

Marzo 1985 - 250 ptas.

# Todospectrum

AÑO 1 - NUMERO 7.

REVISTA EXCLUSIVA PARA USUARIOS



## Del 48 al plus. El kit paso a paso

**Todos los libros de código máquina**

**Análisis de diskettes  
para Spectrum**

**Logo, lenguaje educativo**

**Juegos, Palos,  
romanos en  
Pascal, etc.**

# INVEDISK 200



J.M. PUBLICIDAD

## EL PASO MAS SERIO

### PARA EL SPECTRUM

Lo más nuevo para tu Spectrum,  
por fin ha llegado.

INVESTRONICA te ofrece  
el sistema de discos.

Lo último en la tecnología de microinformática.

Ve e infórmate en  
tu concesionario INVESTRONICA.





Portada: A. Fregenal

AÑO I • NUMERO

7

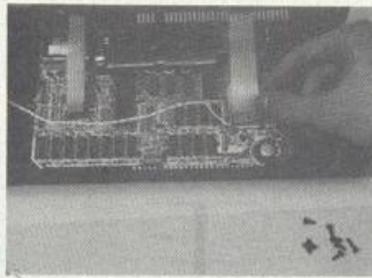
**DIRECTOR:**  
Simeón Cruz  
**COORDINADOR EDITORIAL:**  
Emiliano Juárez  
**REDACCION:**  
Juan Arencibia, Fernando García, José C. Tomás, Gumersindo García, Luis M. Brugarolas, Ricardo García, Santiago Gala  
**DISEÑO:** Ricardo Segura

Editado por  
**PUBLINFORMATICA, S. A.**  
Presidente: Fernando Bolín  
Director Editorial: Norberto Gallego

**Administración:**  
INFODIS, S. A.  
**Gerente de Circulación y ventas:**  
Luis Carrero  
**Producción:**  
Miguel Onieva  
**Director de Marketing:**  
Antonio González  
**Servicio al cliente:**  
Julia González. Tel. 733 79 69  
**Administración:**  
Miguel Atance y Antonio Torres  
**Jefe de Publicidad:**  
María José Martín  
**Dirección y redacción:**  
Bravo Murillo, 377-5.º A. Tel. 733 74 13  
Telex: 48877 OPZX e 28020 Madrid  
**Administración y Publicidad:**  
Bravo Murillo, 377-3 E. Tels. 733 96 62/96  
**Publicidad Madrid:**  
María Lavallo  
**Publicidad Barcelona:**  
María del Carmen Ríos, Jorge González, Pelayo, 12.  
Tel. (93) 301 47 00 ext. 27 y 28.- 08001 Barcelona.  
Depósito legal: M-29041-1984  
Distribuye S.G.E.L.  
Avda. Valdeparra, s/n.  
Alcobendas-Madrid.  
Fotomecánica: Karmat, C/  
Pantoja, 10. Madrid.  
Fotocomposición: Artecomp.  
Imprime: Héroes, C/ Torrelara, 8. Madrid.

Esta publicación es miembro de la Asociación de Revistas de Información **ARI** asociada a la Federación Internacional de Prensa Periódica, FIPP.

**SUSCRIPCIONES:**  
Rogamos dirijan toda la correspondencia relacionada con suscripciones a:  
**TODOSPECTRUM**  
EDISA: Tel. 415 97 12  
C/ López de Hoyos, 141-5.º  
28002 MADRID  
(Para todos los pagos reseñar solamente TODOSPECTRUM)  
Para la compra de ejemplares atrasados dirijan a la propia editorial  
**TODOSPECTRUM**  
C/ Bravo Murillo, 377-5.º A  
Tel. 733 74 13-28020 MADRID



**DEL 48 AL PLUS. EL KIT PASO A PASO.** Montamos el kit en menos de una hora. **4**

**¿PLOTTER PARA EL SPECTRUM?** Experiencias de un instituto con desarrollos prácticos. **10**

**JUEGOS.** Spectrum computing y hunter killer **12**

**DESPLAZAMIENTO PIXEL A PIXEL.** Los lectores tienen la palabra. **15**

**ESPECIAL LIBROS CODIGO MAQUINA.** Todos los títulos del mercado nacional y extranjero. **18**

**LAPIZ OPTICO.** Programación del montaje que tratamos en el número anterior. **30**

**EL LOGO EN LA ESCUELA.** El lenguaje de la educación. **34**

**DESCUBRIMIENTO DE UN NUEVO LENGUAJE: PASCAL.** Cuarto capítulo. **41**

**FLOPPYS PARA SPECTRUM.** Análisis de los principales sistemas de almacenamiento de datos. **48**

**PROGRAMAS:** Palos, amortizaciones, romanos en pascal... **52**

**GUSANEZ:** ¡La amenaza del plus tiene éxito! **62**

**PREGUNTAS Y RESPUESTAS:** Más preguntas y una reflexión sobre los piratas de *software*. **62**

**EL CORCHO.** El mercado de segunda mano va en aumento. **65**

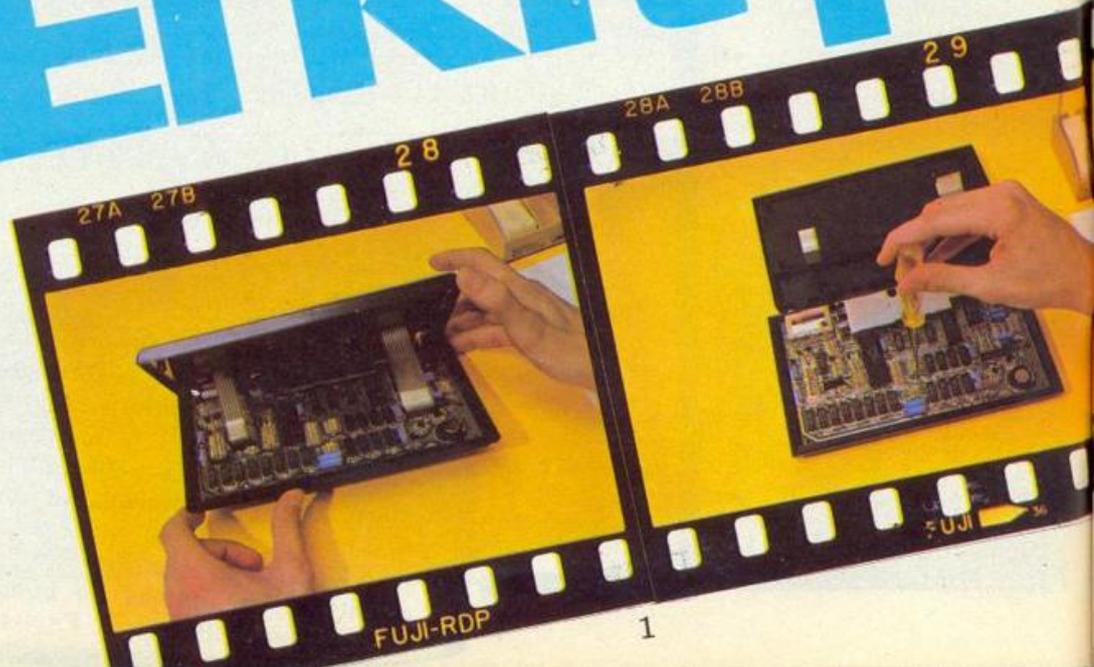
**DIGASELO EN VOZ ALTA: VERBOT LE ENTENDERA.** Un robot amigo de los niños. **66**

Dejamos para el próximo número el análisis del QL, coincidiendo con su lanzamiento oficial. Ordenador del que sin duda se hablará largo y tendido, al igual que del Kit que permitirá a su Spectrum trabajar con un teclado "Plus" y del que damos buena cuenta en este número. Número en el que encontraréis más páginas dedicadas a artículos y a secciones fijas como Preguntas y Respuestas o El Corcho. Con alegría comprobamos vuestra creciente participación, aunque no podamos responderos a todos con la rapidez necesaria. Paciencia.



Del 48 al plus.

El kit pa



Más de un usuario de Spectrum se siente insatisfecho con algunos detalles de acabado de su aparato: por ejemplo con la ausencia de un botón RESET, que obliga a desenchufar y enchufar el aparato para salir de los programas protegidos, con el consiguiente sufrimiento de la clavija de conexión RED. La pobre calidad del teclado, que impide la escritura tipo "máquina de escribir" es también un argumento habitual de quienes niegan la posibilidad de aplicaciones serias para el Spectrum. Otro problema derivado de la economía de diseño del Spectrum es su refrigeración insuficiente, que causa problemas de estabilidad en algunos aparatos, después de largo tiempo de conexión a la red.

Se han multiplicado las soluciones, por parte de fabricantes independientes, para estos problemas: botones de RESET, montajes para cumplir el mismo fin, teclados "profesionales" que suplen las deficiencias del teclado original, tala-

dos para mejorar la refrigeración de la máquina, etcétera.

Hasta el momento, **Sinclair Research** no había dicho una palabra sobre el tema, pero ya con la aparición del Spectrum plus trató de solucionar estos problemas a los futuros compradores mejorando el diseño en varios aspectos: botón de RESET, caja más amplia que refrigera a través de dos rejillas de ventilación, teclado que incorpora como una sola tecla funciones que antes exigían varias pulsaciones, y que permite mayor velocidad de escritura y más seguridad en su uso por sus características semimecánicas.

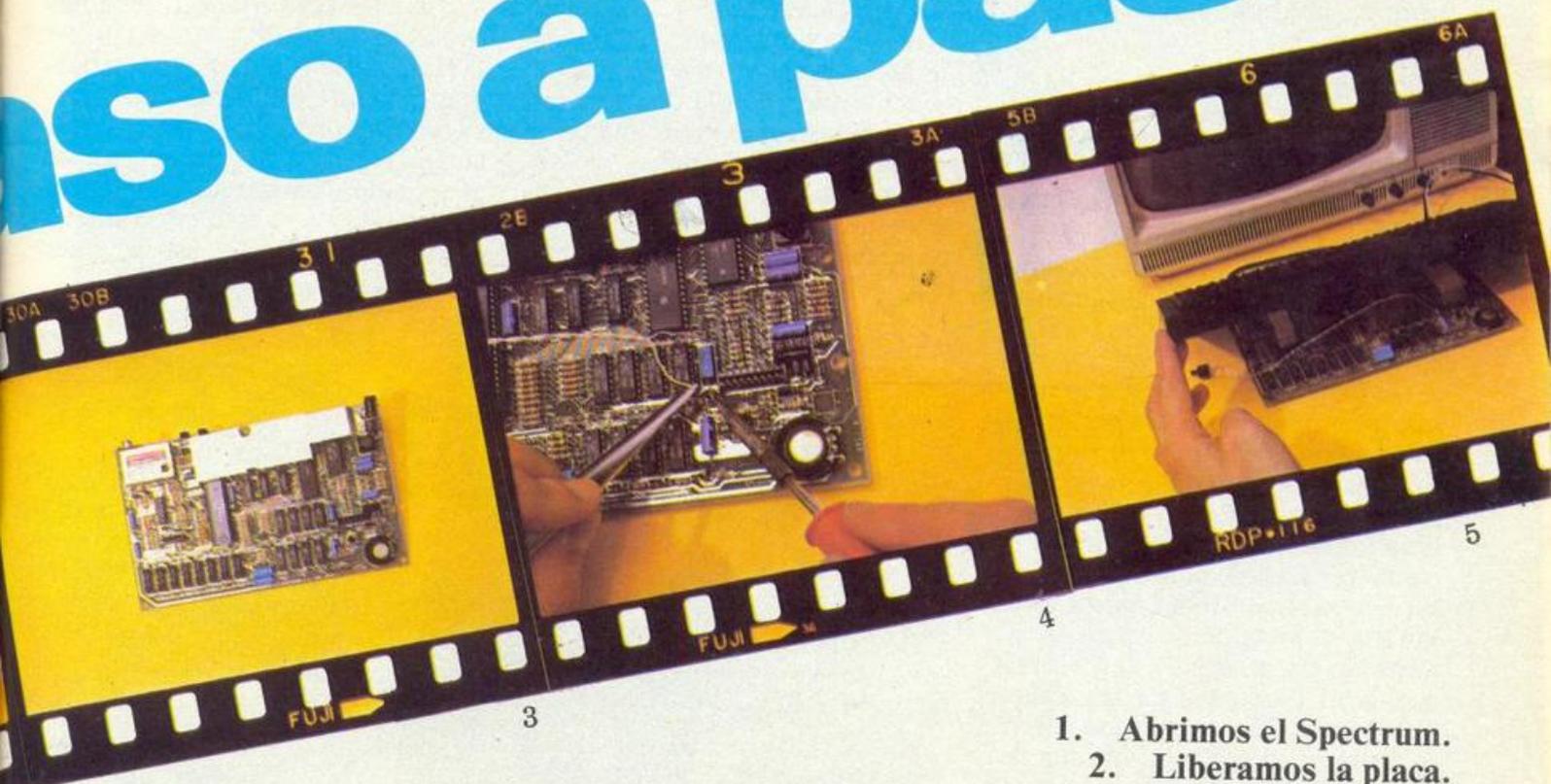
Para la gente que tenía ya el Spectrum, los problemas especialmente con el teclado, habían llevado a muchos usuarios a la adquisición de teclados de mayor calidad proporcionados por diversos fabricantes. En un próximo número publicaremos un informe sobre los teclados adaptables al Spectrum,

donde se analizarán detalladamente las opciones que se le ofrecen al usuario necesitado de un teclado profesional, apto para la mecanografía en aplicaciones comerciales y de proceso de texto.

Otros usuarios esperaban la aparición de ofertas de cambio de su Spectrum por un Plus, para disfrutar así del nuevo aparato. A éstos les podemos decir que no esperen más: no habrá ofertas de cambio y, si las hay, serán en peores condiciones que la oferta lanzada recientemente en el Reino Unido por Sir Clive Sinclair: por 30 libras (unas 6.000 pesetas), los poseedores de Spectrum pueden adquirir un kit que les convierte su máquina en un Plus con la ayuda de un destornillador "de estrella", un soldador y una mesa de trabajo.

Para los que sufrán escalofríos de pensar en acercar el soldador a la placa del Spectrum, los chicos

# so a paso



1. Abrimos el Spectrum.
2. Liberamos la placa.
3. Retiramos la carcasa.
4. Soldadura del Reset.
5. Prueba del teclado.

de Sinclair les hacen el trabajo por otras 20 libras (en total 10.000 pesetas). ¿Y en España qué?, preguntarán. **Investrónica** dará esta opción a los numerosos poseedores de Spectrum que quieran cambiar la apariencia (y el tacto) de su aparato. De precios y de plazos no se sabe todavía nada.

TODOSPECTRUM ha querido analizar este kit para que los lectores conozcan las dificultades que les reportará su montaje, así como las características de este teclado, de cara a una futura comercialización de este producto en nuestro país.

La caja en que viene presentado, incluye un teclado de Plus, una serie de tornillos y accesorios para adaptar a este, el botón de RESET, el manual del Spectrum plus y el *cassette* de demostración, así como las instrucciones de montaje y un sello de garantía para sustituir en el montaje final el que llevaba el Spectrum. La única diferencia que pudimos encontrar entre estos elementos y los correspondientes del Plus corresponden a la palabra *Upgraded* en lugar del número de serie del aparato, justo bajo el *bus* de expansión.

Pasemos a comentar las características de la carcasa (ver Fig. B). Muy parecido al del QL, incluye un teclado de plástico de diseño muy atractivo, compuesto de 58 teclas de color negro. Estas presionan sobre una membrana donde los cruces entre pistas conductoras señalan la pulsación de las teclas al ser presionadas a través de una pieza de goma moldeada que facilita la recuperación del conjunto a su posición de reposo. Por su parte inferior, una lámina plástica de color blanco protege el conjunto. Una mejora con respecto al Spectrum (ver fig. A), donde el teclado mecánico superior no existe, pulsándose directamente sobre la lámina de goma, donde van estampadas las funciones de las teclas. También el mecanismo de cierre es distinto: en el Spectrum una lámina metálica, pegada a la caja, cierra el conjunto por su parte superior, existiendo solo dos ranuras en la caja por las que pasan las

*Realizado el montaje  
no existe diferencia  
con el Plus*

“pistas” que llevan la señal a la placa.

Pasemos al montaje en sí. Las instrucciones son claras. Lo primero que debemos hacer, una vez provistos de las correspondientes herramientas, es desmontar el Spectrum. Este hecho invalida la garantía, aunque las instrucciones informan a los usuarios ingleses de que ésta volverá a entrar en vigor en el momento en que el Spectrum “remozado” vuelva a estar en funcionamiento. Para desmontarlo se deben retirar los cinco tornillos de la parte inferior. Una vez hecho esto, se pueden separar las dos partes con mucho cuidado para no dañar las cintas que unen la membrana del teclado con la placa de circuitos. Estas cintas no están hechas para ser montadas muchas veces y, aunque resistirán perfectamente la operación, no conviene “jugar” con ellas ya que son la parte más delicada del teclado.

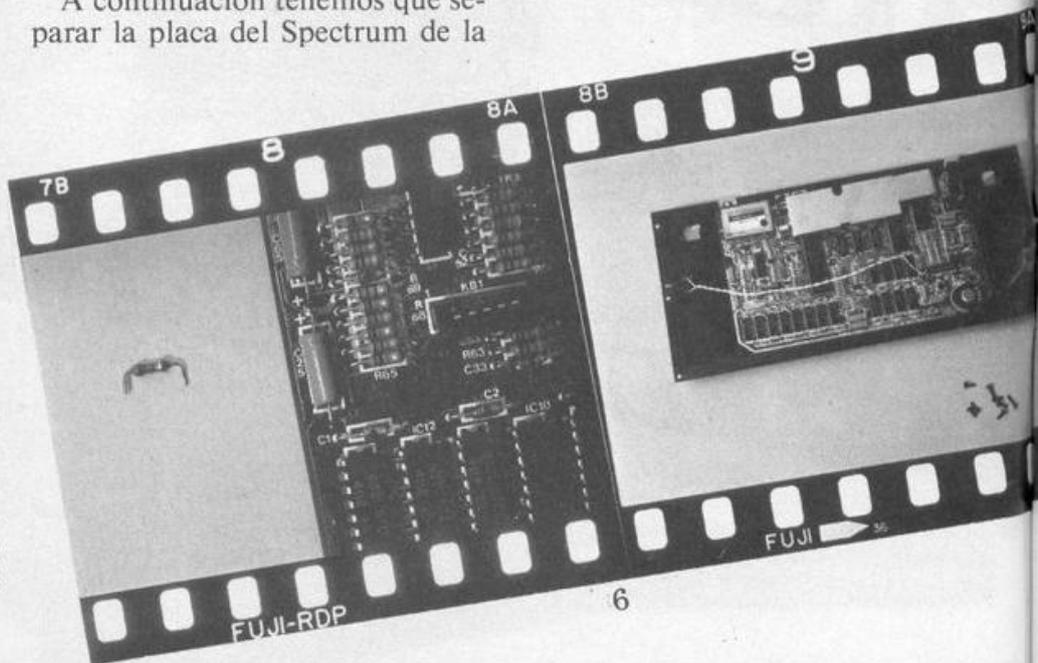
A continuación tenemos que separar la placa del Spectrum de la

parte inferior de la carcasa. Para ello basta retirar un tornillo de la parte central del circuito impreso. Procedemos entonces a retirar cuidadosamente la placa. En este momento podremos ver con claridad la versión que poseemos. A lo largo del tiempo el diseño del Spectrum ha ido variando para solucionar problemas de calentamiento. Existen tres versiones principales: la más antigua tiene la aleta de refrigeración en la parte delantera derecha de la placa, y las 32K extras de los modelos con 48K van montadas sobre una segunda placa por encima de los conectores.

La segunda versión tiene el radiador de refrigeración en la misma posición que la anterior, pero los circuitos se han “apretado” para permitir a toda la memoria residir en la misma placa. Aparecen asimismo componentes (resistencias, condensadores,...) no presentes en la versión anterior.

La versión tres es la que, con pequeños cambios, sigue llevando el Spectrum plus en su interior. Tiene la aleta de refrigeración sobre los conectores, y de mayor tamaño para facilitar la disipación de calor. A ella corresponden las fotos de nuestro artículo.

Hemos realizado este inciso porque el montaje depende de la versión que poseamos. Para las versiones 1 y 2, es necesario cambiar el radiador, ya que el antiguo no encaja correctamente en el nuevo chasis. La hoja de instrucciones



explica claramente cómo realizar esta adaptación, caso de ser necesaria.

El paso siguiente es el más peligroso: fijar el botón de RESET. Para ello, es necesario soldarlo entre las dos "patas" del condensador C27. Si no nos atrevemos con el soldador, se nos recomienda buscar un amigo que lo haga o dejar el RESET sin conectar. Para no dañar los circuitos del Spectrum es conveniente utilizar un soldador de 15 a 25 Watios. Aunque ya hemos publicado instrucciones para realizar una buena soldadura, conviene que tengáis en cuenta una serie de factores: en primer lugar, no acerquéis el estaño a la placa si no tenéis buen pulso. Una gota de estaño entre dos pistas os puede mandar el Spectrum directo al taller de reparaciones. El mejor procedimiento consiste en estañar los cables que realizarán el contacto. Después, con la punta del soldador cubierta de estaño (sin gotas), cubrir las "patas" del condensador. No dejéis calentarse demasiado el condensador. Cuando ambas partes están estañadas, se puede llevar el cable a su posición, y acercar brevemente la punta del soldador. Esto debe bastar para obtener un buen contacto.

