en el ordenador, y este producirá la tabla de multiplicar para ese número en la pantalla. Si desea que la tabla de multiplicar continue oprima cualquier tecla, excepto N, BREAK, o la barra espaciadora. Oprima BREAK y vuelva a ejecutar el programa para crear una nueva tabla. Ahora le mostramos el programa para la tabla y los resultados que se obtienen al introducir 3 y a continuación 146.

TABLA DE MULTIPLICAR

```
10 LET x=1
20 INPUT n
30 PRINT n;"*";x;"=";n*x
40 LET x=x+1
50 GO TO 30
```





Por qué es necesario utilizar paréntesis?

En ciertos casos será necesario emplear paréntesis al realizar cálculos. Por ejemplo, introduzca los siguientes dos comandos y compare los resultados:

```
PRINT 6+2/4
PRINT (6+2)/4
```

En el primer caso el resultado obtenido será 6.5 y en el segundo el resultado será 2. La causa de estos resultados diferentes es que el ordenador tiene incorporado un sistema de prioridades que utiliza durante sus cálculos. El orden de prioridades es el siguiente: primero ejecuta el signo ↑ ; en segundo lugar * o /; y finalmente + o −. Pero en todos los casos siempre ejecutará primero las operaciones contenidas entre paréntesis. Por lo tanto, en el caso del primer ejemplo el ordenador: primero dividió 4 por 2, y luego le sumó el resultado de la división al número 6. En el segundo ejemplo: el computador efectuó en primer lugar la operación entre paréntesis (sumo 6 más 2) y luego dividió el resultado por 4.

La puntuación en su Spectrum

El Spectrum hace uso de una variedad de signos de puntuación. Todos ellos cumplen papeles muy importantes debido a que operan como instrucciones para el ordenador, afectando la forma en que este interpreta las líneas de programa o visualiza información en la pantalla de la televisión.

- **Punto y coma** Cuando utilizada conjuntamente con la palabra clave PRINT, el punto y coma le indica al ordenador que debe visualizar en la pantalla dos elementos uno al lado del otro.
- **Dos puntos** En una línea de probrama señala el final de una sentencia y el comienze de la próxima.
- **Comillas** Cualquier carácter o serie de caracteres contenidos entre comillas son tratados como texto y visualizados en la pantalla, no son considerados como números o variables. Las comillas marcan el principio y el final de una "cadena" (string).
- **Coma** Cuando utilizada con la palabra clave PRINT, la coma le indica al ordenador que visualize el elemento siguiente, ya sea en el centre de la línea o al principio de la línea siguiente.
- Punto decimal o punto.

COLORES, COMO EMPLEARLOS

Su ZX Spectrum + puede producir ocho colores diferentes. Cada color tiene su propio número código. Es posible utilizar cada color en tres formas diferentes: como un color de borde, como un color de tinta, y finalmente como un color de papel.

ZX Spectrum +. Codigos de los colores.

La tabla enumera los colores utilizados por el Spectrum y sus números código respectivos. No es necesario memorizar los números porque las teclas-número que los producen indican cual es el color correspondiente a cada una de ellas.

Número	Color
Ø	Negro
1	Azul
2	Rojo
3	Magenta (Violeta)
4	Verde
5	Verde-azul
6	Amarillo
7	Blanco

El tono de los colores en la pantalla de su televisión dependerá del equipo y el ajuste de sus controles de color, contraste y brillo. Recuerde que se requiere una televisión de colores para este propósito.

Los tres modos de empleo del color en su Spectrum

Los colores de su ordenador pueden ser controlados de tres maneras diferentes. El color de borde (border color) es el color del área en torno del sector central de la pantalla. Los colores de tinta (ink colours) son los colores de los carácteres en la pantalla (letras, números, signos, y formas gráficas), puntos o líneas. El color de papel (paper color) es el color del fondo, ya sea en la totalidad del área de visualización o en cuadrado en torno a cada uno de los carácteres.

Cuando usted enciende su Spectrum, el ordenador emplea una serie de colores preestablecidos. La tinta es de color negro, mientras que borde y papel son de color blanco. Es posible cambiar estos colores simplemente, mediante comandos introducidos directamente desde el teclado. Ya hemos visto antes como cambian los colores en la televisión, en las páginas 6 y 7. Ahora oprima el botón de reajuste (reset) e introduzca el siguiente sencillo programa desde el teclado de su ordenador.

PRUEBA DE LOS COLORES

```
10 PRINT "a";
20 GO TO 10
```



Al ejecutar este programa, el ordenador produce un patrón de estrellas en blanco y negro. Ahora oprima la tecla de BREAK e introduzca algunos nuevos comandos de control para los colores. Introduzca las palabras clave BORDER, INK y PAPER, cada una de ellas seguida de un número de Ø a 7. Después de cada instrucción oprima la tecla ENTER y vuelva a ejecutar el programa. Los siguientes son dos ejemplos, el primero con la instruccion BORDER 4, PAPER 2, e INK 7. El segundo ejemplo contiene las instrucciones BORDER 3, PAPER 5, e INK 1.





Como utilizar colores en sus programas

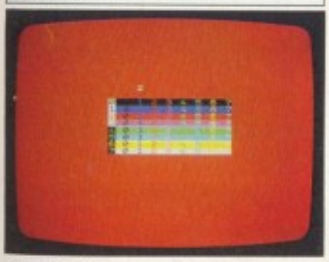
Es posible utilizar las palabras clave BORDER, PAPER, e INK en su programa, para crear textos, tablas, patrones, y dibujos de una variedad de colores.

Cuando se incluye la palabra clave BORDER en una línea de programa, el color del borde cambia cuando el ordenador llega a ese punto. La palabra clave INK independientemente cambia el color de los carácteres o líneas la próxima vez que estas aparezcan en la pantalla. Finalmente, la palabra clave PAPER incluída independientemente en una línea de programa cambia el color del papel, pero solamente en torno de los caracteres (incluyendo puntos y líneas) en la pantalla. Si desea que la totalidad del fondo del área de visualización tenga un color determinado, lo que debe hacer es introducir las palabras clave PAPER y CLS.

También es posible emplear INK y PAPER después de PRINT. En este caso, dichas instrucciones solamente afectarán los carácteres específicos controlados por PRINT. El programa siguiente demuestra todos los colores de borde (border), ink (tinta) y de papel (paper). El programa también le demuestra como utilizar las palabras clave INK y PAPER después de PRINT.

COMBINACIONES DE COLORES

```
 10 FOR b=0 TO 7
 20 BORDER b: PAPER b: CLS
 30 PRINT AT 6,12; INK 9;b
 40 FOR p=0 TO 7
 50 PRINT AT p+8,8; INK p; PAPE
R 9;p
 60 BEEP 0.5,b+p-20+p
 70 FOR i=0 TO 7
 80 PRINT INK i; PAPER p;" ";i;
 90 BEEP 0.01,i+5
100 NEXT i
110 NEXT p
120 NEXT b
```



Al ejecutar este programa, usted verá todas las combinaciones de borde, papel y tinta (o en inglés: border, paper e ink). El programa incluye tres variables: b para el numero del color correspondiente al borde; i controlando el número correspondiente a la tinta (ink); y p, que es el número del papel (paper). La palabra clave BEEP produce el sonido, y las líneas del programas comenzando con FOR y NEXT marcan el principio y el final de un bucle del programa. El propósito de este bucle es cambiar todos los números de los colores desde Ø hasta el 7 consecutivamente. Observe que INK y PAPER pueden tener un valor de 9. Este número código causa que la tinta o el papel se tomen de color blanco o negro, para que resalten contra el fondo o un carácter.

Programa para crear un gráfico de colores

El programa siguiente utiliza los colores de su Spectrum para dibujar un gráfico de barras. El propósito del gráfico es mostrar las temperaturas diumas producidas durante un período determinado. Las temperaturas son mostradas gráficamente como barras amarillas y con números. En la línea del programa 6Ø tenga cuidado de dejar dos espacios en blanco entre las comillas (" ").

GRAFICO DE BARRAS

```
 10 BORDER 0: PAPER 1: CLS
 20 LET c=4
 30 FOR x=1 TO 12
 40 READ t
 50 FOR i=21 TO 21-t STEP -1
 60 PRINT PAPER 6;AT i,c;"  "
 70 NEXT i
 80 PRINT INK 2;AT 20-t,c;t
 90 LET c=c+2
100 NEXT x
110 DATA 20,15,13,16,19,20,18,1
1,12,19,14,17
```



Ahora le sugerimos agregar las siguientes líneas de programa, incluyendo la nueva línea 11Ø, tal como se muestra. Como resultado de estas nuevas instrucciones, el gráfico será de dos colores.

GRAFICO DE BARRAS DOBLE

```
 85 READ t
 86 FOR i=21 TO 21-t STEP -1
 87 PRINT PAPER 3;AT i,c;"  "
 88 NEXT i
 89 PRINT INK 1; PAPER 5;AT 20-
t,c;t
 110 DATA 20,6,15,4,13,5,16,6,19
,10,20,8,18,6,11,4,14,6,19,8,14,
9,17,7
```



GRAFICAS SENCILLAS

Su ordenador ZX Spectrum + le ofrece facilidades para gráficas de alta resolución y de baja resolución. Ambos tipos de gráficas pueden aparecer en la pantalla al mismo tiempo. Las imagenes gráficas en el modo de baja resolución consisten en bloques de color. El propósito de estas dos páginas es explicar como producir estos bloques de color y como ubicarlos en la pantalla mediante instrucciones introducidas en el teclado de su ordenador.

La pantalla de baja resolución

En el modo de baja resolución la pantalla de la televisión se divide en una cuadrícula, cuyas dimensiones son 32 líneas verticales o columnas y 22 líneas horizontales. Cada posición en la pantalla es identificada por dos números, que son sus coordenadas. En primer lugar se indica el número de la línea en la cual se encuentra el punto. Las líneas se cuentan a partir de la parte superior de la pantalla, y el valor de la línea en el tope es , el valor de la línea en la parte inferior de la pantalla es 21.

A continuación se indica el número de la columna, éstas se cuentan de izquierda a derecha. La primera columna a la izquierda tiene el número y la última a la derecha tiene el número 31. (En la página 8 encontrará un diagrama de la cuadrícula de baja resolución)

El siguiente programa llenará las posiciones de los carácteres con colores. La palabra clave RND (la encontrará en la tecla R) seleccionará al azar un color de tinta.

CUADRODOS

```
10 BORDER 1: INK RND*7
20 PRINT "■";
30 GO TO 10
```



El programa genera cuadrados en toda la pantalla de la televisión. Para indicarle al ordenador que visualize un caracter en una posición determinada de la pantalla, deberá insertar las palabras clave PRINT y AT. La palabra clave AT es insertada después de PRINT y es seguida por el número de la líneas, una coma, el número de la columna, y un punto y coma.

Por ejemplo, el comando

PRINT AT 11,16;"*"

visualizará un asterisco en el punto situado en la intersección de la líneas 11 con la columna 16, esto es, el centro de la pantalla.

Como dibujar un arco iris

Un buen procedimiento para producir imagenes de colores es incluir bucles FOR NEXT en los programas de gráficas. Los bucles FOR NEXT son grupos de instrucciones contenidas en un programa, que el ordenador repite una cantidad determinada de veces.

Como seleccionar caracteres gráficos

El teclado de su ZX Spectrum cuenta con un juego de caracteres gráficos, que facilitan la programación de gráficas de baja resolución. Los caracteres se encuentran en las teclas-número del 1 al 8.

Para visualizar los caracteres gráficos en la pantalla se oprime la tecla GRAPH, y a continuación alguna de las teclas-número, del 1 al 8, utilizando la barra espaciadora entre cada número. Los caracteres gráficos aparecen en la pantalla. La parte blanca de cada cáracter en la tecla es el color de tinta (y la parte negra es el color de papel). Vuelva a oprimir las teclas, manteniendo oprimida la tecla CAPS SHIFT al mismo tiempo. El cáracter gráfico aparecerá con los colores de tinta y de papel invertidos.

Este es el procedimiento empleado para incluir caracteres gráficos en las líneas de sus programas. Para abandonar el modo de gráficas y retornar las teclas-número a su estado normal, vasta con oprimir nuevamente la tecla GRAPH.

GRAPH
Tecla empleada para poner el Spectrum en el modo de gráficas.

Tecla 8
Esta tecla esutilizada con GRAPH y CAPS SHIFT para producir un cuadrado de color sólido.



El programador le indicará al ordenador la cantidad de veces que éste debe ejecutar el bucle, mediante una instrucción contenida en la primera línea del bucle. Este método puede ser utilizado, por ejemplo, para visualizar carácteres en la pantalla de la televisión.

Es posible programar más de un bucle al mismo tiempo. También es posible ubicar un bucle dentro de otro, con resultados muy útiles. El siguente programa le enseñará como se utilizan dos bucles (uno "anidado" dentro del otro) para cambiar los colores y las posiciones producidas en la pantalla por las palabras clave INK y AT. Las explicaciones al final de la página le indicarán como programar estos bucles.

ARCO IRIS

```
5 BORDER 0: PAPER 5: CLS
10 LET x=1
20 FOR l=0 TO 21
30 FOR c=1 TO 6
40 PRINT INK c;AT l,c+x;"■"
50 NEXT c
60 LET x=x+1
70 NEXT l
```



Programando imagenes

Cuando en el modo de gráficas de baja resolución, es posible "pintar" imagenes mediante el procedimiento de determinar las coordenadas de las posiciones y los colores de los carácteres gráficos. Para planear una figura le sugerimos que utilize una cuadrícula para imagenes de baja resolución como la incluida en la página 80.

A continuación, seleccione los carácteres gráficos, tal como se explica en la página opuesta, e introduzca en el ordenador, mediante el teclado, las líneas del programa, una a una, para construir la imagen.

El programa siguiente la demuestra el tipo de resultados que puede producir con sus programas. Todas las formas incluidas en la imagen pueden encontrarse en las teclas-número del teclado. Es posible introducir todas las líneas del programa antes de ejecutarlo. También puede observar como el robot es construido elemento tras elemento, al introducirse cada nueva línea de programa, si le indica al ordenador que ejecute las líneas del programa en su memoria, después de introducir una nueva línea.

ZX EL ROBOT

```
5 BORDER 2: PAPER 1: CLS
10 PRINT INK 6;AT 3,15;"■"
15 PRINT INK 6;AT 4,15;"■"
20 PRINT INK 6;AT 5,15;"■"
30 PRINT INK 6;AT 6,13;"■■■■■"
40 FOR l=7 TO 10: PRINT INK 5;
AT l,13;"■ ■": NEXT l
45 PRINT INK 6; PAPER 0;AT 8,1
5;"2x"
50 PRINT INK 2;AT 11,13;"■
■"
60 FOR l=11 TO 15: PRINT INK 6
;AT l,14;"■ ■": NEXT l
70 PRINT INK 3;AT 16,13;"■■";T
AB 17;"■■"
80 FOR l=17 TO 21: FOR c=0 TO
31
90 PRINT INK 4;AT l,c;"■"
100 NEXT c: NEXT l
```



En la línea 70 del programa, la palabra clave TAB, después de PRINT, ubica a carácter a lo largo de la línea donde el ordenador se encuentra trabajando en ese momento. TAB es seguida por un número entre 0 y 31, que especifica la posición de una columna.

Como emplear los bucles FOR NEXT

Un bucle FOR NEXT siempre se inicia con una línea de programa conteniendo las palabras clave FOR y TO, junto con la variable y su valor inicial y final. Por ejemplo

30 **FOR** c=1 TO 6

En el ejemplo, la variable es c. El bucle iniciado de esta forma contendrá una línea (o líneas) que le indican al ordenador que repita una operación. Las líneas también pueden incluir la variable c. Los bucles FOR NEXT siempre terminan con la palabra clave NEXT seguida por la variable. Por ejemplo

50 **NEXT** c

El ordenador executar el programa repitiendo el bucle FOR NEXT completo una cantidad indicada de veces. El valor incial de la variable, indicado antes de TO, se incrementa en pasos de 1, hasta que alcanza el límite establecido después de TO. En el ejemplo, el bucle se repite seis veces, el valor incial de c es 1 y cambia sucesivamente a 2, 3, 4, 5 y finalmente a 6.

En el primer programa de pagina 25, se utilizan tres bucles en un 'nido'. Esto significa que para cada ciclo del bucle 'exterior', el bucle 'intermedio' completa su proceso, y el bucle 'interior' completa sus procesos con más frecuencia.

DIBUJANDO CON SU ORDENADOR

Las posibilidades de su ZX Spectrum + son enormes. Gracias a su capacidad de generar figuras sumamente nítidas, de gran resolución (high-resolution), es posible utilizar el Spectrum para crear imagenes detalladas con líneas claras, y empleando líneas rectas o curvas.

Los dibujos y demas gráficas de gran nitidez se componen de una cantidad de pequeños punto situados uno junto al otro, formando líneas, o rellenando espacios (por ejemplo un rectángulo). El tamaño de cada uno de estos puntos es 1/64 parte del tamaño de los cuadrados empleados en gráficas de baja nitidez o resolución (low-resolution). Cuando se introduce el siguiente comando en su ordenador

```
PLOT 128,87
```

usted podrá apreciar uno de esos puntos en el centro de la pantalla.

Los puntos empleados para generar figuras gráficas de alta nitidez o resolución se denominan "pixels", una abreviación de las palabras inglesas "picture cells" o celdas de colores. Como en el caso de los carácteres de baja resolución, la posición de cada pixel es especificada mediante dos números.

La red de alta resolución o nitidez

La red de alta resolución consiste de 256 pixels distribuidos en hileras horizontales a lo largo de la pantalla y 176 distribuidos en columnas verticales. Sin embargo, y a diferencia de lo que sucede en el caso de las imágenes de baja resolución, el primer número o coordenada es utilizado para indicar la posición sobre una hilera horizontal en la pantalla. Estas posiciones varían entre **Ø** sobre el borde de la izquierda de la pantalla, hasta 255 sobre el borde de la derecha. El segundo numero es la posición sobre la columna vertical. En este último caso los números varían entre **Ø** (parte inferior de la pantalla) y 175 (parte superior de la pantalla). Por ejemplo, las coordenadas **Ø,Ø** corresponden al rincón inferior de la izquierda de la pantalla de su televisor (en el caso de imagenes de baja resolución esas mismas coordenadas corresponderían a la esquina superior de la izquierda). La página **88** contiene un plano de la red de alta resolución.

Ubicando puntos y dibujando sobre la pantalla

Solamente se requieren tres palabras claves para producir gráficas de alta resolución: PLOT, DRAW y CIRCLE. La palabra clave PLOT es seguida por las coordenadas horizontal y vertical del punto requerido separadas por una coma. Esta instrucción ubica un pixel precisamente en la posición definida por las coordenadas. La palabra clave DRAW también es seguida por dos números, separados por una coma, pero estos números no son coordenadas de una posición. Estos números son distancias medidas en pixels entre dos posiciones, ya sea en la dirección vertical u horizontal. El propósito de la instrucción DRAW es trazar una línea entre esas dos posiciones.

Si esta es la primera vez que se utilizan las palabras clave PLOT o DRAW en el programa la primera posición es **Ø,Ø**. Si estas palabras clave han sido utilizadas anteriormente en el mismo programa, entonces la primera posición (el punto de partida para la nueva línea), es la ultima posición PLOT o DRAW anterior, cualquiera sea la más reciente. En este caso, el ordenador ejecutará la instruccion DRAW, dibujando una línea desde aquel punto hasta la nueva posición. Veamos un ejemplo práctico:
La palabra clave PLOT le indica al ordenador que debe mover la posición del punto donde comienza la línea al tope de la pantalla. Luego las cinco sentencias DRAW dibujan las cinco líneas rojas.

ESTRELLA

```
10 INK 0
20 PLOT 128,174
30 DRAW 70,-140
40 DRAW -152,80
50 DRAW 164,0
60 DRAW -150,-60
70 DRAW 70,140
```



A continuación agregue las siguientes líneas de programa a su programa "Estrella"

```
4 BORDER 1: PAPER 6: INK 1: CLS
5 CIRCLE 128, 87, 87
```

Ejecute este programa, y verá como el ordenador dibuja una estrella roja en un círculo azul.

La instrucción CIRCLE requiere tres números. Los primeros dos números le indican al ordenador cual es



la posición del centro del círculo, y el tercer número es el radio de este. También es posible agregar un tercer número a las sentencias incluyendo la palabra clave DRAW.

Pruebe en su programa valores entre 2 y −2, para ver que sucede.

Como rellenar figuras

Crear figuras sólidas en alta resolucion es una operación sencilla, el procedimiento consiste en dibujar una cantidad de líneas paralelas una junta a la otra. Esto puede hacerse mediante un bucle (loop) conteniendo las palabras clave FOR NEXT, el cual cambiará la posición de DRAW, incrementándola en pasos de una unidad.

TRIANGULO SOLIDO

```
  10 BORDER 1: PAPER 6: INK 2: C
LS
  20 FOR x=-100 TO 100
  30 PLOT 128,150
  40 DRAW x,-120
  50 NEXT x
```



Es posible obtener un resultado interesante dibujando líneas con una pequeña separación. Es posible hacerlo agregando la palabra clave STEP y un número, a la sentencia FOR. Introduzca en el ordenador la nueva línea de programa **2Ø** y ejecute nuevamente el programa:

```
20 FOR x = − 100 TO 100 STEP 4
```

Ahora aparece una imagen un forma de abanico. Esto se debe a que cuando el ordenador dibuja las líneas, la palabra clave STEP increments x en pasos de 4 unidades, en lugar de pasos de 1 unidad como anteriormente.



Utilizando la pantalla de su televisión como una pizarra

No es necesario escribir un programa cada vez que desee producir un dibujo o un patrón. Tambien es posible utilizar un programa que le permitira dibujar directamente en la pantalla.

Este programa comienza utilizando la palabra clave INPUT, cuyo propósito es preguntarle cual es el número control para la tinta (ink number). A continuación, y utilizando nuevamente la palabra clave INPUT (esta vez con un signo **$**, para nombrar o "bautizar" la cadena), le indica al ordenador que dibuje líneas cortas cada vez que usted oprima una de cuatro teclas especificas. Estas son U,D,L, o R, seguidas en cada caso por la palabra clave ENTER. El programa utiliza las palabras claves IF y THEN para tomar decisiones.

PIZARRA

```
  10 INPUT "INK ";i
  20 BORDER 3: PAPER 0: INK i: C
LS
  30 PLOT 25,25
  40 LET x$=
  50 INPUT a$
  60 IF a$="U" THEN DRAW 0,x
  70 IF a$="d" THEN DRAW 0,-x
  80 IF a$="l" THEN DRAW x,0
  90 IF a$="r" THEN DRAW -x,0
 100 GO TO 50
```



Tomando decisiones con IF y THEN

En el Programa "Pizarra", las líneas **6Ø** a **9Ø** contienen sentencias con las claves IF y THEN. Estas permiten a su Spectrum tomar una decisión. En este caso el ordenador examinará si la tecla oprimida es alguna de las indicadas en el programa (u,d,l,r). Si cualquiera de estas teclas es oprimida, entonces el ordenador dibujará una línea. No dibujará la línea respectiva si la letra mayúscula es oprimida.

La clave IF es siempre seguida de una condición que el Spectrum comprueba si es verdadera o si se está cumpliendo. Si es verdadera ejecutará la acción indicada después de la clave THEN. Si no es verdadera el programa pasa a la línea siguiente.

Todas las instrucciones en la línea después de THEN dependen de la decisión del ordenador. Por ejemplo en la línea siguiente:

```
110 IF b=5 THEN PRINT "*":GOTO200
```

el ordenador solo irá a la línea **2ØØ** si b=5.

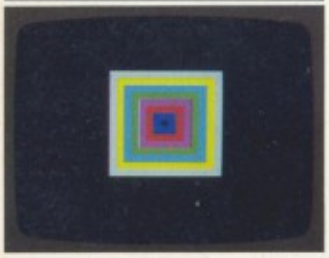
COMO DISENAR SUS PROPIOS PATRONES Y DIBUJOS

Es posible producir toda clase de patrones y dibujos con su ZX Spectrum +, mediante gráficas de baja resolución, gráficas de alta resolución o ambos métodos. El mejor procedimiento para dibujar con su ordenador es hacer previamente un borrador empleando copias de la cuadrícula en la página 80.

Es posible utilizar bucles FOR NEXT para dibujar. Estos bucles repiten una parte del programa total, la cantidad de veces indicada. El programa puede ser redactado de tal forma que cada uno de los bucles cambie las posiciones y colores de los carácteres o la línea, usualmente de acuerdo a un plan regular preestablecido. El siguiente programa hace uso de esta técnica.

CUADRADOS

```
10 BORDER 0: PAPER 0: CLS
20 FOR x=7 TO 0 STEP -1
30 INK x
40 FOR l=11-x TO 11+x
50 FOR c=16-x TO 16+x
60 PRINT AT l,c; ■
70 NEXT c
80 NEXT l
90 NEXT x
```



El programa contiene tres bucles (o en inglés "loops") FOR ... NEXT. El bucle x cambia el color y también el tamaño de los cuadrados grandes producidos por el programa, mientras que el bucle l y el bucle c cambian la posición de la línea y la columna de los cuadrados pequeños cada vez que son visualizados en la pantalla.

Efectos al azar o aleatorios y subrutinas

Los bucles no le limitan a que las figuras sean idénticas cada vez que el programa es ejecutado. Al incluir la palabra clave RND (una abreviación de RaNDom) en el bucle, es posible cambiar los colores, posiciones de los elementos del dibujo o demás características, cada vez que el programa es ejecutado. Examine el programa para el mosaico en la página 10. El programa contiene la instrucción RND*7 que controla el color de la tinta seleccionando cualquier número con un punto decimal entre 0 y 7. La palabra clave INK cambia ese número al número entero más próximo. El resultado es que, cada vez que el ordenador visualize un cuadrado, su color variará entre INK 0 y INK 7.

El siguiente programa dibuja figuras simétricas formadas por carácteres gráficos en la pantalla. El programa emplea la palabra clave RND para cambiar esos carácteres y sus posiciones. Las variables i y p determinan el color de la tinta y el papel. La variable a determina la cantidad de figuras que serán deibujadas por el ordenador (en este caso cuatro figuras). La variable n establece la cantidad de carácteres en cada figura, mientras que la variable x es un número seleccionado al azar por el ordenador entre 129 y 142. La sentencia GOSUB 1000 en la línea 50 dirige el ordenador a una subrutina , que le indica que al llegar a esta línea del programa debe dirigirse a una subrutina, definida en la línea 1000 y siguientes del programa.

FIGURAS SIMÉTRICAS

```
 10 BORDER 4: PAPER 0: CLS
 20 LET i=4: LET p=0
 30 FOR a=1 TO 4
 40 LET x=RND+13+129
 50 FOR n=1 TO 40: GO SUB 1000:
NEXT n
 60 LET i=i+1: LET p=p+1
 70 PAUSE 100
 80 NEXT a
 90 STOP
1000 LET l=INT (RND+11)
1010 LET c=INT (RND+16)
1020 INK i: PAPER p
1030 PRINT AT l,c;CHR$ x
1040 PRINT AT l,31-c;CHR$ x
1050 PRINT AT 21-l,c;CHR$ x
1060 PRINT AT 21-l,31-c;CHR$ x
1070 BEEP 0.01,i*3
1080 RETURN
```



Una subrutina es un grupo de líneas de programa que actuan como un pequeño programa incluído dentro del programa principal. En nuestro ejemplo, la subrutina se encuentra en la línea 1000. La subrutina visualiza un carácter gráfico en cuatro puntos de la pantalla, de forma tal que cada uno de ellos se encuentra a la misma distancia del centro (posición 11,16). La distancia desde el centro es determinada por las líneas de programa 1000 y 1010, la variable l indica la distancia en hileras, y la variable c la distancia en columnas. La palabra clave INT modifica el número seleccionado al azar, transformándolo en cualquier número entero entre 0 y 10 para la variable l , y en cualquier número entero entre 0 y 15 para la variable c. A continuación, las líneas de programa 1030 – 1060 visualizan el carácter gráfico cuyo código es x (consulte la tabla de carácteres en la página 51). La palabra clave BEEP genera un sonido cuya intensidad está relacionada con la posición. Finalmente, la palabra clave RETURN, en la línea de programa 1080 cierra la subrutina y envía nuevamente al ordenador a la instrucción inmediatamente después de la palabra GOSUB, en la línea 50.

La línea 60 varía los colores de la tinta y papel. La palabra clave PAUSE 100 (pausa) en la línea 70 demora la ejecución del programa por un período de dos segundos, antes de que el ordenador reinicie el bucle. La palabra clave STOP (alto) en la línea 90 es una advertencia para el ordenador, que le advierte que no debe continuar directamente a la subrutina después del cuarto bucle.

Es posible modificar el programa, alterando el número 4 en la línea 30, y el número 50 en la línea 50. si aumenta la amplitud de x (línea del programa 40) se cambiarán los carácteres que aparecen en la pantalla.

Empleando los bucles de FOR NEXT para efectos gráficos

Los bucles FOR NEXT pueden ser utilizados con gran eficacia para producir efectos gráficos de alta resolución. Introduzca mediante el teclado el programa siguiente y ejecútelo. Utilizando solamente las palabras clave PLOT y DRAW, los dos bucles FOR NEXT, primero dibujan líneas y luego cinco triángulos o pirámides solidas.

PIRAMIDES

```
 10 BORDER 0: PAPER 1: INK 6
 20 CLS
 30 FOR v=0 TO 20 STEP 2
 40 PLOT 0,v
 50 DRAW 255,0
 60 NEXT v
 70 FOR n=100 TO 220 STEP 30
 80 FOR x=-10-n/10 TO 10+n/10
 90 PLOT n,35+n/10
100 DRAW x,-n/4
110 NEXT x: NEXT n
```



Ahora agregue las líneas en la columna siguiente.Cuando ejecute nuevamente el programa encontrará que un rayo laser dispara en el cielo nocturno, creando una multitud de estrellas. El rayo es dibujado a partir de la esquina de la pantalla, a la posición x, y. Estas dos últimas variables (x, y) son números elegidos al azar por su ordenador.

Tome nota del empleo de la palabra clave OVER 1;


```
120 LET x=RND*255
130 LET y=RND*104+71
140 LET i=INT (175-y)/8
150 LET c=INT (x/8)
160 PLOT 0,0: DRAU OVER 1;x,y
170 BEEP 0.01,x/4
180 PLOT 0,0: DRAU OVER 1;x,y
190 PRINT AT i,c;" "
200 GO TO 120
```



(incluyendo el ;) en las líneas 160 y 180, que son idénticas. OVER 1 permite a la primera línea dibujar el rato del laser, y la segunda línea lo borra, "sin" modificar el resto de la figura.

