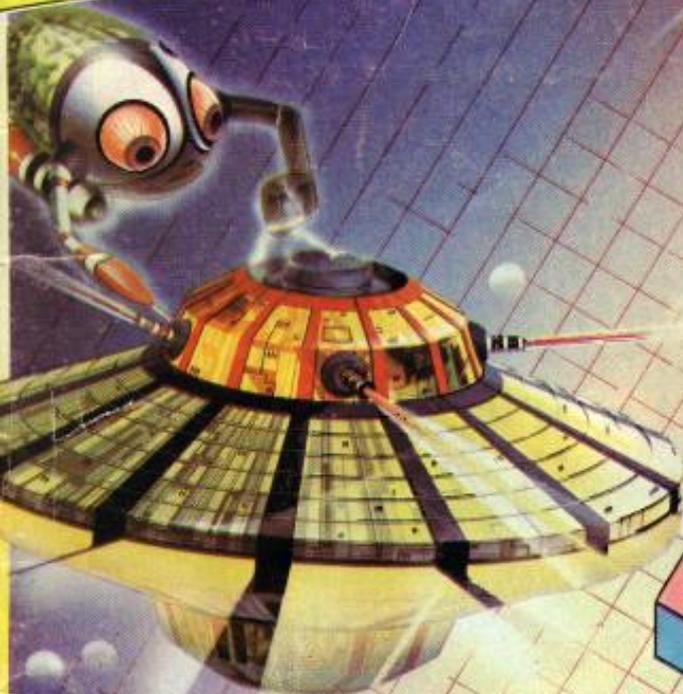


# ZX

REVISTA PARA LOS USUARIOS  
DE ORDENADORES SINCLAIR



**Construya  
su propio  
juego**



# YA ESTAN AQUI LOS REFUERZOS



Te presentamos  
dos auténticos refuerzos  
para obtener mayor  
rendimiento de tu Spectrum.

## EL INTERFACE 1 Y EL MICRODRIVE.

**¡Por fin  
podrás grabar y leer  
información de manera casi instantánea!**



**MICRODRIVE**  
Sólo SINCLAIR podía crear  
para su Spectrum el  
**MICRODRIVE ZX**.  
Todas las ventajas de los dis-  
cos "jumpy" a un precio cuatro  
veces inferior (y en mucho menos  
espacio).

para su Spectrum:

- Mensaje Cartuchos de 85 K con un tiempo medio de acceso de 3,5 segundos.
- Un programa de 48 K que tardaría varios minutos en cargarse mediante una cassette, se puede cargar desde el Cartucho en sólo 9 segundos.

El **INTERFACE 1** se suministra con el conector para los **MICRODRIVES**, un cable para la **RED ZX** y el manual en castellano.  
Todos los **MICRODRIVES** tienen un **Cartucho** de demostración que puede ser posteriormente borrado y utilizado. También puedes adquirir **Cartuchos vírgenes** para tus **MICRODRIVES**.



### INTERFACE 1

- Incluye los siguientes dispositivos:
- Un controlador que permite la conexión de hasta ocho **MICRODRIVES**, consiguiéndose, así, una memoria de hasta 680 K.
  - Una salida **RS 232** para conexión de impresoras profesionales o otro tipo de comunicaciones.
  - Una salida para **RED ZX**, que permite la conexión de hasta 64 Spectrum entre sí, compartiendo juegos, informaciones, impresoras, etc.
- Se atornilla a la parte inferior del Spectrum (elevándose unos centímetros por detrás, quedando el teclado en una posición más cómoda).



VISTA POSTERIOR DEL INTERFACE 1

### IMPORTANTE:

Al adquirir el **Interface 1**, y los **MICRODRIVES**, exige la **TARJETA DE GARANTIA INVESTRONICA**, única válida en todo el territorio nacional y llave para cualquier resolución de duda o reparación.

**INVESTRONICA** no prestará ningún servicio técnico a todos aquellos aparatos que carezcan de la correspondiente garantía.

**DE VENTA EN  
CONCESIONARIOS  
AUTORIZADOS.**



DISTRIBUIDOR  
EXCLUSIVO:

**INVESTRONICA**

CENTRAL COMERCIAL: Torrelavega, 60  
Tel. 484 05 007 - Fax: 231841 P.O. E. Madrid  
C/EL FORTALEZADO CAYULUM, Camp. 80 - Barcelona - 22

AÑO I - No. 6 / MAYO / 1984.

# ZX

REVISTA PARA LOS USUARIOS  
DE ORDENADORES SINCLAIR

Tenemos que comenzar por una confesión. A menudo quienes hacemos esta revista nos sentimos desbordados por la positiva reacción de nuestros lectores. Las cartas, las llamadas telefónicas, los ofrecimientos espontáneos de colaboración son otras tantas muestras de que ZX ha calado hondo entre los aficionados de nuestro país. Otra muestra significativa es la cantidad de programas que recibimos cada día para concursar. Y aunque quisiéramos contentar a todos no podríamos, por falta de espacio.

Al mismo tiempo, como no queremos dejar de ofrecer los artículos didácticos que muchos nos reclaman, hemos tenido que adoptar una decisión salomónica. Para dar cabida al primer capítulo del serial "Construya usted su propio programa", escrito por Fernando García, no tenemos más remedio a partir de este número que alternar la publicación de los dos seriales que veníamos ofreciendo: Aprendiendo Código Máquina, de Juan Martínez Velarde, y los Gráficos en el Spectrum, de Christian Battle. Como Juan y Christian son compañeros de estudios, no ha habido problemas para ponerlos de acuerdo en esta fórmula de alternancia. Esperamos que los lectores también estén de acuerdo.

Gracias a todos y hasta el próximo número.

## EN PORTADA

Iniciamos aquí otro serial didáctico que esperamos interese a un gran número de nuestros lectores. Se trata de un cursillo práctico de programación de juegos en el Spectrum. Contiene una parte genérica y otra más específicamente destinada a los usuarios de este ordenador. No sabemos todavía cuánto espacio nos ocupará en números sucesivos, pero estamos seguros de que estará entre las páginas más leídas de esta revista.

Página 4

## PROGRAMAS

Hay programas para todos. Juegos y aplicaciones. Largos y cortos. Simples y complicados. Para el Spectrum, pero también para el ZX81. Y, como prometíamos en nuestro número anterior, añadimos algunos bingos que completan la serie de los escogidos entre los muchos enviados por nuestros lectores.

Página 25

## LECTORES

ZX se nos está quedando pequeña para atender todas las consultas de nuestros lectores. Cartas de todo tipo llegan cada día a nuestra redacción.

Página 10

## MONTAJES

Reanudamos esta sección publicando un interesante circuito para los amantes de "cacharrear". Se trata de una variante de joystick, que utiliza pulsadores.

Página 20

## LIBROS

Los dos libros que comentamos este mes son radicalmente diferentes. Uno es, que sepamos, la primera recopilación de juegos para el Spectrum que se publica en España en forma de libro. El otro, un eficaz manual para los usuarios del ZX81.

Página 19

## IDEAS

Un solo tema —pero con muchas ideas dentro— ocupa esta sección del mes de mayo: la protección de programas. De ineludible lectura.

Página 16

## SOFTWARE

Juan Martínez Velarde continúa con su erudito tratamiento del código máquina del Spectrum. Esta vez, inicia el tema del movimiento. De manera que quienes hayan seguido con atención los capítulos anteriores de este serial, tendrán ya elementos para meterse bien dentro del ordenador y de sus posibilidades de programación.

Página 58

ZX es una publicación de Ediciones y Suscripciones, S. A. • Presidente: Fernando Bolín • Bravo Murillo, 377, 5.º A. Telf. (91) 733 74 13. Madrid-20. • Director Editorial: Norberto Gallego • Redacción: Alejandro Diges, Aníbal Pardo, Simeón Cruz, Gumersindo García, Juan Arencibia • Diseño: Ricardo Segura • Administración Gerente de Circulación y Ventas: Luis Carrero • Suscripciones: Antonio Zurdo (91) 733 79 69. • Producción: Miguel Onieva • Publicidad Madrid: Nieves Fernández. Telf. (91) 733 96 62 • Publicidad Barcelona: Enrique Alier, M. Carmen Ríos. Pelayo, 12, Barcelona - 1. Telf. (93) 301 47 00 • Distribuye: Sociedad General Española de Librería. Avda. Valdelaparra, s/n. Alcobendas, Madrid • Imprime: Heroes, S. A. Torrelara, 8. Madrid-16 • Depósito Legal: M. 37.432-1983.

# Construya su p



No vamos a empezar el artículo explicando qué es un juego, cuestión innecesaria, pero sí fijando algunas ideas sobre su implementación en los ordenadores y a partir de ese punto desarrollar las demás ideas.

Ya con la aparición de los primeros ordenadores comerciales (en los lejanos sesenta) muchos programadores empezaron a hacer programas como medio de diversión. Debido a las limitaciones de las máquinas estos juegos eran bastante sencillos y no tenían gráficos, sonido o cualquiera de las características a las que estamos acostumbrados hoy en día. Entre estos primeros programas podemos destacar el **STAR TREK** (el primer juego, o casi) que todo ordenador que se preciese debía tener. Asimismo surgió un programa llamado **ADVENTURE** (aventura) y que dio nombre a un tipo de juegos que más adelante explicaremos.

Esos programas y los que le siguieron pusieron la piedra básica de los juegos por ordenador: utilizarlo como máquina pensante para que nos pudiese problemas y situaciones totalmente imprevisibles que había que resolver.

Con la llegada del microprocesador en la década de los setenta empezaron a surgir los ordenadores personales, or-

denadores que cualquiera podía tener en su casa y que tenían características distintas a las de los *mainframes* (grandes ordenadores). Muchos disponían de gráficos y sonido, trabajaban en tiempo real, es decir, siempre le hacían caso al usuario, no como en las grandes máquinas que al tener que hacer caso a muchas personas solían dejar al usuario "colgado" a la espera de los datos que le había pedido. La aparición de los primeros juegos para ordenador personal tuvo un gran éxito que no ha parado hasta hoy día.

## Distintos tipos de juegos

Con la aparición de los microordenadores con capacidad de gráficos y sonidos surgió una gran cantidad de juegos que aprovechaban estas características unidas a la gran velocidad que proporcionaba la máquina, ofreciendo juegos con una gran dinámica en los que la persona se tenía que enfrentar a unos oponentes tontos pero muy veloces y en superioridad numérica. Son los tipos de juego conocidos como *marcianitos* (en inglés se les llama **ARCADE**) y que podemos ver en cualquier bar, esta denominación es genérica, aunque luego el programa no sea de marcianos, sino de bolas que comen puntos, tanques o cualquiera de las múltiples variaciones sobre el tema.

En contraposición a estos juegos existen los juegos inteligentes. En estos se aprovecha la capacidad pensante de la máquina para ofrecer un oponente de la misma categoría que el ser humano, al que muchas veces sustituye (ajedrez, otello, etc.). Aquí ya no prima la rapidez, sino la capacidad pensante y la inteligencia, por algo la rama de la informática denominada "inteligencia artificial" estudia estos juegos con gran interés.

Por último existe un tercer tipo de juegos denominados de aventura (recuerden el apartado anterior) en los que el jugador debe resolver algún problema (encontrar un tesoro, destruir a un enemigo, encontrar a un asesino, etc.) para lo cual puede ejecu-

tar ciertas instrucciones como puede ser el mirar cosas, coger objetos y hablar con otras personas, asimismo se puede desplazar por los distintos lugares en que sucede la acción, que en la mayoría de los casos no caben a la vez en la pantalla del televisor. Este tipo de juegos es el punto intermedio de los dos anteriores, la velocidad no es fundamental, aunque se puede hacer que sea en tiempo real y el jugador debe desarrollar más inteligencia que habilidad para poder ganar, porque estos juegos tienen final como los de "inteligencia", al contrario que los "arcade" en los que los enemigos son infinitos.

Naturalmente estas tres clasificaciones no son estrictas, pudiendo haber juegos que sean mezcla de estos tipos, o que no corresponda a ninguno (bastante raramente) y no por eso tienen que ser malos, al contrario las innovaciones siempre son buenas.

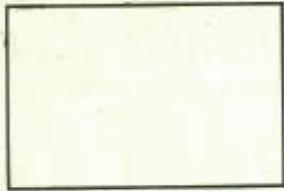
## Vamos a hacer un juego

Una vez que hemos decidido hacer un juego conviene que nos sentemos tranquilamente con papel y lápiz y una buena dosis de café (el ordenador de momento conviene dejarlo guardado). Ante todo debe fijarse bien el objetivo a conseguir: ¿una partida de damas?, ¿los típicos marcianitos?, o quizás una aventura en una isla solitaria imitando a **Robinson Crusoe**? Una vez decidido esto debemos recordar que hacer un juego no es un juego. Aunque esta frase parezca extraña, es real, una de las cosas más difíciles de programar es un juego. Mientras que en cualquier otro tipo de programas sabemos exactamente lo que hay que hacer y cómo se debe hacer, reduciéndose el trabajo a traducir esas instrucciones a un lenguaje comprensible por el ordenador; en un juego además de esto debemos: decidir cuál es el objetivo, de qué medios dispone el jugador para conseguirlo, cómo van a ser sus oponentes, crear la estrategia de éstos, cómo se van a mostrar los datos por pantalla y un largo etcétera de problemas que en otro sitio no se nos presentarían. Una vez

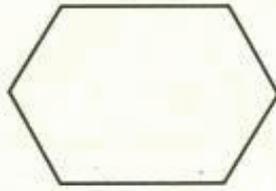
# propio juego

CAPITULO

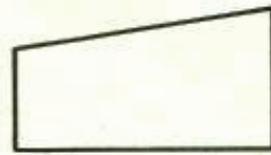
1



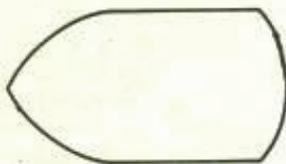
proceso simple.



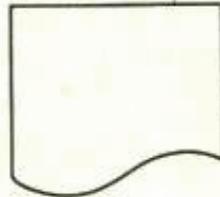
proceso complicado.



entrada por teclado.



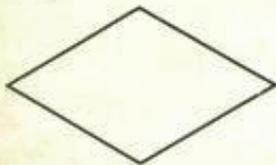
salida por pantalla.



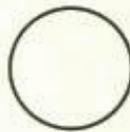
salida por impresora.



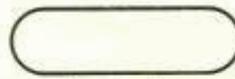
flujo de programa.



decision.



conector.



conector terminal.

Figura 1

definidas y resueltas estas cuestiones debemos pasar a la siguiente fase de diseño.

## Construcción de un organigrama

Aunque bastante odiados por muchos programadores (incluso profesionales), la realización de un organigrama antes de empezar a escribir el programa es sumamente útil porque permite un diseño más racional además de permitirnos descubrir los problemas sobre el papel, siendo más fácil su corrección que si estamos a la mitad del programa cuando lo estamos tecleando "por improvisación".

Para construir estos organigramas existen unos símbolos con un significado y uso bien definido. En la figura 1, podemos ver los más importantes y que son los que nosotros usaremos.

Al principio y al final del programa

Al principio y al final del programa vemente cómo se construye un organigrama y el uso que se da a cada símbolo. Un organigrama es el conjunto de las instrucciones del programa explicadas en forma de bloques, no se explica el funcionamiento instrucción por instrucción, sino en paquetes de instrucciones de fácil programación, así un bloque podría ser: MUEVE FICHA HASTA LA CASILLA ELEGIDA. La traducción de este bloque a instrucciones de programa será, evidentemente, más larga aunque no revierta una excesiva complicación, incluso se podría diseñar una subrutina general que se usase cada vez que tuviésemos que hacer esta operación. Los bloques están unidos por líneas de flujo que salen de uno y van a parar a otro, pudiendo llegar sólo a uno, ya que los programas son lineales (no se pueden seguir dos líneas a la vez).

Al principio y al final del programa



## EN PORTADA

se deben colocar conectores terminales con la palabra PRINCIPIO o FIN escrita dentro de ellos, su finalidad precisamente consiste en indicar esto; dónde empieza y termina el programa indicación que pese a parecer superflua a veces no lo es tanto (además crea un buen efecto). Entre estos dos conectores debe hallarse el resto del programa. Para unir las distintas partes entre sí se utilizan las líneas de flujo; que indican cuál es el siguiente bloque que se

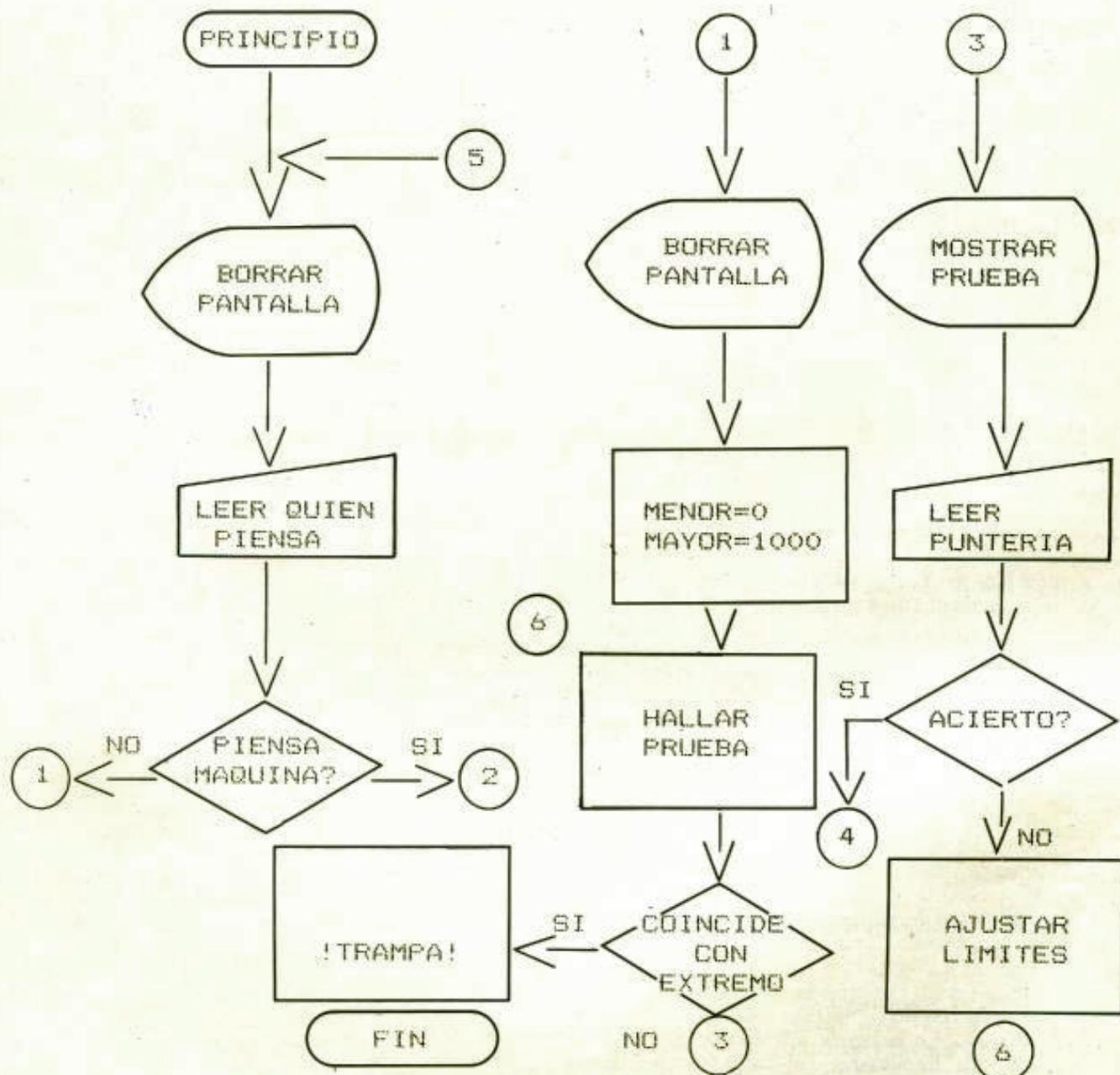
ejecuta después de éste. Los procesos simples son aquellos conjuntos de instrucciones que no hay que especificar más ya que su transformación en líneas de programa es inmediata, el bloque ENCONTRAR EL MAYOR ELEMENTO DE LA MATRIZ A (10) se transforma en las líneas del programa corto de este artículo.

En cambio las operaciones que se designan como proceso complicado requieren una explicación aparte de su

funcionamiento, un caso típico sería: HALLAR LA MEJOR CASILLA POSIBLE PARA MOVER. Esta operación en muchos casos es difícil de plantear y de programar y se debe especificar bien cómo se hace.

La entrada por teclado significa exactamente eso, el programa lee datos del teclado para darles un uso explicado posteriormente (mover el marciano, leer la jugada del jugador humano en una partida de damas, recibir órdenes

Figura 2

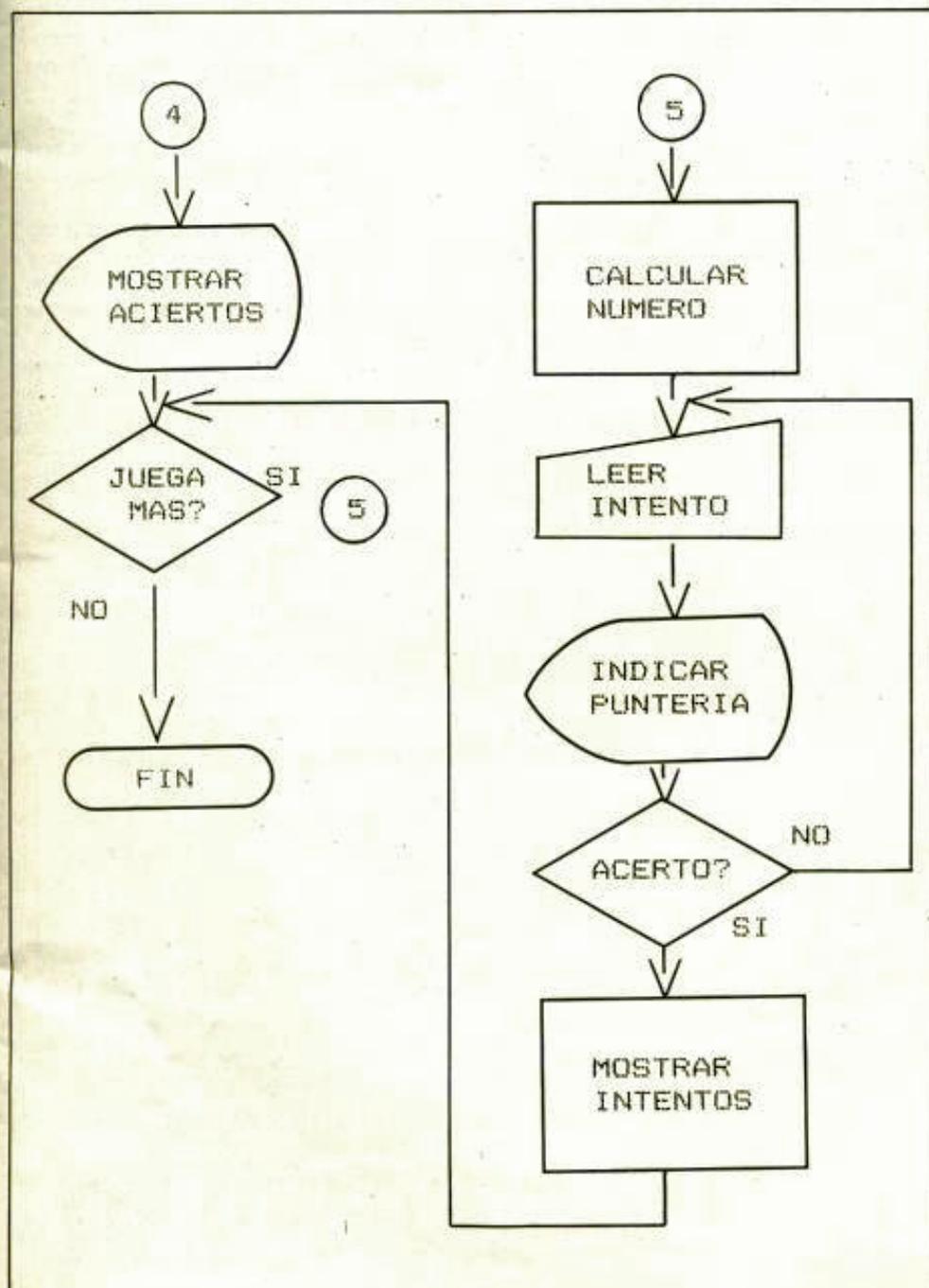


para nuestro explorador...), asimismo la salida por pantalla y/o la salida por impresora dan resultados (números, dibujos o de cualquier otro tipo) por el medio que especifiquen.

El símbolo de toma de decisión es el único del que salen dos líneas de flujo, aunque esto parezca contradecir el postulado que dimos antes de que sólo puede haber una línea de flujo ejecutándose a la vez. No es así, este símbolo representa una toma de decisión en ba-



Figura 3



se a unos datos y según el resultado se sigue una línea o la otra. Pero nunca las dos a la vez.

Por último, el conector (no terminal) es de gran uso cuando se escriben programas largos que no caben en una sola hoja, en este caso al final de la línea de flujo se pone este conector con un símbolo dentro, en la página donde continúa la ejecución ponemos este conector con el mismo símbolo seguido del resto del proceso, por decirlo de algún modo es un indicador de donde continúa el programa.

### Introducción del programa

Ya ha llegado el momento de introducir el programa, podemos desempolvar el ordenador y enchufarlo porque vamos a trabajar con él. En este punto debemos traducir las instrucciones del organigrama a instrucciones comprensibles por la máquina. Si hemos hecho el organigrama bien esto no representará mucha dificultad y será sólo una cuestión de paciencia.

### Eliminación de errores

El apartado anterior no es el último, aunque debería serlo, pero por desgracia los duendes del silicio andan sueltos y un programa que debería funcionar, no funciona o funciona mal. En vez de tirarlo a la basura (opción que atrae) lo que se debe hacer es empezar a repasar el organigrama y el programa de ordenador para ver dónde falla y por qué. Este proceso no es sencillo (muchas veces es el más complicado) y se debe realizar con mucho cuidado y paciencia porque muchas veces el error es un cero que debería ser la letra "o", o un "1" que debería ser un cero. Una vez hecho esto, a disfrutar del juego que para eso lo hemos hecho.

Estos consejos que se dan a continuación son muy importantes para que

## EN PORTADA

el programa sea divertido y nos agrade.

Ante todo debemos recordar que hay que luchar contra un enemigo imaginario y no contra un ordenador que intenta hacer todo lo posible por evitar que juguemos, las instrucciones deben ser claras, el manejo sencillo (evitar que para indicarle que queremos ir a la izquierda tengamos que pulsar cuatro teclas, con una debe bastar), no debe ser demasiado difícil como para frustrar al principiante ni demasiado fácil como para aburrir al experimentado. Un buen sistema para evitar esto último es la dificultad progresiva. Cuanto más avanzamos en el programa más difícil se vuelve. El jugador humano puede ser atacado. Un juego en el que nunca podemos tener algún método de defensa para poder permitirnos la satisfacción de destruir al malvado enemigo cibernético (naturalmente usando medios pacíficos, no clavando un hacha en medio del teclado). Por último a los juegos de acción sería aconsejable dotarles de una tecla que parará momentáneamente la ejecución del juego (no hay nada más horrible que una llamada telefónica en medio de una partida de comecocos) pudiéndose continuar al volver a pulsar la tecla.

### Un ejemplo sencillo

Como ejemplo de las ideas que acabamos de explicar vamos a realizar un sencillo programa.

Idea: El juego consiste en que uno de

los dos contendientes (el ordenador o la persona) piense un número entero (sin decimales) entre el 1 y el 999 y el otro lo debe averiguar poniendo números de prueba y siguiendo las indicaciones del contrario que le debe indicar si el número es más alto o más bajo que el que ha pensado. La estrategia que debe seguir el ordenador es la siguiente: si le toca pensar a él debe generar un número entre el 1 y el 999 aleatoriamente, a continuación debe empezar a "leer" los datos que le dé la persona y decir si son mayores o menores que el que ha pensado. Si coinciden indicará que lo acertó. Si le toca adivinar el número a él, lo hará siguiendo el proceso denominado de búsqueda dicotómica. Este proceso aunque largo para determinados números tiene una efectividad media bastante buena. Básicamente consiste en coger el intervalo en el que está comprendido el número, sabiendo que éste no es ninguno de los dos extremos, por lo tanto de partida cogeremos los números cero y 1.000, que son ilegales y entre los que debe estar comprendido el número a adivinar, el número a probar es el que se encuentra en la mitad del intervalo (en el primer intento será el 500), si no es este número se le cogerá como nuevo extremo inferior o superior dependiendo de si es demasiado alto o demasiado bajo y se vuelve a repetir el proceso hasta que se encuentre el número. Por

ejemplo, si el número pensado es el 700 en el primer intento nos preguntará que si es el 500, al decirle que es demasiado bajo lo tomará como límite inferior quedando el número buscado dentro del intervalo (500, 1.000). Volverá a calcularse el número de prueba y a preguntar (el 750), ajustando los límites otra vez, etc.

Además debe indicar el número de intentos que se tarda en averiguarlo.

Una vez descrito el programa pasamos a escribir el organigrama que es el de la figura 2.

Además, cada vez que se lea o se escriba una prueba hay que incrementar el contador de intentos en una unidad, este contador al principio se habrá puesto a cero. Estos procesos no se han incluido en el organigrama por no llenarlo excesivamente y son de fácil comprensión.

Después del organigrama escribiríamos el programa correspondiente y una vez funcionando lo dejaríamos "bonito" resultando el listado de la figura 3.

Un atento análisis (hágalo como problema de estudio) le revelará la coincidencia del programa con el organigrama, aunque esta semejanza en ningún programa llegará a ser del cien por cien debido a los problemas de última hora (a menos que hagamos el organigrama después del programa, práctica muy bonita pero sin utilidad aparente).

Figura 4

```

5 REM ma=mayor elemento,or=or
ien
6 DIM a(10)
10 LET ma=a(1): LET or=1
20 FOR i=2 TO 10
30 IF a(i)>ma THEN LET ma=a(i)
LET or=i
40 NEXT i

```

Figura 5

```

10 CLS
20 INPUT "Pienso yo?";r$
30 IF r$(1)="s" THEN GO TO 100
40 IF r$(1)="n" THEN GO TO 500
50 GO TO 10
100 LET nv=INT (RND*999)+1
110 LET in=0
120 CLS
130 INPUT "Numeroa probar:";pr
140 LET in=in+1
150 IF pr<nv THEN PRINT "El num
ero ";pr;" es demasiado bajo": G
O TO 130

```

```

160 IF pr>nv THEN PRINT "El num
ero ";pr;" es demasiado alto": G
O TO 130
170 PRINT "Acerto el numero en
";in;" intentos"
180 GO TO 200
200 INPUT "Desea jugar otra vez
?";r$
210 IF r$(1)="s" THEN GO TO 10
220 IF r$(2)="n" THEN STOP
230 GO TO 200
500 CLS
510 LET in=0: LET me=0
520 LET ma=1000
530 LET pr=INT ((ma-me)/2)+me
535 IF (pr=me) OR (pr=ma) THEN
PRINT "Me has mentado. Ya no jue
go": STOP
540 LET in=in+1
550 PRINT "Es el ";nv;"?"
560 INPUT "Alto, bajo o correct
o";r$
570 IF r$(1)="c" THEN PRINT "Lo
acerte en ";in;" intentos": GO
TO 200
580 IF r$(1)="b" THEN LET me=pr
: GO TO 530
590 IF r$(1)="a" THEN LET ma=pr
: GO TO 530
600 GO TO 550

```

PROMOCION  
RECOMPRA



# peek and poke

**AL FINAL GRATIS.** Durante todo este mes de Mayo **Peek & Poke** pone en marcha la **operación recompra** para facilitarte la compra del Sinclair Spectrum.

Nos comprometemos a devolverte, a los 2 años y por el Spectrum usado, el dinero que tú pagues ahora.

Ven a **Peek & Poke** e infórmate.

**Peek & Poke**

C/ Génova, 11. MADRID-4. Tel.(91) 419 8100

## Peek and Poke, está clarísimo.

**Pregunta:** El otro día miré la cinta de demostración del Spectrum y me fijé en los diferentes rótulos que aparecían en algunos programas. ¿Cómo podría crear rótulos variando la altura y la anchura?

**Miguel Alonso Fischer Reus (Tarragona)**

**Respuesta:** Para ello puede utilizar la rutina en código máquina que viene en la cinta Horizontes y que se reproduce en el programa Othelo publicado en este número. En dicho programa se explica el funcionamiento de la rutina.

**Pregunta:** Desearía encontrarme con programa de cálculo de todo tipo: elección de transistores, cálculo de diodos, lógica digital, etc. Me interesa especialmente programas de ficheros, almacenamiento de datos y todo lo que se refiere a almacenar datos técnicos.

**Alberto Ibarretxa Galdácano (Vizcaya)**

**Respuesta:** A nosotros también nos gustaría. Estamos seleccionando material. El almacenamiento de datos es muy importante, especialmente con la aparición del microdrive.

**P.:** ¿Cómo limitar el número de decimales que aparecen en la pantalla en el Spectrum? Y otra pregunta, escribo PRINT 20/4 y aparece en la parte superior de la pantalla 5, ¿cómo bajo este número para seguir operando con él?