Las instrucciones recomiendan que "pasemos" del RESET o se lo encarguemos a un amigo "mani-

tas" en caso de duda. Pero supongamos que ya lo habéis hecho.

Pasamos entonces a probar el nuevo teclado. El conjunto incluye una resistencia de 22k Ohmios, que deberemos soldar en paralelo con la R68 si tenemos problemas con alguna de las teclas en el nuevo teclado. Pero no adelantemos acontecimientos.

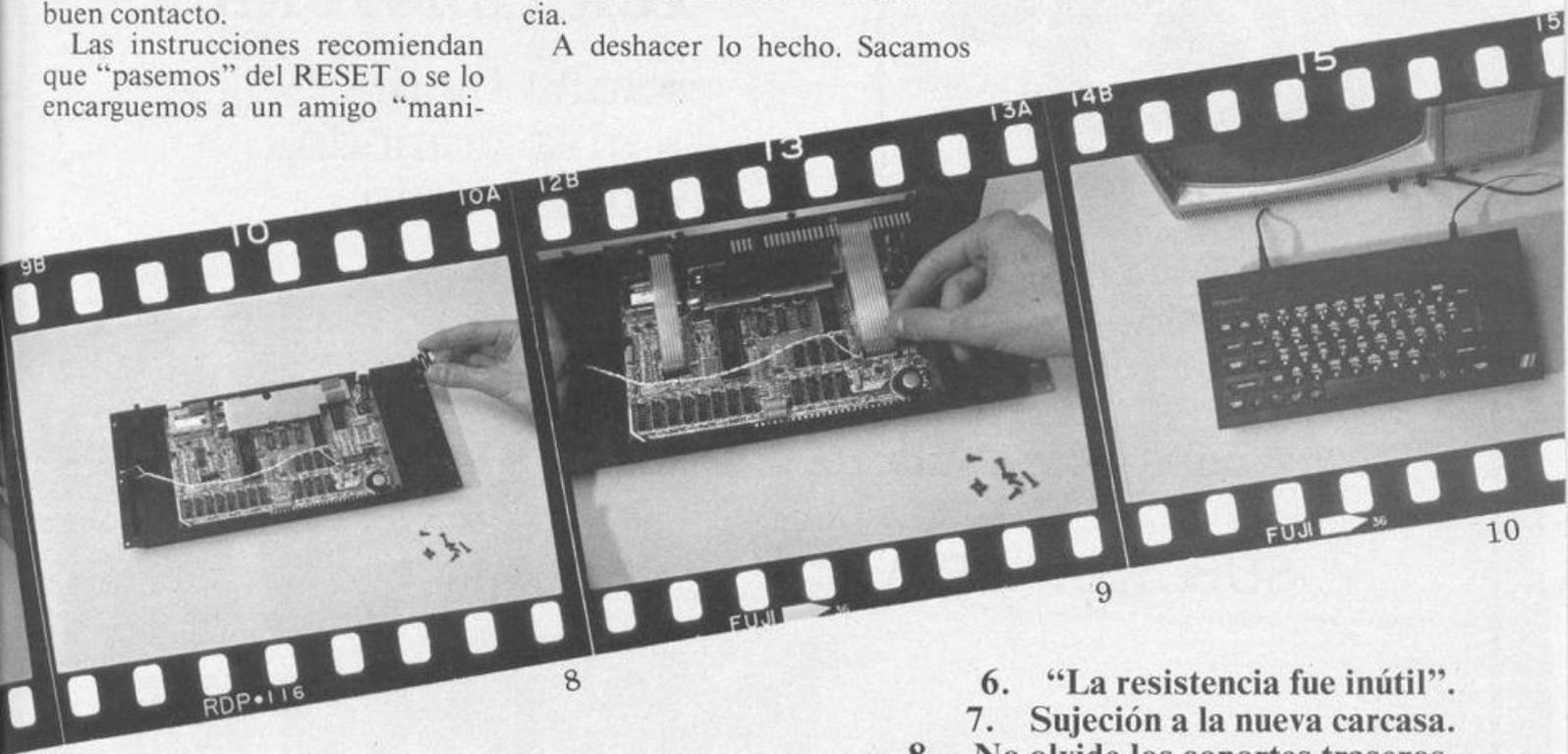
Para probar el teclado se deben conectar las cintas correspondientes a la caja del Plus sobre la placa del Spectrum. Esta se debe haber situado previamente sobre una superficie no conductora (unos periódicos bastan). Una vez conectadas las dos cintas se coloca la parte superior del Plus en posición, y se conecta la toma de TV y alimentación.

El Spectrum debe arrancar con el mensaje de *Copyright* y responder perfectamente a todas nuestras pulsaciones. Se deben probar todas las teclas, y muy especialmente la de STOP, que suele dar los mayores problemas. Si todas funcionan correctamente, no es necesaria la soldadura. En caso contrario, deberemos pasar nervios otra vez. Cuando realizamos el montaje, el teclado funcionó perfectamente sin necesidad de añadir la resistencia.

A deshacer lo hecho. Sacamos

nuevamente las cintas del teclado y procedemos a montar la placa sobre la parte inferior de la caja del Plus. Se nos proporcionan dos tornillos que se deben fijar por la parte delantera de la placa. La tecla de RESET va asomando por un orificio en la parte izquierda de la caja, insertado a presión. Una vez fijado el teclado (tened cuidado de que no "baile", pero no apretéis los tornillos tanto como para romper la placa), procederemos a colocar tres pequeñas almohadillas de espuma sobre el modulador de TV y la aleta del radiador. La misión de estas piezas es doble: proteger las cintas del teclado de ser "pilladas" por estos elementos y evitar que su contacto con la aleta las caliente excesivamente. Para finalizar el montaje, basta insertar las dos "patas" del Plus en su posición, colocar sobre ellas dos muelles de material plástico que las fijan, y cerrar el aparato. Cuidado con no estropear las cintas de teclado en esta última operación. Ocho tornillos completan la operación.

Después, volvemos a probar el aparato. Sí la primera vez nos funcionó correctamente, pero ahora no, el problema residirá probable-



6. "La resistencia fue inútil".
7. Sujeción a la nueva carcasa.
8. No olvide los soportes traseros.
9. Conexión del teclado a la placa.
10. Y esto es todo: Un "Plus" en una hora.



Figura A. Teclado Spectrum

mente en una mala colocación de los conectores del teclado. Otros posibles problemas vienen descritos en el manual. Especialmente hacemos hincapié en que la máquina no debe funcionar NUNCA sin la aleta de refrigeración colocada.

Y ya tenemos un Plus. Para que no podamos envidiar a nadie, el conjunto incluye el manual de instrucciones del Plus, así como la cinta de demostración. En esto Sinclair se ha esmerado en hacer la

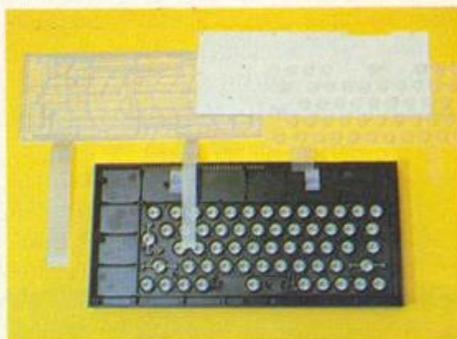


Figura B. Teclado Plus.

máquina más agradable: pasamos de un manual pensado para un público de "enterados", que prefieren la densidad de información a la buena presentación, a un manual donde, manteniendo la cantidad de información, se ha cuidado la presentación y la facilidad de lectura. El buen uso del color, unido a la estructuración del contenido, hace este manual un modelo a imitar por otros fabricantes.

La cinta de demostración ha sido también muy cuidada, con ex-

plicaciones sobre el uso y las distintas funciones del teclado que facilitan al usuario el aprendizaje. En su cara B incluye dos juegos para que veamos parte de las posibilidades de la máquina.

Y el principal defecto del Plus es su color. Aclaremos: no nos quejamos de que se negro, sino de que la estética del aparato haya hecho al diseñador imprimir todos los nombres del aparato en blanco, lo que hace muy difícil distinguir los *tokens* normales de los de modo extendido, y éstos de los que requieren SIMBOL SHIFT. A los usuarios acostumbrados al Spectrum les costará un tiempo familiarizarse con el nuevo teclado.

En cualquier caso, esperamos ayudarlos a conocer cuánto puede dar de sí este paquete de ampliación y si os decidís a adquirirlo, las dificultades que os brindará su puesta en funcionamiento. En la secuencia fotográfica se muestra el montaje paso a paso. ★



## 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

# Todospectrum

# SPECTRUM COMPUTING

DOS GRANDES JUEGOS EN CODIGO  
CON MAQUINA OPCION DE JOYSTICK

## JUEGOS

**Chopper** PILOTANDO UN HELICOPTERO, TENDRA QUE DIRIGIR EL EQUIPO DE RESCATE PARA SALVAR A LOS ABANDONADOS EN UN CAMPO PETROLIFERO, DE UNA MUERTE SEGURA.

**Convoy** DESTRUIR LOS ALIENIGENAS Y SUS NAVES ES LA UNICA FORMA DE SALVAR LA TIERRA DE LA INVASION.

## ARTICULOS

**Twiddler** MUESTRA LOS MISTERIOS DEL RAPIDO CAMBIO DE COLOR.

**Cartoon** APRENDA A PROGRAMAR DIBUJOS ANIMADOS.

**Hangout** CONOZCA MAS A FONDO LAS POSIBILIDADES DE SU ORDENADOR.



BIENVENIDO A

**SPECTRUM**

**COMPUTING**

**CHOPPER  
TWIDDLER  
SHOOT  
HANGOUT  
TOMATOES  
CARTOON  
CONVEYOR  
TALLER  
CONVOY  
LIGHTBIKE  
LA TUMBA DE ELLAK**

**875**  
PTAS.

y mucho más

**SORTEO  
ESPECIAL**

MAS DE  
150.000 PTAS. EN  
PREMIOS BASES EN  
EL INTERIOR

*¡Ya está a la venta!  
¡Comprala en su quiosco  
habitual!*

Solicítela a: INFODIS C/ Bravo Murillo, 377 - 5.ª A. - 28020 MADRID

**CUPON DE PEDIDO**

Si enviemme el precio de 875 ptas.  ejls de SPECTRUM COMPUTING  
El importe lo abonare:  Contra reembolso  Con mi tarjeta de crédito   
Adjunto cheque  American Express  Visa  Interbank   
Número de mi tarjeta   
Fecha de caducidad   
NOMBRE   
DIRECCION   
CIUDAD  PROVINCIA   
Sin gastos de envío

# Plotter para

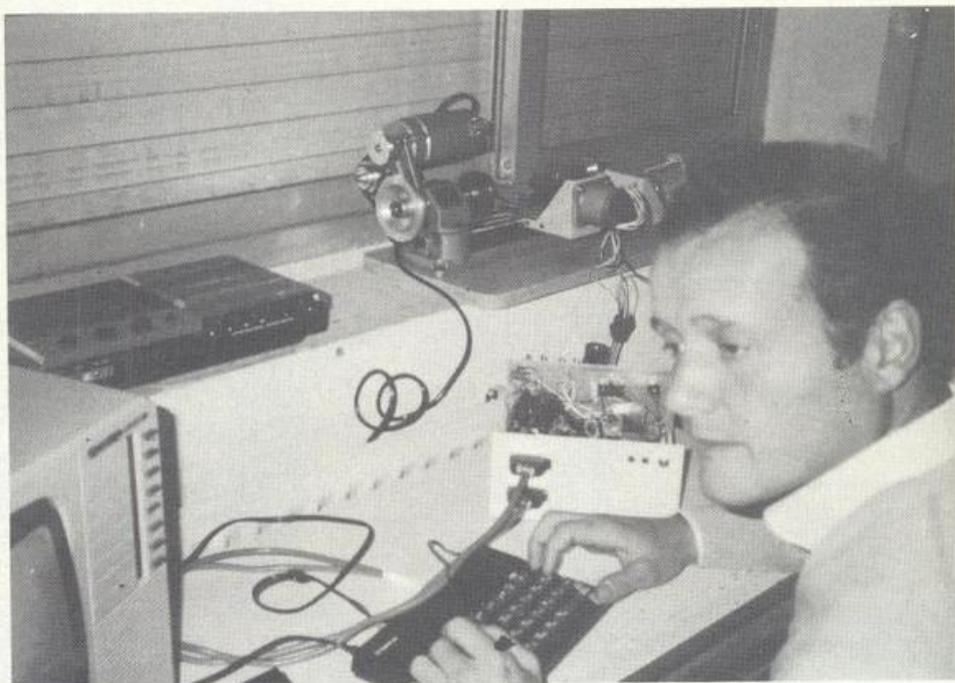
Efectivamente, en el Instituto Politécnico de formación profesional de Sevilla se es consciente de la falta de aprovechamiento del Spectrum en su vertiente de *hardware*.

Un plotter y un pequeño torno son los primeros mecanismos manejados por el Spectrum, al que seguirá en breve, un pequeño robot capaz de cortar chapas.

**T**ras largas esperas burocráticas, el Instituto recibía en marzo del año pasado la ayuda de 250.000 pesetas de la Junta de Andalucía con cargo a los fondos destinados para Proyectos de Innovación y Experimentación en centros docentes no universitarios. Los sueños de José Luna, Lorenzo Morbán y José Guzmán, profesores de matemáticas, metal y electrónica, respectivamente, por fin se hacían realidad, y con ellos los de decenas de estudiantes que corrían por apuntarse a los cursos voluntarios.

En una pequeña sala (los grupos de estudiantes no pueden sobrepasar la docena) se encuentran el enorme tablero de dibujo, el torno y el Spectrum en medio de una multitud de cables y enchufes. "Estos cables deberían sustituirse por los de circuito impreso de ocho hilos, pero no los hemos podido encontrar aquí, en Sevilla", nos comentaba José Guzmán. "De todas formas estas conexiones aparentemente provisionales sobre cinta aislante son buenas. Si el alumno se encuentra con una caja cerrada, con aparatos sofisticados, se produce un rechazo psicológico. De esta forma está todo más a la vista y los chicos ven que ellos también pueden hacerlo."

La ayuda fue inferior a lo solicitado, pero suficiente para la adquisición de los equipos, y para empezar a trabajar en lo que ellos denominan el "bricolage industrial", consistente en modificar un equipo ya existente en otro controlado por ordenador. El primero en su-



## LA INFORMATICA C

De todos es conocida la preocupación oficial por el tema de la informática en las escuelas. Pero no menos conocidos son sus problemas reales de implementación. Recientemente tuvieron lugar en Madrid las Jornadas sobre informática y educación en la enseñanza básica y media, ampliamente desbordadas por profesores de todos los ámbitos y regiones de España, que quieren dar respuesta al Proyecto Atenea (plan gubernamental con el que el ministerio de Educación y Ciencia se propone

introducir los ordenadores en la enseñanza). Mientras esto se lleva a cabo, las necesidades de información y formación en materia informática se vienen llevando a cabo a través de las EATP o Enseñanzas de Actividades Técnico-Profesionales, complementadas por distintos tipos de ayudas de los organismos locales. Este es el caso de la Junta de Andalucía. De acuerdo con las declaraciones de un portavoz de la misma, durante el curso 84-85 fueron aprobados 17 proyectos para la zona de Andalucía, dentro de las

# el Spectrum?

frir esta transformación fue el tablero de dibujo. "En el soporte de las reglas se ha colocado otro con rotuladores, unos motores y, en definitiva, se ha transformado un antiguo tablero de dibujo en un plotter. El sistema es sencillo. Lleva tres circuitos integrados, uno que es el *port* propiamente dicho y dos de decodificación de direcciones. Se completa con otros dos circuitos integrados de control de motores, un bloque de transistores de potencias para la salida y la fuente de alimentación. No hemos empleado fotoacopladores porque lleva un diseño en el que la única

zona común entre potencia y el Spectrum es la mesa. En cuanto al funcionamiento, se crea un *port*, el control y la fuente de potencia. Se coloca una máscara en un *port* de salida y según la señales que se le dé, se activan o desactivan los relés, se le da un avance al motor en un sentido u otro o se le da la dirección de rotación del motor."

Es un poco lento, pero funciona. El único problema es que el ordenador se "resetea" como nos decía lacónicamente José Guzmán. "Ocurre al desconectarlo a través de la fuente de alimentación de la mesa. Tendremos que colocar un

estabilizador mejor al Spectrum. En cuanto a la velocidad, estamos estudiando la colocación de ordenadores más potentes para realizar los cálculos y usar el Spectrum como esclavo para el control directo de los motores. Si el Spectrum tiene que hacerlo todo, aunque sea en ensablador, no va a poder hacerlo a una velocidad aceptable."

Después del plotter vino el torno. En el centro ya había uno, sólo lo cambiaron un poco: "se sustituyeron los dos mandos tradicionales por dos motores que maneja el ordenador. Un pequeño programa indica al ordenador el perfil de la pieza y mueve el carro portaherramientas para realizar este perfil. La precisión es importante y tiene cinco centésimas de diferencia paso a paso."

Un solo Spectrum y 250.000 pesetas para material no parecen ser de gran ayuda, si se piensa que sólo el torno vale las 100.000 pesetas y que el número de estudiantes del instituto se eleva a 1.700 alumnos. Pero ha servido en este caso para que unos jóvenes profesores con ideas lleven adelante su proyecto, que tan buena acogida ha tenido por los estudiantes. ★

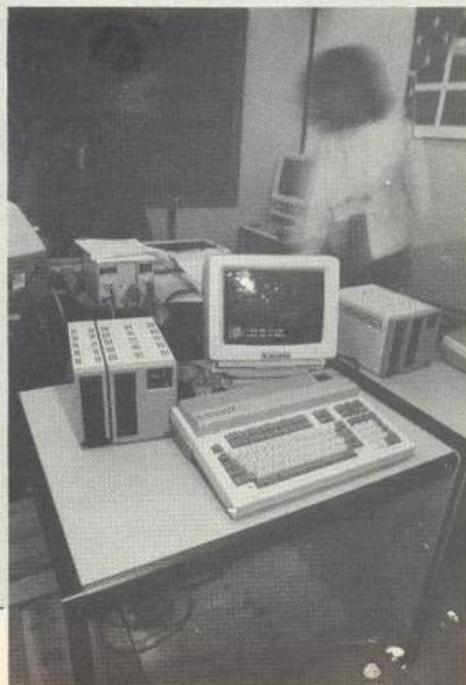


## COMO ASIGNATURA

ayudas a proyectos de innovación y experimentación en centros docentes no universitarios. De estos proyectos sólo ocho contaron con una subvención económica que ascendió a 1.755.000 pesetas, quedando el resto pendiente de dotación.

Por su parte, la Consejería de Cultura tiene muy avanzados los programas para la implantación de la informática como asignatura, que se desarrollan en tres niveles: formación del profesorado existente, implantación de la propia enseñanza de informática y, finalmen-

te, la aplicación de los medios informáticos que de momento, no se considera urgente. En este último sentido, no se ha determinado la elección de marcas concretas de ordenadores, sino tan sólo el número inicial de equipos con el que se llevaría a cabo la experiencia que, de obtener resultados positivos, se ampliaría para ofrecer una cobertura total y poder contar finalmente con la informática como una asignatura reglada en el ámbito de las tecnologías, tanto en EGB como en las enseñanzas medias.



## SPECTRUM COMPUTING

Distribuidor: Compulogical.  
Formato: *cassette*.  
Memoria requerida: 16/48 K.  
Precio: 875 ptas.

Con carácter bimensual, Spectrum Computing es en realidad una "revista en *cassette*". En esta ocasión nos ha llamado la atención el juego "Bitman" con el que abre el número 2.

Junto a este programa, que se trata de una buena versión del conocido comecocos en código máquina, es de destacar el artículo "Scroller", rutina localizable para el desplazamiento *pixel a pixel* y el de "64 columnas", antes rutina en código máquina con posibilidad de ser utilizada por el usuario en sus propios programas. A estos artículos les acompañan diversos juegos, algunos muy completos como el "Randit" (máquina tragaperras) y otros más clásicos como el poker. En general una buena selección, aunque el tiempo entre programa y programa no sea lo suficientemente dilatado y los mensajes de pantalla no siempre sean claros.