Las palabras clave FLASH, BRIGHT e INVERSE

Estas tres palabras clave son muy útiles para controlar los colores de su Spectrum. Cada una de ellas es seguida por los números 0 o 1. Es posible incluirlas en las sentencias conteniendo la palabra clave PRINT, siempre y cuando se agregue un punto y coma (;) después del 0 o el 1. FLASH 1 hace relampagear las posiciones de los carácteres entre los colores de la tinta y el papel. BRIGHT 1 hace que los colores sean más brillantes. INVERSE 1 cambia el color de la tinta al del papel y viceversa. Cuando se introduce una de estas tres palabras clave seguida del número 0, se causa el efecto opuesto: la pantalla vuelve a la situación normal. (El número 1 despues de la palabra clave inicia el efecto especial y el número 0 lo termina).

Pruebe introducir los cambios siguientes en los programas en estas dos páginas, para aprender a utilizar las palabras clave. En el pograma de los cuadrados, modifique el cuadrado en la línea 60 transformándolo en una estrella y a continuación agregue el comando

15 **INVERSE 1**

Como resultado las estrellas aparecerán en color negro (el color del papel), resaltando contra las franjas de colores (los colores cambiantes de la tinta). Introduzca el comando INVERSE 0 antes de continuar.

En el programa de las Figuras Simétricas, agregue las líneas a continuación, para apreciar el efectos de las palabras clave BRIGHT y FLASH.

15 **BRIGHT 1**
16 **FLASH 1**

Observe como FLASH hace "relampaguear" las figuras, lo que produce el efecto de que estas parezcan moverse en la pantalla. Introduzca la sentencia **FLASH 0:CLS** para interrumpir el efecto de "relampagueo" de las figuras en la pantalla.

El PROCEDIMIENTO PARA CREAR CARACTERES DE ORDENADOR

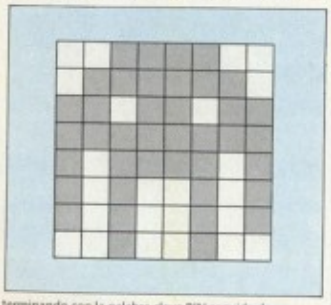
El ZX Spectrum + no se encuentra limitado a los carácteres gráficos que se encuentran indicados en el teclado del ordenador. Este cuenta con una sección especial en su memoria que puede ser utilizada para almacenar carácteres definidos por usted. Estos son denominados "carácteres gráficos definidos por el usuario" (del inglés: "user defined graphics"), y cada programa puede tener una cantidad máxima de 21 de ellos.

Cada uno de los carácteres gráficos definidos por el usuario consiste de 64 puntos minúsculos, llamados pixels, de color de tinta. Los puntos son ordenados en hileras de ocho pixels cada una. Cada carácter ocupa la posición de uno de los carácteres en la cuadrícula de baja resolución.

Diseñando un carácter gráfico

El primer paso es dibujar una cuadrícula de 8x8 como se ilustra más abajo. A continuación llene algunos de los cuadrado representa un pixel pintado con colores de tinta (ink). Cada cuadrado lleno vale 1 y cada cuadrado vacío vale Ø. Por ejemplo, a el siguiente es un diseño para una araña.

Cada uno de los carácteres gráficos definidos por el usuario es identificado por una letra entre la letra a y la letra u (mayúsculas o minúsculas, es lo mismo). Para programar el carácter, usted debe introducir desde el teclado ocho sentencias POKE USR, cada una de ellas



terminando con la palabra clave BIN seguida de un número binario. Este número binario consiste de ocho grupos de números Ø o 1, y cada grupo corresponde a una línea de su cuadrícula. Por ejemplo, la figura de la araña en la cuadrícula será traducida a números binarios de la siguiente forma:

```
10 POKE USR "s",BIN 00111100
20 POKE USR "s"+1,BIN 01111110
30 POKE USR "s"+2,BIN 11011011
40 POKE USR "s"+3,BIN 11111111
50 POKE USR "s"+4,BIN 10111101
60 POKE USR "s"+5,BIN 10100101
70 POKE USR "s"+6,BIN 00100100
```

Ahora ejecute este programa en su ordenador, y esta vez oprima las teclas GRAPH y S. Sorpresa! En lugar de una letra S aparece la araña. A continuación agregue las líneas siguientes, con las instrucciones para la araña en la línea 3Ø del programa, y ejecute el programa. Las arañas se multiplican y cubren la pantalla.

ARAÑAS

```
20 BORDER 1: PAPER 0: CLS
30 PRINT INK RND*7;"s";
40 GO TO 30
```



Al disenar sus carácteres recuerde que no podrá verlos en la pantalla hasta que ejecute el programa donde se encuentran definidos. Hasta entonces, solamente podra verlos en los listados del programa, donde solamente figuran como letras.

Como mezclar colores

Es muy fácil simular combinaciones de colores en su ordenador Spectrum. Todo lo que necesita es crear un carácter que genere un cuadrado en la pantalla, conteniendo una mitad de color de tinta y la otra mitad de color de papel.

```
10 FOR x=0 TO 6 STEP 2
20 POKE USR "a"+x,BIN 10101010
30 POKE USR "a"+x+1,BIN 01010101
40 NEXT x
```

Es procedimiento es sencillo: defina dos líneas de pixels y a continuación indique al ordenador que utilice cada una de ellas alternadamente en un carácter. Al ejecutar el programa debe visualizarse un cuadrado con puntos. Si emplea la misma técnica en un programa con palabras clave para colores, el color del cuadrado será una mezcla de los colores del papel y la tinta en el programa.

Simplificando las gráficas con las palabras clave READ y DATA

Su Spectrum dispone du un procedimiento más sencillo para crear sus propias gráficas, y consiste en emplear números decimales conjuntamente con las palabras clave READ y DATA. El primer paso es transformar los números binarios en números decimales. Puede hacerlo introduciendo por el teclado de su ordenador el comando PRINT BIN seguido por el número binario, por ejemplo:

```
PRINT BIN 00111100
```

En respuesta su Spectrum visualizará el número decimal 6Ø, que es el equivalente del binario 00111100. En el ejemplo de nuestra araña, los ocho números decimales son: 6Ø, 126, 219, 255, 189, 165, 165 y 36.

Ahora puede utilizar las palabras clave READ y DATA. Ambas le suministran un procedimiento sencillo para alimentar cantidades de valores, como números, en las variables del programa. READ es seguido por una variable – una o más letras si está trabajando con números, o una sola letra seguida del símbolo $ cvando se trata de cadenas (string). Cuando su Spectrum encuentra una palabra clave READ, inmediatamente se dirige a la primera sentencia con la palabra clave DATA en el programa. Estas sentencias contienen una lista de valores separados por comas. El procedimiento seguido en este caso es el siguiente: – el ordenador toma el primer valor en esta lista, – retorna a la línea del programa donde había encontrado la palabra clave READ – y pone ese valor en la primera variable que encuentra después de la palabra clave READ.

El ordenador continua ejecutando el programa, y la próxima vez que encuentre la palabra clave READ, repetirá el procedimiento, pero esta vez tomará el segundo valor en la lista, y así sucesivamente.

```
10 FOR x=0 TO 7
20 READ v
30 POKE USR "s"+x,v
40 NEXT x
50 DATA 60,126,219,255,189,165
,165,36
```

El siguiente es el nuevo programa para producir la araña.

Este programa puede almacenar cualquier grupo de ocho números decimales en la memoria del ordenador para crear un carácter. Bastará para ello, con cambiar la letra s en la línea 3Ø por la letra que usted prefiera, e introducir ocho números en la línea 5Ø, despues de la palabra clave DATA, separados entre si por comas. Acto seguido oprima la tecla GRAPH y la letra elegida para obtener el carácter gráfico deseado al ejecutar el programa.

Dibujando un tablero de ajedrez

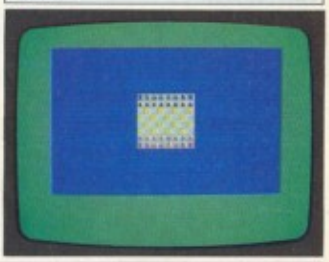
El programa visualiza un tablero de ajedrez en la pantalla, y a continuación dibuja las piezas. Para seleccionar los colores consulte la tabla de códigos de control en esta página.

TABLERO DE AJEDREZ

```
10 FOR x=1 TO 6
20 READ a$
30 GO SUB 500
40 NEXT x
50 BORDER 4: PAPER 1: CLS
60 FOR l=7 TO 14 STEP 2: FOR c
=16 TO 19 STEP 2
70 PRINT AT l,c;" ";AT l+1,c;
" "
80 NEXT c: NEXT l
90 PRINT AT 7,12;"RNBQKBNR"
100 PRINT AT 8,12;"PPPPPPPP"
110 PRINT AT 15,12;"pppppppp"
120 PRINT AT 16,12;"rnbqkbnr"
130 GO TO 130
500 FOR n=0 TO 7
510 READ v
520 POKE USR a$+n,v
530 NEXT n
540 RETURN
```

Las piezas son definidas por una subrutina que comeinza en la línea 5ØØ.

```
550 DATA "P",0,0,16,56,56,16,12
4,0
560 DATA "r",0,84,124,56,56,124
,124,0
570 DATA "n",0,16,56,120,24,56,
124,0
580 DATA "b",0,16,40,68,108,56,
124,0
590 DATA "K",0,16,56,16,56,68,8
6,0
600 DATA "q",0,84,40,16,108,124
,124,0
```



Utilizando los códigos de control para los colores

También es posible emplear otro tipo de instrucciones para los colores, en lugar de utilizar palabras clave como INK y PAPER. El otro método es agregar los códigos de control para los colores en sentencias conteniendo la palabra clave PRINT, después de la primera comilla. Los carácteres gráficos a ser visualizados cambiarán de color en el listado mismo y también aparecerán con esos mismos colores en la pantalla, al ejecutarse el programa.

Para obtener estos códigos, primero oprima la tecla EXTEND MODE, y luego una de las teclas-número con o sin CAPS SHIFT. La tabla le muestra como seleccionar cualquiera de los colores:

PRESS	1	2	3	4	5	6	7	8	9	Ø
EXTEND MODE + CAPS SHIFT	BLUE INK	RED INK	MAGNTA INK	GREEN INK	CYAN INK	YELLW INK	WHITE INK	FLASH OFF	FLASH ON	BLACK INK
EXTEND MODE	BLUE PAPER	RED PAPER	MAGNTA PAPER	GREEN PAPER	CYAN PAPER	YELLW PAPER	WHITE PAPER	BRIGHT OFF	BRIGHT ON	BLACK PAPER

DIBUJOS ANIMADOS

Los elementos gráficos del ordenador son ideales para producir carácteres o líneas con movimiento en la pantalla. Producir estos dibujos animados con su Spectrum no es difícil. Todo lo que se requiere es cambiar continuamente la posición de la pantalla donde el ordenador visualiza los carácteres o dibuja la línea. El mejor procedimiento es utilizar uno o más bucles FOR NEXT.

Movimientos verticales y horizontales

Introduzca en el ordenador el programa siguiente. Si su Spectrum todavía conserva almacenado en su memoria el programa para producir la araña, descripto en la sección anterior (página 32), entonces no será necesario volver a introducir las líneas 10–50. El carácter gráfico generado por el usuario en aquella oportunidad todavía se encontrará disponible y podrá ser llamado mediante la letra "s".

LA ARAÑA SALTADORA

```
   5 BORDER 3: PAPER 5: CLS
  10 FOR x=0 TO 7
  20 READ y
  30 POKE USR "s"+x,y
  40 NEXT x
  50 DATA 60,126,219,255,189,165
,165,36
  60 FOR x=0 TO 7
  70 READ y
  80 POKE USR "t"+x,y
  90 NEXT x
 100 DATA 16,16,16,16,16,16,16,1
6
 110 FOR i=0 TO 20
 120 PRINT AT i,3; INK 0;"t"
 130 PRINT AT i+1,3; INK 2;"s"
 140 NEXT i
```

En este programa la araña desciende a lo largo de la pantalla colgada de su hilo, cada vez que se ejecuta el programa.

Las líneas 60–100 del programa producen otro carácter gráfico ("t"), para el hilo. La animación se produce entre las líneas 110 y 140. Estas construyen un bucle FOR NEXT, en el cual la variable *i* (el número de la línea) cambia de 0 a 20. Cada vez que el bucle se repite, el ordenador dibuja una parte del hilo de la araña en la pantalla, en una posición determinada, y la araña es dibujada precisamente en la posición inferior siguiente. En el próximo bucle, la araña anterior es sustituida por un sector de hilo y el ordenador dibuja una nueva araña un poco más abajo. Mediante estos bucles que se repiten rápidamente, se produce la ilusión de que la araña baja colgada de su hilo, hasta que llega a la línea 21, que es la parte inferior del área de visualización.

Para disminuir la velocidad de la acción, introduzca la línea siguiente:

135 PAUSE 10

El efecto de esta sentencia es causar que el ordenador espere un quinto de un segundo cada vez, antes de dibujar la araña en la posición siguiente.

Ahora le sugerimos agregar las siguientes líneas extra al programa anterior, para producir una animación en otra dirección.

LA ARAÑA VELOZ

```
150 FOR c=2 TO 30
160 PRINT AT 21,c-1; "
170 PRINT AT 21,c+1; INK 2;"s"
180 NEXT c
```



Esta vez, nuestra amiga comienza a correr hacia uno de los lados de la pantalla tan pronto como toca el suelo. El efecto se logra agregando un nuevo bucle FOR NEXT al programa. Este bucle controla la posición horizontal *c* de la figura medida en columnas. Observe que primero se visualiza en la pantalla un espacio y que la araña es dibujada en la posición correspondiente a la columna siguiente. El resultado es que la araña desaparece de una posición y aparece en la siguiente, lo que produce la ilusión de movimiento de derecha a izquierda. Siempre es mejor borrar un carácter antes de volverlo a visualizar en la posición siguiente, para evitar que las figuras animadas parezcan oscilar en la pantalla.

Tiro al blanco

En muchos juegos de ordenador, uno de los elementos más importantes de la acción es el choque entre dos figuras, o cuando un objeto es tocado por un rayo, por ejemplo un rayo de laser.

Es fácil detectar el punto de colisión. Si dos carácteres son visualizados por el ordenador en la posicion *l,c* (por línea y columna) y en la posición *v,h* (por vertical y horizontal) respectivamente, entonces es posible asumir que si *l=v* y *c=h* entonces ambos carácteres se encuentran en la misma posición. Esta identidad puede ser redactada como una sentencia en su programa de ordenador utilizando las palabras clave IF THEN (en este caso unidas por otra palabra clave: AND)

160 IF l=v AND c=h THEN PRINT "CRASH!"

(es decir: si la variable l es igual a la variable v, y si la variable c es igual a la variable h, entonces visualize lo siguiente en la pantalla del televisor: "CRASH!")

Otro procedimiento para determinar si dos figuras entran en colisión es utilizar colores. Remueva el programa de la araña mediante la palabra clave NEW. A continuación introduzca el programa completo detallado en la página 31, o cárguelo en el ordenador si ha tomado la precaución de almacenarlo en un casette. Ahora podemos perfeccionar este programa combinándolo con el programa de la araña (el programa de la araña continua en la memoria del ordenador, excepto naturalmente si usted oprimió el control de reajuste – reset – o apagó el ordenador).

En primer lugar, agregue las líneas siguientes, las cuales crean una gráfica representando una explosión denominada "e".

```
   5 FOR x=0 TO 7
   6 READ y
   7 POKE USR "e"+x,y
   8 NEXT x
   9 DATA 145,82,44,121,158,52,7
4,137
```

A continuación borre la línea 190 y agregue o cambie las líneas siguientes.

ARAÑAS Y PIRÁMIDES

```
 114 LET h=RND*31
 116 FOR v=0 TO 20
 117 PRINT AT v,h;" ";AT v+1,h;
INK 4;"h"
 200 NEXT v
 205 PRINT AT 21,h; FLASH 1; INK
 2; PAPER 5;"S"
 210 GO TO 114
```



Debido a los cambios que hemos realizado, el programa ya no genera estrellas, en cambio ahora tenemos voraces arañas que caen sobre las pirámides y el suelo consumiendolos rápidamente. Lo que ha sucedido es que hemos agregado un nuevo bucle FOR NEXT en el cual las variables v y h suministran la posición de la araña. La variable h es aleatoria o random, lo que causa que las arañas aparezcan en diferentes posiciones de la pantalla. Ahora agregue las líneas siguientes:

ARAÑAS EXPLOSIVAS

```
 190 IF ATTR (v+1,h)=14 THEN GO
TO 500
 500 PRINT AT v+1,h; FLASH 1; PA
PER 2;"E"
 510 PAUSE 100
 520 GO TO 114
```



Cuando el rayo laser hace impacto en una de las arañas, esta se torna amarilla. El procedimiento para lograr este efecto es el siguiente: cuando la línea producida por DRAW entra en la posición del cáracter representando la araña, se produce un cambio en el color de la tinta de este último, y pasa a ser del mismo color que la línea (amarillo en este caso). En la línea 190 del programa, ATTR detecta si la araña se torna amarilla. Cuando esto sucede la instrucción le indica al ordenador que debe dirigirse a la subrutina de la explosión, en la línea 500.

SIMULANDO LOS REBOTES DE UNA PELOTA

Muchos de los programas con efectos gráficos muestran figuras que rebotan contra los bordes de la pantalla. Este programa le explicará como lograr tal efecto. Las variables v y h operan de la misma manera que en el programa de las arañas explosivas, pero se agregan los valores +1 o −1 a las variables v y h para hacer que la pelota se mueva hacia arriba o abajo o hacia la derecha o izquierda. La sentencia SCREEN$ comprueba si existe un X en la posición v,h.

PELOTA REBOTANDO

```
  10 BORDER 1
  20 FOR z=1 TO 10
  30 LET h=INT (RND*25): LET v=I
NT (RND*21)
  40 PRINT INK 2; PAPER 6; FLASH
 1;AT v,h;"X"
  50 NEXT z
  60 LET x=1: LET y=1
  70 PRINT AT v,h;" "
  80 LET v=v+y: LET h=h+x
  90 IF h=0 OR h=31 THEN LET x=-
x: BEEP .2,24
 100 IF v=0 OR v=21 THEN LET y=-
y: BEEP .2,12
 110 IF SCREEN$ (v,h)="X" THEN P
RINT INK 1; PAPER 5;AT v,h;"!#!
!": STOP
 120 PRINT AT v,h;"O"
 130 PAUSE 2
 140 GO TO 70
```



Utilizando atributos

La palabra clave ATTR detecta los atributos o características de una posición determinada en la pantalla. Los atributos son los colores de la tinta o el papel, y si la posición se encuentra relampagueando (FLASH) o brillante (BRIGHT). En el programa de las arañas explosivas, la palabra clave ATTR asegura que cuando la araña cambia de color, tomándose amarilla, el ordenador la haga desaparecer. Su color de tinta ahora es amarillo (número 6). El color de papel es azul (número 1). Esto significa que el total de atributos de esa posición es en total 14 (6 por la tinta amarilla más 8 por el papel azul). La sección correspondiente en la "Guía de Referencia del Programador" le explicará como se llega a estos números.

UN INSTRUMENTO MUSICAL Y EFECTOS SONOROS

El ZX Spectrum + posee un sintetizador de sonidos capaz de producir una gran variedad de sonidos musicales y efectos sonoros, que pueden ser utilizados para dar vida a sus programas. El sintetizador es fácil de usar, y no requiere conocimientos musicales. El sintetizador produce una señal de sonido al altavoz interno de su Spectrum.

Programando sonidos

Para producir sonidos en su Spectrum, solamente se emplea una palabra clave: BEEP. Esta palabra clave es seguida por dos números o variables representando números. El primer número o variable le indica al ordenador cual es la duración (en segundos) del sonido, y el segundo le hace saber cuan agudo o grave debe ser el sonido. El tono es medido en semitonos. Los valores del tono son Ø para do central; 1 para do sostenido; −1 para si (bemol), etc.

Introduzca este programa en su ordenador y ejecútelo.

El Spectrum producirá toda la gama de sonidos de que es capaz, desde las notas más altas o

```
10 FOR p=69 TO -60 STEP -1
20 BEEP .1,p
30 PRINT AT 0,0;"  ";AT 0,0;p
40 NEXT p
```

agudas (69) hasta las más bajas o graves (−60). Como puede apreciar las notas más altas son casi inaudibles, las más bajas suenan como 'clicks'.

La tabla en esta página le indicará cuales son los valores de tono correspondientes a una amplia gama de notas. Con esta información es posible transformer música escrita en parte de un programa para su ordenador Spectrum.

Efectos sonoros

El ordenador Spectrum tiene la capacidad de producir na amplia variedad de efectos sonoros, usualmente mediante un bucle incluyendo la palabra clave BEEP que cambia rápidamente el valor de tono. Pruebe los programas siguientes. Observe que los valores determinando la duración son muy breves, como de la centésima parte de un segundo. Oprima la tecla BREAK para terminar los programas conteniendo los bucles.

BURBUJEO

```
10 LET p=INT (RND*60)-30
20 BEEP 0.05,p: BEEP 0.05,p+7
30 GO TO 10
```

Este programa interpreta un grupo de tres notas, repitiéndolas varias veces mientras varia los tonos al azar. La amplitud de los tonos es considerable, y se la puede modificar, simplemente alterando los valores en la linea 10 del programa.

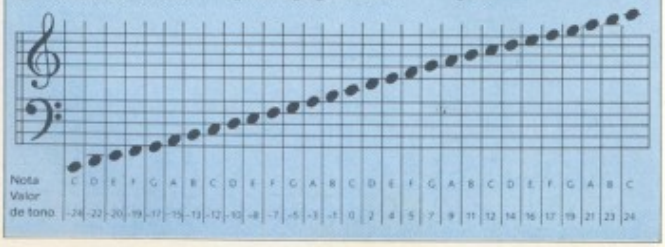
MÁQUINAS

```
10 FOR x=12 TO 36
20 BEEP .01,x
30 BEEP .01,24-x
40 NEXT x
50 GO TO 10
```

El programa produce dos sonidos, el tono de uno aumenta mientras que el del otro baja. Este efecto es el resultado de la combinación de dos sentencias en el programa con la palabra clave BEEP. Las sentencias correspondientes repiten dos notas de tonos diferentes con una separación de apenas un centésimo de segundo.

Valores de tono para efectos musicales

Los siguientes son los valores de los tonos para su Spectrum, desde los tonos más bajos hasta los más altos de los pentagramas graves y agudos. Agregar 1 al valor del tono para una nota aguda y restar 1 para una nota grave.



Nota	C	D	E	F	G	A	B	C	D	E	F	G	A	B	C	D	E	F	G	A	B	C
Valor de tono	-24	-22	-20	-19	-17	-15	-13	-12	-10	-8	-7	-5	-3	-1	0	2	4	5	7	9	11	12	14	16	17	19	21	23	24

ESCALA

```
10 FOR p=1 TO 48 STEP 0.2
20 BEEP .01,p: BEEP .01,p-6
30 NEXT p
```

Este programa es similar al programa "Máquinas" anterior, pero ahora las dos notas aumentan conjuntamente, con una separación de seis semitonos. Además los valores de tono cambian en pasos de Ø.2 – un quinto de un semitono – cada vez, lo que determina que el tono del sonido se incremente más lentamente. Pruebe cambiar la magnitud de los cambios de tono alterando el valor de la instrucción STEP (en inglés STEP significa "paso").

TECLADO MUSICAL

```
10 LET p=CODE INKEY$
20 IF p=0 THEN GO TO 10
30 BEEP .04,(p-38)/2
40 GO TO 10
```

Este programa adjudica un sonido diferente a cada una de las teclas en el teclado de su ordenador. Si se oprime la tecla CAPS SHIFT mientras se mantiene oprimida otra tecla, el sonido baja. El programa hace uso de la sentencia CODE INKEY$ que atribuye un valor diferente a la variable p cada vez que se oprime una nueva tecla.

Sonido y visión

Los efectos de sonido producidos por su Spectrum pueden ser combinados con la acción que se desarrolla en la pantalla de la televisión. Para demostrar como agregar sonido a un programa, examinar errors un ejemplo. Volvamos al programa de la "Araña Veloz" en la página 34.

Recuerde que insertamos una sentencia de PAUSE en la línea 135, para disminuir la velocidad de la acción. En lugar de reducir la velocidad del programa de esta forma, es posible programar una pausa que produce sonidos. Cambie la línea 135 del programa como se indica a continuación:

```
135 GOSUB 500
```

a continuación agregue las siguientes líneas al programa:

```
200 STOP
500 FOR p=40-l TO 38-l STEP -1
510 BEEP Ø.Ø2,p
520 NEXT p
530 RETURN
```

Ejecute el programa y verá como la araña hace un sonido especial cuando desciende. La nueva subrutina controlando el sonido le indica al ordenador que emita tres notas muy rápidamente, cuyo tono baja a medida que la araña desciende a la próxima posición en la pantalla. Pruebe agregar más notas, cambiando el contenido de la línea 500. También es posible aumentar o disminuir la velocidad de las notas, alterando el valor 0,02 en la línea 510.

Como amplificar los sonidos producidos por su Spectrum

Es posible conectar el enchufe de EAR o MIC a un amplificador o un altavoz para incrementar el volumen de los sonidos producidos por su Spectrum.

El procedimiento más sencillo es utilizar el cable del cassette del Spectrum para conectar el punto de conexión EAR o MIC al enchufe MIC de una grabadora en cassettes. Si fuera necesario extraiga el cassette de la grabadora. Ponga en marcha la grabadora de cassettes y oprima los controles PLAY, REWIND (REVERSE), o FAST FORWARD (CUE) (AVANCE RÁPIDO). Ajuste el control del volumen de la grabadora de cassettes. Ahora debe oirse el sonido producido por el ordenador, emitido por el altavoz de la grabadora. También es posible utilizar audífonos con la grabadora de cassettes.

Otra posibilidad es conectar su Spectrum a un equip de música de alta fidelidad, si desea obtener sonidos de alta calidad. En este caso, necesitará un cable especial con un enchufe macho de 3,5mm, para conectar su Spectrum y un enchufe para la conexión al equipo de música. El Spectrum produce una señal de linea similar a la señal de salida de las grabadoras de cassettes, por lo tanto debe ser posible conectar su ordenador a los puntos de conexión REPLAY o LINE IN del equipo de música.



COMO REGISTRAR SUS PROGRAMAS

Tarde o temprano encontrará que es conveniente almacenar sus programas en cintas de cassette. El primer paso es conectar una grabadora de cassettes a su Spectrum, para luego proceder a registrar (SAVE) el programa que se encuentra en la memoria del ordenador. La ventaja de registrar y almacenar los programas en cassettes es obvia: en el futuro cuando los vuelva a necesitar, podrá cargarlos directamente del cassette, utilizando el procedimiento descripto en ls páginas 14 y 15. El propósito de estas páginas es enseñarle a registrar programas y también a verificar que el programa ha sido grabado correctamente.

Registrando los propios programas

1 Primero conecte su Spectrum con una grabadora de cassettes de tipo adecuado, utilizando con tal propósito el cable especial, tal como se describe en la página 14. Compruebe que solamente el enchufe MIC del Spectrum se encuentra conectado a la grabadora de cassettes.

2 Si la grabadora de cassettes cuenta con controles de volumen y de volumen de la grabación, ajústeles a dos tercios del máximo. Si no cuenta con estos controles, no se preocupe, porque los niveles de la grabación se ajustarán automáticamente.

3 Introduzca por el teclado de su ordenador la palabra clave SAVE seguida por el nombre del programa que desea registar entre comillas, por ejemplo

SAVE "prog2"



Es posible utilizar cualquier combinación de hasta diez letras o números para designar un programa. A continuación introduzca la palabra clave ENTER. La línea conteniendo la palabra clave SAVE desaparecerá de la pantalla y en su lugar podra ver la instrucción de operación de la grabadora de su ordenador.



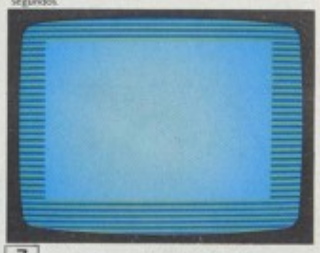
4 Ajuste los controles de la grabadora de cassettes para registrar o grabar el programa. El procedimiento usual es presionar simultáneamente las teclas RECORD y PLAY de la grabadora. A continuación oprima cualquier tecla en su Spectrum.

5 Espere mientras el Spectrum registra el programa. Al principio verá bandas de color rojo y azul ascendiendo lentamente en la pantalla.



Entonces el ordenador proyecta una serie de franjas azules y amarillas. Esto indica que el Spectrum se encuentra enviando el nombre del programa a la cinta de cassette.

6 Se produce un breve intervalo seguido otra vez por las franjas rojas y azules. Nuevamente aparecen las franjas amarillas y azules, cuando el Spectrum envía el programa a la cinta. La operación de grabar un programa externo puede tomar varios segundos.



7 Cuando el programa ha sido grabado en la cinta, su ordenador se lo hace saber mediante el reporte **0 OK,0:1**. Ahora detenga la cinta. El programa ha sido grabado, y si lo desea puede controlarlo o "verificarlo".

'Verificando' el programa

Aunque el ordenador ha transmitido el programa en su memoria a la cinta en la grabadora de cassettes, usted no tiene la seguridad de que aquel fuera grabado correctamente. Afortunadamente, nuestro Spectrum puede verificar la calidad de la grabación.

Este procedimiento se denomina "verificación". En primer lugar, vuelva a enrrollar la cinta hasta el punto donde comenzó la grabación (utilize el contador de revoluciones de su grabadora para este propósito). Luego se conecta el enchufe EAR del Spectrum al enchufe EAR de la grabadora de cassettes (no es necesario retirar la conexión entre los enchufes MIC). Una vez completada esta conexión, introduzca por el teclado la palabra clave VERIFY seguida por el nombre del programa entre comillas. Oprima la tecla ENTER y ponga en funcionamiento la grabadora. A continuación se repetirá la secuencia de las bandas de colores en la pantalla de la televisión: rojas y azules, y luego azules y amarillas. Aparecerá el nombre del programa en la pantalla y permanecerá allí hasta que se complete el proceso de verificación.



Al final de la segunda serie de franjas azules y amarillas el siguiente informe aparecerá en la pantalla:

0 OK,0:1

Este informe significa que el Spectrum ha comparado el programa registrado en la cinta con el programa almacenado en su memoria, y verificado que son idénticos.

Ahora puede tener la seguridad de que su programa ha sido grabado correctamente en el cassette.



Como almacenar sus programas en cintas de cassette

1. Escriba el nombre del programa en el rótulo del cassette o en su tarjeta en el momento de registrarlo. Utilize el mismo tipo de letras que aparecen en la pantalla de la televisión (mayúsculas o minúsculas). Si su grabadora de cassettes tiene un contador de revoluciones, tome nota de las revoluciones antes de iniciar la grabación y después, lo que facilitará la tarea de ubicar el programa en el futuro.

2. Antes de registrar el programa, anote el nombre del programa en una línea de programa al principio de éste, precedido por la palabra clave REM, por ejemplo:

5 **REM** programa LA ARAÑA VELOZ. Versión 3

Al ejecutar el programa, el ordenador ignorará todas las sentencias procedidas por la palabra clave REM. Esta palabra clave es muy útil porque le permitirá incluir comentarios y notas en su programa que le servirán como referencias en el futuro.