**Javier Lainez Vizcaya**

**R.:** Para controlar el número de decimales puede definir una función o fabricarse una subrutina. Veamos un ejemplo de esto último:

```
5 INPUT "No. DECIMALES "; D
10 INPUT X: GO SUB 100
15 PRINT X: GO TO 10
100 LET X$ = STR$ X
110 FOR I = 1 TO LEN X$
120 IF X$(I TO I) <> "." THEN GO
    TO 160
130 LET L = (LEN X$) - I
140 AIF D <= L THEN LET X = VAL X$ (I
    TO I + D)
150 RETURN
160 NEXT I: RETURN
```

13.233  
23.444  
52.54  
—12343.232  
90.5

Respecto a su segunda pregunta no nos da suficiente información sobre lo que quiere hacer para contestarle.

**P.:** Soy un chico de 13 años, poseo un ZX Spectrum y me gustaría que me contestaran a estas preguntas:

1. Tengo una "interface" RS232C-Centronics y me gustaría conocer sus usos.

2. A mi padre le han dado un aparato para convertir las señales de Morse en señales de ordenador pero tiene que hacer un listado en código máquina; ésta en hexadecimal y quiero saber como introducirlo.

3. El interface 2 es, sin duda, mejor que el interface 1, pero, ¿es muy grande la diferencia de precio?

4. El manual de la interface RS232C Centronics es pequeñísimo y muy poco claro, por ello quiero que me resolváis estas cuestiones: ¿cómo se hace para conectar el Spectrum con otro ordenador (el Vic 20, por ejemplo)? ¿Se necesita alguna clase de programa para su funcionamiento?

5. ¿Se puede construir un Joy-Stick?

6. Hay una galleta en el Spectrum para conectar periféricos pero, ¿cuáles son los nombres de sus conectores, con qué números o letras se designan?

7. ¿Qué son y para qué sirven los comandos PEEK y POKE?

8. Si se les envía un programa, ¿devuelven el cassette?

**José Vaño Sempere Alcoy (Alicante)**

**R:** Intentaremos contestar a todas tus preguntas por el orden en el que nos las formulas.

— La interface RS232C te sirve para poner tu Spectrum en comunicación con todos aquellos periféricos que posibiliten ésta comunicación. La utilización más frecuente es con las impresoras, pero sirve en general para todo tipo de periféricos ya que su misión es la de servir de "intérprete" para que el Spectrum se entienda con cualquier periférico, e

incluso con otros ordenadores.

— La introducción de instrucciones en código máquina no se puede hacer directamente. Para ello ha de utilizarse un programa Basic que permita alojar los distintos valores en unas determinadas posiciones de memoria. Te remitimos a los artículos sobre código máquina que venimos publicando. Existen también libros sobre este tema. Además en el próximo número, puedes ver este mismo problema que nos comentas en el programa OTHELO, un magnífico programa que utiliza una rutina en código máquina.

— En España todavía no se comercializan ningún tipo de interface, aunque próximamente aparecerá el interface 1 para la utilización del microdrive. Una mayor información sobre ésto la puede encontrar en Inversión.

— Conectar un ordenador con otro distinto no es sencillo, y lo primero es saber para qué se van a conectar. En cualquier caso, no le será posible conectar el VIC-20 con el Spectrum, ya que los microprocesadores son distintos.

— Por supuesto que te puedes construir un Joy-Stick, pero no olvides que también se ha de construir la interface para que Spectrum e interface "se entiendan".

— En el número 3 publicábamos el esquema del Spectrum donde se hablaba de esto. No obstante en el manual lo puedes encontrar con mayor detalle.

— Las instrucciones PEEK y POKE sirven para acceder a las distintas posiciones de memoria. La instrucción PEEK permite leer, en tanto la POKE permite modificar o grabar en una localización determina-

da. De esta forma se puede almacenar el programa en código máquina a que antes aludíamos.

— Finalmente, como respondemos a otro lector, todos los cassettes, junto con la documentación que nos enviáis, nos lo quedamos en archivo.

**Pregunta:** ¿Qué hay que hacer para enviar programas a su revista?

Miguel Angel García.  
Madrid.

**Respuesta:** Siga las normas que a tal efecto aparecen en cualquier número de ZX.

**P.:** ¿Cómo es posible que los señores de Ventamatic anuncien en vuestra revista que el Spectrum de 48K vale sólo 43.950 pesetas? ¿Es verdad o es tongo? ¿Podrías especificar si los programas que publicáis son en color? ¿Podrías publicar algún programa de buenos gráficos, de marcianos, comecocos o semejantes?

Carlos Navarro Bilbao  
Orense

**R.:** La comercialización del Spectrum dentro y fuera de Gran Bretaña tiene en razón del volumen que ha alcanzado, varios circuitos paralelos. Oficialmente, el distribuidor exclusivo en España es Investrónica, pero hay otras empresas que se las arreglan para importar máquinas directamente y para venderlas más baratas. La diferencia, si acaso, reside en los manuales y en la calidad del servicio postventa que pueden ofrecer esos distribuidores no autorizados por Sinclair Research. Por lo demás, se trata de la misma máquina.

La información sobre si los programas son o no en color creemos que es innecesaria. Basta con ver si hay instrucciones INK, BORDER o PAPER para saber que se está utilizando la capacidad del Spectrum para realizar vistosos programas. Lo realmente importante es saber qué ordenador y qué capacidad de memoria se necesita para utilizar el programa, ¿no cree?

En cuanto a su última pregunta, tratamos de seleccionar los mejores programas y la representación gráfica es uno de los criterios de selección, junto con el tema del programa, más importante.

**P.:** Deseo solicitarles me informen cuál es la línea correcta (90 PLOT, x, y) del programa que hace referencia al dibujo del joystick de su revista número 2, ya que da error y yo no doy con la línea correcta.

Fermín Martín García  
Pamplona

**R.:** Tenga en cuenta que dicho programa se realizó con el joystick de Indescomp, por lo que su utilización con otros joysticks puede no funcionar. Si no es así, el programa ha de funcionar correctamente y con él hicimos el dibujo que apareció a su derecha.

**P.:** Soy un chico que tengo 13 años y en estos momentos no tengo ningún ordenador. Me gustaría poder mandar algún programa que he preparado en una tienda de Murcia donde me dejan el ZX81, pero el problema está en que al no

tener ninguno no pudo mandarle el cassette. ¿Si el programa fuera elegido ganaría las 5.000 pesetas?

Raul Morales  
Murcia

**R.:** Por supuesto. Todos los programas elegidos son premiados con 5.000 pesetas. Y entre todos los recibidos, se publiquen o no, se sortea cada mes un *Microdrive*. Además, para el ZX81 recibimos pocos programas, lo que aumenta tus posibilidades de que el programa que envíe sea elegido.

**P.:** He visto que en muchos programas comerciales y de esta revista (por ejemplo el programa Bazooka de la 2a. edición de ZX) definen variables como "LET A = PI/PI", "LET C = PI-PI", "LET B = CODE "A", etc. ¿No sería más sencillo y ocuparía menos memoria ponerlo como LET A = 1, LET C = 0, etc.? ¿Tiene, por el contrario, alguna función o finalidad especial el escribirlo de esa manera?

Pedro Pérez Chaves  
Madrid

**R.:** El realizarlo de esa forma es precisamente para ocupar menos memoria. Por ello se emplea cuando se trabaja con el ZX81 de 1K.

**Pregunta:** Tengo el siguiente problema. El sitio en que vivo tiene luz muy inestable y el estabilizador del Spectrum no es lo suficientemente potente para estabilizarla. Donde vivo hay unas oficinas que usan

ordenadores más potentes y me dijeron que quizás con un estabilizador de TV se arreglaría, enchufándolo al Spectrum además del que lleva incorporado. Así lo hice y funcionó durante un tiempo, pero después volvió a fallar y se me estropeó el Spectrum. ¿Qué puedo hacer?

José M.<sup>a</sup> Perello  
Tarragona

**Respuesta:** Lo mejor es poner un fusible previo a la toma de corriente. Consulte en cualquier tienda especializada para que le indiquen la potencia adecuada.

**P.:** ¿La empresa Sinclair fabricará algún tipo de funda o maleta para el transporte del Spectrum? También me gustaría que me enviaran, si es posible, un catálogo de *software* para el Spectrum?

Francisco Borrás Gras  
Barcelona

**R.:** A Sinclair Research sólo le preocupa la fabricación de ordenadores. Los accesorios los deja a otros. Hemos visto en alguna tienda especializada un pequeño maletín importado de Gran Bretaña, que ha sido fabricado para el uso que usted busca. Es todo lo que podemos decirle al respecto.

Respecto a su pedido de catálogo de *software*, nadie tiene uno completo. Nosotros estamos trabajando en uno para publicarlo oportunamente en las páginas de la revista. Mientras tanto, puede usted leer detenidamente las páginas de publicidad, o coleccionar los comentarios que publicamos en las páginas o sección correspondiente.

## LECTORES

**Pregunta:** ¿Es posible adaptar un Spectrum a un monitor de video de alta resolución? ¿Existe algún medio de incrementar el número de caracteres por línea en su salida por pantalla?

**J. García Nieto**  
Alcalá de Henares (Madrid)

**Respuesta:** No hay ningún problema para adaptar el ordenador a un monitor de video de alta resolución. En cuanto a la posibilidad de incrementar el número de caracteres por línea, vamos a preparar un artículo sobre ello, por lo que ya le adelantamos que es posible.

**Pregunta:** En la salida de la unidad transformadora de mi ordenador Spectrum 48K, medida con un tester, me da un voltaje de 14,7 voltios (medidos en vacío) y este voltaje tiende a aumentar conforme transcurre el tiempo después de enchufar la unidad transformadora. ¿Es normal este voltaje de salida sin estar conectado el Spectrum?

**José Angel Fontela**  
Pontevedra

**Respuesta:** Es perfectamente normal. El efecto de carga hace que la tensión disminuya.

**Pregunta:** Sería de mucha utilidad para los usuarios del Spectrum que publicaran el esquema para poder realizar en casa el *interface* para la aplicación de una palanca de mando a cuatro posiciones para diversos juegos, evitando así el deterioro de las teclas del ordenador.

**Joan Batllori Tarres**  
Barcelona

**Respuesta:** Lo publicamos en este mismo número de la revista.

**Pregunta:** Tengo muchas ganas de colaborar en su revista, así que les voy a mandar algunos trabajos para ver si son de su agrado y son publicados. Por el momento les envío un proyecto de un modem "muy especial". Especial por varias razones: no hay que ser un manitas para hacerlo, ni saber mucho de electrónica. Sólo hay que saber soldar; funciona perfectamente, y la razón más convincente, un modem puede costar cuando llegue a España más de 15.000 pesetas. Mi "prototipo" sale por menos de 100 pesetas. Tan sólo se necesita cable aislado, un conector de tipo jack, dos pinzas metálicas o algo similar.

El funcionamiento es el siguiente: se desenrosca el micrófono del teléfono y se enganchan las pinzas a las "patas" que hacen contacto con la cápsula. El conector se enchufa al cassette por la toma EAR, y se ajusta el volumen y los agudos al máximo. El receptor hace lo mismo, desenrosca el auricular, engancha las pinzas y enchufa el jack a la toma MIC del *cassette*. Conviene que el receptor desenrosque el micrófono también para evitar ruidos. Simple, ¿verdad? (Ver gráfico).

**Fernando Doutel García**  
Madrid

**Respuesta:** Es un método ciertamente original, pero no tan fácil y simple como usted cree, y ello por tres razones: es ilegal toda conexión no autorizada por telefónica, existe posibilidad de dañar el ordenador, y además, no existe posibilidad de controlar el volumen. Lamentamos decepcionarlo.

**Pregunta:** Quisiera referirme al programa Repaso de Geometría, publicado en el número 4, página 42. En él se pregunta por mis conocimientos sobre esta ciencia y si me atrevería a ampliarlo... Pues no, no me atrevo si no se corrigen las líneas 3010 y 4010 las que, según parece, han sido limpiamente traducidas de la versión inglesa, pero sin haber tenido la precaución de repararlas y hacer uso de los mínimos y precisos conocimientos de geometría para darse cuenta de los dos olímpicos errores copiados literalmente del original. Según mis matemáticas de bachillerato (de hace cincuenta años) el área de la esfera era  $4 \cdot \pi \cdot m \cdot m$  y su volumen  $\frac{4}{3} \cdot \pi \cdot m \cdot m \cdot m$ . No creo que con la matemática moderna hayan cambiado, ya que esta ciencia habría dejado de llamarse exacta. Cuanto he expresado es pura broma, pero va en serio, aunque sin la más mínima intención de ofender a nadie.

**Juan Guillén Serra**  
Barcelona

**Respuesta:** En el campo de la informática, ya sabrá usted, las cosas cambian constantemente y a un ritmo vertiginoso. No obstante, la inmutable verdad matemática permanecerá siempre intacta, por lo que le damos la razón. Pero se aventura usted en la suposición de que sea una traducción de un programa inglés. Ese programa, que recibimos de un lector, fue premiado por su originalidad y si hubiésemos tenido algún indicio de tal copia no habría sido publicado ni, desde luego, recibido el premio. Quizás su autor, si así lo desea, pueda decir algo al respecto.

**Pregunta:** Al recibir el número 4 de ZX me llevé una alegría al ver un programa para calcular los números primos, pues tenía yo de hecho uno que, desgraciadamente, me introducía como primos los números  $361=19 \times 19$ ,  $841=29 \times 29$ ,  $961=31 \times 31$ ,  $1.369=37 \times 37$ , etc., que no son primos. Observé que en el ejemplo que publicábais no figuraba el dos (único número par primo) y, en efecto, el programa no lo da como primo y además de los mismos números erróneos que el hecho por mí, es decir, 361, 641, 961, 1.369, etc. El error se produce por alguna circunstancia que no comprendo, pues el ordenador toma dos números que son iguales como distintos (en realidad un número y la parte entera de otro). ¿Es que ocurre en BASIC como en Fortran, que se distinguen las variables enteras de las reales, considerándolas el ordenador distintas aunque tengan el mismo valor?

**Santiago Pérez González**  
Salamanca

**Respuesta:** Como le ocurría a otro lector, en ocasiones el ordenador le puede engañar. Así, por ejemplo, puede ocurrir que le diga que  $A=5$  y  $B=5$  y, sin embargo, le informa que A no es igual a B. En estos casos, pídale qué vale A-B o la diferencia de las variables con las que trabaja. Verá que la diferencia es insignificante, pero distinta de cero y lo suficiente para que el ordenador lo note. Para solucionarlo, basta con indicarle el nivel de exactitud que desea. Vea la respuesta que damos al lector Arístides López.

**Pregunta:** Les escribo para que me manden información sobre los caracteres

# UNA GALAXIA DE PROGRAMAS PARA EL ZX-SPECTRUM

## Video-juegos

### PHEENIX (16K/48K)

La mayor versión del clásico video-juego de los bases. Debes aniquilar a los pájaros que se lanzan como desgranados para eliminarlos, después a los que están de los huecos y al final al pájaro avanzado en la nave comandante. 5 pantallas. C.M. J.K. 1.600,—

### WRECKAGE (16K/48K)

Increíbles gráficos en 3 dimensiones. Debes eliminar a las naves enemigas antes de que destruyan tu ciudad con sus bombas. Dificultad progresiva hasta imposible. 10 pantallas. C.M. J.K. 1.600,—

### ANDROIDE UNO (16K/48K)

Misión Reactor. Debes conducir a Androide a través de distintos salas y laberintos, rompiendo las paredes y eliminando a los guardianes con su rayo láser, para poner una bomba en el centro del reactor y volver a la base antes de que se agote el tiempo o las baterías de energía. 7 pantallas. Extraordinarios gráficos animados. C.M. J.K. 1.600,—

### ANDROIDE CERO (16K/48K)

Conduce a Androide a través de un laberinto muchas veces mayor que el pantalla, y encuentra la salida. Destruye a todos los enemigos y recarga tu láser y campo de energía con las cápsulas que encontrarás en el camino. C.M. J.K. 1.600,—

### LA MINA (16K/48K)

Debes recoger las 40 bolas de dinero escondidas en la mina y meterlas en la carromata. Puedes subirte en las vigas y ascensores, pero cuidado con los trampolines, cables y los pediceros que te persiguen. 3 pantallas. Extraordinarios gráficos animados. C.M. 1.600,—

### BEDLAM (16K/48K)

Viveva a los horribles monstruos del laberinto en el planeta prisión. Cuidado con los guardianes del terreno. Extraordinarios gráficos animados. 3 niveles de dificultad progresiva. C.M. J.K. 1.600,—

### GULPMAN (16K/48K)

Una nueva versión del clásico conocido con el nombre de 10 laberintos, 10 velocidades y 10 grados de aceleración seleccionables. C.M. 1.400,—

### CYBOTRON (16K/48K)

Destruye a los robots y procura no chocar con la nave electrificada. Dificultad progresiva hasta 50 robots en pantalla, y algunos indestructibles. 1.000,—

### LA RANA + Z-MAN (16K/48K)

Las mejores versiones de dos de los video-juegos más populares reanuda en un solo cassette. La rana debe alcanzar la autopista de la muerte y el río hasta llegar a su río. Z-MAN es una réplica exacta de gráficos, sonido y pantallas al clásico conocido. Ambos con pantallas sucesivas de dificultad progresiva hasta llegar a tope la memoria disponible. C.M. J.K. (solo Z-MAN) 1.800,—

### CIEMPIES + STORM-FIGHTERS (16K/48K)

Destruye al enemigo que va avanzando hasta él a través del bosque de setas, pero cuidado: cada vez que se alcanzan en el medio se divide en dos. Atención al caracol y a la araña araña que también tratan de destruirte. En Storm-Fighters evita los aviones y destruye las naves enemigas que avanzan y disparan hacia ti. Dificultad progresiva hasta llenar la pantalla de naves enemigas y asteroides. C.M. 1.800,—

### EL DETECTIVE (48K)

A través de 25 pantallas elimina a todos los malvados que tratan de destruirte con bombas, aviones, helicópteros y toda clase de trampas para evitar que descubras los secretos de su caja fuerte después de averiguar la combinación. Evita ser aplastado por los neumáticos viejos y sobre para que no te atañe el peso sobre-punto. Extraordinarios gráficos animados. C.M. J.K. 1.700,—

### GRAND PRIX (48K)

Extraordinarios gráficos en 3 dimensiones. Corre con la fórmula 1 en un circuito de carreras contra otro coche. Circuitos y clase de opones seleccionables. C.M. 1.700,—

### TUNELES MARCHANOS (48K)

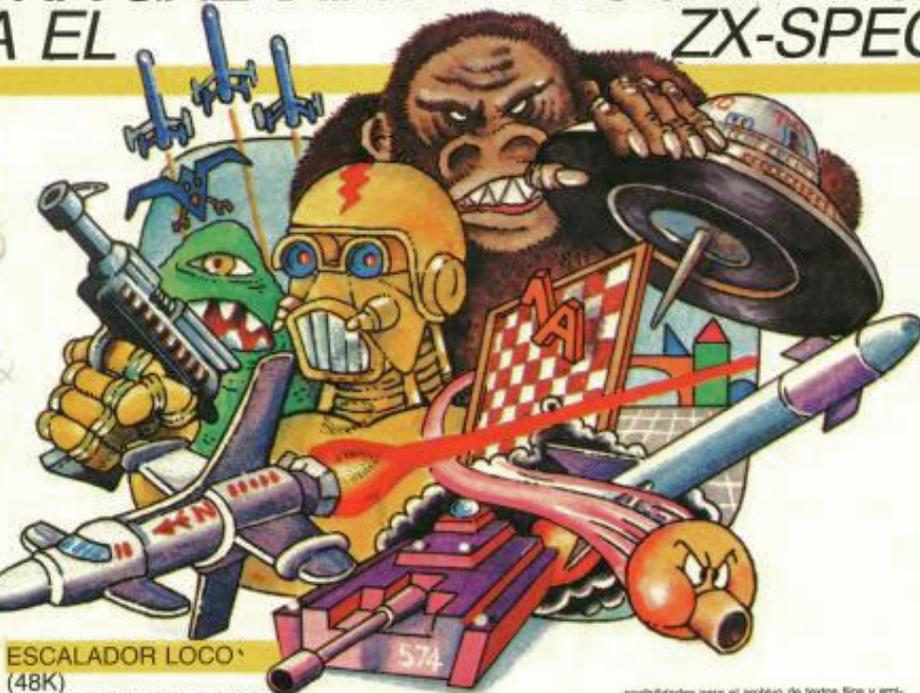
Recoge las siete llaves que abren el coche del último nivel, en las distintas zonas (laberinto de la muerte, zona pasadizo y las zonas bajas) y volver a la cabina tele-transportadora. Cuidado con los criaturas y las minas. Extraordinarios gráficos animados 3D. C.M. J.K. 1.700,—

### SUPER-COMELOCOS (16K/48K)

1.400,—

### GALAXIANS + SPYNADS (16K/48K)

1.000,—



### ESCALADOR LOCO (48K)

Sube por las ventanas del edificio. Cuidado con pillarle los dedos en las que se cierran, y con los besitos, pequeños y bolas de béisbol que te lanzan desde arriba. Hay diferentes zonas de ventanitas donde la dificultad es mayor. C.M. J.K. 1.700,—

### STOMPING STAN (48K)

Conduce al monstruo STAN a su casa después de recoger las llaves que se hallan en lo alto de la pantalla. Para subir debes saltar por los agujeros de las banderas móviles y evitar ser atrapado por los persiguidores. En la segunda pantalla debes recoger las llaves y evitar a los persiguidores en el laberinto. C.M. 1.700,—

### KILLER KONG (48K)

Rescata a la chica de las garras del gorila loco que te lanza barriles para aplastarte. 5 pantallas con escaleras, fuego, ascensores e incluso hamburguesas. Extraordinarios gráficos animados y efectos de sonido. C.M. 1.700,—

### BARMY BURGERS (48K)

Recorre los cuatro ingredientes de las hamburguesas subiendo escaleras y pasando sobre ellos. Poco cuidado con las salsichas y el huevo frito que te persiguen y a los que puedes dejar fuera de combate por un momento lanzándoles rubes de ginebra. 5 pantallas de dificultad progresiva. C.M. 1.700,—

### BUSQUEDA EN LA MINA MALDITA (48K)

Recoge todos los diamantes en los 31 niveles de la mina y así a la superficie. Cuidado con las rocas que te lanzan el monstruo y los murciélagos, fantasmas y toros de arena. Extraordinarios gráficos animados. C.M. 1.700,—

### AUTOSTOPISTA GALACTICO (48K)

Te hallas en medio del espacio y debes recoger bolitas de oxígeno para sobrevivir y dinero para pagar las naves-taxi. Encuentra un planeta y atábase en el para hacer auto-stop. Hay naves-600, naves-ROLLS, naves-policia, naves-beasuro, naves-luz y naves-dudosa que te pueden robar el dinero. De este modo debes llegar al planeta central de la inmensa galaxia ayudado de la mapa galactico. Extraordinarios gráficos. C.M. J.K. 1.700,—

### ROBOT FACTORY (48K)

Dirige la cadena de montaje de robots para fabricar el mayor número posible. Llena los depósitos antes de que se vacíen, ajusta la velocidad de la cadena, pero cuidado con los robots defectuosos que te atoran y las cadenas. C.M. J.K. 1.700,—

### ULTIMO AMANECER EN LATTICA (48K)

A través del enorme laberinto de las calles de Lattica, recoge las llaves que te llevan a los distintos niveles y enciende y desactiva la bomba antes de que explote y destruya a Lattica. Deberás enfrentarte con multitud de guardianes del laberinto. Extraordinarios gráficos animados. C.M. J.K. 1.700,—

### ANDROIDE DOS (48K)

Misión Megatón. Debes destruir a 5 megatóns en las distintas zonas (laberinto de la muerte, zona pasadizo y las zonas bajas) y volver a la cabina tele-transportadora. Cuidado con los criaturas y las minas. Extraordinarios gráficos animados 3D. C.M. J.K. 1.700,—

### VIDEO-DISPLAY (16K/48K)

2.000,—

### EMISION/RECEPCION MORSE (48K)

3.000,—

## Juegos inteligentes

### AJEDREZ 2002 (48K)

3.500,—

### DALLAS (48K)

1.400,—

### GUERRA DE BARCOS + SUBMARINO (16K)

1.400,—

## Utilidades

### ENSAMBLADOR/ DEENSAMBLADOR: ULTRAVIOLET/ INFRARED (16K/48K)

2.500,—

### COMPILADOR (16K/48K)

2.000,—

### MASTER-DISEÑO (48K)

2.000,—

### 3D VISION (16K/48K)

2.000,—

### VIDEO-DISPLAY (16K/48K)

3.000,—

### EMISION/RECEPCION MORSE (48K)

3.000,—

posibilidad para el archivo de textos tipo y emisión directa. La velocidad de recepción puede variar hasta un 40% sin entrar en la decodificación. 2.000,—

### FORTH (16K/48K)

3.000,—

### ADAPTADOR PROGRAMAS BASIC ZX81 (16K/48K)

1.500,—

## Gestión

### MASTERFILE: BASE DE DATOS (48K)

1.500,—

Disponibles también los mejores programas de importación (IMAGINE, ULTIMATE, BUG-BYTE, SOFTWARE PROJECTS, SINCLAIR, OCEAN, MELBOURNE HOUSE, MICROMEGA, etc.) y programas para ZX81. Solicite listas.



## VENTAMATIC

VENTA POR CORREO: Avda. de Rhode, n.º 253 - Apartado de Correos n.º 168 - Tel.: (972) 25 56 16 (24 horas) - ROSAS (Gerona).

EXPOSICION Y VENTA AL PUBLICO: c/Córcega, n.º 89, entlo. - BARCELONA-29.

Recorte o copie este cupón.

Fecha: 16-5-84

Señores de VENTAMATIC ( ), envíenme VIDEO-DISPLAY 48K

NOMBRE: LAIME G.

APELLIDOS: LOPEZ GARCIA

DOMICILIO: AVDA. FUERZAS ARMADAS 3

PROVINCIA: ALICANTE DP: —

Envío giro postal/telón conformado ptas.: fecha: n.º

Para pagos reembolso o tarjeta crédito envíe 500,— ptas. gastos envío. Tarjeta VISA/MASTERCARD n.º

CADUCA:

Gastos de envío: • Pedidos + 7.000 ptas. pagados por adelantado, sin gastos. • Pedidos - 7.000 ptas. pagados por adelantado, 300 ptas.

de cualquier tipo de fichero y listado, así como la selección y ordenación por cualquier concepto. Los datos creados son accesibles incluso desde programas BASIC del usuario. Totalmente traducidos al castellano y adaptable a todo tipo de impresoras de papel normal, o ZX. 3.000,—

### CONTABILIDAD PERSONAL (48K)

2.500,—

### CONTROL DE STOCKS 64 COLUMNAS (48K)

3.000,—

### CONTEXT: PROCESADOR TEXTOS 64 COLUMNAS (48K)

3.000,—

### SIMPLEX: PROGRAMACION LINEAL (16K/48K)

2.500,—

## Educativos

### EL ARQUITECTO (48K)

1.700,—

C.M.: Código máquina. J.K.: Joystick KEMPSTON-VENTAMATIC.

## LECTORES

gráficos del Spectrum, pues tan sólo una vez los vi reflejados en pantalla y no he vuelto a tener acceso a ellos.

Valle Moreno Peine  
Toledo

**Respuesta:** Le remitimos a los números 1 y 2 de esta revista, especialmente al segundo, donde estudiábamos la definición de caracteres en profundidad. No obstante, volveremos a tratar el tema en un próximo número para aquellos que no hayan podido conseguir aquellos, hoy agotados después de sucesivas reimpresiones.

**Pregunta:** Somos dos principiantes en este mundo apasionante. Tenemos varios problemas:

— ¿Qué significa *software*?

— ¿Cómo se graba un programa de los de la revista en una cinta?

— ¿Qué es un megabyte?

— ¿Podrían explicar en un próximo número, paso a paso, cómo hacer un programa?

— ¿Podrían hacer un apartado para principiantes?

— ¿Cómo podríamos conseguir los números 1 y 2 de su revista, ya que en los kioscos está agotado?

Jordi Ruiz Zamorano y  
Bernardino Ramón Cañellas  
Palma de Mallorca

**Respuesta:** En informática hay dos palabras clave que no tienen equivalente en castellano: *hardware* y *software*. La primera se refiere a las máquinas y aunque es aplicable a todo lo que es equipo físico (*hard* significa duro), en la práctica suele limitarse a los ordenadores en sí mismos, olvidando que también los pe-

riféricos merecen ser incluidos bajo la denominación *hardware*. La segunda palabra, que tanto os preocupa, se refiere a la lógica del ordenador, su sistema operativo y a las instrucciones o programas que le permiten resolver el problema planteado. Es, por así decirlo, la parte intangible del ordenador (*soft* significa blando, en esta acepción).

Para grabar los programas basta con darle al ordenador la instrucción SAVE correspondiente. Creemos que no habéis prestado suficiente atención al Manual. Consultar las páginas 149 y 180.

Un Megabyte es una magnitud de medida de la memoria de un ordenador. El ordenador trabaja con unos y ceros, y a cada dígito (0 ó 1) se le denomina un bit. Cuando tenemos ocho bits se habla de 1 byte, y 1 Kilobyte (o 1 K, en su forma abreviada) son 1.024 bytes. Así, ustedes habrán observado que en esta revista tratamos del ZX81 de 1 K o de 16 K y del Spectrum de 16 o de 48 K. O del Microdrive de 80 K. Los grandes ordenadores pueden trabajar con millones de bytes, por lo que al hablar de su memoria se habla de Megabytes (1 M equivale a un millón de bytes o, digámoslo de otro modo, a 1.000 K).

Siempre tratamos de incluir explicaciones más detalladas de algunos programas de los que publicamos. Pero no hay "norma fija" para programar. Simplemente, hay que tener una idea clara de lo que se quiere y pensar cómo lo puede hacer el ordenador, valiéndose de las instrucciones que incorpora.

Hace tiempo que estamos pensando en una sección para principiantes, pero no encontramos todavía la fórmula que la haga, a la

vez, de interés general. Tengan paciencia.

Es cierto que esos números se han agotado en los kioscos, pero hemos hecho sucesivas reimpresiones y lo mejor será que os dirijáis a nuestro Departamento de Suscripciones, por carta o por teléfono. Ellos resolverán vuestro problema.

**Pregunta:** Me dirijo a ustedes para que me aclaren algunas dudas:

— ¿Se puede mandar más de un programa cada mes por persona? Si su respuesta es afirmativa, ¿se pueden mandar en una misma cinta?

— Si algún programa de los que envían los lectores no puede ser publicado, ¿pueden indicar por qué?

— En la sección de comentarios de programas les sugiero que pongan más imágenes, para dar una mejor idea de la resolución gráfica, efectos, etc.

— Me gustaría que publicaran algún número especial y que incluyeran algún programa para el Spectrum de los "largos", es decir, de más de 18, 16 K.

— ¿Es posible que un programa en BASIC pueda tener subrutinas en código máquina?

— Creo que los programas de juegos son necesarios y muy prácticos para aprender nuevos trucos.

Juan Pamos  
Valencia

**Respuesta:** No nos oponemos, en principio, a que nos manden cuantos programas deseen. Pero, puesto que tratamos de premiar al mayor número de lectores posible, será difícil que publiquemos más de uno de los que nos llegan en una misma cinta o procedentes de un lector en el mismo mes.

Sin embargo, que no se publiquen en el mes de recepción no impide que sean publicados más tarde, si su interés lo justifica.

Cuando no publicamos un programa puede ser por diversos motivos: funcionamiento defectuoso, falta de calidad, tema resuelto de mejor forma por otro programa, etc. No obstante, no los desechamos: todos quedan archivados por si pudiesen ser de interés en futuras ediciones.

De paso, aclaremos lo que otros lectores nos han preguntado telefónicamente: en principio, no devolvemos los programas no publicados ni mantenemos correspondencia sobre ellos, pero si algún lector se empeña en ello, no le negaremos la devolución si pasa por nuestras oficinas.

En cuanto a las ilustraciones de los comentarios, creemos que con una es suficiente. Desde luego, ello no excluye la posibilidad de hacer ocasionalmente lo que usted nos sugiere.

Sobre los programas "largos", habrá usted visto en nuestro número 5 que ya vamos presentando algunos más complicados. Estamos en ello, pero tampoco es demasiado el material que se produce para el Spectrum de 48 K. Tal vez porque, siendo más caro, se ha vendido menos en España que el de 16 K. También en esta cuestión esperamos la colaboración de los lectores.

Para un programa en código máquina puede utilizar la instrucción USR dirección, donde se indica al ordenador comenzar a utilizar el programa en código máquina existente a partir de la dirección indicada. El retorno al programa en BASIC se produce cuando el programa en código máquina encuentra la instrucción RET, equivalente al RETURN de la subrutina.

**Pregunta:** Poseo un Spectrum de 48 K y el uso principal para el que lo he adquirido es el educativo, tanto para mí como para mis hijos. Profesionalmente podría serme de utilidad —soy titulado mercantil— y esta es mi pregunta: ¿con la adquisición de los *microdrives* y la consiguiente ampliación de memoria, podría llevar contabilidades de empresas? ¿Es posible la conexión de una impresora de papel continuo para le listado de diarios y balances? Actualmente poseo el programa de Contabilidad General editado por Investrónica, y una impresora ZX, que cumplen su cometido pero no sirven (por su formato) para presentar ante un posible cliente que solicite de mis servicios.