Hablábamos del primer programa, el "Bitman" o "Comecocos", el gran plato fuerte con el que se inicia este número 2. Enteramente realizado en código máquina, las instrucciones y mensajes se encuentran en inglés, aunque ello no es demasiado problema dada la simplicidad de su funcionamiento: basta con pulsar las teclas de monto del cursor para producir los desplazamientos del comecocos. En la carrera, y este es el mejor detalle que podemos mencionar de este juego, el comecocos come a una velocidad que podemos calificar de "endiablada". A su paso podrá comer los puntos que encuentre y con ello conseguir mayor puntua-

El laberinto es siempre el mismo... al igual que la velocidad de los "cocos" azules.

ción. Y no sólo come él, ya que cuatro *bugs* o fantasmas le persiguen implacablemente a una velocidad nada despreciable. En cada una de las cuatro esquinas existe una "fruta" especial con la que podemos comer a sus perseguidores y aumentar notablemente su puntuación..., pero sólo por breves

**Control:** teclado.

**Jugadores:** uno.

**Gráficos:** muy aceptables, con sensación de relieve y colores cambiantes según el juego.

**Sonido:** bueno.

**Nivel de dificultad:** no dispone de niveles de dificultad.

**Originalidad:** está basado en el video-juego conocido, no incorpora ningún rasgo propio.

**Conclusión:** es un juego difícil al principio y que requiere de tiempo para dominar el movimiento del comecocos. Es una lástima que no pueda jugarse con *joystick*.



momentos, pues su efecto dura escasos segundos.

Una vez devorados todos los puntos se obtiene una nueva pantalla... para no quedarse con hambre. La parte inferior de la pantalla informa continuamente de la puntuación del jugador, máxima puntuación, comecocos que quedan (se parte de 3) y número de pantalla.

La presentación y despedida se acompaña de una bonita música. También se produce una música especial en el momento en que se come una fruta, además del cambio de color de los fantasmas, y en el momento en que el comecocos es comido, que también aparece una cruz en el lugar del "genocidio".

El trazado del recorrido está bien conseguido y permite gran movilidad, especialmente en los laterales, pero se echa en falta alguna salida por los extremos de la pantalla, como ocurre con los video-juegos clásicos de este tipo. También hubiera sido deseable contar con varios niveles de juego, ya que la velocidad puede resultar excesiva para los novatos.

La adicción y la competición están garantizadas, conservándose la máxima puntuación de los diez mejores jugadores.

## HUNTER KILLER

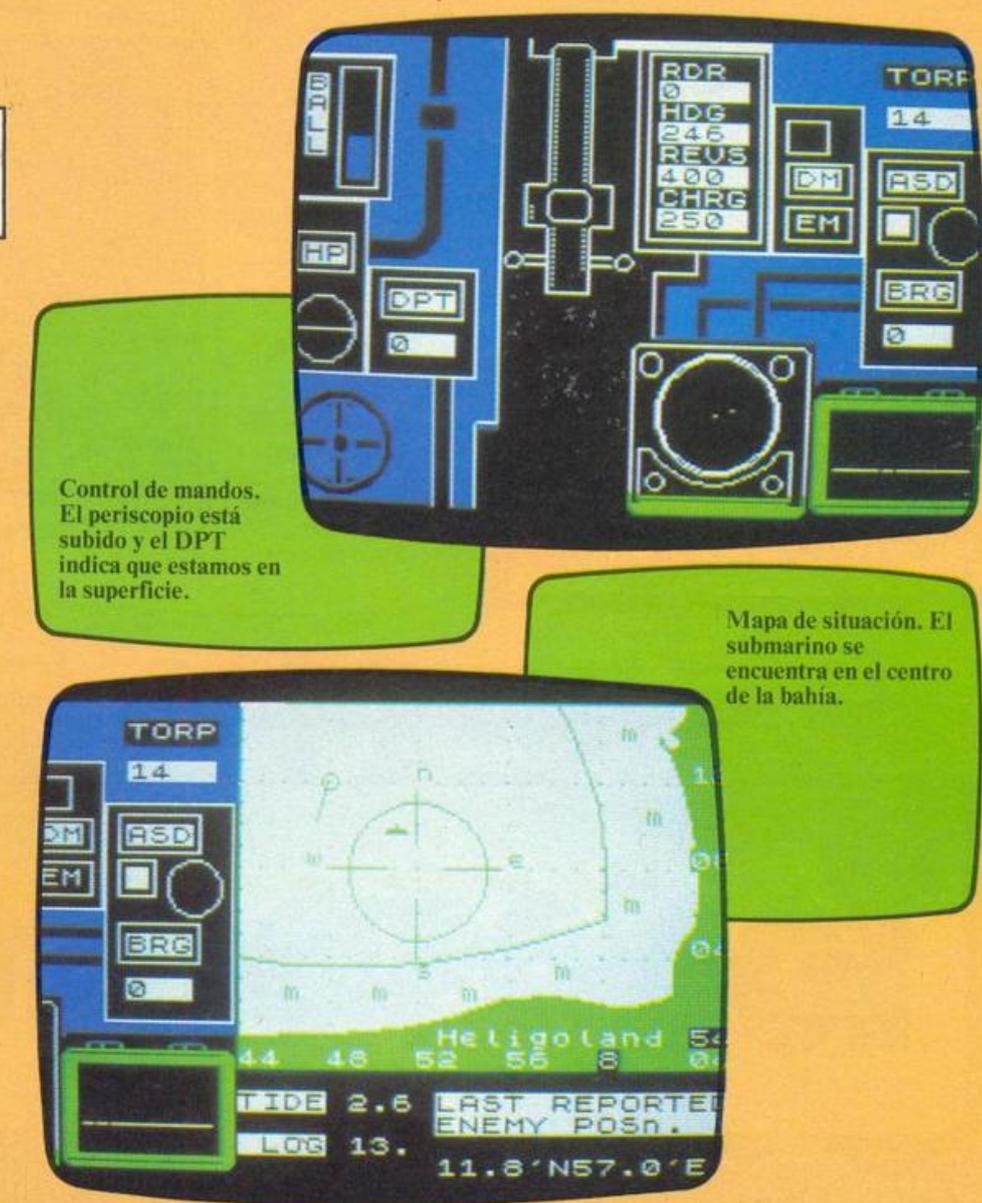
Distribuidor: ABC Soft.  
Formato: *Cassette*.  
Memoria requerida: 48 K.  
Precio: 2.250 ptas.

A primera vista parece un "juego de barcos". Nada más lejos de la realidad. Navegando en el submarino "S" de la Royal Navy (famoso por su reducida dimensión y su decisiva actuación en la segunda guerra mundial), su objetivo es destruir al submarino enemigo. Y para hacer más difícil su misión, no falta el ataque aéreo (que le obligará a sumergirse rápidamente), los destructores (mejor abandonar su zona) y las minas cercanas a la costa (no aguantan el más mínimo rozamiento).

Tiene dos modalidades de funcionamiento: contra el ordenador o para dos jugadores a través de la red del *interface 1*. Para un jugador el enemigo está siempre en la superficie. Para dos jugadores, además de utilizar la red con sus correspondientes Spectrum y pantallas, presenta algunas diferencias: no hay ataque aéreo y las baterías se encuentran parcialmente cargadas, lo que les obligará a subir a la superficie.

El programa presenta tres pantallas: control de mando, mapa de situación y periscopio.

El control de mandos informa de la situación del submarino (profundidad, motores utilizados, número de torpedos, etc.) donde destaca el periscopio que, lógicamente, puede elevarse en cualquier momento. Parte de esta información —el lateral derecho— se conserva al pasar al mapa de situación. Dicho mapa corresponde a la costa de Helegoland en Alemania, donde se desarrolla la batalla. En el mapa se puede ver la situación



del submarino y su proximidad a la costa. Es la tercera pantalla, la vista a través del periscopio, la más atractiva. Aunque el objetivo no esté acorde con la vistosidad general del juego (apenas se puede distinguir el submarino enemigo), el nivel del agua está bien conseguido, variando según la profundidad, así como el avión enemigo. El control del periscopio es total, siempre

que no esté por debajo de los 38 pies, pudiéndose mover en un ángulo de 360 grados.

Como aspecto negativo, es de destacar el impacto del torpedo, apenas imperceptible y el hecho de que todo el teclado permanezca bloqueado hasta que se produce el impacto: intentamos lanzar el torpedo y sumergirnos rápidamente... ¡y nos quedamos con las ganas! ★

# Juegos

**Control:** joystick, teclado.

**Jugadores:** 1 ó 2.

**Gráficos:** tres pantallas con gráficos nada espectaculares.

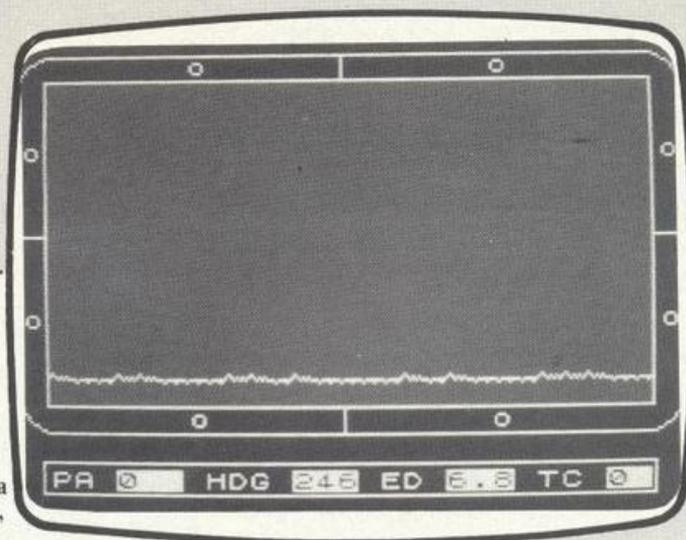
**Sonido:** bastante deficiente, salvo en el aviso de emergencia o hundimiento.

**Nivel de dificultad:** cinco niveles.

**Originalidad/conclusión:**

Bastante novedoso, al luchar bajo el agua, frente a las clásicas luchas galácticas.

Visión a través del periscopio. La línea inferior informa de la distancia al enemigo, ángulo de giro, torpedos lanzados...



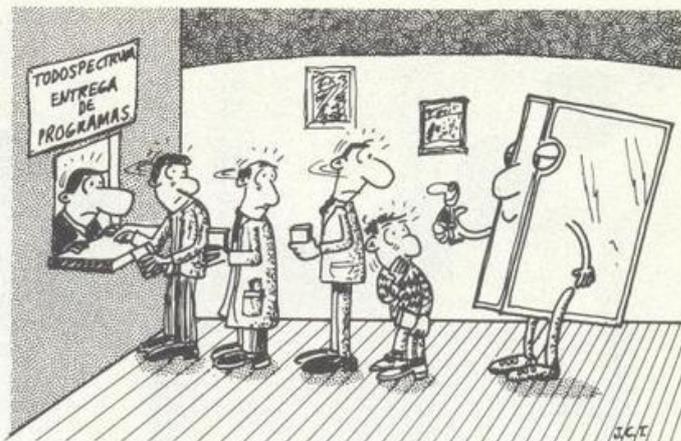
# Todospectrum

**ANUNCIESE  
por  
MODULOS**

**MADRID  
(91) 733 96 62  
BARCELONA  
(93) 301 47 00**

# Desplazamiento

# Pixel a Pixel



“Este es su turno”, se decía en un artículo aparecido en el número 2 de TODOSPECTRUM bajo el título de Desplazamiento Pixel a Pixel. Varios lectores nos han escrito para contarnos cómo veían los programas que ahora mostramos como 1 y 2.

Alfredo Llorente nos explica para qué sirve el programa:

“Como pretendía ser interesante me dispuse a teclearlo. No fue tarea fácil y cuando terminé necesitaba una espalda nueva. Pero el esfuerzo valió la pena, pues el programa en código máquina posibilitaba el empleo de un SPRITE en el Spectrum. Y de un auténtico SPRITE, me explicaré, un SPRITE permite mover por la pantalla un bloque de caracteres...”

Un SPRITE es un caracter o conjunto de caracteres —normales o definidos por el usuario (UDG)—, que puede imprimirse en cualquier lugar de la pantalla mediante una sencilla orden. Algunos ordenadores implementan de entrada esta posibilidad, pero no es el caso del Spectrum. Como

nos comentaba en su carta Gabriel Ortas, otro de los lectores que nos escribió, la rutina en código máquina “permitía escribir un caracter en cualquier lugar de la pantalla disponiendo 256 por 176 posiciones en vez de las 22 por 32 convencionales”.

De su mano se nos invita a comprender el funcionamiento del programa:

“Con el programa 1 se carga el código máquina desde la dirección 64600 a la 65099. El programa lleva incluido un *checksum* por lo que no le ha de preocupar demasiado si comete algún error al introducir las DATAS ya que aparecería el mensaje DATA ERROR; en caso contrario la rutina en código máquina se grabará automáticamente con el nombre MOVERAREA. Encienda su *cassette* para grabar y pulse la tecla. No se olvide de poner en MIC los cables de interconexión.

Pará que la rutina funcione ha de introducir diversos datos relativos al movimiento (derecha, izquierda, arriba o abajo), velocidad,

etc. He aquí la lista de las direcciones de memoria en las que hay que introducir algún dato:

65188. En esta dirección hay que introducir el número de pausa.

65359 y 35360. En estas direcciones hemos de colocar las coordenadas, fila y columna, respectivamente, del caracter superior izquierdo del bloque que queremos mover. Estas coordenadas coinciden con las del comando AT (fila, columna) del BASIC.

65361 y 65363. En estas dos direcciones se colocan los números de caracteres que miden la altura y la base, respectivamente, del bloque rectangular que se va a desplazar.

Desde la 64512 a la 64517. En estas direcciones se le indica al ordenador las direcciones y los movimientos que se efectuarán. Los números que se colocarán en estas posiciones han de seguir la fórmula siguiente:  $n1 \times n2 + n3$ , en la que:

—  $n1$  es el 32 si lo que se quiere es indicar la longitud del desplaza-

## PROGRAMA 1

```

1 CLEAR 50000
2 LET T=0
3 FOR I=64600 TO 65099
4 READ X: POKE I,X: LET T=T+X
5 NEXT I
6 IF T(<)59504 THEN PRINT "DAT
A ERROR"
7 IF T(<)59504 THEN STOP
9 SAVE "MOVERAREA" CODE 64600,5
00
10 PRINT "ahora MOVERAREA"
11 PRINT "Grabado en cinta."
12 STOP
13 DATA 243,205,95,252,251
14 DATA 201,8,205,44,254
15 DATA 24,3,24,242,255
16 DATA 1,8,8,197,42
17 DATA 75,255,126,230,252
18 DATA 32,2,193,201,125
19 DATA 20,165,254,205,171
20 DATA 192,58,165,205,230
21 DATA 202,16,188,201,61
22 DATA 202,16,253,201,202
23 DATA 202,16,202,202,210
24 DATA 203,58,165,204,230
25 DATA 240,32,10,42,75

```

```

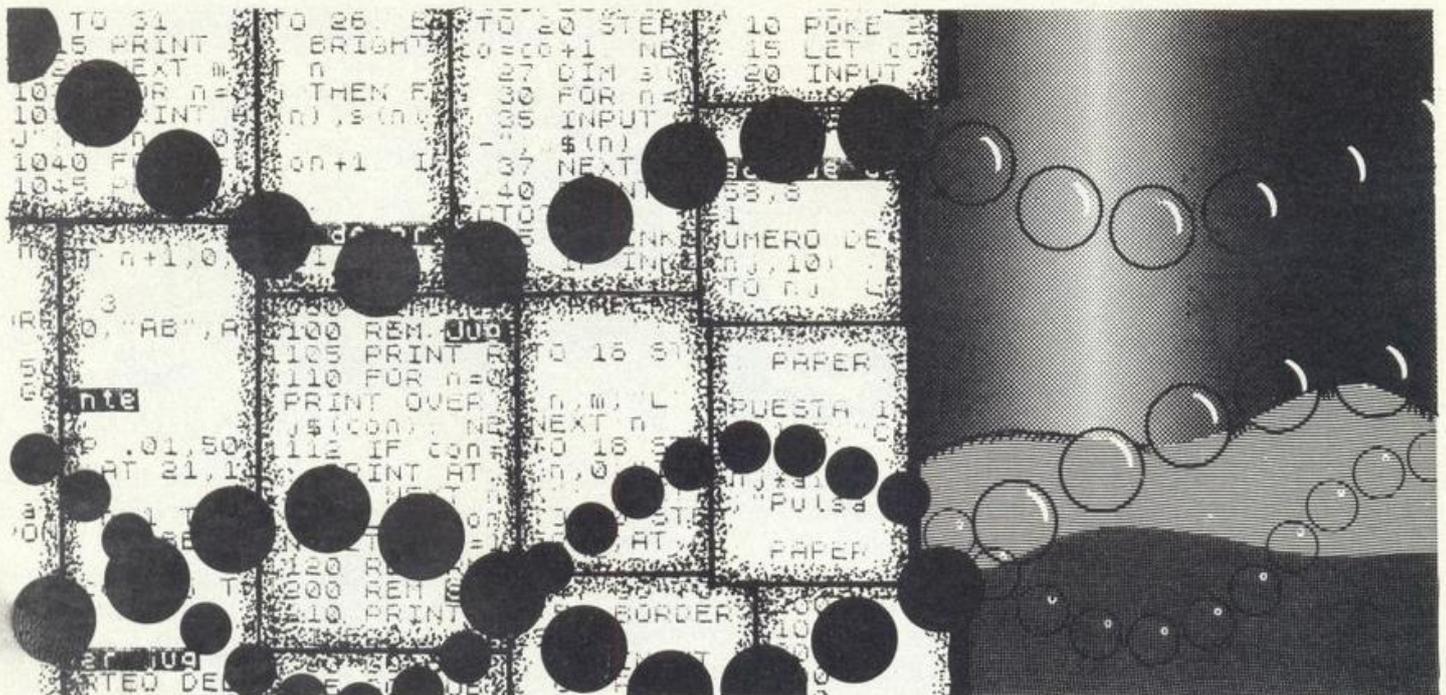
35 DATA 255,35,237,99,75
36 DATA 255,24,202,58,165
40 DATA 254,214,4,58,165
42 DATA 254,24,205,58,164
44 DATA 254,71,197,71,197
46 DATA 71,16,254,193,16
50 DATA 249,193,16,203,16
52 DATA 60,8,8,203,16
54 DATA 60,77,255,122,128
56 DATA 71,197,203,214,255
58 DATA 95,217,203,214,1
60 DATA 83,255,198,1,71
62 DATA 201,203,108,214,48
64 DATA 4,201,203,203,198
66 DATA 221,36,16,242,193
68 DATA 16,2,24,14,200
70 DATA 123,198,32,95,48
72 DATA 4,122,24,108,6,57
74 DATA 213,24,203,201,193
76 DATA 16,191,193,16,90
78 DATA 58,77,255,61,58
80 DATA 77,255,5,8,167
82 DATA 195,144,252,255
84 DATA 197,5,42,252,58
86 DATA 124,16,77,255,58
88 DATA 81,255,58,1,71
90 DATA 81,255,58,198,1,71
92 DATA 22,8,90,22,8

```

```

94 DATA 203,62,48,2,22
100 DATA 128,126,131,119,35
102 DATA 16,241,193,16,90
104 DATA 24,14,225,125,198
106 DATA 32,8,48,4,124
108 DATA 165,111,48,229,24
110 DATA 213,225,193,16,198
112 DATA 193,4,62,16,144
114 DATA 60,58,77,255,58
116 DATA 60,77,255,144,5
118 DATA 80,80,197,255,32
120 DATA 203,197,17,255,6
122 DATA 81,255,60,71,4
124 DATA 214,32,95,48,4
126 DATA 122,214,8,87,24
128 DATA 10,123,198,32,95
130 DATA 48,4,122,198,8
132 DATA 87,213,167,122,246
134 DATA 71,103,123,214,214
136 DATA 111,48,4,124,214
138 DATA 8,103,6,8,197
140 DATA 213,229,58,83,255
142 DATA 71,25,119,28,44
144 DATA 16,250,225,209,107
146 DATA 68,20,103,16,203
148 DATA 193,209,16,203,58
150 DATA 83,255,71,54,60
152 DATA 44,16,251,193,121
154 DATA 61,79,32,20,237

```



miento de caracteres; o 4 si se indica en *pixels*.

— *n2* será el número de caracteres o *pixels* que queremos que se desplace el bloque.

— *n3* indica la dirección de desplazamiento y será:

- 0 si es a la izquierda.
- 1 si es a la derecha.
- 2 si es hacia arriba.
- 3 si es hacia abajo.

Después de haber indicado todos los movimientos que se desearía que se produjesen, en la siguiente dirección pondríamos un 0, regresando la rutina al BASIC. De esta forma si en la dirección 64512 hay un 82 y en la 64513 un cero, el movimiento será hacia arriba 20 *pixels* es decir, dos caracteres y medio, ya que  $82 = 4 \times 20 + 2$ .

Una vez introducidos estos datos, ya sólo tendrá que llamar a la rutina situada a partir de la dirección 64612 de la siguiente manera:

```
LET M=USR 64612.
```

El programa 2 ilustra claramente el funcionamiento de esta rutina. Seguro que antes de verlo funcionar ya habrá adivinado lo que va a pasar cuando pulse RUN".

Pero no piense que la colaboración de nuestros lectores acaba aquí, nada más lejos de la realidad. Alvaro Mateos nos manda una mejora del programa en código máquina que le dota de una mayor potencia, permitiendo que los SPRITES tengan atributos diferentes de los de la pantalla. Reservamos para otro número la completa descripción de este programa.

