

51
185 pts.
IVA INCLUIDO

ALON

Enciclopedia Práctica del Spectrum



Nueva Lente/Ingelek



SPECTRUM 128: EL NUEVO SINCLAIR



uando en el año 1983 los aficionados a la informática recibíamos la noticia de la aparición en el mercado de un nuevo microordenador a un

precio revolucionario comparado con los manejados por aquel entonces, un cierto rictus de susceptibilidad se apoderó de todos nosotros.

Se trataba del Sinclair ZX Spectrum. ¿Pero quién era Sinclair? Sobre esta marca las únicas noticias que teníamos conducían directamente al nanoordenador ZX 81, un modesto equipo con teclado a base de membranas sensibles y con 1 Kbyte de memoria en su configuración base.

Pero el nuevo Spectrum le desbordaba en todos los aspectos: memoria, velocidad de proceso, facilidad de manejo y programación, posibilidades sonoras y gráficas, ampliaciones, etc. En fin, todo lo necesario para tomar contacto con la microinformática a un precio asequible, hasta entonces reservada a los usuarios de equipos profesionales y con economías a prueba de toda solvencia. Las primeras pruebas efectuadas con aquel ordenador despejarían pronto todas nuestras dudas: se trataba de un equipo sencillamente revolucionario.

Por supuesto, no todo eran virtudes y, por ejemplo, el modesto teclado de gomas o la lentitud experimentada en todas las operaciones de transferencia de información entre ordenador y cassette, comenzaba a traer de cabeza a todos los usuarios ansiosos de obtener un rendimiento más profesional de su nueva máquina.

Los fabricantes renovaron entonces su oferta y pusieron a nuestra disposición un nuevo medio de almacenamiento de bajo coste y gran velocidad de acceso: el Microdrive.

Poco después aparecería en el mercado el ZX Spectrum +, el cual incorporaba como principal novedad un teclado semiprofesional ampliado que permitía minimizar las pulsaciones necesarias al introducir la información, así como aumentar la precisión y rapidez de estos procesos.

Ya por aquel entonces, primavera de 1985, el mercado apuntaba hacia una nueva generación de microordenadores, los cuales dotaran al usuario de una mayor potencia en el tratamiento de grandes bloques de información.

Los principales fabricantes del momento se lanzaron a una desenfadada carrera al objeto de cubrir la demanda existente en este terreno, y fruto de ello aparecieron entre otros el Commodore 128 o el Amstrad CPC 6128, con una relación precio/prestaciones, francamente formidable.

Sinclair no podía ser menos y en colaboración con Investrónica efectuó en España durante el SIMO 85 la presentación exclusiva a nivel mundial del nuevo Spectrum 128 K.

LA PRIMERA IMPRESION

Tal acontecimiento pronto suscitó una enconada polémica entre el distribuidor y los primeros usuarios del nuevo prototipo. Opiniones para todos los gustos fueron vertidas en las revistas especializadas en ordenadores Sinclair.

Temas como la absoluta compatibilidad augurada por el distribuidor con los modelos anteriores, el nuevo sistema operativo en modo 128, la ausencia de un medio de almacenamiento rápido y

El nuevo Spectrum 128 K.





fiable que no fuera el sempiterno Microdrive o la mínima información que el usuario recibe en los manuales del equipo no tardaron en ser cuestionados.

Pues bien, tratemos de aclarar todas estas circunstancias comenzando desde el principio. Analicemos cuidadosamente las novedades que incorpora el ZX Spectrum 128 K.

Tras desembalarlo, observamos que la carcasa exterior tiene el mismo aspecto del Spectrum +, de hecho este nombre aparece grabado en la zona superior izquierda, aunque en la parte inferior se añade el distintivo 128 K.

La primera novedad consiste en la aleta de refrigeración exterior situada a la derecha del teclado principal. Su justificación es obvia. El 128 consume algo más que sus predecesores y además, en ellos se observaba una alarmante facilidad para sobrecalentarse tras varias horas de trabajo, y por tanto, la decisión de mejorar la disipación del calor es completamente acertada.

En la parte trasera encontramos los conectores para la fuente de alimentación exterior, la antena de televisión y la tarjeta de expansión, y en el lugar de los enchufes EAR y MIC se ha instalado una salida para monitor RGB.

Estos se hallan situados ahora en el lateral izquierdo de la carcasa y aunque los manuales

(más adelante trataremos de ellos) no lo señalan, es posible mantener conectados a la grabadora los dos cables durante las operaciones de lectura/escritura.

Junto a ellos, un conector marcado con RS 232 permite el enlace con impresoras que sigan este protocolo, modems o instrumentos musicales en la norma MIDI.

Finalmente, en el frontal, un pequeño enchufe permite la conexión mediante un cable extensible, del tipo empleado en los teléfonos, del elemento exterior más novedoso: el teclado numérico independiente.

LOS TECLADOS

Dos octavas contiguas se solapan.

CANALES						
TONO			RUIDO			
	A	B	C	A	B	C
CODIGO	1	2	4	8	16	32

A primera vista el teclado principal es idéntico al del Spectrum + con sus 58 teclas dispuestas en la misma forma, pero con una diferencia, muchas de ellas están traducidas al español. Tal es el caso de PARAR (BREAK), BORRAR (DELETE), CAMBIO (SYMBOL SHIFT), MODO EXTENDIDO (EXTEND MODE), etc.

Además, la serigrafía ha cambiado en otras muchas de lugar. Por ejemplo, en las teclas 4 y 5 encontramos las letras Ñ y ñ, respectivamente. Para obtenerlas se pulsán simultáneamente éstas y la tecla CAMBIO. Observemos las fotografías y encontraremos bastantes novedades en este aspecto.

Esta circunstancia, a priori, no plantea ningún problema, pues se ha respetado el código ASCII y un programa que antes seleccionaba una op-

OCTAVA X+1																				
ESCALA INFERIOR							ESCALA SUPERIOR													
c	d	e	f	g	a	b	C	D	E	F	G	A	B	C	D	E	F	b	A	B
ESCALA INFERIOR							ESCALA SUPERIOR													
OCTAVA X																				

FRECUENCIA	DO	RE	MI	FA	SOL	LA	SI
NOMBRE	C	D	E	F	b	A	B

Códigos para la selección de canales de tono y ruido.



ción tras recibir un carácter dólar (\$), seguirá haciéndolo de igual manera.

Los 18 pulsadores del teclado auxiliar constituyen el principal añadido al sistema para simplificar la introducción de datos numéricos y optimizar el trabajo de edición de programas y textos. Nos referiremos a todas sus funciones cuando tratemos el modo de trabajo 128.

EN MARCHA

Cuando conectamos el ordenador a la red aparece sobre la pantalla el nuevo mensaje del fabricante:

(c) 1985 Sinclair Research Ltd
ESPAÑOL

y tras él, un pitido indicando que la inicialización fue correcta. La primera sorpresa es que éste no procede del minialtavo, sino del televisor.

El buzzer ha sido suprimido y toda emisión de sonido se efectúa vía el altavoz del televisor lo que mejora con creces la altura de la señal audible. Si nuestra pantalla fuera un monitor sin altavoz, podemos recoger el sonido del conector MIC y llevarlo a un amplificador.

A partir de este momento estamos bajo las órdenes del nuevo sistema operativo del 128. En efecto, si pulsamos una tecla aparece en la esquina superior izquierda de la pantalla un cursor cuadrado parpadeante.

Una línea en la parte inferior nos informa constantemente del modo en que nos encontramos: extendido, gráfico, mayúsculas, etc. Al principio estaremos en este último, y letra a letra debemos teclear los comandos del BASIC Sinclair que utilizemos en nuestros programas, y además, en mayúsculas.

De no hacerlo así, el editor considerará las entradas erróneas (no es lo mismo **PRINT** que **print**). Es de especial importancia seleccionar nombres de variables que no contengan entre sus letras palabras del BASIC. Por ejemplo, si tecleamos en una línea **LET CALOR=0**, el error estará garantizado, pues tratará de interpretar el grupo **OR** por separado, y por supuesto, no lo conseguirá.