Si el ordenador no visualiza el reporte **0 OK,0:1**, esto significa que la grabación no ha sido correcta. En este caso, deben tomarse las medidas siguientes: En primer lugar consulte la sección de **Preguntas y respuestas sobre programas pre-registrados** en la página 16. Recuerde que siempre es posible que el programa fuera grabado correctamente, y que la dificultad se deba a que no es posible volver a cargarlo en el ordenador para la verificación. Examine el procedimiento de verificación y corrija cualquier error. A continuación vuelva a enrrollar la cinta y repita el procedimiento de verificación. Si el ordenador todovía no verifica el programa, entonces le sugerimos que consulte la sección sobre **Soluciones para los problemas que puedan presentarse al registrar programas** en la página siguiente.

– NOTA: No oprima la tecla NEW, no reajuste (reset), ni apague el ordenador! Si lo hace, perderá el programa almacenado en la memoria del ordenador, sin tener la seguridad de que cuenta con una copia correcta en la cinta de cassette.

Iniciando un programa automáticámente

Es posible agregar el nombre del programa (siempre entre comillas) a continuación de la palabra clave SAVE, seguido por la sentencia LINE 1, por ejemplo:

SAVE "ARAÑA VELOZ" **LINE** 1

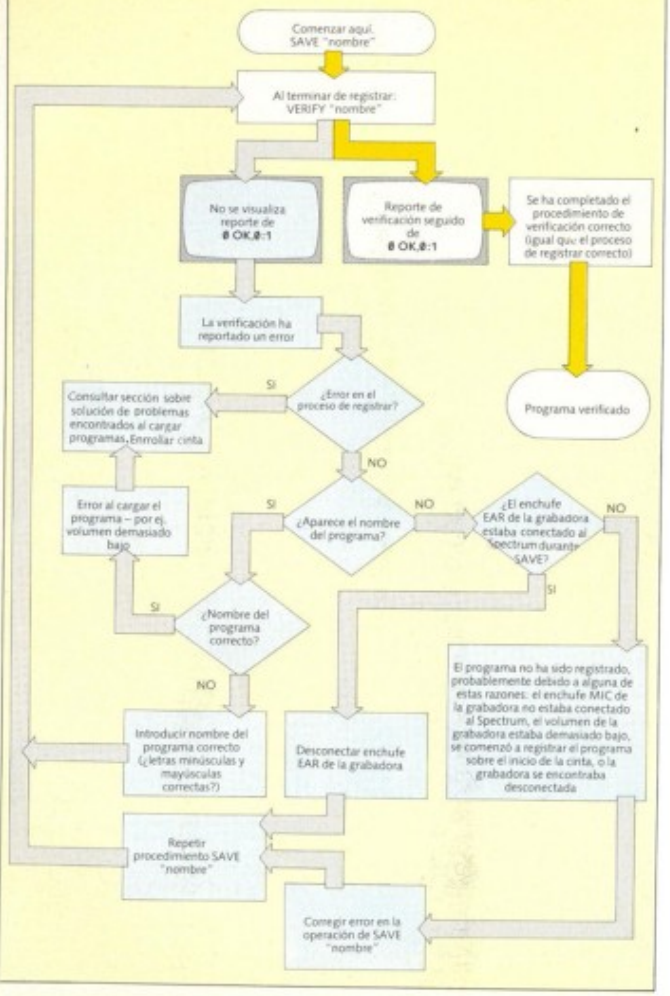
En este caso, el procedimiento para grabar el programa es similar al explicado anteriormente, pero al verificar la grabacion, no debe incluir la sentencia LINE 1 despues de VERIFY y el nombre del programa. La ventaja de agregar esta sentencia es que, en el futuro, al cargar el programa al ordenador, este lo ejecutará automaticamente, tan pronto como se de cargarlo. En este caso no sera necesario introducir la instrucción RUN para indicarle al ordenador que ejecute el programa. (Recuerde detener la grabadora una vez que el programa ha sido cargado al ordenador).

Lo que sucede en este caso es lo siguiente: el programa comienza en la línea 1, si el ordenador no encuentra una línea 1, entonces avanza a la primera línea del programa. Cuando se cambia el número 1 a otro número, el ordenador iniciará el programa automaticamente en la línea con ese número.

Registrando programas con las palabras clave CODE, SCREEN$ o DATA

La palabra clave SAVE también puede ser utilizada en programas conteniendo las palabras clave CODE o SCREEN$, para registrar una sección determinada de la memoria del ordenador, y conjuntamente con DATA para registrar una matriz (array). Consulte al respecto las secciones sobre SAVE CODE, SAVE SCREEN$ y SAVE DATA, en la **Guía de referencia del programador.**

Soluciones para problemas al registrar programas en cinta



APRENDIENDO ACERCA DE SU ZX SPECTRUM +

Este capítulo describirá los diversos componentes de su ZX Spectrum + y explicará su funcionamiento. También le explicaremos como utilizar equipos o accesorios "periféricos": estos son conectados al ordenador para incrementar su rendimiento y transformarlo en un completo sistema de procesamiento de datos. Finalmente, tendrá la oportunidad de aprender más acerca de los aspectos (técnicos de su ordenador, incluyendo la forma en que se organiza su memoria, y también le suministraremos información sobre las especificaciones técnicas del Spectrum.

3

COMPONENTES

Comenzaremos esta sección con un buen consejo: si desea aprender más acerca de su Spectrum lea este capítulo, pero *no abra* su ordenador! Recuerde que si lo abre invalidará su garantía, y, lo que es peor, corre el riesgo de dañar alguno de los componentes.

En el interior de caja se encuentran dos conectores de cinta que interconectan el teclado del ordenador Spectrum con los demás componentes. Estos últimos vienen instalados en ún único tablero de circuitos impresos. El tablero contiene elementos eléctricos de tipo estandar, tales como resistencias y condensadores, pero sus elementos más importantes son las microchips rectangulares de color negro, dispuestas independientemente o en bloques.

El interior de una chip

La parte funcional de una microchip es, en realidad, bastante más pequeña que la envoltura de plástico que la contiene. La envoltura de la chip es diseñada principalmente para albergar las conexiones requeridas por aquella. Estas conexiones permiten conectarla a los puntos de conexión ubicados sobre el panel de circuitos. La chip en si misma es una pequeña placa de silicio que contiene varios miles de bifurcaciones eléctricas. Cada una de las bifurcaciones eléctricas actúa como un conmutador que permite detener, pasar, o almacenar las señales que llegan a él. El procedimiento es muy sencillo, pero existen tantas bifurcaciones operando conjuntamente en la misma chip, que el sistema como un todo, puede producir señales complejas que almacenan o procesan información a una velocidad y precision enormes.

¿Cómo se conectan las chips con el resto del sistema?

Por lo tanto, en general, podemos decir que el Spectrum es un circuito eléctrico de gran complejidad. Cuando el ordenador se encuentra en funcionamiento, un flujo continuo de pulsos eléctricos es transmitido a través de los senderos en el interior de cada chip, y entre estas y los demás componentes del sistema, para hacer funcionar el ordenador.

Pero, ¿cómo es que se controlan estos procesos, para asegurar que la señal correcta llega al punto indicado, cuando se la necesita? Escondido en el interior de una de las chips se encuentra el reloj del ordenador. El tic tac de este reloj es el resultado de una serie de pulsos de electricidad constantes.

El interior de su Spectrum
En esta vista del interior de su ordenador se han retirado las dos cintas conectoras del teclado.

Uncommitted Logic Array (ULA)
Esta chip genera las imagenes en la pantalla del televisor, partiendo de información conservada en RAM y también opera como control del sistema.

Punto de conexión del teclado
Uno de los conectores de cinta al tablero es conectado en este punto.

Cuando el Spectrum se encuentra operando, al oprimirse una tecla se ponen en contacto un par de cables situados debajo del teclado. Esto genera una señal código que es enviada a la Unidad Procesadora Central (en inglés, Central Processing Unit o CPU).

Salida a la Tv
Produce la señal que es transmitida a la televisión.

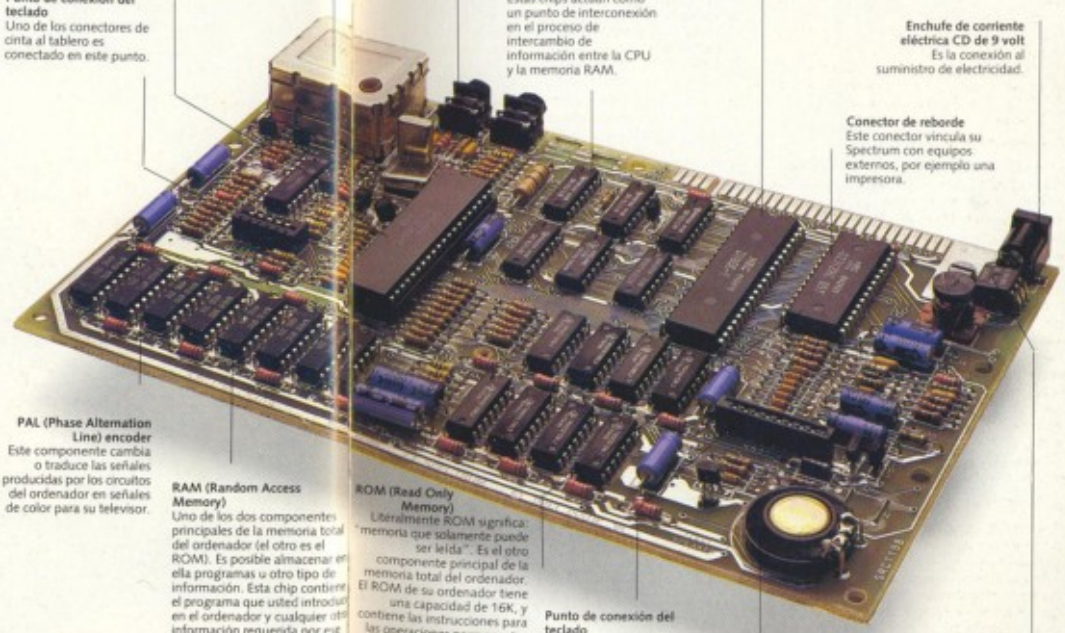
Enchufes para la grabadora de cassettes
Son empleados para transmitir información y programas desde la grabadora a la memoria RAM del ordenador, o para transmitirlos desde la memoria a la cinta en el cassette para almacenarlos.

Central Processing Unit o CPU (Unidad Procesadora Central)
Este es el cerebro de su ordenador. El CPU del Spectrum es un micro-procesador Z80. La unidad CPU lleva a cabo todas las tareas de computación, cálculos y controla la operación total del Spectrum.

Chips lógicas
Estas chips actúan como un punto de interconexión en el proceso de intercambio de información entre la CPU y la memoria RAM.

Enchufe de corriente eléctrica CD de 9 volt
Es la conexión al suministro de electricidad.

Conector de reborde
Este conector vincula su Spectrum con equipos externos, por ejemplo una impresora.



PAL (Phase Alternation Line) encoder
Este componente cambia o traduce las señales producidas por los circuitos del ordenador en señales de color para su televisor.

RAM (Random Access Memory)
Uno de los dos componentes principales de la memoria total del ordenador (el otro es el ROM). Es posible almacenar en ella programas u otro tipo de información. Esta chip contiene el programa que usted introduce en el ordenador y cualquier otra información requerida por ese programa, por ejemplo valores para sus variables. La capacidad de memoria RAM de su Spectrum es 48K, y todo su contenido puede ser modificado desde el teclado del ordenador. También es posible borrar el contenido de la memoria RAM mediante el control de reajuste (reset) o apagando el ordenador.

ROM (Read Only Memory)
Literalmente ROM significa "memoria que solamente puede ser leida". Es el otro componente principal de la memoria total del ordenador. El ROM de su ordenador tiene una capacidad de 16K, y contiene las instrucciones para las operaciones permanentes requeridas por la CPU. Entre otras tareas traduce las instrucciones en BASIC de los programas a un lenguaje que puede ser comprendido por la CPU. El contenido de esta memoria ROM no puede ser modificado o alterado desde el teclado, a diferencia de lo que sucede con la memoria RAM.

Punto de conexión del teclado
Aqui se conecta uno de los conectores de cinta del tablero.

Altavoz
Este componente produce los sonidos.

Regulador de voltaje
Controla el voltaje para evitar cualquier cambio en la electricidad que pueda afectar al ordenador.

¿CÓMO FUNCIONA SU ZX SPECTRUM +?

Desde el punto de vista de su operación, el ordenador ZX Spectrum + se compone de cuatro partes principales: unidades de introducción de datos, como el teclado, las mediante las cuales se introduce en el ordenador la información necesaria, memorias temporarias y permanentes que almacenan la información, incluyendo programas, e instrucciones de operación; la Unidad Procesadora Central (CPU) que ejecuta las instrucciones en el programa; y las unidades de salida o output, al usuario.

Introduciendo y ejecutando un programa

¿Qué sucede en el interior de su Spectrum cuando usted introduce y ejecuta un programa? Por ejemplo, un programa de una sola línea como el siguiente:

10 PRINT 6+2

Primero, usted ha operado el teclado. Debajo del teclado se encuentra una red de cables. Cada vez una tecla es oprimida, un par de estos cables entra en contacto, lo que produce una señal que es enviada a la CPU. La CPU transmite el código a la memoria RAM, donde la señal es almacenada.

Al ejecutar el programa, la CPU recibe los códigos almacenados en la memoria RAM uno a uno, en el mismo orden que en el programa.

En nuestro ejemplo, la CPU recibirá en primer lugar el código para la palabra clave PRINT. Es un código particular que le indica a la CPU que debe tomar un código de operación especial guardado en su memoria ROM. Este código de operación es transmitido a la CPU, la cual se prepara para ejecutar las instrucciones. En este ejemplo la instruccion es sencilla: escribir algo sobre la pantalla de la televisión. A continuación la SPU recibe el valor 6 contenido en la memoria RAM. Este valor también ha sido traducido al código. La CPU lo recibe y lo guarda en una pequeña memoria interna denominada "registro". La señal siguiente es el código para la suma. la CPU toma el valor 2 de la memoria RAM como en el caso del número 6. El valor 2 también ha sido traducido al código comprendido por la CPY y es sumado al valor 6 que se encuentra almacenado en el registro. El resultado es 8. La CPU traduce este resultado en otro código y finalmente, el número 8 es visualizado en la pantalla.

Almacenando un programa

Cuando se le indica a su procesador que almacene el programa en una cinta de cassette, la CPU nuevamente toma los códigos de la memoria RAM. Pero en lugar de ejecutar las instrucciones, las envía a una unidad convertidora que los transforma en señales de sonido. Estas son las señales que seran transmitidas a la grabadora de cassettes y registradas en la cinta.

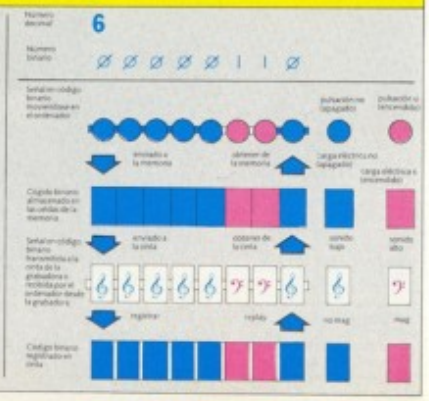
Al cargarse el programa en el ordenador, se produce el proceso inverso: las señales de sonido son transformadas en códigos de ordenador por el convertidor.

Códigos binarios

Todos los códigos que hacen funcionar su Spectrum son binarios. Se denominan binarios porque están compuestos por solamente dos tipos de señales. Estas señales pueden ser representadas por números binarios, esto es números que solamente contienen dos numerales: Ø y 1. Por ejemplo, el número binario para 6 es ØØØØ11Ø.

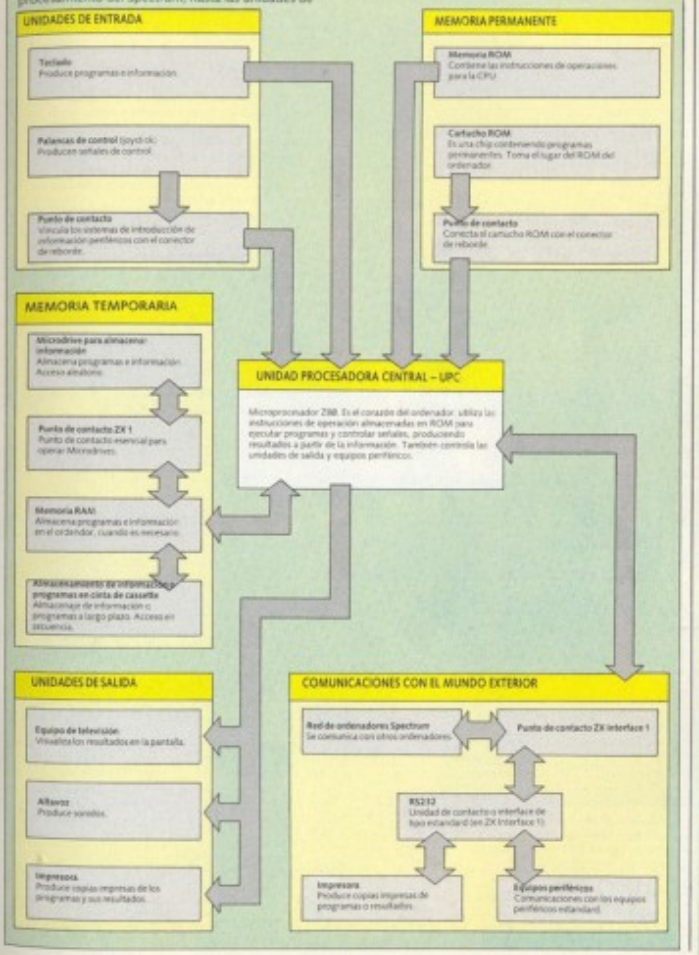
En el interior de su Spectrum, los códigos consisten en rápidas secuencias de pulsaciones de electricidad. Cuando una de estas pulsaciones llega a cualquier punto en el sistema, esta breve ráfaga de electricidad equivale a 1 en binario. Si no se produce una pulsación en un punto específico, en un momento determinado, esto equivale a Ø en binario. Por lo tanto, en el código empleado por el ordenador el valor 6 es equivalente a: *¿pulsación? si-si-si-si-no-no-si.*

En el diagrama adjunto es posible apreciar como el ordenador utiliza diversos tipos de información traducida al código binario para mover información de un punto del sistema al otro.



Los senderos de entrada – procesamiento en el ordenador salida de su Spectrum

El diagrama muestra como la información codificada es procesada en el ordenador. Es posible apreciar las diferentes etapas, desde la introducción de información en el ordenador, por ejemplo desde el teclado, su pasaje a través de los sistemas de procesamiento del Spectrum, hasta las unidades de salida, por ejemplo la pantalla de la televisión. Las flechas en un solo sentido indican los senderos que operan solamente en una dirección. Las flechas de dos puntas señalan los senderos que pueden operar en dos direcciones.



EL PROCEDIMIENTO PARA CONECTAR EQUIPOS ACCESORIOS

Es posible complementar su ZX Spectrum + con una serie de equipos accesorios que incrementarán su rendimiento, transformándolo en un poderoso sistema de procesamiento de datos. Existen disponibles una amplia variedad de equipos periféricos Sinclair y de otros tipos, compatibles con su Spectrum. El componente central es el punto de contacto ZX Interface 1, que permite conectar Microdrives para el manejo rápido y sencillo de programas e información. Este punto de contacto también sirve para conectar una amplia gama de equipos periféricos, incluyendo otros Spectrums. Mediante este punto de contacto, es posible conectar su Spectrum con impresoras de tipo estandard, o con un modem (modulador-demodulador) para transmitir información a grandes distancias. También existen disponibles otros tipos de puntos de contacto o interfaces, que conectan cartuchos ROM enchufables al ordenador, y que sirven para cargar programas en forma instantánea. Con estos elementos, es posible instalar bastones de mando (joysticks en inglés) para sus juegos.

Impresoras compatibles con su Spectrum

Algunas impresoras pueden ser conectadas directamente a su ordenador mediante el conector de reborde de su Spectrum. Por ejemplo, si ya posee una impresora Sinclair ZX, la puede conectar a su ordenador sin necesidad de agregar un punto de contacto (interface). Sin embargo, si desea hacer uso de impresoras requiriendo una salida del tipo RS232, entonces debe hacer uso del enchufe D en el punto de contacto ZX Interface 1.

El punto de contacto ZX Interface 1

La unidad de punto de contacto ZX Interface 1, se conecta a la parte posterior y la base de su Spectrum. Su propósito es vincular su Spectrum con una serie de equipos periféricos. Estos pueden incluir un máximo de hasta ocho Microdrives, otros 63 ordenadores Spectrum, o, mediante su unidad de punto de contacto RS232 estandard, una amplia variedad de equipos periféricos estandard.

Las microdrives y los cartuchos Microdrive pueden reemplazar la grabadora de cassettes y las cintas para almacenar información. Cuando se emplean los cartuchos Microdrive, es posible registrar, verificar y cargar programas en cuestión de segundos. Cada cartucho tiene suficiente capacidad como para almacenar 85K de información y utilizando el máximo de ocho Microdrives, su Spectrum tendrá una capacidad de almacenamiento en línea de 680K! Cualquier programa será ubicado automáticamente, con un tiempo de acceso típico de apenas 3,5 segundos.



Cargando una Microdrive
Los cartuchos Microdrive son insertados en la ranura situada al frente de la Microdrive.



Instalación de un cartucho ROM
El cartucho ROM es insertado en el punto de contacto. Al conectarse el ordenador, el programa en el cartucho ROM es cargado automáticamente en el Spectrum, salteándose el ROM en el interior del ordenador.

Conectando los equipos periféricos
El punto de conexión ZX Interface 1 es enchufado permanentemente en el conector de reborde del ordenador, y se encuentra situado detrás y debajo de éste. La ilustración muestra el sistema antes de conectar el ordenador.



Unidades Microdrive
Es posible conectar hasta ocho de estas unidades a su Spectrum.

Cable de cinta
Conecta la Microdrive con el ordenador, vía el punto de contacto ZX Interface 1.

Nota
El ZX Spectrum + tiene dos soportes que pueden ser empleados para inclinar el teclado. No es necesario emplearlos cuando se instala el conector ZX Interface 1.

Mediante el cable de red incluido con la unidad de punto de contacto (interface), es posible vincular su ordenador con otro ordenador – ya sea un ZX Spectrum o un ZX Spectrum +. De esta forma el ordenador ofrece la posibilidad construir una red de ordenadores, con un máximo de 63 unidades Spectrum.

La unidad de punto de contacto ZX Interface 1 también incluye un punto de contacto RS232 con un enchufe de 9 vías en D (9-way D). Además se encuentra disponible un cable de conexion de tipo estandard.



Las impresoras de tipo estandard son conectadas al ordendor mediante el punto de contacto ZX Interface 1.

Cartuchos ROM y palancas de control

Los puntos de contacto como la ZX Interface 2 le permiten conectar cartuchos ROM y palancas de control. En el caso de los cartuchos ROM, la información contenida en ellos se carga automáticamente en el ordenador, cuando este es conectado, no siendo necesario invertir un tiempo considerable en cargar programas o información desde cinta magnética.

Punto de contacto (interface) para el cartucho ROM/palanca de control.

Conector o enchufe de reborde
Este enchufe es utilizado para conectar equipos periféricos al ordenador.

Advertencia
Los accesorios periféricos siempre deben ser conectados antes de encender el ordenador.

MAPA DE LA MEMORIA DEL ZX SPECTRUM +

La fotografía del interior del ordenador Spectrum, en las páginas 42–43, muestran entre otros componentes una chip para la memoria ROM y 16 chips más pequeñas para la memoria RAM. En conjunto estas chips constituyen la totalidad de la memoria del Spectrum. La memoria del Spectrum consiste de 65536 unidades de almacenamiento (storage units), cada una de las cuales contiene un byte (un número entre Ø y 255). Cada unidad es identificada por un número denominado su *dirección*.

En inglés ROM significa *Read Only Memory*, es decir la memoria que solamente puede ser leída por el ordenador, pero que no puede ser utilizada por el usuario para almacenar información ni borrada. Esta memoria permanente contiene las instrucciones de operación de la Unidad Procesadora Central (en inglés Central Processing Unit o CPU). La capacidad de la chip ROM es de 16K. Esto significa que la chip puede contener 16x1Ø24 bytes o direcciones (en total 16384 bytes). Estos bytes solamente pueden ser leídos, no es posible modificarlos. (Si fuera posible cambiarlos la consecuencia sería que el ordenador no podría continuar funcionando). Para examinar el contenido de una posición o dirección en la memoria del ordenador se utiliza la palabra clave PEEK (en inglés "peek" literalmente significa "mirar").

RAM significa *Random Access Memory*, es la memoria a la cual el usuario tiene acceso y contiene todos los programas e información introducida en el ordenador desde el teclado o por otros medios. La capacidad de la memoria RAM del Spectrum es 48K, es decir, este sector de la memoria total contiene 48x1Ø24 bytes o direcciones (en total 49152 bytes). Random Acces significa en castellano "Acceso aleatorio" o "Acceso al azar", lo que indica que cualquiera de las direcciones de este sector de la memoria puede ser modificada por el programador. Para cambiar el contenido de una dirección en la memoria RAM se utilizan sentencias conteniendo la palabra clave POKE.

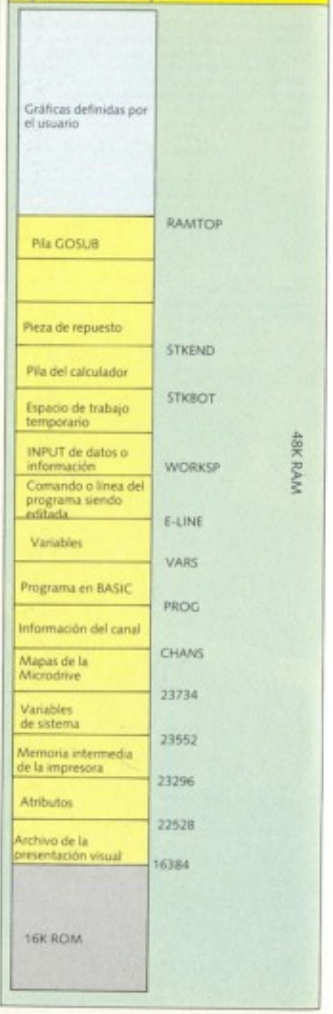
Las direcciones de la memoria se extienden desde Ø hasta 65535, la primera cuarta parte siendo memoria ROM y el resto memoria RAM.

Variables de sistema

La columna a la derecha muestra la organización de la memoria del Spectrum. El diagrama indica la posición de las diversas secciones que controlan el ordenador. Es posible cambiar la posición de varias de estas secciones. Los límites de estas secciones son determinados por *variables de sistema*.

Las *variables de sistema* utilizadas por el ordenador Spectrum no son variables como las que existen en el idioma BASIC. Las variables de sistema son nombres para ciertos valores útiles ubicados en direcciones o posiciones particulares de la memoria del ordenador. El propósito de estos nombres es ayudarle a recordar la importancia del valor almacenado en cada una de esas posiciones. Por ejemplo, la variable de sistema RAMTOP es la dirección en el tope o punto más alto del espacio destinado a la memoria RAM. El valor de RAMTOP establece la cantidad de espacio disponible para almacenar programas en BASIC y los valores de sus variables.

Mapa de la memoria del Spectrum.



APRENDIENDO ACERCA DEL BASIC DE SINCLAIR

Este capítulo describe los detalles del BASIC empleado por Sinclair, suministrando un sumario acerca del empleo de cada una de las palabras clave y otros detalles adicionales acerca de aquel. La información cubre los aspectos requeridos para crear desde los programas en BASIC más sencillos hasta los más complejos y avanzados. El capítulo no ha sido organizado para ser leído desde el principio al final. Su estructura es la de un diccionario para la programación, y le permitirá sacar el máximo provecho de su Spectrum.

4

GUÍA DE REFERENCIA PARA EL PROGRAMADOR, UTILIZANDO LAS PALABRAS CLAVE DEL BASIC DE SINCLAIR

Esta guía contiene una descripción completa de todas las palabras claves en BASIC disponibles en el ZX Spectrum +. Cada entrada incluye información sobre los siguientes aspectos:

- Posición de la palabra clave
- Clase de palabra clave
- Propósito de la palabra clave
- Empleo de la palabra clave
- Formato utilizado al redactar un programa conteniendo la palabra clave
- Código del juego de carácteres

Los detalles referentes a la posición, propósito, empleo y código del juego de caracteres no requieren más explicación. La información sobre clase y formato es algo más compleja. Por lo tanto, le sugerimos que, para hacer un mejor uso de la guía, primero estudie cuidadosamente la información en esta página.

Clases de Palabras Clave

La palabras clave pueden clasificarse entre una y cuatro clases.

Comandos
Un comando es una palabra clave que le indica al ordenador que ejecuta determinada acción. Ciertos comandos serán puestos en práctica inmediatamente, por ejemplo: RUN, LOAD.

Sentencias
Como los comandos, las sentencias son palabras claves que le indican al ordenador que ejecute determinada acción. La diferencia entre ambas radica en que las sentencias son incluidas en un programa, y por lo tanto serán llevadas a la práctica únicamente cuando el ordenador ejecuta el programa.
Ejemplos: DRAW, INPUT.

Función
Es una palabra clave que produce un valor. La función puede formar parte de un comando o una sentencia.
Ejemplos: RND, INT.

Operador lógico
Es una palabra clave empleada para realizar procesos lógicos, y puede ser incluida en un comando o una sentencia. Puede determinar si una determinada condicion es verdadera o falsa, o cambiar tal condición. El Spectrum tiene tres operadores lógicos: AND, OR y NOT.

Números y variables

Números
Almacenados con una precisión de 9 o 10 dígitos. La amplitud en la manipulación de números es de entre aproximadamente 10^{38} y $4*10^{-39}$.

Variables aceptadas
Numéricas. Cualquier longitud, comenzando con una letra. Se ignoran los espacios y todas las letra son convertidas en minúsculas. No se distingue entre mayúsculas y minúsculas.
Cadena. Cualquier letra sola seguida de un signo de $. No se distingue entre mayúsculas y minúsculas.
Matriz. Para variables de matriz ver la sección sobre DIM.

Formato de las palabras clave

La sección sobre formato de las palabras claves describe la sintaxis correcta de cada palabra clave, explicando cual es la combinación correcta de palabras claves, y demás factores incluidos en una instrucción introducida en el ordenador. Los elementos frecuentemente combinados con palabras claves incluyen variables y valores. En el texto se utilizarán las siguientes abreviaciones al describir el formato correcto para una palabra clave determinada.