Conozco los ordenadores profesionales, pero su precio, por ahora, no es asequible o más bien amortizable, y mi idea era empezar con lo que tengo y conforme crezcan mis necesidades crecerán mis posibilidades de adquirir alguno.

**José Ramón Macía  
Málaga**

**Respuesta:** Es posible poder llevar contabilidades de empresas. El único problema está en el volumen de datos. La cinta del *microdrive* le permite tener una memoria externa de 80 K, pero sólo es eso, una cinta. Es más rápida que el *cassette*, pero no es comparable a un disco y, por tanto, no podrá acceder de forma rápida y eficiente a la información. Estamos preparan-

do un programa que utiliza archivos de datos en un *microdrive* para ilustrar su utilización, con lo que podrá usted ver las diferencias respecto del disco. Acerca de la impresora, ya hay en el mercado diferentes modelos. Si sólo se trata de sustituir la criticada ZX Print (que ha dejado de fabricarse) por otra de similar precio y formato, una buena idea es comprar la GP-100, de Seikosha, que trabaja con rollos de papel normal. Si lo que prefiere es trabajar con una de papel continuo, supongamos una del tipo Centronics, sólo necesitará un *interface* paralelo para su Spectrum. La firma Ventamatic comercializa uno.

**Pregunta:** He leído su artículo en el que hacían referencia a un circuito que permitía, aprovechando la salida de impresora del ordenador, conecta relés electromagnéticos de dos posiciones, que se accionarían por programa. ¿Es posible controlar relés electromagnéticos utilizando la salida de impresora del ZX Spectrum? Si es así, ¿podrían dedicar el apartado de Montajes de un futuro ejemplar a mostrar el circuito necesario para ello?

**Javier Ortega  
Valladolid**

**Respuesta:** Es posible dicha conexión y tomamos buena nota de su sugerencia.

¿TIENES? O  
¿NO TIENES?

**ZX Spectrum**

en ambos casos, tomate  
una copa  
con nosotros.



**World-Micro s.a.**

Avenida del Mediterráneo, 7  
Teléfonos 251 12 00 - 251 12 09  
Madrid-7

**DIGICO**

Los mejores  
y más populares programas.  
Todos originales.

PROGRAMAS SPECTRUM		PROGRAMAS ORIC	
1	JETPAC 16K	21	MAZE DEATH RACE "
2	COOKIE "	22	ATIC ATAC "
3	PSSST "	TODOS A 1800 Ptas.	
4	TRANS AM "	PROGRAMAS ORIC	
5	JUMPIN JACK "	23	CENTIPED 16K
6	AH DIDDUMS "	24	THE ULTRA "
7	BRAIN DAMAGE "	25	LIGHT CICLE "
8	MOLAR MAUL "	26	HARRIER ATACK "
9	HARRIER ATACK "	27	GALAXY 5 "
10	JUNGLE TROUBLE "	28	DINKY KONG 48K
11	LUNAR JET MAN 48K	29	STAR FIGHTER "
12	PANIC "	30	JOGGER "
13	KRAZY KONG "	TODOS A 2100 Ptas.	
14	ZIP ZAP "	Y muchas más, manda un sobre con sello y dirección para catálogo.	
15	STONKERS "		
16	ZOOM "		
17	ALCHEMIST "		
18	ROBOT RIOT "		
19	FREZ BEES "		
20	STAR ENTERPRISE "		

Envía a DIGICO. Plaza de Baix, 2, Eliche (Alicante)

Nombre _____	Cantidad	No. Art	Precio
Apellido _____			
Dirección _____			
Población _____			
<input type="checkbox"/> Incluye talón bancario	Total		
<input type="checkbox"/> Giro postal			

Muchos son los lectores que nos han preguntado sobre la forma de proteger un programa ante la posible manipulación de "intrusos". De eso vamos a hablar a continuación, de conservar intacta la idea del programador. Lo que no es posible evitar de manera eficaz es que un programa sea duplicado o copiado. Esto es debido a que existe un sistema para hacer copias contra el que no hay una protección eficiente; copiar de un aparato a otro. El defecto que tiene este sistema es la mala calidad de las copias. Ocurre, a menudo que un programa copiado con este sistema no funcione al cargarlo en el Spectrum, debido a que la posición de la cabeza magnética o *azimuth* varía en mayor o menor grado de unos aparatos a otros.

Para evitar que alguien cambie nuestros programas, lo primero que hay que hacer es grabar con AUTO RUN, es decir, escribir cada vez SAVE "nombre" LINE x y así cuando se cargue de nuevo el programa con LOAD "" o LOAD "nombre", este se ejecutará automáticamente, o sea, sin necesidad de escribir RUN, a partir de la línea de programa referenciada.

De esta manera, para ver el listado del programa éste habrá de ser parado pulsando "BREAK", y será necesario escribir LIST seguido de "ENTER", naturalmente. A no ser que estemos en un INPUT, en donde deberemos pulsar STOP.

Sabiendo esto, ya conocemos la manera de proteger nuestro programa para que no sea interrumpido: inutilizar el BREAK y el STOP (siempre y cuando grabemos con AUTO-RUN). Veamos distintas formas de realizar esta protección:

a) Cambiar a cero el valor contenido en la variable

del sistema DESZ, cuya dirección es 23659. Su función es indicar el número de líneas de que consta la parte inferior de la pantalla. Esta es usada para introducir los datos correspondientes a una sentencia INPUT, para escribir comandos directos (aquellos que han de ser ejecutados directa o inmediatamente, al contrario de las líneas de programa que son almacenadas previamente), para escribir, editar y corregir líneas de programación y finalmente para escribir sus mensajes el ordenador.

Por tanto si introducimos el valor cero en esta posición (mediante POKE 23659,0) suprimiremos la parte inferior de la pantalla, con la consecuencia de que los mensajes no podrán ser escritos en ninguna parte. Lógicamente, en cuanto pulsemos "BREAK" o intentemos de cualquier otra manera interrumpir el programa en su ejecución, causaremos un mensaje de error como por ejemplo el correspondiente a pulsar "BREAK", "BREAK into program". Pero al no existir las líneas en las que han de ser escritos los mensajes, el ordenador, o mejor dicho, el sistema operativo quedará inutilizado indefinidamente en un intento desesperado de escribir un mensaje en una parte de la pantalla no existente. Nosotros lo único que veremos será cómo la pantalla entera se torna de color negro y nos daremos cuenta de que el ordenador ya no reacciona ante ninguna tecla. Se habrá producido el llamado "CRASH" del sistema. En este caso la información seguirá en la RAM. Usando otros métodos, como ya veremos, ésta se pierde al producirse una especie de auto-destrucción.

En conclusión, el programa que esté grabado con AUTO-RUN y que conten-

ga un POKE 23659,0 será imposible de parar. La cuestión tiene un fallo: Al suprimirse las dos líneas de la parte inferior de la pantalla no se podrá usar el INPUT. Para remediar esto se pueden hacer dos cosas, susti-

tuir el INPUT con un diminuto programa o deshacer la protección inmediatamente antes del mencionado comando y volver a poner POKE 23659,0 inmediatamente después. Veamos un ejemplo de cada:

```

10 POKE 23659,0
20 REM
30 REM
40 POKE 23659,2: INPUT "Datos
? "; a$: POKE 23659,0
50 REM
60 REM

```

```

10 POKE 23659,0
20 REM
30 REM
40 POKE 23659,2: INPUT "Datos
? "; LINE a$: POKE 23659,0
50 REM
60 REM

```

```

10 POKE 23613,0
20 REM
30 INPUT "Datos ? "; LINE a$
40 REM

```

```

10 POKE 23613,0
20 REM
30 INPUT "Datos ? "; a$
40 REM

```

Como bien sabemos, un INPUT se puede interrumpir (en caso de variable alfanumérica borrando previamente las comillas), introduciendo "STOP". Por tanto, el programa número uno se puede parar, porque si detenemos el programa en el INPUT, no se llegará a ejecutar el POKE 23659,0 que hay detrás.

Este punto vulnerable se puede proteger usando una variante del INPUT, el INPUT LINE (véase programa número 2). Dicho LINE provoca que no aparezcan las comillas y por tanto no se pueda "salir" del INPUT. Si escribiésemos "STOP" sería aceptado como el valor de la variable a\$.

Después del INPUT LINE volvemos a proteger el programa (ver programa 2) con un POKE 23659,0.

b) Introducir el valor cero en la posición de memoria 23613. Esta posición es la variable de sistema ERRSP. Su longitud no es de un byte como en DFSZ, sino de dos. Estos dos bytes significan una dirección. Esta ha de ser calculada del siguiente modo:

La posición 23613 contiene el llamado *low order byte* o byte de orden bajo de la dirección. La posición 23614 contiene el *high order byte* o byte de orden alto.

Para saber la dirección tenemos que multiplicar el byte *high* por 256 y sumarle al resultado el byte *low*. En este caso la dirección indica dónde está el programa en C/M (código máquina) que ha de ser ejecutado en caso

de que ocurra un error. Este programa cuyo comienzo es indicado por ERRSP suele ser uno en la ROM. Normalmente este asunto sólo debería de preocupar al intérprete BASIC y no a nosotros.

Pero nosotros somos muy entrometidos y lo que hacemos es introducir O en dicha posición. O mejor dicho, nos basta con cambiar a O el byte bajo. De este modo el programa que es ejecutado al ocurrir un error ya no es el que debiera. El programa actual es un programa que provoca un RANDOMIZE USR O.

La sentencia que acabamos de mencionar provoca un abandono del BASIC (aunque a veces sólo es momentáneo) y la ejecución del programa en C/M que exista a partir de la posición indicada (no tiene por qué ser cero).

RANDOMIZE USR O provoca la ejecución del programa en C/M a partir de la posición O. Esta es la misma por la que comienza a funcionar el ordenador inmediatamente después de ser encendido éste. Como ya sabemos, lo que hace al ser puesto en marcha es llenar toda la memoria RAM de ceros y escribir en la parte inferior de la pantalla el mensaje "© 1982 Sinclair Research Ltd". En resumen, esto significa que al ocurrir un error se borrará absolutamente todo lo que esté contenido en la RAM, nuestro programa incluido, por supuesto.

Una cosa importante a tener en cuenta al usar los dos métodos hasta ahora mencionados, es que nuestros programas no deben contener ningún fallo. De lo contrario el error provocado por éste causaría el "CRASH" del sistema o la autodestrucción del programa, y no sería este el "efecto" realmente deseado.

Por cierto, borrar un programa mediante RANDOMIZE USR O es un modo mucho más efectivo que hacerlo con NEW, puesto que este último comando sólo borra hasta una posición de memoria determinada. Esta posición está contenida en la variable de sistema RAMTOP (23730 y 23731). Así podrá eliminar también los caracteres gráficos definidos en su caso.

c) Otra manera de evitar la manipulación de nuestros programas es introducir controles de color en cada una de las líneas de programa. Como ya sabrán los lectores, éstos se introducen directamente a través del teclado. Lo primero que tenemos que hacer es cambiar el cursor a modo "E". Para ello pulsaremos primero uno de los dos "SHIFT" (*symbol shift* y *caps shift*) y después, manteniendo el primero, pulsaremos el otro "SHIFT".

Una vez en modo "E" pulsaremos una tecla numérica entre el cero y el siete para obtener el color deseado del "PAPER" o fondo. Después tenemos que volver al modo "E" y pulsaremos de nuevo una tecla entre cero y siete, pero ahora apretaremos a la vez el "CAPS SHIFT". Así obtendremos el color necesario para el "INK" o la tinta.

Estos controles los tenemos que introducir detrás del número de la línea programa que estemos escribiendo.

Lo importante en este método es usar el mismo color para el papel o fondo y para el "INK" o la tinta. De este modo lograremos "invisible" nuestro listado. Para ver el listado habrán de ser borrados todos los controles de color. Este sistema no es el más efectivo, pero también funciona y se puede además combinar con los otros.

d) Lo que vamos a proponer a continuación también está basado en hacer molesto e incómodo el listar, ver y modificar un programa.

La cuestión es cambiar el juego de caracteres. Este está en la ROM, pero se puede usar otro cambiando la variable de sistema CHARS (posiciones 23606 y 23607). Esta contiene una dirección que ha de ser calculada como ya explicamos en ERRSP. La dirección indica dónde se halla la información de las letras, símbolos, etc. que son usados normalmente. La información está en forma de bytes, siendo cada byte la fórmula de los puntos de la matriz de puntos de cada carácter. Esta matriz viene

explicada con más detalle en el capítulo 14 del manual de programación. Lo que nos interesa a nosotros es cambiar este valor de CHARS. Al hacer esto cuando usemos un PRINT o cuando listemos el programa no veremos los caracteres normales, sino aquellos cuya información esté contenida a partir de la dirección indicada por CHARS.

Realmente esta variable del sistema contiene la dirección del juego de caracteres menos 256. Cuando cambiemos la dirección el nuevo valor deberá estar entre cero y 15105 o entre 15515 y 65535.

Este método también puede ser combinado con los demás.

Antes de cada PRINT

*photos copy*  
**Compra tu SINCLAIR**  
 en *photos copy*

RECLAM.

(Servimos a toda España)

<b>sinclair ZX Spectrum</b>	48K	52.000.-
<b>sinclair ZX Spectrum</b>	16K	39.900.-
<b>sinclair ZX81</b>		14.975.-

Incluimos con el equipo:

- Un libro de programación ó 5 programas.
- La inscripción gratuita por 6 meses, al CLUB SINCLAIR DE PHOTO COPY, con lo que conseguirás los mejores precios en Software y Hardware, además de la más completa información sobre SINCLAIR.

*photos copy*

Envíos contra giro  
o talón bancario

C/Teresa Herrera, 9  
Tel: (981) 21 34 21 LA CORUÑA

## IDEAS

deberemos volver a cambiar CHARS a su valor inicial, 15360. Después podremos volver a introducir un valor distinto. La fórmula para cambiar una dirección de dos bytes de longitud es: POKE

posición, dirección-256 \* INT (dirección/256) y POKE posición + 1,INT (dirección/256).

El programa cuatro es un ejemplo de cómo cambiar el juego de caracteres.

como si hubiésemos apretado la tecla dos veces. Esto es el llamado "AUTO-REPEAT" o la autorrepeticion.

El valor del tiempo que tenemos que tener la tecla apretada está contenido en la posición de memoria 23561, la variable de sistema REPDEL. Una vez apretada la tecla (apretada y mantenida, por supuesto), ésta empezará a "repetir" continuamente hasta que la soltemos. El valor del tiempo que transcurre entre una repetición y otra está en la variable de sistema REPPER (pos. 23962).

Podemos introducir los mínimos valores posibles en estas dos posiciones para conseguir hacer indomitable el teclado. Nótese que los valores mínimos no son

el cero, sino que, debido a la manera de la que son interpretadas las mencionadas dos posiciones, son el uno.

Al tener valores tan pequeños en REPDEL y REPPER cuando apretemos una tecla esta, o mejor dicho, el carácter que presenta, aparecerá no sólo una, sino siete u ocho veces en donde estemos escribiendo. Si por ejemplo intentamos escribir LIST, aparecerán 7 u 8 LIST's en la parte inferior de la pantalla y si pulsamos "ENTER" esto no será aceptado por la computadora y aparecerá un interrogante detrás del primer LIST.

Por supuesto, al igual que los métodos 3.º, 4.º; éste puede ser combinado con los demás.

```

36100 CLEAR = 000000
36200 +256*PEEK 23561+256*PEEK 2
36300 FOR t=0 TO 7
36400 POKE 30000+7-t*8*j,PEEK (t+
e+8*j)
36500 NEXT t
36600 NEXT j
36700 POKE 23561,48
36800 POKE 23562,116
36900 STOP
    
```

e) Y por último otro método: imposibilitar el manejo del teclado. Antes que nada hemos de saber que

cuando presionamos una tecla y la mantenemos apretada durante cierto tiempo, el ordenador interpretará esto

**NOS  
HEMOS  
TRASLADADO**



\*\*\*

les comunica su nueva dirección:

**C/BRAVO MURILLO, 377**

en la plaza de castilla

**Telf. (91) 733 96 62**

Madrid-20

## LIBROS

**Guía Práctica del  
BASIC ZX81  
y del Spectrum**  
Ramón Rovira Soligo  
Ediciones Técnicas  
Rede/Ventamatic  
148 páginas

Muchos son los temas que trata este ambicioso libro en sus casi 150 páginas. Está compuesto de 20 capítulos dedicados al ZX81 y un apéndice sobre el Spectrum. Por tanto, aunque los caracteres de la portada ponen énfasis en el ZX81, pensamos que el título del libro puede inducir a error.

Sin ánimo de ser exhaustivos éstos son los temas de los 20 capítulos, bien estructurados por el autor: Introducción a los ordenadores, Instalación y Mantenimiento, Estructura del teclado y errores de sintaxis, Variables y Constantes, Instrucciones principales del BASIC, Funciones, Caracteres, Condiciones y operadores lógicos, Bucles, Tablas, Gráficos, Impresora, Almacenamiento de datos en *cassette*, Estructura de la memoria, Instrucciones PEEK y POKE, Variables del sistema, Código Máquina, Ahorro de memoria, Comparación con otros BASIC y Códigos de error.

Demasiados temas, tal



vez, para tratar en tan pequeño volumen. Pero a pesar de esa limitación, Ramón Rovira consigue que el lector pueda llegar a tener una visión general del ZX81. En algunos casos, esa visión es excelente, como ocurre en el capítulo dedicado a las Tablas, donde pueden verse variables de tres dimensiones, o en la parte de ahorro de memoria, algo que resulta vital para trabajar con el ZX81 de 1 K. En cambio, la visión se reduce a apenas media página cuando se aborda el tema de la impresora.

Finaliza el libro, como decíamos, con un apéndice sobre el Spectrum. Hay que decir que es francamente bueno, y una magnífica guía para conocer sus diferencias.



**Spectrum. Libro de  
Juegos.**  
Mike James, S. M. Gee  
y Kay Ewbank  
Ed. Díaz de Santos  
134 páginas

Díaz de Santos es una librería afincada en Madrid y Barcelona, desde siempre especializada en libros técnicos y científicos, que últimamente está prestando una gran atención a la informática. No sólo cuenta con sectores específicos en sus locales, sino que ha comenzado a editar en castellano libros originalmente publicados por Granada Publishing en Inglaterra. En nuestro número anterior hemos comentado otro libro de los mismos editores y

toca ahora el turno a este Libro de Juegos.

Todo el libro está dedicado, en efecto, a presentar juegos para el Spectrum (son 21 en total, lo que no es mucho para un libro de su precio). Pero sería erróneo verlos sólo como juegos. Se proporciona el listado, una imagen gráfica de pantalla y la información precisa para utilizar el programa, además de (esto es lo más importante) indicaciones sobre el listado para aclarar la utilización de caracteres gráficos, caracteres en blanco, etc., estructura de subrutinas (se realiza una buena programación separando claramente las distintas subrutinas y explicando su significado), detalles de programación y mejoras posibles.

La importancia didáctica de este libro es, pues, su mejor característica. No es una mera colección de programas, sino un instrumento de aprendizaje, al detallarse en sus páginas las distintas partes de cada programa, su funcionamiento y su estructuración. De ello surge una programación clara, sencilla y accesible para cualquiera.

En definitiva, se trata de un libro ameno e interesante para todos aquellos que deseen contar con una biblioteca de juegos.

\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*

# ¡PEGATINAS!

\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*°\*

NUESTROS JOVENES  
LECTORES QUE QUIERAN  
RECIBIR GRATUITAMENTE  
PEGATINAS DE ZX  
TIENEN QUE SOLICITARLAS  
POR CARTA

(no se aceptan pedidos por Tel.)

A NUESTRAS OFICINAS:  
C/BRAVO MURILLO, 377 - 5º A.  
MADRID - 20.

INDICANDO CLARAMENTE  
SU NOMBRE Y DOMICILIO

El circuito que proponemos es muy simple, consiste esencialmente en dos circuitos integrados, un decodificador (IC1) y un *buffer* tri-estado (IC2).

Cinco pulsadores, uno para posicionar el cursor de disparo hacia arriba, otro hacia abajo, a la derecha, a la izquierda y un quinto botón para efectuar el disparo ejercen el control. Unas pocas resistencias, tres condensadores y un conector de 28 patillas (para conec-

tar el circuito al Spectrum completan el montaje.

En el circuito se pueden distinguir dos partes totalmente diferenciadas. Una la componen los pulsadores y un generador de código, de tal manera que al actuar sobre los pulsadores se genera un código determinado. La otra es un *interface* para aplicar dicho código al ordenador.

Mi teclado se compone de los cinco



# Joystick de disparo para el Spectrum

# MONTAJES

pulsadores mencionados, que se montarán en la parte superior de la caja que alberga a la placa de circuito impreso. Un contacto de cada uno de los pulsadores, se conecta a masa (potencial de OV.).

El otro contacto de cada pulsador se conecta al potencial positivo de +5v, a través de una resistencia de 4 K7 .

El pulsador mantiene su circuito normalmente abierto cuando no se presiona sobre él, por tanto los terminales 11, 12, 13, 14, 15, 16, 17 de IC2 están puestos a potencial positivo ("1" lógico) a través de las resistencias R1 a R7. Al pulsar cualquiera de ellos, estamos poniendo a masa ("0" lógico) el terminal correspondiente del inte-

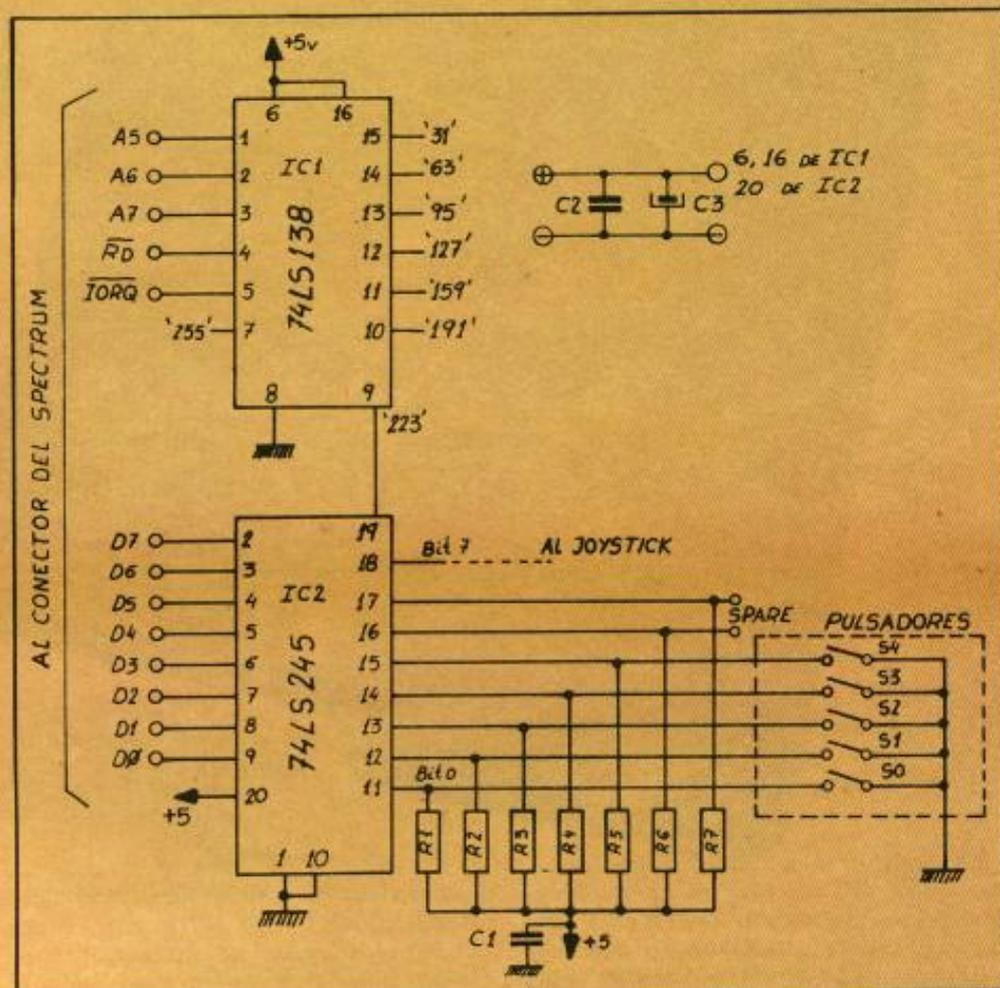


Fig. 1 Esquema eléctrico del circuito.

## LISTA DE COMPONENTES

### Resistencias

R1 = 4K7 1/8 w  
 R2 = 4K7 1/8 w  
 R3 = 4K7 1/8 w  
 R4 = 4K7 1/8 w  
 R5 = 4K7 1/8 w  
 R6 = 4K7 1/8 w

Varios 5 pulsadores  
 1 zócalo 16 pastillas  
 1 zócalo 20 pastillas

### Condensadores

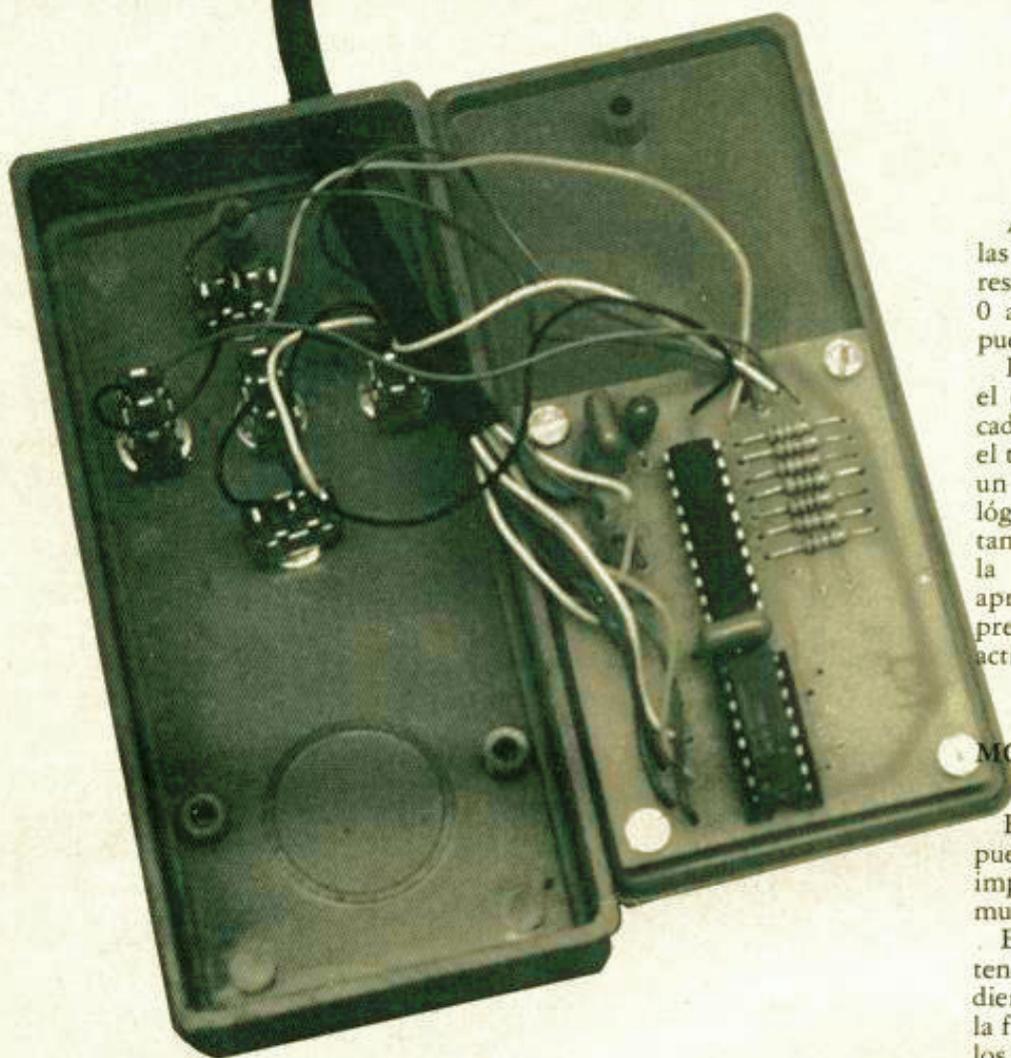
C1 = 100 kpf Miniatura  
 C2 = 100 kpf Miniatura  
 C3 = 0,1µf Tántalo 16v

### Integrados

IC1 = 74LS 138  
 IC2 = 74LS 245

cables de colores  
 1 conector de 28 pines en doble cara para el bus del Spectrum.

# Joystick de disparo para el Spectrum



grado IC2. En una palabra, el hecho de pulsar un botón, significa poner a "0" ó "1" el bit correspondiente y con ello variar el código de entrada al integrado IC2.

Los circuitos integrados IC1 e IC2 forman un *interface* entre los pulsadores y el Spectrum.

El circuito integrado IC1 es un decodificador, concretamente el 74 LS 138, conectado de tal manera que detecte cuando la CPU Z80 (CPU del Spectrum) efectúa la lectura de un número especificado en las direcciones de entrada/salida.

Cuando esta señal de lectura llega a la patilla correspondiente de IC1 (4), este integrado genera una señal a nivel bajo en una de sus patillas de salida. El nivel bajo en la patilla 9 de IC1 (corresponde a la dirección 192) se usa

para activar la salida de líneas del integrado IC2. Este integrado es el 74L245 y se trata de un *buffer* triestado. La señal de las líneas procedentes de los pulsadores se conectan a las entradas del integrado IC2 y así, cuando la dirección 223 es leída (IN 223), las señales procedentes de los pulsadores, se transfieren al ordenador.

Las ocho líneas de salida (D0 a D7) del circuito integrado IC2 corresponden a las ocho líneas del bus de datos del Spectrum. Este es el camino a través del cual las instrucciones se transfieren entre la CPU Z80 y la memoria ROM, RAM o los *ports* de entrada/salida. Todas las palabras del ordenador, se representan en el formato de 8 bits (unos o ceros, H o L) correspondientes a las ocho líneas del bus de datos.

Así en la figura 1 podemos ver que las señales procedentes de los pulsadores S0 a S4 se conectan vía IC2 a los bit 0 a 4 del bus de datos (un joystick se puede conectar al bit 7).

Las palabras de 8 bits representan en el ordenador un número binario y así cada combinación de señales leídas en el teclado de pulsadores corresponden a un número en particular. La palabra lógica (compuesta de 0s y 1s) y por lo tanto el valor del número obtenido en la entrada del *port* (determinado al apretar uno de los pulsadores) está presente en el momento en que se activa el *buffer*.

## MONTAJE

Este simple y sencillo circuito, se puede montar en la placa de circuito impreso diseñada a tal fin y que se muestra en la figura 2.

Empezaremos montando las resistencias de 4K7 en su lugar correspondiente, para ello podemos guiarnos de la figura 3 en la que se pueden apreciar los componentes montados sobre la placa. Seguiremos montando los tres condensadores de desacoplo, observando la polaridad correcta del condensador electrolítico de tantalio C3, pues en caso contrario se podía destruir. Los condensadores C1 y C2 no tienen una polaridad determinada, pudiéndose montar en una u otra posición. A continuación montaremos los circuitos integrados, aunque se recomienda utilizar zócalos ya que frente al inconveniente que puede suponer el aumentar un poco el presupuesto del proyecto, tenemos la gran ventaja de poder sustituir los integrados en caso de avería, sin tener que desoldar el integrado. Esto supone casi siempre la rotura o desperfecto del circuito impreso ya que es difícil desoldar todas las patillas a la misma vez.

Para el buen funcionamiento del

# MONTAJES

circuito, es muy importante realizar unas buenas soldaduras, por lo que a lo menos expertos en el campo de la soldadura le remitimos al número 3 de esta revista, donde se dan unos consejos muy útiles para llevar a cabo y con éxito cualquier montaje.

Mucha atención a la hora de colocar los integrados ya que es imprescindible su correcta colocación.

Una vez montados los componentes sobre la placa podemos soldar los cables que unen la placa con el conector de 28 patillas, el cual hace de unión con el Spectrum (fig. 4). También se soldarán los cables de los cinco pulsadores como podemos ver en la figura 5.

Por último, no nos queda más que mecanizar la caja que alojará a la

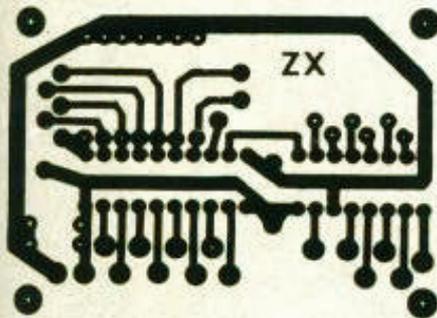


Fig. 2 Circuito impreso a tamaño natural sobre el que se montará el circuito.

plaquita de circuito impreso y sobre la que podemos montar los pulsadores.

Recordamos que este circuito se alimenta a través del conector recibiendo la alimentación del propio Spectrum.

## EL PROGRAMA

Para probar el teclado de posicionamiento, quitar el conector de alimentación del Spectrum, colocar el conector de expansión y teniendo cuidado de conectarlo correctamente en su posición adecuada, volver a conectar la alimentación. El Spectrum trabajará exactamente igual que lo hacía anteriormente, pero si aparece alguna dificultad desconecte el circuito inmediatamente y no lo vuelva a conectar hasta que haya sido verificado correctamente o se haya detectado el problema.