Por su parte, el ya mencionado Luis Vicente, nos decía:

"Como dicen que una imagen vale más que mil palabras, les mando un pequeño programa que utiliza la misma rutina de código máquina para desarrollar una vieja idea de Julio Verne. Se puede mejorar el movimiento logrando mayor continuidad si reducimos la amplitud de los movimientos y aumentamos la velocidad, pero eso lo dejo para entretenimiento del curioso lector."

A todos ellos, nuestro agradecimiento y el premio prometido. ✨

Alfredo Llorente  
Gabriel Ortos  
Alvaro Mateos  
Luis Vicente

```

156 DATA 91,77,255,123,214
158 DATA 32,95,48,4,122
160 DATA 214,8,87,237,183
162 DATA 77,255,14,8,107
164 DATA 195,144,252,2,107
166 DATA 77,255,58,83,255
168 DATA 71,197,24,2,107
170 DATA 19,213,38,8,188
172 DATA 31,255,71,4,107
174 DATA 34,11,197,4,123,198
176 DATA 32,95,48,4,123,198
178 DATA 198,8,87,122,230
180 DATA 248,8,7,6,24
182 DATA 1,20,108,26,103
184 DATA 125,18,16,248,193
186 DATA 18,228,209,193,18
188 DATA 16,193,121,80,79
190 DATA 2014,16,32,20,237
192 DATA 31,77,255,20,123,198
194 DATA 32,95,48,4,123,198
196 DATA 198,8,87,4,237,183
198 DATA 77,255,58,14,255,197
200 DATA 195,144,252,2,308
202 DATA 77,255,203,39,2,308
204 DATA 39,39,203,39,2,308
206 DATA 134,58,77,255,58
208 DATA 60,255,77,24,198
210 DATA 64,58,78,238,24,198
212 DATA 64,58,78,255,201
PROGRAMA 2
1 CLEAR 50000: LOAD ""CODE
2 GO TO 200
3 REM -----
4 POKE 65355,0: POKE 65355,25
5 POKE 65188,DELAY
6 POKE 65361,HEIGHT
7 POKE 65363,LENGTH
8 POKE 65360,Y
9 POKE 65359,X
10 LET M=USR 64612
11 RETURN
12 REM -----
13 REM Secuencia movimiento
14 REM -----
15
17 POKE 64512,20+4: REM izquie
18 POKE 64513,20+4+3: REM abajo
19 POKE 64514,20+4+1: REM dere
20 POKE 64515,20+4+2: REM arri
21 POKE 64516,0
22 RETURN
23 REM -----
25 POKE 64512,5+32+3: REM abaj
26 POKE 64513,3+32: REM izquie
27 POKE 64514,6+32+1: REM dere
28 POKE 64515,3+32: REM izquie
29 POKE 64516,5+32+2: REM arri
30 POKE 64517,0
31 RETURN
32 REM -----
33 REM Adicionales secuencias
34 REM de movimiento
35 REM -----
36 BORDER 5
37 PRINT AT 10,10: ""
38 LET LENGTH=4: LET HEIGHT=4
39 LET Y=10: LET X=8
40 LET DELAY=20
41 GO SUB 17: GO SUB 4
42 REM -----
43 PRUZE 50: CLS: LIST 201
44 LET LENGTH=18: LET HEIGHT=5
45 LET Y=7: LET X=10
46 LET DELAY=1
47 GO SUB 25: GO SUB 4
48 PRUZE 30: GO TO 214

```

# Todospectrum

REVISTA EXCLUSIVA PARA USUARIOS

## OFERTA DE SUSCRIPCION

Te ofrece la posibilidad de suscribirte con unas condiciones muy ventajosas para ti:

- 1** Recibir puntualmente, durante 12 meses, en tu domicilio, la revista mensual con mayor difusión para usuarios de ordenadores Sinclair.
- 2** Consigue un práctico regalo:



Una obra fundamental en la biblioteca de los aficionados al Spectrum:

- Reglas y herramientas del Basic.
- La técnica de los organigramas.
- El fabuloso mundo de las rutinas.
- Variables y cadenas.
- Funciones matemáticas usuales.

Esta publicación, escrita con estilo ameno y práctico, te ayudará a sacar todo el partido posible a tu máquina.

- 3** La opción de ser protagonista. Tú puedes tener una participación directa con tus comentarios, programas, sugerencias, etc.
- 4** Obtener premios importantes con tus programas, y temas de interés.

## EN DEFINITIVA, TODO SON VENTAJAS

No dejes pasar esta oportunidad, suscríbete a "TODOSPECTRUM", cumplimentando hoy mismo el cupón de respuesta adjunto.

# Todospectrum

BRAVO MURILLO, 377 - 5.º A  
TELEFONO: 733 74 13/47/63/97 28020 MADRID

ESPECIAL LIBROS

# CODIGO MAQUINA



Entre los usuarios de Spectrum y ZX81, son cada vez más las personas que no quedan satisfechas con el uso del BASIC, debido a su lentitud y a sus limitaciones. Entre los que deciden aprender código máquina, una buena parte se desaniman al intentar adquirir un buen libro con el que guiar su aprendizaje. Se trata de elegir entre la selva de material impreso una buena obra, adecuada a sus conocimientos, barata, y pedagógica...

Pero nos tememos que esos chollos son difíciles de encontrar, al menos en castellano. He aquí otro problema: los lectores que no lean inglés tendrán serias dificultades para utilizar la bibliografía sobre este tema, aunque las traducciones son cada vez más numerosas y de mejor calidad.

Con el fin de orientarles en su búsqueda. Publicamos a continuación una guía de los libros de código máquina Z80 más representativos, en especial aquellos dirigidos

## LO QUE USTED DEBE LEER PARA EMPEZAR A SABER

a la programación de ZX81 y Spectrum. Dividimos arbitrariamente los libros en cuatro categorías:

- Para principiantes. **1**
- Nivel avanzado. **2**
- Manuales de referencia del microprocesador Z80. **3**
- Ayudas a la programación para Spectrum. **4**

Las dos primeras categorías incluyen aquellos libros destinados al aprendizaje de personas que no tengan grandes conocimientos de código máquina o de la arquitectura del Spectrum. La presentación se hace de manera pedagógica y las

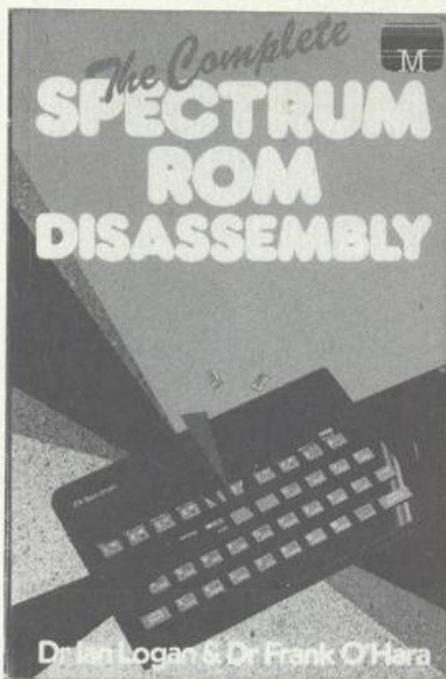
distintas secciones se introducen dosificando la dificultad.

En el tercer grupo hemos incluido libros que no tienen en cuenta específicamente el diseño del Spectrum o del ZX81, sino que nos describen de una forma general la programación de sistemas basados en el microprocesador Z80 y su familia de circuitos auxiliares. Son muy adecuados para personas que ya posean conocimientos a este nivel y que deseen un manual de referencia del procesador. Imprescindibles para los "artistas" del soldador.

El cuarto nivel incluye aquellos libros cuyo objeto principal es, bien proporcionar subrutinas básicas que nos ayudan a programar en código máquina, bien a conocer mejor la ROM del Spectrum, para utilizar sus puntos de entrada en nuestros programas o para conocer el sistema operativo de nuestra máquina. Muy útiles para el programador experimentado.

**4** The Complete Spectrum ROM disassembly  
Autor: Ian Logan  
Editorial: Melbourne House, 1982  
236 páginas

El Spectrum con los "sesos" al aire. Si en otros libros se nos muestra el *hardware* del Spectrum; en este es el sistema operativo el que se nos "desnuda" sin ninguna vergüenza. Es un libro ya veterano, que se puede encontrar en las estanterías de los que conocen las interioridades del Spectrum "de memoria".



No se trata de un libro de aprendizaje. Es, en cambio, un manual de uso imprescindible para comprender el sistema operativo, para saber por qué se nos "cuelga" el ordenador en determinados momentos.

El principal defecto de este libro es que no se hace referencia a la ROM del Interface I, ya que no existía en el momento de la edición del libro. Por tanto, el uso de las rutinas del Interface I nos exigirá la compra de un libro aparte.

Un buen libro aconsejable para aquellos que ya posean unos mínimos conocimientos sobre el código máquina.

**3** The programming guide to the Z80 chip  
Autor: Phillip R. Robinson  
345 páginas  
Editorial: Tab Books, 1984.  
Z80 Assembly language programming  
Autor: Lance A. Leventhal  
Editorial: Osborne/McGraw Hill, 1979

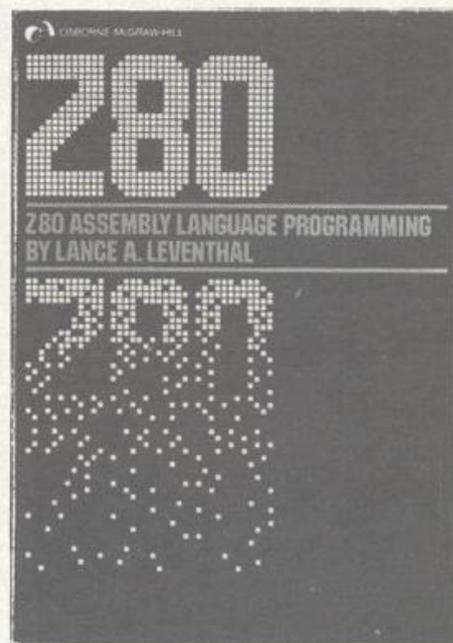
Para el microprocesador, un par de libros de utilidad en dos casos: Si queremos aprender a programar en código máquina Z80 sin importarnos la arquitectura concreta de la máquina para la que vamos a



trabajar, o bien si queremos utilizar las posibilidades *hardware* del Spectrum para realizar montajes, añadir periféricos, etcétera.

Ambas obras cumplen ambos objetivos. Se nos enseña el juego de instrucciones del chip, los periféricos de la "familia", diseñados específicamente por el fabricante para utilizarlos con el Z80 (PIO, SIO, etc.). Incluyen además explicaciones sobre el uso de los programas ensambladores, así como técnicas de programación en "assembler".

Si bien los dos volúmenes están más allá de lo que un aficionado medio está dispuesto a estudiar, pueden resultar útiles si utilizamos el Spectrum como un banco de



aprendizaje para el estudio y programación de microordenadores basados en el mismo procesador.

**4** Las 40 mejores subrutinas en código máquina para el Spectrum  
Autor: John Hardman y Andrew Hewson  
Editorial: Indescomp, 1984  
162 páginas  
Precio: 1.950 ptas.

Una obra encuadrada entre aquellas que no están escritas con la idea de enseñarnos lenguaje máquina, sino de facilitarnos la programación en ese idioma. A pesar de ello, se nos ofrece una primera sección en la que se repasa el juego de instrucciones del microprocesador y el mapa de memoria del Spectrum de una manera esquemática.

Se pasa a continuación a proponernos 40 subrutinas en código máquina, con unas explicaciones relativamente detalladas de su funcionamiento. El listado viene dado en ensamblador y en decimal, con información para permitir la relocalización de los programas, caso de ser éstas dependientes de la posición de memoria en que las situemos.

Las cuarenta subrutinas que nos promete el libro se completan con

una amplia colección de programas útiles para trabajar en BASIC. Con ellos podremos borrar bloques de programa, reenumerar líneas, compactar programas, cargar código máquina en sentencias DATA, buscar dentro de nuestro



programa una cadena de caracteres determinada, etcétera.

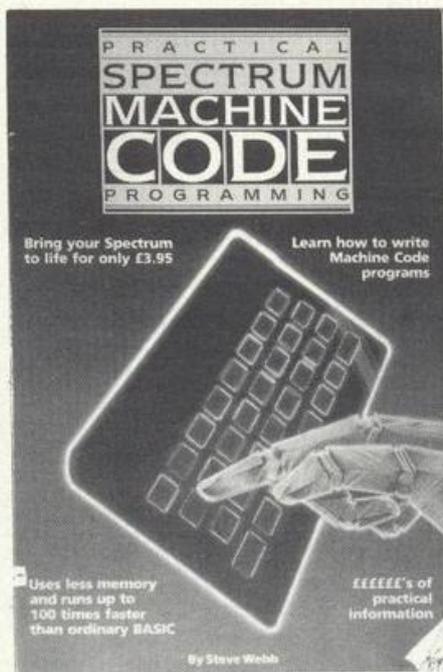
Salvando la traducción, que utiliza una terminología poco usual, este libro resultará útil para aquellos que quieran estudiar técnicas de programación en código máquina mediante unas rutinas interesantes, tanto por su contenido como por las técnicas de programación usadas.

**1** Practical Spectrum Machine Code Programming  
 Autor: Steve Webb  
 Editorial: Virgin Books, 1984  
 116 páginas

Un libro breve, para aquellos que no quieran ser tratados como ignorantes pero deseen un enfoque del tema "desde el principio".

Tras una corta introducción sobre lo que es el código máquina, donde se nos proporciona un cargador hexadecimal que nos facilitará el trabajo de las secciones posteriores, se explican las instrucciones de código máquina comparándolas con instrucciones BASIC que realicen cometidos parecidos.

Este volumen gustará a aquellos



que deseen ir viendo sus progresos con realizaciones prácticas. Su principal defecto es el escaso espacio dedicado al sistema operativo del Spectrum, que nos quita posibilidades de trabajo en aplicaciones más serias.

**1** Código Máquina, ZX Spectrum  
 Autor: Beam Software  
 Editorial: Indescomp, 1982.  
 243 páginas  
 Precio: 2.100 ptas.

Los distintos temas se introducen por preguntas del tipo. Eso está bien, pero ¿cómo haría yo...?, lo que hace que los problemas se aborden de una manera gradual. En primer lugar se nos presenta la

estructura de la CPU del Z80, hablándonos de los registros como las "manos y pies" con las que cuenta el ordenador.

La obra prosigue con una descripción detallada del juego de instrucciones del procesador, para continuar con una amplia discusión de las particularidades del Spectrum (teclado y pantalla).

Un buen libro, cuyo único defecto es la falta de explicaciones sobre la estructura de la ROM, lo



que impide la simplificación de muchas rutinas mediante llamadas al sistema operativo. Asimismo la traducción incluye algunos términos que pueden sonar extraños a los oídos de personas acostumbradas a programar.

**1** Spectrum. Introducción al código máquina. Cómo obtener más velocidad y potencia  
 Autor: Ian Sinclair  
 Editorial: Díaz de Santos, 1983  
 158 páginas  
 Precio: 1.450 ptas.

Para los principiantes. A un nivel muy sencillo y en castellano, esta obra nos explica los fundamentos del código máquina pedagógicamente y "desde el principio". Se nos enseñan los fundamentos de los microordenadores,

manejo de los caracteres como números, etc. El mapa de memoria de Spectrum y su forma de almacenar los programas, un estudio detallado del funcionamiento de la CPU, una descripción de las instrucciones más utilizadas del Z80, son los conceptos que, de una manera clara y muy asequible al profano, se estudian en los primeros capítulos. Se nos explica también la forma de depurar (limpiar de

nos explica los cambios de arquitectura entre las dos máquinas describiendo las modificaciones necesarias para adaptar los ejemplos del libro al Spectrum. Aunque el diseño de ambas máquinas guarda cierta semejanza, las variaciones son suficientes para no permitir una compatibilidad, siquiera parcial, sin "meterle mano" a los programas.

La obra nos muestra progresiva-

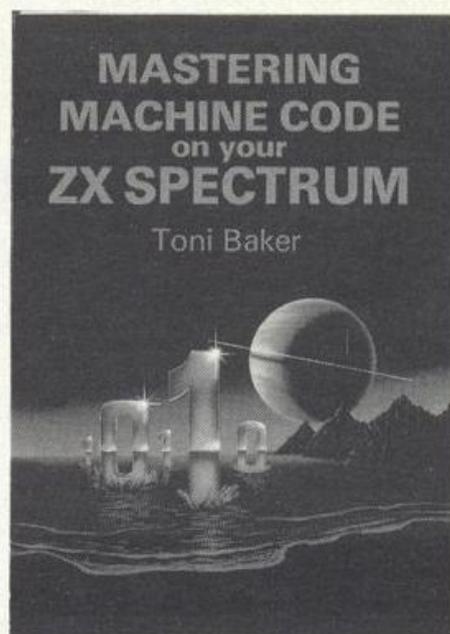
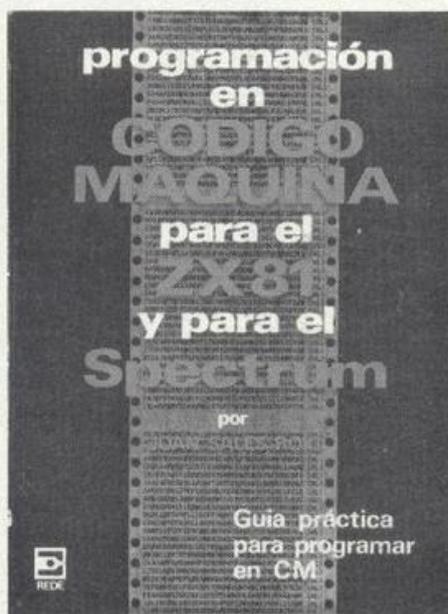
## 2 Mastering Machine Code on your ZX81

Autor: Toni Baker

Editorial: Interface Publications, 1981

186 páginas

Un libro imprescindible para todo programador en código máquina de ZX81 que se precie de serlo. El libro muestra una característica que puede interesar o no:



errores) nuestros programas y el uso de los ensambladores y monitores. Y aquí reside el principal defecto del libro: está orientado al uso del ensamblador ULTRAVIOLET, de ACS, un programa pionero en Gran Bretaña, pero difícil de encontrar en España.

Para los profanos a los que la lectura de otros libros les haga salir el error "Nonsense in Spanish", he aquí un libro adecuado para dar los primeros pasos en este campo.

## 1 Programación en Código Máquina para el ZX81 y para el Spectrum

Autor: Joan Sales Roig

Editorial: Ediciones Técnicas REDE, 1984

158 páginas

Precio: 1.200 ptas.

Orientado hacia la programación con ZX81. Un apéndice final

mente los distintos tipos de instrucciones máquina del Z80, trabajando cada sección mediante ejemplos concretos. Más avanzado el libro, se nos explica el funcionamiento del teclado y la pantalla. Son estas las partes del libro más difíciles de adaptar al Spectrum.

También el capítulo destinado a las direcciones útiles de la ROM experimenta algunos cambios, aunque los "restarts" cumplen funciones parecidas en ambas máquinas. Las direcciones restantes de la ROM no coinciden, por lo que habrán de ser adaptadas para su uso con el Spectrum.

Un libro útil para la programación en código máquina de ZX81 y, con las salvedades antes citadas, también interesante si nuestra máquina es un Spectrum, aunque en este caso debemos realizar una lectura más "inteligente" del texto.

está construido de principio a fin sobre la arquitectura del ZX81. Esto puede ser muy útil si el usuario posee un ZX81; si, en cambio, poseemos un Spectrum, el libro pierde interés, ya que existe otro del mismo autor, comentado más adelante, específicamente para los usuarios de este último.

Aunque el microprocesador de ambas máquinas sea el Z80, y por tanto los códigos de programa sean compatibles entre ambas máquinas, las direcciones de las subrutinas de la ROM son distintas, así como las variables del sistema y la memoria de pantalla. Por esta razón, un programa escrito para ejecutar sobre ZX81 no "correrá" correctamente sobre el Spectrum si utiliza este tipo de llamadas al sistema operativo o dibuja sobre la pantalla.

Este manual resulta un auxiliar imprescindible para los veteranos poseedores de ZX81 que se han in-

troducido ya en el código máquina, a pesar de que tiene numerosas erratas en las direcciones de la ROM y en los listados de los programas. Los que posean un Spectrum y hayan dejado el libro arrinconado con el ZX81, que lo desempolven, ya que todo aquello que no se refiera al sistema le será de indiscutible utilidad.

**1 ZX Spectrum. Programación en lenguaje ensamblador**

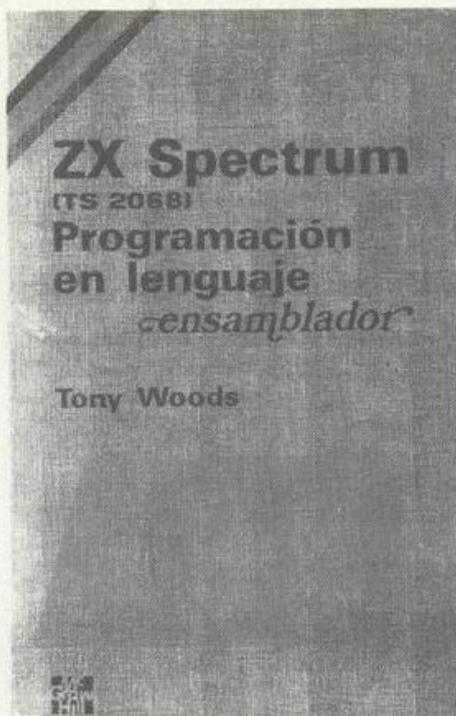
**Autor:** Tony Woods  
**Editorial:** McGraw-Hill, 1983  
**174 páginas**  
**Precio:** 1.290 ptas.