Una forma de evitarlo consiste en asignar nombres de variables en minúsculas, y la otra, como ya hemos señalado, poner especial atención en no incluir dentro del nombre posibles comandos. Cada vez que separemos dos sentencias median-



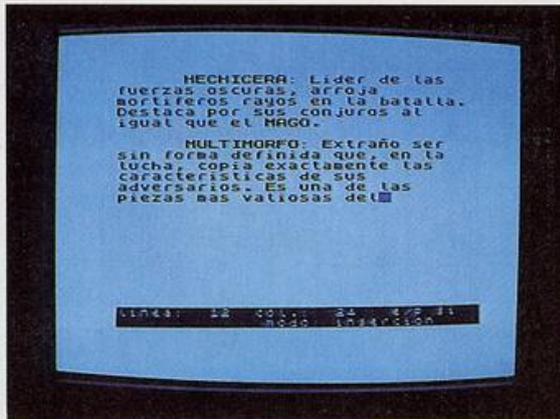
En el lateral izquierdo de la carcasa han sido instalados los conectores del casete EAR, MIC y el de impresora RS 232.

te dos puntos (:), el editor automáticamente provoca un salto a la siguiente línea. Por otra parte, si cometemos algún error al efectuar las entradas desde el teclado, en lugar del signo de interrogación (?) parpadeante de todos conocido sobre la posición donde éste ocurrió, ahora encontraremos una especie de cucaracha (*bug*, en su acepción inglesa).

Otra particularidad es que ya no existe el cursor de programa y salvo que nosotros indiquemos lo contrario ejecutando **LIST** o **EDIT** siempre nos encontraremos en la primera página de éste.

Una toma RGB ha sido habilitada para la conexión directa de un monitor en color.





Las dos líneas inferiores del procesador de textos incorporado en el sistema nos informa sobre línea y columna actuales, así como, del modo de escritura implícito.

Pulsando CMND en el teclado independiente o EDITAR en el principal, el cursor se traslada a la última línea de la pantalla. Esta puede emplearse, igualmente, para introducir programas, pero con la salvedad de limitar la entrada a 32 caracteres.

Aquí nos está permitido hacer uso de los nuevos comandos del editor:

- EDIT o E seguido por un número de línea muestra el programa a partir de ésta con el cursor preparado al principio de ella para comenzar la edición.
- NUM o N y a continuación dos números sepa-

El 128 K pretende venir a sustituir a su antecesor: el sorprendente Spectrum 48 K.



rados por una coma, indican al editor la nueva base y el incremento entre líneas elegido para efectuar la reenumeración de todo el programa BASIC almacenado en ese momento en la memoria. En este sentido se echa de menos un comando, el cual permitiera la reenumeración parcial, por bloques y no siempre total.

— BORRAR o B con dos parámetros numéricos a continuación eliminará del programa las líneas comprendidas entre ambos, incluidas éstas también.

En el teclado numérico auxiliar están previstas algunas funciones que permiten corregir los programas con cierta comodidad. A parte de cuatro teclas de cursor, las cuales facilitan el movimiento en todos los sentidos por el listado, es factible avanzar con una sola pulsación al principio o final de una misma línea o de la siguiente.

Dos teclas de borrado para eliminar el carácter situado bajo el cursor o el ubicado a su izquierda, más un duplicado de ENTER completan las teclas de edición de programas.

DISCO RAM

De los 128 Kbytes disponibles en este modelo, la cantidad aprovechable para la programación en BASIC sigue manteniéndose inalterada respecto a sus predecesores, es decir, casi 41 K. El resto de memoria adicional sólo es accesible desde código máquina.

Pero además, ofrecen una posibilidad adicional: la de servir de almacenamiento de datos y programas. Para hacer uso de ella deberemos encontrarnos en modo 128. Los comandos son los mismos que los de casete con el añadido del símbolo de admiración (!).

Una ventaja fundamental frente a otros soportes de información es la transferencia prácticamente instantánea de ésta y la total fiabilidad (no es necesario verificar la información una vez almacenada). Pero por contra perdemos su contenido cuando desconectamos el ordenador o pasamos al modo 48 K.

No obstante en programas que manejen un volumen apreciable de datos, el cual nos obligue a continuas operaciones de carga o grabación de éstos, el Disco RAM será una inestimable ayuda, siendo posible visualizar en cualquier momento su contenido mediante el comando **CAT!**.

El paso al modo 48 K se efectúa ejecutando como comando directo o dentro de un programa la orden **SPECTRUM**. Tras esto, el teclado auxiliar



queda deshabilitado y podemos desconectarlo si solamente vamos a trabajar en este sistema. A partir de entonces el ordenador se comporta como si de un Spectrum + se tratara.

El contenido de la memoria no se borra por esta operación, si bien todos los comandos especiales comentados anteriormente no serán operativos, y en algunos casos conducirán al mensaje de error "NO EXISTE EN BASIC". Tanto éste, como los restantes mensajes de la ROM han sido traducidos al castellano, y así, en vez del habitual *Scroll?* aparecerá ¿SIGO?, por ejemplo.

El paso 128-48 se realiza también tecleando letra a letra **RUN USR 0**, pero en este caso perderemos el contenido de la memoria. El salto inverso, es decir, de modo 48 a modo 128 no está permitido y nos veremos obligados a desconectar la alimentación o pulsar el botón de RESET para conseguirlo.

PROCESANDO TEXTOS

Entre las novedades incorporadas en el 128 se encuentra presente el acceso mediante el comando **EDIT**, y tras él una variable alfanumérica, al procesador de documentos residente en el sistema.

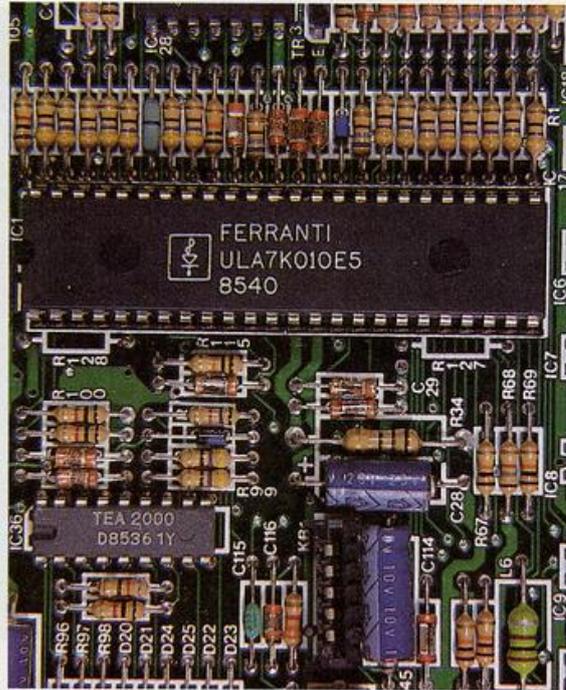
Los distribuidores anuncian entre sus virtudes la posibilidad de creación de largos documentos, la facilidad de corrección de errores y desplazamiento por todo el texto, así como la impresión de éste a base de cómodos comandos y manejando el teclado auxiliar.

Pues bien, en la prueba efectuada encontramos que cada línea contiene como máximo 32 caracteres. No es posible justificar los textos por la derecha. La búsqueda de palabras y el movimiento de bloques no se ha previsto.

Las vocales acentuadas no están en el juego de caracteres de serie. No obstante, es posible crearlas como gráficos definidos y teclearlas cuando sea necesario en este modo.

Si en un texto de 320 líneas (la mitad de la capacidad, aproximadamente, disponible en los procesadores para Spectrum comerciales), le ordenamos mediante la tecla correspondiente del teclado auxiliar que salte de la primera línea a la última, con cierta sorpresa comprobamos que esta operación se realiza en aproximadamente 56 segundos.

Cuando se trata de grabar el documento, la única opción posible es a cinta, pues el paso a Microdrive provoca el error NO EXISTE EN BASIC.

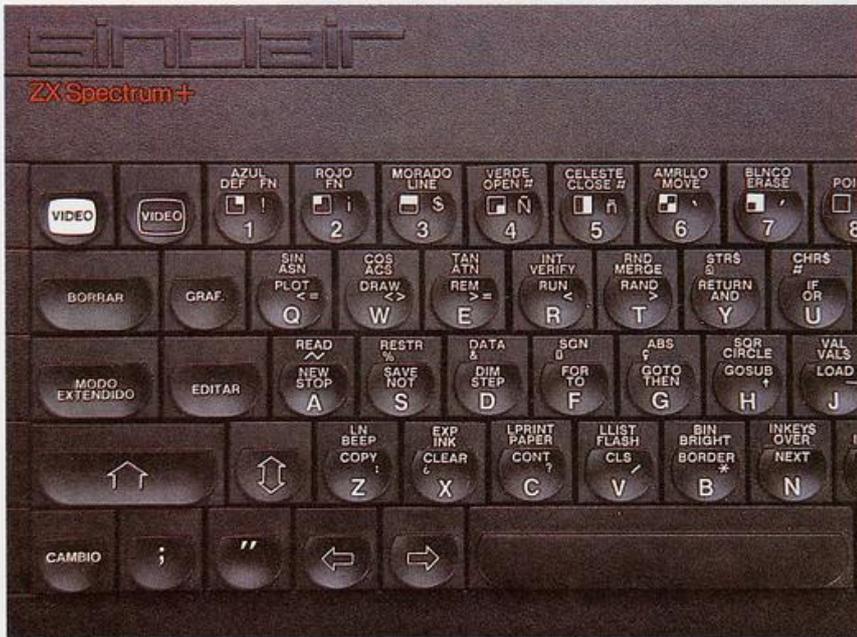


La nueva ULA de 48 pines: 8 más que en las versiones anteriores.