Abreviación	Explicación	Ejemplo
const-num	Una constante numérica (número)	24.5
var-num	Una variable numérica (una variable que puede contener un número)	suma
expr-num	Expresión numérica (cualquier combinación de constantes y variables numéricas que producen un número como resultado)	suma*24.5 RND*7
const número entero var número entero entero expres número entero	Una constante, variable o expresión numérica cuyo valor es redondeado al número entero más próximo	
const de cadena	Una constante de cadena o cadena (cualquier combinación de carácteres entre comillas)	"ZX Spectrum+"
var de cadena	Una variable de cadena (una variable que puede contener un cadena)	a$
expr de cadena	Una expresión de cadena (cualquier combinación válida de constantes de cadena y variables de cadena que suministre una cadena)	a$+"ZX Spectrum+" a$ (6 TO 8)
letra	Cualquier letra mayúscula o minúscula	Y x
letra$	Cualquier letra mayúscula o minúscula seguida del signo $	B$ a$
cond	Una condición, o una sub-condición insertada dentro de una condición	x=10 AND t<=10
sentencia	Cualquier sentencia en BASIC que es válida cuando utilizada	IF t<>10 THEN STOP
[ ]	Un elemento opcional que puede ser repetido	

NOTA: En el texto los términos valor numérico (numeric value) y valor de cadena (string value) denotan cualquier elemento numérico o de cadena respectivamente.

Los signos en el Sinclair BASIC

Signo	Ubicación	Acción/Use
$	4	Variable de cadena
'	7	Inicia nueve línea
(	8	Abre paréntesis
)	9	Cierra paréntesis
<=	Q	Menos o igual que
<>	W	No es igual a
>=	E	Más o igual que
<	R	Menor que
>	T	Mayor que
↑	H	Elevar a la potencia
-	J	Resta. Número negativo.
+	K	Suma. Número positivo.
=	L	Igual
:	Z	Separa sentencias en la de programa
/	V	División
*	B	Multiplicación
.	Tecla .	Punto decimal
;	Tecla ;	Controla visualización de información en la pantalla: Separa sentencias dentro de la sentencia del programa.
"	Tecla "	Abre y cierra una cadena
,	Tecla ,	Controla visualización de información en la pantalla: Ubica información en las columnas 0 o 16 de la pantalla. Separa los valores insertados a continuación de una palabra clave.

Juego de carácteres del ZX Spectrum +

	0	1	2	3	4	5	6	7	8	9
0							PRINT coma	EDIT PAPER control	↑cursor izquierda	gráficas derecha
10	cursor abajo	cursor arriba	DELETE	ENTER	número		INK control		FLASH control	BRIGHT control
20	INVERSE control	OVER control	AT control	TAB control						
30			espacio	!	"	#	$	%	&	'
40	(	)	*	+	,	-	.	/	0	1
50	2	3	4	5	6	7	8	9	:	;
60	<	=	>	?	@	A	B	C	D	E
70	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y
90	Z	[	/	]	↑	_	£	a	b	c
100	d	e	f	g	h	i	j	k	l	m
110	n	o	p	q	r	s	t	u	v	w
120	x	y	z	{	|	}	~	©		
130										
140					GRAPHICS A	GRAPHICS B	GRAPHICS C	GRAPHICS D	GRAPHICS E	GRAPHICS F
150	GRAPHICS G	GRAPHICS H	GRAPHICS I	GRAPHICS J	GRAPHICS K	GRAPHICS L	GRAPHICS M	GRAPHICS N	GRAPHICS O	GRAPHICS P
160	GRAPHICS Q	GRAPHICS R	GRAPHICS S	GRAPHICS T	GRAPHICS U	RND	INKEY$	PI	FN	POINT
170	SCREEN$	ATTR	AT	TAB	VAL$	CODE	VAL	LEN	SIN	COS
180	TAN	ASN	ACS	ATN	LN	EXP	INT	SQR	SGN	ABS
190	PEEK	IN	USR	STR$	CHR$	NOT	BIN	OR	AND	<=
200	>=	<>	LINE	THEN	TO	STEP	DEF FN	CAT	FORMAT	MOVE
210	ERASE	OPEN #	CLOSE #	MERGE	VERIFY	BEEP	CIRCLE	INK	PAPER	FLASH
220	BRIGHT	INVERSE	OVER	OUT	LPRINT	LLIST	STOP	READ	DATA	RESTORE
230	NEW	BORDER	CONTINUE	DIM	REM	FOR	GO TO	GO SUB	INPUT	LOAD
240	LIST	LET	PAUSE	NEXT	POKE	PRINT	PLOT	RUN	SAVE	RANDOMIZE
250	IF	CLS	DRAW	CLEAR	RETURN	COPY				

ABS ABSolute value — Valor Absoluto

Posición en el teclado
EXTEND MODE
G

Función

La palabra clave ABS produce la magnitud absoluta de un valor numérico. Esto es, el valor sin un signo de positivo o negativo.

Como emplear ABS
ABS es seguido por un valor numérico. Una expresión debe estar encerrada *entre paréntesis*, por ejemplo:

```
50 LET x=ABS(y-z)
```

ABS retorna el valor absoluto del valor numérico.

Ejemplo
El comando

```
PRINT ABS -34.2
```

produce como resultado que el ordenador visualize el número 34.2

Formato
ABS const-num
ABS var-num
ABS (expr-num)

ACS Arc CoSine — Arco CoSeno

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT W

Función

ACS calcula el valor de un ángulo a partir de su coseno.

Como emplear ACS

SVD rd seguido por un valor numérico. Una expresión debe estar *encerrada entre paréntesis*, por ejemplo

```
60 LET x=ACS(y*z)
```

El valor a continuación de la palabra clave ACS (y*z en el ejemplo) es el coseno del ángulo requerido y puede variar entre +1 y −1. ACS retorna el valor del ángulo en radianes (no en grados). Para convertir los radianes en grados se multiplica el valor retornado por ACS por 180/PI.

Ejemplo

```
PRINT 180/PI * ACS 0.5
```

El resultado será 60, el ángulo en grados correspondiente a un coseno de 0.5.

Formato
ACS const-num
ACS var-num
ACS (expr-num)

AND

Posición en el teclado
SYMBOL SHIFT

Operador lógico/función

AND opera como un operador lógico, para determinar la veracidad de una combinación de condiciones. Solamente produce el efecto deseado si todas las condiciones enumeradas en la sentencia son verdaderas. AND también actúa como una función para ejecutar operaciones binarias en dos valores de cadena o dos valores numéricos.

Como emplear AND
Como un operador lógico, AND vincula dos condiciones contenidas en una sentencia, donde es necesario establecer la veracidad del conjunto, por ejemplo:

```
90 IF x=y+z AND time<10
   THEN PRINT "Correcto"
```

Unicamente si ambas condiciones (x=y+z) y (time < 10) son verdaderas, el ordenador visualizará la palabra Correcto en la pantalla. Si cualquiera de las dos condiciones es falsa, entonces la totalidad de la sentencia será considerada como falsa por el ordenador continuará a la próxima línea del programa.

AND como una función
Como una función AND puede operar en relación a dos valores numéricos, por ejemplo

```
50 LET x=y AND z
```

AND retorna el primer valor (y) si el segundo valor (z) no es igual a 0, y retorna 0 si el segundo valor (z) es 0.
AND también puede operar con un valor de cadena (string value) siempre y cuando éste precedía dicha palabra clave. Un valor numérico debe siempre seguir AND, por ejemplo

```
50 LET a$=b$ AND z
```

AND retorna el primer valor (b$) si el segundo valor (z) no es cero. AND retornará una cadena nula o vacía si el segundo valor (z) es cero.
Observe que el ZX Spectrum + asigna un valor de 1 a una condición verdadera y un valor de 0 a una condición falsa. El ordenador reconoce cualquier valor que no sea cero como verdadero y el valor cero como falso. El Spectrum no evalúa combinaciones de valores numéricos de acuerdo con las tablas de veracidad estandard.

AND retorna el primer valor (b$) si el segundo valor (z) no es cero. AND retornará una cadena nula o vacía si el segundo valor (z) es cero.
Observe que el ZX Spectrum + asigna un valor de 1 a una condición verdadera y un valor de 0 a una condición falsa. El ordenador reconoce cualquier valor que no sea cero como verdadero y el valor cero como falso. El Spectrum no evalúa combinaciones de valores numéricos de acuerdo con las tablas de veracidad estandard.

Formato
cond. **AND** cond.
expr-num **AND** expr-num.
expr de cadena **AND** expr-num.

ASN Arc Sine — Arco Seno

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT Q

Función

ASN calcula el valor de un ángulo a partir de su seno.

Como emplear ASN
ASN es seguido por un valor numérico. La expresión debe estar *encerrada entre paréntesis*, por ejemplo:

```
60 LET x=ASN(y*z)
```

El valor siguiendo la palabra clave ASN (en el ejemplo: y*z) es el seno del ángulo requerido, y puede varia entre −1 y +1. Obedeciendo la palabra clave ASN el ordenador retornará el valor del ángulo en radianes (no en grados). Para convertir los radianes en grados se multiplica el valor retornado por ASN por 180/PI.

Ejemplo

```
PRINT 180/PI * ASN 0.5
```

el resultado será 30, el ángulo en grados equivalente a un seno de 0.5.

Formato
ASN const-num
ASN var-num
ASN (expr-num)

AT

Posición en el teclado
SYMBOL SHIFT I

Ver: **INPUT**;**LPRINT**;**PRINT**

ATN Arc TaNgent — Arco Tangente

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT E

Función

ATN le indica al ordenador que calcule un ángulo a partir de su tangente.

Como emplear ATN
La palabra clave ATN es seguida por un valor numérico. La expresión debe estar *encerrada entre paréntesis*, por ejemplo:

```
60 LET x=ATN(y*z)
```

El valor a continuación de ATN (en el ejemplo: y*z) es es la tangente del ángulo requerido. TN retornará el valor del ángulo en radianes (no en grados). Para convertir los radianes a grados multiplique el resultado suministrado por ATN por 180/PI.

Ejemplo

```
PRINT 180/PI * ATN 1
```

retornará el resultado 45, que es el ángulo en grados correspondiente a una tangente de 1.

Formato
ATN const-num
ATN var-num
ATN (expr-num)

ATTR ATTRibutes — Atributos

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT L

Función

ATTR suministra información sobre los atributos de una determinada posición de carácter en la pantalla. Los atributos son los colores de la tinta y el papel, y si los efectos de brillo, y "relampagueo" se encuentran operando o no.

Como emplear ATTR
ATTR es seguido de dos valores numéricos, *separados por una coma y encerrados entre paréntesis*. Por ejemplo:

```
150 IF ATTR(v,h)=115 THEN
    GOSUB 2000
```

El primer valor a continuación de la palabra clave ATTR ('v' vertical) puede variar entre 0 y 23, y se refiere a la posición de una de las líneas o hileras en la pantalla de la televisión. El segundo valor ('h' horizontal) puede variar entre 0 y 31, y es el número de la comumna correspondiente. ATTR retornará un número es la suma de los atributos existentes en el punto especificado. Este número se compone de los siguientes elementos:

Color de la tinta	Código de color (entre 0 y 7)
Color del papel	8 veces el código de color.
Brillo	64
"Relampagueo" (Flash)	128

Ejemplo
Supongamos que un carácter situado en la posición 11,16, es visualizado en un color de tinta número 3 (magenta), color de papel número 6 (amarillo), y es brillante pero no relampaguea. En este caso cuando el ordenador recibe el comando

```
PRINT ATTR (11,16)
```

Responderá: 115(3+8X6+64+0)

ATTR en forma binaria
ATTR retorna un byte. Si el bit 7 de este byte, (muy significativo) es 1 esto indica que el efecto de relampagueo se encuentra en operación y 0 que está apagado. Lo mismo sucede en el caso del bit 6 que se refiere al efecto de brillo: el bit 1 indica que se encuentra en operación, y el bit 0 que está apagado. Los bits 5 a 3 se refieren al color del papel (en binario), y los bits de 2 a 0 ofrecen información sobre la tinta.

Formato
ATTR(expr-num,expr-num)

BEEP

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT Z

Sentencia/comando

La palabra clave BEEP causa que el altavoz del ordenador emita una nota de duración y tono predeterminados.

Como emplear BEEP
BEEP puede ser utilizado para formar una sentencia contenida en un programa o también puede ser empleado como un comando directo. Esta palabra clave es seguida por dos valores numéricos, *separados por una coma*, por ejemplo

```
60 BEEP x,y
```

El primer valor (x) puede variar entre 0 y 10. Su propósito es definir la duración en segundos de la nota emitida. El segundo valor (y) puede variar entre −60 a +69 y define el tono de la nota, en semitonos. Los valores negativos son por debajo del "do" central, y los positivos por encima del "do" central.

Ejemplo
El comando

```
BEEP 0.5, 1
```

producirá la nota do sostenido por encima del do central, con una duración de medio segundo.

Formato
BEEP expr-num, expr-num

BIN BINary number — Número BINario

Posición en el teclado
EXTEND MODE
B

BIN transforma un número binario en un número decimal.

Como emplear BIN
BIN es seguido por el número binario. El número binario puede consistir de hasta dieciseis números 1 o 0, por ejemplo

```
50 POKE USR "a",BIN 10101010
```

La palabra clave producirá el número decimal correspondiente al número binario. BIN es utilizado comunmente en conjunción con las palabras clave POKE y USR, como en el ejemplo anterior, para crear carácteres gráficos definidos por el usuario del ordenador. En este caso 1 significa un pixel de color de tinta y 0 un pixel de color de papel.

Ejemplo
El comando

```
PRINT BIN 11111110
```

visualizará el número decimal 256, el equivalente del valor binario indicado en la sentencia.

Formato
BIN [1] [0]

BORDER Borde

Posición en el teclado
B

Sentencia/Comando

BORDER especifica el color del borde en torno del area visualización en la pantalla.

Como emplear BORDER
La palabra clave BORDER puede ser utilizado como un comando directo o como una sentencia contenida en un programa. BORDER es seguida por un valor numérico, por ejemplo

```
30 BORDER RND*7
```

El valor a continuación de la palabra clave BORDER es redondeado al número entero más proximo. Su función es especificar el color del borde como sigue:

0 Negro
1 Azul
2 Rojo
3 Magenta (morado)
4 Verde
5 Azul-verde
6 Amarillo
7 Blanco

Observe que BORDER también define el color del papel en la parte inferior de la pantalla. A diferencia de lo que sucede en el caso de las palabras clave INK y PAPER, una sentencia conteniendo la palabra clave BORDER no puede ser incorporada (insertada) en una sentencia conteniendo la palabra clave PRINT

Ejemplo

```
BORDER 2
```

produce un borde de color rojo en torno del área de visualización en la pantalla.

Formato
BORDER expr número entero

BRIGHT

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT B

Sentencia/comando

El resultado de BRIGHT es incrementar el brilo de los colores de los carácteres visualizados en la pantlla por el ordenador.

Como emplear BRIGHT
La palabra clave BRIGHT puede ser utilizado como un comando directo, pero normalmente es empleada para formar una sentencia incluida en un programa. BRIGHT es seguida de un valor numérico. Por ejemplo

```
30 BRIGHT 1
```

El valor a continuación de BRIGHT es redondeado al número entero más próximo si fuera necesario, entre 1 y 8. Un valor de 1 determina que todos los carácteres visualizados a continuación en la pantalla, en ejecución de sentencias PRINT o INPUT, aparezcan en colores de tinta y papel más brillantes. Un valor de 8 causa que al visualizarse nuevos carácteres en las posiciones de los carácteres las posiciones brillantes aquellos se mantengan brillantes y que las posiciones de carácteres normales se mentengannormales. La palabra clave BRIGHT seguida de 0 cancela las instrucciones BRIGHT 1 y BRIGHT 8, y como resultado, todos los carácteres visualizados a continuación serán de brilo normal.

BRIGHT también puede ser insertada dentro de sentencias afectando la visualización de información en la pantalla y conteniendo las palabras clave PRINT, INPUT, PLOT, DRAW, y CIRCLE. La palabra clave BRIGHT es ubicada a continuación de una de acquellas palabras clave, pero precede los parámetros de la data o visualización. BRIGHT es seguida por los mismos valores y un punto y coma, por ejemplo

```
5Ø PRINT BRIGHT 1;
   "ADVERTENCIA"
```

El efecto de BRIGHT entonces será local y se aplicará exclusivamente a los carácteres visualizados, punto dibujado (plotted), o línea trazada por el ordenador en la pantalla, en ejecución de ese comando. Recuerde que BRIGHT 1 aumenta el brillo del color del papel de la totalidad de la posición del caracter, de 8×8 pixels, si cualquiera de los pixels en dicha posición es dibujado en color de tinta.

Formato
BRIGHT expr número entero [;]

CAT CATalogue

Comando para el manejo de información almacendada en Microdrives. Ver el Manual sobre la Microdrive y el punto de contacto Interface 1.

CHR$ CHaRacter string Cadena de carácteres

Posición en el teclado
EXTEND MODE
U

Fonction

El juego de carácteres del Spectrum se compone de los carácteres y palabras clave en el teclado del ordenador más cualquier carácter gráfico definido por el usuario. Mediante el empleo de la palabra clave CHR$ y un número-código, cada uno de los componentes del juego de carácteres puede ser obtenido como una cadena. El juego de carácteres también incluye varios códigos de control que afectan la visualización de los carácteres en la pantalla. Estos códigos pueden ser puestos en operación, y los carácteres visualizados en la pantalla, incorporando la palabra clave PRINT antes de CHR$. El juego de carácteres completo y sus números-códigos se detallan en la página 51.

Como emplear CHR$
CHR$ es seguido por un valor numérico, por ejemplo

```
8Ø PRINT CHR$ x
```

Una expresión debe ser *encerrada entre paréntesis*. El valor a continuación de CHR$ (x en el ejemplo) es redondeado al número entero más próximo. Los números enteros entre 32 y 255 producen como una cadena alguno de los carácteres en el teclado del ordenador, elementos gráficos definidos por el usuario, o una palabra clave. El Spectrum emplea los valores en el código ASCII para valores entre 32 y 95, y entre 97 y 126. En el ejemplo, si se asigna un valor de 65 a la variable x, el ordenador visualizará la letra A.

Códigos de control CHR$
Los valores entre 1 y 31 solamente se utilizan para producir códigos de control. Las sentencias CHR$6 (PRINT coma), Ø8 retroceder un espacio) y 13 (nueva línea o ENTER) afectan las imagenes visualizadas en la pantalla cuando incluidos en una sentencia con la palabra clave PRINT. CHR$ puede ser seguido por el valor-código y un punto y coma, por ejemplo

```
6Ø PRINT "A"; CHR$6; "B"
```

Este sentencia le indica al ordenador que visualize lo siguiente:

A B

Los códigos de control CHR$ también pueden ser utilizados para formar una cadena compuesta, que los contiene. La sentencia

```
6Ø PRINT "A" +CHR$6+ "B"
```

tiene exactamente el mismo efecto que la anterior.

Los códigos 16 a 23 afectan el color y la posición, y pueden ser incluidos en cadenas compuestas conjuntamente con CHR$. En este último caso, CHR$ 16 (control de la tinta o INK) y CHR$ 17 (control del papel o PAPER) serán seguidos por un valor código para los colores entre Ø y 7; y en el caso de las palabras claves para los efectos especiales (FLASH, BRIGHT, INVERSE y OVER) los CHR$18 a CHR$21 serán seguidos por el número Ø (pone en operación el efecto especial) o 1 (lo termina). Por ejemplo, el comando

```
PRINT CHR$16+CHR$3+CHR$17
+CHR$6+CHR$18+CHR$1
+"ZX Spectrum +  "
```

tendrá el resultado de visualizar las palabras ZX SPECTRUM + en letras a destellos, rojas y amarillas. Como en los ejemplos precedentes, es posible sustituir los símbolos de + por puntos y comas.

CHR$22 (control AT) es seguido por dos valores CHR$ para indicar los números de la línea y columna. El comando

```
PRINT CHR$22+CHR$11
+CHR$16+CHR$42
```

le indica al ordenador que visualize una estrella en el centro de la pantalla.

CHR$23 (control TAB) también es seguido por dos valores de la misma manera. El segundo valor normalmente es Ø y el primero indica la posición TAB.

```
PRINT CHR$23+CHR$16+
CHR$Ø + CHR$42
```

produce la estrella en otra posición.

Tenga en cuenta que estos son los únicos controles disponibles. Si se utiliza PRINT CHR$ con un valor de palabra clave más de 164 solamente se visualizará el nombre de la palabra clave sin ponerla en ejecución.

Formato
CHR$ const número entero [,] [+]
CHR$ var número entero [,] [+]
CHR$ (expr número entero) [,] [+]

CIRCLE Circulo

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT H

Sentencia/comando

CIRCLE dibuja un circle en la pantalla.

Como emplear CIRCLE
La palabra clave CIRCLE es seguida de tres valores numéricos, *cada uno de ellos separado por una coma*. Ejemplo

```
8Ø CIRCLEx,y,z
```

Si fuera necesario cada uno de los tres valores será redondeado al número entero más cercano. El ordenador, traza un círculo en la cuadrícula de gráficos de alta resolución, utilizando el color de tinta vigente en ese momento. Los primeros dos valores (x,y) definen coordinadas horizontal y vertical del centro del círculo. El tercer valor (z), define el largo del radio. Las dimensiones deben ser tales que el círculo no se extienda fuera del área de visualización de la pantella.

CIRCLE es afectado por sentencias o comandos controlando el color de las figures en la pantalla, y puede incluir instrucciones referentes al color, con los mismos efectos que en las sentencias conteniendo las palabras clave PLOT y DRAW.

```
CIRCLE 128,88,87
```

dibujará un círculo ocupando la mayor parte del área de visualización.

Formato
CIRCLE [sentencia;] número entero, número entero, número entero

CLEAR Borrar

Posición en el teclado
X

Sentencia/comando

CLEAR borra los valores corrientes de todas las variables, liberando el espacio de la memoria ocupado por aquellos valores y el espacio hasta el límite RAMTOP, la dirección superior del área regida por el sistema BASIC. La palabra clave CLEAR también puede ser utilizada para reajustar el límite RAMTOP (literalmente RAMTOP significa "el tope del área RAM").

Como emplear CLEAR
CLEAR puede ser utilizado como un comando directo o puede ser incluido como parte de una sentencia incorporada en un programa. No requiere parámetros, por ejemplo

```
5Ø CLEAR
```

Obedeciendo la instrucción CLEAR, el ordenador borra los valores asignados corrientemente a todas las variables, incluyendo las matrices (arrays). También ejecuta las palabras clave CLS y RESTORE para limpiar la pantalla y ristaurar el indicador de información o datos (data pointer) al primer elemento de datos o información. Además, la posición de PLOT (dibujar) es reajustada, poniendola en el punto inferior a la izquierda del área de visualización. La pila de GOSUB también es borrada.

Observe que no se requiere incluir CLEAR antes de proceder a dimensionar matrices (arrays), ya que la palabra clave DIM utilizada en ese caso borra cualquier otra matriz preexistente de su mismo nombre. Tenga presente, además, que RUN ejecuta la sentencia o comando CLEAR.

CLEAR et RAMTOP
CLEAR puede ser seguida por un valor numérico, por ejemplo

```
CLEAR 65267
```

A continuación, CLEAR opera como se explicó más arriba, y acto seguido adjusta el RAMTOP – la dirección más alta del área en BASIC del ordenador – al valor indicado. En el ZX Spectrum + el valor máximo de RAMTOP es ajustado a 65367, y se encuentra ubicado debajo del área reservada para los gráficos definidos por el usuario. La palabra clave NEW borra el contenido de la memoria hasta el límite establecido en el RAMTOP. Como resultado, es posible recurrir al empleo de CLEAR para definir un sector de la memoria que es inmune a la palabra clave NEW. En el ejemplo, esta área se encuentra entre el límite máximo RAMTOP del ordenador ZX Spectrum (65367) y el límite establecido mediante la palabra clave CLEAR (65267), y su capacidad es de 1ØØ bytes. El elevar el límite RAMTOP incrementa la memoria disponible para BASIC, sacrificando parte del espacio reservado para los gráficos definidos por el usuario. Observe que la pila GOSUB será situada en el RAMTOP, siempre y cuando el RAMTOP permanezca por encima del la pila del calculador.

Para determinar cual es el RAMTOP, introducir en el ordenador las instrucciones siguientes.

```
PRINT PEEK 23730+256*PEEK23731
```

Formato
CLEAR (exp-num)

CLOSE

Comando para el manejo de información almacenada en Microdrives. Ver el maual sobre Microdrives y el punto de contacto Interface 1.

CLS Clear Screen Limpiar la pantalla

Posición en el teclado
V

Sentencia/comando

CLS borra todos los textos y gráficos en el área de visualización de la pantalla, dejándola en blanco y con el color de papel (fondo) corriente.

Como emplear CLS
CLS puede ser utilizado como un comando directo o como un elemento de una sentencia contenida en un programa. No requiere parámetros. Por ejemplo

```
25Ø IF a$= "NO" THEN CLS
```

Si se cumple la condición estipulada, el ordenador borrará el área de visualización (pero no el borde de la pantalla), adoptando el color de papel seleccionado por el comando o sentencia PAPER inmediatamente anterior. Si el programa no incluye tal sentencia o comando, el ordenador adoptará el color de papel (PAPER) blanco.

Note que CLS debe ser utilizada después de la palabra clave PAPER y antes de PRINT, o cualquier otra sentencia afectando la visualizacion, para producir un fondo de colores abarcando la totalidad del área de visualización.

Formato
CLS

CODE Código

Posición en el teclado
EXTEND MODE
I

Fonction

La función CODE suministra el número código de un carácter determinado en el juego de carácteres del Spectrum (ver página 51).

Como emplear CODE
CODE es seguido por una cadena (sting). Por ejemplo

```
9Ø IF CODE a$<65 OR CODE
   a$>9Ø THEN GOTO 8Ø
```

Las expresiones parte de la cadena deben ser *encerradas entre paréntesis*. CODE suministra el número de codigo del *primerr* caracter en la cadena. Si la cadena es una "cadena nula" o "vacia" entonces CODE suministrará el valor Ø.

CODE suministra los valores de código ASCII desde 32 a 95, y desde 97 a 126.

Ejemplo

```
PRINT CODE "ZX Spectrum +"
```

visualizará en la pantalla el número código **9Ø**, correspondiente a la letra Z.

SAVE CODE y LOAD CODE
CODE es utilizado en forma diferente cuando combinado con SAVE y LOAD. Ver: SAVE CODE Y LOAD CODE.

Formato
CODE const de cadena
CODE var de cadena
CODE (expr de cadena)

CONTINUE Continuar

Posición en el teclado
C

Comando

Si el programa se detiene, la palabra clave CONTINUE puede ser utilizada para volverlo a iniciar a partir del punto donde se detuvo. Si la detención en la ejecución del programa se debe a un error, este deberá ser rectificado antes de que CONTINUE Permita que el programa se reinicie.

Como emplear CONTINUE
Esta palabra clave es utilizada como un comando directo cuando un programa se ha detenido. No requiere parámetros. Normalmente, después de CONTINUE el programa se reinicia en la sentencia donde se produjo la interrupción. Si la causa de ésta fué error, es posible introducir un comando para rectificarlo y CONTINUE permitirá que el programa continue a partir de esa sentencia. Cuando el programa se ha detenido en una sentencia conteniendo la palabra clave STOP, causando un reporte 9, o si se detuvo debido a que la tecla BREAK fué oprimida, causando un reporte L, entones CONTINUE reinicia el programa a partir de la sentencia siguiente. Antes de introducir CONTINUE es posible introducir un comando para rectificar 1 situación si fuera necesario.

Cuando se emplea CONTINUE para reasumir un comando directo: esta palabra clave entrará en un bucle (loop) si el comando se detuvo en la primera sentencia del comando. En este caso la imagen desaparece, pero es posible recuperar control oprimiendo la tecla BREAK. CONTINUE causará un reporte Ø si el comando se detuvo en la segunda sentencia, y un reporte N si se detuvo en la tercera sentencia o alguna de las subsecuentes.

Formato
CONTINUE

COPY

Posición en el teclado
Z

Comando

COPY causa que la impresora Sinclair produzca una copia de la información visualizada en la pantalla de la televisión.

Como emplear COPY
La palabra clave COPY es utilizada como un comando directo cuando un programa ha sido ejecutado completamente o detenido. No requiere parámetros. Si la impresora se encuentra conectada y funcionando, después del comando COPY producirá una copia de las primeras 22 líneas de la imagen en la pantalla. Los colores de tinta (primer plano) serán impresos en negro; los colores de papel (fondo) no son impresos. Es posible detener la impresora oprimiendo la tecla BREAK.

Cuando se ha visualizado el listado de un programa en la pantalla, es posible copiarlo mediante el comando COPY, si el listado fué producido mediante el comando o una sentencia LIST. Recuerde que un listado también aparecerá en la pantalla al presionarse le tecle ENTER, después que el programa ha sido completado o detenido. Sin embargo, en este último caso, de "listado automático", el listado no podrá ser copiado mediante el comando COPY.

Formato
COPY

COS COSine Coseno

Posición en el teclado
EXTEND MODE W

Fonción

COS suministra el coseno de un ángulo.

Como emplear COS
COS es seguido por un valor numérico, por ejemplo

```
14Ø LET x=COS y
```

Una expresión debe *ser encerrada entre paréntesis*. El valor a continuación de COS es el ángulo en radianes. COS suministrará el coseno del ángulo. Es posible convertir grados a radianes multiplicándolos por PI/18Ø.

Observe que COS suministra un valor negativo para ángulos de entre 9Ø a 27Ø grados, y un valor positivo

para ángulos entre 0–90 y 270–360 grados.

Ejemplo
El comando

```
PRINT COS (60*PI/180)
```

produce como resultado el valor 0.5, el coseno correspondiente a un ángulo de 60 grados.

Formato
COS const-num
COS var-num
COS (expr-num)

DATA Datos

Posición en el teclado
EXTEND MODE
D

Sentencia

DATA suministra una lista de elementos de información o datos dentro de un programa. Por ejemplo, estos elementos de información pueden ser los valores adjudicados a variables o cadenas a ser visualizadas por el ordenador. Cada elemento de información es asignado a una variable mediante una sentencia de READ.

La asignación de datos o información a las variables en el programa se efectua siguiendo el orden en el cual dichos datos aparecen en el programa. Sin embargo, es posible utilizar la palabra clave RESTORE para comenzar la asignación de datos a variables en el primer elemento de información contenido en una sentencia de DATA.

Como emplear DATA
DATA solamente puede ser empleada para formar una sentencia parte de un programa. Normalmente esta palabra clave es seguida por una lista de constantes numéricas o de cadena, *separadas entre si por comas*. Por ejemplo

```
50 DATA 31, "ENERO", 28,
"FEBRERO"
```

Cada una de las constantes es asignada a una variable determinada mediante una sentencia de READ, que le indica al ordenador que lea un determinador elemento de información o dato. La sentencia de DATA puede ser ubicada en cualquier parte del programa. El número, tipo (numérica o de cadena), y orden de las constantes debe corresponder a la cantidad de veces que el ordenador debe ejecutar la sentencia de READ. La naturaleza y orden de las variables en la sentencia de READ también debe estar de acuerdo con los elementos en la sentencia de READ también debe estar de acuerdo con los elementos en la sentencia de DATA. Si existen demasiados elementos de información para ser ubicados en la misma sentencia la lista de DATA puede ser separada en varias sentencias de DATA sucesivas.