Una obra adecuada para la persona que ya tenga una cierta experiencia en programación en BASIC. Está bien adaptada al Spectrum, enseñando la utilización de las rutinas de la ROM para realizar los "trabajos pesados". Así el principiante no tendrá que hacer una complicada subrutina sólo para poder ver caracteres por la pantalla, y los programas resultarán más cortos, con la consiguiente facilidad para el aprendizaje. La entrada de datos y el uso del altavoz se realizan asimismo mediante llamadas a la ROM, introduciéndose después las subrutinas para realizar esta actividad uno mismo.

El libro realiza una revisión casi exhaustiva del conjunto de instrucciones del Z80, dejando en el tintero solamente los temas más complicados, como la gestión de interrupciones. Durante todo el tiempo los ejemplos han sido seleccionados de forma que nos ayuden a conocer mejor la estructura de memoria del Spectrum.

El volumen está bien traducido y presentado, siendo quizá su defecto principal la confusión causada por la descripción de los programas usando el ensamblador de

la casa inglesa ACS. Los lectores que no dispongan de este programa deberán modificar algunas pseudoinstrucciones, así como la forma de introducción de los programas.



**1 Spectrum machine code made easy. Vol 1. For Beginners**

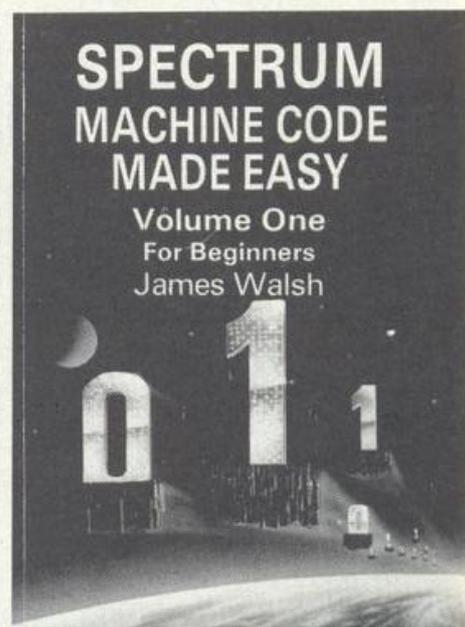
**Autor:** James Walsh  
**Editorial:** Interface Publications, 1983  
**224 páginas**

Un texto realmente agradable, lo que hace doblemente lamentable el hecho de que no haya sido traducido al castellano. Los que hablen inglés pueden disfrutar con este libro, escrito por un chico de dieciséis años, que sólo presupone un conocimiento superficial de BASIC a sus lectores.

La obra comienza preguntándose por qué utilizar código máquina. Después nos introduce en la forma de representación de datos usada por el ordenador, en las direcciones de memoria, registros, operaciones, etcétera.

Otro capítulo nos introduce en los ensambladores, desensambladores y monitores de código máquina, preguntándose nuevamente cómo y por qué utilizarlos. Se nos explica también la mecánica que debemos seguir para escribir un programa.

El libro ha sido escrito pensando en el Spectrum, y existe un capítulo



lo destinado a la entrada-salida de datos usando la ROM. Todos los temas están pensados para la persona que no tiene ningún conocimiento informático, siendo su principal problema la falta de un índice temático que nos facilite la búsqueda de palabras concretas.

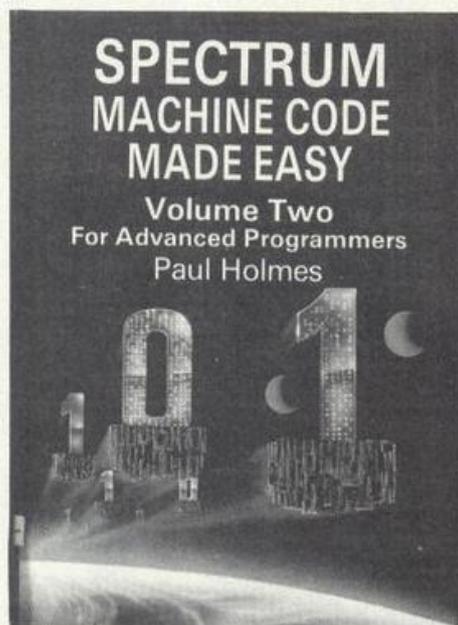
**2 Spectrum. Machine Code made easy. Vol 2. For Advanced programmers**

**Autor:** Paul Holmes  
**Editorial:** Interface Publications, 1983  
**150 páginas**

En este libro, segundo de la serie que la Editorial Interface ha dedicado al código máquina del Spectrum, nos encontramos con las instrucciones que no se habían discutido en el volumen anterior: las instrucciones de desplazamiento y lógicas bit a bit, las instrucciones input-output, y las interrupciones. Todos estos temas están tratados a

un nivel asequible, pero que presupone ya unos conocimientos previos de código máquina.

Los ejemplos son progresivamente más complicados (y más útiles) que en la obra anterior, por ejemplo se desarrolla una subrutina para imprimir a doble alto en la pantalla. Se desarrolla asimismo



un programa «máquina de escribir» que podría ser el embrión de un procesador de textos.

Nuevamente echamos en falta la presencia de un índice a la hora de buscar soluciones a una pregunta concreta sin repasarnos todos los capítulos. Sin embargo, las personas con poco conocimientos de programación y que conozcan el inglés encontrarán la lectura de los libros que componen esta serie instructiva y amena, si es que se puede considerar ameno un tema como el código máquina.

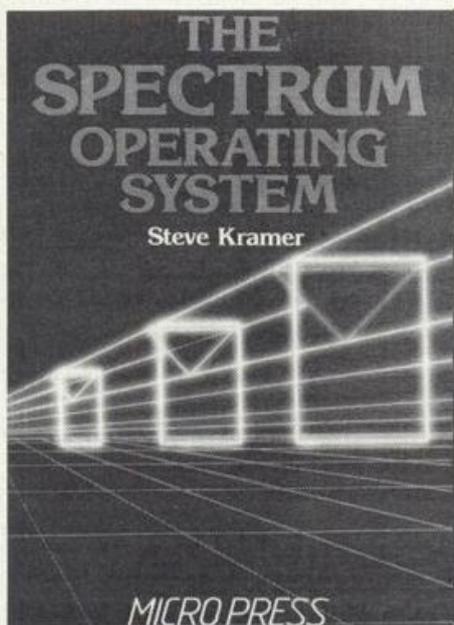
---

**4** The Spectrum operating System  
Autor: Steve Kramer  
Editorial: Micropress, 1984  
132 páginas

Un manual destinado a los que ya "controlan" algo de código máquina. El autor nos comenta las subrutinas de la ROM, tanto la básica como la primera versión de la ROM del Interface I, dándonos explicaciones detalladas del funcio-

namiento del sistema operativo del Spectrum.

La utilidad de este tipo de obras es doble: Por un lado, el conocimiento del sistema operativo nos permite la utilización de los programas residentes en la ROM, ahorrándonos así espacio y esfuerzo de programación al desarrollar nues-



tras aplicaciones en Código máquina.

Por otro, sabiendo cómo hace las cosas el BASIC, podremos optimizar, a veces apreciablemente, nuestros programas. Incluso, los más hábiles, pueden llegar a inventar nuevas técnicas de protección, que los "genios" ingleses no hayan descubierto todavía.

Las interrupciones, el mapa de memoria, las variables del sistema, la ampliación del BASIC mediante el Interface I, etc., son algunos de los temas que trata esta obra, muy útil para los que quieren ir "más allá".

---

**2** Mastering Machine Code on your ZX Spectrum  
Autor: Toni Baker  
Editorial: Interface Publications, 1983  
315 páginas

Un autor ya veterano en estas lides. En estas mismas páginas se co-

menta su "clásico" sobre código máquina para ZX81. Aparentemente, la obra que tratamos ahora no se diferencia apenas de la anterior, habiéndose mantenido los títulos de capítulo. Sólo una lectura atenta revela que las características del Spectrum han sido tenidas en cuenta, y que los dos libros son



menos parecidos de lo que creíamos.

Se trata de un manual muy completo, que cubre todos los campos relacionados con la programación del Spectrum en su lenguaje nativo. El autor explica el mapa de memoria, la manera de introducir el código en memoria sin que desaparezca al ser "invadido" por el BASIC, las direcciones más útiles de la ROM, etc., lo que nos facilita una comprensión mejor del trabajo del Spectrum en BASIC. Incluso, como una característica poco habitual, la manera que utiliza el Spectrum para almacenar números en coma flotante se estudia detalladamente, explicándonos asimismo la manera de manejar este tipo de números mediante llamadas a la ROM.

Un libro interesante para aprender a manejarse en código máquina, y que proporcionará conocimientos interesantes de la ROM del Spectrum a quienes quieran saber cómo y por qué funciona el Spectrum. ★

## Un poco de terminología

Como todos estamos ya cansados de oír, los procesadores sólo entienden como instrucciones secuencias de números binarios. Puesto que es difícil recordar sus significados y corregir los errores, es muy útil que el propio ordenador nos traduzca los programas, de un lenguaje que entendamos nosotros (más o menos), a su lenguaje interno. Ese es el trabajo (por ejemplo), del intérprete BASIC residente en la ROM.

En el caso de la escritura de programas en código máquina, existen dos tipos de "ayudas" básicas: los cargadores hexadecimales (o decimales) y los ensambladores. En estos últimos programas las instrucciones se escriben, como ya sabréis, en una jerga que los fabricantes llaman mnemónica, mucho más relacionada con la forma de pensar de las personas (o por lo menos de los ingleses). La diferencia entre usar código máquina en binario, hexadecimal o ensamblador se puede ver con un pequeño ejemplo:

00111010,01100000,00000000  
 en binario.  
 3A,60,00  
 en hexadecimal.  
 LD A,(0060H)  
 en ensamblador.

Son estas tres maneras de decir al ordenador que cargue el acumulador con el número contenido en la localidad de memoria de dirección 96 (en decimal). Como podemos ver, el cargador hexadecimal nos permite reducir la cantidad de cifras a teclear, pero sigue siendo bastante ininteligible. El uso de un ensamblador, en cambio, aumenta la legibilidad y facilita la corrección de errores. Además, los ensambladores permiten referenciar las direcciones de memoria mediante símbolos, lo que hace más fácil saber su significado.

Este último hecho resulta muy útil por otra razón: el código máquina del Z80 se debe programar

para ser ejecutado en una posición fija de memoria. Esto quiere decir que los programas no son "relocalizables" salvo muy reducidas excepciones. Los ensambladores, al poder hacer referencia a localidades de memoria de forma simbólica, permiten "reensamblar" un programa en diferentes posiciones, cambiando sólo la dirección de comienzo.

Un problema que hace rechazar a los usuarios el código máquina con mucha frecuencia es la dificultad

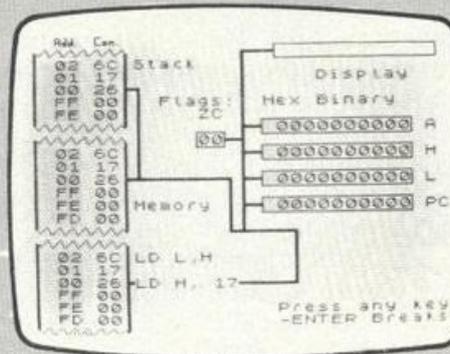
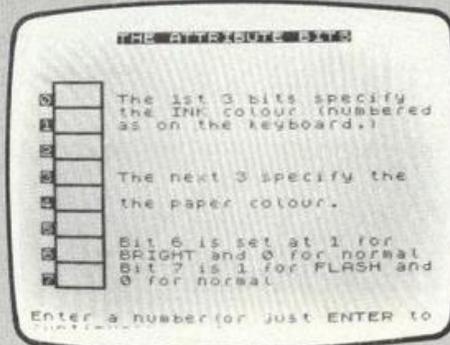
de que debiera, señal de que nos hemos equivocado el algún sitio, por lo que debemos modificar parte de nuestro programa y probar otra vez.

Otro tipo de programas son los paquetes de aprendizaje de código máquina, aún ausentes del mercado español.

El primero, *Machine Code for Beginners*, presentado por la casa inglesa **Dream Software**, es un pequeño monitor-ensamblador pensando para el aprendizaje, y basado en la simulación de un microprocesador inexistente, cuyo juego de instrucciones y registros es un pequeño subconjunto del correspondiente al Z80. La simplificación del juego de instrucciones y la reducción del número de celdas de memoria (256), permite un manejo sencillo al principiante, que puede pasar gradualmente al juego de instrucciones completo, dosificando las dificultades. Una pequeña joya, completada por un excelente manual de instrucciones (en inglés).

La otra cinta que comentamos tiene un planteamiento mucho más convencional. Su nombre ya nos indica mucho sobre sus objetivos: *Beyond Horizons* (Más allá del horizonte), juego de palabras con la cinta "Horizontes" que sirve como introducción al uso del Spectrum. La cinta repasa, a lo largo de once capítulos, el mapa de memoria del Spectrum, las instrucciones PEEK y POKE, el manejo de la memoria de pantalla, etc. A pesar de su nombre, no va mucho más allá del manual del Spectrum.

Las posibilidades gráficas y de color del Spectrum no han sido aprovechadas como debieran (sólo encontramos un capítulo en el que se presentaban algún diagrama gráfico en blanco y negro). Para los fabricantes españoles de software resultaría muy fácil superar, teniendo en cuenta el idioma, cintas como esta de **Robotics**. ✱



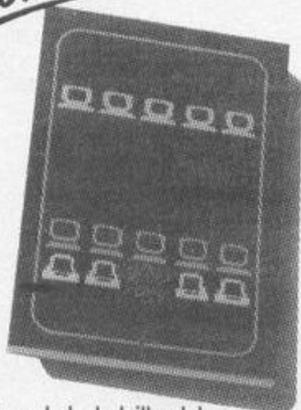
tad de "depurar" los programas escritos. No existe BREAK, ni mensajes de error, por lo que tener un "monitor" nos resultará muy cómodo para comprobar los programas.

Un monitor es un programa que simula la ejecución de un programa código máquina "paso a paso", permitiéndonos generalmente ver los cambios que experimentan los contenidos de los registros, los flags, las localidades de memoria, etc. Si un registro no contiene lo



**Alhambra  
informática**

*novedad*



Manual de bolsillo del  
SINCLAIR SPECTRUM

Esta guía no pretende sustituir al manual de instrucciones del Sinclair-Spectrum, sino proporcionar una información fácilmente asequible sobre todas las cuestiones que el usuario pueda plantearse para obtener un mayor rendimiento de su ordenador personal.

Redactado de forma sencilla y concisa, contiene una valiosa serie de datos sobre el manejo del Sinclair-Spectrum.

**OTROS TITULOS  
PUBLICADOS**

1. *66 Programas en Basic. De la E.G.B. a la Universidad.* A. MARTINEZ I. RIEIRO.
2. *Estudio general del Ordenador.* P. BISHOP.
3. *Proceso de datos.* D. HARRISON.
5. *Adopción de decisiones en actividades comerciales (IBM PC).* W.R. OSGOOD  
J.F. MOLLOY.

De venta en librerías y tiendas de informática.

**EDITORIAL  
Alhambra, s.a.**  
C/ Claudio Coello, 76  
Madrid

# Guía del comprador de Todospectrum

## COMPUTIQUE

Si posees un Spectrum y —o un QL,  
Si dominas el código Máquina,  
Si te gusta la programación y  
puedes escribir un buen programa

**¡CONTACTA CON NOSOTROS!**

**COMPUTIQUE**

C/ Embajadores, 90. 28012 MADRID  
Tfno. 227 09 80 - 227 91 99



## MULTISYSTEM, S. A.

**BOUTIQUE INFORMATICA**

- \* Ordenadores Personales.
- \* Micro-ordenadores de gestión.

**Todas las novedades en:**  
Programas - Periféricos - libros  
(nacionales y de importación)

**Para: Spectrum - Dragón - Base 64  
Spectravideo - Oric - Commodore, etc.**

C/ San Vicente, 53. ALICANTE. Tel. (965) 21 55 66.

## FACTURACION SPECTRUM

Un programa que le permite realizar:

Facturas  
Pedidos  
Ofertas  
Albaranes  
Control de Stocks  
Listas de Precios  
20 Ficheros diferentes  
1000 Artículos  
400 Fichas

Un sólo programa de  
fácil manejo con microdrive  
con 20 ficheros de clientes,  
proveedores, artículos, etc.

**ALSI, S. A.** Antonio López, 154.  
Tel. 91/475 43 39. 28026 MADRID

## K-BITS

- APPLE
- SPECTRUM
- SPECTRAVIDEO
- COMMODORE
- AMSTRAD
- DRAGON

### OFERTA MES DE MARZO

En importes superiores a 25.000 ptas.  
Vale obsequio 10% descuento para su  
próxima compra

Barquillo, 15 - Tel. 232 57 37  
MADRID



- Ordenadores personales Hard y Soft.
- Cursos de Basic.

Oficina **RENOVACION EN MARCHA, S. A.**  
C/ Espronceda, 34. 28003-MADRID  
Tfno. (91) 441 24 78

**REMSHOP 1**  
Galileo, 4. 28015 MADRID  
Tfno. (91) 445 28 08

**REMSHOP 2**  
C/ Dr. Castelo, 14. 28008 MADRID  
Tfno. (91) 274 98 43

**REMSHOP 3**  
C/ Modesto Lafuente, 33. 28003 MADRID  
Tfno. (91) 233 83 19

**REMSHOP BARCELONA**  
C/ Pelayo, 12. Entrésuelo J 08881 BARCELONA  
Tfno. (93) 301 47 00

**REMSHOP LAS PALMAS**  
C/ General Mas de Gamíndez, 45. LAS PALMAS  
Tfno. (928) 23 02 90

**ELECTRONICA  
SANDOVAL S.A.**

DISTRIBUIDORES DE:

COMMODORE-64  
ORIC-ATMOS  
ZX SPECTRUM  
SINCLAIR ZX 81  
ROCKWELL'-AIM-65  
DRAGON-32  
NEW BRAIN  
DRAGON-64  
CASIO FP-200

ELECTRONICA SANDOVAL, S. A.  
C/ SANDOVAL, 3, 4, 6. 28010-MADRID  
Teléfonos: 445 75 58 - 445 76 00 - 445 18 70  
447 42 01  
C/ SANDOVAL, 4 y 6  
Centralita 445 18 33 (8 líneas)

## ULTIMO AVISO

¿Eres aficionado a la programación?  
¿Dominas el código máquina?  
¿Tienes programas originales?  
¿Puedes escribir un buen juego?  
¿Quieres ganar dólares, libras, francos o  
pesetas desde tu casa, en tus horas libres?

## ¡NO TE LO PIERDAS!

Contacta inmediatamente con:

**CIBERCOMP, S. A.**  
Tels. (91) 200 21 00  
(91) 759 22 44

Especialistas en software para Home Computers,  
asociados con primeras firmas internacionales.



## CAMAFEO INC.

**CASSETTES  
DE CALIDAD PROBADA  
PARA ORDENADORES**

Cada uno	Caja de 10	Caja de 30
C-5 199 ptas	1 393 ptas.	3 582 ptas.
C-10 209 ptas	1 463 ptas.	3 762 ptas.
C-15 219 ptas	1 533 ptas.	3 942 ptas.
C-20 229 ptas	1 602 ptas.	4 122 ptas.

Libre de gastos de envío contra reembolso correos

**CAMAFEO INC. Dep.03**  
José Lázaro Galdiano, 1. 28036 Madrid.

Su micro ordenador en Canarias

**sinclair**

**ZX - SPECTRUM 48 K**  
Commodore -64

Periféricos - Programas y juegos

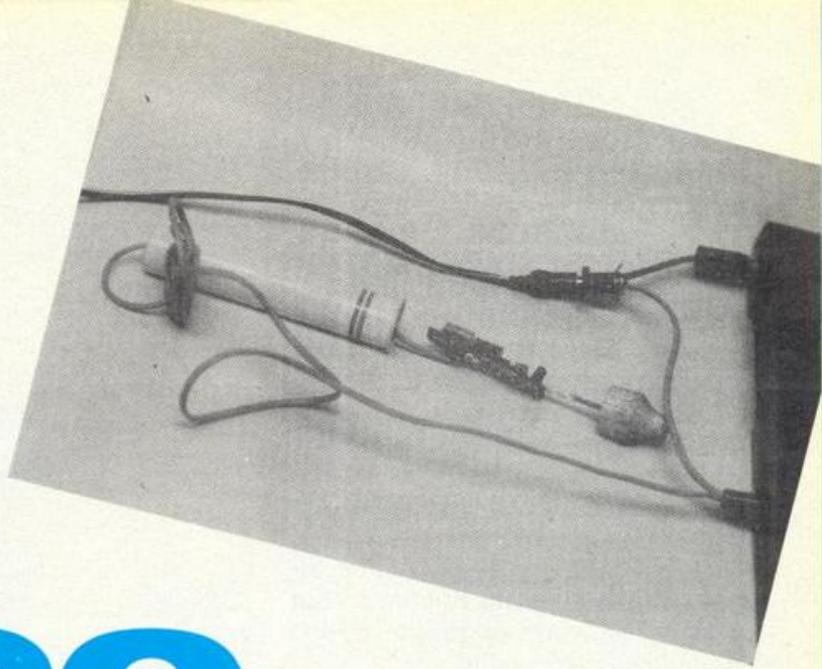
**Precios especiales**

**MEL IMPEX**

La Naval, 143 Las Palmas. Tfno. 928-27 43 11

# Lápiz óptico

y2



En el número anterior describimos la construcción de un eficaz lápiz óptico y prometimos ofrecer los programas que permiten manejarlo con comodidad. Y como dicen que lo prometido es deuda...