En resumen, en la creación de documentos cortos y con pocas pretensiones en cuanto a formato y longitud, su utilidad es dudosa, pero en el caso de grandes documentos la velocidad con que los trata lo convierte en inapropiado para esta labor.

El teclado auxiliar facilita la edición de programas y el control del procesador de textos.





Las inscripciones de muchas teclas ha sido traducida al castellano y entre los caracteres de este idioma aparecen las eñes.

Sorprende, por tanto, que en la publicidad se oferte como una alternativa en el campo del proceso de textos, cuando éste se haya implementado con tan baja calidad.

EL SONIDO EN EL 128

Frente al proceso de textos, donde nuestras quejas quedan más que justificadas, el 128 dispone

de un nuevo PSG (*Programmable Sound Generator*, generador de sonidos programable), el cual amplía con creces las posibilidades musicales ya conocidas de los modelos anteriores.

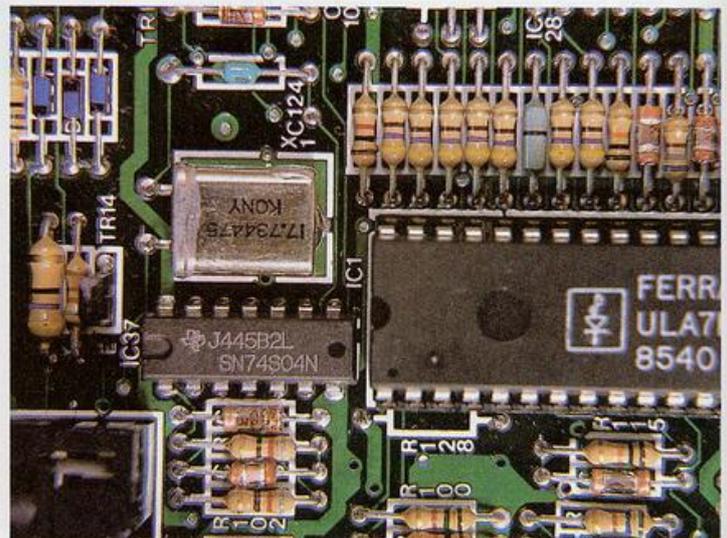
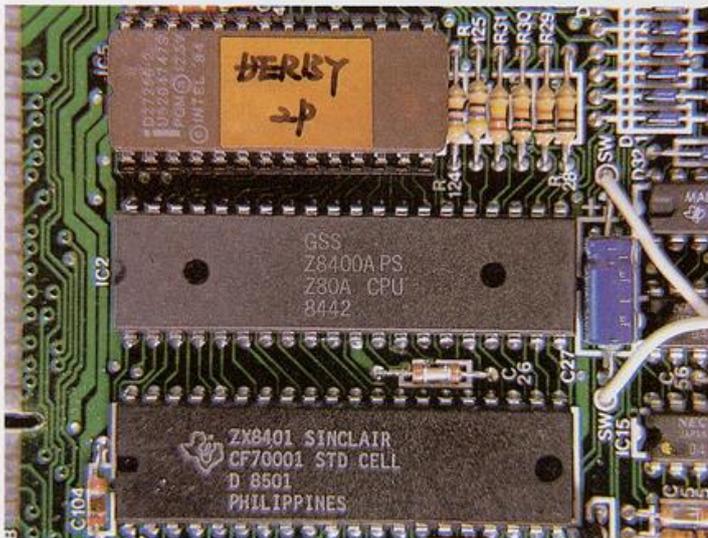
Desde BASIC es controlable mediante **BEEP** (en esta misma sección está recogida toda la información referente a él), y el nuevo comando **PLAY**, solamente operativo cuando nos encontramos en modo 128.

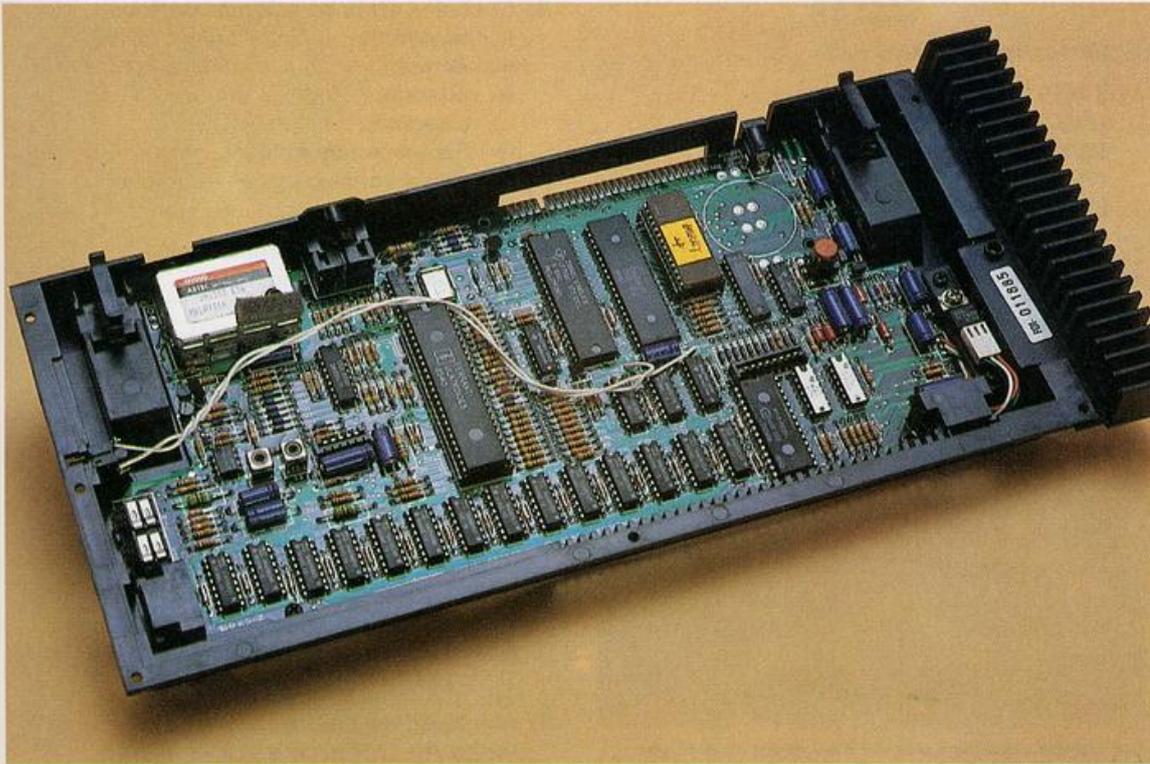
PLAY puede controlar simultáneamente tres canales de sonido independientes denominados A, B y C. La asignación de la música o efecto sonoro a cada canal se realiza mediante cadenas alfanuméricas, es decir, el formato general de este comando tiene el siguiente aspecto: **PLAY cadena1\$, cadena2\$, cadena3\$**.

En la construcción de cadenas actúan varios nuevos comandos, los cuales han sido resumidos en la tabla adjunta. Comentemos los más importantes:

- M permite la selección de canal para la emisión de tono, ruido o mezcla de ambos. Por defecto, los canales A, B y C están preparados como generadores de tono, pero según el valor situado tras el comando M, podemos modificar esta circunstancia efectuando combinaciones de los valores señalados en la tabla.
- Cada nota se obtiene asignándole el nombre reservado para ella según la escala de la figura.
- O selecciona la octava mediante el parámetro situado a continuación. Este puede variar de 0 a 8, siendo la más grave la marcada con 00 y la más aguda la 08.
- Un número entre 1 y 12 colocado antes de una nota establece su duración.
- N actúa de separador de comandos cuando la unión de éstos pueda dar lugar a confusión. Observemos la tabla de comandos y comprobaremos que además de los anteriores parámetros

El cristal de cuarzo ha sido sustituido por una nueva versión regulada a 17.734475 MHz.