Ejemplo
el programa siguiente

```
10 FOR n=1 TO 2
20 READ a$,x
30 PRINT a$,x; "dias"
40 NEXT n
50 DATA 31, "Enero", 28,
   "Febrero"
```

visualizará lo siguiente en la pantalla:

Enero	31 dias
Febrero	28 dias

Utilizando DATA con variables
Cada uno de los elementos de información o datos contenidos en una sentencia de DATA puede consistir de variables numéricas o de cadena, o de expresiones, siempre y cuando previamente se les haya asignado valores. En el ejemplo anterior, la sentencia de DATA puede ser modificada como sigue.

```
50 DATA d,m$,d-3 "Febrero"
```

Si previamente se atribuyó un valor de 31 a la variable d, y un valor "Enero" a la variable m$, entonces el ejecutarse se obtendrá precisamente el mismo resultado que antes.

LOAD DATA, SAVE DATA y VERIFY DATA
La palabra clave DATA también puede ser utilizada con LOAD, SAVE, y VERIFY para almacenar matrices (arrays) en cinta de cassette. Ver: LOAD DATA, SAVE DATA y VERIFY.

Formato
DATA expr-num [,expr-num]
[expr de cadena]
DATA expr de cadena [,expr-num]
[,expr-num]

DEF FN DEFine FuNction — DEFina Función

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT 1

Sentencia

DEF FN habilita al usuario para definir una función que no se encuentra disponible como palabra clave. Es posible pasar una variedad de parámetros a la función en una sentencia de FN. La sentencia de FN reclama aquella función y puede producir como resultado un valor numérico de cadena.

Como emplear DEF FN
DEF FN solamente puede ser utilizada como una sentencia contenida en un programa. Cuando se desea definir una función numérica, DEF FN es seguida por cualquier letra indivual y entonces por una o más variables numéricas, *separadas entre si por una coma y encerradas entre paréntesis*, por ejemplo: DEF FN r(x,y). Esto es seguido por signos de igual y una expresión numérica conteniendo las variables, por ejemplo

```
1000 DEF FN r(x,y)=
     SQR (x ↑ 2+y ↑ 2)
```

La letra a continuación de DEF FN (r en el ejemplo) es un nombre que identifica la función. Las variables también pueden consistir en letras individuales. Observe que en ambos casos, el Spectrum no distingue entre letras mayúsculas y minúsculas.

La expresión que sigue el signo de igual utiliza las variables (x,y en el ejemplo) para definir la función.

Una sentencia de DEF FN puede ser ubicada en cualquier parte del programa.

Para invocar la función definida por DEF FN, se utiliza la palabra clave FN seguida por la letra que designa la función y una lista de valores numéricos, *cada uno de ellos separado por una coma y encerrado entre paréntesis*, por ejemplo:

```
50 PRINT FN r(3,4)
```

Los valores numéricos entre paréntesis son transferidos a la función en el mismo orden en que se encuentran las variables en la sentencia DEF FN.

Asi, en el ejemplo anterior, x recibirá un valor de 3, mientras que y recibirá un valor de 4.

FN evalúa la expresión y produce como resultado un valor.

DEF FN también puede ser seguida por una letra y un par de paréntesis solamente, por ejemplo

```
1000 DEF FNr( )=INT(x+0.5)
```

El valor asignado en este momento a la variable (x en el ejemplo) es pasado a la función al ser llamado por la palabra clave FN.

En este caso, FNr( ) produce el valor asignado a x en este momento redondeado al número entero más próximo.

DEF FN y cadenas (string)
DEF FN también puede ser utilizado de la misma forma, con el propósito de definir e invocar una función de cadena (string function). En este caso, el nombre de la función es una sola letra seguida del signo $, además una o más de las variables en la sentencia es una letra seguida por el signo $.

La definición está formada por una expresión de cadena (string expression)

```
1000 DEF FN a$(b$,x,y)=
     b$(x TO y)
```

En este ejemplo, la expresión de cadena a continuación del signo de igual utiliza un fragmentador de cadena, y las variables x e y son el primary y último carácter de una sección de b$.

FN debe ser seguida por el nombre de la función y un valor de cadena entre paréntesis, conjuntamente con cualquier otro parámetro a ser pasado a la función.

En este caso, el comando

```
PRINT FN a$
("FUNDAMENTAL", 1,3)
```

Produce como resultado que el ordenador visualize en la pantalla lo siguiente FUN. De la misma manera, el comando

```
PRINT FN a$
("FUNDAMENTAL", 5,8)
```

producirá en la pantalla: AMEN.

Formato
DEF FN letra ([letra] [,letra]) =
expr-num
DEF FN letra$ ([letra$] [letra] [,letra]
[,letra$]) = expr de cadena
FN letra ([expr-num] [,expr-num])
FN letra$ ([expr de cadena]
[expr-num] [,expr-num] [,expr de cadena])

DIM DIMension — Dimensión

Posición en el teclado
D

Sentencia

La palabra clave DIM es utilizada para establecer las dimensiones (componer) una matriz (array) conteniendo una cantidad predeterminada de variables numéricas o de cadena.

Una matriz contiene un juego de variables, todas con el mismo nombre, que se distinguen las unas de las otras por sus subindices (en inglés: subscripts). Los subindices son valores que identifican cada una de las variables o elementos en la matriz.

Como emplear DIM can matrices numéricas
DIM es utilizado para formar una sentencia en un programa, siendo seguida por una letra individual que designa la matriz, y uno o mas valores numéricos. *Cada valor numérico es separado por una coma y encerrado entre paréntesis*. Por ejemplo

```
20 DIM x(10)
30 DIM z(20,5)
```

La primera es una matriz uni-dimensional, conteniendo diez elementos con subindices de 1 a 10. El nombre de la matriz es x, y sus variables son identificadas combinando el nombre de la matriz con sus respectivos subindices, por ejemplo: x(1), x(3), x(10), etc.

Cuando el ordenador llega a esta sentencia en el programa borra cualquier matriz con el mismo nombre existente en su memoria, y asigna un valor de 0 a cada una de las vaariables en la matriz.

Tenga presente que al establecer las dimensiones de una matriz, el Spectrum no distingue entre letras mayúsculas – para el ordenador la variable x (2) es idéntica a la variable X (2). Sin embargo las variables numéricas simples, teniendo la misma letra que el nombre de una matriz (en el ejemplo x o X) pueden coexistir con esta última y ser utilizadas separadamente si fuera necesario.

La cantidad de valores entre paréntesis es igual al numéro de dimensiones creadas en una matriz numérica.

El segundo ejemplo muestra una matriz bi-dimensional con un total de 100 elementos: 20 elementos en la primera dimension y 5 en la segunda (el total es 20 x 5). En estecaso los elementos se numeran z(1,1) hasta z(20,5)

Es posible crear matrices con cualquier número de dimensiones.

Los elementos de una matriz numérica pueden ser identificados subsecuentemente por el nombre de la matriz seguida por un valor entre paréntesis, por ejemplo

```
70 PRINT x(a)
160 PRINT z(7,b)
```

DIM y matrices de cadena
La palabra clave DIM es utilizada de la misma forma en las matrices numéricas, con la diferencia de que se utiliza una letra individual seguida por el signo $ para nombrar la matriz.

Además, debe agregarse un valor extra a los valores entre paréntesis para definir el largo de cada cadena. Por ejemplo:

```
30 DIM a$(20,5)
90 DIM b$(20,5,10)
```

La primera sentencia crea una matriz de 20 elementos, cada una de los cuales contiene una cadena de cinco caráctere. Las variables identificadas por subindices se designan desde a$(1) hasta a$(20) inclusive. Al principio, el ordenador asigna a cada variable una cadena "vacia" o "nula" (" ").

Como antes, el ordenador borrará de su memoria cualquier otra matriz del mismo nombre que pueda encontrarse en ella. Además, a diferencia de lo que sucede en el caso de las matrices numéricas, una variable de cadena simple no puede coexistir con una matriz de cadena del mismo nombre.

El segundo ejemplo crea una matriz de cadena bi-dimensional, con un total de 100 elementos: 20 elementos en la primera dimension y 5 en la segunda. Todos los elementos tienen una extensión de 10 carácteres.

Subsecuentemente, cuando se asignan valores a una matriz de cadena, estos son complementados por valores de relleno al final de la cadena, o son cortados, si fuera necesario para ajustarlo a la extension especificada.

Los elementos de una matriz de cadena son identificados por el nombre de la matriz, seguido por uno o más valores numéricos entre paréntesis, proporcionando el numéro o numéros del subindice correspondiente.

Por ejemplo, el elemento a$(2) de la matriz puede ser "PEREZ", y el elemento b$(12,4) puede ser "MADRID".

Sin embargo, es posible agregar un valor extra para definir un carácter particular en la cadena. En estos ejemplos a$(12,4,5) sería R (el cuarto carácter en MADRID).

Matrices de cadena de dimensión cero
Es posible crear una matriz de cadena con una dimensión de cero, utilizando solamente un valor entre paréntesis, por ejemplo

```
10 DIM c$(15)
```

Esta matriz contiene solamente un elemento, c$, cuya extensión es establecida en el valor definido de 15 carácteres (15).

Formato
DIM letra (expr-num [,expr-num])
DIM letra$ (expr-num [,expr-num])

DRAW Trazar

Posición en el teclado
W

Sentencia/comando

La palabra clave DRAW es utilizada para trazar líneas reactas o curvas en la pantalla.

Como emplear DRAW
Normalmente, DRAW es utilizada para formar una sentencia en un programa, cuando se requiere una línea recta, seguida de dos valores numéricos *separados por una coma*. Por ejemplo

```
40 DRAW x,y
```

El ordenador trazara una línea recta en la cuadrícula de gráficos de alta resolución. La línea será trazada a partir de la posición definida por la sentencia de PLOT anterior o la posición donde terminé la línea trazada en ejecución de una sentencia DRAW anterior, cualquiera sea la última.

Los dos valores a continuación de la palabra clave DRAW serán redondeados al numéro entero más próximo si fuera necesario.

El primer valor (x en el ejemplo) define la distancia horizontal desde esta posición, y el segundo valor (y) define la distancia vertical. Estos valores son negativos cuando la línea se trazará hacia la izquierda o hacia abajo en la pantalla. La posición donde la línea termina debe encontrarse dentro del área de visualización de la pantalla.

Cuando el programa no contiene sentencias de PLOT o DRAW anteriores, la nueva sentencia DRAW tendrá el efecto de comenzar a trazar la línea a partir de la esquina inferior izquierda de la pantalla, identificada por las coordenadas 0,0.

DRAW será afectada por sentencias conteniendo instrucciones o comandos relativos al color. También puede incluir sentencias incorporadas, con los mismos resultados que en las sentencias incluyendo las palabras clave PLOT y CIRCLE.

DRAW, trazando curves
DRAW puede ser seguido por un tercer valor para producir una curva, sektor de una circunferencia, por ejemplo

```
40 DRAW x,y,z
```

El tercer valor (z) define el ángulo en radianes de la línea trazada en la pantalla. Cuando el valor es positivo la línea será trazada hacia la izquierda, y hacia la derecha cuando es negativo. Los valores de PI o –PI producen un circulo.

Ejemplo
El programa siguiente traza un triángulo en la pantalla

```
10 PLOT 127, 150
20 DRAW 70, -100
30 DRAW -140, 0
40 DRAW 70,100
```

Cuando se agrega el valor 1 o –1 a la sentencia conteniendo DRAW, los lados se transformaran en curvas hacia adentro o hacia afuera.

Formato
DRAW [sentencia:] expr número entero, expr número entero [,expr número entero)

ERASE Borrar

Comando para la operación de almacenamiento de información mediante Microdrives. Ver el Manual de las Microdrives y puntos de contacto Interface 1.

EXP EXPonent — Exponente

Posición en el teclado
X

Función

EXP es una función aritmética que eleva el exponente a una potencia determinada.

Como emplear EXP
La palabra clave EXP es seguida de un valor numérico, por ejemplo

```
60 LET y=EXPx
```

Una expresión debe estar *encerrada entre paréntesis*. Como resultado EXP suministra el exponente e elevado a la potencia del argumento (x en nuestro ejemplo).

Ejemplo

```
PRINT EXP1
```

produce el valor de e, que es: 2.7182818

Formato
EXP const-num
EXP var-num
EXP (expr-num)

FLASH Relampago

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT
V

Sentencia/comando

La palabra clave FLASH le indica al ordenador que determinadas posiciónes de carácteres "relampagueen" o produzcan destellos. Este efecto óptico es obtenido alternando los colores de la tinta y el papel a una velocidad preestablecida.

Como emplear FLASH
FLASH puede ser utilizado como una comando directo, pero, por lo general, forma parte de una sentencia en un programa. Es seguido por un valor numérico, por ejemplo

```
50 FLASH 1
```

El valor numérico a continuación de FLASH es redondeneado al numéro entero más próximo si es necesario, y puede ser 0, 1 o 8.

El valor 1 causa que "relampagueen" todos los carácteres visualizados subsecuentemente en ejecución de sentencias conteniendo PRINT o INPUT.

FLASH es ubicado a continuación de la palabra clave, pero precede los parámetros de la información (datos) o de la imagen a ser visualizada. FLASH es seguido por los mismos valores y un punto y coma.

Por ejemplo

```
120 PRINT FLASH 1; INK 2;
    PAPER 6; "ATTENTION"
```

este caso el efecto de FLASH será local y se aplicará únicamente a los carácteres visualizados, punto o línea trazada en ejecución de esa sentencia.

Recuerde que FLASH determina el estado de la totalidad de la posición de 8x8 pixels, aunque solamente se dibuje (PLOT) en un color de tinta la posición de uno de ellos.

Formato
FLASH expr.numéro entero [;]

FN FUNction Función

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT
Z

Función

FN invoca una función definida por el usuario. Esta palabra clave siempre es utilizada en conjunción con DEF FN, la cual define la función a ser invocada.

Como emplear FN
Si se trata de una función numérica, FN es seguida por una letra y entonces por un par de paréntesis. Cuando se desea pasar parámetros a la función, estos son *separados por comas y encerrados entre paréntesis*. Por ejemplo

```
120 LETx=FN r(3,4)
```

Los parámetros (3 y 4 en el ejemplo) son pasados a la función, denominada r en este caso. A continuación FN produce el resultado.

El par de paréntesis debe ser incluido aún cuando no se requiera pasar parámetros a la función, por ejemplo

```
70 PRINT FN r( )
```

En este caso la función empleará los valores asignados corrientemente a sus variables.

FN invoca una función de cadena de acuerdo al mismo procedimiento, excepto que en este caso sera necesario agregar el símbolo **$** inmediatamente después de la letra que designa la función. Por más detalles consulte la sección sobre la palabra clave DEF FN.

Formato
FN letra ([expr-num] [,expr-num])
FN letra**$** ([expr de cadena] [expr-num] [,expr-num] [,expr de cadena])

FOR Para

Posición en el teclado
F

Sentencia/comando

FOR es utilizado siempre con las otras dos palabras clave TO y NEXT para construir un bucle (loop) FOR NEXT. Esta estructura permite que una determinada sección del programa se repita una determinada cantidad de veces.

Como emplear FOR
FOR siempre forma una sentencia con TO. La palabra clave FOR es seguida por una letra, un signo de igual, y dos variables numéricos separadas por la palabra clave TO. Por ejemplo

```
60 FOR a=1 TO 9
```

La letra (a en el ejemplo) es una variable de control. A continuación se incluyen las sentencias que deberán ser repetidas, normalmente una o más de estas sentencias hace uso de la variable de control. El bucle termina con una sentencia de NEXT. En esta última sentencia, NEXT es seguida por el nombre de la variable de control (a en el ejemplo). Por ejemplo

```
90 NEXT a
```

Al ejecutarse el programa, FOR cumple una serie de operaciones. Primero: FOR borra cualquier variable del mismo nombre que la variable de control. Segundo: FOR le asigna a la variable de control un valor inicial idéntico al valor especificado antes de TO (1 en el ejemplo). A continuación las instrucciones en las sentencias son ejecutadas. Tercero: cuando el ordenador llega a la palabra clave NEXT incrementará el valor de la variable de control en una unidad (a 2 en el ejemplo). Cuarto: el ordenador comparará el nuevo valor de la variable de control con el valor indicado después de TO (9). Este último valor establece el límite para el bucle. Quinto: si este valor es inferior al del límite, el ordenador reiniciará el proceso, retomando a la sentencia FOR y repitiendo el bucle FOR NEXT. Al finalizar cada bucle el valor de la variable de control se incrementa en una unidad. Cuando el ordenador comprueba que el valor de la variable de control es superior al valor límite (9) terminará el bucle. A continuación el programa continuará con la sentencia después de NEXT.

En el ejemplo, el bucle es repetido nueve veces, y el valor de la variable de control se incrementará de a 9. Al terminarse el último bucle, la variable de control tendrá el valor de 18.

Recuerde que el Spectrum no distingue entre letras mayúsculas y minúsculas cuando estas se utilizan para designar una variable de control.

Utilizando STEP en un bucle FOR NEXT
La palabra clave STEP puede ser incorporada en una sentencia para incrementar o disminuir el valor de la variable de control en pasos de más de 1. STEP es ubicado después del valor límite y es seguido por un valor numérico. Por ejemplo

```
60 FOR a=1 TO 9 STEP 2
```

En este caso el valor de la variable de control será incrementado al cumplirse un bucle en la cantidad indicada por STEP (2 en el ejemplo). El bucle se repetirá hasta que el valor de la variable de control sea superior al valor límite establecido.

En el ejemplo los valores sucesivos de la variable de control serán 1,3,5,7, y 9. El ordinador abandonará el bucle cuando la variable de control alcance el valor de 11 (9+2).

Es posible estipular un valor negativo para STEP, lo que determinará que el valor de la variable de control disminuya, en lugar de incrementarse. Lógicamente, en este caso el valor incidial debe ser mayor que el valor límite. El bucle será terminado cuando el valor de la variable de control sea menor que el valor límite. Por ejemplo

```
60 FOR a=9 TO 1 STEP-1
```

El valor límite disminuye desde 9 a 1, y el ordenador abandona el bucle cuando el valor de la variable de control llega a 0.

Acomodando un bucle dentro de otro
en inglés esta operación es descripta muy gráficamente como "anidar bucles". Es posible anidar uno o más bucles FOR NEXT, uno dentro del otro. En este caso el orden de las variables de control en las sentencias NEXT deberá ser el inverso del orden de las variables de control en las sentencias FOR. Es posible anidar, o acomodar uno dentro de otro, una cantidad ilimitada de bucles.

Formato
FOR letra=expr-num **TO** expr-num [**STEP** expr-num) **NEXT** letra

FORMAT Formato

Comando utilizado para controlar el procedimiento de almacenaje de información en las Microdrives. Consultar el Manual de las Microdrives y puntos de contacto Interface 1.

GOSUB Dirijase a la subrutina

Posición en el teclado
H

Sentencia/comando

La palabra clave GOSUB causa que el ordenador se desvíe a una subrutina. Las subrutinas son secciones separadas del programa, normalmente compuestas de varias líneas. Este procedimiento es muy útil cuando la misma subrutina es utilizada varias veces durante el mismo programa.

Como emplear GOSUB
GOSUB puede ser utilizado como una sentencia o un comando directo. GOSUB es seguida por un valor numérico. Por ejemplo

```
GOSUB 1000
```

Cuando el ordenador ejecuta el programa, el valor a continuación de GOSUB es redondeado al número entero más próximo, y el ordenador toma el desvío indicado, dirijéndose a la línea del programa con ese número. El empleo de una variable o expresión permite que el ordenador tome el desvío a la subrutina, encaminándose a una línea determinada. Tenga presente que el ordenador se desviará al llegar a la sentencia GOSUB, aún si el número de línea mencionado no se encuentra en el programa. En este caso el ordenador continuará con la primera sentencia que encuentre en esa posición.

La subrutina es terminada con la palabra clave RETURN (retorno). Al encontrarla el ordenador retornará a la sentencia inmediatamente después de la sentencia conteniendo la palabra clave GOSUB que inició el proceso.

Es posible "anidar" subrutinas, unas dentro de otras. En este caso el ordenador es enviado de una subrutina a la otra. En todos los casos, la palabra clave RETURN, que termina la subrutina, siempre le indica al ordenador que debe retornar a la sentencia inmediatamente después de la sentencia conteniendo GOSUB ejecutada en último lugar.

La pila de GOSUB
Siempre que se ejecuta una instrucción GOSUB, su número de línea es almacenado en la pila GOSUB de la memoria del ordenador. Cuando se ejecutan dos o más sentencias GOSUB antes de la palabra clave RETURN, el ordenador almacenará el número de cada una de las líneas del programa conteniendo la palabra clave GOSUB ejecutada. Estos números de línea serán almacenados de forma tal, que el último número se encuentre en el tope de la pila. El efecto de la palabra clave RETURN es indicarle al ordenador que retome al número de línea GOSUB que se encuentra en el tope de la pila. Como resultado, el ordenador retornara a la línea inmediatamente después de la última sentencia conteniendo la palabra clave GOSUB.

Tenga presente que cuando no existen suficientes palabras clave RETURN puede producirse el error 4 (Out of memory, o "sin memoria".

Formato
GOSUB expr número entero

GOTO Dirijase a la línea

Posición en el teclado
G

Sentencia/comando

La palabra clave GOTO le indica al ordenador que se desvíe y se dirija a una línea determinada del programa. (A diferencia de GOSUB que le indica que avance a una subrutina).

Como emplear GOTO
La palabra clave GOTO puede ser utilizada como un comando directo para ejecutar un programa a partir de una línea determinada, *sin borrar previamente* el contenido de la pantalla. También puede ser empleada para formar una sentencia incorporada en un programa. GOTO es seguida por un valor numérico. Por ejemplo

```
60 GOTO 350
```

Cuando la instrucción conteniendo la palabra clave GOTO es ejecutada, el valor a continuación de GOTO es redondeado al número entero más próximo, y el ordenador se desvía, encaminándose a la línea indicada. El empleo de una variable o expresión permite que el programa se desvíe a una línea determinada. Tenga presente que el ordenador se desviará al llegar a la sentencia GOTO, aún si el número de línea mencionado en la sentencia no se encuentra en el programa. En este caso el ordenador continuará con la primera sentencia que encuentre en esa posición.

Formato
GOTO expr número entero

IF Si

Posición en el teclado
U

Sentencia/Comando

La palabra clave IF siempre es utilizada conjuntamente con la palabra clave THEN (entonces) para producir una decisión que determina la acción subsecuente. Antes de tomar la decisión, el ordenador examina ciertas condiciones, para determinar si son verdaderas o falsas. El efecto de estas dos palabras claves es indicarle al ordenador que "si se cumplen ciertas condiciones, entonces haga lo siguiente". Cuando llega a este punto del programa el ordenador se encuentra frente a una encrucijada, y el programa le indica que si cierta condición es verdadera debe tomar un camino, y si no es verdadera, debe tomar el otro camino.

Como emplear IF y THEN
Normalmente IF forma parte de una sentencia con THEN. If es seguido, en primer lugar, por un valor numérico o una condición; en segundo lugar por THEN: y a continuación por una o mas sentencias válidas en BASIC. Por ejemplo

```
80 IF x THEN GOTO 250
240 IF a$="NO" THEN PRINT
    "FIN": STOP
```

Una constante, variable o expresión (por ejemplo x) será considerada como verdadera si tiene un valor diferente a 0. En este caso, la sentencia a continuación de THEN y *la demás sentencias en la misma línea del programa* serán ejecutadas. Después el programa continua a la línea que sigue la instrucción conteniendo la palabra clave THEN. La constante, variable o expresión serán consideradas como falsas cuando su valor es 0. En consecuencia, las sentencias a continuación no serán ejecutadas y el programa continuará a la línea inmediatamente después de la línea inmediatamente después de la línea conteniendo la palabra clave IF. En el ejemplo, el ordenador no continuará a la línea 250 si el valor ex es 0.

Si la condición especificada (a$="NO") a continuación de la palabra clave IF es verdadera, entonces el ordenador ejecutará las sentencias a continuación de THEN. Si la condición es falsa, el ordenador continuará a la próxima línea. En el caso de nuestro ejemplo, si la variable a$ tiene el valor "NO", el ordenador visualizará el texto indicado entre comillas ("FIN") y ejecutará la instrucción "STOP" poniendo fin al programa. En cambio, si a$ tiene cualquier otro valor, el ordenador continuará el programa saltando a la próxima línea.

El Spectrum adjudica a una condición verdadera el valor de 1 y a una condición falsa el valor de 0. Como resultado, el ordenador reconocerá cualquier valor excepto cero como verdadero, y considerará a 0 como falso. Es posible asignar el valor de una condición a una variable mediante una sentencia como la siguiente

```
70 LET x=a$="NO"
```

Recuerde que, a diferencia de lo que sucede en otros dialectos del BASIC, en el Spectrum la palabra clave THEN no puede ser omitida antes de GOTO.

Formato
IF expr-num – sentencia conteniendo **THEN** [;sentencia]
IF condición **THEN** sentencia

IN En

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT I

Función

IN controla el estado del teclado y otros equipos que sirven para introducir información al ordenador (input) y de salida (output). Esta palabra clave lee una byte desde una dirección de conector (port address) determinada, la cual indica el estado del equipo o sistema conectado a ese conector.

Como emplear IN
IN es seguido de un valor numérico, por ejemplo:

```
150 LET x=IN y
```

El valor a continuación de IN puede variar entre 0 y 65535. La función de este valor es indicar la dirección del conector que debe ser leída por el ordenador. IN retorna la lectura de ese conector en bytes.

Direcciones del tacado
El taclado tiene ocho direcciones, cada una de las cuales contiene uno de cinco

bytes diferentes, dependiendo de la tecla oprimida en cada caso. Las direcciones son las siguientes: 65278, 65822, 64510, 63486, 61438, 57342, 49510 y 32766. Los valores de las bytes en esas direcciones pueden ser: 175, 183, 187, 189 o 190.

Formato
IN num-const
IN num-var
IN (expr-num)

INK Tinta

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT X

Sentencia/comando

La palabra clave INK especifica el color de los elementos en primer plano visualizados en la pantalla. Estos elementos incluyen carácteres, puntos, líneas reactas y curvas.

Como emplear INK
INK puede ser empleada como un comando directo, pero es utilizada normalmente como parte de una sentencia en un programa. INK es seguida por un valor numérico, por ejemplo

```
70 INK x
```

El valor numérico a continuación de INK es redondeado al número entero más próximo, y puede variar entre 0 y 9. Los numeros clave de los colores para los elementos en primer plano son los siguientes:

0	Negro
1	Azul
2	Rojo
3	Magenta (morado)
4	verde
5	Azul-verde
6	Amarillo
7	Blanco
8	Transparente
9	Negro o blanco de contraste

La sentencia INK8 le indica al ordenador que el color actual de la tinta, en cualquier posición donde se incluye esta sentencia, debe permanecer incambiado. La sentencia INK9 modifica el color de la tinta a blanco o negro que contraste con el color del fondo o papel.

Colores de tinta (INK) globales o locales
Cuando la sentencia está formada solamente por la palabra INK, como en el ejemplo anterior, el efecto es global (el color es global), y como resultado todas las demás imágenes serán visualizadas en este color de primer plano. La palabra clave INK también puede ser insertada en sentencias controlando la visualización de información en la pantalla que incluyen alguna de las siguientes palabras claves: PRINT, INPUT, PLOT, DRAW y CIRCLE. INK es ubicada a continuación de la palabra clave, pero precede la información o datos y los parámetros controlando la visualización de la información. INK es seguida por los mismos valores y *un punto y coma*. Por ejemplo

```
60 CIRCLE INK 4;128,88,87
```

En este caso el efecto de la palabra clave INK es local y se aplica únicamente a los carácteres, puntos, líneas rectas o curvas, ejecutadas por el ordenador en cumplimiento de las instrucciones contenidas en esa línea de programa. La sentencia en el ejemplo le indica el ordenador que dibuje un círculo de color verde. Una vez completada esta instruccion local, el ordenador retornará al color de tinta global o al color normal (negro).

Formato
INK expr número entero [;]

INKEY$ INput KEY

Posición en el teclado
EXTEND MODE
N

Función

La palabra clave INKEY$ es utilizada para detectar si se ha oprimido alguna de las teclas en el teclado del ordenador.

Como emplear INKEY$
La palabra clave INKEY$ no requiere un argumento y por lo general es utilizada para asignar un carácter determinado a una variable de cadena, con el propósito de controlar un carácter en particular. Por ejemplo

```
70 LET a$=INKEY$
130 IF INKEY$="N" THEN STOP
```

Al ejecutarse la instrucción, INKEY$ retorna el valor del carácter producido por la tecla oprimida en ese memento. Si no se oprime una tecla, entonces INKEY$ retorna una cadena vacía (""). Es important recordar que INKEY$ distingue entre mayúsculas y minúsculas y otros carácteres cambiados (shift). Para detectar cualquier tecla sin tomar en cuenta el carácter especifico producido por ella, se utilizará la palabra clave IN.

A diferencia de la palabra clave INPUT, INKEY$ have avanzar inmediatamente el ordenador a la sentencia siguiente. Por lo tanto, normalmente INKEY$ es ubicada en el seno de un bucle que se repite constantemente hasta que la tecla requerida es oprimida.

Ejemplo
Esta línea suspende la operación hasta que se introduce la letra "y", sin cambio a mayúsculas (CAPS SHIFT) ni cierre de mayúsculas (CAPS LOCK).

```
60 IF INKEY$ <> "y" THEN GOTO 60
```

Formato
INKEY$

INPUT Entrada de datos

Posición en el teclado
I

Sentencia/comando

La palabra clave INPUT permite introducir información o datos (data) en el ordenador durante la ejecución de un programa.

Como emplear INPUT
Normalmente, la palabra clave INPUT forma parte de una sentencia contenida en un programa y se utiliza de manera muy similar a PRINT. En su forma más sencilla, INPUT es seguida por una variable numérica o de cadena, por ejemplo

```
60 INPUT x
90 INPUT a$
```

Al llegar a esta instrucción el ordenador espera hasta que algun número o cadena es introducido en el ordenador. A medida que la información es introducida a través del teclado, el valor o dato es visualizado en la línea inferior de la pantalla. Al oprimirse la tecla ENTER el valor o dato introducido es asignado a la variable indicada y el programa continua.

Una sentencia INPUT puede contener más de una variable y puede visualizar mensajes advirtiendo que es necesario introducir ciertos datos o números. Este efecto se logra exactamente de la misma forma que con la palabra clave PRINT, utilizando comillas para delimitar el mensaje pidiendo la información, incluyendo los puntos y comas, y comas, necesarios para separar los diferentes elementos. Es posible incluir sentencias tales como INK, FLASH y PAPER en la sentencia conteniendo INPUT. Por ejemplo

```
80 INPUT INK2;"Cómo te llamas?";n$,("Cuantos años tienes?,"+n$+"?"); edad
```

Tome nota de las siguientes diferencias entre sentencias conteniendo INPUT y PRINT. INPUT espera cuando llega a una variable, por lo tanto todas las variables y expresiones (por ejemplo la variable incluyendo el signo $ en el ejemplo) que serán incluidas en la pregunta, deben ser *encerradas entre paréntesis*. El ordenador visualiza el mensaje al pie de la pantalla, y las líneas ascienden a medida que se van agregando nuevos elementos. Se puede incluir la palabra clave AT en una sentencia incluyendo INPUT, como en el caso de sentencias con PRINT. La palabra clave AT 0,0 visualiza el mensaje al principio de la línea encima de la última línea en la pantalla. La imagen en la pantalla ascenderá en la pantalla si se visualizan más de dos líneas.