Quienes siguieran el artículo sobre la construcción del lápiz recordarán que si éste se halla sobre una zona de la televisión que esté iluminada, recibiremos por la entrada de *cassette* un pulso de tensión. No sabemos a priori en qué instan-

te ocurrirá, pero estamos seguros de que si a los 40 ms. no hemos recibido ninguna señal, el lápiz se halla sobre una zona oscura. La razón de esto es que una pantalla se barre 25 veces por segundo, o lo que es lo mismo, el haz de electro-

nes pasa por un punto una vez cada 40 ms. La luminosidad de cada punto es función de la intensidad del haz en cada instante.

En definitiva, tendremos que muestrear continuamente la entrada EAR durante al menos 40 ms. Si detectamos alguna señal podemos suponer que nuestro lápiz ha localizado un punto luminoso de la pantalla. Si, por contra, no se ha observado cambio alguno a la entrada, podemos estar seguros de

## Programa 1

```

0000      00001 ;*****
0000      00002 ;* LAPIZ OPTICO *
0000      00003 ;* RUTINA *
0000      00004 ;*****
0000      00005
0000      00006 ;@ Luis Miguel
0000      00007 ; BRUGAROLAS 1984
0000      00008
0000      00009 ;rutina RELOCALIZABLE
0000      00010
0000 010000 00011 LD B,0000
0000 116004 00012 LD DE,0460h
0000 3E7F 00013 LOOP LD A,7Fh
0000 DBFE 00014 IN A,(0FEh)
000A E540 00015 AND #01000000b
000C 2006 00016 JR Z,PULSE
000E 1B 00017 ENTRY2 DEC DE
000F 7A 00018 LD A,D
0010 53 00019 OR E
0011 20F3 00020 JR NZ,LOOP
0013 C9 00021 RET
0014
0014 0640 00022 PULSE LD B,40h
0016 10FE 00023 DELAY DJNZ DELAY
0018 3E7F 00024 LD A,7Fh
001A DBFE 00025 IN A,(0FEh)
001C E540 00026 AND #01000000b
001E 2006 00027 JR NZ,ENTRY2
0020 0C 00028 INC C
0021 C9 00029 RET
0022      00030
0022      00031

```

00000 TOTAL ERRORS

DELAY 0016  
ENTRY2 000E  
LOOP 0006  
PULSE 0014

## Programa 2

```

10 RESTORE
20 DATA 1,0,0,17,96,4,62,127,2
19,254
30 DATA 230,64,40,6,27,122,179
,32,243
40 DATA 201,6,64,16,254,62,127
,219,254
50 DATA 230,64,32,238,12,201
60 LET SUMA=0: FOR I=23296 TO
23329: READ A: POKE I,A: LET SUM
A=SUMA+A: NEXT I
70 IF SUMA<>3703 THEN PRINT "
Ha escrito mal el DATA": STOP
80 PRINT "Todo correcto.Grabe l
a rutina conSAVE ""Lapiz"" CODE
23296,33. Recuerde que puede
cargarla en cualquier direccio
n"

```

## Programa 3

```

10 INK 7: PAPER 0: BORDER 0: C
LS
20 CLEAR 65000: LET RUT=65501:
LOAD ""CODE RUT
30 READ X0,Y0,N
40 DATA 8,6,6
50 BEEP .1,30

```

```

100 PRINT AT Y0-3,2:"Escriba su
opcion:"""TAB X0+1;"1<- Opcion
1""TAB X0+1;"2<- Opcion 2""TAB
X0+1;"3<- Opcion 3""TAB X0+1;"
4<- Opcion 4""TAB X0+1;"5<- Opc
ion 5""TAB X0+1;"6<- Opcion 6"
110 FOR I=0 TO N-1: FOR J=1 TO
5: PRINT AT 2*I+Y0,X0: INVERSE 1
;" ": IF USR RUT THEN PRINT AT
2*I+Y0,X0;" ": IF NOT USR RUT TH
EN NEXT J: LET A*=STR# (I+1): G
O TO 200
120 PRINT AT 2*I+Y0,X0;" ": NEX
T I: LET A*=INKEY#: IF A*="" THE
N GO TO 110
200 BEEP .2,35: LET A=VAL A#: I
F A<=N THEN PRINT AT 2*(A-1)+Y0
,X0+5: OVER 1: FLASH 1;"
"

```

```

210 IF A=1 THEN REM .....
220 IF A=2 THEN REM .....
230 IF A=3 THEN REM .....
240 IF A=4 THEN REM ....
250 IF A=5 THEN REM ..
260 IF A=6 THEN REM
270 CLS : PRINT AT 20,3:"Se equ
ivoco pruebe de nuevo": GO TO 10
0

```

que se halla sobre una zona oscura, o simplemente que no está apuntando al tubo de rayos catódicos.

Sin embargo, al experimentar un poco llegamos a convencernos de que nuestro Spectrum está gobernado por una legión de brujas que se divierten enormemente haciéndonos todo tipo de perrerías. Resulta que incluso con los cables de magnetófono desenchufados, se detectan pulsos. Un sistema de filtrado eficaz consiste en comprobar si después de un pequeño tiempo sigue presente la señal.

La rutina definitiva se encuentra bajo el epígrafe de programa 1. Es relocable, esto es, no contiene saltos a direcciones absolutas por lo que puede introducirse en cualquier lugar de la memoria.

Para cargar el programa desde BASIC, se recomienda escribir el programa 2, que introduce el código en forma rápida a partir de la dirección *rut*, evitando así tener que esperar el programa en código máquina.

Cuando hacemos `LET a=USR rut`, la variable "a" toma el valor 0 ó 1 en función de que el lápiz se halle sobre una zona oscura o iluminada, respectivamente. De esta manera:

`IF USR rut THEN... (o USR`

`rut=1)` llevará a cabo la orden correspondiente si el lápiz se halla sobre el punto iluminado.

`IF NOT USR rut THEN... (o USR rut=0)` lo hará si el lápiz no está sobre zona iluminada.

Veamos un ejemplo. Supone-

**Se detectan pulsos incluso con los cables de magnetófono desenchufados.**

mos el programa 2 introducido. Escribimos:

```
100 INK 7:PAPER 0:BORDER
0:CLS
110 PRINT AT 10,10; INVERSE
1;"
120 IF USR rut THEN PRINT
"Lápiz sobre cuadrado blanco":S-
TOP
130 GO TO 120
```

Pero si hubiera sobre la pantalla otro punto blanco, no sabríamos

cuál es el escogido. Tenemos dos alternativas posibles: hacer que los cuadrados no estén iluminados simultáneamente (en el momento de ejecutar la rutina en código máquina, existe una única zona blanca en la pantalla, si detectamos algo, es porque el lápiz se halla sobre dicha región).

El problema que presenta este método es que resulta de poca utilidad tener un único cuadrado testigo. Generalmente habrá textos, dibujos, etc. La salida al problema consiste en comprobar si, modificando el estímulo, varía la respuesta. En la práctica, esto supone ver si al detectar un pulso con el cuadrado iluminado, se dejan de recibir pulsos con el cuadro oscuro. Si repetimos el proceso varias veces y en todas se confirman las previsiones, podemos estar seguros de estar sobre el cuadro seleccionado. Quienes todavía tengan dudas, las aclararán con el siguiente ejemplo.

### Selección de opciones en menú

Es posiblemente una de las aplicaciones más interesantes del lápiz óptico. En la pantalla aparece un

### Programa 4

```
00001 ;*****
00002 ;* LAPIZ OPTICO *
00003 ;* definicion de graficos *
00004 ;*****
00005
00006 ;@ Luis Miguel BRUGAROLAS 1984
00007
00008
00009
00010 BEGIN LD HL,0007h
00011 LD (LONGX),HL
00012 LD L,03
00013 LD (LONGY),HL
00014
00015 LD A,07Fh ;FINAL
00016 IN A,(0FEh) ;BREAK
00017 RRCA
00018 RET NC
00019
00020 LD A,0F7h ;TECLA '1'
00021 IN A,(0FEh) ;Congelar
00022 RRCA
00023 JR NC,BEGIN
00024
00025 LD A,0EFh ;TECLA '0'
00026 IN A,(0FEh)
00027 RRCA
00028 LD HL,5C81h
00029 JR C,NZERO
00030 SET 0,(HL) ;Borrar
00031 LD A,0DFh ;TECLA 'P'
00032 IN A,(0FEh)
00033 RRCA
00034 JR C,INDIC
00035 RES 0,(HL) ;Dibujar
00036
```

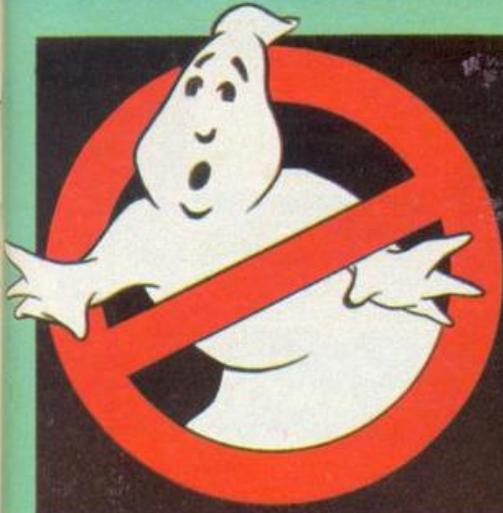
```
00037 ;Indicar modo DIBUJO o BORRADO
00038
00039 INDIC LD B,0Ah
00040 LD DE,22520+96
00041 XOR A
00042 BIT 0,(HL)
00043 JR Z,CORR
00044 LD A,30h
00045 CORR LD (DE),A
00046 INC DE
00047 DJNZ CORR
00048 LD E,126
00049 XOR A
00050 LD B,0Ah
00051 BIT 0,(HL)
00052 JR NZ,ESCR
00053 LD A,30h
00054 ESCR LD (DE),A
00055 INC DE
00056 DJNZ ESCR
00057
00058 LD A,0FFh
00059 LD (XPAT),A
00060 AND A
00061 EX AF,AF
00062 EX AF,AF
00063 CONT2 EX AF,AF
00064 ; CF:0-X,1-Y
00065 LD HL,(LONGX)
00066 JR NC,COMMON
00067 LD HL,(LONGY)
00068 COMMON EX AF,AF
00069
00070 ENTRY XOR A
00071 LD B,L
00072 INC B
```

```
00073 LOOP3 RLCA
00074 OR 01
00075 DJNZ LOOP3
00076 LD C,A
00077 LD A,L
00078 CPL
00079 AND 0000111b
00080 AND H
00081 LD B,A
00082 LD A,C
00083 JR Z,SAVE
00084 LOOP4 RLCA
00085 DJNZ LOOP4
00086
00087 SAVE LD C,A
00088 EX AF,AF
00089 LD A,C
00090 JR NC,XJORK1
00091 LD (YPAT),A
00092 JR JUMP1
00093 XJORK1 LD (XPAT),A
00094 JUMP1 EX AF,AF
00095 PUSH HL
00096
00097 LD A,(YPAT)
00098 LD C,A
00099 LD A,(XPAT)
00100 CALL PRINT
00101 CALL SIGNAL
00102 JR C,TEST; No detectado
00103 LD A,(YPAT)
00104 CPL ; Dibujo complement
00105 LD C,A
00106 LD A,(XPAT)
00107 CALL PRINT
00108 CALL SIGNAL
```

# commodore

## Magazine

AÑO 2 - Núm. 13 Marzo 1985 - 250 Ptas

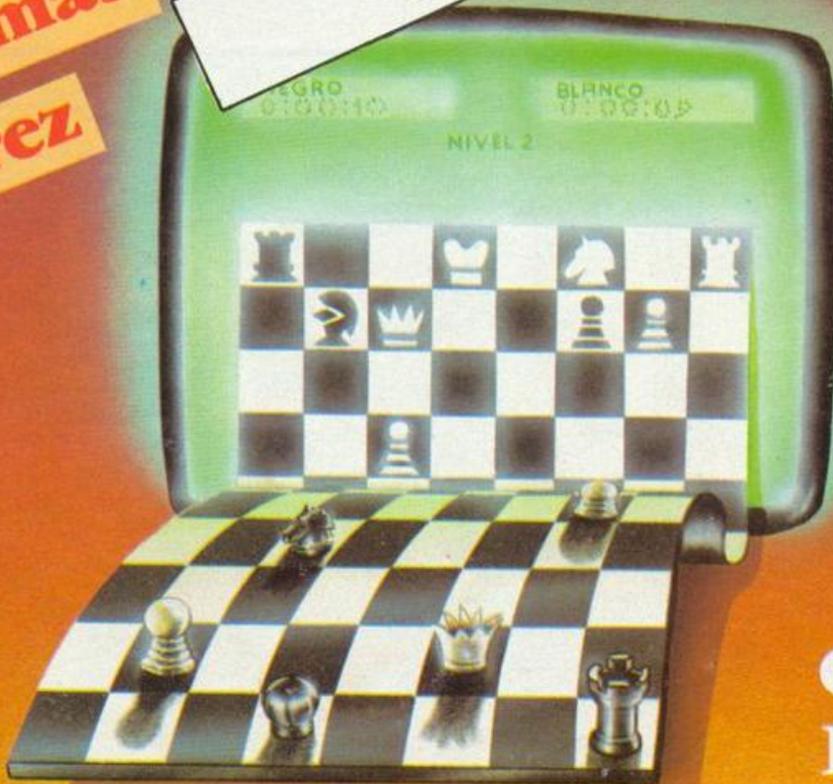


**64**  
**Los Cazafantasmas**



**YA ESTA A LA VENTA**  
que dibuja

**Análisis:  
Programas  
de ajedrez**



**Centronics  
Interface**

menú que tiene la particularidad de que delante de todas las opciones se desliza rápidamente un cuadro luminoso. (Siempre que se trabaja con lápiz óptico, debemos hacerlo con papel negro y tinta blanca).

La selección de opción podrá realizarse, bien apretando la tecla correspondiente, bien apuntando con el lápiz la casilla anterior al número de opción (ver programa 3).

Para utilizarlo en algún programa propio, debe saberse que N representa el número de opciones posible, y que X0 e Y0 son las coordenadas del comienzo del texto de la primera opción. El número de comprobaciones que se hacen antes de dar por válida una elección (línea 110, parámetro extremo del FOR J) puede variarse a voluntad.

Para usar el programa, será necesario dar los nombres correspondientes a las opciones (modificando su número si es necesario) y definir las acciones en los casos correspondientes (líneas 210 a 260).

Aunque el BEEP de la línea 50 pudiera parecer superfluo, no lo es en absoluto. Para comprobarlo, intenten ejecutar el programa cuando no está presente.

## GENERACION DE CARACTERES

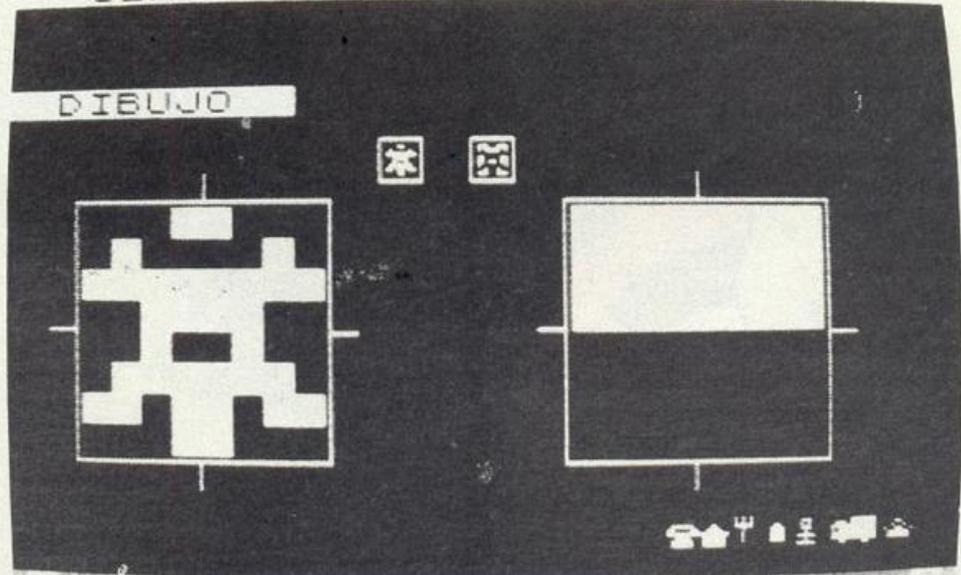


Figura 1.—Ejemplo de definición de caracteres gráficos con el lápiz óptico.

### Definición de gráficos

Todos los lectores habrán comprobado que la definición de gráficos vía BASIC es muy tediosa. Incluso los programas que facilitan la operación, como el que se presenta en la cinta de demostración HORIZONTES, no alivian mucho el problema. Proponemos aquí un ejemplo de utilización del lápiz

óptico para simplificar al máximo la operación hasta el punto de convertirla en un divertido juego. Se podrán definir, no sólo los UDG, sino el total de caracteres de los que dispone el Spectrum, a excepción de los 16 gráficos asignados a los números. De esta manera podemos conseguir que nuestro ordenador escriba en un complicado estilo gótico, o que los listados

```

00109 CCF
00110 JR NC,TEST; Correcto
00111 INC A; Error
00112
00113 TEST POP HL
00114 JP NZ,BEGIN; ERROR
00115 JR NC,CONT1; DETECTED
00116 LD A,L; Patron complex
00117 INC A
00118 OR H
00119 LD C,A
00120 CP H
00121 JP Z,BEGIN;Ya habiamos
00122 ;analizado el complementario
00123 LD H,C
00124 JR ENTRY
00125
00126 CONT1 LD A,H
00127 EX AF,AF'
00128 JR NC,XUORK
00129 EX AF,AF'
00130 LD (CORDY),A
00131 LD A,(LONGY)
00132 RRA ;Carry esta a 0
00133 LD (LONGY),A
00134 JR CONT2
00135
00136 XUORK EX AF,AF'
00137 LD (CORDX),A
00138 LD A,(LONGX)
00139 AND A; Carry a 0
00140 JR Z,POINT
00141 RRA
00142 LD (LONGX),A
00143 JP CONT2
00144

00145 POINT LD DE,0008h;Duration
00146 LD HL,0008h;Pitch
00147 CALL 0305h;BEEP
00148
00149 LD A,(YPAT)
00150 LD C,A
00151 LD A,(XPAT)
00152 HL,SC01h
00153 BIT 0,(HL)
00154 JR Z,INSERT
00155 CPL ;
00156 Borrar punto
00157 INSERT LD D,A; D:XPAT*
00158 EX AF,AF' C:YPAT
00159 LD HL,ARRAY
00160 LD B,00
00161 LOOP1 LD A,D
00162 SLA C
00163 JR NC,NEXT1
00164 EX AF,AF'
00165 JR NZ,DELETE
00166 EX AF,AF'
00167 OR (HL)
00168 JR JUMP2
00169 DELETE EX AF,AF'
00170 AND (HL)
00171 JUMP2 LD (HL),A
00172 NEXT1 INC HL
00173 DUNZ LOOP1
00174
00175 ;Banda central negra
00176 LOOP2 LD HL,5900h
00177 LD (HL),7
00178 INC L
00179 JR NZ,LOOP2
00180

00181 ;Modelo pequeno
00182 LD HL,16384+32*6+13
00183 LD DE,ARRAY
00184 LD B,00
00185 LD A,(DE)
00186 LD (HL),A
00187 CPL
00188 INC L
00189 INC L
00190 INC L
00191 LD (HL),A
00192 DEC L
00193 DEC L
00194 DEC L
00195 INC H
00196 INC DE
00197 DUNZ LITTLE
00198
00199 ;Dibujo de la izquierda
00200 LD HL,5903h
00201 LD DE,ARRAY
00202 LD A,(DE)
00203 LD B,06
00204 NEXTF LD B,06
00205 NEXTB RLCA
00206 JR NC,JUMP3
00207 LD (HL),38h
00208 INC L
00209 DUNZ NEXTB
00210 INC DE
00211 LD A,18h
00212 ADD A,L
00213 LD L,A
00214 JR NC,NEXTF
00215

```

aparezcan en los claros trazos del alfabeto griego.