Levantando la carcasa aparece ante nuestros ojos el nuevo circuito impreso y sus componentes.

destinados al control del volumen, velocidad de ejecución, repetición de grupos de notas, etc., confieren al Spectrum 128 unas posibilidades sonoras formidables.

CONCLUSIONES

Entre las virtudes del nuevo sistema quedan contrastadas sus posibilidades musicales o la mejora en el proceso de programas con gran volumen de datos de entrada/salida valiéndonos del Disco RAM.

El nuevo teclado auxiliar simplifica la edición de programas y la introducción de datos numéricos, así como el empleo del ordenador como calculadora. En este sentido, el sistema, tras una operación, retiene en su memoria el resultado por si el usuario necesita encadenarlo con el siguiente. Se echa de menos un BASIC ampliado que mejore la capacidad de proceso y la potencia operativa general del nuevo ordenador, así como, un medio de almacenamiento masivo de información de alta velocidad de acceso y fiabilidad.

La traducción de los mensajes de la ROM al castellano facilita al operador profano en inglés la comprensión de éstos, si bien, parece un síntoma de improvisación encontrar como traducción de "Out of screen", "FUERRA DE LA PANTALLA", donde efectivamente, se ha deslizado una "R" de más.

El procesador de textos incorporado constituye el punto más débil de todo el sistema, y por supuesto, quien desee emplear su nuevo ordenador en este campo, más vale que vaya pensando en alguno de los comerciales.

El tema de la total compatibilidad a nivel de software y hardware conviene ponerlo cuando menos en tela de juicio. En principio todo el software disponible para el Spectrum + es implementable en el 128 procesándolo en modo 48 K, aunque sólo el tiempo confirmará o rebatirá tal posibilidad.

A nivel de hardware, la cuestión es muy diferente, y todos aquellos periféricos que para su funcionamiento aprovechan la primitiva rutina de inicialización de la ROM son totalmente incompatibles. Tal es el caso del SOUND JOYSTICK INTERFACE de INVESTRONICA. El poseedor de uno de estos dispositivos, puede ir pensando en adquirir un nuevo modelo.

En resumen, un ordenador en el cual ha primado en su diseño la ampliación de las posibilidades lúdicas del Spectrum por aprovechamiento de una mayor capacidad de memoria, pero inadecuado para todo usuario que con el cambio de modelo pretenda trabajar bajo una óptica algo más profesional.

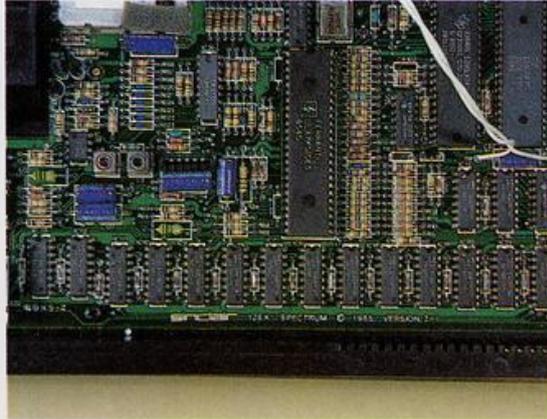


TABLAS

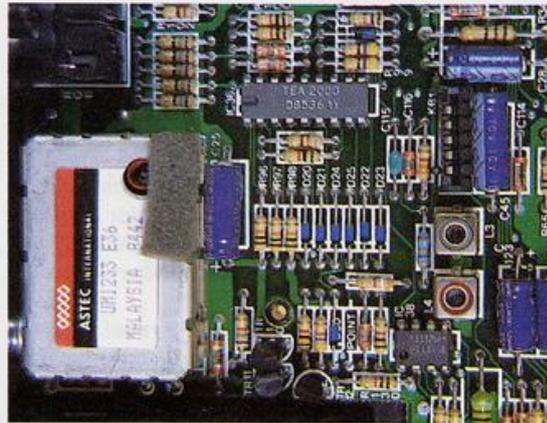
1) Comandos del Disco RAM

SAVE!"nombre"
 LOAD!"nombre"
 MERGE!"nombre"
 CAT!
 ERASE!"nombre"

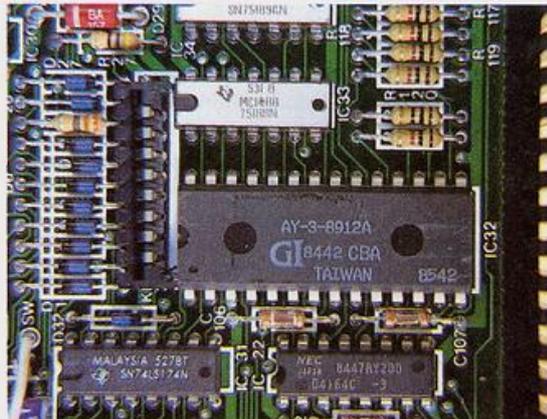
Los pequeños chips bajo la ULA soportan los 112 Kbytes de memoria RAM.



Los nuevos circuitos cerca del modulador se encargan de enviar las señales sonoras hacia el altavoz del televisor.



El chip AY-3-8912A es el nuevo PSG (Program Sound Generator) encargado de la producción de sonidos a través de tres canales independientes.



2) Símbolos en la creación de sonidos

) Sostenido
 \$ Bemol
 & Silencio
 — Ligaduras
 () Repetición de notas
 !! Comentarios

3) Comandos de sonido

M- Selección de canal.
 O- Selección de octava.
 1-12- Duración de las notas.
 N- Separador de comandos.
 V- Volumen de las notas (0-15).
 W- Selección de envolvente.
 X- Longitud de la envolvente.
 U- Selección de canales afectados por el generador de envolventes.
 T- Tempo (bpm o golpes por minuto).
 H- Corta las repeticiones indefinidas.
 a-g Escala inferior de notas.
 A-G Escala superior de notas.
 Y- Habilita la salida MIDI (1 a 16).
 Z- Cambia la programación del sintetizador.

5) Mensajes del casete

PREPARE LA CINTA Y PULSE ENTER
 PROGRAMA:
 MATRIZ NUM.:
 MATRIZ CARACT.:
 BYTES:

4) Mensajes del 128

0 OK
 1 NEXT SIN FOR
 2 VARIABLE INDEFINIDA
 3 INDICE ERRONEO
 4 SIN MEMORIA
 5 FUERRA DE PANTALLA
 6 NUMERO MUY ALTO
 7 RETURN SIN GOSUB
 8 FIN DE FICHERO
 9 SENTENCIA STOP
 A ARGUMENTO ERRONEO
 B ENTERO FUERA DE RANGO
 C NO EXISTE EN BASIC
 D BREAK/CONT SIGUE
 E FALTAN DATOS
 F NOMBRE/FICHERO ERRONEO
 G NO HAY SITIO
 H STOP EN INPUT
 I FOR SIN NEXT
 J DISPOSITIVO E/S ERRONEO
 K COLOR INVALIDO
 L BREAK EN PROGRAMA
 M RAMTOP MALA
 N SENTENCIA PERDIDA
 O CORRIENTE INVALIDA
 P FN SIN DEF
 Q PARAMETRO ERRONEO
 R ERROR DE CARGA
 ¿SIGO?



ENTRADA Y SALIDA



El sistema de comunicaciones que un microprocesador debe mantener con los periféricos que le rodean, es uno de los más importantes factores considerado por los fabricantes de un ordenador, cuando han de tomar la decisión de incorporarlo a sus equipos.

En el caso del Z 80, la interacción con el teclado, la ULA, la cual controla la imagen que parte hacia el televisor, el casete, las impresoras, etc. se encuentra satisfactoriamente gestionada a través de las rutinas almacenadas en la ROM de nuestro Spectrum.

Pero no olvidemos que cuando un microprocesador necesita enviar o recoger información destinada o procedente de un dispositivo exterior, lo hace siempre byte a byte a través de los canales habilitados a tal efecto del bus de datos.

Este como sabemos, está formado por 8 líneas, y a través de cada una de ellas puede circular un bit cada vez, es decir, hablamos de un bus de da-

tos de 8 bits o un byte, el cual podríamos decir que comienza en el microprocesador y termina en el puerto (port) habilitado para el periférico.

IN Y OUT

IN y OUT son las dos instrucciones preparadas por los fabricantes del Z 80 encargadas de gestionar cualquier transferencia de datos entre microprocesador y periférico. La CPU no necesita saber como funciona un dispositivo exterior a ella en detalle, sino tan solo, de donde (IN) o a donde (OUT) recoger o enviar la información que con él intercambia.