Como interrumpir INPUT
Cuando INPUT es seguida por una variable numérica y se introduce la palabra clave STOP, el programa se detendrá. En el caso de las variables de cadena para interrumpir la operación de la palabra clave INPUT se seguirá el procedimiento siguiente: borrar (delete) el primer par de comillas que aparezca en la pantalla, y a continuación introducir la palabra clave STOP.

Empleando INPUT con LINE
La sentencia INPUT LINE solamente puede ser utilizada con variables de cadena. Normalmente la combinación de INPUT con una variable de cadena causa que el ordenador visualize dos pares de comillas en la pantalla. A medida que la cadena es introducida mediante el teclado en el ordenador, su contenido aparecerá en la pantalla, entre las comillas. Para borrar las comillas, utilize la sentencia INPUT LINE seguida de la variable de cadena. Si se requiere un mensaje al usuario para que introduzca la información, ese mensaje es situado entre las palabras clave INPUT y LINE, por ejemplo

```
70 INPUT "Cómo te llamas?"; LINE n$
```

Formato
INPUT [pregunta][;][,][;]["]
var-num
INPUT [pregunta][;][,][;]"]
var de cadena
INPUT [pregunta][;][,][;"] **LINE**
var de cadena
[pregunta]=[const de cadena]
[(expr de cadena)][**AT** expr número entero, expr número entero]
[sentencia][:][,][;]["]

INT INTeger Número Entero

Posición en el teclado
EXTEND MODE
R

Función

La palabra clave INT transforma números fraccionarios en números enteros suprimiendo la parte fraccionaria.

Como emplear INT
La palabra clave INT es seguida por un valor numérico, por ejemplo

```
70 LET x=INT y
```

Una expresión debe estar *encerrada entre paréntesis*. INT produce el valor del número fraccionario *redondeado en menos* al número entero más proximo.

Ejemplo:
La siguiente sentencia contiene dos números fraccionarios, uno de ellos negativo. Al introducirse el comando PRINT

```
PRINT INT 45.67,INT -77.6
```

el ordenador visualizará lo siguiente en la pantalla

```
45          -8
```

Formato
INT const-num
INT var-num
INT (expr-num)

INVERSE Invertir

Posición en el teclado
EXTEND MODE
M

Sentencia/comando

La palabra clave INVERSE produce el efecto de invertir los colores en las posiciones de los carácteres, con el resultado de que los colores de tinta se toman colores del papel y viceversa.

Como emplear INVERSE
Normalmente INVERSE es utilizado para formar una sentencia contenida en un programa. INVERSE es seguido por un valor numérico. Por ejemplo

```
70 INVERSE 1
```

El valor numérico a continuación de la palabra clave INVERSE es redondeado al número entero más próximo, y puede ser 0 o 1. El efecto de INVERSE 1 es causar que todas las imagenes visualizadas en ejecución de las palabras clave PRINT y OUTPUT sean producidas en colores invertidos. En cambio, INVERSE 0 restituye el color normal del papel y la tinta.

Observe que la palabra clave INVERSE puede ser incluida en el seno de una sentencia controlando la visualización de información en la pantalla, como sucede con la palabra clave INK. INVERSE puede causar algunos problemas cuando se utiliza conjuntamente con CIRCLE, PLOT o DRAW. En este caso el efecto de la sentencia INVERSE 1 será que el punto, linea o circulo sea trazado en el mismo color que el papel o fondo de la imagen, y en consecuencia desaparecerán.

Formato
INVERSE expr número entero

LEN LENgth Largo o extensión

Posición en el teclado
EXTEND MODE
K

Función

La palabra clave indica el largo o extensión de una cadena (string).

Como emplear LEN
LEN es seguida por un valor de cadena, por ejemplo

```
50 LET x=LEN a$
```

Una expresión debe estar *encerrada entre paréntesis*. LEN informa cual es la cantidad de carácteres en la cadena.

Ejemplo
La siguiente línea

```
120 INPUT a$ IF LEN a$>9 THEN GOTO 120
```

pasa solamente cadenas conteniendo más de nueve carácteres.

Formato
LEN const de cadena
LEN var de cadena
LEN (expr de cadena)

LET

Posición en el teclado
L

Sentencia/comando

La palabra clave LET es utilizada para asignar un valor a una variable. En el dialecto de BASIC utilizado por el Spectrum siempre es necesario incluir LET en las sentencias que asignan valores a variables.

Como emplear LET
Usualmente LET forma parte de una sentencia componente de un programa, aunque también puede ser utilizada como un comando directo. Esta palabra clave es seguida por una variable de cadena o una variable numérica, un signo de igual, y a continuación un valor. El valor puede ser numérico o una cadena, dependiende de la naturaleza de la variable precediendo LET. Por ejemplo

```
60 LET x=x+1
80 LET a$="Correcto"
```

Este valor es asignado entonces a la variable indicada (x en el primer ejemplo y a$ en el segundo). Las variables simples (son variables sin subíndices) se mantienen indefinidas hasta que se les asigna un valor mediante sentencias conteniendo las palabras clave LET, READ o INPUT.

Las variables contenidas en matrices (arrays) reciben inicialmente los valores 0 o cadena vacía (""). (Ver DIM).

Formato
LET var-num=expr-num
LET var de cadena=expr de cadena

LIST Hacer una lista

Posición en el teclado
K

Comando/sentencia

Al ejecutar la instrucción LIST el ordenador visualiza en la pantalla una lista del programa contenido en su memoria en ese momento.

Como emplear LIST
Normalmente LIST es utilizado como un comando directo, aunque también puede formar parte de una sentencia en un programa. Cuando se trata de hacer una lista del contenido de un programa completo, bastará con introducir la palabra clave LIST sin mas información. Después del comando directo

LIST

el ordenador visualizará en la pantalla la primera y subsecuentes páginas conteniendo las líneas del programa. Este proceso se cumplirá en páginas. Para desplazar las páginas hacia arriba (scroll) bastará con oprimir cualquier tecla excepto: N, la barra espaciadora, STOP o BREAK.

LIST también puede ser seguido del número de una línea determinada, como un valor numérico, por ejemplo

LIST 100

El valor numérico a continuación de LIST será redondeado al número entero más próximo si fuera necesario. A continuación el ordenador visualizará una lista comenzando a partir de esta línea. Si no existe una línea con ese número, el ordenador comenzará su lista en la próxima línea.

Formato
LIST [expr número entero]

LLIST

Posición en el teclado
EXTEND MODE
V

Comando/sentencia

Esta palabra clave le indica a la impresora Sinclair que imprima una copia de la lista del programa contenido en la memoria del ordenador en ese momento.

Como emplear LLIST
LLIST es utilizado exactamente de la misma forma que LIST (Ver LIST por más detalles). Observe que la información visualizada en la pantalla no cambiará mientras la lista está siendo imprimida.

Formato
LLIST [expr número entero]

LN Logarithm (Natural) Logaritmo (Natural)

Posición en el teclado
EXTEND MODE
Z

Función

LN suministra el logaritmo natural (en base e) de un valor especificado. Actúa como el inverso de EXP.

Como emplear LN
La palabra clave LN es seguida por un valor numérico. Por ejemplo

60 LET x=LN y

Una expresión debe estar *encerrada entre paréntesis*. El valor numérico a continuación de LN debe ser mayor que 0. LN produce el logaritmo natural de este valor.

Formato
LN const-num
LN var-num
LN (expr-num)

LOAD Cargar

Posición en el teclado
J

Comando/sentencia

La palabra clave LOAD le indica al ordenador que cargue un programa completo almacenado en cinta, depositándolo en su memoria.

Como emplear LOAD
Normalmente LOAD es utilizado como un comando directo, pero también puede formar parte de un programa, para cargar otro programa.

La palabra clave LOAD es seguida del nombre del programa a ser cargado. Los nombres de los programas son un valor de cadena con una extensión máxima de diez carácteres. Por ejemplo

LOAD "nombre"

Los efectos de LOAD son drásticos: al ejecutarse esta instrucción el ordenador borra el programa corrientemente en la memoria del ordenador, así como todos los valores de sus variables. A continuación el Spectrum busca el nombre del nuevo programa en la cinta y lo carga.

Recuerde que en el caso de los nombres de programas el ordenador distingue entre letras mayúsculas y minúsculas.

Cuando LOAD es seguido por una cadena vacía, como en el ejemplo:

LOAD""

el Spectrum cargará el primer programa completo que encuentre.

Observe que LOAD es utilizado en forma diferente cuando el ordenador ha sido conectado a una unidad Microdrive. Ver el Manual de la Microdrive y el punto de contacto Interface 1, por detalles adicionales.

Formato
LOAD expr de cadena

LOAD CODE

Posición en el teclado
EXTEND MODE
J

Comando/sentencia

La palabra clave LOAD CODE es empleada para indicarle al ordenador que cargue en su memoria cierta información, que ha sido previamente almacenada en cinta.

La información consiste en una serie de bytes, las cuales serán enviadas a diversas direcciones en la memoria del ordenador.

Es posible hacer uso de LOAD CODE, por ejemplo, para cargar una imagen gráfica, u otro tipo de información para gráficas definidas por el usuario.

Como emplear LOAD CODE
LOAD CODE puede ser utilizado como un comando directo o, alternativamente, como parte de una sentencia en un programa.

La sintaxis de esta palabra clave es la siguiente: en primer lugar LOAD, luego el nombre del programa entre comillas, finalmente CODE.
Por ejemplo

LOAD "datos" CODE

El nombre de programa a continuación de LOAD es el nombre correspondiente a la información registrada en cinta que se desea cargar al ordenador. A este nombre se le aplican las mismas limitaciones que en el caso de la palabra clave LOAD (Ver: LOAD).

Cuando el ordenador llega a esta sentencia en el programa – o cuando recibe el comando directo – procede a buscar en la cinta la información almacenada bajo ese nombre. Cuando el ordenador encuentra la información, visualizará los bytes, seguidos por el nombre de ese grupo de datos.

A continación el Spectrum carga los bytes en su memoria distribuyéndolos en las direcciones a partir de las cuales fueron registradas previamente. La nueva información cargada en ejecución de este comando es superpuesta sobre toda otra información anterior en la memoria del ordenador.

CODE también puede ser seguido por uno o dos valores numéricos, separados por una coma

LOAD "dibujo" **CODE** 16384,6912

Los valores numéricos incluidos a continuación de CODE serán redondeados al número entero más cercano si fuera necesario. Su propósito es definir la dirección inicial desde donde se cargará la información indicada en la sentencia (16384 en el ejemplo), y la cantidad de bytes que se enviarán a las ubicaciones, comenzando a partir de esa dirección.

Si el número indicado no es el correcto, el ordenador visualizará el mensaje de error de carga de la cinta (tape error).

Cuando LOAD es seguido por solamente un valor numérico, el ordenador interpreta que se trata del número definiendo la posición de la dirección inicial a partir de la cual deberán ubicarse todos los bytes.

El resultado del ejemplo más arriba también puede ser obtenido mediante las palabras claves LOADSCREEN$.

Por detalles acerca del procedimiento empleado para almacenar bytes, ver la sección SAVE CODE.

Formato
LOAD expr de cadena **CODE** [expr entero] [,expr número entero]

LOAD DATA

Posición en el teclado
EXTEND MODE
D

Sentencia/comando

La palabra clave LOAD DATA es empleada para cargar matrices (arrays) almacenadas en cinta. Las matrices son registradas en cinta mediante una instrucción SAVE DATA.

Como emplear LOAD DATA
Es posible emplear las palabras claves LOAD DATA como parte de una sentencia en un programa o como un comando directo. El orden seguido es el siguiente: a continuación de LOAD se incluye el nombre que identifica la información registrada en la cinta (un valor de cadena). A continuación se incluye la palabra clave DATA y una letra o una letra y un signo de $. El último elemento de la sentencia es un par de paréntesis vacíos. Por ejemplo

270 **LOAD** "números" **DATA** n()
300 **LOAD** "nombres" **DATA** n$()

El nombre incluido a continuación de LOAD, es el nombre de la matriz registrada en la cinta. Este nombre se encuentra sujeto a las mismas restricciones que los demás nombres de programas cuando se utiliza la palabra clave LOAD. La letra o letra$ insertada a continuación de DATA es la letra nombre seleccionada para designar la matriz en el programa contenido en la memoria del ordenador, y que hará uso de aquella matriz.

Al ejecutarse esta instrucción, el Spectrum busca la matriz indicada en la cinta. Cuando la encuentra, el ordenador visualiza el mensaje "Number array:" o "Character array:" en la pantalla, seguido del nombre. A continuación el ordenador carga la matriz en su memoria.

Cuando el ordenador carga la matriz registrada en cinta, la deposita en su memoria; el ordenador borrará cualquier matriz designada con la misma letra nombre preexistente en su memoria (en el ejemplo los nombres de las matrices son "n" y "n$"); finalmente el ordenador formará una nueva matriz designada con la letra de nombre especificada.

Formato
LOAD expr de cadena **DATA**
letra [$]()

LOAD SCREEN$

Posición en el teclado
J
EXTEND MODE
SYMBOL SHIFT
K

Sentencia/comando

LOAD SCREEN$ le indica al ordenador que cargue la información referente a una imagen de la pantalla de la televisión almacenada en cinta.

Como resultado de esta instrucción, el ordenador tomará la información de la cinta y la enviará a las secciones de su memoria controlando la visualización de imágenes en la pantalla de la televisión, para generar la imagen.

Como emplear LOAD SCREEN
LOAD SCREEN$ puede ser integrado en una sentencia incluida en un programa o puede ser utilizado como un comando directo.

LOAD es seguido del nombre bajo el cual la información fué registrada en la cinta. Este nombre es un valor de cadena. Finalmente se agrega SCREEN$. Por ejemplo

LOAD "dibujo" **SCREEN$**

El nombre entre paréntesis a continuación de LOAD es el nombre que identifica la información almacenada en la cinta, y se encuentra sometido a las mismas restricciones que los nombres de programas cuando se utiliza la palabra clave LOAD.

La imagen se forma gradualmente en la pantalla con los colores de tinta y papel activos en ese momento y luego toma los colores definidos como atributos.

Por más detalles acerca del procedimiento para almacenar información contenida en la pantalla, ver SAVE SCREEN$.

Formato
LOAD expr de cadena **SCREEN$**

LPRINT Line printer PRINT

Posición en el teclado
EXTEND MODE
C

Sentencia/comando

La palabra clave LPRINT le indica a la impresora Sinclair tipo L que imprima una determinada información, de la misma forma que la palabra clave PRINT le indica al ordenador que visualize una determinada información en la pantalla de la televisión.

Como emplear LPRINT
LPRINT puede ser incluido como parte de una sentencia contenida en un programa, o puede ser utilizado como un comando directo. Es seguido por elementos de información o datos (data) que pueden ser separados por puntos y comas, comas o apóstrofes, por ejemplo

60 LPRINT "Número";x
"Nombre";n$,"Edad" ;a

Al ejecutar esta instrucción, el ordenador transfiere la información indicada la impresora, la cual lo imprime. El formato será el mismo que hubiera sido utilizado para visualizar la información en la pantalla en ejecución de la instrucción PRINT.

Una instrucción conteniendo LPRINT también puede incluir sentencias TAB, algunos controles CHR$, sentencias INVERSE y OVER, y códigos de control, con los mismos resultados que cuando se utiliza PRINT.

Formato
LPRINT [**TAB** expr número entero;)] [**AT** expr número entero,expr número entero;] [**CHR$**(expr número entero);] [sentencia,][expr número entero][expr de cadena][;][,][']

MERGE Fundir o combinar

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT
T

Sentencia/comando

La palabra clave MERGE permite fusionar dos programas.

Como emplear MERGE
MERGE puede ser incluida en una sentencia parte de un programa, o como un comando directo.

MERGE es seguida por el nombre de la información registrada en cinta como un valor de cadena, por ejemplo

500 MERGE "prog2"

El nombre a continuación, de MERGE ("prog2" en el ejemplo) es el nombre del programa almacenado en cinta que se desea combinar con el programa que ya se encuentra en la memoria del ordenador.

Este nombre se encuentra sometido a las mismas limitaciones que los nombres de programas utilizados en sentencias con las palabras claves LOAD.

El ordenador ejecuta la instrucción MERGE, cargando el programa almacenado en cinta sin borrar el otro programa en su memoria.

MERGE expr de cadena

MOVE

Comando utilizado para controlar el

almacenamiento de información en las Microdrives. Por más información consultar el Manual de las Microdrives y el punto de contacto Interface I.

NEW

Posición en el teclado
A

Comando/sentencia

Cuando se introduce la palabra clave NEW el ordenador borra el contenido del área de la memoria en BASIC (el área hasta el límite RAMTOP), removiendo todos los programas que se encuentran en ese sector de su memoria.

Como emplear NEW
Por lo general NEW es utilizado como un comando directo, pero también puede formar parte de una sentencia en un programa. Es utilizado independientemente.

Al ejecutar esta instrucción, el ordenador borra el programa y variables en su memoria hasta el límite RAMTOP. Como resultado, las gráficas definidas por el usuario conservadas en un área por encima del límite RAMTOP no son afectadas por NEW.

Formato
NEW

NEXT Siguiente

Posición en el teclado
N

Sentencia/comando

La palabra clave NEXT siempre es utilizada en conjunción con FOR para construir un bucle FOR NEXT.

Como emplear NEXT
NEXT se utiliza normalmente en una sentencia contenida en un programa y como parte de un bucle FOR NEXT. Es seguida por una letra. Esta última es la variable de control del bucle.
Por ejemplo

```
90 NEXT a
```

En el dialecto de BASIC utilizado por Sinclair, la variable de control debe ser especificada.

Por más detalles acerca de los bucles (loops) FOR NEXT ver la sección sobre FOR NEXT.

Formato
NEXT letra

NOT No

Posición en el teclado
SYMBOL SHIFT
S

Operador lógico/función

La palabra clave NOT es empleada para invertir la veracidad de una condición, causando que una condición falsa se torne en verdadera y viceversa.

Como emplear NOT
NOT es seguida por una condición o un valor numérico. Dos ejemplos:

```
90 IF NOT x=y THEN PRINT "Error"
90 LET correcto=x=x+z:
   IF NOT correcto THEN PRINT
   "Error"
```

Cuando NOT es seguida por una condición (x=y+z en el ejemplo), el Spectrum opera de la siguiente forma: primero asigna un valor de 1 a la condición si es verdadera y Ø si es falsa. NOT opera entonces como una función, invirtiendo el valor producido, con el resultado de que es posible examiner la validez del reverso de la condición.

Observe que si la condición contiene las palabras clave AND u OR debe *estar encerrada entre paréntesis*.

Cuando NOT es seguida por un valor numérico, retornará un Ø si el valor numérico que la sigue no es Ø, y un número 1 si el valor que la sigue es Ø. Por lo tanto, en los ejemplos anteriores: el Spectrum visualizará "Error" si x<>y+z, o si *correcto* tiene un valor de Ø.

Formato
NOT cond
NOT expr-num

OPEN

Comando utilizado para controlar el almacenamiento de información en las Microdrives. Por más información consultar el Manual de las Microdrives y el punto de contacto Interface 1.

OR

Posición en el teclado
SYMBOL SHIFT U

Operador lógico/función

OR actua como un operador lógico para establecer la veracidad de una combinación de condiciones. Si una o más de las condiciones son verdaderas, entonces la condición total es verdadera. OR tambien opera como una función para ejecutar operaciones binarias envolviendo dos valores numéricos.

Como emplear OR
Como un operador lógico. OR enlaza dos condiciones en una sentencia donde se trata de establecer la veracidad o falsedad del conjunto total. Por ejemplo:

```
70 IF INKEY$="N" OR
   INKEY$="n" THEN STOP
```

Si cualquiera de la dos condiciones indicadas en la sentencia es verdadera, entonces el ordenador llega a la conclusión de que la totalidad de la sentencia es verdadera y obedece la instrucción (THEN STOP).

En el ejemplo una de las condiciones (INKEY$="N" o INKEY$="n") se transforma en verdadera tan pronto como se oprime la tecla "N" en el teclado del ordenador, indiferentemente de que se trate de la "N" mayúscula o minúscula. En los dos casos (mayúscula o minúscula) la totalidad de la combinación en la sentencia se ha transformado en verdadera y el programa se detiene obedeciendo la instrucción THEN STOP.

OR como una función
El ZX Spectrum + asigna un valor número de 1 a la condición verdadera y de Ø a la condición falsa. La palabra clave reconoce cualquier valor excepto Ø como verdadero y Ø como falso. Por lo tanto, OR puede ser precedido o seguido por un valor numérico, por ejemplo

```
40 LET x=y OR z
```

En el ejemplo, el ordenador asignará un valor de 1 a la variable x si el valor de z es cualquier número excepto Ø. Alternativamente, si el valor de z es Ø o una condición falsa, el ordenador le asignará un valor y.

Recuerde que el Spectrum no evalúa combinaciones de valores numéricos de acuerdo con las tablas de veracidad estandard.

Formato
cond OR con
expr-num OR expr-num.

OUT Salida

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT O

Sentencia/comando

La palabra clave OUT envía un byte a la dirección correspondiente a un "port" o punto de conexión de entrada/salida del ordenador.

Como emplear OUT
OUT puede ser empleada para formar una sentencia contenida en un programa o como un comando directo independiente. OUT es seguida por dos valores numéricos, *separados por una coma*.
Por ejemplo

```
40 OUT 254,3
```

Ambos valores son redondeados al número entero más proximo. El primer valor (254 en el ejemplo) puede variar entre Ø y 65535 y la dirección de la "port" o punto de conexión. El segundo valor (3 en el ejemplo) puede variar entre Ø y 255, siendo el byte enviado a la dirección especificada anteriormente.

Como se explicó, cada byte se compone de una serie de bits. En el ejemplo los bits Ø–2 del byte enviado a la dirección de punto de contacto (port address) 254 definen el color de borde o border. En el ejemplo, este instrucción cambiará el color del borde a magenta. El bit número 3 enviado a la misma dirección, controla el enchufe de MIC, y el bit 4 controla el altavoz.

La dirección de "port", o punto de contacto, 251 controla la impresora, y las direcciones de punto de contacto 254, 247, y 239 son utilizadas para controlar otros equipos periféricos conectados al ordenador.

Formato
OUT expr número entero, expr número entero

OVER Sobreimpresión

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT N

Sentencia/comando

La palabra clave OVER es utilizada para sobreimprimir un carácter encima de otro. También puede ser empleada para dibujar puntos o trazar líneas rectas o curvas de color de papel (paper) en lugar de color de tinta (ink).

Como emplear OVER
Por lo general OVER es utilizada para formar una sentencia parte de un programa. Es seguida por un valor numérico. Por ejemplo

```
80 OVER 1
```

El valor numérico a continuación de la palabra clave OVER es redondeado al número entero más próximo, y puede ser Ø o 1. El valor normal es OVER Ø, y determina que un nuevo carácter impreso en una posición de carácter borre completamente lo que había antes en la misma posición, tomando su lugar. En cambio el valor OVER 1 pone en operación un efecto especial: el nuevo carácter se agregará al carácter anterior en la misma posición, combinándose con él.

OVER puede ser insertado en una sentencia conteniendo las palabras claves PRINT o INPUT, como sucede con INK. En este caso su efecto será local, OVER afectará únicamente los caracteres visualizados en ejecución de esa sentencia. Por ejemplo, la sentencia siguiente produce el efecto de subrayar una palabra

```
60 PRINT AT 11,15; "Sí";
   OVER 1; AT 11,15;"  "
```

Sin embargo, observe que los carácteres se combinan de tal forma que, en el sector donde los dos carácteres se sobreponen, el ordenador visualiza el color del papel (paper).

OVER en el modo de alta resolución
OVER puede ser utilizado conjuntamente con PLOT, DRAW, y CIRCLE. Cuando no se incluye la palabra clave OVER las líneas rectas y curvas pueden sobreponerse las unas a las otras, pero deben ser del mismo color de tinta, para evitar que cambie de color de la totalidad de la posición donde ambas líneas se cruzan. En cambio, si se incluye la palabra clave OVER 1, cuando dos líneas rectas o curvas se cruzan, o cuando se encuentran con algún otro carácter, se visualizará el color del papel. Otro efecto interesante es que si se dibujan puntos, o se trazan dos líneas rectas o curvas, exactamente en las mismas posiciones, en presencia de la instrucción OVER 1, los elementos sobrepuestos desaparecerán.

Formato
OVER expr número entero

PAPER Papel

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT C

Sentencia/comando

PAPER es utilizado para seleccionar el color del papel o fondo de la pantalla. Este color puede ser el de la totalidad del fondo, o solamente del fondo de ciertos carácteres, puntos o líneas visualizados en las posiciones de carácteres individuales.

Como emplear PAPER
La palabra clave PAPER puede ser utilizada para formar una sentencia incluida en un programa o como un comando directo. PAPER es seguido por un valor numérico, por ejemplo

```
80 PAPER x
```

Este valor numérico es redondeado al número entero más próximo, y puede variar entre Ø y 9. Los números código para los colores de PAPER son similares a los utilizados con la palabra clave INK (ver la sección correspondiente). Como en el caso de INK, los colores pueden ser globales o locales. Para hacerlos locales, la palabra clave PAPER es insertada en sentencias controlando la visualización de imágenes en la pantalla, exactamente de la misma manera que con los colores de tinta. (Ver la sección INK).

Cuando se visualizan carácteres en la pantalla a continuación de una sentencia incluyendo la palabra clave PAPER, ya sea que la instrucción tenga un efecto global o local, el color del fondo de la posición del carácter correspondiente cambiará al color seleccionado. Esto se aplica también cuando se dibujan puntos, o se trazan líneas rectas o curvas, mediante instrucciones conteniendo una sentencia de PAPER (efecto local), pero no a continuación de una sentencia o comando de efecto global.

Para producir un fondo de color que abarque la totalidad del área de visualización en la pantalla, es necesario utilizar la palabra clave CLS después de una sentencia conteniendo PAPER. Como resultado toda la imagen en la pantalla se toma del color indicado, el cual permanece como el color de fondo.

Formato
PAPER expr número entero [;]

PAUSE Pausa

Posición en el teclado
M

Sentencia/comando

El efecto de la palabra clave PAUSE es suspender la ejecución del programa por un período de tiempo definido o indefinido.

Como emplear PAUSE
La palabra clave PAUSE es utilizada normalmente como parte de una sentencia en un programa. Es seguida de un valor numérico, por ejemplo

```
130 PAUSE 100
```

El valor numérico a continuación de PAUSE es redondeado al número entero más próximo. El número puede variar entre Ø y 65535 y define la duración de la pausa en términos de imágenes de televisión (50 imágenes por segundo en Europa o 60 en los Estados Unidos). Por ejemplo, un valor de 50 produce una pausa de un segundo en Europa y en otros países donde la frecuencia de las imágenes es de 50Hz.

Sin embargo, la pausa puede ser interrumpida presionando una tecla. La instrucción PAUSE Ø crea una pausa de duración ilimitada que continua hasta que es interrumpida oprimiendo una tecla.

Formato
PAUSE expr número entero

PEEK Mirar

Posición en el teclado
EXTEND MODE
O

Función

PEEK suministra el valor del byte almacenado en una dirección particular de la memoria del ordenador.

Como emplear PEEK
La palabra clave PEEK es seguida por un valor numérico, por ejemplo

```
80 LET x=PEEK (256*y)
```

Observe que la expresión debe ser *encerrada entre paréntesis*. El valor numérico a continuación de PEEK es redondeado al número entero más próximo si fuera necesario. El número puede variar entre Ø y 65535 e identifica una determinada dirección en la memoria. Como resultado de esta instrucción el ordenador suministra el valor del byte (un número entre Ø y 255) en la dirección especificada.

Ejemplo
La cantidad de imágenes (frames) de televisión ocurridas desde el momento en que se encendió el Spectrum es almacenada en las direcciones 23672 a 23674. Como esas imágenes son producidas a una velocidad constante, explorar estas ubicaciones de la memoria mediante una sentencia conteniendo la palabra clave PEEK, ofrece un método preciso para medir el tiempo. En el siguiente ejemplo, la línea de programa le indica al ordenador que indique cuanto tiempo ha transcurrido desde que el Spectrum fué encendido (medido en segundos y

```
10 PRINT (PEEK 23672+256*
   PEEK 23673+65536*PEEK
   23674)/50
```

NOTA: *Cuando la frecuencia de la electricidad es 60 Hz, el valor 50 debe ser cambiado a 60.*

Formato
PEEK const número entero
PEEK var número entero
PEEK (expres número entero)

PI π

Posición en el teclado
EXTEND MODE
M

Función

PI suministra el valor de pi (π) para ser utilizado en cálculos.

Como emplear PI
Esta palabra clave no requiere valores o variables complementarias cuando utilizada en una sentencia o comando, por ejemplo

DRAW 25,Ø,–PI

PI suministra un valor de 3.1415927, y como resultado el comando en el ejemplo le indicará al ordenador que trace un semicírculo de gran tamaño en la pantalla.

Formato
PI

PLOT Dibujar

Posición en el teclado
O

Sentencia/comando

PLOT es utilizado len la producción de gráficas de alta resolución, para dibujar un pixel o punto de color, en una posición específica de la pantalla.

Como emplear PLOT
La palabra clave PLOT puede formar parte de una sentencia incluida en un programa o también puede ser utilizada como un comando. Normalmente es seguida por dos valores numéricos *separados por una coma*, por ejemplo

50 **PLOT** 128,87

Los dos valores numéricos después de PLOT serán redondeados al número entero más próximo si fuera necesario. El primer valor puede variar en Ø y 255, su función es definir la coordenada horizontal de la posición en la pantalla. El segundo valor puede variar entre Ø y 175, su función es definir la coordenada vertical de la posición. Normalmente el ordenador ejecuta las instrucciones dibuando un pixel en el punto indicado, utilizando el color de tinta efectivo en ese momento.

Tome nota de los siguientes efectos producidos sobre la instrucción PLOT por las sentencias o comandos referentes a los colores. Después de OVER 1, el color de un punto existente en la misma posición es cambiado al color del papel (paper). A continuación de INVERSE 1, el punto es dibujado en el mismo color del papel en ese momento. Después de BRIGHT 1 o FLASH 1, toda la posición del carácter en la pantalla de baja resolución, donde se dibuja el pixel será de color brillante o "relampagueara" (flash).

Estas cuatro palabras claves e INK también pueden ser insertadas en una sentencia conteniendo la palabra clave PLOT. El procedimiento a seguir es el mismo que en el caso de la palabra clave PRINT. Por ejemplo

160 **PLOT** INK 2;x,y

El efecto de estas palabras claves es el *mismo* que antes, con una diferencia importante: será un efecto limitado al pixel dibujado en ejecución de la sentencia. Cuando se inserta la palabra clave PAPER en una sentencia conteniendo PLOT, el efecto de la instrucción será cambiar el color de papel de toda la posición del carácter donde se encuentra el pixel.