El programa en BASIC incorpora la selección de menús con el lápiz. Tenemos, además del programa general de gestión, otros tres programas en código máquina. El primero utiliza la rutina de detección de pulsos del programa de gestión y la adapta a nuestros requisitos particulares. Las otras dos se encargan de transferir a RAM los datos referentes a los perfiles de los caracteres.

Dado que la descripción completa del programa principal (en Ensamblador) ocuparía posiblemente todo el ejemplar, vamos a discutir brevemente los puntos más sobresalientes (ver programas 4 y 5).

En primer lugar comentaremos el método empleado para hallar el punto seleccionado. Disponemos de un cuadrado de  $8 \times 8$  puntos, lo que resulta un total de 64. El método más inmediato sería el de barrer cada uno de los *pixels* hasta localizar el punto al que apunta el lápiz. Esto supone que, en el peor de los casos, necesitaríamos hacer 64 intentos antes de tener alguna idea de por dónde empezar a comprobar.

Con un poco de astucia, lograre-

mos reducir enormemente este índice. Para determinar cualquier número comprendido entre 0 y 7 necesitamos 3 bits. Para determinar dos (coordenadas x e y) necesitamos..., muy bien, 6 (en efecto  $2^6 = 64$ ). Debemos añadir todavía otro que indique si el lápiz se halla

elegido. El precio: relativa complicación en los programas de gestión.

El estado de la pantalla puede verse en la figura 1. En la parte izquierda se muestra el perfil actual del gráfico. La parte derecha es el área de trabajo. Al dibujar sobre ella, los puntos que resulten localizados, pasarán a la región izquierda, donde se añadirán al dibujo o se borrarán corrigiendo algún error.

Vamos a centrarnos sobre la "zona de trabajo". En primer lugar colorearemos de blanco la mitad superior del cuadrado. Si no se detecta ninguna señal, pasaremos a la inferior. Si de nuevo volvemos a tener una negativa, repetiremos el ciclo. Al detectar el lápiz en cualquiera de las dos mitades, pondremos blanca la parte izquierda de la región correspondiente. Si falla probaremos con la derecha... De esta manera vamos reduciendo sucesivamente el campo de posibilidades hasta localizar el punto. Siempre que detectemos el lápiz, borraremos la imagen para ver si de esta manera la señal desaparece. Es por eso por lo que si el lápiz se encuentra a caballo entre dos *pixels*, no se admite ninguno.

En cuanto al programa propia-

**Si a los 40 ms no se ha recibido ninguna señal, el lápiz se halla sobre una zona oscura.**

o no apuntando a la pantalla. Resumiendo, la cantidad mínima de intentos será de 7.

Nuestro procedimiento requiere un número nunca superior a 12 intentos, pero una vez acabada la búsqueda, tenemos la certeza absoluta de hacer determinado el punto

```

00216      JP  BEGIN
00217
00218      ;
00219      ; PRINT
00220      ; -----
00221
00222 PRINT LD  HL,5913h
00223      LD  D,A
00224 NEXTF2 SLA  C
00225      JR  C,FOR
00226      XOR A
00227 FOR   LD  B,08
00228 NEXTC RLCA
00229      LD  (HL),00 ; NEGRO
00230      JR  NC,INCR
00231      LD  (HL),38h ; BLANCO
00232 INCR  INC  HL
00233      DJNZ NEXTC
00234      LD  A,32-8
00235      ADD A,L
00236      LD  L,A
00237      RET C
00238      LD  A,D
00239      JR  NEXTF2
00240
00241      ;
00242      ; SIGNAL
00243      ; -----
00244      ; Carry a 0 si se detecta pulso
00245
00246 SIGNAL LD  DE,0460h
00247 LOOP  LD  A,7Fh
00248      IN  A,(0FEh)
00249      AND 40h

```

```

00250      JR  Z,PULSE
00251 ENTRY2 DEC  DE
00252      LD  A,D
00253      OR  E
00254      JR  NZ,LOOP
00255      SCF
00256      RET
00257
00258 PULSE  LD  B,40h
00259 DELAY  DJNZ DELAY
00260      LD  A,7Fh
00261      IN  A,(0FEh)
00262      AND 40h
00263      RET Z
00264      JR  ENTRY2
00265
00266      ;
00267      ; VARIABLES DEL SISTEMA
00268      ; -----
00269
00270 LONGX  DEFB 0
00271 CORDX  DEFB 0
00272 LONGY  DEFB 0
00273 CORDY  DEFB 0
00274 ARRAY EQU 0
00275      DEFW 0,0,0,0
00276 XPAT  EQU 0
00277      DEFS 1
00278 YPAT  EQU 0
00279      DEFS 1

```

#### Programa 5

10 REM GENERADOR DE CARACTERES

```

20 INK 7: PAPER 0: BORDER 0: C
LS
30 CLEAR 59999: LET men=23296:
LET car=60000: LET rut=65135: L
ET tabla=65519: LET udg=car+896:
POKE udg-1,62: POKE 23676,INT (
udg/256): POKE 23675,udg-256*INT
(udg/256): LOAD ""CODE:CLS
40 PRINT " GENERACION DE C
ARACTERES ""
50 LET m$="" : LET B$
=""
60 DATA 205,204,255,62,0,71,14
3,79,201,1,248,2,33,0,61,17,96,2
34,237,176,201,1,160,0,33,8,62,1
7,224,237,237,176,201
70 RESTORE : FOR i=0 TO 32: RE
AD a: POKE i+men,a: NEXT i: RAND
OMIZE USR (men+9): RANDOMIZE USR
(men+21)
800 BEEP .1,30: PRINT AT 5,4:"E
lija su opcion:"TAB 7:"1<-Def
inir UDG""TAB 7:"2<-Definir tod
os carac.""TAB 7:"3<-Corregir a
lguno""TAB 7:"4<-Grabar caracte
res""TAB 7:"5<-Ver caract. defi
nidos""TAB 7:"6<-Parar el progr
ama"

```

mente dicho vamos a discutir algunas líneas.

Dos de las variables de nuestro sistema reciben los nombres de LONGX y LONGY (LONG, generalizando). Cada una de ellas tiene dos bytes. Al primero se le conoce por el mismo nombre y el segundo es llamado CORD (X e Y). Su misión es la de guardar la coordenada de inicio de la zona blanca del patrón. En cuanto a los LONG el contenido indica el número de cuadrados en blanco menos uno que tiene el byte patrón las dos direcciones. Por ejemplo al empezar, el número de blancos es de 8 en X. Restando uno queda 7:0000 0111. En Y tenemos 4.4-1=3:0000 0011.

La creación de un byte con el número de cuadrados adecuado es trivial (ver líneas 70-75). Como se observa, al pasar a la etapa siguiente, debemos dividir por dos la longitud de blancos del modelo. Esto se consigue inmediatamente rotando el byte LONG a la derecha.

Sin embargo, no es esta la propiedad más interesante. Si incrementamos el valor de LONG tendremos a 1 el bit de la coordenada que tenemos que determinar. Por ejemplo, cuando LONGX=4 pode-

mos estar en la mitad derecha o en la izquierda. Esto es CORDX=0 o CORDY=4 (0011+1=0100. En general tenemos 0X00. Si X=0, CORDX=0; si X=1 CORDY=4).

**Un sistema de filtrado eficaz consiste en comprobar si después de un pequeño tiempo sigue presente la señal.**

En 116-118 hallamos el método de cálculo. Conocida la coordenada podemos hallar el patrón definitivo fácilmente: líneas 81-85.

Una vez que el punto ha sido detectado, se actualiza el modelo del gráfico que se almacena en el espacio llamado ARRAY, y se dibuja en la pantalla.

Para acabar vamos a explicar

cómo utilizar desde el BASIC los gráficos definidos:

— Caracteres normales: si se empiezan a almacenar a partir de la dirección *car*, hacer:

```
POKE 23606,car-256*INT(car/256)
POKE 23607,INT(car/256)-1
```

Al pasar a los caracteres normales:

```
POKE 23606,0
POKE 23607,60
```

— UDG, gráficos definidos por el usuario: si se almacenan a partir de *udg*

```
POKE udg-1,62
POKE 23675,udg-256*INT(udg/256)
POKE 23676,INT(udg/256)
```

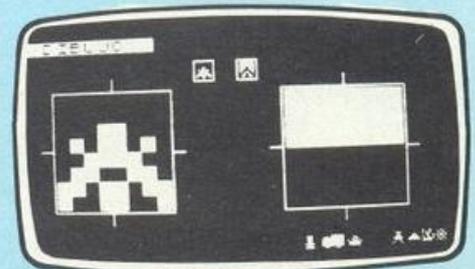
De esta forma ya no tiene excusa para utilizar un lápiz óptico, salvo que le "asuste" meterse en "montajes caseros", en cuyo caso le aconsejamos tome el camino más fácil: el lápiz comercial, al que en un futuro número pasaremos revisión. ★

Luis Miguel

```
805 FOR I=1 TO 6: FOR J=1 TO 6:
PRINT INVERSE 1;AT 2*I+6,6;" "
: IF NOT USR men THEN PRINT AT
2*I+6,6;" ": IF USR men THEN NE
XT J: LET a$=STR$ i: GO TO 810
806 PRINT AT 2*I+6,6;" ": NEXT
I: LET A$=INKEY$
810 IF a$="1" THEN LET cod=144
: LET l=21: GO SUB 9500: GO TO 1
000
820 IF a$="2" THEN LET cod=32:
LET l=132: GO SUB 9500: GO TO 1
000
830 IF a$="3" THEN INPUT " Esc
riba el CODE del caracter ";co
d" Numero de caracteres a defin
ir:";l: LET l=l-1: GO SUB 9500:
GO TO 1000
840 IF a$="4" THEN GO TO 9000
850 IF a$="5" THEN GO SUB 9500
: GO TO 2000
860 IF a$="6" THEN GO SUB 9500
: STOP
870 GO TO 805
1000 PRINT AT 3,0;"CORRECCION""
DIBUJO ";AT 20,0: PAPER 4: INK
7;" 1<-Interrup.":TAB 20:"Corre
gir->0 "" SPACE<-Salir":TAB 20:
```

```
"Dibujar ->P "
1010 FOR I=0 TO 1: PLOT 102+24*I
,129: DRAW 11,0: DRAW 0,-11: DRA
W -11,0: DRAW 0,11: PLOT 22+128*
I,113: DRAW 34,0: DRAW 0,7: DRAW
0,-7: DRAW 33,0: DRAW 0,-34: DR
AW 7,0: DRAW -7,0: DRAW 0,-33: D
RAW -34,0: DRAW 0,-7: DRAW 0,7:
DRAW -33,0: DRAW 0,34: DRAW -7,0
: DRAW 7,0: DRAW 0,33: NEXT I
1020 POKE 23606,96: POKE 23607,2
33: FOR i=cod TO cod+1: PRINT AT
4,28;CHR$ i: RANDOMIZE USR rut
1030 FOR j=0 TO 7: POKE (car+8*(
i-32)+j),PEEK (tabla+j): POKE ta
bla+j,0: PRINT AT 8+j,3;"
": NEXT j
1040 LET M$=M$(2 TO )+CHR$ i: PR
INT AT 18,20;M$: NEXT i: POKE 23
606,0: POKE 23607,60: GO SUB 950
0: GO TO 800
2000 PRINT AT 2,0: POKE 23606,96
: POKE 23607,233: FOR i=32 TO 16
4: PRINT CHR$ i;" ": NEXT i: PO
KE 23606,0: POKE 23607,60: PRINT
AT 20,0;" Apriete cualquier te
cla para volver al MEN
U": PAUSE 0: GO SUB 9500: GO TO
```

```
800
9000 INPUT " Escriba los caracte
res primero y ultimo"" Primero:
";p$" Ultimo:";u$: LET p=CODE p
$: LET u=CODE u$: LET s=car+8*(p
-32): LET e=car+8*(p-31)
9010 INPUT "Que nombre va a pone
rles? 10 max:";n$: CLS : SAVE n$C
ODE s,e-s: PRINT AT 10,5;"Rebobi
ne para verificar": VERIFY ""COD
E : GO TO 800
9500 REM OTRA PAGINA
9510 BEEP .2,35: PRINT AT 1,0:
FOR I=0 TO 20: PRINT b$: PAUSE 4
: NEXT I: RETURN
```



Exclusivo para lectores de

Todospectrum

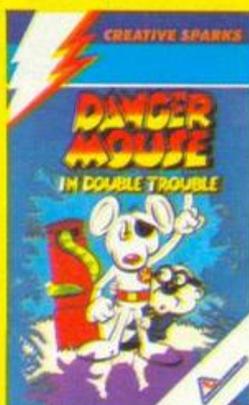
# CLUB DE SOFTWARE

Ha seleccionado ocho excelentes programas de juego para su ordenador **SPECTRUM**



**BLACK HAWK**  
Usted es un piloto de pruebas altamente entrenado para pilotar el avión más destructivo hasta ahora creado. Dos pantallas con casi 30 niveles de habilidad.

Precio: 1.550 ptas.



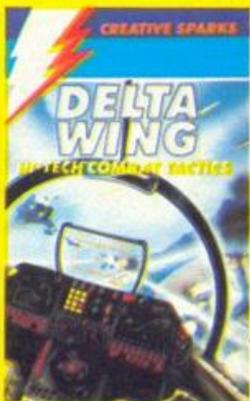
**DANGER MOUSE**  
El malvado Barón Silas Greenback ha construido un androide que Danger Mouse debe destruir sin perder un momento. Guíe a Danger Mouse y a Penfold a través de la espesa jungla hasta el lugar donde se encuentra el Barón, y después deténgale antes de que sea demasiado tarde. ¿Puede usted ayudar a Danger Mouse a salvar el mundo?

Precio: 1.550 ptas.



**TOWER OF EVIL**  
¿Arriesguese! ¿Puede usted recuperar el tesoro perdido? Una acción que discurre a través de 40 o más habitaciones le aseguran horas de juego que cautivará su atención.  
• Número de jugadores: uno.  
• Control joystick o teclado.  
• Dos pantallas de gráficos de acción rápida.  
• El VIC-20 requiere un cartucho 8K.

Precio: 1.550 ptas.



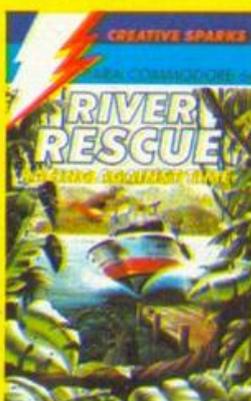
**DELTA WING**  
Delta Wing es un avanzado simulacro de combate en vuelo. Ha sido diseñado para dar la excelente "sensación" de los giros y volteretas de la alta velocidad en vuelo. Con 15 controles independientes y aviones de ataque enemigos para derribar, este programa le lleva a usted y a su Spectrum al límite. Delta Wing también permite conectar dos ordenadores Spectrum para jugar uno contra otro mediante el interface 1.

Precio: 1.550 ptas.



**SPECIAL DELIVERY**  
Santa Claus se encuentra metido en un terrible lío: se durmió en los laureles, y ahora no tiene suficientes regalos para visitar a todos los impacientes niños. Ayude a Santa Claus a recoger regalos adicionales y después a descender sobre los tejados para entregar los regalos introduciéndose por las diversas chimeneas.  
• Tres pantallas rotantes.  
• Amplia animación gráfica y música.

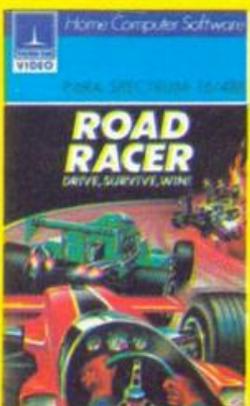
Precio: 1.550 ptas.



**RIVER RESCUE**  
Un equipo de importantes científicos han quedado atrapados en una peligrosa jungla, y la única manera de sacarlos de allí es a través del río. Usted es el propietario de una potente fuorborda, y su trabajo es rescatarlos. Navegar a través de uno de los más salvajes y torcidos ríos del mundo no resulta fácil pero usted debe dirigir su magnífica nave corriente arriba sorteando pequeñas islas evitando cocodrilos y leños flotantes a su paso. Disfrute la rápida y excitante acción en este juego de alta energía. La acción en la pantalla rotante y el magnífico sonido y gráficos le aseguran largas horas de diversión.

Precio: 1.550 ptas.

Recorte y envíe este cupón HOY MISMO a: **INFODIS, S.A. - Bravo Murillo, 377-5.º-A. 28020 MADRID**



Precio: 1.550 ptas.

**ROAD RACER**  
Adelanta al primer coche y a otro más, su pulso se acelera y la tensión crece... ¿podrá tomar la delantera y ganar el trofeo?



Precio: 1.550 ptas.

**GOLD RUSH**  
La emocionante aventura de la búsqueda de oro en un extraño planeta llamado Orón. ¿Qué peligros le aguardan...?

## ENVIE A MI DOMICILIO

LA CASSETTE O CASSETTES RELACIONADAS A CONTINUACION EL IMPORTE LO ABONARE:

POR CHEQUE  CONTRA REEMBOLSO  CON TARJETA DE CREDITO

American Express  Visa  Interbank  Fecha de caducidad \_\_\_\_\_

Número de mi Tarjeta

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Nombre \_\_\_\_\_

Dirección \_\_\_\_\_

Ciudad \_\_\_\_\_ D.P. \_\_\_\_\_

Provincia \_\_\_\_\_

cantidad	producto	ptas.	total

SIN MAS COSTOS DE ENVÍO

# EL LOGO EN LA ESCUELA

*Y seguimos con el Logo, analizando en esta ocasión la aritmética y la gestión de pantalla de este curioso lenguaje, para terminar con un pequeño repaso de la bibliografía más destacada hasta el momento*

## La aritmética del Logo

**E**L Logo de Sinclair opera con números racionales enteros y con números fraccionarios de hasta siete decimales. La suma, la resta, la multiplicación y la división pueden realizarse de forma "infija", es decir, intercalando el operador (+ - \* /) entre las cantidades. Además, la suma, la división y la mul-

tiplicación pueden ser realizadas de forma "prefija" en cuyo caso utilizaremos los primitivos SUM, DIV y PRODUCT, precedidos por los operandos. Ejemplo:

```
PRINT 4.5 + 5.5
10
PRINT SUM 4.5 5.5
10
```

Obviamente, también respeta la jerarquía de las operaciones. Per-

mite calcular raíces cuadradas y funciones trigonométricas en donde los ángulos deben ser expresados en grados. Ver figura 1.

Las operaciones INT y ROUND nos otorgan ambas la parte entera de un número pero de manera diferente. INT siempre nos elimina la parte decimal del número prescindiendo de su valor. ROUND nos "redondea" teniendo en cuenta el valor decimal.





el valor del resto cuando a es dividido por b.

### Gestión de la pantalla

Existen dos modos de trabajar en pantalla: texto y gráfico. Al cargar el Logo, la pantalla se inicializa en modo texto con 22 líneas disponibles y SCROLL para paginar.

En modo gráfico, donde interviene la tortuga, las 22 líneas superiores se destinan a gráficos dejando las dos líneas inferiores para la introducción de instrucciones.

Las opciones del BASIC del Spectrum en lo referente a colores, brillo, FLASH, OVER, etc., se mantienen con alguna variante. Por ejemplo, cuando usamos SETTC [n n] podemos especificar el color del fondo y del borde simultáneamente.

SETCURSOR [a b] sitúa el cursor en las coordenadas especificadas de línea y columna cuyos rangos son de 0 a 21 y de 0 a 30 ya que se utiliza un carácter (!) para indicar el final de cada línea.

En modo gráfico, la tortuga puede desplazarse libremente por la pantalla. Si el parámetro de desplazamiento es superior a las dimensiones de trabajo de la pantalla, la tortuga reaparece en el lado opuesto sobre el mismo eje hori-

zontal o vertical. Si su dirección es un ángulo, dibujará rayas oblicuas.

Pero esta facultad puede alterarse de dos maneras distintas. Con FENCE podemos acotar el límite de acción de la tortuga a las dimensiones de la pantalla. Si el argumento para el desplazamiento es superior a estos límites, nos impedirá moverla y nos dará el mensaje de que se encuentra fuera de éstos.

Para poder movernos fuera de la pantalla y que la tortuga continúe obedeciendo nuestras órdenes, utilizamos el comando WINDOW. En este modo podremos mover a la tortuga entre + 32767 y - 32768.

Si deseamos volver a la situación original basta teclear el comando WRAP. Ver figura 2.

### El editor

Para definir procedimientos existen dos posibilidades: modo Texto y modo Editor. Desde el modo texto el primitivo TO más el nombre del procedimiento indica que vamos a introducir instrucciones para definirlo, aunque sin posibilidades de corrección cada vez que entremos una línea. Por ello, si abordamos procedimientos complejos conviene hacerlo en el modo editor. EDIT " nombre del procedimiento, nos deja en panta-

Nombre de la función	Significado
SINE n (SIN n)	Seno
COSINE n (COS n)	Coseno
TANGENT n (TAN n)	Tangente
COTANGENT (COT n)	Cotangente
ARCSIN n	Arcoseno
ARCCOS n	Arcocoseno
ARCTAN n	Arcotangente
ARCCOT n	Arcocotangente

Figura 1. Funciones trigonométricas.