MNEMONICO	CODIGO MAQUINA	REGISTRO F								Nº BYTES	CICLOS		NOTAS
		7	6	5	4	3	2	1	0		MAQ.	RELOJ	
		S	Z		H		P/V	N	C				
DI	1 1 1 1 0 0 1 1	•	•	•	•	•	•	•	•	1	1	4	
EI	1 1 1 1 1 0 1 1	•	•	•	•	•	•	•	•	1	1	4	
IMO	1 1 1 0 1 1 0 1									2	2	8	
	0 1 0 0 0 1 1 0	•	•	•	•	•	•	•	•				
IM1	1 1 1 0 1 1 0 1									2	2	8	
	0 1 0 1 0 1 1 0	•	•	•	•	•	•	•	•				
IM2	1 1 1 0 1 1 0 1									2	2	8	
	0 1 0 1 1 1 1 0	•	•	•	•	•	•	•	•				
RETI	1 1 1 0 1 1 0 1									2	4	14	
	0 1 0 0 1 1 0 1	•	•	•	•	•	•	•	•				
RETN	1 1 1 0 1 1 0 1									2	4	14	
	0 1 0 0 0 1 0 1	•	•	•	•	•	•	•	•				

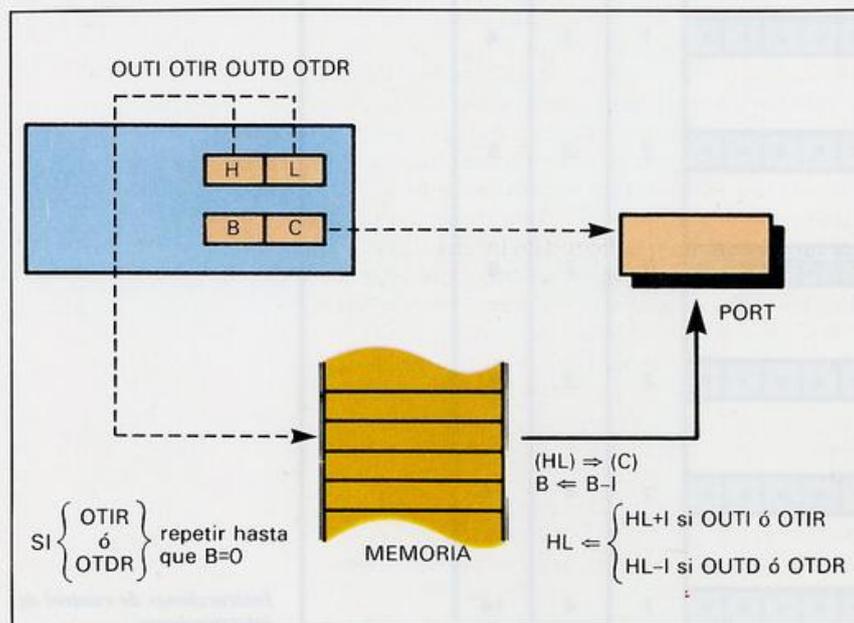
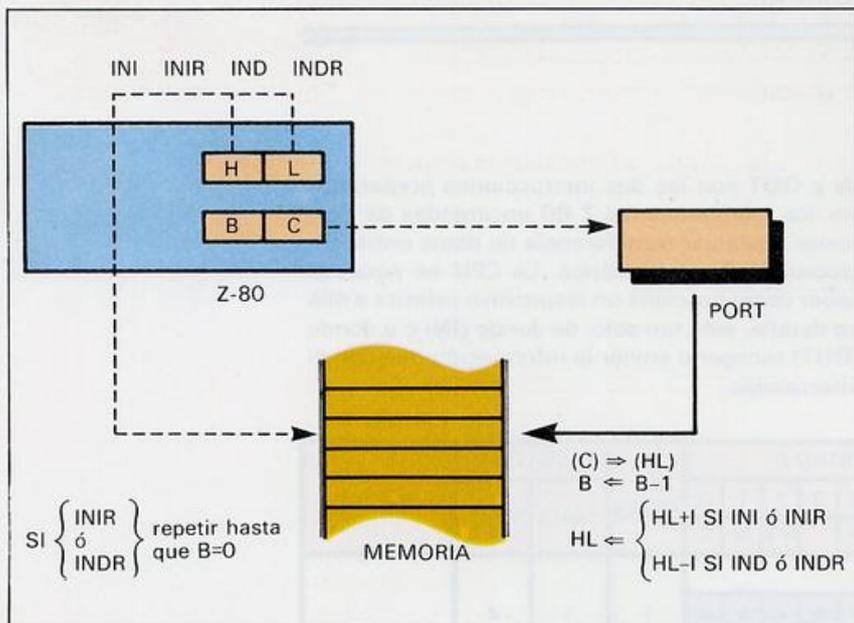
Instrucciones de control de interrupciones.

Este lugar es lo que antes hemos denominado puerto, o port en su acepción inglesa, y puesto que el Z 80 no le interesa como llega o sale de allí cada byte desde/hacia el periférico, envía o recoge un octeto, de uno en uno, cada vez que sea necesario. Y éste es, precisamente, el trabajo encomendado a las instrucciones **IN** y **OUT**. El formato más sencillo de las instrucciones de entrada (**IN**) de datos hacia el microprocesador es:

IN DESTINO, PORT

Mecanismo de las instrucciones de entrada y salida.

donde DESTINO puede ser cualquiera de los registros de 8 bits A, B, C, D, E, H o L y PORT, el puerto del cual nosotros necesitamos recavar la



información definido por el contenido del par BC en las instrucciones **IN r,(C)** y por el par **An** en **IN A,(n)**.

Respecto de las instrucciones de salida (**OUT**), podemos efectuar idénticas apreciaciones, pero considerando ahora que la operación es inversa, es decir, enviamos a un puerto determinado el byte almacenado en el registro **ORIGEN** del microprocesador. El formato será, por tanto:

OUT (PORT), ORIGEN

donde tanto **PORT** como **ORIGEN**, quedan definidos exactamente igual que en las instrucciones de entrada pero, por supuesto, el sentido de la transferencia de información es opuesto (del Z 80 al periférico).

De forma similar a lo que ocurría con las instrucciones de búsqueda y transferencia de bloques, las de entrada/salida pueden funcionar semiautomáticamente o de manera totalmente automática para recoger o enviar la información contenida o destinada a un determinado bloque de memoria desde/a un port específico.

En estos casos el par HL debe señalar al primer octeto del bloque (instrucciones **INI**, **INIR**, **OUTI** y **OTIR**) o al último (caso de **IND**, **INDR**, **OUTD** y **OTDR**) de donde o hacia donde recoger/enviar la información, puesto que en las primeras se incrementa de uno en uno y en las segundas se decreta en la misma cantidad cada vez que se realiza una lectura/escritura del port correspondiente.

El registro B es utilizado como contador, y en todas estas instrucciones se decreta en uno por cada lectura o escritura que se realice del puerto especificado por el registro C.

Si las instrucciones son de mecanismo automático, la transferencia de datos entre ordenador y periférico se ejecuta una y otra vez hasta que el contenido del registro B sea 0. En las figuras hemos representado esquemáticamente todos estos procesos.

INTERRUPCIONES

Una interrupción es una señal enviada por un periférico hacia el microprocesador con la intención de que éste abandone cualquier trabajo que estuviera realizando y atienda su solicitud.

Cuando esta circunstancia se produce, la CPU completa la ejecución de la instrucción en curso en ese instante y salta a una dirección de memoria donde encontrará almacenada la subrutina de

tratamiento de la interrupción, anotando previamente en el stack, donde paralizó el programa principal.

Una vez concluida la rutina de interrupción retorna al punto de salto, siempre que al final de esta hayamos incluido una instrucción de retorno, y continúa implementando el resto del programa. Sobre este particular conviene puntualizar que, en principio, bastaría colocar al final de la subrutina una instrucción de retorno de las ya conocidas, por ejemplo, RET. Pero solamente las de retorno de interrupción (RETI y RETN), garantizan que el microprocesador encontrará restablecidas las condiciones de partida previas, al tratamiento de la solicitud del periférico en cuestión.

En el Z 80, microprocesador del Spectrum, dos de sus patillas están dispuestas para recibir las peticiones de interrupción por parte de un dispositivo exterior.

El pin 17 denominado NMI (*Non Maskable Interruption*) se encargaría de recibir las interrupciones no enmascarables y hemos puntualizado «se encargaría», pues el Spectrum no utiliza esta línea.