PULSADOR N.º	VALOR BINARIO							VALOR
	6	5	4	3	2	1	0	
L. de datos	6	5	4	3	2	1	0	
Ning. p. arret.	1	1	1	1	1	1	1	127
Puls. 0	1	1	1	1	1	1	0	126
Puls. 1	1	1	1	1	1	0	1	125
Puls. 2	1	1	1	1	0	1	1	123
Puls. 3	1	1	1	0	1	1	1	119
Puls. 4	1	1	0	1	1	1	1	111
Puls. 1 y 3	1	1	1	0	1	0	1	117
Puls. 2 y 4	1	1	0	1	0	1	1	107
Todos los p.	1	1	0	0	0	0	0	96

TABLA 1: Valores que toma el bus de datos al presionar los diferentes pulsadores.

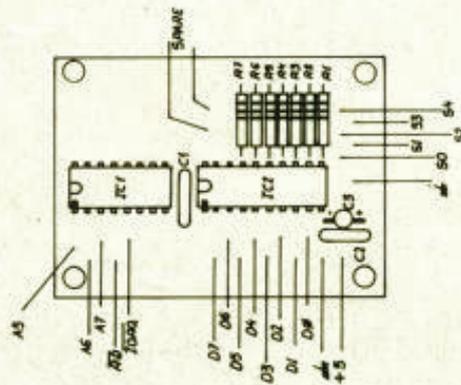


Fig. 3 Vista de la colocación de los componentes sobre la placa del circuito impreso.

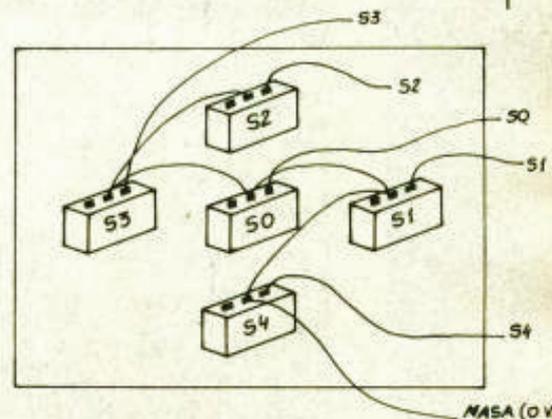


Fig. 5 Cableado de los pulsadores S0 a S4.

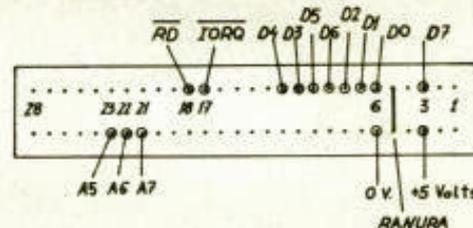


Fig. 4 Diagrama de cableado del conector de 28 pines para conectar el circuito al Spectrum. Visto por el lado del cableado.

Si todo está correcto, escriba y corra este programa:

```

10 REM Prueba del teclado
20 LET A = IN 223: REM Leer la entrada del port
30 IF A > 127 THEN LET A = A - 128
40 PRINT A
50 POKE 23692,2
    
```

60 GOTO 20  
Si ninguna de las teclas está pulsada, el valor que tenemos será 127. Probar a pulsar cada una de las demás teclas y observar qué valores se obtienen en el port.

La tabla 1 muestra cuáles serán los valores que representan a cada pulsador, cuando cada uno de ellos sea accionado. Vemos que en la tabla se representan tanto los valores en código binario, como en decimal generador al apretar cada pulsador. Si al accionar cada uno de los pulsadores por separado no obtenemos los valores indicados en la tabla 1, con toda certeza nos hemos equivocado al conectar los pulsadores o algún otro error habremos cometido. En este caso se recomienda reparar todas las conexiones.

Todos los componentes necesarios para realizar este montaje se detallan en la lista de componentes.

**COMPRA EL BRIC DE MAYO EN TU QUIOSCO**

# Bric

**LA TECNOLOGIA QUE SE ENTIENDE**

**El mayor túnel de viento para investigaciones espaciales**

**Televisores de bolsillo en color**

**Omac 1: pequeño avión de diseño revolucionario**

## Así se pilotan **LOS NUEVOS ULTRALIGEROS**

**Construya una casa de espuma sintética**

**Novísimas aplicaciones de la holografía**

**FOTOGRAFIA**  
Los últimos modelos de cámaras autofocus

**VIDEO**  
Los generadores de efectos especiales

**ELECTRONICA PASO A PASO**  
Montaje de un cuentarrevoluciones digital

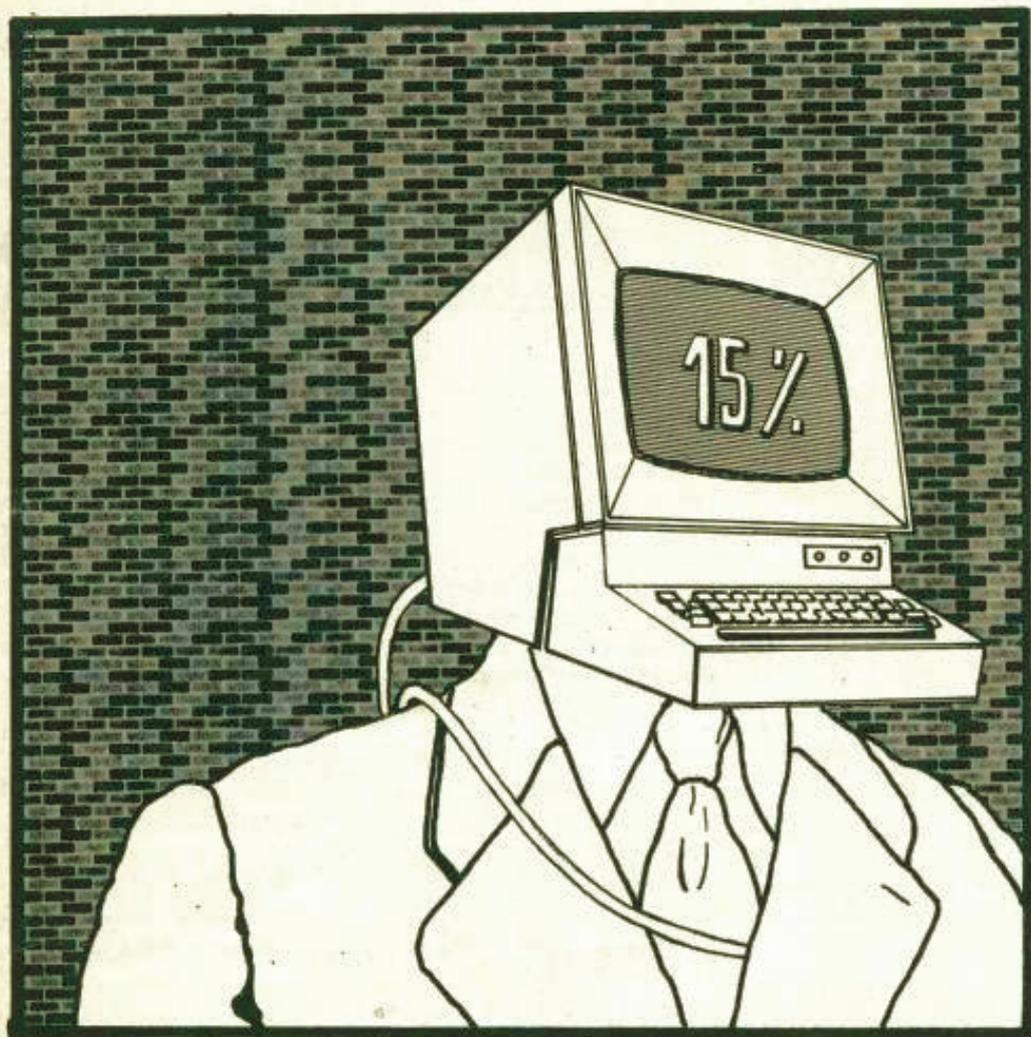
**INFORMATICA**  
Las impresoras más económicas para su ordenador personal

**BRICOLAGE**  
Cómo hacer un panel para sus herramientas

Mayo 1984 150 Pta

**NUEVO PROGRAMA DE  
JUEGOS PARA SU  
SPECTRUM**

El más brillante avance mensual de los hallazgos tecnológicos en todo el mundo. La revista de los que aman la tecnología.



GANADOR  
DEL **ZX** DE  
ESTE MES  
**MICRODRIVE**

Este mes el azar ha querido beneficiar a uno de nuestros lectores de Andalucía, Jesús Benítez, que vive en Chiclana, provincia de Cádiz, es el ganador del sorteo del ZX Microdrive correspondiente a los programas recibidos en abril. Y esto, a pesar de que su programa todavía no ha sido publicado. Esperamos que sea un aliciente para que Jesús siga colaborando con nosotros.

Si Vd. pretende conocer cuanto dinero tendrá en su cuenta bancaria al cabo de un cierto tiempo, seguro que multiplica el saldo de su cuenta por el tipo de interés. Pero se equivocará, ya que las cantidades variarán a lo largo de ese tiempo por efecto de los intereses. Lo que ha de calcular es un interés compuesto, y de eso este programa sabe mucho. (ZX81-1K).

# INTERES COMPUESTO

```

100000S CLS
100010 PRINT "CAPITAL INICIAL ";
100020 INPUT CI
100030 PRINT CI
100040 PRINT "TASA DE INTERES ANUA
100050 O/O) ";
100060 INPUT TI
100070 PRINT TI
100080 LET TI=TI/100
100090 PRINT "N. DE PERIODOS ANUAL
100100 DE COMPO-SICION DE INTERES ";
100110 INPUT NP
100120 PRINT NP
100130 PRINT "N. DE ANNOS TRANSCUR
100140 RIDOS ";
100150 INPUT NA
100160 PRINT NA
100170 LET CA=CI*(1+TI/NP)**(NP*NA
100180 PRINT "CAPITAL AL FINAL DEL
100190 ANNO ";
100200 PRINT CA
100210 PRINT "OTRA VEZ?"
100220 INPUT K$
100230 IF K$="S" THEN GOTO 140
100240 IF K$="N" THEN GOTO 5
100250 IF K$="?" THEN GOTO 210
100260 GOTO 120

```

```

CAPITAL INICIAL 500000
TASA DE INTERES ANUAL (EN O/O) 1
N. DE PERIODOS ANUALES DE COMPO-
SICION DE INTERES 5
N. DE ANNOS TRANSCURRIDOS 5
CAPITAL AL FINAL DEL ANNO
525500.3

```

OTRA VEZ?



Caza 4

# EL CAZADOR

¿A qué hacía tiempo que no iba de caza? Le proponemos que deje las botas guardadas, vende su escopeta y se venga con nosotros a cazar patos o lo que quiera. Le espera un maravilloso cielo azul con una nube blanca no menos maravillosa. Sólo tiene que apuntar con su Spectrum (teclas "5, 6, 7 y 8) y disparar (tecla f). No es fácil acertar. Además, cada vez que dispare —dado que su Spectrum tiene un gran retroceso— ha de ajustar de

nuevo su tiro. Otro problema con el que se encontrará es que su pájaro tiene una predilección especial por la única nube que hay sobre el firmamento, con lo que le será un poco más difícil acertarle. Como hay huelga de controladores sólo salen cinco pájaros y siempre por separado. Adelante, afine su puntería y compre algún "bichito" en la tienda de al lado para no quedar mal.

(16K-Spectrum).

```

10 REM Caza.
20 GO SUB 8000
30 FOR g=1 TO 5
40 GO SUB 3000
45 PRINT AT 21,5;"Caza ";g;AT
50 GO SUB 2000
60 NEXT g
70 GO TO 9000
90 STOP

1000 PLOT x-6,y
1010 DRAW 12,0
1020 PLOT x,y-6
1030 DRAW 0,12
1060 RETURN

2000 LET a$=INKEY$
2005 GO SUB 1000
2006 LET j=b(i): GO SUB 4000
2010 IF a$="5" AND x>10 THEN LET
x=x-2
2020 IF a$="6" AND y>10 THEN LET
y=y-2
2030 IF a$="7" AND y<160 THEN LE
T y=y+2
2040 IF a$="8" AND x<240 THEN LE
T x=x+2
2050 LET i=i+1: LET j=b(i): GO S
UB 4000
2060 IF a$="f" THEN GO SUB 5000
2070 GO SUB 1000
2080 IF end=1 THEN LET end=0: RE
TURN
2100 GO TO 2000
3000 CLS
3020 LET j=INT (RND*3)+1
3030 LET r=INT (RND*10)
3040 PRINT AT j,r+9; INK 7; PAPE
R 7;" "; REM 2 espacios.
3050 PRINT AT j+1,r+8; INK 7; PA
PER 7;" "; REM 4 espacios.
3060 PRINT AT j+2,r+10; INK 7; P
APER 7;" "; REM 3 espacios.
3080 LET end=0
3090 LET x=INT (RND*60)+40
3100 LET y=75
3110 PAPER 5
3120 INK 0
3140 LET i=3
3150 GO SUB 1000

```

```

3160 LET j=b(i): GO SUB 4000
3170 RETURN

4000 INK 8: PAPER 8
4005 PLOT i-3,j+2
4010 PLOT i-2,j+2
4020 PLOT i-1,j+1
4030 PLOT i,j
4040 PLOT i+1,j+1
4050 PLOT i+2,j+2
4060 PLOT i+3,j+2
4070 PLOT i-4,j+1
4080 IF i>250 THEN LET end=1
4090 RETURN
5000 BEEP .1,-5
5010 IF POINT (x,y)<>1 THEN LET
x=x+10-INT (RND*20): RETURN
5020 LET x=x+10-INT (RND*20)
5030 BEEP .3,-5
5040 PRINT "Impacto!!"
5050 LET h(g)=h(g)+1
5060 RETURN

8000 LET hit=0
8010 CLS
8020 PRINT AT 10,2;"S P E C T R
U M
C A Z A"
8050 DIM b(254)
8070 FOR i=1 TO 254
8080 LET b(i)=150-(125-i)*(125-i
)/200
8100 NEXT i
8120 OVER 1
8130 DIM h(5)
8140 RETURN

9000 CLS
9010 LET t=0
9020 FOR i=1 TO 5
9030 PRINT AT i+5,10;"Pajaros ";
i;"--5 Acertados ";h(i)
9040 LET t=t+h(i)
9050 NEXT i
9060 PRINT AT 12,10;"Total impac
tos=";t
9070 INPUT "Continuamos (s/n)";a
$
9075 IF a$="s" OR a$="5" THEN RU
N
9080 PAPER 7
9090 INK 0
9100 OVER 0

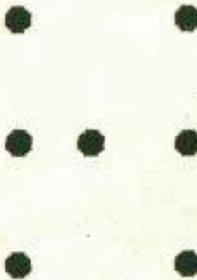
```

# PRUEBE CON LOS DADOS

¿Que no encuentra su cubilete? ¿Que dejó su ajedrez y no se lo han devuelto?... No se preocupe. No pierda de vista a su Spectrum, ya sabe que tiene en él el cajón donde meter su ajedrez, su contabilidad, su sistema de ecuaciones... o el juego de dados que le ofrecemos a continuación. Le aseguramos que el dado no irá a parar al último

rincón de la habitación. El manejo es sencillo: concentre y presione cualquier tecla. Cuando se canse de jugar pulse BREAK.  
(16K-Spectrum).

Notas gráficas:  
Línea 1100 y siguientes:  
Graphic a.



```

10 GO SUB 5000
20 GO SUB 2000
30 INPUT " "
40 PRINT AT 20,0; PAPER 0; INK
  6;"Presione cualquier tecla par
  a lanzar el dado."
50 PAUSE 0
60 LET t=RND*10+5
    
```

```

70 FOR i=1 TO t
80 PAUSE 1+i
90 INK 6
100 GO SUB 1000
110 LET r=INT (RND*6) +1
120 INK 2
130 GO SUB 1000
140 NEXT i
150 GO TO 50
1000 GO SUB 6000
1010 GO TO 1000+r*100
1100 PRINT AT 10,13;"●"
1110 RETURN
1200 PRINT AT 5,10;"●"
1210 PRINT AT 15,17;"●"
1220 RETURN
1300 GO SUB 1100
1310 GO TO 1200
1400 PRINT AT 5,17;"●"
1410 PRINT AT 15,10;"●"
1420 GO TO 1200
1500 GO SUB 1400
1510 GO TO 1100
1600 PRINT AT 10,10;"●"
1610 PRINT AT 10,17;"●"
1620 GO TO 1400
2000 FOR i=1 TO 13
2010 PRINT AT 3+i,9; PAPER 6;"
      ": REM 13 espacios.
2020 NEXT i
2030 LET r=1
2040 GO TO 1000
5000 POKE USR "a"+0,BIN 00111100
5010 POKE USR "a"+1,BIN 01111110
5020 POKE USR "a"+2,BIN 11111111
5030 POKE USR "a"+3,BIN 11111111
5040 POKE USR "a"+4,BIN 11111111
5050 POKE USR "a"+5,BIN 11111111
5060 POKE USR "a"+6,BIN 01111110
5070 POKE USR "a"+7,BIN 00111100
5080 INK 2
5090 PAPER 0
5100 CLS
5110 PAPER 6
5120 BORDER 0
5130 RETURN
6000 OUT 254,16
6010 OUT 254,0
6020 RETURN
    
```





# PROGRAMAS

```

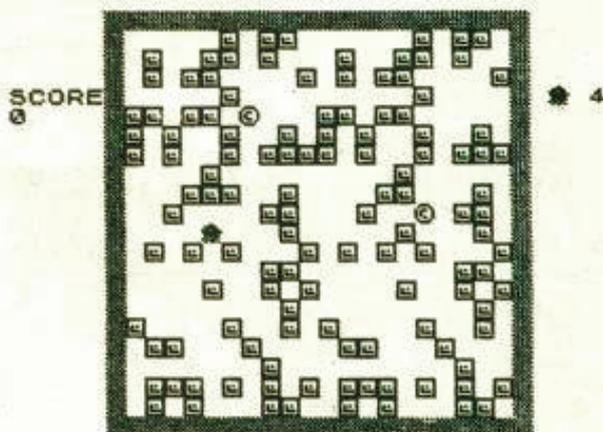
,x1-w;"@": LET w=w+1: IF CODE SC
REEN$ (y1,x1-w)=0 THEN RETURN
468 NEXT i: RETURN
470 IF CODE SCREEN$ (y1+w-1,x1)
=32 OR y1=9 THEN RETURN
472 FOR i=1 TO 4: PRINT PAPER 7
;AT y1+w-1,x1;" "; PAPER 5;AT y1
+w,x1;"@": LET w=w+1: IF CODE SC
REEN$ (y1+w,x1)=0 THEN RETURN
473 NEXT i: RETURN
475 IF CODE SCREEN$ (y1-w+1,x1)
=32 OR y1=2 THEN RETURN
476 FOR i=1 TO 4: PRINT PAPER 7
;AT y1-w+1,x1;" "; PAPER 5;AT y1
-w,x1;"@": LET w=w+1: IF CODE SC
REEN$ (y1-w,x1)=0 THEN RETURN
477 NEXT i: RETURN
480 IF CODE SCREEN$ (y1,x1+w-1)
=32 OR x1=25 THEN RETURN
483 FOR i=1 TO 4: PRINT PAPER 7
;AT y1,x1+w-1;" "; PAPER 5;AT y1
,x1+w;"@": LET w=w+1: IF CODE SC
REEN$ (y1,x1+w)=0 THEN RETURN
484 NEXT i: RETURN
500 FOR i=1 TO 28: READ n: READ
m: BEEP n+0.1,m: NEXT i: RESTOR
E 1000: FOR i=0 TO 21: PRINT PAP
ER 0;AT i,5;"
": NEXT i: LET sn=rt+2: GO T
O 100
1000 DATA 0.6,4,0.4,5,0.4,7,0.4,
7,0.4,5,0.4,4,0.4,2,0.4,0,0.4,0,

```

```

0.4,2,0.4,4,0.6,4,0.2,2,0.6,2,0.
5,4,0.4,5,0.4,7,0.4,7,0.4,5,0.4,
4,0.4,2,0.4,0.4,0.4,0.4,2,0.4,4,
0.6,2,0.2,0.9,0

```



## SPEN

C/. SAN QUINTIN, 10  
(PLAZA DE ORIENTE)  
TELEF. 247 62 82  
MADRID - 13

## LA INFORMÁTICA SIN ESFUERZO

De los invasores al inglés.

Del tratamiento de texto a la música.

### De los ficheros al Fútbol.

Si posees un ordenador tipo Sinclair, Commodore, Oric..... has comprobado la cantidad de aplicaciones que tienen. El ordenador "familiar" está preparado para dialogar contigo. Educación, gestión, pasatiempos son sólo posibilidades. Ayuda a los niños a aprender, a distraerse, al padre o a la madre a descubrir el mundo de la informática y a administrar la vida doméstica.

SPEN S.A., centro especializado en la enseñanza de la informática, ha elaborado unos cursos de corta duración para que disfrutes de tu microordenador y le saques el máximo rendimiento.



En la exposición y contenido de los mismos se ha valorado el hecho de que prácticamente todos los asistentes poseáis un microordenador, y por consiguiente después de cada tema, estareis en condiciones de realizar programas y aplicaciones correctas. Además os resolveremos todas las dificultades que surjen en la programación y fuera de ella, tales como:

### BASIC II

Duración. Uno o dos meses.

#### Objetivos.

- \* Aprender cómo se desarrolla una idea para convertirla en un programa.
- \* Modificación de programas.
- \* Características técnicas y científicas del Spectrum.

Nivel requerido. Dominio del lenguaje Basic.

Precio. 7.000 pesetas/mes (curso de dos meses).

### BASIC I:

Duración. Uno o dos meses.

Objetivos. \* Establecer una estrecha relación entre el usuario y su equipo a través del Basic, de tal forma que se pueda obtener el máximo rendimiento del mismo conociendo su capacidad, potencia y limitaciones.

Nivel requerido. Este curso no requiere conocimientos de partida en informática.

Precio. 7.000 pesetas/mes (curso de dos meses).

**CURSOS POR CORRESPONDENCIA** Basic I, Basic II, se imparten con cassettes y apuntes.

Si desea más información, llame o escriba a: SPEN S.A., C/San Quintín, 10, Tel. 247 62 82 - Madrid - 13.

— Protección de programas, utilización de variables, conocimiento de la memoria disponible, mantenimiento de archivos, programación en Código Máquina.

NO OLVIDES que estos microordenadores no sirven sólo para jugar, la informática ya no es un lujo. Dentro de muy poco tiempo será necesidad real para acceder al mercado del trabajo. Para los estudios, dominar los ordenadores será tan importante como hoy lo es una calculadora. APROVECHA ESTA OPORTUNIDAD.

### CODIGO MAQUINA

Duración. Dos meses.

#### Objetivos.

- \* Traducción del lenguaje Basic al Código Máquina a través de programas ensambladores.

- \* Dominio del juego de instrucciones del microprocesador.

Nivel requerido. Nivel BASIC II.

Precio. 10.000 pesetas/mes.

Deseo más información sobre los cursos:

Nombre \_\_\_\_\_

Empresa \_\_\_\_\_

Cargo \_\_\_\_\_

Dirección \_\_\_\_\_

Ciudad \_\_\_\_\_ Telf. \_\_\_\_\_

# MASTER MIND

Ya hemos publicado dos programas de Master mind para el Spectrum. Ahora le toca al hermano pequeño: el ZX81. Sólo necesitará 1K para utilizar el programa que nos envía. Ramón Vila desde La Coruña. Por supuesto, el programa le informa del número de jugadas que necesitó para acertar la clave (ZX81-1K).

## PROGRAMA GANADOR DE 5.000 PTAS

Este Master Mind ha sido escrito por Ramón Vilas de Escariaba, de La Coruña, a quien enviamos de inmediato un talón nominativo por 5.000 pesetas. Gracias.

```

10 PRINT "CIFRAS?"
20 INPUT A
30 LET T=PI/PI
40 LET A$=""
50 FOR I=1 TO A
60 LET B=INT (RND*10+20)
70 LET A$=A$+CHR$ B
    
```

```

80 FOR J=1 TO I-1
90 IF A$(I)(<>)A$(J) THEN GOTO 1
20
100 LET A$=A$( TO I-1)
110 GOTO 100
120 NEXT I
130 NEXT I
140 PRINT "SU NUMERO", "AC AP"
150 LET I=PI
160 INPUT PH
170 INPUT PI-PI
180 INPUT BS
190 FOR I=1 TO A
200 IF A$(I)=BS THEN GOTO 180
210 FOR J=1 TO A
220 IF A$(J)=BS THEN GOTO 400
230 IF A$(I)=BS AND A$(I)=BS(I) THEN
240 I=I+1
250 IF I=I+1 AND A$(I)=BS(I) THE
260 GOTO 100
270 NEXT I
280 PRINT T
290 PRINT BS;" ";H
300 GOTO 180
310 PRINT BS;" O.K. EN ";T
410 GOTO 10
    
```

# PLAN DE AHORRO

A la hora de ahorrar puede calcular el dinero que tendrá dentro de unos años, de acuerdo con un determinado tipo de interés, o puede hacerlo a la inversa, es

decir, decidir qué dinero desea tener en una fecha concreta para, acorde con el tipo de interés, determinar las cantidades a ingresar. Si su plan de ahorro es el

segundo puede valerse de este pequeño programa. Aquí si desea disponer de 100.000 ptas. al cabo de 12 meses para comprar el QL,

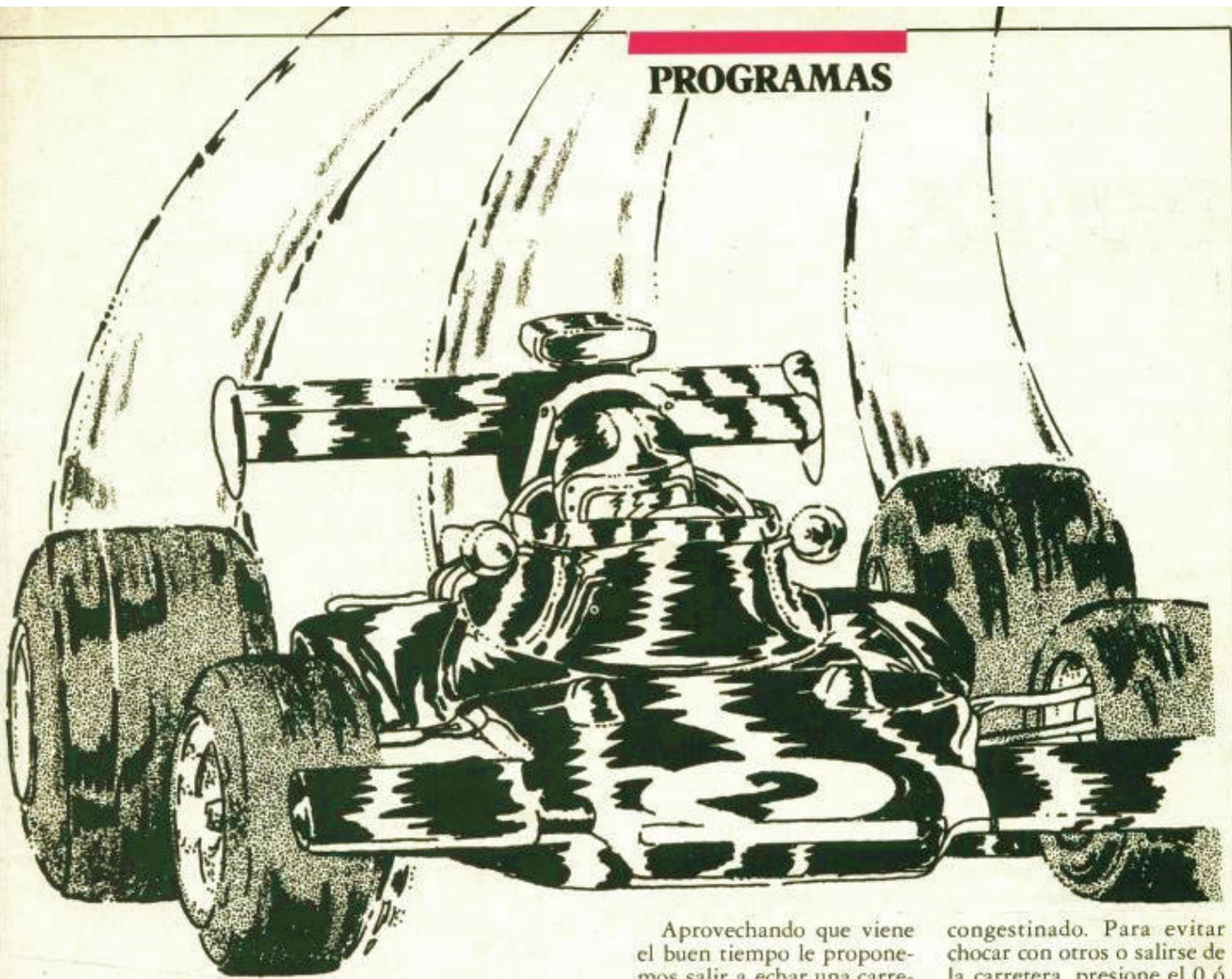
con un tipo de interés del 3,8 %, realizando ingresos mensuales necesitará abonar en su cuenta 8.028 pts./mes. (1K-ZX81).

```

5 CLS
10 PRINT "PLAN DE AHORRO"
20 PRINT "CAPITAL FINAL "
30 INPUT CF
40 PRINT CF
50 PRINT "TASA ANUAL DE INTERE
SE (EN O/O) "
60 INPUT TI
70 LET TI=TI/100
80 PRINT TI
90 PRINT "N. DE PAGOS ANUALES
"
100 INPUT NU
110 PRINT NU
120 PRINT "N. DE ANOS ";
130 INPUT NA
140 PRINT NA
150 LET UR=CF*TI/NU/((1+TI/NU)*
*(NA-1))
160 PRINT "PAGOS REGULARES ";
170 PRINT UR
180 PRINT "OTRA VEZ?"
190 INPUT K$
200 IF K$="S" THEN GOTO 200
210 IF K$="N" THEN GOTO 5
220 GOTO 190
    
```

```

PLAN DE AHORRO
CAPITAL FINAL
500000
TASA ANUAL DE INTERESES (EN O/O)
.01
N. DE PAGOS ANUALES 6
N. DE ANOS 3
PAGOS REGULARES 27366.338
OTRA VEZ?
    
```



# CIRCUITO

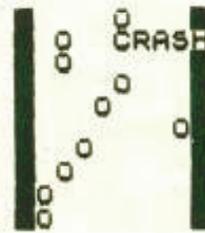
Aprovechando que viene el buen tiempo le proponemos salir a echar una carrera de coches con su ZX81. Pero tenga cuidado, hay muchos que tuvieron la misma idea y el tráfico está un poco

congestinado. Para evitar chocar con otros o salirse de la carretera, presione el 0 ó el 1, según quiera desplazarse a la derecha o izquierda respectivamente. (1K - ZX81).

```

100 LPRINT AT N, Z; "CRASH"; AT 20,
1; "RECORRIDOS "; INT (U/4); " KILO
METROS"
170 GOTO 40
160 LET U=U+1
150 SCROLL
140 IF Z>17 OR Z<10 THEN GOTO 2
130 LET Z=Z+(INKEYS="0")-(INKEY
S="1")
120 PRINT AT N, Z; A$
110 NEXT G
100 LET A$=INT (RND*8)+2
90 LET P$1 TO INT (U/170)+1
80 FOR P$1 TO INT (U/170)+1
70 LET A$="■"
60 PRINT "I"
50 IF INKEY="0" THEN GOTO 25
40 LPRINT AT N, Z; "CRASH"; AT 20,
1; "RECORRIDOS "; INT (U/4); " KILO
METROS"
300 LPRINT AT N, Z; "CRASH"; AT 20,
1; "RECORRIDOS "; INT (U/4); " KILO
METROS"
200 LPRINT AT N, Z; "CRASH"; AT 20,
1; "RECORRIDOS "; INT (U/4); " KILO
METROS"
100 LPRINT AT N, Z; "CRASH"; AT 20,
1; "RECORRIDOS "; INT (U/4); " KILO
METROS"

```



RECORRIDOS 20 KILOMETROS

# DIBUJOS

El programa se basa en las funciones circulares SIN y COS para el trazado de rectas y curvas en relación a una circunferencia, dando lugar a la formación de vistosas figuras.

El programa se halla dividido en 4 partes bien diferenciadas, en cada una de las cuales se traza un tipo distinto de dibujo, y a las que se accede por medio de las líneas 10-40, al responder afirmativamente a las preguntas del ordenador.

En las líneas 5-80 se elige el dibujo que se desea realizar y otras opciones como el borrado de pantalla, color de fondo o impresión en papel. Desde la línea 200 a 290 se realiza el primero de los cuatro tipos de dibujos posibles, la "figura giratoria", que consiste en el dibujo de un polígono regular en las distintas posiciones que se van generando al rotar en el interior de una circunferencia. El número de posiciones distintas que queremos que se dibujen se introduce en la línea 230 (densidad).

Desde la 400 a la 520 se traza el segundo tipo de dibujo, el "polígono de hilos", que se realiza dividiendo una circunferencia en tantas partes como queramos (número de lados del polígono) y uniendo entre sí todos los puntos que ha originado la división.