Formato
PLOT [sentencia;] expr número entero, expr número entero

POINT Punto

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT 8

Función

La palabra clave POINT es empleada para determinar si el color de una posición determinada en la pantalla de alta resolución es un color de tinta o de papel.

Como emplear POINT
POINT es seguido por dos valores *numéricos separados por una coma y encerrados entre paréntesis.*
Por ejemplo

248 **IF POINT** (x,y)=1 **THEN**
GOSUB 600

Los dos valores después de POINT son redondeados al número entero más próximo si fuera necesario. La función del primer valor es definir la coordenada horizontal de un pixel en la pantalla y puede variar entre Ø y 255. El segundo valor define la coordenada vertical y puede variar entre Ø y 175. Al recibir la instrucción POINT, el ordenador responderá con un número 1, si el pixel en la posición indicada es de color de tinta y con un Ø si es de color de papel.

Formato
POINT (expr número entero, expr número entero)

POKE Introducir

Posición en el tablero
O

Sentencia/comando

La palabra clave POKE es utilizada para cambiar el valor de un byte situado en una dirección específica de la memoria del ordenador.

Como emplear POKE
POKE puede formar parte de una sentencia incluida en un programa o también puede operar como un comando. Es seguido por dos valores numéricos *separados por una coma.*
Por ejemplo

POKE 23609,255

Los dos valores a continuación de POKE son redondeados al número entero más próximo si fuera necesario. El primer número es una dirección en la memoria RAM del ordenador, pudiendo variar entre 16384 y 65535. El segundo valor puede variar entre Ø y 255, siendo la byte cuyo valor se desea modificar.

Formato
POKE expr número entero, expr número entero

PRINT Imprimir

Posición en el teclado
P

Sentencia/comando

La sentencia conteniendo PRINT puede incorporar otras palabras clave para definir la posición y color de la información visualizada.

Como emplear PRINT
PRINT puede ser empleada independientemente o puede ser seguida por datos. Estos datos pueden ser cualquier tipo de expresión numérica o de cadena, o una combinación de ambos tipos.

Cuando se emplea PRINT con datos, dos o más elementos deben ser separados por un punto y coma, una coma o un apóstrofe.

También es posible insertar ciertas palabras claves entre PRINT y los datos, en cualquier orden; siempre y cuando la sentencia formada por aquella palabra clave *termine en un punto y coma.* Estas palabras claves son las siguientes: CHR$, TAB, AT, INK, PAPER, FLASH, BRIGHT, INVERSE y OVER.

Empleando PRINT con cadenas
El efecto de la palabra clave PRINT sola, o seguida por una cadena vacía (""), será visualizar, "imprimir", una línea blanca en la pantalla y desplazar el cursor al principio de la línea siguiente.

PRINT seguida por una constante de cadena tendrá el efecto de visualizar en la pantalla los carácteres entre las comillas. El comando

PRINT "3/542/76/21"

producirá el resultado siguiente en la pantalla:

3/542/76/21

PRINT seguida por una variable o expresión de cadena visualizará o "imprimirá" en la pantalla la cadena.

PRINT con números
PRINT seguido por cualquier variable numérica visualizará en la pantalla el valor de la expresión. Los números son notación decimal con hasta ocho dígitos significativos de un número.

Los números muy grandes y muy pequeños son visualizados en notación científica, como dos figuras separadas por la letra E. Esta indica un número en el cual la primera parte (la mantisa) es elevado a la potencia de la segunda parte (el exponente). El comando

PRINT 3/542/76/21

por ejemplo, producirá el siguiente resultado en la pantalla

3.4680798E-6

Impresos con PRINT mediante los signos de puntuación.
La información "impresa" en la pantalla, en ejecución de una instrucción PRINT. PRINT seguido por elementos de información, datos, separados por un punto y coma, visualiza dichos elementos uno junto al otro. El comando

PRINT 1;2;3

producirá

123

PRINT es seguida por elementos de información separados por comas, el ordenador visualizará cada uno de esos elementos en el centro o al principio de una línea en la pantalla, dependiendo de la posición del primer elemento de información. El comando

PRINT 1,2,3

producirá el siguiente resultado:

1 2
3

PRINT seguida por elementos de información separados por apóstrofes, visualizará cada uno de dichos elementos al principio de la línea siguiente. El apóstrofe opera como un "punto y aparte". El comando

PRINT 1'2'3

producirá el siguiente resultado

1
2
3

PRINT termina en un punto y coma, coma o apóstrofe, la información visualizada por el ordenador ejecutando la próxima sentencia PRINT será afectada de la misma forma.

PRINT y otras palabras claves.
PRINT puede ser seguida por TAB, un valor numérico, punto y coma y un elemento de información, por ejemplo

60 **PRINT TAB** x;a$

El valor a continuación de TAB (x en el ejemplo) será redondeado al numero entero más próximo si fuera necesario. A continuación ese valor será dividido por 32 y se lleva el resto para producir un valor entre Ø y 31.

PRINT puede ser seguido por la palabra clave AT y dos valores *numéricos separados por una coma, un punto y coma y un elemento de información.* Por ejemplo

50 **PRINT AT** l,c; "Informacion"

El primer valor (la variable l en el ejemplo) define la linea horizontal en la pantalla donde se "imprimirá" o visualizará la información. Este valor puede variar entre Ø y 21. El segundo valor, la variable c en el ejemplo, define el número de la columna donde se imprimirá el primer carácter o dígito de la información. El número de columna puede variar entre Ø y 31. El comando PRINT AT 11,16;"*" visualize una estrella en el centro de la pantalla.

PRINT también puede ser seguida por una o más funciones CHR$. Por más detalles consulte la sección sobre CHR$.

PRINT y las palabras claves controlando los colores.
PRINT en la pantalla de la televisión son afectadas por las sentencias y comandos referentes al color y conteniendo la palabra clave INK, PAPER, FLASH, BRIGHT, INVERSE y OVER que se encontraban incluidas en las líneas anteriores del programa. Sentencia conteniendo PRINT y terminada en un punto y coma; palabra clave seguida por un punto y coma; y a continuación el elemento de información, por ejemplo

50 **PRINT AT** 11,16; **INK** 2;
FLASH 1; "*"

Después de haber ejecutado la sentencia, el ordenador retornará a los valores normales o los valores globales preestablecidos. PRINT también obedecerá los códigos de control del color locales insertados conjuntamente con la información (ver la página 33).

Formato
PRINT [**TAB**expr número entero;]
[**AT** expr número entero, expr número entero]
[**CHR$** (expr número entero);]
[sentencia;][expr numérica]
[expr de cadena][;][,]

RANDOMIZE (RAND)

Posición en el teclado
T

Sentencia/comando

La palabra clave RANDOMIZE es abreviada en el teclado como RAND, y es utilizada en conjuncion con RND (sin embargo no debe confundirse la una con la otra) para generar una secuencia de números aleatorios (al azar) y que, por lo tanto, no pueden ser predecidos, o una secuencia de números predecibles.

Como emplear RANDOMIZE
La palabra clave RANDOMIZE es utilizada dentro de una sentencia incluida en un programa o como un comando. Esta palabra clave puede ser utilizada independientemente o seguida de un valor numérico, por ejemplo

RANDOMIZE 1
10 **RANDOMIZE**

El valor numérico a continuación de RANDOMIZE será redondeado al número entero más próximo si fuera necesario. El valor numérico puede variar entre Ø y 65535. Un valor superior a Ø ajusta el valor de la variable de sistema SEED a este valor. Como resultado la palabra clave RND siempre generará la misma secuencia de números (por información referente a variables de sistema ver página 48). La secuencia de los números dependerá del valor de RANDOMIZE.

En cambio, cuando RANDOMIZE es seguida por Ø o ningún valor, entonces la variable de sistema SEED recibirá el valor de otra variable de sistema denominado FRAMES. Esta variable FRAMES cuenta la cantidad de imágenes visualizadas por la televisión desde el momento cuando se encendió el Spectrum. SEED se incrementa cada 50 o 60 imágenes – dependiendo del tipo de televisión (Ver sección sobre PAUSE). Debido a que SEED cambia cada 50 o 60 segundos, la secuencia de números generados por una sentencia conteniendo RANDOMIZE y a continuación RND o solamente RANDOMIZE Ø, ofrecerá un efecto sumamente aleatorio.

Formato
RANDOMIZE [expr número entero]

READ Leer

Posición en el teclado
EXTEND MODE
A

Sentencia/comando

La palabra clave READ es utilizada en combinación con la palabra clave DATA, y tiene el propósito de asignar valores a variables, haciendo uso de los valores especificados en sentencia de DATA.

Como emplear READ
Normalmente READ es utilizada para formar una sentencia contenida en un programa.

Es seguida por una o más variables numéricas o variables de cadena, *separadas entre si por comas.* Por ejemplo

20 **READ** a$,x

Cuando la instrucción conteniendo la palabra clave READ se ejecuta por primera vez, el ordenador se dirigirá a la línea conteniendo la palabra clave DATA y tomará la misma cantidad de valores que variables, comenzando a partir del primer elemento en la lista de valores a continuación de DATA. Los valores contenidos en la lista de DATA son atribuidos a las variables en la sentencia READ sucesivamente y siguiendo su orden en la sentencia.

Formato
READ var-num [,var-num]
[var de cadena]
READ var de cadena [,var-num]
[,var de cadena]

REM REMark Comentario

Posición en el teclado
E

Sentencia

REM tiene el propósito de indicarle al

ordenador que la información a continuación es solamente un comentario u observación. Esta puede consistir, por ejemplo, en el título y nombre del autor del programa, explicaciones acerca de las líneas siguientes en el programa, por ejemplo el propósito de una variable. La sentencia REM es incluida en el programa para la información del usuario, pero no interviene en la operación del programa. La sentencia REM no es visualizada en la pantalla excepto cuando se hace un listado del programa (LIST).

Como emplear REM
REM forma parte de una línea independiente del programa o también puede ser incluida como el elemento final de otra línea. La palabra clave REM es seguida por los comentarios, y no requiere comillas, por ejemplo

```
10 REM El autor de este programa
     es Pedro
20 INPUT n$: REM n$ es el nombre
```

Cuando el ordenador llega a una línea encabezada por la palabra clave REM ignorará el resto de la información en la misma línea.

Formato
REM [cualquier carácter]

RESTORE Restaurar

Posición en el teclado
EXTEND MODE
S

Sentencia/comando

La palabra clave RESTORE es utilizada junto con READ y DATA para modificar el mecanismo normal de READ DATA. Su función es indicarle al ordenador que, al ejecutar la instrucción READ, tome los valores de una determinada sentencia conteniendo la palabra clave DATA, en lugar de seguir el procedimiento normal, de dirigirse a la primera sentencia de DATA en el programa (ver DATA y READ).

Como emplear RESTORE
Por lo general RESTORE es incluida dentro de una sentencia parte de un programa. El programador tiene la opción de agregar a continuación de ella algun valor numérico, por ejemplo

```
160 RESTORE 800
```

El valor numérico a continuación de RESTORE es redondeado al número entero más próximo si fuera necesario. El ordenador asume que el valor numérico es el número de la línea de programa conteniendo una sentencia de DATA. Después de haber encontrado la palabra clave RESTORE, el ordenador procederá de la manera siguiente: al encontrar una línea de programa con la palabra clave READ, se encaminará a la línea 800 del programa y tomará los valores de DATA especificados en ella, para asignarlos a las variables en aquella línea. El procedimiento mediante el cual un valor es adjudicado a una variable es similar al utilizado con READ DATA (ver las secciones DATA y READ).

Si la línea de DATA (línea 800 en el ejemplo) no existe o no contiene una palabra clave DATA, entonces el ordenador continuará hasta la próxima sentencia de DATA.

Cuando RESTORE es seguida por el número 0, o si no es seguida por un valor numérico, entonces el ordenador, al encontrar una sentencia con la palabra clave READ avanzará hasta la próxima línea conteniendo la palabra clave DATA.

Formato
RESTORE [expr número entero]

RETURN Regresar

Posición en el teclado
Y

Sentencia/comando

La palabra clave RETURN es utilizada para terminar una subrutina e indicarle al ordenador que regrese al programa principal o una subrutina previa.

Como emplear RETURN
Normalmente, RETURN es utilizada como parte de una sentencia en un programa. RETURN es utilizada independientemente, ubicada al final de la subrutina. Por ejemplo

```
1080 RETURN
```

Al ejecutar esta instrucción, el ordenador regresa a la sentencia inmediatamente después de la última palabra clave GOSUB ejecutada.

Por más detalles consultar la sección sobre GOSUB.

Formato
RETURN

RND RaNDom number — Número aleatorio

Posición en el teciado
EXTEND MODE
T

Función

Esta palabra clave es utilizada para generar un número aleatorio (al azar)

Como emplear RND
La Palabra clave es utilizada independientemente, como parte de una sentencia en un programa o un comando. Por ejemplo

```
60 LET x= RND
```

Al ejecutar esta instrucción el ordenador genera un número al azar o aleatorio, menor que 1 y mayor o igual a 0.

Cuando el Spectrum es encendido, reajustado (control reset), o se introduce la instrucción NEW, los números invocados mediante RND son producidos en la misma secuencia. Esta secuencia es generada tomando las potencias de 75 (75, 75*75, 75*75*75, y así sucesivamente) y dividiendo cada potencia por 65537, y utilizando solamente el residuo, entonces se sustrae 1 del residuo y se divide este resultado por 65536, etc.

Cuando se requiere una secuencia más aleatoria u otra secuencia predeterminada se incluirá la palabra clave RANDOMIZE antes de RND.

RANDOM y números enteros
Si se desea obtener números enteros entre 1 y x, se utilizará la palabra clave INT, en la fórmula INT(RND*X)+1. Para generar un número entero entre 0 y x, se utilizará la fórmula INT(RND*x+0.5)

Formato
RND

RUN Ejecutar

Posición en el teclado
R

Comando/sentencia

La palabra clave RUN le indica al ordenador que ejecute el programa, normalmente a partir de su primera línea.

Como emplear RUN
RUN puede ser empleado como un comando directo o puede forma parte de una sentencia de un programa. Es posible agregar un valor numérico a continuación de RUN. Por ejemplo

```
RUN 50
```

Cuando RUN no es seguido por un valor numérico, el ordenador ejecutará el programa a partir de la primera línea del programa. Cuando se incluye un valor numérico después de RUN, este será redondeado al número entero más próximo si fuera necesario y el programa será ejecutado a partir de esa línea. Si el ordenador no encuentra la línea de ese número, entonces comenzará a ejecutar el programa a partir de la línea inmediatamente siguiente.

Cuando se carga en la memoria del ordenador un programa almacenado en cinta mediante la instrucción LINE, el ordenador lo ejecutará inmediatamente, y no será necesario utilizar la instrucción RUN.

Formato
RUN [expr número entero]

SAVE Registrar

Posición en el teclado
S

Comando/sentencia

La palabra clave SAVE le indica al ordenador que registre en cinta el programa en su memoria, para almacenarlo.

Como emplear SAVE
SAVE es utilizado por lo general como un comando directo, aunque también puede formar parte de una sentencia contenida en un programa.

```
SAVE "ProgramaX"
```

El nombre del programa puede contener hasta diez carácteres. Al ejecutarse esta instrucción el ordenador visualizará el siguiente mensaje en la pantalla

Ponga en funcionamiento la grabadora, luego oprima cualquier tecla

Cuando la cinta comienza a pasar frente a la cabeza de grabación de su grabadora el usuario oprimirá cualquier tecla, y continuación el ordenador comenzará a transmiter el contenido de su memoria RAM a la cinta. Al terminar de grabar, el ordenador visualizará en la pantalla, el reporte 0K,0,1.

Ejecución automática
Para indicarle al ordenador que ejecute un programa pregrabado automáticamente, inmediatamente después de haberlo cargado en su memoria, se utilizarán las palabras claves SAVE y LINE al registrar en cinta el programa. La sintaxis es la siguiente: el nombre del programa, seguido por la palabra clave LINE y un valor numérico, por ejemplo

```
SAVE "programaX" LINE 1
```

El valor numérico a continuación de LINE será redondeado al número entero más próximo si fuera necesario, y será un valor 1 o el número de una línea del programa.

Al ejecutar estas instrucciones, el ordenador registrará el programa en cinta, como cuando ejecuta la instrucción SAVE.

Cuando el ordenador carga el programa pregrabado, lo pondrá en ejecución automáticamente a partir de la línea indicada por el valor numérico. Si no encuentra una línea de ese número, el ordenador comenzará a ejecutar el programa a partir de la línea siguiente en el programa.

Formato
SAVE expr de cadena (**LINE** expr número entero)

SAVE CODE

Posición en el teclado
S
EXTEND MODE
I

Comando/sentencia

La palabra clave SAVE CODE le indica al ordenador que envíe una determinada sección de información en su memoria a la grabadora de casseltes para ser registrada en cinta y almacenada. Esta información podrá ser utilizada nuevamente en el futuro. El procedimiento para cargar la información almacenada en cinta en la memoria del ordenador es mediante la palabra clave LOAD CODE.

Como emplear SAVE CODE
La palabra clave SAVE CODE puede ser utilizada como un comando directo o ser incluida en una sentencia contenida en un programa. La primera parte de la palabra clave, SAVE, es seguida por el nombre que designará la información registrada en la cinta, este nombre es un valor de cadena (string value). A continuación se inserta la segunda parte de la palabra clave: CODE. CODE es seguida por dos valores numéricos separados por una coma. Por ejemplo

```
SAVE "dibujo" CODE 16384,6912
```

El nombre que identifica la información registrada en cinta, insertado entre SAVE y CODE puede tener un máximo de diez carácteres.

Los dos valores numéricos después de CODE son redondeados al número entero más próximo si fuera necesario.

El primer número indica la dirección inicial (16384) de la información almacenada en la memoria. El segundo valor le indica al ordenador la cantidad de bytes que deben ser registradas (6912 bytes en el ejemplo).

El ordenador ejecuta esta instrucción y envía la información a la grabadora de cassettes para ser registrada, como cuando se ejecuta la instrucción SAVE. (Ver la sección sobre SAVE)

Formato
SAVE expr de cadena **CODE**
expr de número entero, expr de número entero

SAVE DATA Registrar data

Posición en el teclado
S EXTEND MODE
D

Sentencia/comando

La palabra clave SAVE DATA le indica al ordenador que registre una matriz (array) en cinta magnética. Más tarde la matriz puede ser vuelta a cargar en el ordenador utilizando la palabra clave LOAD DATA.

Como emplear SAVE DATA
La palabra clave SAVE DATA puede ser utilizada formando parte de una sentencia en un programa o como un comando directo. SAVE es seguida por el nombre bajo el cual se registró la matriz, a continuación se incluye la palabra DATA y luego una letra o una letra con el signo **$**, y finalmente un par de paréntesis vacíos. Por ejemplo

```
450 SAVE "números" DATA n()
750 SAVE "nombres" DATA n$()
```

El nombre de la matriz puede incluir hasta un máximo de diez carácteres. La letra sola o con el signo de **$** ubicada después de DATA será el nombre identificando la matriz en el programa registrado en cinta. La matriz es registrada en cinta siguiendo el mismo procedimiento que en el caso de los programas registrados con la palabra clave SAVE.

Formato
SAVE expr de cadena **DATA** letra [$]()

SAVE SCREEN$ Registrar pantalla

Posición en el teclado
S
EXTEND MODE
SYMBOL SHIFT
K

Comando/sentencia

SAVE SCREEN$ le indica al ordenador que registre la información visualizada en la pantalla en cinta magnética. En el futuro la información podrá ser cargada nuevamente en el ordenador mediante la instrucción LOAD SCREEN$.

Como emplear SAVE SCREEN$
La palabra clave SAVE SCREEN$ puede ser incluida en una sentencia componente de un programa o como un comando directo. El primer término, SAVE, es seguido por el nombre bajo el cual se registrará la imagen en la cinta. A continuación se inserta SCREEN$.

```
SAVE "imagen" SCREEN$
```

El nombre que identificará la imagen o información visualizada en la pantalla, registrada en el cassette, puede tener hasta diez carácteres.

Cuando el ordenador recibe esta instrucción, enviará la imagen o información a la grabadora para ser registrada en cinta, como cuando se ejecuta la palabra clave SAVE. (Ver la sección sobre SAVE)

Formato
SAVE expr de cadena SCREEN$
[**LINE** expr de numero entero]

SCREEN$ Pantalla

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT
K

Función

El propósito de la función SCREEN$ es detectar cual carácter aparece en una posición determinada de la pantalla.

Como emplear SCREEN$
La palabra clave SCREEN$ es seguida por dos valores numéricos separados por una coma y encerrados entre paréntesis, por ejemplo

```
160 IF SCREEN$ (1,c)= "x" THEN
PRINT
"CRASH!"
```

Los valores numéricos después de SCREEN$, redondeados al número entero más próximo si fuera necesario, son las coordenadas del punto en la pantalla. El primero de los valores (indicado en el ejemplo por la variable 1) define el número de la línea horizontal sobre la cual se encuentra el punto, pudiendo variar entre 0 y 21. El segundo valor numérico (c) define la columna sobre la cual se encuentra el punto. Este valor puede oscilar entre 0 y 31.

Cuando el ordenador recibe esta instrucción, examinará el contenido de la posición especificada en la pantalla, y visualizará el carácter en esa posición como una constante de cadena (el carácter entre paréntesis, "x" en el ejemplo).

Si el ordenador no encuentra un carácter en la posición indicada entonces se limitará a visualizar una cadena vacía "".

Formato
SCREEN$ (expr número entero, expr número entero)

SGN SIGN — Signo

Posición en el teclado
EXTEND MODE
F

Función

La palabra clave SIGN sirve para determinar si un número es positivo, negativo o cero.

Como emplear SGN
SGN es seguido por un valor numérico, por ejemplo

```
50 LET x=SGN y
```

Una expresión debe ser *encerrada entre paréntesis*. Obedeciendo la palabra clave SGN, el ordenador visualizará un número 1 si el argumento (y en el ejemplo) es positivo, –1 si el argumento es negativo y 0 si el argumento es cero.

Formato
SGN const-num
SGN var-num
SGN var-num
SGN (expr-num)

SIN SINe — Seno

Posición en el teclado
EXTEND MODE
Q

Función

La función SIN suministra el valor del seno de un ángulo.

Como emplear SIN
SIN es seguido por un valor numérico, por ejemplo

```
80 LET x=SIN y
```

Una expresión debe ser *encerrada entre paréntesis*. El valor numérico insertado a continuación de SIN es el ángulo en radianes, y la función SIN indicará cual es el seno correspondiente a ese ángulo.
Los grados pueden ser convertidos en radianes multiplicándolos por PI/180.
Observe que SIN suministra un valor positivo para ángulos de entre 0 y 180 grados, y un valor negativo para ángulos de entre 180 y 360 grados.

Formato
SIN const-num
SIN var-num
SIN (expr-num)

SQR SQuare Root — Raíz cuadrada

Posición en el teclado
EXTEND MODE
H

Función

SQR calcula la raíz cuadrada de un número.

Como emplear SQR
La palabra clave SQR es seguida por un valor numérico, por ejemplo

```
70 LET x=SQRy
```

Una expresión debe ser *encerrada entre paréntesis*. El valor a continuación de SQR (y en el ejemplo) siempre debe ser mayor que cero, y la función SQR produce su raíz cuadrada.

Formato
SQR const-num
SQR var-num
SQR (expr-num)

STOP

Posición en el teclado
SYMBOL SHIFT
A

Sentencia/comando

STOP detiene la ejecución del programa en un punto determinado. Por ejemplo, esta palabra clave puede ser utilizada para terminar la sección principal de un programa, lo que permitirá incluir las subrutinas en una sección separada, a continuación de la principal. STOP también es un auxiliar útil cuando se aplican técnicas para detectar y corregir los errores en programas.

Como emplear STOP
Normalmente, STOP forma parte de una sentencia contenida en un programa. Es utilizado independientemente, por ejemplo

```
650 STOP
```

Durante la ejecución del programa el ordenador se detendrá al llegar a esta línea y producirá el siguiente mensaje de error en la pantalla:

```
9 STOP Statement
```

agregando el número de la línea y de la sentencia donde se ha detenido el programa.

Si se introduce la instrucción CONTINUE el ordenador reanudará el programa a partir de la sentencia siguiente, con los nuevos valores.

Formato
STOP

STR$ STRing$ — Cadena

Posición en el teclado
EXTND MODE
Y

Función

La palabra clave STR$ convierte números en cadenas (string).

Como emplear STR$
STR$ es seguido por un valor numérico, por ejemplo

```
90 LET a$=STR$ x
```

Una expresión debe estar *encerrada entre paréntesis*. Al ejecutar la instrucción STR$, el ordenador visualizará en la pantalla el valor de su argumento (el número) como una constante de cadena.
En el ejemplo, si se asigna un valor de 65 a x, entonces la sentencia en el ejemplo le asignará un valor de "65" a a$.

Formato
STR$ const-num
STR$ var-num
STR$ (expr-num)

TAB Ver LPRINT, PRINT

TAN TANgent — Tangente

Posición en el teclado
EXTEND MODE
E

Función

TAN suministra la tangente de un ángulo.

Como emplear TAN
La palabra clave TAN es seguida por un valor numérico, por ejemplo

```
130 LET x=TAN y
```

Una expresión debe estar *encerrada entre paréntesis*. El valor a continuación de TAN es el ángulo en radianes, y TAN suministra el ángulo de la tangente.
Es posible convertir grados en radianes multiplicándolos por PI/180.

Formato
TAN const-num
TAN var-num
TAN (expr-num)

TO Hasta

Posición en el teclado
SYMBOL SHIFT
F

Función

La palabra clave TO tiene dos usos diferentes en el dialecto de BASIC utilizado por Sinclair.
En primer lugar, TO puede ser utilizado conjuntamente con la palabra clave FOR para construir un bucle FOR NEXT (ver la sección FOR por más detalles).
En segundo lugar TO también puede ser utilizado para fragmentar ("slicing") cadenas (strings). La fragmentación es la operación mediante la cual una cadena es dividida en subcadenas más pequeñas.

Como se emplea TO para fragmentar cadenas
En este caso TO es utilizado para designar el primer y último carácter de la subcadena contenida en la cadena principal. La palabra clave TO es precidida por una variable de cadena, a continuación se abre un paréntesis, seguido de un valor numérico. Después de TO se agrega otro valor numérico y se cierra el paréntesis. Por ejemplo

```
80 PRINT a$ (4 TO 7)
```

Una expresión de cadena (string expression) también debe ser *encerrada entre paréntesis*. El valor de cadena (en el ejemplo a$) identifica la cadena a ser fragmentada. Los dos valores numéricos (en el ejemplo los números 4 y 7) le indican al ordenador las posiciones del primer y último carácter en la subcadena dentro de la cadena principal. Al ejecutar la instruccion TO el ordenador tomará el fragmento definido en la sentencia de la cadena principal y visualizará la subcadena (caracteres 4 a 7 de $).
Si no se incluye un número identificando el comienzo de la subcadena, el ordenador asumirá que el valor numérico es 1. De la misma manera, si no se incluye un segundo número, para indicar donde termina la subcadena, entonces el ordenador asume que esta termina en el último carácter de la cadena principal.

Formato
const de cadena ([expr-num] **TO** [expr-num])
var de cadena ([expr-num] **TO** [expr-num])
(expr de cadena) ([expr-num] **TO** [expr-num])

USR User SubRoutine — Subrutina del usuario

Posición el en teclado
EXTEND MODE
L

Función

La palabra clave USR es empleada para invocar una subrutina en código de máquina depositada en una dirección especificada de la memoria del ordenador. USR también puede ser utilizada para depositar los datos (data) para gráficas definidas por el usuario en las ubicaciones reservadas del espacio en el tope de la memoria del ordenador.

USR y el código de máquina
Para utilizar el código de máquina, USR es seguido por un valor numérico, por ejemplo

```
80 PRINT USR 65000
100 RANDOMIZE USR 65000
```

Una expresión debe ser *encerrada entre paréntesis*. El valor numérico a continuación de la palabra clave USR será redondeado al número entero más próximo. Este valor es la dirección inicial en la memoria donde se ha depositado la subrutina en código de máquina. Cualquier sentencia en el programa conteniendo USR invocará dicha subrutina, y USR producirá el valor del registro bc.

USR y gráficas definidas por el usuario
Para crear gráficas definidas por el usuario se empleará la palabra clave USR junto con la palabra clave POKE. USR es seguida por una constante de cadena o una variable para invocar una dirección para la sentencia conteniendo POKE. Por ejemplo

```
5 POKE USR "a",255
```

El valor de cadena insertado a continuación de USR puede ser una sola letra, en minúsculas o mayúsculas, entre A y U. El ordenador no distinguirá entre mayúsculas y minúsculas.
USR también suministrará la dirección inicial (dirección del primer byte) de una de las 21 secciones de la memoria especialmente reservadas para las gráficas definidas por el usuario. Cada sección contiene ocho bytes, y mediante la palabra clave POKE es posible almacenar un número directamente en la posición en la memoria, para crear un carácter gráfico.

Formato
USR const número entero
USR var número entero
USR (expr número entero)
USR const de cadena
USR var de cadena

VAL VALue — Evaluar

Posición en el teclado
EXTEND MODE
J

Función

La palabra clave transforma una cadena con un valor numérico en un número.

Como emplear VAL
VAL es seguida por una constante o una variable de cadena. Por ejemplo

```
70 LET x=VAL a$
```

El ordenador elimina las comillas del valor de la constante o variable de cadena y lo que queda se evalua como un valor numérico. VAL evalua esos elementos y dará un resultado numérico.
Ejemplos
Si a$ tiene el valor "435", la sentencia en el ejemplo anterior asignará un valor de 435 a la variable x. Sin embargo VAL también puede evaluar expresiones, por ejemplo

```
10 INPUT a$,x
20 PRINT VAL a$
```

El valor de cadena asignado a a$ debe ser una expresión utilizando x, por ejemplo "x*x". El ordenador asignará un valor numérico a x, por ejemplo 5. VAL elimina las comillas para obtener x*x, y evalua el resultado utilizando el valor asignado a x. Por ejemplo si el valor de x era 5 en este caso el resultado será 5*5=25.

Formato
VAL const de cadena
VAL var de cadena

VAL$ VALue (string) — Evaluar (cadena)

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT J

Función

VAL$ evalua una cadena como una expresión de cadena. Su resultado es una cadena, como en caso de VAL, pero en lugar de producir un valor numérico como este último, VAL$ produce otra cadena.

Como emplear VAL$
VAL$ es seguido por una variable de cadena, por ejemplo

```
130 PRINT VAL$ a$
```

El ordenador elimina las comillas del valor de la variable de cadena y lo que queda se evalua como una cadena. VAL$ evalua esta última expresion y produce el resultado como una constante de cadena.