El hecho es que las instrucciones NMI, de haberlas, gozarían de prioridad absoluta y sería imposible desactivarlas por parte del programador. El microprocesador saltaría siempre a la rutina ubicada a partir de la dirección 0066h (102 decimal) y ésta le indica que provoque la reinicialización del sistema sin llegar nunca a la instrucción RETN (retorno de interrupción no enmascarable) con la que finaliza dicha subrutina.

Visto así, parece que este tipo de interrupciones carece de utilidad práctica (prueba a ejecutar **RANDOMIZE USR 102**). Desde el punto de vista de la programación de nuestro Spectrum, probablemente sí.

Pero algunos periféricos de los existentes en el mercado son capaces de bloquear la ROM de forma que a partir de la dirección 0066h, el Z 80 en-

Códigos de las instrucciones de interrupción.

MNEMONICO	HEXA-DECIMAL	DECIMAL
DI	F3	243
EI	FB	251
IMO	ED 46	237, 70
IM1	ED 56	237, 86
IM2	ED, 5E	237, 94



cuentre una rutina diferente de la preprogramada en el sistema, adquiriendo de esta manera prioridad frente a cualquier otro dispositivo.

Protocolos.

INTERRUPCIONES ENMASCARABLES

Un periférico conectado con el pin 16 del Z 80, INT (*Interrupt request*) puede efectuar en cualquier momento una solicitud para ser atendido, pero a diferencia de las interrupciones que llegan a la patilla NMI, la petición será o no atendida en función de si previamente, por programa, hemos bloqueado o no estas solicitudes. De ahí, el nombre de interrupciones enmascarables.

Cuando conectamos el Spectrum a la red, las interrupciones se encuentran deshabilitadas y el sistema comienza a procesar las rutinas programadas en la ROM a partir de la dirección 0000. Por tanto, cualquier solicitud de interrupción será desatendida hasta que el microprocesador encuentre una instrucción EI (*Enable Interrupt*, habilitar las interrupciones), momento a partir del cual, podrán ser atendidas.

Es decir, cuando recuperamos el control sobre el sistema, tras inicializarse e imprimirse el mensaje de Sinclair, las interrupciones estarán habilitadas.

A partir de ese momento, podemos inhibirlas mediante la instrucción DI (*Disable Interrupt*, deshabilitar interrupciones), aunque otras operaciones, como los procesos de gestión del casete o la emisión de sonidos por órdenes BEEP, las deshabi-



litarán, pues el control del tiempo es parte fundamental de su funcionamiento y parece lógico evitar que el Z 80 sea molestado mientras los ejecuta.

MODOS DE INTERRUPCION

En el juego del Z 80 existen tres instrucciones IM (*Interruption Mode*) las cuales fijan otros tantos modos de interrupción: IM 0, IM 1 e IM 2.

Cuando un periférico interrumpe en modo 0 coloca en el bus de datos un byte el cual provoca que el microprocesador salte a alguna de las primeras direcciones de la página cero. En el Spectrum, este modo no se utiliza, salvo en el caso de un RESET, tras el cual, la CPU comienza a procesar la rutina de inicialización.

El modo 1 es similar al anterior, pero en este caso la dirección de salto es constante (0038h). Es el

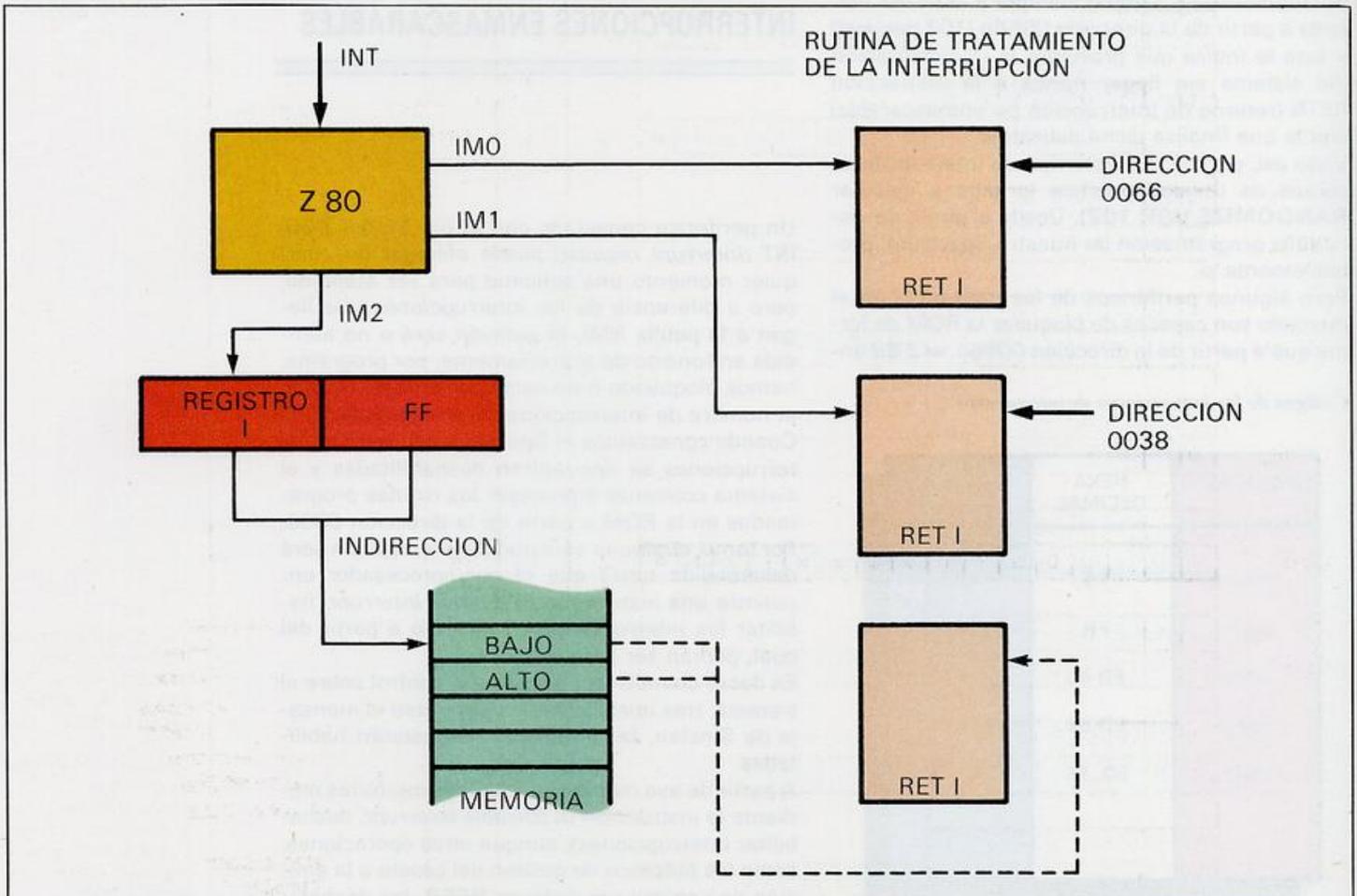
modo habitual en el Spectrum, pues a partir de dicha dirección se encuentra la rutina de tratamiento del teclado, y como ya sabemos, la ULA cursa 50 veces cada segundo una solicitud de interrupción para que este sea examinado.

Finalmente, mediante la instrucción IM 2, se accede al modo de interrupción 2 el cual, desde el punto de vista de la programación es el más interesante para nosotros. Cuando una solicitud es efectuada al microprocesador por este sistema, el Z 80 conforma una dirección absoluta de salto entre el contenido del registro I (parte alta) y el byte entregado por el periférico (parte baja) al bus de datos.

La dirección así formada (byte colocado en el bus de datos + 256 × contenido de I) señala a una posición de memoria determinada. Ahora la CPU lee el contenido de ésta y de la siguiente celda, y con estos dos octetos forma la dirección absoluta donde encontrar la subrutina de tratamiento de la interrupción (1.ª posición + 256 × siguiente), método conocido técnicamente como de «indirección».

Cuando en un programa nosotros hacemos uso de este mecanismo de interrupción, al no existir el periférico que lo solicite, el Spectrum deposita en el bus de datos, siempre el valor FFh. Como

Los modos de interrupción.



vemos, el valor que nosotros asignemos al registro I es determinativo.

En principio, su valor podrá ser cualquiera con tal que la posición de memoria obtenida contenga los valores que finalmente conformarán la dirección efectiva de tratamiento de la interrupción. Como demostración hemos preparado una pequeña rutina la cual pasa del modo IM1, habitual en

el Spectrum, al IM2. Cada vez que se genera una interrupción para ser analizado el teclado, se producen ahora unos efectos de borde a través del puerto 254 (FEh).