Entre las líneas 600-710 se crea el tercer tipo de dibujo, la "figura hecha de semicírculos", al unir con curvas dos puntos tantas veces como queramos. La curvatura de la línea que une los dos puntos se introduce en la línea 660 (ángulo de trazado): la distancia angular entre los dos puntos se introduce en la línea 650, y el número de líneas que queremos que se dibujen se introduce en la línea 640 (densidad). Desde la línea

800 a la 900 se realiza el último tipo de dibujo, la "figura hecha de círculos", al trazar una serie de círculos tomando, como centro de éstos diversos puntos pertenecientes a una circunferencia central. El número de círculos a dibujar se introduce en la línea 850 (densidad), el radio de los círculos que se trazan se introduce en la línea 830 (radio) y el radio de la circunferencia central, en torno a la cual se trazan los círculos, se introduce en la línea 820 (distancia al centro).

En cuanto a la utilización del programa. La pregunta "coordenadas" se refiere a las del centro de la circunferencia que da lugar a las figuras anteriores. Los valores de las coordenadas son los de los *pixels* coordenadas x (horizontal) entre 0 y 255, coordenada y (vertical) entre 0 y 175.

Ejemplos: Pruebe las siguientes figuras:

— Figura giratoria: coordenadas 127, 87; radio 80; no. lados 3; densidad 7.

— Polígono de hilos: coordenadas 127, 87; radio 80; no. lados 15.

— Figura de semicírculos: coordenadas 127, 87; radio 80; densidad 25; ángulo entre puntos 1; ángulo de trazado PI.

— Figura de círculos: coordenadas 127, 87; distancia al centro 50; radio 30; densidad 30.

16K Spectrum.

**PROGRAMA GANADOR DE  
5.000 PTAS**

Estos bonitos dibujos son el fruto del trabajo de Ignacio Jiménez Guerrero, lector a quien mucho agradecemos el envío de su programa

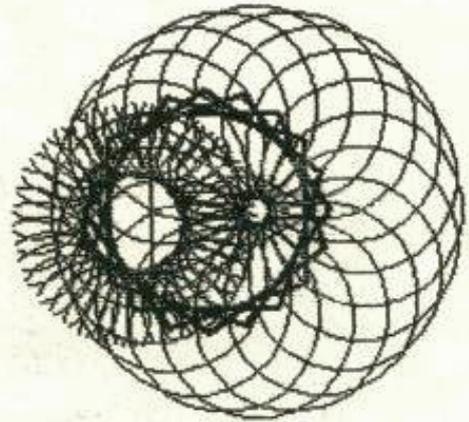
```

1 REM DIBUJOS
  © IGNACIO JIMENEZ GUERRERO,
  1984
5 CLS : INPUT "color fondo ";
c: BORDER c: PAPER c: CLS : INK
c
10 INPUT "figura giratoria? (s
/n) ";o$: IF o$="s" THEN GO TO 2
00
20 INPUT "poligono de hilos (s
/n) ";o$: IF o$="s" THEN GO TO 4
00
30 INPUT "figura de semicircul
os s/n ";o$: IF o$="s" THEN GO T
O 500
40 INPUT "figura de circulos s
/n ";o$: IF o$="s" THEN GO TO 80
0
50 INPUT "elige una opcion, pu
lsa s ";o$: IF o$="s" THEN GO TO
10
55 STOP
60 INPUT "paso a impresora s/n
";i$: IF i$="s" THEN COPY
70 INPUT "borro? (s/n)";o$: IF
o$="s" THEN GO TO 5
60 GO TO 10
200 REM figura giratoria
205 INPUT "color tinta (0-7)";t
: IF t<0 OR t>7 THEN GO TO 205
210 INPUT "coordenadas (x,y)";f
,"g
212 INPUT "radio";r: IF f-r<0
OR f+r>255 OR g-r<0 OR g+r>175 T
HEN GO TO 210
215 INPUT "no. lados";n: IF n<3
OR INT n<>n THEN GO TO 215
230 INPUT "densidad";d
240 FOR a=0 TO 2*PI STEP (2*PI)
/d
243 LET ax=f+COS a*r: LET ay=g+
SIN a*r
246 PLOT ax,ay
250 FOR b=1 TO n
255 LET x=f+COS (a+b*2*PI/n)*r:
LET y=g+SIN (a+b*2*PI/n)*r
260 DRAW INK t;x-ax,y-ay
264 LET ax=x: LET ay=y
270 NEXT b
280 NEXT a
290 GO TO 60
400 REM poligonos hechos con hi
los
405 INPUT "color tinta (0-7)";t
: IF t<0 OR t>7 THEN GO TO 405
410 INPUT "coordenadas (x,y) ";
f,"g
420 INPUT "radio";r
430 IF f-r<0 OR f+r>255 OR g-r<
0 OR g+r>175 THEN GO TO 410
440 INPUT "no. lados";n: IF n<3
OR INT n<>n THEN GO TO 440
450 FOR a=0 TO 2*PI STEP 2*PI/n
460 LET x=INT (f+COS a*r): LET
y=INT (g+SIN a*r)
470 FOR b=1 TO n
480 LET k=INT (f+COS (b*2*PI/n)
*r): LET l=INT (g+SIN (b*2*PI/n)
*r)
490 PLOT x,y: DRAW INK t;k-x,l-
y
500 NEXT b
510 NEXT a
520 GO TO 60
600 REM semicirculos
605 INPUT "color tinta (0-7) ";
ti: IF ti<0 OR ti>7 THEN GO TO 6
05
610 INPUT "coordenadas (x,y) ";
f,"g
620 INPUT "radio ";r
630 IF f-r<0 OR f+r>255 OR g-r<
0 OR g+r>175 THEN GO TO 610
640 INPUT "densidad";d
650 INPUT "angulo entre puntos
(0-PI) ";a: IF a>PI OR a<0 THEN
GO TO 650

```

## PROGRAMAS

```
660 INPUT "angulo de trazado (0  
-PI) ";t: IF t>PI OR t<0 THEN GO  
TO 660  
670 FOR b=0 TO 2*PI STEP 2*PI/d  
680 LET x1=f+COS b*r: LET y1=g+  
SIN b*r: LET x2=f+COS (b+a)*r: L  
ET y2=g+SIN (b+a)*r  
690 PLOT x1,y1: DRAW INK t;x2-  
x1,y2-y1-t  
700 NEXT b  
710 GO TO 60  
800 REM circulos  
805 INPUT "color tinta (0-7)";t  
IF t<0 OR t>7 THEN GO TO 805  
810 INPUT "coordenadas (x,y)";f  
";g  
820 INPUT "distancia al centro  
";d  
830 INPUT "radio ";r  
840 IF f-d-r<0 OR f+d+r>255 OR  
-d-r<0 OR g+d+r>175 THEN GO TO  
810  
850 INPUT "densidad ";i  
860 FOR a=0 TO 2*PI STEP 2*PI/i  
870 LET x=f+d*COS a: LET y=g+d*  
SIN a  
880 CIRCLE INK t;x,y,r  
890 NEXT a  
900 GO TO 60
```



## EL LIDER EN MICROORDENADORES EN MADRID

**MICRO** **M** **WORLD**  
**W**

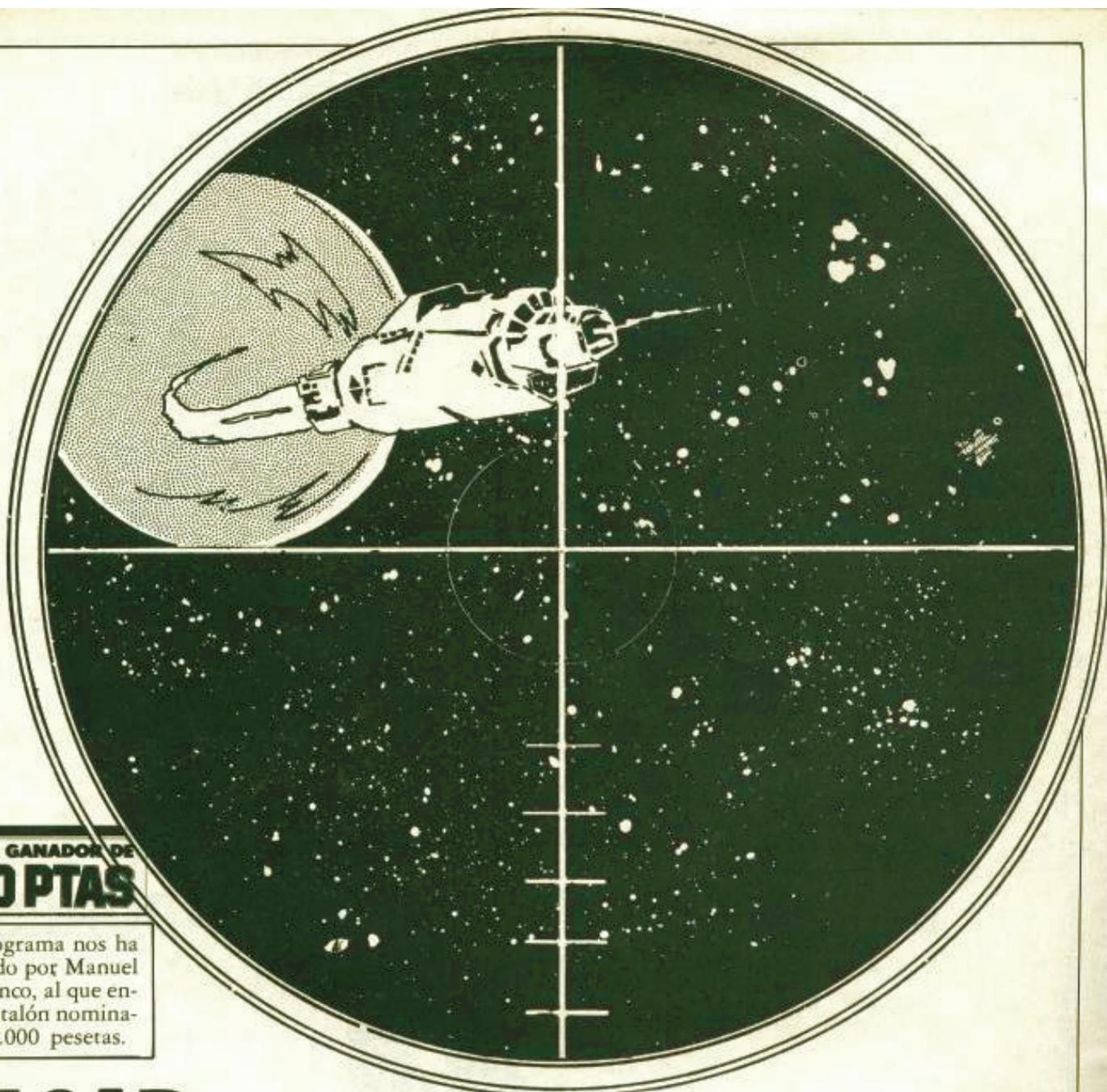
Distribuidor Autorizado de:

- Sinclair Spectrum
  - Oric-1 48K
  - New Brain
- Katson (compatible Apple II)
  - Apple IIe
  - APD/Altos

**MICRO** **M** **WORLD**  
**W**

en Madrid:  
Modesto Lafuente, 63 - Telf. 253 94 54  
Colombia, 39-41 Telf. 458 61 71  
en Villalba:  
Honorio Gonzalo, 2 - Telf. 850 66 02

Todo en microordenadores y ordenadores de gestión.



PROGRAMA GANADOR DE  
**5.000 PTAS**

Este programa nos ha sido enviado por Manuel Cuadra Ronco, al que enviamos un talón nominativo por 5.000 pesetas.

# QUASAR

Nuevamente el radar ha detectado objetos voladores no identificados, aproximándose a la tierra. Los ordenadores indican "formación de ataque". Aproveche la oportunidad para probar su nuevo avión con mira "anti-intrusos" y realizar un disparo certero. Para disponer presione el O para

desplazarse por la bóveda celeste presione las teclas 5, 6, 7, 8. (16K Spectrum).

Notas gráficas:

Graphic A: Línea 215, 224, 2000, 2001.

```

1 PAPER 0: BORDER 1: INK 5: C
L5
2 LET p=0: LET t=1000
100 REM Inicio de mira
105 GO SUB 1000
110 PLOT 125,62: DRAW 0,60: PLO
T 95,92: DRAW 60,0
115 CIRCLE 125,92,30: CIRCLE 12
5,92,25: CIRCLE 125,92,10
120 FOR n=0 TO 100: PLOT (INT (

```

```

255*RND)), (INT (175*RND)): NEXT
n
122 PRINT FLASH 1: INK 2: BRIGH
T 1: AT 21,0: " "; FLASH 1: INK 2;
BRIGHT 1: AT 21,31: " "
125 INPUT INK 0: PAPER 2: "GRADO
DE DIFICULTAD? (1-10)"; gr
200 REM Inicio de mira
201 LET x=10: LET y=15
210 LET x=x+INT (2*RND)-INT (2*
RND)+(INKEY$="6")-(INKEY$="7"):
IF x>=21 OR x<=0 THEN LET x=0
211 LET y=y+INT (2*RND)-INT (2*
RND)+(INKEY$="8")-(INKEY$="5"):
IF y>=31 OR y<=0 THEN LET y=0
215 PRINT INK 6: OVER 1: AT x,y;
" "
223 IF INKEY$="0" THEN PLOT 0,0
: DRAW INK 3: OVER 1: 125,92: PLO
T 255,0: DRAW INK 3: OVER 1: -130
,92: BEEP .007,10: PLOT 0,0: DRA
U OVER 1: 125,92: PLOT 255,0: DRA
U OVER 1: -130,92: IF x=10 AND y=
15 THEN FOR n=40 TO 50: PRINT OV
ER 1: AT x,y: CHR$ (n): BEEP .001,
20: PRINT OVER 1: AT x,y: CHR$ (n)
: NEXT n: GO SUB 2000
224 PAUSE 4: PRINT OVER 1: AT x,
y: " "
225 LET t=t-gr: PRINT PAPER 2:
INK 0: AT 21,13: "TIEMPO :"; t:
: IF t<=0 THEN GO TO 3000
230 GO TO 200

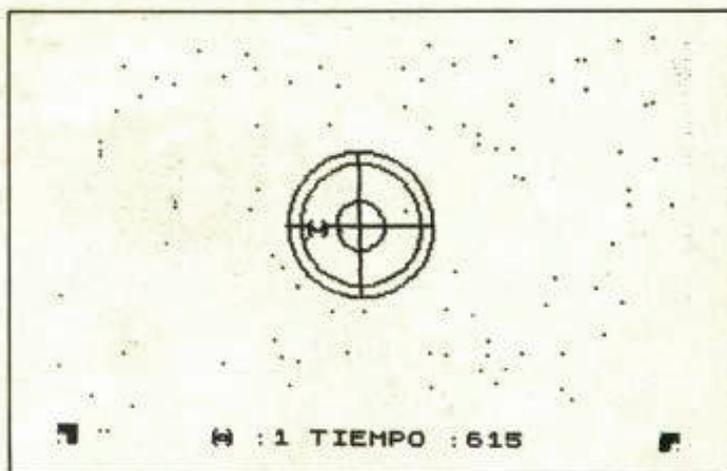
```

# PROGRAMAS

```

1000 FOR k=0 TO 7: READ c: POKE
USR "a"+k,c: NEXT k: RETURN
1001 DATA BIN 00000000
1002 DATA BIN 01000010
1003 DATA BIN 10000001
1004 DATA BIN 10111101
1005 DATA BIN 11100111
1006 DATA BIN 10111101
1007 DATA BIN 10000001
1008 DATA BIN 01000010
2000 FOR n=7 TO 1 STEP -1: PAUSE
5: BORDER n: NEXT n: LET p=p+1:
PRINT PAPER 4; INK 0; AT 21,8;"@
...:p
2001 PRINT OVER 1; AT x,y;"@"
2002 IF p=10 THEN GO TO 4000
2004 GO TO 200
3000 FOR h=0 TO 200: BEEP .009,h
/10: POKE (22528+(INT (768*RND))
), (INT (255*RND)): NEXT h: BEEP
1,40: PRINT INK 0; PAPER 5; FLAS
H 1; AT 10,8;" P E R D I S T E "
3002 PRINT #1;"PULSA UNA TECLA P
ARA EMPEZAR": PAUSE 0: RUN
4000 BEEP .2,20: BEEP .5,40
4001 BEEP .2,20: BEEP .5,40
4002 PRINT INK 0; PAPER 5; FLASH
1; AT 10,8;" V I C T O R I A "
GO TO 3002
5000 REM EL JUEGO CONSISTE EN
TRATAR DE COLOCAR LA NAVE ENEM-
IGA EN EL PUNTO DE MIRA Y DISPA
RAR EL LASER DE TU NAVE.
5001 REM PARA ELLO SE UTILIZAN
    
```

LAS TECLAS DEL CURSOR Y EL "0"  
 PARA DISPARAR.  
 5002 REM PARA COMPLETAR LA MI-  
 SION SE DEBEN DE HACER 10 IMPAC-  
 TOS SOBRE LA NAVE ENEMIGA ANTES  
 DE QUE SE ACABE EL TIEMPO.  
 5003 REM EXISTEN 10 GRADOS DE  
 DIFICULTAD.



ENVIENOS LA  
 HOJA DE PROMOCION

# C.O.S.E.S.A.

COMPAÑIA ESPAÑOLA DE SUMINISTROS ELECTRONICOS, S. A.  
 Distribuidores oficiales de INVESTRONICA

**LOS MEJORES PRECIOS  
 EN LA MAS AMPLIA GAMA  
 DE ORDENADORES  
 PERSONALES**

- Sinclair (ZX81 y Spectrum)
- Commodore ● Unitrón
- Laser ● Dragón

**Tenemos todos  
 los programas y periféricos  
 del mercado**

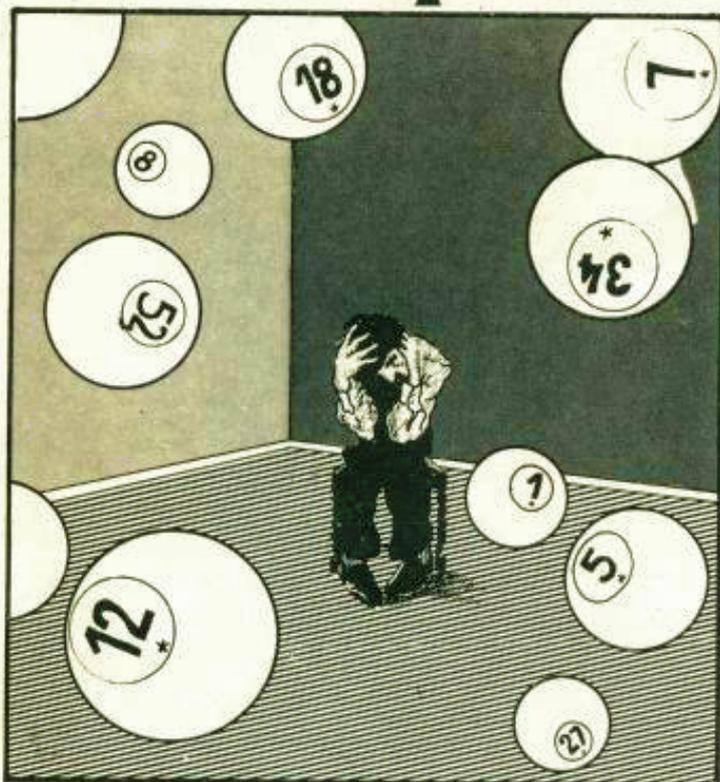
BARQUILLO, 25 - MADRID-4  
 Tfnos. 221 55 07 - 222 69 49  
 232 36 44 - 231 29 18



**HOJA DE PROMOCION**  
 Sirvase remitirnosla a COSESA  
 C/ Barquillo, 25 - Madrid-4

Por favor, envíeme más información sobre los ordenadores  
 .....  
 Nombre .....  
 Dirección .....  
 Localidad ..... D. P. ....  
 Provincia ..... Tfno. ....

# La multiplicación de los bingos



8

Un buen programa el que nos manda Joaquín desde Málaga, a pesar del reducido número de instrucciones. Cuando tenga línea pulse la "l" o "b" cuando consiga el bingo. Junto al listado, nos ha enviado un comentario de las principales líneas del programa, especialmente interesante para todos aquellos que busquen algo más que pasar un buen rato.

— Líneas 1 y 5: borra la pantalla y pone el puntero

**PROGRAMA GANADOR DE 5.000 PTAS**

Desde Málaga nos ha llegado este bingo para el ZX Spectrum. Su autor es José Joaquín Heredia, a quien felicitamos sinceramente y premiamos con 5.000 pesetas.

de números aleatorios en función del tiempo de encendido.

— Línea 7: Pone a 0 parte del *buffer* de la impresora.

— Línea 15: Asigna a la variable "e" el número de la extracción.

— Líneas 16 a 23: Saca en pantalla los números 1 al 90 tabulados.

— Líneas 24 a 35: Impresión rótulos en pantalla.

— Línea 40: Dibuja la bola.

— Líneas 42 a 45: Impresión número bola extraída.

— Líneas 50 y 51: Obtención al azar de un número entre 1 y 90.

— Líneas 52 a 55: Borra el número anterior e imprime el nuevo.

— Líneas 60 a 80: Coloca el fondo rojo en la parte superior de la pantalla y el número extraído, cambian-

do el color según la extracción.

Si usted leyó nuestro número anterior y tiene un Spectrum, seguro que ya ha elegido su bingo particular. Pero tenemos más, para aquellos que todavía no hayan hecho su elección. Y, como prometíamos, también hay para quienes poseen un ZX81.

En principio, elegir bingos para el Spectrum parecía una tarea sencilla. Puestos a la tarea ¡sólo tuvimos que probar las 66 cintas que nos remitieron los lectores! Por el contrario, para el ZX81 sólo recibimos seis cintas, por lo que decidimos esperar a tener más material que, no obstante, no llegó. Lo cual, por cierto, nos ha extrañado, habida

cuenta de la cantidad de lectores que demandan más programas para el ZX81. Esto viene a evidenciar, en todo caso, el enorme potencial que existe en nuestro mercado.

Con este número cerramos, pues, el capítulo dedicado a los bingos, sin perjuicio de que recibamos algún rezagado que, por su interés, merezca ser publicado. Y lo cerramos con cinco bingos, tres para el Spectrum y dos para el ZX81. Esperamos que sean de su agrado. Reiteramos nuestra sorpresa por la extraordinaria respuesta a nuestra convocatoria y el agradecimiento a los lectores que nos han enviado programas.

— Líneas 100 a 400: Detección de línea o bingo.

— Línea 1000: Vuelve a la línea 42. El programa principal va de la línea 42 a la 1000.

— Líneas 2000 y 2020:

Comprueba que el número extraído no esté repetido.

— Líneas 3000 a 3060: Verificación línea.

— Líneas 4000 a 4060: Verificación bingo.

Extracción: automática.

16K spectrum.

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90

NUMERO DE LA EXTRACCION

8



# PROGRAMAS

```

1 CLS
2 RANDOMIZE
3 FOR a=23300 TO 23400: POKE
a,0: NEXT a
15 LET x=11: LET e=0
16 PAPER 6
17 PRINT INK 6; PAPER 7; AT 0,0
;
18 PRINT AT 1,0;" 1 2 3 4
5 6 7 8 9 10 "
20 FOR a=2 TO 9: FOR b=0 TO 28
STEP 3: PRINT AT a,b;" ";x;" "
LET x=x+1: NEXT b: NEXT a
23 PRINT INK 7; PAPER 6; AT 10,
0;"
24 PAPER 7
30 PRINT AT 15,2;"NUMERO DE LA
35 PRINT AT 16,3;"EXTRACCION"
40 FOR a=0 TO 6 STEP 2: CIRCLE
INK 2,207,35,25-a: NEXT a
42 LET e=e+1
45 PRINT AT 18,7;e
50 LET c=INT (RAND*90)+1
51 GO TO 2000
52 PRINT AT 17,25;" "
53 PAUSE 10
55 PRINT AT 17,25;c
60 LET d=INT (c/10)
70 IF d*10=c THEN PRINT OVER 1
PAPER 2; INK 9; AT d,28;" " : G
O TO 100
80 LET f=c-d*10: PRINT OVER 1;
PAPER 2; INK 9; AT d+1,f*3-2;"

```

```

100 FOR a=1 TO 350
200 IF INKEY$="l" THEN GO TO 30
00
300 IF INKEY$="b" THEN GO TO 40
00
400 NEXT a
1000 GO TO 42
2000 IF PEEK (23300+c)=255 THEN
GO TO 50
2010 POKE 23300+c,255: GO TO 52
3000 PRINT PAPER 2; INK 6; BRIGH
T 1; AT 12,6;"SE HA CANTADO LINEA
"
3010 IF INKEY$("<") THEN BEEP .1,
0: BEEP .1,7: GO TO 3010
3020 IF INKEY$="" THEN BEEP .1,0
: BEEP .1,7: GO TO 3020
3030 LET a$=INKEY$
3040 IF a$="s" THEN GO TO 42
3050 IF a$="n" THEN PRINT AT 12,
60;" " : GO TO 4
2000
3060 GO TO 3010
4000 PRINT BRIGHT 1; PAPER 1; IN
K 5; FLASH 1; AT 21,10;"BINGO BIN
GO"
4010 IF INKEY$("<") THEN BEEP .1,
0: BEEP .05,7: BEEP .1,12: GO TO
4010
4020 IF INKEY$="" THEN BEEP .1,0
: BEEP .05,7: BEEP .1,12: GO TO
4020
4030 LET a$=INKEY$
4040 IF a$="s" THEN RUN
4050 IF a$="n" THEN PRINT AT 21,
10;" " : GO TO 42
4060 GO TO 4010

```

9

René Zaragueta ha pretendido realizar un programa muy completo y lo ha conseguido. Esquemáticamente, el programa permite varias cosas:

- Introducir porcentajes para banca, bingo y línea.
- Introducción del número de participantes y nombre de cada uno.
- Petición del precio del cartón y número de cartones que cada jugador compra.
- Impresión tabla con datos de jugadores, número de cartones comprados e importe.
- Al finalizar el juego,

se informa de las ganancias de cada jugador.

— Se posibilita echar nuevas partidas, informándose del saldo acumulado según haya sido su suerte.

Extracción: automática. 16K Spectrum.

**PROGRAMA GANADOR DE  
5.000 PTAS**

Como decimos en el texto, René Zaragueta, domiciliado en Barcelona, es el autor de este otro programa de bingo, que le ha valido el premio de 5.000 pesetas.

```

10 BORDER 7: PAPER 7: INK 0: I
NVERSE 0: BRIGHT 0: CLS : LET pa
rtidas=1: LET tcar=0:
20 GO SUB 160: GO SUB 340
30 BORDER 0: PAPER 5: INK 1: C
LS : INVERSE 1

```

```

40 GO SUB 1100
50 FOR n=19 TO 21: PRINT AT n,
14; PAPER 7; BRIGHT 1;" " : NÉ
XT n
60 PRINT #1; PAPER 2;" Cualq
uier tecla para PARAR "
70 PLOT 112,0: DRAW 0,24: DRAW
32,0: DRAW 0,-24
80 INK 0: PAPER 6: BRIGHT 1
90 DIM a(90): FOR n=1 TO 90: L
ET a(n)=0: NEXT n
100 LET y=(INT (RAND*90))+1
110 IF a(y)=1 THEN GO TO 100
120 BEEP .05,30: LET a(y)=1: PR
INT AT 20,15; PAPER 7; INK 2; BR
IGHT 1;y;" "
130 IF INKEY$("<") THEN GO SUB 1
120
140 GO SUB 850: GO SUB 1080
150 GO TO 100
160 REM Subrutina de partici-
pantes
170 LET a$=" ZX BINGO ZX BINGO
ZX BINGO ZX HECHO POR RENE ZARA
GUETA PARA ZX"
180 FOR n=1 TO LEN a$: PRINT PA
PER 6; INK 0; FLASH 1;a$(n);: BE
EP .05,30: NEXT n
190 PRINT AT 2,0;"Entre al repa
rto de premios en %""La BANCA
es para contribuir a los gasto
s del anfitrión, pero puede ser
0."

```

```

200 FOR n=8 TO 16: PRINT AT n,7
; PAPER 5; " : RE
M 16 espacios
210 NEXT n
220 PRINT AT 10,8; " BANCA:
% " AT 12,8; " BINGO: % " AT
14,8; " LINEA: % "
230 PRINT "La BANCA absorbe
las diferencias por redondeo."
240 INPUT "% para la BANCA? (de
0 a 10) " : tb
250 IF tb<0 OR tb>10 THEN PRINT
#1; PAPER 2; INK 7; FLASH 1; "Es
de 0 a 10 !! " : OR n=0 TO 500:
NEXT n: GO TO 240
260 PRINT AT 10,16; tb
270 INPUT "% para el BINGO? (de
0 a 80) " : tbi
280 IF tbi<0 OR tbi>80 THEN PR
INT #1; PAPER 2; INK 7; FLASH 1;
"Es de 0 a 80 !! " : FOR n=0 TO 5
00: NEXT n: GO TO 270
290 PRINT AT 12,16; tbi
300 LET tl=100-(tb+tbi): PRINT
AT 14,16; tl
310 PRINT #1; "Si quieres cambia
r datos pulsa c Si no, otra tecl
a para seguir."
320 PAUSE 0: IF INKEY$="c" OR I
NKEY$="C" THEN GO TO 220
330 RETURN
340 REM Participantes
350 INPUT "Numero de participan
tes?" : np
360 LET npmax=np+3: DIM n$(npma
x,7)
370 FOR n=1 TO np
380 PRINT AT 21,0; "Nombre del j
ugador n " : n; " ? " : INPUT n$(n)
390 NEXT n
400 GO SUB 1720
410 IF z$="s" THEN LET np=np+1:
PRINT AT 21,0; "Nombre del Jugad
or n " : np: INPUT n$(np): IF np>n
pmax THEN PRINT AT 21,0; "Ya no p
uede participar nadie mas": PAUS
E 0: GO TO 440
420 IF z$="s" THEN GO SUB 1720:
GO TO 410
440 DIM c(npmax): DIM i(npmax):
DIM g(npmax): DIM s(npmax): DIM
p(npmax): DIM q(npmax): DIM b(n
pmax): DIM l(npmax)
450 FOR n=1 TO npmax
460 LET c(n)=0: LET g(n)=0: LET
s(n)=0: LET q(n)=0: LET p(n)=0:
LET b(n)=0: LET l(n)=0: LET i(n
)=0: NEXT n
530 BORDER 2: PAPER 2: INK 7: C
L S
540 FOR n=0 TO 6
550 PRINT AT 15+n,0; PAPER 7; "
560 NEXT n
570 PRINT AT 5,7; "P A R T I D A
n " : partidas
580 PRINT AT 17,7; INK 0; PAPER
7; "Precio del carton?": INPUT c
ar
590 PRINT AT 7,6; "PRECIO DEL CA
RTON: " : car
600 PRINT AT 17,4; PAPER 7; INK
0; "Cuantos cartones compra?"
610 FOR n=1 TO np
620 PRINT AT 19,11; PAPER 7; IN
K 0; n$(n)
630 INPUT cj: LET c(n)=c(n)+cj
635 FOR m=1 TO cj: PRINT AT 20,
0; FOR v=1 TO 90 STEP 6: PRINT
INT (RND*10)+v+1; " " : NEXT v: P

```

```

AUSE 0: PRINT AT 20,0; INVERSE 1
;
; NEXT m
640 LET tcar=tcar+car*cj
650 NEXT n
660 PRINT AT 17,4; PAPER 7; "
670 PRINT AT 15,7; BRIGHT 1; IN
K 0; PAPER 7; "Recaudacion=" : tcar
680 LET lin=INT (tcar*(100-(tbi
+tb)/100)
690 PRINT AT 19,10; PAPER 7; IN
K 0; BRIGHT 1; "LINEA, A: " : lin
700 LET bin=INT (tcar*tbi/100)
710 PRINT AT 21,10; PAPER 7; IN
K 0; BRIGHT 1; "BINGO, A: " : bin
720 LET ban=tcar-lin-bin
730 PRINT AT 17,10; PAPER 7; IN
K 0; "BANCA... " : ban
740 FOR n=1 TO np: LET i(n)=c(n
)*car: NEXT n
750 GO SUB 1760
760 REM Tabla 2
770 BORDER 1: PAPER 1: INK 7: C
L S
780 PRINT AT 0,0; "n Nombre
Cart (*) Importe(*) (*)Acumulado"
790 PRINT " " : PRINT
800 FOR n=1 TO np
810 PRINT AT 3+n,0; n; AT 3+n,4; n
$(n); AT 3+n,16; c(n); AT 3+n,25; i
(n)
820 NEXT n
830 PRINT #1; "Cualquier tecla p
ara empezar la PARTI
DA": PAUSE 0
840 RETURN
850 LET l=0: LET c=0
860 IF y<=10 AND y>0 THEN LET l
=1
870 IF y<=20 AND y>10 THEN LET
l=3
880 IF y<=30 AND y>20 THEN LET
l=5
890 IF y<=40 AND y>30 THEN LET
l=7
900 IF y<=50 AND y>40 THEN LET
l=9
910 IF y<=60 AND y>50 THEN LET
l=11
920 IF y<=70 AND y>60 THEN LET
l=13
930 IF y<=80 AND y>70 THEN LET
l=15
940 IF y>80 THEN LET l=17
950 LET w=y-((INT (y/10))*10)
960 IF w=1 THEN LET c=1
970 IF w=2 THEN LET c=4
980 IF w=3 THEN LET c=7
990 IF w=4 THEN LET c=10
1000 IF w=5 THEN LET c=13
1010 IF w=6 THEN LET c=16
1020 IF w=7 THEN LET c=19
1030 IF w=8 THEN LET c=22
1040 IF w=9 THEN LET c=25
1050 IF w=0 THEN LET c=28
1060 PRINT AT l,c,y; (" " AND y<1
0)
1070 RETURN
1080 FOR r=0 TO 50: BEEP .01,r:
IF INKEY$<>" " THEN GO SUB 1120
1090 NEXT r: RETURN
1100 REM tablero
1110 FOR y=1 TO 90: INK 1: PAPER
3: GO SUB 850: NEXT y: INVERSE
0: RETURN
1120 REM parar
1130 FOR n=0 TO 50: NEXT n
1140 PRINT AT 19,0; PAPER 5; BRI
GHT 0; "c = Continuar" : l = Lin
ea
"b = Bingo "

```

## PROGRAMAS

```

1150 IF INKEY$="c" THEN FOR n=19
TO 21: PRINT AT n,0; PAPER 5; B
RIGHT 0; " " : NEXT n
: RETURN
1160 IF INKEY$="l" THEN LET comp
=5: LET prem=1: GO TO 1190
1170 IF INKEY$="b" THEN LET comp
=15: LET prem=2: GO TO 1190
1180 GO TO 1150
1190 FOR n=19 TO 21: PRINT AT n,
0; PAPER 5; BRIGHT 0; " "
: NEXT n
1200 PRINT AT 19,0; PAPER 5; BRI
GHT 0; "Entre los nu- " "meros a
com- " "probar:"
1210 DIM e(comp): FOR n=1 TO com
p: LET e(n)=0: NEXT n: FOR h=1 T
O comp
1220 INPUT nu: IF a(nu)<>1 THEN
GO SUB 1280: GO TO 100
1230 LET a(nu)=2
1240 LET e(h)=nu
1250 FLASH 1: LET y=nu: GO SUB 8
50: FLASH 0
1260 NEXT h
1270 GO SUB 1360
1280 REM Numero falso
1290 FOR n=19 TO 21: PRINT AT n,
0; PAPER 5; BRIGHT 0; " "
: NEXT n
1300 IF h-1=comp AND prem=1 THEN
PRINT AT 19,0; PAPER 5; "FELICID
ADES" " LINEA " : INPUT "Juga
dor n°: "; jn: LET l(jn)=l(jn)+1:
GO SUB 1420: RETURN
1310 IF h-1=comp AND prem=2 THEN
PRINT AT 20,0; PAPER 6; FLASH 1

```

```

: " FELICIDADES " " " BINGO
: " INPUT "Jugador n°: "; jn: LET
b(jn)=b(jn)+1: GO SUB 1420: RETU
RN
1320 PRINT AT 19,0; FLASH 1; " " "
FALSO!! " " " " " "
c="Continuar": PAUSE 0
1330 GO SUB 1360
1340 IF INKEY$="c" THEN PRINT AT
19,0; PAPER 5; BRIGHT 0; " " "
" " : FOR n=0 TO 100: NEX
T n: GO TO 100
1350 GO TO 1340
1360 REM Num. comprobados
1370 FOR r=1 TO h-1
1380 LET a(n)=1
1390 LET y=e(n): GO SUB 850
1400 NEXT n
1410 RETURN
1420 REM Sumar premio
1430 IF prem=2 THEN LET s(jn)=-i
(jn)+bin: PRINT AT 18,20; PAPER
5; BRIGHT 0; n$(jn); AT 19,20; "HA
GANADO"; AT 20,24; bin; AT 21,23; "P
tas.": LET g(jn)=g(jn)+bin: GO T
O 1470
1440 IF prem=1 THEN LET s(jn)=-i
(jn)+lin: PRINT AT 18,20; PAPER
5; BRIGHT 0; n$(jn); AT 19,20; "HA
GANADO"; AT 20,24; lin; AT 21,23; "P
tas.": LET g(jn)=g(jn)+lin
1450 RETURN
1460 LET partidas=partidas+1: GO
TO 30
1470 REM final partida, tablas, em
pezar otra partida

```

# SAQUELE JUGO A SU SPECTRUM

BOALOX INFORMATICA tiene programas que le resolverán problemas en su trabajo. O que le permitirán aprender en forma entretenida.