Ejemplos
Pruebe este programa

```
10 INPUT a$,x$
20 PRINT VAL$ a$
```

El valor de cadena asignado a a$ debe ser una expresión empleando x$, por ejemplo "x$+x$"". El ordenador evaluará x$ como una cadena, por ejemplo "DO". La palabra clave VAL$ elimina las comillas del valor atribuido a a$ para obtener x$+x$, y lo evalua utilizando el valor asignado previamente a x$. En este caso el ordenador visualizará el resultado DODO.

Formato
VAL$ var de cadena

VERIFY Verificar

Posición en el teclado
EXTEND MODE
SYMBOL SHIFT R

INFORMES EN LA PANTALLA

Cuando el ordenador Spectrum de ja de escribir BASIC, un informe aparece al fondo de la pantalla de la televisión, indicando que un comando o un programa ha sido completado o que un error ha ocurrido. Cada informe consiste en un numéro de código o una letra seguida por un mensaje breve, y a consecuencia los números de la linea y la sentencia en los cuales se paró el ordenador. Un comando se indica como linea Ø y dentro de una linea, sentencia 1 se ubica al principio de la linea, sentencia 2 se ubica después de los primeros dos puntos o THEN, y asf sucesivamente. CONTINUE por lo general hace que el programa comienza de nuevo en la sentencia especificada en el informe.

Ø OK
Se ha completado un éxito una operación determinada. Si se inserta la palabra clave CONTINUE el ordenador ignorará este último informe y reanudará la ejecución del programa en la sentencia especificada en el reporte previo.

1 NEXT without FOR
Al ejecutar el programa, el ordenador ha encontrado una palabra clave NEXT pero no ha podido encontrar su palabra clave complementario FOR, pero existe una variable ordinaria con el mismo nombre.

2 Variable not found.
Se ha utilizado una variable de control con la palabra clave NEXT, pero sin incluir previamente su palabra clave complementario FOR.
Se ha incluido una variable con subíndice pero sin haber definido sus dimensiones mediante una sentencia de DIM, o cargado una matriz desde cinta magnetica.

3 Subscript wrong
El subíndice de una variable se encuentra fuera de las dimensiones de la matriz correspondiente.

4 Out of memory
No queda suficiente espacio en la memoria del ordenador para completar la sentencia o comando.

5 Out of screen
Una sentencia INPUT ha generado más de 23 líneas en la mitad inferior de la pantalla, o se ha utilizado un número de línea de 22 o más con la palabra clave PRINT AT.

6 Number too big.
El ordenador ha intentado producir un número mayor que 1Ø(38) apróximadamente.

7 RETURN without GOSUB
El ordenador ha encontrado una palabra clave RETURN pero no puede ubicar la palabra clave complementaria GOSUB correspondiente.

8 End of file
Informe sobre la manipulación de información registrada en la Microdrive.

9 STOP statement
El ordenador ha encontrado una sentencia conteniendo la palabra clave STOP, CONTINUE, reanudará el programa a partir de la sentencia siguiente.

A Invalid argument (Argumento erróneo)
Se ha dado un argumento o valor incorrecto.

B Integer out of range
El ordenador ha redondeado un número al número entero más próximo. El resultado queda fuera de los límites del ordenador y no puede ser aceptado por éste.

C Nonsense in BASIC
El ordenador ha encontrado una sentencia que no tiene sentido en BASIC y en el contexto del programa.

D BREAK-CONT repeats (Se repite BREAK)
Se ha oprimido la tecla BREAK. El efecto de la palabra clave CONTINUE es repetir la sentencia donde el ordenador se detuvo.

E Out of DATA (No quedan datos)
Obedeciendo una instrucción READ, el ordenador se ha dirijido a la lista de datos o información contenida en una sentencia encabezada por la palabra clave DATA.

F Invalid file name
SAVE Ha encontrado que el nombre agregado a continuación de SAVE tiene más de diez carácteres.

G No room for line
No hay suficiente sitio en la memoria del ordenador para admitir una nueva línea de programa.

H STOP in INPUT (STOP al ejecutar instrucción INPUT)
Al ejecutar una instrucción INPUT, el ordenador ha encontrado la palabra clave STOP. La palabra clave CONTINUE tiene el efecto de repetir la sentencia conteniendo INPUT

I FOR without NEXT (FOR sin NEXT)
El ordenador ha encontrado un bucle de FOR ... NEXT donde sus límites o los valores definidos por STEP están incorrectos. **FOR** x=5 **TO** Ø y además no se ha encontrado la palabra clave NEXT complementaria.

J Invalid I/O device
Informe sobre la manipulación de infoemación registrada en la Microdrive.

K Invalid colour (Color inválido)
El valor especificado en sentencias conteniendo las palabras claves INK, PAPER, FLASH, BRIGHT, INVERSE u OVER, o el caracter de control correspondiente se encuentra fuera de los límites del ordenador.

L BREAK into program
Durante la ejecución de un programa se ha oprimido la tecla BREAK. Si se desea continuar ejecutando el programa bastará con introducir la palabra clave CONTINUE.

M RAMTOP no good
El valor especificado para RAMTOP es demasiado grande o demasiado pequeño.

N Statement lost
Durante la ejecución del programa el ordenador ha encontrado una instrucción indicandole que avance a una sentencia que ya no existe.

O Invalid stream
Información registrada en la Microdrive.

P FN without DEF
La palabra clave FN, ha contrado encontrar la sentencia DEF FN correspondiente.

Q Parameter error
Una sentencia conteniendo la palabra clave FN contiene una cantidad equivocada de valores o argumentos para ser pasados a la función. (Una cadena en lugar de número o viceversa).

R Tape loading error
Información, o verificación no se ha completado correctamente. El ordenador no especificará cual ha sido el error.

MAS ALLA DEL BASIC

BASIC es un lenguaje de propósito general para ordenadores que se se desempeña bien en la mayoría de las aplicaciones. Existen disponibles programas (software) que le suministrarán otros lenguajes, tales como FORTH, micro-PROLOG, y LOGO. Estos lenguajes operan de una forma muy diferente a BASIC y le abren nuevas posibilidades a su ordenador.

Debido a que BASIC es un lenguaje de propósitos generales, a veces puede llegar a ser algo incómodo cuando se trata de utilizar el ordenador para ciertas aplicaciones especiales. También es comparativamente lento. Otros lenguajes pueden ofrecer una mayor flexibilidad combinada con una programacion más sencilla y velocidades mayores. El lenguaje FORTH, por ejemplo, le permite definir sus propias palabras para programar. Estas palabras serán incluidas en instrucciones que su ordenador comprenderá y ejecutará a velocidades diez veces superiores a las de los comandos equivalentes en BASIC.

En el caso del lenguaje micro-PROLOG, el ordenador adquiere la capacidad de comprender frases en inglés sencillas, almacenándolas en su memoria como una base para desarrollar un diálogo con el usuario. LOGO es un lenguaje para ordenadores desarrollado con fines educacionales.

El código de máquina o "machine code"

El propósito del lenguaje BASIC es permitir que el usuario pueda darle instrucciones al ordenador, en una forma fácil de comprender. La Unidad Procesadora Central. Las instrucciones en BASIC introducidas en el ordenador, por ejemplo desde el teclado, deben ser traducidas mediante un programa permanente, denominado Intérprete de BASIC (Basic Interpreter). Este programa se encuentra almacenado en una sección de la memoria del ordenador y convierte la instrucción en BASIC en una secuencia de señales de código, las cuales son enviadas a la CPU.

El intérprete tomará algún tiempo para traducir las instrucciones en BASIC al lenguaje comprendido por la CPU ZXØØ, conocido como código de máquina (machine code en inglés).

Es posible saltearse el intérprete, enviando instrucciones redactadas en código de máquina directamente a la ZXØØA en la CPU del ordenador. En este caso su programa será ejecutado con gran rapidez.

Para tener una idea acerca de la rapidez con que el ordenador ejecuta instrucciones en código de maquina, le sugerimos que introduzca en el ordenador, a traves del teclado, el programa siguiente.

FRANJAS INSTANTÁNEAS

```
 10 FOR x=Ø TO 15
 20 READ n: POKE 65ØØØ+x,n
 30 NEXT x
 40 DATA 33,255,63,1,1,24,22
 50 DATA 55
 60 DATA 35,11,120,177,200,114,
24,245
 70 RANDOMIZE USR 65ØØØ
```



Pruebe cambiar el valor numérico 55, en la línea número 5Ø del programa, a cualquier valor entre 1 y 255, para cambiar las franjas.

También es posible producir franjas de colores insertando comandos conteniendo la palabra clave INK. Lo que deseamos demostrarle es la *velocidad* con que cambian las imagenes visualizadas en la pantalla por el ordenador, cuando las instrucciones son en código de máquina.

Este programa funciona porque las sentencias de DATA contienen 16 códigos que son almacenados a partir de la dirección 65ØØØ de la memoria, en ejecución de las instrucciones contenidas en las líneas 1Ø a 3Ø del programa. La línea 7Ø envia los códigos a la microchip ZXØØA, lo que produce inmediatamente la imagen en la pantalla.

Para ayudarle a redactar sus propios programas en código de máquina, existen programas especiales denominados ensambladores (assemblers). Los ensambladores operan con una lista de instrucciones para ser introducidas en el ordenador, en lugar de la lista de números utilizada por el código de máquina. El propósito de ensamblador es traducir estas instrucciones a los números del código de máquina. Las instrucciones del ensamblador no son palabras clave en inglés como en BASIC.

LA TERMINOLOGIA DE LOS ORDENADORES: UN BREVE DICCIONARIO

A continuación explicaremos los significados de algunos de los términos asociados con el manejo de ordenadores que usted encontrara en esta publicación.

Las palabras en bastardilla han sido explicadas en otra parte de esta sección. Si desea averiguar el significado de algún término que no se encuentra incluido en esta sección le sugerimos que consulte el índice de esta publicación.

Almacenar (save). Es el procedimiento mediante el cual se deposita un *programa* o datos (data) en un sistema de almacenamiento de datos, por ejemplo un cassette o un cartucho microdrive.

Anidar o insertar (nesting). Ese término se aplica en el caso de programas con bucles, cuando uno o más bucles se encuentran "anidados" en el interior de otro bucle.

Argumento (argument). Es un *valor*, empleado por una *función* para obtener un resultado.

Atributos (attributes). Códigos que le atribuyen colores a los *carácteres*.

BASIC. Es el idioma de ordenadores utilizado por el ZX Spectrum + y la mayoría de los micro-ordenadores utilizados en el hogar.

Binario, Codigo (Binary Code). Es un tipo de código utilizado por los ordenadores.

Bit. Uno de los dos elementos fundamentales del código binario.

Bucle (loop). Una sección de un programa que es repetida varias veces.

Byte. Es un conjunto de ocho bits. Representa un número. El valor de este número puede variar entre Ø y 255. Cada una de las *direcciones* en la *memoria* del ordenador alberga un byte.

Cadena (string). Es un grupo de uno o más *cáracteres*, encerrados entre comillas para distinguirlos de los números y variables numéricas.

Cadena, Variable de. (string variable). Es una *variable* conteniendo una *cadena*. Las variables de cadena siempre consisten de una sola letra seguida del signo **$**.

Carácter (character). Cualquier letra, número (Ø a 9), signo o unidad *gráfica*, que puede ser visualizada por el ordenador en la pantalla o impresa.

Carácteres, juego de. (character set). Es el conjunto de los *carácteres* y ciertos códigos de control preestablecidos, empleados por el ordenador.

Cargar (load). Alimenter un *programa* o datos (*data*) en el ordenador, a partir de un equipo de *almacenamiento*, por ejemplo una grabadora de cassettes o un *cartucho rom*.

Código de máquina (machine code). Es el lenguaje comprendido por su ZX Spectrum +. Los programas redactados en *BASIC* son traducidos el código de máquina por el ordenador, antes de ejecutarlos.

Comando (command). Una instrucción individual que es ejecutada por el ordenador, o un *comando directo*.

Comando directo (direct command). Un juego de una o más instrucciones que el ordenador ejecuta inmediatamente.

Constante (constant). Un número o un grupo formado por una o más letras o cualquier otro tipo de *carácteres*.

CPU (Central Processing Unit). Ver: Unidad Procesadora Central.

Cursor. Es una letra parpadeante que indica la posición en la pantalla donde se visualizará algún elemento. Puede indicar el *modo* en el cual se encuentra el ordenador en ese momento.

Datos (Data). Son los datos transferidos al ordenador por un programa, o por otros métodos, para producir resultados.

Dirección (address). Una unidad individual de memoria. Existen 65536 direcciones en la ZX Spectrum +.

Editar (edit). Es la operación de modificar algunos detalles de un *programa*.

Expresión (expression). Una combinación de *constantes, variables*, y *palabras claves*.

Función (function). Es una operación donde el ordenador toma uno o más *valores* (o *argumentos*) y los utiliza para obtener un resultado. El resultado es otro valor.

Gráficas (graphics). La producción de imágenes, tales como figuras, esquemas, o diagramas por el ordenador.

Hardware (equipos). El ordenador y otros accesorios o máquinas, por ejemplo los *accesorios periféricos*.

Imprimir (print). Visualizar resultados o *gráficas* en la pantalla de la televisión, o imprimirlos mediante una impresora.

Información (information). Palabras, números, y signos individualmente o combinados, que el ordenador debe manipular.

Input. *Programas* y *datos* alimentados o introducidos en el ordenador.

Introducir (enter). Es la operación por la cual se comunica al ordenador una instrucción o información completa.

K. Es una medida de la capacidad de la *memoria* de un ordenador. 1K es equivalente a 1 kilobyte o 1024 bytes. La capacidad de la memoria indicada en K, es la cantidad total de *direcciones* en la memoria. Cada una de estas direcciones puede albergar un *byte*. El ZX Spectrum + tiene una memoria de 48K RAM y 16K ROM, lo que ofrece una capacidad total de 64K.

Línea (line). Una instrucción o juego de instrucciones en un **programa.** Cada línea tiene un número, que le indica al ordenador la secuencia que debe seguir para ejecutar el programa.

Listado (listing). Una lista ordenada de las líneas de programa.

Lógico (logic). Es el proceso mediante el cual un ordenador decide si los resultados son correctos o incorrectos, o determina si determinados elementos son *verdaderos* o *falsos*.

Matriz (array). Un grupo de unidades de *información* vinculadas entre si, que son conservadas en un grupo en la *memoria* del ordenador.

Memoria (memory). Es la parte del ordenador que almacena el *programa*, los *datos* y también las instrucciones de operación permanentes.

Modo (mode). En el Spectrum, uno de los cinco estados que determinan cuales serán las *palabras clave* y los *carácteres* producidos por cada tecla en el teclado del ordenador.

Operador (operator). Una instrucción que ejecuta operaciones aritméticas o *lógicas*.

Output (salida o rendimiento). Son los resultados producidos por el ordenador.

Palabra clave (keyword). Una instrucción para el ordenador en BASIC.

Periférico (peripheral). Cualquier accesorio conectado a su ordenador.

Pixel. Es el punto de color más pequeño que puede aparecer en la pantalla de su televisión.

Programa (program). Una secuencia de instrucciones para ser ejecutadas por el ordenador.

Punto de contacto (interface). Es una unidad que conecta el ordenador con componentes *periféricos* (o componentes *periféricos* entre si) y que asegura que estos puedan comunicarse los unos con los otros.

RAM (Random Access Memory). Es aquel sector de la *memoria* total del ordenador que puede almacenar *programas* o *datos*, y otros *valores* cambiantes. También conocida como la memoria volátil, el contenido de la memoria RAM es borrado al apagarse el ordenador. El ZX Spectrum + tiene una memoria RAM con una capacidad de 48K.

Registro (register). Una pequeña unidad de la memoria, separada de la *memoria* principal. Los registros dentro de la CPU (Unidad Procesadora Central) son utilizados para ejecutar el proceso de computación.

Reporte (report). Un mensaje visualizado en la pantalla por el ordenador, describiendo sus acciones.

Resolución (resolution). La nitidez o cantidad de detalles que pueden ser producidos por las *gráficas* del ordenador.

ROM (Read Only Memory). Es el sector de la memoria total del ordenador conteniendo *programa* e instrucciones permanentes para el ordenador. Esta memoria solamente puede ser leída. El ZX Spectrum + tiene 16K de ROM.

Save (ver almacenar)

Scroll (enrrollar). El movimiento de los *caracteres* visualizados en la pantalla de la televisión, que permite proyectar una imagen.

Sentencia (statement). Ya sea una *palabra clave* que es utilizada para forma una instrucción contenida en una línea de programa, o la instrucción misma.

Sintaxis (syntax). Es la secuencia correcta de las *palabras clave, constantes, variables* y *expresiones*, requeridas para formar una instrucción válida en BASIC

String (ver: cadena)

Unidad Procesadora Central (Central Processing Unit o CPU). Es el elemento central de su ordenador. Ejecuta las funciones de ordenación y controla las demás unidades. El ordenador ZX Spectrum + utiliza un microprocesador Z80.

Valor (value). Cualquier número o grupo de *constantes, variables* o *expresiones* requeridos del ordenador o producidos por éste.

Variable (variable). Una o más unidades de *memoria* conteniendo una *constante* particular para ser utilizada por el ordenador. El ZX Spectrum + distingue entre *variables numéricas y variables de cadena*.

INDICE

Los números de página en bastardilla se refieren a ilustraciones ó secciones especiales en las páginas indicadas.

Aleatorio (Random). Efectos 30
Almacenaje de programas 44,45
Alta resolucion. Gráficas 26,*28-29*
Altavoz 43
Alteración de programas 9
Amplificación de sonido 37
Animación 34-35
Antena, cable y enchufe 4,5
Aritméticos. Operadores 22,*22*
ATTR 35

Baja resolución. Gráficas de *26-27*
Barra espaciadora (teclado) *19*
BASIC 18; 49-73
BASIC de Sinclair 49-73
BEEP 36; *18*
BIN 33
Binario. Código. 44
Borde (Border). Color. 25-5; *6*
BREAK *19*
BRIGHT 31
Bucles (loops) 26-27,30

Cálculos 22-23; *22-23*
Cadenas (Strings) *22*
CAPS LOCK *21*; *18*
CAPS SHIFT 8, 21; *18*
Carácteres
Creación de. 32-33
Selección 20
Carácteres Definidos por el Usuario (CDU) 80; *32,33*
Cargando programas en el ordenador 13,14-15; *14-16*
Cartuchos, Microdrive 12, 46; *46*
Cartuchos, ROM 12,47; *47*
Cassettes, grabadora. V: Grabadora de cassettes
Central Processing Unit (CPU) 43,44,48,*75*; *43,45*
Cinta. Cable de. 46
Cintas de cassette 12; *44,45*
Almacenaje 12
Cuidado de las cintas 12
Rotulos 14,39
Sonido 12
CIRCLE 28
Código Binario 44
Código de Máquina 75
Colisiones 34,35
Color 24, 25;*24,25*
Códigos *24*
Color del borde (Border) 24-5; *6*
Combinaciones 25
Control, códigos *33*
Mezclas 32
Pruebas 6;*24*
Teclas *19*
Comas 23, 51
Comandos 22, 50
Comillas 23, 51
Conector de reborde *5,43,47*
Conexiones 5
Grabadora de cassettes *13*
Electricidad 5
Televisión *4*
Contacto. Puntos de (interfaces). *45,46-47*
CPU. V: Central Processing Unit
Cuadrícula de alta resolución. 28;*80*
De baja resolución. *26*;*80*
Cursor. Controles de los *19*
Chips 42-43
Chips lógicos *43*

DATA 33
DELETE 10
Dos puntos 23, 51
DRAW 28,29

EAR. Enchufe *37*; *5*, *13*
EDIT *18*,*21*
Editamoto líneas de programa 21
Ejecución de programas (RUN) 8-9
Electricidad. Enchufe 9V CD *5,43*
Electricidad. Suministro de. 4,5;*5,43*
Enchufes 5
Enchufe MIC 37;*5,13*
ENTER *9,10,11,19*
Entrada-Salida. Caminos de. 45
Equipos (hardware). Definición 12
Equipos accesorios *45,46-47*
Errores, Corrección 10, 21
Reportes en la pantalla 74
EXTEND MODE 8,*21*; *18*
Extend mode 21; *20*

Figuras. Diseño de. *30-31*
De baja resolución *26-27*
Llenando figuras 29; *29*
FLASH 31
FOR NEXT 26-27, 29, 30, 31, 34
FORTH 75
Funciones 50

GOTO 23
Grabadoras de cassetes
Cargando programas 14-16
Como amplificadores 37;*37*
Conexiones 5,13; *13*
Contador de revoluciones 14
Control de tono 14,15; *16*
Control de volumen 14,15; *16*
Registran do programas 38-40
Selección 12
V: Cintas de cassette.
Gráficas. Aminación. 34-5
Alta resolución 26;*28-29*
Baja resolución *26-27*
Color 24-25
Creando carácteres *32-33*
Dibujos *30-31*
Efectos aleatorios 30
Llenando figuras 29;*29*
Gráficos *25*;*25*
GRAPH *21*,*18*,*26*
Graphics mode. V: Modo gráficas

IF THEN 29
Impresoras 45,47;*45,47*
Interferencia de radio 4
Ink. Color de 24-25
INPUT 23,29
Interfaces (puntos de contacto) *45,46-47*
Introduciendo información 8,9
INV VIDEO *18*
INVERSE 31

Juego de carácteres *51*

LET 23
Letra. Modo. *21*;*20*
Líneas 8
Borrando líneas 21
Editando líneas 21
LIST 21
Listado de programas 8,21
LOAD 14-16
LOAD, V:Cargando programas
LOGO 75
Loops. V: Bucles

Memoria *12*;*42,43 44-48*
Memoria. Mapa de la. *48*
MIC. Enchufe 37;*5,13*
Microdrives 46;*5,46*
Cartuchos 12,46;*46*
Cargando 46
Micro-PROLOG 75
MODEM
(Modulador-Demodulador) 46
Modos 20-21
Modo Gráficas *21*;*20*
Modo Letras *21*;*20*
Modo Mayúsculas *21*;*20*
Modo Palabras Clave 20;*20*
Música 36-37
Tono *36*

NEW 11,12; *18*
Numeros 50
Nuevos programas 11

Operadores aritméticos 22;*22*

Palabras clave 9, 18-19,5,52-73
Selección de, 19,20
Palabras clave. Modo 20;*20*
Palancas de control 45,47
Pantalla. Movimiento vertical. (Scroll) 9
Papel (Paper). Color. 24-25
Paréntesis *23*
Pixel 28
PLOT 28
POKE 48
PRINT 22
Programación 17-40
Programas
Borrando líneas del, 21
Cargando programas. 12,13,14-15; *14-15*
Corrección de errores 10
Edición 21
Ejecución (RUNning) 8-9,44
Iniciando nuevos pr. 11
Introducción en el ordenador 8-9,44
Registrando 13,38-40
Verificación de, 39
Volviendo a comenzar 10
Programas pre-grabados 12-13; *13*
Cargamdp estp s-rpgra,as 14-16; *14-16*
Programas adeuados para el Spectrum 12
Tipos 12
Puntuación. Signos de. 23,51
Punto 23, 51
Punto y coma 23,51

RAM (Random Access Memory)
Memoria RAM 42,48; *42-45*
Cartuchos RAM 4
RAMTOP 48
Random. V: Aleatorio
READ 33
Reajuste (Reset) 11,12; 5
Registrando programas 13,38-40
Reiniciando programas 10
REM 30
Reportes en la pantella 74
RND 26,30
ROM (Read Only Memory)
Memoria ROM 48; *43,45*
ROM. Cartuchos 12,47; *46-47*
RS232. Punto de conexión (Interface) 47;*45*
RUN. V: Ejecución de programas

SAVE 39-39
SAVE V: Registrando programas
Scroll. V: Pantalla. Movimiento vertical
Sentencias 22, 50
Signos matemáticos 22,50
Seleccionando, 19
Símbolos. Selección. 20
Sintonización de la televisión 6;*6*
Software. V: Programas. V: Programas Pre-grabados.
Sonido. Efectos de sonido 36-27
STEP 29
Strings (cadenas) 22
Subrutinas 30-31
SYMBOL SHIFT 3,21; *19*

Teclado *18*, *19*
Carácteres gráficos *26*
Introduciendo información 8,9
Modos 20-21
Teclas 18-19; *18-19*
Número *19*
Operación 20-21;*20-21*
Televisión.
Codificador 42
Conexión. 5
Tipos adecuados. 4
Sintonización. 6;*6*
Tinta (ink). Color de, 24-25
Tono. Controles en la grabadora 12,14,15
Tono musical 36
TRUE VIDEO *18*

Uncommitted Logic Array (ULA) 42
Unidad Procesadora Central. V:Central Processing Unit

Variables 22-23,50
Variables de sistema 48
Voltaje. Regulador. *43*
Volumen. Controles en la grabadora. 12,14,15

ZX80 Microprocesador. 43,75; *45*
ZX Interface 1 (Conector) 45,46-47
ZX 16K RAM 4

Primera edición, 1984, por Dorling Kindersley Ltd., 9 Henrietta Street, Londres WC2E 8PS, Inglaterra. Editado en asociación con Sinclair Research Ltd., 25 Willis Road, Cambridge, Inglaterra.

© 1984 by Sinclair Research Ltd. y Dorling Kindersley Ltd., Londres, Inglaterra. Ilustraciones © 1984 por Dorling Kindersley Ltd., Londres, Inglaterra.

Los nombres sinclair y ZX Spectrum + son marcas registradas de Sinclair Research Limited.

Se reservan todos los derechos. No está autorizada la reproducción de ninguna parte d esta publicación, o su totalidad, almacenamiento en sistemas de almacenamiento de datos, o transmisión en cualquier forma o por cualquier medio, incluyendo medios electrónicos, mecánicos, fotocopias, almacenamiento en cinta, etc., sin la autorización previa y por escrito de los propietarios de los derechos de autor (copyright).

Editor David Burnie
Editor de Arte Peter Luff
Diseño Debra Lee
Fotógrafo Trevor Melton
Fotografía screen-shot Vincent Oliver
Editor Gerente Alan Buckingham

Reproducción por A. Mondadori, Verona, Italia. Impreso y encuadernado en Italia, por A. Mondadori, Verona.

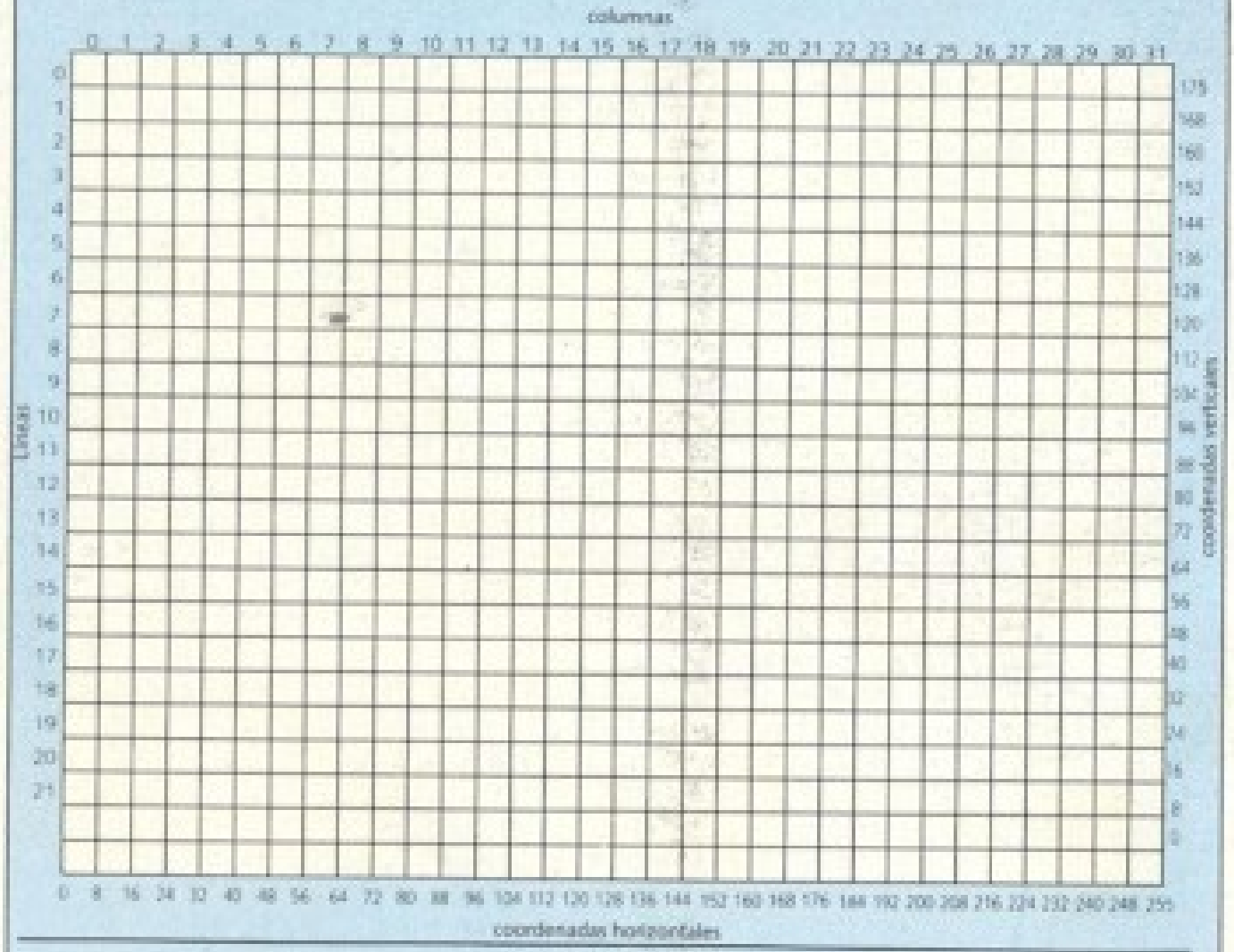
Cuadrícula para la presentación visual en la pantalla de la televisión

La cuadrícula a continuación muestra las coordenadas para gráficas de alta y baja resolución. Si bien las palabras claves individuales que controlan los efectos gráficos operan solamente con coordenadas de alta o de baja resolución, es posible utilizarlas al mismo tiempo cuando produciendo una presentación visual.

La cuadrícula de baja resolución se divide en dos partes principales, el área de visualización principal y dos líneas en la parte inferior de la pantalla. La palabra clave PRINT AT producirá carácteres en la parte superior y la palabra clave INPUT AT producirá palabras en la parte inferior.

Las coordenadas para la pantalla de baja resolución se indican sobre los márgenes superior y de la izquierda de la cuadrícula.

La cuadrícula de alta resolución ocupa solamente el área de visualización principal. Las palabras claves PLOT, DRAW y CIRCLE son empleadas para producir gráficas de alta resolución. Las coordenadas de la cuadrícula de alta resolución se indican sobre los márgenes inferior y de la derecha de la cuadrícula.



Cuadrícula para las gráficas definidas por el usuario
Sumar los números en la tabla para obtener el valor decimal de cada línea o hilera horizontal de la cuadrícula, y a continuación utilizar la palabra clave POKE USR para programar el carácter.

128	64	32	16	8	4	2	1	Totales de las líneas o hileras