Una vez en marcha, tecleemos comandos del tipo **BEEP 2,-5: PAUSE 20: BEEP 3,0: PAUSE 30: BEEP 4,10** y comprobaremos como durante su ejecución las interrupciones son deshabilitadas.

El grupo de entrada/salida.



MNEMONICO	CODIGO MAQUINA	REGISTRO F								Nº BYTES	CICLOS		NOTAS																	
		7	6	5	4	3	2	1	0		MAQ.	RELOJ																		
		S	Z		H		P/V	N	C																					
INA,(n)	1 1 0 1 1 0 1 1									2	3	11	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>	REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111	
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
x		n			x	•	•	x	•	x	•	•		•																
INr,(c)	1 1 1 0 1 1 0 1									2	3	12		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>	REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
0	1	r	r	r	0	0	0	↓	↓	x	↓	x	p		0	•														
INI	1 1 1 0 1 1 0 1									2	4	16	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>		REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
								①																						
1	0	1	0	0	0	1	0	x	↓	x	x	x		x	x	1	•													
INIR	1 1 1 0 1 1 0 1									2	5/4	21/16		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>	REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
1	0	1	1	0	0	1	0	x	1	x	x	x	x		x	1	•													
IND	1 1 1 0 1 1 0 1									2	4	16	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>		REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
								①																						
1	0	1	0	1	0	1	0	x	↓	x	x	x		x	x	1	•													
INDR	1 1 1 0 1 1 0 1									2	5/4	21/16		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>	REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
1	0	1	1	1	0	1	0	x	1	x	x	x	x		x	1	•													
OUT(n),A	1 1 0 1 0 0 1 1									2	3	11	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>		REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
x			n			x	•	•	x	•	x	•		•	•	•														
OUT(c),r	1 1 1 0 1 1 0 1									2	3	12		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>	REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
0	1	r	r	r	0	0	1	•	•	x	•	x	•		•	•														
OUTI	1 1 1 0 1 1 0 1									2	4	16	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>		REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
								①																						
1	0	1	0	0	0	1	1	x	↓	x	x	x		x	x	1	•													
OTIR	1 1 1 0 1 1 0 1									2	5/4	21/16		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>	REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
1	0	1	1	0	0	1	1	x	1	x	x	x	x		x	1	•													
OUTD	1 1 1 0 1 1 0 1									2	4	16	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>		REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
								①																						
1	0	1	0	1	0	1	1	x	↓	x	x	x		x	x	1	•													
OTDR	1 1 1 0 1 1 0 1									2	5/4	21/16		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th>REG</th> <th>r</th> </tr> <tr> <td>B</td> <td>000</td> </tr> <tr> <td>C</td> <td>001</td> </tr> <tr> <td>D</td> <td>010</td> </tr> <tr> <td>E</td> <td>011</td> </tr> <tr> <td>H</td> <td>100</td> </tr> <tr> <td>L</td> <td>101</td> </tr> <tr> <td>A</td> <td>111</td> </tr> </table>	REG	r	B	000	C	001	D	010	E	011	H	100	L	101	A	111
	REG	r																												
B	000																													
C	001																													
D	010																													
E	011																													
H	100																													
L	101																													
A	111																													
1	0	1	1	1	0	1	1	x	1	x	x	x	x		x	1	•													

1 Significa que el indicador Z será colocado a 1, si 3-1 es cero; en caso contrario será puesto a 0. Cuando en CICLOS se dan dos cantidades la primera indica ciclos si B≠0 y la segunda ciclos cuando B=0.



EMBOSCADA



mboscada es un juego de habilidad con el cual pasaremos unos momentos agradables, sin duda una buena despedida.

propina. De esta forma es imposible finalizar la partida con cero puntos.

El objetivo del juego es dirigir una calabaza mágica, porque si no añadimos lo de mágica no sabemos como justificar que una calabaza ande sola, por el recinto en el cual está prisionera, sin que tropiece ni resulte atrapada por los cuadros centelleantes que irán surgiendo a su paso. Para hacer más sencilla esta misión, cada trescientos puntos desaparecerán todos los obstáculos, recomenzando el ciclo.

Para el desplazamiento de nuestra calabaza se utilizarán las siguientes teclas:

- "O" "Izquierda"
- "P" "Derecha"
- "Q" "Arriba"
- "A" "Abajo"

INTRODUCCION DEL PROGRAMA

A la hora de introducir el listado del programa, debemos destacar que los caracteres que figuran en el listado con subrayado simple, corresponden a los caracteres gráficos de las teclas afectadas,

i!

A la hora de introducir el listado del programa, debemos destacar que los caracteres que figuran en el listado con subrayado simple, corresponden a los caracteres gráficos de las teclas afectadas, y los de doble subrayado, a las mayúsculas de los caracteres gráficos correspondientes.

*

Para la ejecución automática del programa debemos grabarlo mediante la utilización del siguiente comando: **SAVE "EMBOSCADA" LINE 1**. Si prefiriésemos la grabación sin autoejecución teclearíamos la siguiente orden **SAVE "EMBOSCADA"**.



Para hacer más sencillo el juego, cada 300 puntos desaparecerán todos los obstáculos, recomenzando el ciclo.

Nuestra calabaza posee tres vidas, gracias a lo cual es posible conseguir una buena puntuación. Para aquellos que sean extremadamente malos en este tipo de juegos, cada vez que aparezca una calabaza, o por cada punto se le anotarán diez de



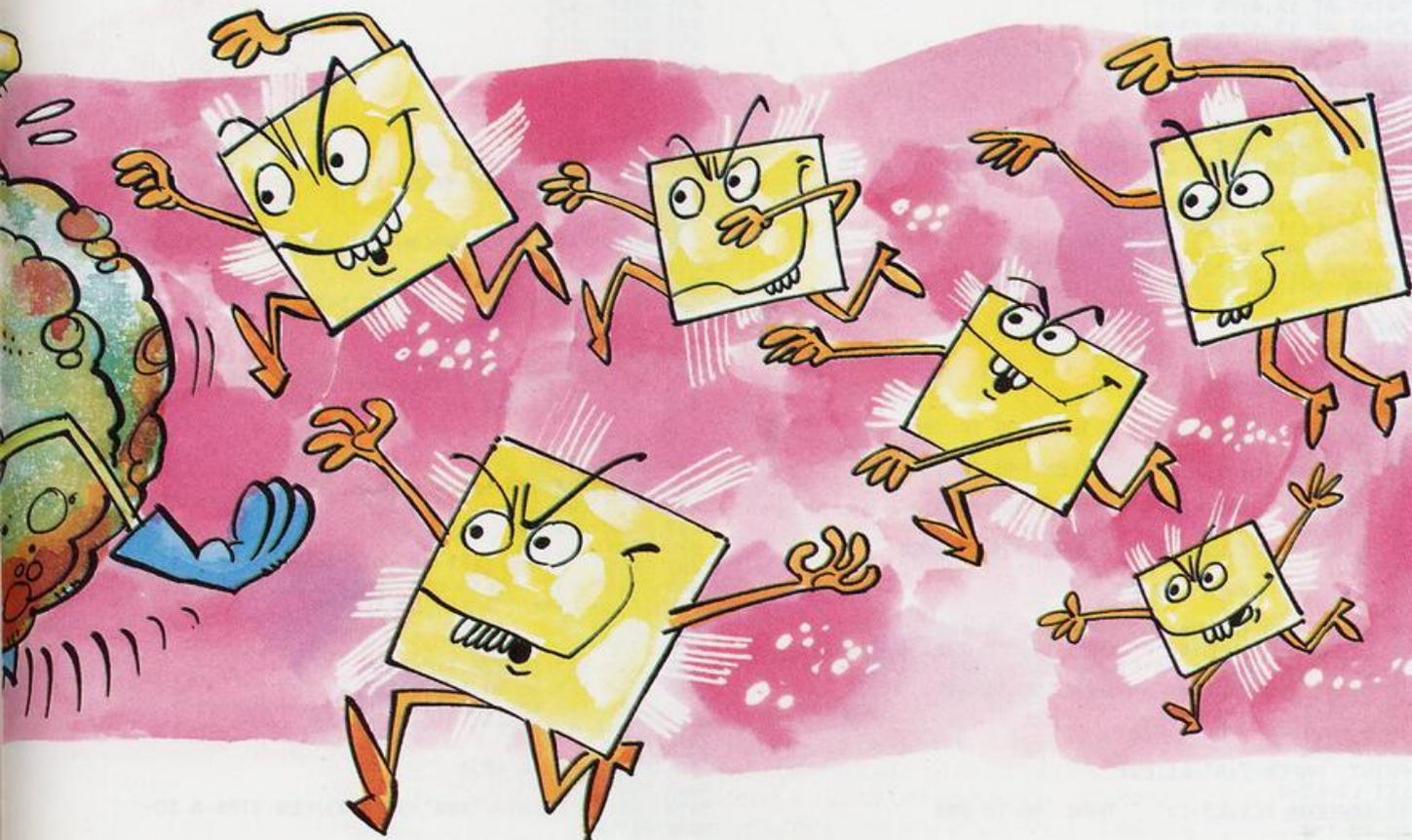
y los de doble subrayado, a las mayúsculas de los caracteres gráficos correspondientes.