### PROGRAMAS EN EXISTENCIA (todos para Spectrum 48K)

**"CUERPO".** Explica el funcionamiento de los principales órganos y sistemas del cuerpo humano, mediante gráficos con color, movimiento y sonido. Menú con 17 posibilidades; 55 dibujos, con movimiento, y 51 textos explicativos. Diccionario médico con 90 términos.

**"DIGESTIVO".** Vea el sistema digestivo humano en funcionamiento; gráficos con color y movimiento.

**"TEST".** Paquete de programas para docentes, que permite construir sin esfuerzo pruebas objetivas de distintos tipos (pruebas de verdadero/falso, y pruebas de elección múltiple con 3, 4 y 5 respuestas posibles). dos de los pro-

gramas contienen pruebas de vocabulario de francés y de inglés —con vocabulario incluido, sustituible por otro—; y el resto de ellos utilizables para exámenes sobre cualquier materia. Todos con posibilidad de aplicación individual y colectiva, con instrucciones para elaborar plantillas de corrección rápida. Con salida para impresora ZX.

**"TUTOR".** Muy útil para iniciarse en programación. Facilita el estudio de las palabras inglesas que se usan en el BASIC del Spectrum. Contiene un test de elección múltiple para hacer más ameno el aprendizaje, y dos diccionarios.

**"TUTOR 1".** Diviértase mientras aprende vocabulario inglés. Sobre un vocabulario de 600 palabras inglesas de las más usuales, contiene un test de

elección múltiple, un test para aprender a escribir correctamente las palabras inglesas, dos diccionarios y pronunciación figurada. Ideal para niños y adolescentes. Tan "adictivo" como un juego.

**"TUTOR F".** Características similares a "TUTOR 1", pero sobre vocabulario francés.

**"ESTADISTICAS".** Paquete de programas de estadísticas de dos variables. Calcula los principales estadísticos, correlación y recta de regresión. Hace escalas de intervalos; traza histogramas y nube de puntos. Conjunto de tres programas que permite procesar gran cantidad de datos. Muy útil para Psicólogos, Economistas, Ingenieros, Médicos, estudiantes, etc.

### PRECIOS

- Cassette con "DIGESTIVO" y "CUERPO" ..... 2.000 Ptas.
- Paquete de programas "TEST" ..... 3.500 "
- Cassette con "TUTOR" y "TUTOR 1" ..... 2.000 "
- Cassette con "TUTOR" y "TUTOR F" ..... 2.000 Ptas.
- Paquete de Programas de "ESTADISTICA" ..... 3.000 "

Ventas en su proveedor habitual o contra reembolso (en el precio se incluirán 300 Ptas. de gastos de envío).



**BOALOX**

INFORMATICA C/Progreso, 98 - ORENSE Telf. (988) 22 16 47

# PROGRAMAS

```

1480 GO SUB 1760
1490 FOR n=1 TO np: LET q(n)=q(n)
+c(n): NEXT n: BORDER 7: PAPER
7: INK 0: BRIGHT 0: INVERSE 0: C
LS
1500 PRINT AT 0,0; PAPER 2; INK
7: " Resumen: no partidas Jugadas
...partidas
1510 PRINT "Nombre: Part Cart
Lineas Bingos"
1520 PRINT INK 2; "
1530 FOR n=1 TO np: LET p(n)=p(n)
+1*(c(n)<>0): PRINT TAB 0;n$(n)
1540 PRINT TAB 11;p(n);
1550 PRINT TAB 15;q(n);
1560 PRINT TAB 22;l(n);TAB 29;b(
n);
1570 PRINT : PRINT : NEXT n
1580 GO SUB 1760
1590 CLS : PRINT "Nombre: Gan
a: Debe: Saldo: "
1600 FOR n=1 TO np
1610 LET col=7-5*(s(n)<0): LET c
olt=0+7*(s(n)<0)
1620 PRINT TAB 0;n$(n);TAB 12;g
(n);TAB 18;i(n);

```

```

1630 PRINT TAB 26; PAPER col; IN
K colt; ("-" AND s(n)<0);TAB 27;A
BS s(n)
1640 NEXT n
1650 GO SUB 1720: IF INKEY$="n"
THEN GO TO 1690
1660 LET np=np+1: IF np>npmax TH
EN PRINT "No puedo aceptar mas j
ugadores"
1670 LET l(np)=0: LET i(np)=0: L
ET b(np)=0: LET p(np)=0: LET s(n
p)=0: LET g(np)=0: LET q(np)=0
1680 PRINT AT 21,0;"Nombre del j
ugador no:";np; "?": INPUT n$(np):
GO TO 1650
1690 GO SUB 160: CLS : GO SUB 47
0: GO TO 30
1700 LET partidas=partidas+1
1710 STOP
1720 PRINT AT 21,0;"Se incorpora
alguien mas? (s/n)"
1730 FOR n=0 TO 50: NEXT n: PAUS
E 0: LET z$=INKEY$
1740 CLS
1750 RETURN
1760 PRINT #1;"Cualquier tecla p
ara seguir": PAUSE 0: RETURN
1770 SAVE "ZX BINGO" LINE 10

```

BINGO ZX BINGO ZX BINGO ZX  
HECHO POR RENE ZARAGUETA PARA ZX  
Entre el reparto de premios en %  
La 'BANCA' es para contribuir a  
los gastos del anfitrión, pero  
puede ser 0.

BANCA: %  
BINGO: %  
LINEA: %

La BANCA absorbe las diferencias  
por redondeo.

PRECIO DEL CARTON: 250

Cuantos cartones compra?

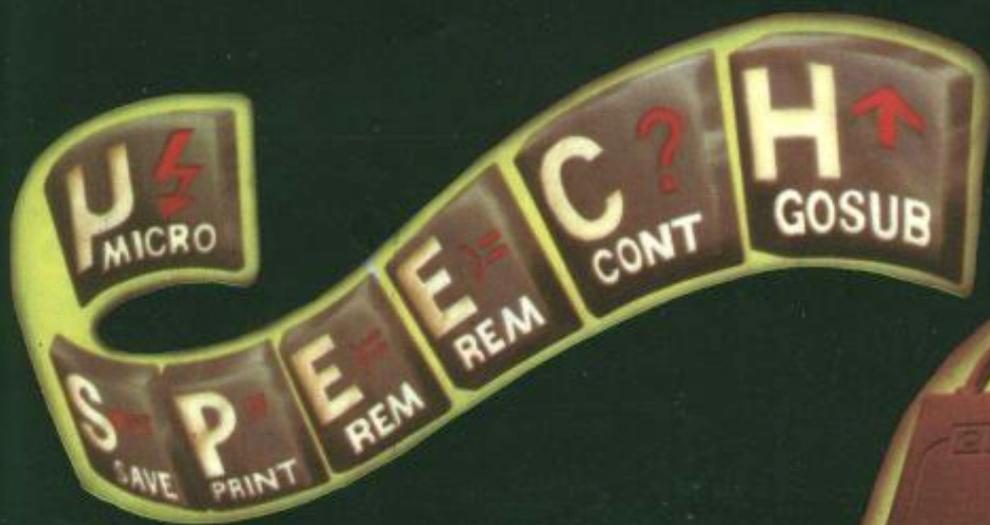
ricardo  
11, 11, 22, 21, 26, 40, 38, 50, 52, 65, 64,  
72, 82, 83, 90,

n <sup>o</sup>	Nombre	Cart(*)	Importe(*)
1	ricardo	2	500
2	Juanito	1	250

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90

# COMPUTER AUTO DATA RECORDER C. 108

Cassette  
especial para  
ordenadores.



CURRAH

μSPEECH

Ahora puedes  
hacer hablar  
tu Spectrum.



EN LAS BUENAS TIENDAS DE INFORMATICA



**CECOMSA**

Castelló, 25 - 3.º E - Madrid-1 - Telef.: 435 37 01

Fabrica periféricos para Spectrum ZX81 y VIC 20

## PROGRAMAS

10

Fernando García nos ofrece un programa de bingo con números gigantes. Pero lo suyo le ha costado, como puede verse en las líneas 5133 a 5142. Es un buen programa, aunque utilizando la rutina assembler para generación de números de mayor tamaño, se habría ahorrado bastantes instrucciones. El programa genera 4 cartones al azar, detectando si hay línea o bingo y poniéndolo en pantalla. Esto es otra importante novedad, aunque lógicamente ne-

cesita más tiempo de comprobación cuanto mayor es el número de cartones.

Extracción: automática.  
16K Spectrum.

### PROGRAMA GANADOR DE 5.000 PTAS

Fernando García Romero, de Madrid, se ha hecho acreedor a nuestro premio de 5.000 pesetas como autor de este programa de bingo, que hemos seleccionado.

```

1 FOR a=0 TO 7: READ x: POKE
USR "a"+a,x: NEXT a
2 DATA 1,3,7,15,31,63,127,255
3 FOR a=0 TO 7: READ x: POKE
USR "b"+a,x: NEXT a
4 DATA 128,192,224,240,248,25
2,254,255
5 FOR a=0 TO 7: READ x: POKE
USR "c"+a,x: NEXT a
6 DATA 255,127,63,31,15,7,3,1
7 FOR a=0 TO 7: READ x: POKE
USR "d"+a,x: NEXT a
8 DATA 255,254,252,248,240,22
4,192,128
9 FOR a=0 TO 7: READ x: POKE
USR "e"+a,x: NEXT a
10 DATA 255,127,63,31,31,63,12
7,255
11 FOR a=0 TO 7: READ x: POKE
USR "f"+a,x: NEXT a
12 DATA 255,254,252,248,248,25
2,254,255
15 DIM t(90): DIM m(15): DIM b
(4): DIM l(4): DIM c(4,15): DIM
d(30,15)
20 LET linea=0: LET bingo=0
30 FOR a=1 TO 15: READ m(a): N
EXT a
40 DATA 0,3,6,9,12,0,3,6,9,12,
0,3,6,9,12
50 FOR q=1 TO 90: LET t(q)=0:
NEXT q
60 PRINT "Despues de cada nume
ro, se pon- dran dos cuadros de
la parte in- ferior intermitentes
, para conti- nuar pulse cualquier
tecla." : INK 2: "La(p) sirve para
parar."
70 PRINT "Si quiere jugar con
cartones su- yos, pulse (P)."
80 INPUT d$: IF d$="P" OR d$="
P" THEN GO TO 5000
90 INPUT "Numero de cartones (2
-4)",nc
95 IF nc<2 OR nc>4 THEN GO TO
90
100 CLS : PRINT AT 0,0: INK 1;"
** TABLERO **
110 FOR a=1 TO 9
120 PRINT AT a,0: INK 1;"■"; AT
a,31: INK 1;"■"
130 NEXT a

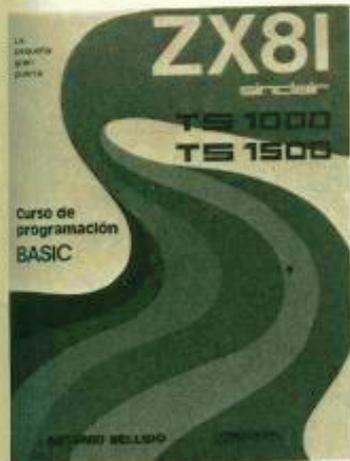
```

```

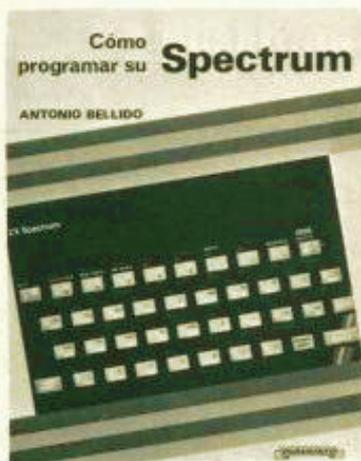
140 LET fi=10: GO SUB 3000
150 LET fi=11: GO SUB 4000
160 LET fi=14: GO SUB 3000
170 LET fi=16: GO SUB 3000
180 LET fi=17: GO SUB 4000
190 LET fi=20: GO SUB 3000
200 PRINT AT 15,1;"Carton 1
Carton 2"; AT 21,1;"Carton 3
Carton 4"
250 REM Entrada de los numeros
en los cartones
260 FOR a=1 TO nc: FOR b=1 TO 1
5
270 LET va=INT (90*RND)+1
280 FOR c=b-1 TO 1 STEP -1
290 IF va=c(a,c) THEN GO TO 270
300 NEXT c
310 LET c(a,b)=va
320 NEXT b: NEXT a
350 REM Escritura de los carton
es
360 FOR a=1 TO nc: FOR b=1 TO 1
5
370 GO SUB 2000
380 IF c(a,b)<10 THEN GO TO 400
390 PRINT AT fila,colu;c(a,b):
GO TO 410
400 PRINT AT fila,colu;" ";c(a,
b)
410 NEXT b: NEXT a
420 GO SUB 500: GO TO 600
500 REM Comienzo
510 LET nu=INT (90*RND)+1: IF t
(nu)=1 THEN GO TO 510
520 LET t(nu)=1: LET filan=1+IN
T (nu/10,1)
530 LET colun=1+3*(nu-INT (nu/1
0)*10-1)
540 IF colun<0 THEN LET colun=2
9
580 PRINT AT filan,colun: FLASH
1: PAPER 6;" ";nu
590 RETURN
600 REM Comprobacion de los car
tones
610 FOR a=1 TO nc
620 FOR b=1 TO 15
630 IF c(a,b)<>nu THEN GO TO 68
0
640 LET c(a,b)=0: GO SUB 2000
650 LET memoria=22528+fila*32+c
olu
660 POKE memoria,32: POKE memor
ia+1,32
670 GO TO 690
680 NEXT b
690 NEXT a
700 REM lineas
710 IF linea=1 THEN GO TO 900
720 FOR a=1 TO nc
730 FOR i=0 TO 10 STEP 5
740 LET tot=0
750 FOR b=1+i TO 5+i
760 LET tot=tot+c(a,b): NEXT b
770 IF tot=0 THEN LET l(a)=1: G
O TO 790
780 NEXT i
790 NEXT a
800 LET tot=0
810 FOR a=1 TO nc: LET tot=tot+
l(a): NEXT a
820 IF tot<>0 THEN LET linea=1
830 REM Detectar si hay linea
840 IF linea=0 THEN GO TO 1200
850 FOR a=1 TO nc
870 IF l(a)=1 THEN GO SUB 2500:

```

# Su BIBLIOTECA de INFORMATICA



**ZX81. Curso de programación BASIC.** Bellido. 128 páginas. 2ª edic. 1983. 850,- Ptas.  
**INDICE EXTRACTADO:** Introducción general. Manejando el ZX81. Programando en BASIC. Practicando el BASIC con el ZX81.



**Cómo programar su SPECTRUM.** Bellido. 132 páginas. 2ª edición. 1984. 850,- Ptas.  
**INDICE EXTRACTADO:** Cómo utilizar el Spectrum. Cómo programar el Spectrum. Introducción para principiantes. Introducciones específicas. Expresiones y operadores lógicos.



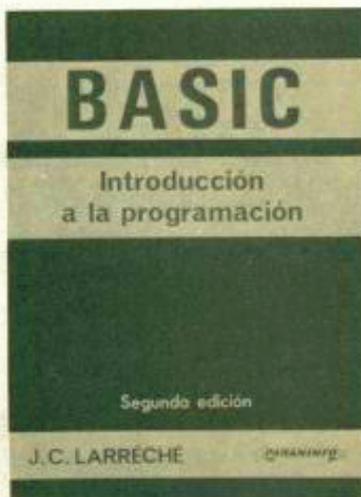
**Cómo usar los colores y los gráficos en el SPECTRUM.** Bellido. 96 páginas. 1984. 850,- Ptas.  
**INDICE EXTRACTADO:** Caracteres gráficos. Colores. La casualidad. El movimiento. La resolución de gráficas. Procesos internos. PEEK y POKE. Memoria. Pantalla. Gráficos profesionales.



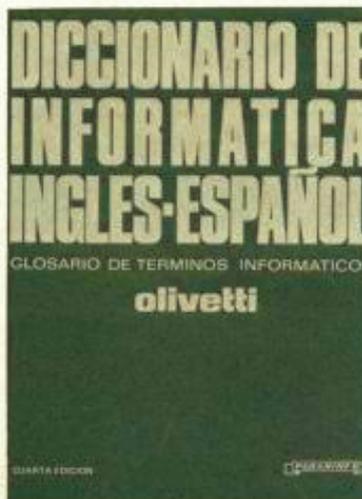
**BASIC. Curso acelerado.** De Rossi. 224 páginas. 1984. 850,- Ptas.  
**INDICE EXTRACTADO:** Uso eficiente del sistema. Las sentencias LET, IF, PRINT, GOTO y END, READ y DATA. Resolución de problemas usando siete tipos de sentencias. Diagrama de flujo. Aritmética. Comandos del sistema. Sentencias FOR/NEXT. Funciones INT y RND. Sentencias INPUT y RESTORE, GOSUB, RETURN y ON. Matrices y subíndices. Tablas o vectores. Sentencias alfanuméricas. Funciones definidas. Cintas de papel. Ejercicios.



**BASIC. Programación de microordenadores.** Checroun. 112 páginas. 2ª edición. 1984. 500,- Ptas.  
**INDICE EXTRACTADO:** Los microordenadores. El lenguaje BASIC. Extensiones del BASIC. Los ficheros. Problemas de aplicación. Recapitulación de las instrucciones de BASIC. Las funciones incorporadas o de bibliotecas.



**BASIC. Introducción a la Programación.** Larreché. 132 páginas. 2ª edic. 1984. 550,- Ptas.  
**INDICE EXTRACTADO:** Introducción a la programación. Definición del lenguaje BASIC. Realización de un programa en BASIC. Programas útiles. Extensión del BASIC. Lenguaje de comandos de los sistemas HP 2000 B, C y F.



**Diccionario de informática. Inglés-español.** Olivetti. 272 páginas. 4ª edición. 1984. 750,- Ptas.  
 Facilita la traducción literal y el concepto científico y técnico de cada vocablo. Glosario de términos.



**Programación con el lenguaje COBOL.** Galán Pascual. 328 páginas. 1983. 975,- Ptas.  
**INDICE EXTRACTADO:** Conceptos previos sobre informática básica. Introducción. "Identification division", "Environment division", "Data division", "Procedure division". Instrucciones de entrada/salida. Instrucciones de manipulación de datos, de bifurcación, de cálculo, condicionales y de control. Creación y manejo de tablas. Generación automática de informes. Clasificación de ficheros. Su organización en dispositivos de acceso directo. Sugerencias de orden práctico. Apéndices.

## COPIE O RECORTE ESTE BOLETIN DE PEDIDO

Deseo recibir a reembolso los siguientes libros:

.....  
 .....  
 .....

Enviar a  
 Editorial PARANINFO, S.A.  
 Magallanes, 25 - MADRID-15

Nombre .....

Domicilio .....

Ciudad .....

```

GO SUB 2550
880 NEXT a
890 GO TO 1200
900 REM Bingo
910 FOR a=1 TO nc
920 LET tot=0
930 FOR b=1 TO 15: LET tot=tot+
c(a,b): NEXT b
940 IF tot=0 THEN LET b(a)=1
950 NEXT a
960 LET tot=0
970 FOR a=1 TO nc
980 LET tot=tot+b(a): NEXT a
990 IF tot=0 THEN GO TO 1200
1000 FOR a=1 TO nc
1010 IF b(a)=1 THEN GO SUB 2500:
GO SUB 2500
1020 NEXT a
1030 GO SUB 1500
1040 CLS
1050 FOR a=1 TO nc
1060 IF b(a)=0 THEN GO TO 1030
1070 PRINT "Ha obtenido bingo el
carton: "; a
1080 NEXT a
1090 PRINT AT 12,3; FLASH 1;"Qui
ere jugar otra vez(s,n)"
1100 GO SUB 1500: IF INKEY$<>"s"
AND INKEY$<>"n" THEN STOP
1110 CLS: CLEAR: RUN
1200 GO SUB 1500
1205 LET memor=22528+filan*32+co
lun
1208 POKE memor,48: POKE memor+1
,48
1210 GO TO 420
1500 PRINT AT 21,15; FLASH 1;"
1510 IF INKEY$="" THEN GO TO 151
0
1515 IF INKEY$="p" OR INKEY$="P"
THEN STOP
1520 PRINT AT 21,15;" ": RETURN
1600 GO SUB 1500
1610 FOR Q=1 TO 7: PRINT AT 12+Q
,8;" ": NEXT Q
1620 RETURN
2000 LET fil=11: LET col=1
2010 IF a=2 THEN LET col=17: GO
TO 2050
2020 IF a=3 THEN LET fil=17: GO
TO 2050
2030 IF a<>4 THEN GO TO 2050
2040 LET fil=17: LET col=17
2050 LET fila=fil+INT (b/5.5)
2060 LET colu=col+m(b)
2070 RETURN
2500 LET f=15: LET c=10
2510 IF a=2 THEN LET c=26: GO TO
2545
2520 IF a=3 THEN LET f=21: GO TO
2545
2530 IF a<>4 THEN GO TO 2545
2540 LET f=21: LET c=26
2545 RETURN
2550 PRINT AT f,c; FLASH 1;"LINE
": GO SUB 1500
2560 PRINT AT f,c;" ": RETUR
N
2600 PRINT AT f,c; FLASH 1;"BING
O": GO SUB 1500: RETURN
3000 PRINT AT fi,0; INK 5;"
TURN
4000 FOR i=0 TO 2

```

```

4010 PRINT AT fi+i,0; INK 5;"
4020 PRINT AT fi+i,15; INK 5;"
4030 PRINT AT fi+i,31; INK 5;"
4040 NEXT i
4050 RETURN
5000 PRINT "Si desea que la maqu
ina detecte las lineas y bingos
pulse(L).
5010 INPUT es: IF es<>"L" AND es
<>"l" THEN GO TO 5030
5020 INPUT "numero de cartones(2
-30)",nc
5030 IF nc<2 OR nc>30 THEN GO TO
5020
5040 PRINT "Introduzca los numer
os por line-as."
5050 FOR a=1 TO nc: PRINT "Carto
n "; a
5060 FOR b=1 TO 15
5062 INPUT d(a,b)
5063 IF d(a,b)<1 OR d(a,b)>90 TH
EN GO TO 5062
5065 PRINT "/";d(a,b);: NEXT b
5070 PRINT: NEXT a
5080 CLS: PRINT AT 0,0; INK 1;"
** TABLERO **
5090 FOR a=1 TO 9
5100 PRINT AT a,0; INK 1;" ";AT
a,31; INK 1;" ";: NEXT a
5110 PRINT AT 10,0; INK 1;"
5120 GO SUB 500
5122 LET d=INT (nu/10)
5124 LET u=nu-d*10
5126 FOR i=0 TO 1
5128 IF d=0 AND i=0 THEN GO TO 5
150
5130 LET x=10+i*7
5132 GO TO 5133+d
5133 PRINT AT 13,x;" " : AT 14
,x;" " : AT 15,x;" " : AT 16
,x;" " : AT 17,x;" " : AT 1
8,x;" " : AT 19,x;" " : GO
TO 5150
5134 PRINT AT 13,x;" 0 " : AT 14
,x;" " : AT 15,x;" " : AT 16
,x;" " : AT 17,x;" " : AT 1
8,x;" " : AT 19,x;" " : GO
TO 5150
5135 PRINT AT 13,x;" 1 " : AT 14
,x;" " : AT 15,x;" " : AT 16
,x;" " : AT 17,x;" " : AT 1
8,x;" " : AT 19,x;" " : GO
TO 5150
5136 PRINT AT 13,x;" 2 " : AT 14
,x;" " : AT 15,x;" " : AT 16
,x;" " : AT 17,x;" " : AT 1
8,x;" " : AT 19,x;" " : GO
TO 5150
5137 PRINT AT 13,x;" 3 " : AT 14
,x;" " : AT 15,x;" " : AT 16
,x;" " : AT 17,x;" " : AT 1
8,x;" " : AT 19,x;" " : GO
TO 5150
5138 PRINT AT 13,x;" 4 " : AT 14
,x;" " : AT 15,x;" " : AT 16
,x;" " : AT 17,x;" " : AT 1
8,x;" " : AT 19,x;" " : GO
TO 5150
5139 PRINT AT 13,x;" 5 " : AT 14
,x;" " : AT 15,x;" " : AT 16
,x;" " : AT 17,x;" " : AT 1
8,x;" " : AT 19,x;" " : GO
TO 5150

```

# PROGRAMAS

```

5140 PRINT AT 13,x;" 7 ";AT 14
,x;" ";AT 15,x;" ";AT 16
,x;" ";AT 17,x;" ";AT 1
0,x;" ";AT 19,x;" ";GO
TO 5150
5141 PRINT AT 13,x;" 4 ";AT 14
,x;" ";AT 15,x;" ";AT 16
,x;" ";AT 17,x;" ";AT 1
0,x;" ";AT 19,x;" ";GO
TO 5150
5142 PRINT AT 13,x;" 9 ";AT 14
,x;" ";AT 15,x;" ";AT 16
,x;" ";AT 17,x;" ";AT 1
0,x;" ";AT 19,x;" ";GO
TO 5150
5150 LET d=u
5160 NEXT i
5200 IF e$="L" OR e$="L" THEN GO
TO 5220
5210 GO SUB 1600: LET memor=2252
0+filan*32+colun: POKE memor,48:
POKE memor+1,48: GO TO 5120
5220 FOR a=1 TO nc: FOR b=1 TO 1
5230 IF d(a,b)=nu THEN LET d(a,b
)=0
5240 NEXT b: NEXT a
5250 IF linea=1 THEN GO TO 5360
5260 FOR a=1 TO nc
5270 FOR i=0 TO 10 STEP 5

```

```

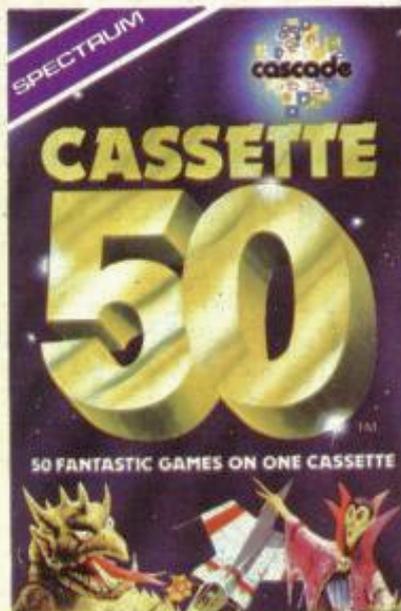
5320 LET tot=0
5330 FOR b=1+i TO 5+i
5340 LET tot=tot+d(a,b): NEXT b
5310 IF tot<>0 THEN GO TO 5320
5312 PRINT AT 21,0;"Ha obtenido
linea el carton ";a
5313 FOR g=1 TO 100: NEXT g
5316 PRINT AT 21,0;" ": LET linea
=1: GO TO 5330
5320 NEXT i
5330 NEXT a
5340 GO TO 5430
5360 FOR a=1 TO nc
5370 LET tot=0
5380 FOR b=1 TO 15
5390 LET tot=tot+d(a,b): NEXT b
5400 IF tot<>0 THEN GO TO 5410
5402 PRINT AT 21,0;"Ha obtenido
bingo el carton ";a
5403 FOR g=1 TO 100: NEXT g
5406 PRINT AT 21,0;" ": LET bingo
=1
5410 NEXT a
5420 IF bingo=1 THEN CLS: GO TO
1090
5430 GO SUB 1600: LET memor=2252
0+filan*32+colun: POKE memor,48:
POKE memor+1,48: GO TO 5120

```

**cascade**

**cascade**

**cascade**



Dirección:  
C/ CRISTOBAL BORDIU, 35  
MADRID-3  
TFNO. 254 24 30

**LA CINTA  
DEL AÑO**

N.º Y NOMBRE DEL JUEGO

- |  |                                    |  |
|--|------------------------------------|--|
| 1. MUNCHER (MASTICADOR)                  | 17. RACE TRACK (PISTA DE CARRERAS) | 35. FIELD (EL CAMPO)                       |
| 2. SKI JUMP (SALTO DE SKI)               | 18. SKI RUN (SLALON)               | 36. DRAGOLE (DRAGON DE ORO)                |
| 3. BASKETBALL (BALONCESTO)               | 19. TANKS (TANQUES)                | 37. SPACE SEARCH (BUSQUEDA EN EL ESPACIO)  |
| 4. FROGGER (RANAS)                       | 20. SOLAR SHIP (NAVE SOLAR)        | 38. INFERNO (INFERNO)                      |
| 5. BREAKOUT (ESCAPE)                     | 21. TEN PINS (10 ALFILERES)        | 39. NIM (NIM)                              |
| 6. CRUSHER (APLASTADOR)                  | 22. CARS (COCHES)                  | 40. VOYAGER (VIAJANTE)                     |
| 7. STARTRK (VAJE A LAS ESTRELLAS)        | 23. STUMPER (BEISBOL)              | 41. SKETCH PAD (CUADERNO DE ESBOZOS)       |
| 8. MARTIAN KNOCK OUT (KAO MARCHANO)      | 24. PINBALL (FLIPPER)              | 42. BUTZ (ALARMA ANTIAREA)                 |
| 9. BOOGLES (DESORBITADO)                 | 25. CAVERN (CUEVA)                 | 43. FISHING MISSION (MISION DE PESCA)      |
| 10. ALIEN ATTACK (ATAQUE EXTRATERRESTRE) | 26. LASER (LASER)                  | 44. MYSTICAL DIAMONDS (DIAMANTES MISTICOS) |
| 11. LUNAR LAMER (ATERIZAJE LUNAR)        | 27. ALIEN (EXTRATERRESTRE)         | 45. GALAXY DEFENSE (DEFENSA DE LA GALAXIA) |
| 12. MAZE EATER (COMEDOR DE LABERINTO)    | 28. CARD (CARGA)                   | 46. CYBER (CIFRADO)                        |
| 13. MIGHTYTRAP (ESQUEJA TRAMPA)          | 29. THE RACE (CARRERA)             | 47. JETMOBILE (COCHE JET)                  |
| 14. MOTORWAY (AUTOPISTA)                 | 30. THE SKILL (EL CRANEJO)         | 48. BARREL JUMP (SALTAR EL BARRIL)         |
| 15. LABYRINTH (LABERINTO)                | 31. SHIT (ORBITA)                  | 49. ATTACKER (ATACADOR)                    |
| 16. SKITTLES (JUEGO DE BOLOS)            | 32. MUNCH (MUNCH)                  | 50. SPACE MISSION (MISION ESPACIAL)        |
|  | 33. BOWLS (BOLOS)                  |  |
|  | 34. RAIDERS (SAQUEADORES)          |  |

Remita este cupón a Cascade, C/ Cristóbal Bordiu, 35, Madrid-3

NOMBRE .....

APELLIDOS .....

DIRECCION .....

POBLACION .....

PROVINCIA ..... D.P. ....

TELEFONO .....

FORMA DE PAGO:  TÁLON CONFORMADO  GIRO POSTAL  CONTRA REEMBOLSO



VEN A LA TIENDA  
Nº1 DE MADRID  
**REMSHOP-1**  
INAGURACION 3 DE MAYO 1984

# REM

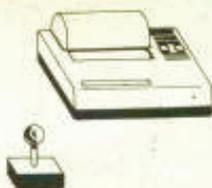
- REM** Somos profesionales.
- REM** Nace para dar mejor servicio.
- REM** Como oferta de lanzamiento, entrega un Bono del 10% del importe de su compra para adquirir nuevo material o programas.
- REM** CAMBIO acepta equipos de 2ª mano al adquirir otro nuevo
- REM** Consultanos tus necesidades.

**RENOVACION EN MARCHA, S.A.**  
c/. Espronceda, 34-2º int. - MADRID-3  
Teléfono (91) 441 24 78

**REM SHOP 1**  
c/. Galileo, 4 - MADRID-15  
Teléfono (91) 445 28 08

## HARD SPECTRUM JUEGOS

1 Spectrum 16 K .....	39.900 Ptas.
1 Ampliación a 48 .....	9.500 Ptas.
1 Interface programable con joystick .....	17.500 Ptas.
1 Impresora Alphacom 32 .....	22.500 Ptas.
Interface Microdrive .....	19.500 Ptas.
1 Microdrive .....	19.500 Ptas.
<b>TOTAL</b> .....	<b>128.400 Ptas.</b>



**PRECIO PTAS.  
TOTAL  
115.560 Ptas.**

## HARD SPECTRUM UTILIDADES

1 Spectrum 48 K .....	51.900 Ptas.
1 Teclado profesional .....	16.800 Ptas.
1 Interface paralelo .....	13.100 Ptas.
1 Impresora GP 550 .....	69.990 Ptas.
1 Máquina escribir electrónica SILVER-REED .....	99.900 Ptas.
1 Interface máquina .....	32.990 Ptas.
1 Plotter 4 colores .....	42.500 Ptas.
<b>TOTAL</b> .....	<b>327.180 Ptas.</b>



**PRECIO PTAS.  
TOTAL  
291.000 Ptas.**

## SOFT SPECTRUM JUEGOS TOP TEN

<b>SUPER AJEDREZ 3</b> 48 K 4.600 Ptas.	<b>VALHALLA</b> 48 K 5.500 Ptas.	<b>JET SET WILLY</b> 48 K. 3.500 Ptas.	<b>MAD MARTHA II</b> 48 K 3.500 Ptas.
<b>HOBBIT</b> 48 K 5.000 Ptas.	<b>3D DEATH CHASE</b> 16 K 2.500 Ptas.	<b>ATIC-ATAC</b> 48 K 4.500 Ptas.	<b>ZOOM-3D</b> 48 K 2.500 Ptas.
<b>FIGHTER PILOT</b> 48 K. 3.500 Ptas.	<b>SCUBA DIVE</b> 48 K. 3.500 Ptas.	<b>10 CINTAS PRECIO TOTAL</b> 34.900 Ptas. INCLUYENDO ZX SPECTRUM- 48 K + JOYSTICK <b>PRECIO TOTAL</b> 94.900 Ptas.	

## SOFT SPECTRUM UTILIDADES/EDUCACION TOP TEN 48 K

<b>TRAT. TEXTOS ESPAÑOL</b> 5.000 Ptas.	<b>SUPER QUINIELAS</b> 5.000 Ptas.	<b>ARCHIVO MASTER FILE</b> 4.500 Ptas.	<b>HOJA CÁLCULO OMNICALC</b> 4.500 Ptas.
<b>ESTADISTICA</b> 3.500 Ptas.	<b>64 CARACTERES</b> 2.000 Ptas.	<b>TRIANGULOS</b> 2.000 Ptas.	<b>PIANO ELECTRONICO</b> 2.000 Ptas.
<b>CONTABILIDAD OFICIAL</b> 5.000 Ptas.	<b>SUPERCODE</b> 4.500 Ptas.	<b>10 CINTAS PRECIO TOTAL</b> 33.750 Ptas. INCLUYENDO SPECTRUM 48 K <b>PRECIO TOTAL</b> 80.460 Ptas.	

## REM NOTICIAS

### REM CLUB

Funciona como un club de video. Se adquiere una cinta y se intercambia con otras a 200 Ptas./semana. En cintas inglesas 400 Ptas./semana. Sólo versiones originales.

### REM FRANCHISING

Si quieres montar tu propia mini-tienda de informática o una tienda especializada, envíanos tu dirección y recibirás información completa.

### REM DETALL

Si quieres vender nuestros productos, envíanos tu dirección y recibirás puntual información.

### REM SOFT

Pagamos 20% de royalties de programas originales.

### REM COMPRA

Cintas inglesas 2ª mano en buen estado versiones originales.

## BOLETIN DE PEDIDO

Nombre y Apellidos \_\_\_\_\_  
 Dirección y Teléfono \_\_\_\_\_  
 Deseo recibir más información \_\_\_\_\_  
 Deseo adquirir \_\_\_\_\_  
 Precio total (incluye 300 Ptas. de gastos de envío) \_\_\_\_\_  
 Giro Postal  Giro Telegáfico  Transferencia Bancaria   
 Ingreso en cuenta 3769 8 BANCO DE ESPAÑA - Ptas. Rosas, 44 - MADRID-3  
 Talón adjunto  Talón conformado adjunto   
 Tarjeta VISA número \_\_\_\_\_  
 Fecha Caducidad \_\_\_\_\_ Firma \_\_\_\_\_

**REM** c/. Espronceda, 34-2º int. - MADRID-3

# PROGRAMAS

```

230
1070 IF U$(1)(M)="6" THEN GOTO 1
250
1080 IF U$(1)(M)="7" THEN GOTO 1
270
1090 IF U$(1)(M)="8" THEN GOTO 1
290
1100 IF U$(1)(M)="9" THEN GOTO 1
310
1110 NEXT M
1120 RETURN
1130 PRINT AT 5,7*M+2;" ";AT
6,7*M+2;" ";AT 7,7*M+2;" ";
" ";AT 8,7*M+2;" ";AT 9,7*M+
2;" ";AT 10,7*M+2;" ";AT
11,7*M+2;" "
1140 GOTO 1110
1150 PRINT AT 5,7*M+2;" ";AT 6
,7*M+2;" ";AT 7,7*M+2;" ";AT
8,7*M+2;" ";AT 9,7*M+2;" ";
AT 10,7*M+2;" ";AT 11,7*M+2;"
"
1160 GOTO 1110
1170 PRINT AT 5,7*M+2;" ";AT
6,7*M+2;" ";AT 7,7*M+2;" ";
" ";AT 8,7*M+2;" ";AT 9,7*M+
2;" ";AT 10,7*M+2;" ";AT 11
,7*M+2;" "
1180 GOTO 1110
1190 PRINT AT 5,7*M+2;" ";AT
6,7*M+2;" ";AT 7,7*M+2;" ";
" ";AT 8,7*M+2;" ";AT 9,7*M+
2;" ";AT 10,7*M+2;" ";AT
11,7*M+2;" "
1200 GOTO 1110
1210 PRINT AT 5,7*M+2;" ";AT 6
,7*M+2;" ";AT 7,7*M+2;" ";AT
8,7*M+2;" ";AT 9,7*M+2;" ";
AT 10,7*M+2;" ";AT 11,7*M+2
"
1220 GOTO 1110
1230 PRINT AT 5,7*M+2;" ";AT
6,7*M+2;" ";AT 7,7*M+2;" ";
AT 8,7*M+2;" ";AT 9,7*M+2;"
";AT 10,7*M+2;" ";AT 11,
7*M+2;" "
1240 GOTO 1110
1250 PRINT AT 5,7*M+2;" ";AT
6,7*M+2;" ";AT 7,7*M+2;" ";AT 8
,7*M+2;" ";AT 9,7*M+2;" ";
AT 10,7*M+2;" ";AT 11,7*M+
2;" "
1260 GOTO 1110
1270 PRINT AT 5,7*M+2;" ";AT
6,7*M+2;" ";AT 7,7*M+2;" ";
" ";AT 8,7*M+2;" ";AT 9,7*M+2
";AT 10,7*M+2;" ";AT 11,
7*M+2;" "

```

SI PULSAN ALGUNA TECLA  
LES DARE A CADA UNO/A  
LA TABLA DE NUMEROS

6 13 14 25 26

30 31 33 34 44

50 58 61 78 87

CARTON PARA MI AMIGO JUAN

PULSEN UNA TECLA PARA CONTINUAR

CON ESTE PROGRAMA

UDS. PUEDEN JUGAR AL FAMOSO

B I N O

CUANTOS DESEAN PROBAR SUERTE?

3 23 24 29 32

33 35 42 47 48

53 56 63 79 85

CARTON PARA MI AMIGO SIMEON

PULSEN UNA TECLA PARA CONTINUAR

La versión española de Popular Computing

# ORDENADOR POPULAR

LA REVISTA QUE INTERESA TANTO AL AFICIONADO COMO AL PROFESIONAL



Una publicación que informa con amenidad acerca de las novedades en el campo de las computadoras personales.

**ORDENADOR POPULAR**, la revista para el aficionado a la informática.

Ya está a la venta

Cómprala en su kiosco habitual o solicítela a:

**ORDENADOR POPULAR**

Bravo Murillo, 377  
Tel. 7339662  
MADRID-20

# PROGRAMAS

# 34

```
1280 GOTO 1110
1290 PRINT AT 5,7*M+2;"███";AT
6,7*M+2;"███";AT 7,7*M+2;"███";
███";AT 8,7*M+2;"███";AT 9,7*M+
```

```
2;"███";AT 10,7*M+2;"███";AT
11,7*M+2;"███";
1300 GOTO 1110
1310 PRINT AT 5,7*M+2;"███";AT
6,7*M+2;"███";AT 7,7*M+2;"███";
███";AT 8,7*M+2;"███";AT 9,7*M+
2;"███";AT 10,7*M+2;"███";AT
11,7*M+2;"███";
1320 GOTO 1110
1330 FOR I=1 TO N
1340 FOR J=1 TO 14
1350 FOR O=1 TO 14
1360 IF A(I,O) <= A(I,O+1) THEN GO
TO 1400
1370 LET ZX=A(I,O)
1380 LET A(I,O)=A(I,O+1)
1390 LET A(I,O+1)=ZX
1400 NEXT O
1410 NEXT J
1420 NEXT I
1430 RETURN
1470 FOR F=0 TO 100
1480 NEXT F
1490 PRINT "███","DESEAN JUGAR DE
NUEVO ? (S/N) "
1500 INPUT A$
1510 CLS
1520 IF A$="S" THEN GOTO 120
1530 PRINT "███"," PUES ADIOS
Y HASTA OTRA"
```

12

Antonia García nos ha mandado este lento pero seguro programa de Bingo. La lentitud es sólo al principio, ya que tarda cinco minutos, pero después funciona normalmente. Cada vez que se pulsa una tecla, se selecciona una bola, y así sucesivamente hasta completar las 90.

Veamos cómo trabaja el programa: Primero se prepara la memoria para recibir los 90 números (duplicados). La duplicación la realiza para poder efectuar la comparación (líneas 10 a 40). A continuación la selección de los números que completan todas las bolas (líneas 50 a 120). La parte más importante es el algoritmo de comparación (líneas 130 a 199). Esta parte del programa elige las bolas en el orden seleccionado en la parte de "selección" y

coloca los números por columnas y filas para un fácil repaso. Las líneas 200 a 280 corresponden a la subrutina de instrucciones de juego.

El programa está escrito para el ZX81 con 16 K de memoria, pero puede pasarse en el Spectrum con sólo suprimir las líneas 40 y 120.

Extracción: manual.  
(ZX81 - 16 K)

**PROGRAMA GANADOR DE  
5.000 PTAS**

Antonia García Ochoa de Gracia nos ha enviado desde Chueca (provincia de Toledo) este programa de bingo para el ZX81 que publicamos con mucho gusto, atendiendo al reclamo de los muchos usuarios de este ordenador. Antonia se ha ganado 5.000 pesetas.

```
5 REM "BINGO"
10 RAND
15 GOSUB 200
20 DIM A(90)
30 DIM B(90)
```

```
40 FAST
50 FOR N=1 TO 90
50 LET A(N)=INT (RND*90)+1
70 FOR Z=1 TO N-1
80 LET B(Z)=A(N-Z)
90 IF B(Z)=A(N) THEN GOTO 60
100 NEXT Z
110 NEXT N
120 SLOW
130 FOR H=1 TO 90
140 LET A$=STR$ A(H)
150 PRINT AT 15,3;"ULTIMA";TAB
23;"BOLAS";TAB 4;"BOLA";TAB 21;"
EXTRAIDAS"
160 IF INKEY$="" THEN GOTO 160
170 PRINT AT 10,5;A$;" ";TAB 25
;H
180 IF LEN A$=1 THEN PRINT AT U
AL A$,1;A$
190 IF LEN A$=2 THEN PRINT AT U
AL A$(2),VAL A$(1)+3;A$
195 NEXT H
199 STOP
200 PRINT AT 0,3;"I N S T R U C
C I O N E S";TAB 3;"=====
=====";TAB 0
205 PRINT "PARA COMENZAR PULSE
UNA TECLA.";TAB 0
210 PRINT "CADA VEZ QUE PULSA U
NA TECLA SE SELECCIONA OTRA BOLA
";TAB 0
220 PRINT "EN LA PARTE SUPERIOR
DE LA PANTALLA, APARECEN LOS NU
MEROS EXTRAIDOS.";TAB 0
230 PRINT "EN LA INFERIOR IZQUI
ERDA, LA UL TIMA BOLA EXTRAIDA."
;TAB 0
240 PRINT "EN LA INFERIOR DERE
CHA, EL NUMERO DE ORDEN DE LA MI
SMA.";TAB 0
250 PRINT "LA PREPARACION DEL J
UEGO ES LENTA, ASI QUE... PACIE
NCIA."
260 IF INKEY$="" THEN GOTO 260
270 CLS
280 RETURN
290 SAVE "BINGO"
300 RUN
```

# ZX

# OFERTA DE INTRODUCCION

Una revista para los usuarios de los ordenadores personales SINCLAIR.  
Una publicación mensual que ayuda a obtener el máximo partido al ZX-81  
y al SPECTRUM.

**ZX** le trae cada mes programas, juegos y montajes, además de reportajes  
sobre programación, y la posibilidad de ganar premios realizando programas,  
y otros temas siempre de gran interés.

**GRATIS  
PARA USTED  
si se suscribe a ZX**

Una obra imprescindible  
en la biblioteca de todo  
poseedor de un ordenador  
personal SINCLAIR Spectrum.  
**Cómo usar los colores y los gráficos  
en el Spectrum**, un regalo de 96  
páginas, tamaño  
210 X 270 mm., cuyo precio  
de venta es de 850 Ptas.



Aproveche ahora esta irrepetible oportunidad para suscribirse a **ZX**. Envíe hoy mismo  
la tarjeta adjunta, que no necesita sobre ni franqueo. Dépositela en el buzón más cercano.  
Inmediatamente recibirá su primer ejemplar de **ZX** más el **REGALO** y así durante un año  
(12 ejemplares).

# ZX

*Bravo Murillo, 377  
Tel. 7339662  
MADRID-20*





## ZX-SPECTRUM 16K:

34.950 ptas.

## ZX-SPECTRUM 48K:

45.950 ptas.

Con garantía VENTAMATIC válida en todo el territorio nacional

**Incluido en el precio:** fuente de alimentación, cables para TV y cassette, manuales en inglés, su traducción en castellano, cassette de demostración, cassette de juegos y catálogo de accesorios, libros y programas.

### ACCESORIOS PARA ZX-SPECTRUM

**INTERFACE JOYSTICK:** Compatible KEMPSTON, el más usado por la mayoría de los juegos disponibles hoy y por todos los nuevos que van apareciendo. **3.500 ptas.**

**AMPLIFICADOR DE SONIDO:** Se adapta perfectamente al ordenador e incluye altavoz interno, control de volumen, pulsador de RESET y conmutador SAVE/LOAD. **4.500 ptas.**

**INTERFACE CENTRONICS PARA IMPRESORA:** Para adaptar la mayoría de las impresoras en papel normal al ZX-SPECTRUM. Incluye software para LLIST y LPRINT programable hasta 80 columnas, y COPY de pantalla para impresoras gráficas (en color para la SEIKOSHA GP-700). Cable incluido en el precio. **11.450 ptas.**

**KIT DE AMPLIACION A 48K CON CHIPS:** Suministrado con detalladas instrucciones de montaje para las versiones 2, 3 y 3B del ZX-SPECTRUM 16K. **7.950 ptas.**

**JOYSTICK SPECTRAVIDEO QUICKSHOT:** El más vendido en todo el mundo. Con 4 ventosas para fijarlo a la superficie de juego, empuñadura perfectamente adaptada a la forma de la mano con pulsador de disparo en la misma y en la base del joystick. **2.900 ptas.**

**FUNDA DE PLASTICO:** 400 ptas.

**KIT DE GRAFICOS:** 990 ptas.

**CONECTOR HEMBRA:** 1.000 ptas.

**CONECTOR MACHO:** 300 ptas.

**JUEGO 5 PLANTILLAS TECLADO:** 100 ptas.

**MAS DE 150 PROGRAMAS Y LIBROS DISPONIBLES:** Solicite lista

**ATENCION A LAS PROXIMAS NOVEDADES:** Teclado profesional con sonido, ZX INTERFACE 1 y ZX MICRODRIVES, Trazador digital, Tablero digitalizador, Lápiz de luz, Floppy Disk, Modem, etc.

### IMPRESORAS SEIKOSHA

INTERFACE GRATIS POR LA COMPRA CONJUNTA DE SPECTRUM E IMPRESORA

GP-250 serie/paralelo: 54.900 ptas.

GP-550: 69.900 ptas.

GP-700 color: 98.500 ptas.

GP-50: 26.900 ptas.



**ZX81 1:** 13.450 ptas.

**ZX81 16K:** 17.950 ptas.

Con garantía VENTAMATIC válida en todo el territorio nacional.

**Incluido en el precio:** fuente de alimentación, cables para TV y cassette, manual en inglés, manual en castellano, cassette de juegos y catálogo de programas, libros y accesorios.

**Teclado de pulsadores:** 1.500 ptas.

Solicite lista de programas y libros.



## VENTAMATIC

### VENTA POR CORREO:

Avda. de Rhode, n.º 253 - Apartado de Correos n.º 168  
Tel.: (972) 25 56 16 (24 horas) - ROSAS (Gerona).

### EXPOSICION Y VENTA AL PUBLICO:

c/Córcega, n.º 89, entlo. - BARCELONA-29.

Recorte o copie este cupón.

Fecha: .....

Señores de VENTAMATIC (.....), envíenme:

.....

.....

NOMBRE: .....

APELLIDOS: .....

DOMICILIO: .....

POBLACION: ..... DP: .....

PROVINCIA: .....

Envío giro postal/talón conformado ptas.: ..... fecha: .....

n.º: .....

Para pagos c/reembolso o tarjeta crédito envíe 500,— ptas. gastos envío.

Tarjeta VISA/MASTERCARD n.º .....

CADUCA: ..... Firma

Gastos de envío:

- Pedidos + 7.000 ptas. pagados por adelantado, sin gastos.
- Pedidos - 7.000 ptas. pagados por adelantado, 300 ptas.

**GARANTIA DE DEVOLUCION 14 DIAS EN PEDIDOS PAGADOS POR ADELANTADO**

**CLUB sinclair de**  
*photo copy*

**Hazte socio del 1º Club de Informática de Galicia**

Si quieres formar parte del CLUB SINCLAIR, rellena el cuestionario y envíalo o tráelo personalmente a PHOTO COPY, c/. Teresa Herrera, 9. La Coruña

Pronto recibirás noticias nuestras.

RECIBIR

Apellidos: .....  
Nombre: .....  
Dirección: .....  
Teléfono: .....  
Paseo: .....

MODELO DE ZX-81   
SINCLAIR ZX-Spectrum

Recibido  N.º entrada

*photo copy* DISTRIBUIDOR OFICIAL **sinclair** Teresa Herrera, 9 Telf. 213421 LA CORUÑA

**ELECTRONICA SANDOVAL S.A.**

DISTRIBUIDORES DE  
ROCKWELL-AIM-65  
VIDEO GENIE-EG-2000  
CASIO FX-9000P - SINCLAIR ZX81  
OSBORNE 1 - DRAGON-32  
NEW BRAIN - EPSON HX-20

ELECTRONICA SANDOVAL S.A.  
C/. SANDOVAL 3, 4, 5 - MADRID-10  
Teléfono: 445 75 59-445 76 00-445 18 70-447 42 01

**TRONIK**

Higay, 11-13  
Tel. (93) 212 85 86  
Barcelona-02

HOLA, SOY TRONIK  
TU AMIGO INFORMATICO

- Todo sobre el ZX SPECTRUM:  
• Periféricos  
• Múltiples programas  
• Libros y revistas.  
• Recomendamos tu ordenador como entrada de otro nuevo.  
• Cursos de BASIC a todos los niveles.



**Microshop, s.a.**  
BOUTIQUE INFORMATICA

Juan Sebastián Etxano, 6  
Teléfono 27 20 55. SEVILLA - 11

ZX 81 .....	14.975
ZX SPECTRUM 16K .....	39.900
ZX SPECTRUM 48K .....	52.000
ZX PRINTER .....	17.100

TODOS LOS JUEGOS DEL ZX 81 Y SPECTRUM

INDESCOMP, SOFTWARE, HARDWARE,  
y libros para los micros anteriormente mencionados.

**MAJADAHONDA TECNICOS INFORMATICOS**

SPECTRUM (Juegos, P. Educativos, etc...)  
DRAGON (Más de 400 programas)  
REALIZAMOS PROGRAMAS A SU MEDIDA  
APLICACIONES STANDARD  
CURSILLOS DE BASIC

Urbanización Parque Res. de Madrid  
Parcela A - Local 2  
Tfno. 638 55 15 - Majadahonda (Madrid)

**CONTROL**  
MECANIZACION Y SISTEMAS  
CONDE TORRES CABRELA, 9  
47400 CORDOBA

Ordenadores de gestión, Ordenadores personales, Periféricos, Accesorios y Programas. DISTRIBUIDOR EXCLUSIVO SINCLAIR ZX SPECTRUM  
COMMODORE Microdrive (ya disponible)  
SHARP

**MICROGESA**

Especialistas en Sinclair  
Lápiz óptico; Amplificador de sonido.  
MODEM telefónico,  
MICRODRIVES interface 2 software standard y a medida  
reparamos tu Spectrum

C/ SILVA, 5 - 4.º - Tel. 242 24 71

**ACADEMIA GH**  
MONCLOA

CURSOS DE ANALISIS, PROGRAMACION E INFORMATICA INDUSTRIAL  
LENGUAJES: BASIC, COBOL, PASCAL Y FORTRAN  
GRUPOS PARA JOVENES DE 11 A 16 AÑOS  
GRUPO ESPECIAL COMBINADO CON INGLES ESPECIAL POR CORRESPONDENCIA Y SABADOS  
MICROS IBM, ORIC Y SPECTRUM  
AUTOMATA PROGRAMABLE ALLEN BRADLEY  
PROXIMO CURSO DE BASIC (Comienzo: 2 de Mayo)

Estamos en Hilarión Eslava, 34, 2 B  
MADRID - 15 (junto a Galaxia)  
Tfno. 449 04 40 y 449 75 27

**VALMAR DIV. INFORMATICA**

SU TIENDA DE INFORMATICA EN CADIZ LE OFRECE:

- \* Ordenadores:  
- Sinclair - Sharp - Toshiba  
- Honeywell Bull - Dragon - Apple
- \* Gran biblioteca de programas y libros
- \* Programas conformes a sus necesidades
- \* Periféricos

C. CIUDAD DE SANTANDER, 8  
Tfno. 28 10 69/27 60 42 - TX: 76171 VLVYA E

**AHORRE CASI UN 25 POR CIENTO**

**ZX SPECTRUM**

48K. ....	39.900 Ptas.
16K. ....	31.000 Ptas.
ZX81 1K. ....	11.500 Ptas.
IMPRESORA . . . . .	12.000 Ptas.

SEIS MESES DE GARANTIA  
SERVICIO DE REPARACIONES

¡NUEVO!  
**ORIC ATMOS**  
48K. .... 47.500 Ptas.

VENTA DIRECTA O REEMBOLSO  
**COMPUTER DISKONT**  
Plaza Blasco de Garay, 17, 1  
BARCELONA - 4  
Teléfono 241 55 18

**micrcenter**

MAJADAHONDA

**Especializado en SINCLAIR**

- Micro-ordenadores
- Periféricos y accesorios
- Programas
- Librería de Microinformática

Urb. Jardín de la Ermita  
Majadahonda ( Madrid )

**EPSILON** 

micro-informática

DISPONEMOS DE UN GRAN SURTIDO DE CINTAS DE JUEGOS, ACCESORIOS Y PERIFERICOS: NACIONALES E IMPORTADOS, TANTO PARA LOS SINCLAIR ZX-81 Y SPECTRUM, COMO PARA NUESTRA AMPLIA GAMA DE MICROORDENADORES, ENTRE LOS QUE SE ENCUENTRAN LOS: VIC-20, COMMODORE 64, VICTOR LAMBDA, JUPITER ACE, TOSHIBA Y TODOS LOS QUE DESEES

ECHEGARAY, 9 - Telef. 22 10 27 - 27 00 73  
MALAGA - 15

INGESCON, S. A.

electronica

**LUVI**

**ORDENADORES PERSONALES**

Vizcaya, 6 - Tfno. 230 44 84/ 227 89 62  
MADRID

SINCLAIR / ZX - SPECTRUM

TU  
DISTRIBUIDOR  
EN  
VALENCIA

**CESPEDES**  
COMPONENTES ELECTRONICOS

C/ San Jacinto, 6  
Tfno. 370 35 81 / 370 17 24

**LIBROS - PROGRAMAS  
ACCESORIOS**

**MICRO M WORLD**

ORDENADORES PERSONALES Y  
MICROORDENADORES DE  
GESTION

- SPECTRUM
- KATSON
- ORIC-1
- APPLE
- NEW BRAIN
- ALTOS

SOFTWARE STANDARD  
Y A MEDIDA

CURSILLOS Y FORMACION

En Madrid:

MODESTO LAFUENTE, 63 - Telef. 253 94 54

**ENERGIA SOLAR  
CALEFACCION  
SUELO RADIANTE**

Programas para Cálculo y Dimensionamiento de  
Instalaciones de Energía Solar y Bombas de Calor

Balanza energética  
Análisis de rentabilidad

Programas en cassettes para Spectrum 16 K y 48 K

Pida información a:

J. CANALES - Apartado 129  
Tfno. 79 36 51 - Javea (Alicante).

**ZX SPECTRUM  
en BILBAO**

Programas, libros, información...

**gi gesco-**  
**informática, s. a.**

C/ Telesforo Aranzadi, 1  
Tfno. 1941 431 87 60

**Academia Matemáticas**

**CURSOS DE  
INFORMATICA**

DISTINTOS LENGUAJES

CALLE RECOLETOS, 5 - Teléfono 276 00 15  
MADRID - I

**m micro-**  
**B Bierzo**

Carretera 72 - Telef. 417421 - Apartado 141 - PONFERRADA

CENTRO DE MICROINFORMATICA Y ELECTRONICA  
ORDENADORES PERSONALES ZX-81 SPECTRUM  
VIC-20 COMMODORE 64

Ordenadores de Gestión

Programas Profesionales - Docentes  
de Gestión y de Juegos

CLUB DE USUARIOS - FORMACION



**COMPUTEST**

SERVICIO  
DE REPARACION  
EN 7 DIAS

**SPECTRUM  
ZX81**

**IMPRESORA**

Clara del Rey, 2 - 3.  
Tfno. (91) 415 95 88

**¡ATENCIÓN!**

**USUARIOS DEL MICRO DRIVE  
ZX SPECTRUM**

A partir del próximo 25 de Mayo  
disponemos de Plan Nacional  
Contable para Microdrive

- Archivo para contable.
- Archivo asientos.
- Extracto de cuentas.
- Balances.

**WORLD MICRO Tel. 251 12 00.**



**SUSCRIBASE  
POR TELEFONO**

- \* más fácil,
- \* más cómodo,
- \* más rápido

**Telf. (91) 733 79 69**

**7 días por semana, 24 horas a su servicio**

SUSCRIBASE A





# COMPRO·VENDO·CAMBIO·COMPRO·VENDO·CAMBIO

Me interesaría todo lo relativo con máquinas traductoras y programas de traducción. Colaboraría en esta parcela a cualquier nivel. Javier, Aptdo. 247, Valencia - 1.

**Se vende** para ZX81, teclado profesional Indescomp más 16K soldados internamente al mismo más 3 cintas más 8 boletines Club Nacional Usuarios, 20.000 ptas. Francisco Pérez Lis, Tel. (96) 374 74 28, Valencia.

**Vendo** ZX81, su fecha de compra 27/7/83, con manual, transformador y conectores; en perfecto estado, precio a convenir. También vendo cintas de juegos de 16K (Ventamatic) a muy buen precio. Juan Vallcorba, C/Montcada, 8, 2o., Mataró Tel. (93) 798 36 38.

Nos gustaría tomar **contacto** con usuarios del Spectrum preferentemente de las zonas central y oriental de Asturias, para intercambio de programas e información en general. Benigno o

Marcos. Niembro-Llanes. Tel. 40 76 84.

Estoy interesado en formar un pequeño club entre amigos (sin ningún fin económico) para intercambiar programas del Spectrum 16/48K. Josep María Escudedo García. C/Balmes, 66, 2o., 3a. Barcelona-7.

**Vendo** ZX81 con ampliación a 16 K original Sinclair, con teclado móvil, manual de instrucciones en inglés (regalaría otro en castellano), fuente de alimentación, cables, tres libros con programas y una docena de programas grabados. Todo por 23.000 ptas. Preguntar por Juan (solo tardes) Tel. (976) 49 22 73.

**Desearía** poseer una buena biblioteca sobre Informática, para lo cual aceptaría revistas, libros, publicaciones, etc., a cambio de libros que poseo sobre Energía Solar, Física, Geometría Analítica, Álgebra Lineal, Cálculo Infinitesimal, Electrotécnica, Estadística, etc.

Contactar con Elías Hernández. C/Martínez Cubells, 2, 6ta. Valencia-2, Tel. 351 96-35155 Tfno. (96) 351 55 12.

**Intercambio** programas para Spectrum 48K. Joaquín Bayón López. C/San Mateo, 24, 10mo C. Oviedo-8

Busco **contactos** con usuarios del Spectrum 16/48K. Cambiamos y duplicamos programas. Poseemos gran cantidad de juegos. Preguntar por: Alberto. Tel. (93) 239 27 53 (7 hs. en adelante).

**Desearía** recibir de algunos usuarios programas para ZX81 de 1 y 16K. También para ZX Spectrum de 16 y 48K. Juegos, dibujos, gráficos. Tel. (965) 26 76 63.

**Vendo** Sinclair ZX81 (1/12/83) más 64K (20/2/84) más dos cintas comerciales (una de 16K City Patrol y otra de 1K) más

libro programación ZX81 y algunas revistas. Todo por 28.000 Ptas. Todo en perfecto estado. También cambiaría más 3.000 Ptas. por Spectrum 16K en buen estado; José Luis González. Sevilla. Tel. (954) 76 40 08.

**Vendo** ZX81 en perfecto estado, con ampliación de memoria de 16K, con manual, todos los cables, un libro de juego 1K, varios cassettes de juegos y de aplicación y varios libros para el ZX81. Garantía hasta el 10/4/84. Todo por 16.500 Ptas. Manuel García Álvarez. Barcelona. Tel. (93) 301 62 56.