Para la ejecución automática del programa debemos grabarlo mediante la utilización del siguiente comando: **SAVE "EMBOSCADA" LINE 1**. Si prefiriésemos la grabación sin autoejecución teclearíamos la siguiente orden **SAVE "EMBOSCADA"**.

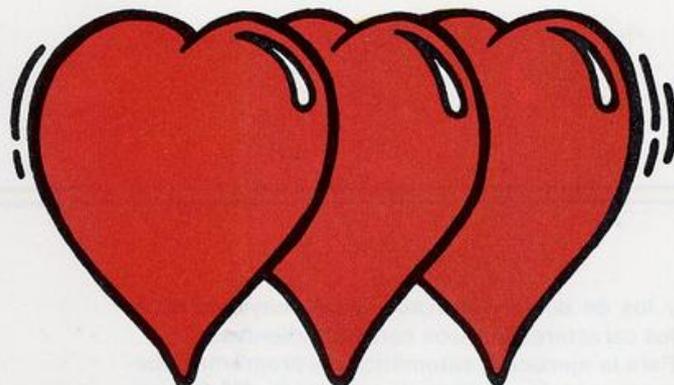
En cuanto a los literales **INV.** y **TRUE** que aparecen subrayados y entre corchetes, corresponden a los caracteres de control de video inverso o **INVERSE VIDEO (CAPS SHIFT + 4)** y video normal o **TRUE VIDEO (CAPS SHIFT + 3)**.



El objetivo del juego es dirigir la calabaza por el recinto en el cual está prisionera, sin que tropiece ni resulte atrapada.



PROGRAMA



Nuestra calabaza posee tres vidas, gracias a lo cual es posible conseguir una buena puntuación.

```

1 REM *****
2 REM * C.E.IGLESIAS MTNEZ. *
3 REM *****
4 REM * ATRAPADA (C) 1985 *
5 REM *****
6 REM * ADAPTACION: *
7 REM *****
8 REM * J.M.MAYORAL SERRANO *
9 REM *****
10 REM * EMBOSCADA (C) 1986 *
11 REM *****
15 POKE 23658,8
20 LET CONTA=0
30 GO SUB 5000
40 CLS
41 LET R=0
42 LET V$=" A A A "
50 INK 6: PAPER 1: BORDER 1
51 CLS
52 LET P=0
53 LET V=3
60 CLS
61 PRINT AT 9,3;"PUNTOS":
62 PRINT AT 10,4;"0000"
63 PRINT AT 12,3;"RECORD"
64 PRINT AT 13,4;"0000"
70 PRINT PAPER 7; INK 5; AT 3,13;"43333333333333333333"
7.
90 FOR F=4 TO 19
91 PRINT INK 5; PAPER 7; AT F,13;"5"
5.
92 NEXT F
93 PRINT PAPER 7; INK 5; AT 20,13;"13333333333333333333"
32.
110 PRINT AT 2,2; INK 5;"EMBOSCADA"; TAB 18; V$
120 PLOT 14,150
121 DRAW 75,0
122 DRAW 0,12
123 DRAW -75,0
124 DRAW 0,-12
130 PRINT AT 2,18;" "; AT 2,18; V$( TO 2*V)
140 LET L1=11
150 LET L2=21
160 PRINT AT 10,4; FN P$(P)
170 PRINT AT 13,4; FN P$(R)
180 PRINT INK 2; PAPER 7; AT L1,L2;"A"
181 BEEP .1,0
182 BEEP .1,7
190 LET P=P+10
191 PRINT AT 10,4; FN P$(P)
200 IF R<P THEN LET R=P: PRINT AT 13,4; FN P$(R)
210 IF CONTA=100 THEN LET CONTA=0: RESTORE : GO TO 60
60
220 LET V1=L2+INT (RND*-3)+2
230 LET V2=L1+INT (RND*-3)+2
240 IF V2=L1 AND V1=L2 THEN GO TO 220
245 IF ATTR (V2,V1)=244 THEN GO TO 220
250 PRINT BRIGHT 1; PAPER 6; INK 4; FLASH 1; AT V2,V
1;"B"
251 BEEP .05,7
252 BEEP .05,4
253 BEEP .05,7
254 LET CONTA=CONTA+1
255 GO SUB 600
260 IF SCREEN$ (L1-1,L2)<>" " THEN GO TO 280
270 GO TO 330
280 IF SCREEN$ (L1+1,L2)<>" " THEN GO TO 300
290 GO TO 330
300 IF SCREEN$ (L1,L2+1)<>" " THEN GO TO 320
310 GO TO 330
320 IF SCREEN$ (L1,L2-1)<>" " THEN GO TO 3480
330 LET C=1
335 IF INKEY$="" THEN GO TO 335
336 RESTORE
340 IF L2>15 AND INKEY$="0" THEN GO TO 400
350 IF L2>28 AND INKEY$="P" THEN GO TO 420
360 IF L1>5 AND INKEY$="Q" THEN GO TO 440
370 IF L1<18 AND INKEY$="A" THEN GO TO 460
380 LET C=C+1
381 IF C=10 THEN GO TO 220
390 GO TO 340
400 PRINT PAPER 7; AT L1,L2;" "
401 LET L2=L2-1
402 IF SCREEN$ (L1,L2)<>" " THEN GO TO 480
410 GO TO 180

```

```

420 PRINT PAPER 7; AT L1,L2;" "
421 LET L2=L2+1
422 IF SCREEN$ (L1,L2)<>" " THEN GO TO 480
430 GO TO 180
440 PRINT PAPER 7; AT L1,L2;" "
441 LET L1=L1-1
442 IF SCREEN$ (L1,L2)<>" " THEN GO TO 480
450 GO TO 180
460 PRINT PAPER 7; AT L1,L2;" "
461 LET L1=L1+1
462 IF SCREEN$ (L1,L2)<>" " THEN GO TO 480
470 GO TO 180
480 FOR F=0 TO 7
481 PRINT PAPER 7; INK F; AT L1,L2;"A"
482 PAUSE 10
483 NEXT F
484 PRINT AT L1,L2; BRIGHT 1; FLASH 1;"B"
485 LET V=V-1
490 BEEP .3,7
491 BEEP .1,7
492 BEEP .1,7
493 BEEP .1,7
494 BEEP .1,7
495 BEEP .1,9
496 BEEP .2,7
497 BEEP .1,12
498 LET CONTA=0
500 IF V THEN GO TO 60
510 FOR F=15 TO -15 STEP -1
520 BEEP .01,F
530 NEXT F
540 PRINT AT 11,15; BRIGHT 1; INVERSE 1;"PULSA UNA T
ECLA"
550 RESTORE
560 IF INKEY$="" THEN GO TO 560
565 LET P=0
570 GO TO 50
600 REM [INV_] SBR.COMPROBACION [TRUE]
610 IF ATTR (L1-1,L2)=244 AND ATTR (L1+1,L2)=244 AND
ATTR (L1,L2-1)=244 AND ATTR (L1,L2+1)=244 THEN GO T
O 630
620 RETURN
630 FOR N=1 TO 10
640 BEEP .05,30
650 BEEP .05,35
660 BEEP .05,40
670 NEXT N
680 PRINT FLASH 1; AT 21,13;" ESTAS ACORRALADO "
690 PRINT #0;" PULSA UNA TECLA PARA EMPEZAR"
700 IF INKEY$="" THEN GO TO 700
710 GO TO 50
720 STOP
5000 DATA 102,24,126,219,255
5010 DATA 189,66,60,0,126,126
5020 DATA 126,126,126,126,0
5030 FOR F=0 TO 15
5040 READ A
5050 POKE USR "A"+F,A
5060 NEXT F
5070 DEF FN P$(A)=( "000"+STR$(A))(LEN STR$(A) TO )
5080 RETURN

```