**Cambio** programas e impresiones con usuarios de Spectrum de Oviedo y Avilés e incluso toda Asturias. Preguntar por: José Manuel Crespo Fernández. C/Valdés Salas, 2, 2do. dcha. Oviedo. Tel. (985) 23 88 66

# COMPRO·VENDO·CAMBIO·COMPRO·VENDO·CAMBIO

**electrónica**  
**LUGO**  
ORDENADORES.

**DRAGON 32**

**ATARI**

**SORD**

**ZX**

**sinclair**

**CABLES ESPECIALES  
CONECTORES  
PROGRAMAS**

BARQUILLO. 40  
MADRID-4

# APRENDIENDO EL

## CUARTA PARTE: EL MOVIMIENTO

Después de tres artículos sobre el código máquina y la memoria del ZX Spectrum, queremos presentar a partir de este número ciertas rutinas y subrutinas en lenguaje ensamblador. Deben tener la función de servir de prototipo a otros programas en Assembler que los lectores podrán realizar en el futuro.

Este artículo tratará sobre el tema del movimiento. Una rutina en código máquina que se encuentra al final del texto servirá como demostración del uso y la velocidad del lenguaje Assembler frente al BASIC.

Rompiendo el hilo del tema y dadas unas consultas que nos han realizado sobre cómo almacenar cualquier programa en código máquina, debemos responder lo siguiente: debe ser guardado en memoria RAM libre al usuario. Sin embargo, hay que fijarse dónde, porque existen determinadas zonas que almacenan valores valiosos que no deben ser destruidos. Resultaría conveniente seguir la explicación que viene a continuación con la ayuda de un plano de la memoria interna del ordenador y de la distribución de la información tal y como lo hemos hecho en el primer artículo de esta serie (ZX n.º 3).

La memoria de pantalla (direcciones 16384-22527) y la memoria de atributos (direcciones 22528-23295) no son muy recomendables. Esta parte de la memoria RAM varía constantemente de información, por el frecuente uso de la orden CLS y los controles de atributos. El comando CLS cambia a 0 el contenido de todas las direcciones de memoria del archivo de pantalla. Con los controles de color se modifican el contenido de las direcciones de memoria del archivo de atributos. La memoria disponible para almacenar código máquina deben encontrarse, por lo menos, por encima (en direcciones de memoria superiores) a las variables del sistema y a los mapas del *microdrive*, si están conectados. Es ahí donde comienza el sistema BASIC.

Si no existiera un programa BASIC se podrían almacenar aquí los códigos. Sin embargo, a partir del momento en el que se escribiera un programa BASIC, estos códigos pasarían a ocupar aquellas direcciones de memoria y el código máquina sería destruido.

Como norma general, que debe ser adecuada a cada caso particular, recomendamos almacenar cualquier programa en código máquina a partir del RAMTOP o a partir de la dirección de memoria, por lo menos, 28000. De esta manera se dispone de aproximadamente cuatro o cinco K de memoria para un corto programa BASIC y sus variables. En caso de necesitar más memoria habría que subir el RAMTOP a direcciones superiores. Cada programa adecúa la posición de RAMTOP a sus necesidades.

Para poder seguir adelante con este artículo partimos de la base de que se han leído y estudiado los tres anteriores. Damos por supuesto que el lector ya conoce conceptos tales como registro, mnemónico y otros.

Volviendo al tema que nos ocupa, el movimiento, es preferible comenzar por lo más sencillo. Vamos a plantearnos primeramente la finalidad del programa, antes de empezar a codificarlo (tómese este sistema como una norma general en la programación, sea cual sea el lenguaje que se utilice). Por ejemplo: conseguir que un cuadrado negro parpadeante se mueva a nuestra voluntad por la pantalla del televisor.

Lo más fácil sería conseguir que este cuadrado negro se imprimiera en pantalla. Vamos a utilizar para ello el archivo de atributos, que es donde el ordenador va almacenando el FLASH, BRIGHT, PAPER e INK de un carácter.

# 26

DONDE  
ALMACENAR EL  
CODIGO MAQUINA

# 27

EL ESQUEMA DEL  
PROGRAMA

# CODIGO MAQUINA

La memoria de atributos ocupa las direcciones que van de la 22.528 a la 23.295 ambas inclusives. La dirección de memoria 22.528 corresponde al cuadrado de la fila 0 columna 0. La siguiente dirección de memoria (22529) corresponderá a la fila cero columna 1 y así sucesivamente hasta la dirección 23295, que corresponde a la fila 23 columna 31.

El programa número 1 es el equivalente en Basic del listado Assembler número 1. Este realiza la impresión del carácter.

Fíjese en el listado número 1 y compruebe cómo el registro doble HL recibe el valor correspondiente a la dirección de memoria 22928 (fila 11, columna 16), aproximadamente en la mitad de la pantalla. El registro simple A, o acumulador, toma el valor 07H cuyo binario es 00000111. Recordando la función de cada bit de un byte de la memoria de atributos y su apareamiento, el número binario 00000111 corresponde al atributo: FLASH 0, BRIGHT 0, PAPER 0, INK 7.

Mediante direccionamiento indirecto el valor del acumulador pasa a la dirección de memoria contenida en el registro doble HL. Puede ser comparado con: POKE (valor de HL), valor de A. De esta manera hemos modificado el valor de una dirección de memoria de un atributo. En la pantalla aparecerá un cuadro en negro. El mnemónico RET nos devuelve el control al BASIC.

```
10 REM > BAJA RAMTOP A 28670
20 CLEAR 28670
30 REM >      INTRODUCE
                CODIGO MAQUINA
40 FOR F=28672 TO 28678
50 READ A: POKE F,A: NEXT F
60 DATA 33,144,89,62,7,119,201
```

*El código máquina introducido por el programa 1 se pone en marcha con RANDOMIZE USR 28672.*

## LISTADO ASSEMBLER 1

Especificaciones: "list ass1" para ZX Spectrum.

Descripción General: Impresión en pantalla de un cuadrado fondo negro, tinta blanca en la dirección de memoria 5990H (22928 dec).

Entrada: —

Salida: Dirección 5880H con valor 07H.

Registros Usados: A, H, L.

```
7000          00100  ORG 7000H ;
7000 219059 00110  LD HL,5990H ;fila 12, columna 16
7003 3E07 00120  LD A,07H ;PAPER 0: INK 7
7005 77 00130  LD (HL),A ;POKE (HL),A
7006 C9 00140  RET ;
0000          00150  END
0000 TOTAL ERRORES
```

```
10 REM > BAJA RAMTOP A 28670
20 CLEAR 28670
30 REM >      INTRODUCE
                CODIGO MAQUINA
40 FOR F=28672 TO 28680
50 READ A: POKE F,A: NEXT F
60 DATA 62,223,219,254,230,4,3
2,248,201
```

*El código máquina introducido por el programa 2 se pone en marcha con RANDOMIZE USR 28672.*

# 28

## ESPERANDO A QUE SE PULSE UNA TECLA

Todos los programas que existen para ordenadores domésticos hacen uso de las teclas para realizar diferentes funciones, a no ser que se utilice el opcional *Joystick*. A menudo se observa:

- Tecla 5 → mueve hacia la izquierda.
- Tecla 8 → mueve hacia la derecha.
- Tecla 6 → mueve hacia abajo.
- Tecla 7 → mueve hacia arriba.

La definición de las teclas está siempre sujeta a la voluntad del programador. En lenguaje BASIC es necesario el uso del comando INKEY\$ junto a la orden IF...THEN. Es corriente ver en programas escritos en BASIC: IF INKEY\$ = "5" THEN... o IF INKEY\$ = "8" THEN... Dada la lentitud del BASIC, la mayoría de los programas, sobre todo los juegos, utilizan el código máquina para el movimiento de un gráfico en pantalla.

Como ejemplo podemos tomar el programa tres. La línea diez tiene la finalidad de modificar el cursor de modo L a modo C. Este POKE puede usarse sin peligro en cualquier programa. La sentencia siguiente hará que el programa "espere", es decir, repitiendo sucesivamente la línea 20, hasta que la tecla I es pulsada. El programa parará en ese momento, por el STOP de la sentencia 30.

```

10 POKE 23658,8
20 IF INKEY$ <> "I" THEN GO TO 2
30 STOP

```

Aunque en código máquina las cosas van de otra manera, se puede asimilar a esto. Comencemos por la distribución de las teclas. La parte superior de la carcasa del ZX Spectrum está ocupada por el teclado. Este consta de 40 teclas, divididas en cuatro filas de diez teclas cada una. Su distribución es igual que en una máquina de escribir convencional. Para el microprocesador el teclado está organizado de diferente manera: existen ocho semifilas, cada una de cinco teclas. Fíjese en la figura número 1 para hacerse una idea.

FIGURA N.º 1

SEMIFI- LA	BIT N.º					VALOR DE ENTRA- DA
N.º	4	3	2	1	φ	
φ	V	C	X	Z	CAPS SHIFT	φ
1	G	F	D	5	A	φFDH
2	T	R	E	W	Q	φFBH
3	5	4	3	2	1	φF7H
4	6	7	8	9	φ	φEFH
5	Y	U	I	O	P	φDFH
6	H	J	K	L	ENTER	φBFH
7	B	N	M	SYMBOL SHIFT	BREAK SPACE	φ7FH
AND	16	8	4	2	1	
	1φφφφ	φ1φφφ	φφ1φφ	φφφ1φ	φφφφ1	

## SOFTWARE

El teclado es un periférico de entrada de información. Cada tecla, como se comprueba en la figura 1, se encuentra en la intersección de una semifila y una columna. La tecla "I" es la intersección de la semifila cinco y la columna dos. Podemos decir que cada tecla es un elemento de un conjunto bidimensional, cuya primera dimensión sería una fila y la segunda, una columna.

La entrada de información a través del teclado se realiza mediante determinados "canales". Uno de los canales o *ports* utilizados es el 254d o OFEH. Observe el cero escrito antes de cualquier número hexadecimal de un byte. El uso de los canales o *ports* se realiza en lenguaje BASIC a través de las órdenes IN y OUT.

Al explorar el teclado debe darse una dirección de memoria para cada semifila del teclado, donde se encuentra la tecla seleccionada. En un programa en lenguaje máquina es el acumulador el que debe contener un valor de entrada antes de encontrarse con el mnemónico IN. Pasemos a explicarlo mediante el listado Assembler 2.

### LISTADO ASSEMBLER 2

Especificaciones: "List ass2" para ZX Spectrum.  
Descripción General: Espera hasta que se pulse la tecla "I".  
Entrada: —  
Salida: —  
Registros Usados: A

```
7000      00100  ORG 7000H ;
7000 3EDF 00110  N. P. LD A,DFH ; semifila Y-P
7002 DBFE 00120  IN A, (OFEH) ; port de entrada OFEH
7004 E604 00130  AND 4 ; verificación tecla "I"
7006 20F8 00140  JR NZ, N.P ; repetir si no pulsada
7008 C9      00150  RET ;
0000      00160  END
7000 N.P (NO PULSADA)
```

El acumulador es cargado con el valor hex 0DFH, que es el valor de entrada para la semifila Y-P. Mediante el mnemónico IN y tras el port OFEH se comprueba el estado de las teclas de esa semifila. Si una tecla está pulsada tiene el valor cero. En su estado normal (no pulsada) tiene el valor uno. El estado de las teclas de una semifila se expresa mediante un byte, que es almacenado en el acumulador tras la instrucción IN. Este byte es llamado **byte de retorno**. Dado que una semifila consta de cinco teclas, el byte de retorno almacena el estado de esas cinco teclas. Son los cinco bits más bajos, los bits 0-4 los que se ocupan de guardar el estado de las cinco teclas de la semifila.

El estado "normal" de una semifila (ninguna tecla pulsada) tras 1) cargar el acumulador con el valor de entrada para una semifila; 2) recoger el estado de las cinco teclas de la semifila escogida con el mnemónico IN y el port OFEH, y 3) introducir (automático) ese estado, expresado en un byte, en el acumulador, es el siguiente: ???11111. Los cinco bits tienen valor 1. Los bits 5-7 carecen de importancia para nosotros, ya que no almacenan el estado de ninguna tecla. En cualquier byte de retorno el bit 0 (el que se encuentra más hacia la derecha) almacena el estado de la tecla de la columna 0 (ver figura 1). Asimismo el bit 1 almacena el estado de las teclas de la columna 1... hasta el bit 4 que almacena el estado de las teclas de la columna 4. Al existir sólo cinco teclas por semifila, los bits 5-7 no almacenan el estado de ninguna tecla y carecen de importancia en este caso.

Cargando en el acumulador el valor correspondiente a una semifila podemos saber el estado de un grupo de teclas, es decir, si están pulsadas o no. Sólo necesitamos el estado de un determinado bit para saber si una tecla está pulsada o no. Existen dos métodos para comprobar esto:

1. Usando la instrucción lógica AND.
2. Usando la instrucción BIT.

# 29

## LA INSTRUCCION LOGICA AND

Dado que últimamente la longitud de mis artículos son centro de las quejas de algunos lectores, este mes explicaré sólo el uso de la instrucción lógica AND, que aunque es más complicada, su uso está más difundido en programas Assembler.

La instrucción lógica AND compara el contenido del registro simple A con otro registro o con un valor n. La comparación se realiza a través del contenido binario del comparador (el acumulador) y del valor a comparar. Tome la figura 2 como ejemplo y observe cómo se comparan todos los bits. El bit 0 de uno, con el bit 0 del otro; el bit 1 de uno, con el bit 1 del otro, y así hasta el bit 7 del acumulador con el bit 7 del elemento a comparar. El resultado de ANDear esos

**FIGURA 2**  
REGISTRO A : 01010101  
ELEMENTO COMPARADO A: 10100101  
RESULTADO AND : 00000101  
(nuevo registro A)  
0 AND 0 = 0  
0 AND 1 = 0  
1 AND 0 = 0  
1 AND 1 = 1

dos valores es introducido automáticamente en el registro simple A. El nuevo contenido binario del registro A es el resultado de ANDear los bits del mismo rango. Solamente si el bit del acumulador es igual a uno y el bit del elemento a comparar es igual a uno, el resultado será uno. Este resultado se almacenará en la misma posición (dentro del byte) que los bits comparados. En el caso que cualquiera de los dos bits a comparar sea cero, el resultado también será cero. La instrucción lógica AND es muy usada en programas ASSEMBLER. Entre otras funciones, puede controlar el estado de determinados bits de un byte. Tome como ejemplo la figura 3.

La comprobación del estado de un bit culmina con el uso del indicador de cero, que ya ha sido tratado aquí en bastantes ocasiones. Si el resultado de utilizar la instrucción lógica AND es igual a cero quiere decir que no existe ninguna pareja de bits del mismo nivel, en la cual ambos sean igual a uno. El indicador de cero toma el valor uno, como tras cualquier operación cuyo resultado sea igual a cero. Si el resultado es diferente de cero, es una prueba que existen por lo menos

**FIGURA 3**  
REGISTRO A: 01101100      01001100  
AND 32 : 00100000      00100000  
RESULTADO A 00100000      00000000  
BIT 5 = 1      Bit 5 = 0  
INDICADOR CERO = 0      INDICADOR CERO = 1

dos bits del mismo nivel con valor uno. En este caso el indicador de cero toma el valor 0.

Una vez explicada la instrucción lógica AND y su uso, podemos volver al listado Assembler 2. La línea 00130 compara con la instrucción lógica AND el valor de retorno de la semifila escogida anteriormente. Este valor está expresado en un byte en el registro simple A. El valor escogido para ANDear es cuatro, cuya configuración binaria es 00000100, puesto que el bit, cuyo estado queremos conocer es el dos (2 elevado a 2 = 4). Compruébelo en la figura 1, donde la tecla "I" ocupa el bit n2. Si la tecla "I" no está pulsada (situación normal), el bit n.º 2 tiene valor uno. En caso contrario, la tecla "I" pulsada, el bit n.º 2 contiene el valor cero. Analicemos estos dos casos posibles:

1) Tecla "I" no pulsada. BIT n.º 2 = 1

AND 4. Resultado = 0

Zero Flag = 0

2) Tecla "I" pulsada. BIT n.º 2 = 0

# SOFTWARE

AND 4. Resultado = 0  
Zero Flag = 1

Aprovechando el cambio del indicador de cero dependiendo de cada resultado, podemos adivinar el estado de una tecla, es decir, si está pulsada o no. Saltos relativos condicionales (JR NZ o JR Z) modifican el control del programa a otras direcciones. En el listado Assembler 2 y en el caso que la tecla "I" no esté pulsada (indicador de cero = 0) el control del programa vuelve a la línea 00110 del listado, a comenzar todo el proceso de nuevo, mediante un salto relativo condicional de cero (JRNZ). Si la tecla "I" está pulsada, el indicador de cero toma el valor uno y la instrucción RET se ejecuta. Este método realiza una función similar a la del programa 3.

El uso de la instrucción AND en el listado ASSEMBLER 2 permite, por ejemplo, que sean dos las teclas que se comprueben al mismo tiempo. Sólo es necesario modificar la línea 00130 del listado por AND x, donde x representa la suma de los valores decimales a sumar para chequear una tecla individualmente. Cambiando la línea 00130 del listado por AND 0CH, el control no vuelve al BASIC si no se pulsan las teclas "I" y "U". Como práctica y antes de pasar al centro del artículo, usted debe ejercitar con este mismo programa Assembler cambiando las teclas a pulsar.

Considero imprescindible para poder seguir este artículo a partir de ahora, el dominar todo el sistema hexadecimal y binario. Para ello puede ayudarse de tablas especiales como las que se publicaron en este mismo espacio hace dos números. Para la total comprensión de este programa deben seguirse todas las instrucciones frente a un ZX SPECTRUM y un televisor o un monitor. Coja su manual de programación y ábralo por la página 183 (apéndice A) para tener a mano los mnemónicos del microprocesador Z 80.

# DINAMIC

Spectrum Software

*En el país del tiempo, acechado por miles de peligros, tu vida corre contra reloj y la única salida es...*

1000 pts  
unidad

... **Denght** (48k)

*"la fuente de la juventud  
la 1ª  
aventura gráfico-conversacional  
en español"*

Autor:  
S. T. C. R. O.

**ARTIST** (48k)

*y saca a flote tus gráficos*

- Ploter de precisión
- Diseño 3D
- 9 bancos gráficos

Autor:  
VICTOR

"Mansión" Dinamic C/Tilos 2 n:21 Montepincipe Boadilla MADRID

Dirección:  
Nombre:  
Teléfono:

Giro postal n°  
 Reembolso  
 Artist  
 Denght  
1000

# 30

## LISTADO ASSEMBLER

El programa en sí tiene la finalidad de demostrar el uso de las teclas en un programa en código máquina. Un cuadrado parpadeante se moverá a velocidades de vértigo (compruébelo usted mismo introduciendo el programa 4) a las órdenes de dos teclas. La tecla "1" para moverse hacia la izquierda y la tecla "0" para moverse hacia la derecha. Pulsando en cualquier momento la tecla "J", el sistema vuelve al BASIC.

Estructura: este programa consta de cuatro partes.

1. Mnemónicos etiquetados con INIC e IMPR (inicialización e impresión). El registro doble HL se inicializa con el valor 5990H (22.928 dec) que corresponde en la memoria de atributos a la fila 12 columna 16. Se va a utilizar este registro HL como puntero del gráfico que va a desplazarse por la pantalla. El valor 87h (135 dec) pasa al acumulador. Este tiene el correspondiente binario 10000111 que en la memoria de atributos representa el conjunto de colores: FLASH 1, BRIGHT 0, PAPER 0, INK 7. Precisamente estos atributos van a ser introducidos en la dirección de memoria contenida en el registro HL mediante direccionamiento indirecto (1.00130, comparar con el listado Assembler n.º 1). El direccionamiento indirecto se expresa mediante el mnemónico LD (HL), A. Observe la importancia de los paréntesis que indican que el contenido es el elemento de trabajo y no el propio registro HL. Este mismo valor de HL se almacena en la dirección de memoria representada por la etiqueta STORE (Guardar) = 6FFDH. Hubiera sido posible utilizar los mnemónicos PUSH y POP para almacenar y recuperar HL. Dado que este último sistema lo utilice en el anterior artículo, hemos preferido presentar un nuevo método. Mediante el mismo (direccionamiento absoluto) los valores de registros se almacenan en direcciones de memoria elegidas a voluntad del programador. Así se guardan valores determinados en direcciones específicas. El resultado final es una ampliación de lugares de información. Las direcciones se expresan mediante etiquetas, que por su uso pueden ser consideradas como "variables".

2. Mnemónicos etiquetados con TECLAS, TEST DER, TEST IZQ Y TEST RET. La etiqueta Teclas da nombre a una importante parte del programa que tiene la función de comprobar si determinadas teclas están pulsadas. El sistema utilizado es el mismo que el explicado en el listado Assembler n.º 2. El acumulador se carga con un valor correspondiente a una semifila. El mnemónico IN A, (0FEH) comprueba al registro A. La instrucción lógica AND verifica el estado de una tecla en particular, en caso de estar pulsada y de acuerdo con el indicador de cero, el control del programa se modifica a rutinas explicadas más adelante. El programa comprueba el estado de las teclas 0, 1 y J quedando a la espera que se pulse una de éstas. Si se pulsa la tecla J el mnemónico RET devuelve el control al Basic. Si se pulsan 1 ó 0 entran en funcionamiento las rutinas IZQ o DER.

3. Mnemónicos etiquetados con DER, IZQ. En la parte del programa etiquetada con DER el puntero (registro HL) se INCrementa en 1 con la orden INC HL. Al incrementarse el registro HL se pasa automáticamente a la dirección de memoria del archivo de atributos situada a la derecha del anterior elemento de pantalla.

En IZQ pasa sistemáticamente lo mismo. La orden DEC HL, DECRementa el puntero en 1. De esta manera el registro HL contiene la dirección de memoria situada a la izquierda del anterior valor de HL.

Todo ello es controlado por la pulsación de dos teclas (0 y 1). Si el cuadrado parpadeante ocupa la columna 31 de cualquier línea y se pulsa la tecla 0 (mover derecha), el cuadrado pasará a ocupar la columna 0 de la fila inmediatamente inferior. Si el cuadrado parpadeante ocupa la columna 0 de una línea y se pulsa la tecla 1 (mover izquierda), el cuadrado pasará a ocupar la columna 31 de la fila inmediatamente superior.

No creo que pueda comprobar esto, por la velocidad que alcanza en su movimiento el cuadrado, al estar programado el mismo en código máquina.

4. Mnemónicos etiquetados con TEST POST, VALIDO y NO VALIDO. Una vez alterado el valor del registro doble HL hace falta comprobar si su posición no rebasa los límites del archivo de atributos (5800H - 5B00H). Para ello utilizaremos la instrucción CP (comparar) y los saltos relativos condicionales.

La instrucción CP (comparar) es de uso muy frecuente en programación

# SOFTWARE

## LISTADO ASSEMBLER 3

Especificaciones: "list ass3" para ZX Spectrum.

Características generales: Un cuadrado parpadeante se mueve por la pantalla a las órdenes de las teclas 1 (derecha) y 0 (izquierda).

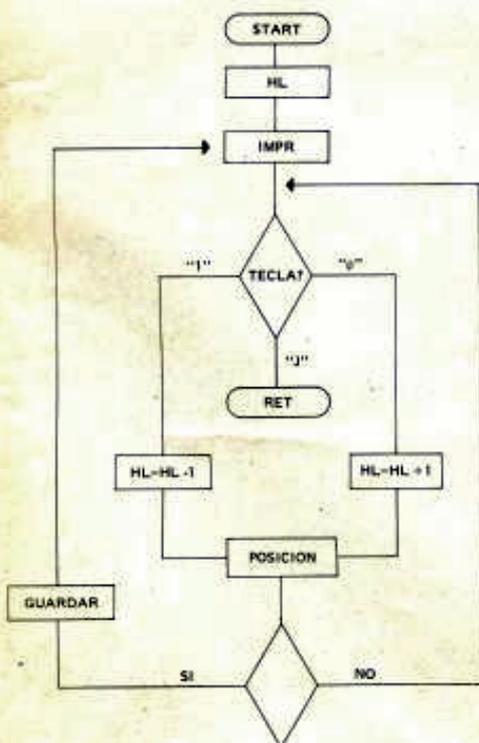
Entrada: —

Salida: Elemento del archivo de atributos con valor 87H.

Registros Usados: A, B, C, H, L.

7000		00100	ORG 7000H ;
7000	215990	00110 INIC	LD HL,5990H ;medio del arvhivo
			de ;atributos
7003	3E87	00120 IMPR	LD a,87H ;FLAH 1:PAPER 0:
			INK 7
7005	77	00130	LD (HL) ,A ;
7006	22FD6F	00140	LD(STORE) ,HL ;almacena HL
7009		00145 TECLAS	;
7009	3EEF	00150TESTDER	LD A,0EFH ;semifila 6-0
700B	DBFE	00160	IN A,(0FEH) ;
700D	E601	00170	AND 01H ;tecla 0
700F	2002	00180	JR NZ TEST IZQ ;
7011	1813	00190	JR DER ;
7013	3EF7	00200TESTIZQ	LD A, 0F7H ;semifila 1-5
7015	DBFE	00210	IN A, (0FEH) ;
7017	E601	00220	AND 01H ;tecla 1
7019	2002	00230	JR NZ TEST RET ;
701B	180C	00240	JR IZQ ;
701D	3EBF	00250TESTRET	LD A,0BFH ;semifila H-ENTER
701F	DBFE	00260	IN A, (0FEH) ;
7021	E608	00270	AND 08H ;tecla J
7023	20E4	00280	JR NZ TEST DER ;
7025	C9	00290	RET ;
7026	23	00300 DER	INC HL ;
7027	1801	00310	JR TEST POS ;
7029	2B	00320 IZQ	DEC HL ;
702A	7C	00330TESTPOS	LD A,H ;
702B	FE57	00340	CP 57H ; HL menor que 22528?
702D	280D	00350	JR Z, NO VALIDO ;
702F	FE5B	00360	CP 5BH ; HL mayor que 23295?
7031	3009	00370	JR NC, NO VALIDO ;
7033	ED4BFD6F	00380 VALIDO	LD BC, (STORE) ;C=antiguo HL
7037	3E38	00390	LD A,38H ;PAPER 0: INK 7
7038	02	00400	LD (BC) ,A ;
703A	18C7	00410	JR IMPR ;
703C	2AFD6F	00420NO VALIDO	LD HL, (STORE) ;recupera HL
703F	18C8	00430	JR TECLAS ;
0000		00440	END ;

STORE 6FFD  
 INIC 7000  
 IMPR 7003  
 TECLAS 7009  
 TEST DER 7009  
 TEST IZQ 7013  
 TEST RET 701D  
 DER 7026  
 IZQ 7029  
 TEST POS 702A  
 VALIDO 7033  
 NO VALIDO 703C



# SOFTWARE

## PROGRAMA 4 (correspondiente al listado ASSEMBLER 3)

```
10 REM BAJA RAMTOP A 28650
20 CLEAR 28650
30 REM INTRODUCE C/M
40 FOR F=28672 TO 28736
50 READ A: POKE F,A: NEXT F:LET T=0
60 DATA 33, 144, 89, 62, 135, 119, 34, 253, 111, 62, 239, 219, 254, 230, 1, 32, 2,
24, 19, 62, 247, 219, 254, 230, 1, 32, 2, 24, 12, 62, 191, 219, 254, 230, 8, 32, 228
70 DATA 201, 35, 24, 1, 43, 124, 254, 87, 40, 13, 254, 91, 48, 9, 237, 75, 253, 111,
62, 56, 2, 24, 199, 42, 253, 111, 24, 200
80 FOR F= 28672 TO 28736
90 LET T= T + PEEK F: NEXT F
100 IF T < > 7242 THEN PRINT "ERROR EN DATAS" : STOP
110 PRINT "PULSA UNA TECLA PARA PONER EN MARCHA EL
CODIGO MAQUINA": PAUSE 0: RANDOMIZE USR 28672
```

Assembler. Posibilitan al programador el poder comprobar el valor almacenado en el registro A con el almacenado en o por otro registro o con un valor N.

Sólo hay tres casos posibles, fíjese en la figura número 4. Dependiendo del resultado de la comparación indicadores toman valores determinados. Los dos indicadores que toman parte en instrucciones CP son el de cero y el de arrastre (CARRY FLAG).

En el primer caso dos valores a comparar iguales, el indicador de cero recibe valor uno y el de arrastre, cero.

En el segundo caso el acumulador menor que el valor comparado, el indicador de cero toma valor cero y el de arrastre uno.

### FIGURA 4

- (1)  $A = V \rightarrow$  INDICADOR DE CERO = 1;  
INDICADOR DE ARRASTRE =  $\phi$
- (2)  $A < V \rightarrow$  INDICADOR DE CERO =  $\phi$   
INDICADOR DE ARRASTRE = 1
- (3)  $A > V \rightarrow$  INDICADOR DE CERO =  $\phi$   
INDICADOR DE ARRASTRE =  $\ll$

En el tercer y último caso el acumulador mayor que el valor comparado, el indicador de cero recibe valor cero y el de arrastre también.

Aquí se puede demostrar la gran utilidad de la instrucción CP. En el programa Assembler el acumulador toma el valor del registro H (byte de orden alto del puntero) y se compara con el 57H. Para poder ser impreso, el registro debe ser mayor a 57H. Mediante CP 57H se determina si es igual, mayor o menor.

El registro HL no será válido si H es igual a 57H (fuera de pantalla). Por ello si tras la comparación el indicador de cero toma valor uno ( $H = 57$ ), el control pasa a la parte etiquetada con NO VALIDO.

Igualmente, en el caso que el registro H sea mayor o igual a 5BH, no será válido. La comparación se realiza normalmente y si tras ella el indicador de arrastre (CARRY FLAG) toma el valor cero (registro H mayor o igual a 5BH) el puntero no será válido.

De esta manera tenemos dos casos posibles: VALIDO o NO VALIDO.

Si el puntero ha sido aceptado como válido la dirección de memoria correspondiente al último elemento tratado toma valor 38H (56 dec = PAPER 0: INK 7). Se consigue el efecto de "borrar" el cuadrado.

Esto se realiza con el registro doble BC y de nuevo el direccionamiento indirecto. El control del programa se modifica hasta la posición etiquetada con IMPR y el proceso vuelve a comenzar.

Si el puntero no ha sido aceptado, HL recupera su valor anterior almacenado en store (1.00420) y el proceso continúa a partir de las rutinas de teclas.

Continuaremos en el próximo número.

Juan Martínez Velarde

Como ejercicio proponemos establecer una pausa para observar bien el continuo movimiento del cuadrado parpadeante. Aceptaré de buen grado todas las versiones realizadas por ustedes, bien a nivel personal en PEEK AND POKE, GENOVA 11 (tel. 91/419 56 48). MADRID-4, o por correo a mi nombre en la Redacción de la revista ZX.

# ELLOS SON FAMOSOS EN EL MUNDO DE LOS VIDEOJUEGOS

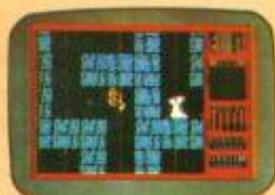
(BUGABOO. LA PULGA. N.º 1 EN INGLATERRA)



BUGABOO C.B.M. 64



BUGABOO Spectrum



FRED Spectrum



TU JUEGO

## ¿QUIERES SERLO TU TAMBIEN?

¿Te gusta programar? ¿Eres capaz de escribir en código máquina? ¿Tienes una buena idea? ¿Has conseguido alguna rutina espectacular? ¿Estás creando un juego?..... INDESCOMP. SOFTWARE, invita a todos los entusiastas de la programación con conocimientos y práctica del lenguaje ensamblador (código máquina 6502 y Z80) a participar y colaborar con nuestro Departamento de videojuegos en la creación de nuevos "best sellers".

Envíanos urgentemente tus ideas y experiencias a:

**indescomp**  
SOFTWARE

DIRECTOR DE PROGRAMAS

P.º DE LA CASTELLANA, 179 - MADRID-16 - TELEF. 279 31 05

# OPERACION CAMBIO

ZX SPECTRUM:  
48K: 52.000 Ptas.  
16K: 39.000 Ptas.  
ó desde 1.393 Ptas al mes



## OFERTAS MES DE MAYO

- Joystick	2.600 Ptas
- Programas In- gleses	1.100 Ptas
- Interface impreso- ra KEMPSTON	8.000 Ptas
- 29 Programas Melbourne	7.500 Ptas

La idea es así de sencilla: si tienes un SINCLAIR ZX 81 y deseas comprarte un SPECTRUM, nosotros te ayudamos. SINCLAIR STORE va a valorarte tu ZX 81 esté como esté: nuevo, ... viejo, ... funcioando, ... estropeado. Te lo vamos a aceptar como entrada del SPECTRUM que nos compres, pagando al mes, tan solo, 1.393.- Ptas. Así, ahora, el conseguir tu SPECTRUM es más sencillo. Ven a vernos o ponte en contacto con nosotros desde cualquier parte del Territorio Nacional.

**Tener ahora tu Spectrum es más fácil.**

**Abierto sábados por la tarde.**

Aparcamiento gratuito para nuestros clientes.  
(C/. Magallanes, 1)

**sinclair store**  
Bravo Murillo, 2 - Telf. 446 62 31 - Madrid