Existen dos operaciones más con las que podemos trabajar y que son de suma importancia. El azar, que está también presente con la instrucción RANDOM, se utiliza de forma diferente a la que los usuarios del Spectrum están acostumbrados. Su notación es RANDOM n. Si n es un número entero positivo, nos devuelve un número al azar entre 0 y n - 1. Por ejemplo, RANDOM 6 nos puede dar al azar los valores 0, 1, 2, 3, 4 ó 5.

A diferencia de la aritmética del BASIC del Spectrum, tenemos en el Logo la función resto (REMAINDER a b), que nos devuelve

Nombre de la instrucción	Descripción	Significado
BRIGHT n	Igual que en el BASIC. Ver manual del Spectrum.	Brillo
FLASH	Igual que en BASIC. Ver manual del Spectrum.	Centelleo
INVERSE	Igual que en el BASIC. Ver manual del Spectrum.	Inverso
OVER	Igual que en el BASIC. Ver manual del Spectrum.	Superpone
NORMAL	Anula INVERSE y FLASH.	
CLEARTEXT CT	Borra la pantalla en modo texto y las dos últimas líneas en modo gráfico.	Limpia textos
TEXTSCREEN TS	Coloca el cursor en el margen superior izquierdo e inicializa la pantalla en modo texto.	Modo texto
CLEARSCREEN	Limpia la pantalla en modo gráfico ubicando la tortuga en su posición original.	Limpiagráficos

Figura 2. Comandos para tratamiento de pantalla.

lla al primitivo TO y su nombre listos para introducir las instrucciones y con la posibilidad de insertar o borrar cualquier carácter desplazando el cursor libremente por toda la pantalla.

Podemos definir en este modo varios procedimientos simultáneamente. Las opciones del editor para mover el cursor, borrar e insertar líneas o caracteres son mucho más complejas que las del BASIC. Incluso si el texto ocupa más de una pantalla podemos movernos hacia las páginas anteriores o posteriores.

Otras cuestiones generales que son de interés para comentar y con las que deseamos acabar esta primera parte para adentrarnos luego en la gramática del Logo son:

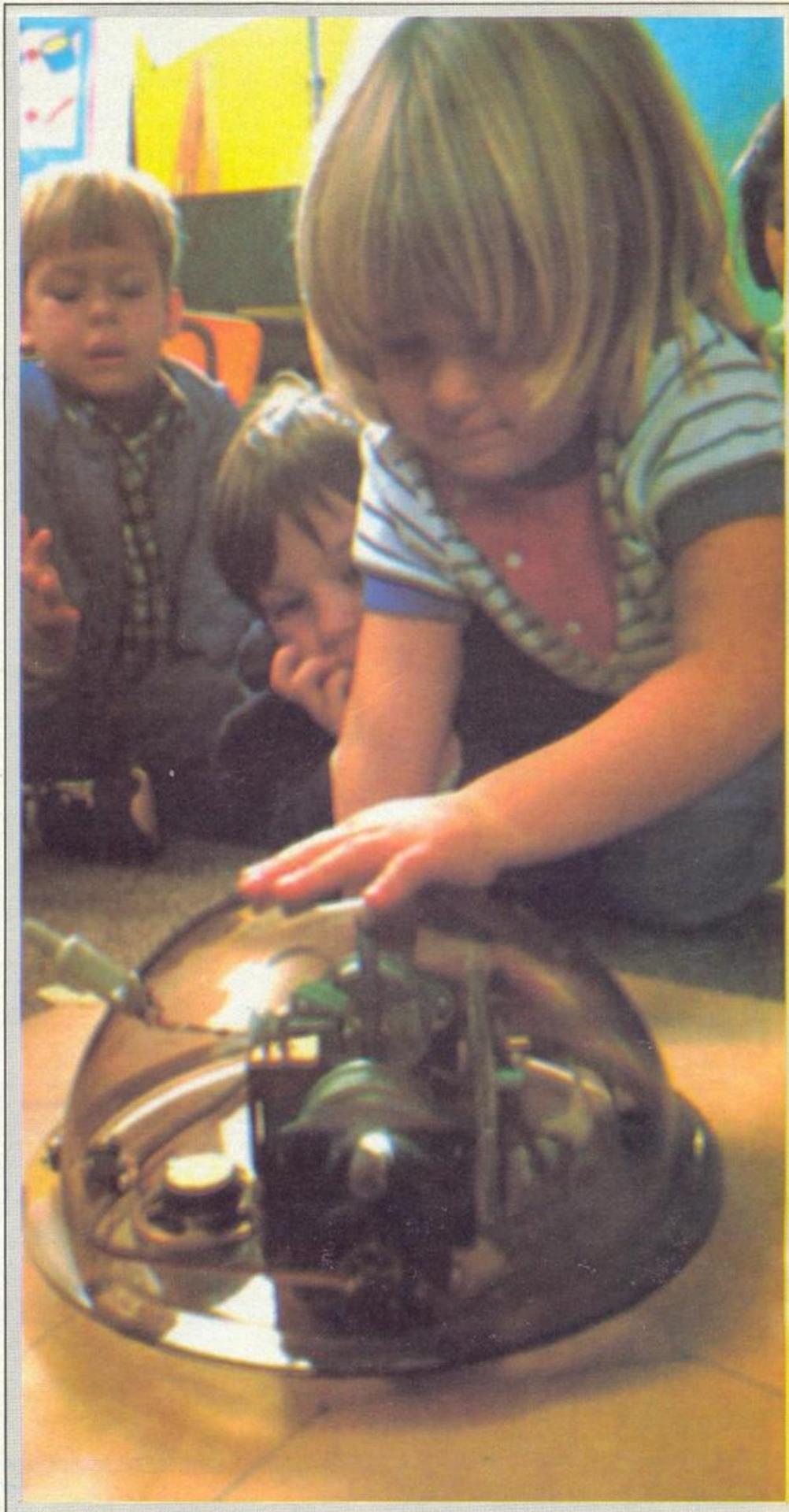
— El comando SOUND, con los parámetros de duración y altura nos permite hacer sonidos. La duración nos la da en segundos y la altura en semitonos. Los parámetros para la duración están entre 0 y 255, y para la altura entre -62 y 75. Con el siguiente procedimiento podemos hacer que cada tecla tenga un sonido.

```
TO CANTAR
SOUND SE 0.5 (ASCII
RC)-65
CANTAR
END.
```

— STARTROBOT nos permite que los comandos de movimiento de la tortuga los podemos aplicar a un robot mecánico que se encuentre conectado al Spectrum. Con STOPROBOT volvemos a la situación original. De no haber un robot conectado interpretará esta instrucción como el nombre de un procedimiento que hemos almacenado.

— Las instrucciones PEEK y POKE del BASIC se encuentran también en el Logo pero con nombres diferentes. En el primer caso .EXAMINE y la dirección nos da el contenido en decimal. Y con .DEPOSIT, la dirección y el contenido en decimal podemos definir caracteres que luego podremos imprimir.

Miguel Figini,  
Manuel Tristán-Polo  
y Paco Rivière Tell



# COMO DIBUJA LA TORTUGA

**Veamos cómo dibujar con un ejemplo concreto:** Un cielo estrellado con una luna. Con esto podréis ver cómo pueden combinarse distintos procedimientos en uno más complejo.

Para hacer la luna necesitábamos arcos de círculo. Definimos el procedimiento arco en el que el avance y el giro eran variables. Así podríamos controlar la curvatura de la silueta de la luna.

```
ARCO : avance : giro
>REPITE 15 [AV : avance
GD : giro]
>FIN
```

La repetición de 15 la elegimos porque sí! y experimentamos con diferentes parámetros de avance y giro hasta encontrar una luna que nos gustara.

La filosofía del Logo es más constructiva que teórica, por eso decidimos hacerlo así.

```
PARA LUNA
>SINLAPIZ AV 10 CONLA-
PIZ GD 90; orienta la tortuga an-
tes de empezar el primer arco.
>ARCO 4 14; primer arco.
>SL CENTRO AV 10 CL GD
130; primero orienta antes del se-
gundo.
>ARCO 2.4 8; segundo arco.
>FIN
```

La primitiva CENTRO sitúa a la tortuga en el centro de la pantalla y orientada hacia arriba.

Con esto ya teníamos la luna en el centro de la pantalla.

Decidimos hacer estrellas de diferentes tamaños.

```
? PARA ESTRELLA
>ASIGNA "tamaño AZAR 8
>REPITE 5 [AV : tamaño GD
144]
>FIN
```

AZAR 8 hizo que la variable tamaño oscilara entre 1 y 8. Así tuvimos una estrella de cinco puntas (averiguar por qué el giro de 144 nos da una estrella de cinco puntas y de tamaños diferentes). Pero un cielo tiene muchas estrellas, nuestro problema era poderlas «repartir».

Decidimos situar la tortuga al azar en la pantalla y hacerla dibujar una estrella cada vez.

```
? PARA ESTRELLAS
>REPITE : 100 [SL GD AZAR
360 AV 15+AZAR 70 CL ES-
TRELLA]
>FIN
```

GD AZAR 360 AV 15 + AZAR 70 nos situaba la tortuga aleatoriamente cada vez que dibujaba una estrella. (¿Por qué le sumamos 15 al avance?)

Cien era la cantidad de estrellas que queríamos que hubiera.

Finalmente realizamos el procedimiento cielo:

```
? PARA CIELO
>LUNA
>ESTRELLAS
>FIN
```

En este procedimiento las estrellas pueden situarse sobre la luna. ¿Cómo lo evitaríais? Y ¿si quisiéramos ubicar la luna en otro lugar de la pantalla?

Con las pocas instrucciones que hemos visto se puede hacer perfectamente (ver pantallas).

Existe la posibilidad de que definamos un procedimiento que se

utilice a sí mismo. Esto se llama recursividad. Por ejemplo, definamos ESTRELLAS de otra manera:

```
? PARA ESTRELLAS
>SL GD AZAR 360 AV
15+AZAR 70 CL ESTRELLA
>ESTRELLAS
>FIN
```

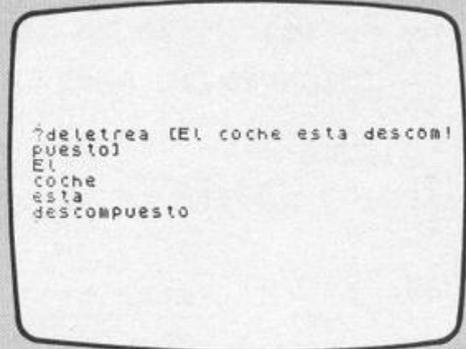
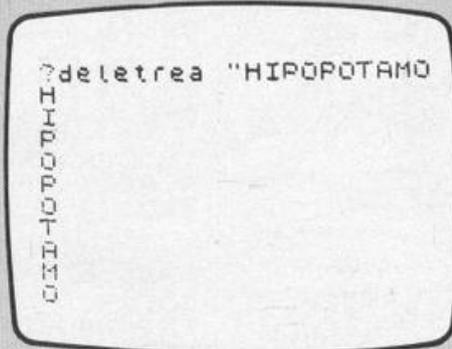
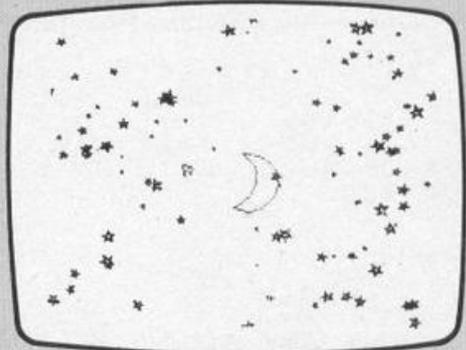
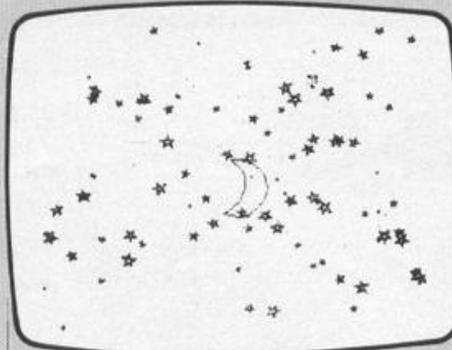
El procedimiento estrellas se llama a sí mismo y cada vez que se ejecuta va quedando pendiente la realización del FIN de cada llamada. Si no ponemos ninguna condición que lo detenga nunca acabará de dibujar estrellas.

## Palabras y listas

Logo no sólo ordena qué hacer a la tortuga, también manipula palabras y listas. Una palabra es una serie de caracteres. Por ejemplo:

```
"Tortuga
"Logo
"ZXXY24
```

Las comillas identifican una palabra. Una lista está compuesta por una serie de palabras u otras listas. (Pantallas inferiores.)



[Col patata lechuga ajo]  
 [Esta lista contiene [Esta lista contiene]]

El corchete es usado para acotar las listas. Podemos crear variables cuyo contenido sean listas y palabras.

ASIGNA "animal" hipopótamo  
 IMPRIME : animal  
 Responde: HIPOPOTAMO  
 Análogamente:  
 ASIGNA "vegetales" [col patata ajo cebolla]  
 IMPRIME : vegetales  
 Responde: COL PATATA AJO CEBOLLA

Disponemos de primitivas Logo que permiten añadir o quitar elementos a una palabra o lista.

IMPRIME MENOSPRIMERO : animal  
 Responde: IPOPOTAMO

Tengamos en cuenta que los dos puntos (:) indican el contenido de una variable.

IMPRIME MENOSPRIMERO "animal"  
 Responde: NIMAL

Es decir, toma animal como palabra (literal), y no como variable.

IMPRIME MENOSPRIMERO : vegetales

Obtenemos: PATATA AJO CEBOLLA

De igual manera, la orden PRIMERO nos da el primer carácter de una palabra o la primera palabra de una lista.

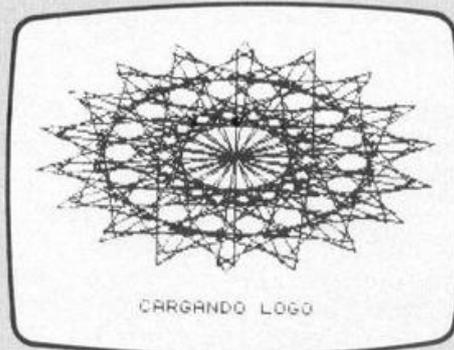
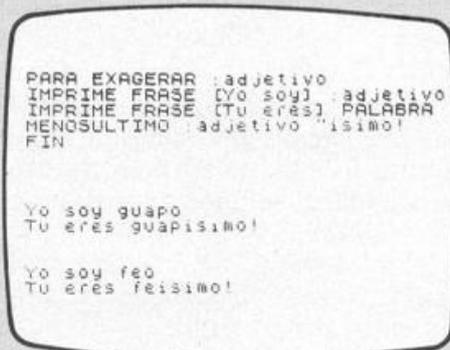
IMPRIME PRIMERO : animal  
 Responde: H (de hipopótamo, contenido de animal)

Definamos un procedimiento para deletrear palabras y descomponer frases en palabras:

```
? PARA DELETREA : texto
> SI VACIA? : texto [PARATE]
> IMPRIME PRIMERO : texto
> DELETREA MENOSPRIMERO : texto
> FIN
```

Este procedimiento es recursivo. Tiene un parámetro asociado que contendrá la palabra o lista a deletrear.

Hemos querido dar sólo un somero vistazo a algunas de las potencialidades del Logo. Muchas otras cosas han quedado por explicar. Nuestra principal intención es animaros a penetrar en el fascinante mundo de la tortuga, y que descubráis con ella un nuevo sentido a la programación y el uso de los microordenadores. Sobre todo animar a los más pequeños para que con la tortuga puedan comenzar a utilizar el microordenador ya que, por primera vez, tienen un lenguaje que pueden entender y con el que pueden crecer. ★



Esta lista esta hecha con la instrucción POALL, que imprime los títulos y definiciones de todos los procedimientos y el valor de cada variable en el área de trabajo

```
?
TO JARDIN
CASA
PU LT 150 FD 50 RT 120 PD
ARBOLES
VALLA
PU FD 100 LT 90 FD 15 RT 90 PD
SOL
PR [BIENVENIDO A MI JARDIN]
END

TO SOL
HT
REPEAT 26 [fd 15 bk 15 rt 15]
END

TO VALLA
REPEAT 45 [fd 10 bk 10 rt 90 fd 5 lt 90]
END
```

```
TO ARBOLES
ARBOL
PU LT 90 FD 30 LT 90 FD 50 RT 15
0 PD
ARBOL
END

TO ARBOL
FD 50
REPEAT 36 [fd 30 bk 30 rt 10]
BK 50
END

TO TRIANGULO
RT 30
FD 45 RT 120
FD 45 RT 120
FD 45 RT 120
END

TO CUADRADO
REPEAT 4 [fd 45 rt 90]
END

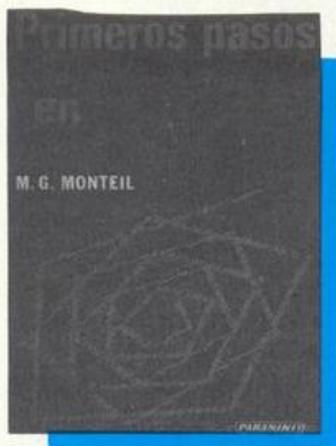
TO CASA
CUADRADO
FD 45
TRIANGULO
END
```

# LOGO-LIBROS

**Título:** Primeros pasos en Logo.  
**Autor:** Marie Gaelle Monteil  
**Editorial:** Paraninfo, S. A.  
**Páginas:** 90.

El libro que presentamos tiene un interés especial para todos aquellos usuarios (grandes y pequeños) que desean comenzar a aprender Logo. La autora nos ofrece en este pequeño manual un método de aprendizaje fácil y adaptable a todos los ordenadores que incluyan este lenguaje de programación (Spectrum, Commodore, Apple, etc.). Los objetivos del manual son, por tanto, introducir al lector en el estudio y aprendizaje del Logo.

El manual está dividido en dos partes, en la primera se enseñan las posibilidades del Logo en modo texto; y en la segunda, en modo gráfico, resultando bastante útil y

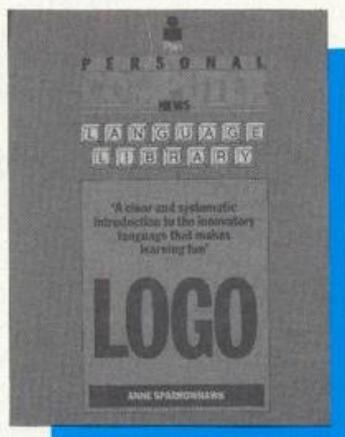


fácil de manejar. Las explicaciones de las instrucciones y procedimientos de Logo no plantean problemas al usuario (hemos de recordar que este manual va dirigido "también" para el aprendizaje de los niños). Todas ellas vienen en español, pero incorporan el término original en inglés, lo que permitirá que los usuarios cuyas máquinas sólo admitan las instrucciones en inglés no tengan dificultad en utilizarlas. También es necesario tener en cuenta otro dato que es las diferencias que existen entre los distintos Logo implementados en los ordenadores. La autora sólo recoge aquellos comandos y términos más comunes, explicando con más detalles los comandos, sintaxis y procedimientos específicos de cada ordenador al final.

**Título:** Logo  
**Autor:** Anne Sparrowhawk  
**Editorial:** Personal Computer News  
**Páginas:** 165.

Logo es un interesante libro en versión inglesa orientado a la educación y diseñado por Seymour Papers en los años 60.

En este libro se recoge una gran variedad de ideas y sugerencias que aportarán una gran utilidad al alumno cuando comience a desarrollar sus primeros programas. Estos primeros pasos con Logo comenzarán con un aprendizaje de las formas y gráficos básicos (cuadrados, círculos, estrellas, etc.)



hasta iniciarse en la programación estructurada y el diseño.

Se incluye, también, un análisis de los puntos específicos que caracterizan a este lenguaje en relación con otro lenguaje muy conocido, el BASIC. Aunque tanto uno como otro plantean una dificultad, la variedad de dialectos que existen de estos lenguajes; máquinas como Apple II, BBC Micro, Sinclair Spectrum, Commodore 64 y Atari 600XL, tienen su "propio Logo". Debido a esto, algunos comandos utilizados en este manual pueden ser efectivos sólo en determinadas máquinas.

Por último, la autora recoge diversas razones que cree han motivado la aceptación del Logo para ser usado en la enseñanza, en lugar de otros lenguajes.

**Título:** Aprendiendo con Logo  
**Autor:** Daniel Watt  
**Editorial:** MacGraw-Hill  
**Páginas:** 360.

La versión utilizada por Watt es la suministrada por Terrapin Inc. y Krell Software, Inc. para los ordenadores Apple II y compatibles y el Texas Instruments. El libro recoge también una serie de aclaraciones para su utilización con la versión de Apple Logo. Esto implica que los usuarios que vayan a utilizar este manual con otras versiones de Logo y otras máquinas deberán tener en cuenta las posibles diferencias que puedan existir entre las distintas versiones.

El manual está estructurado en tres partes, cada una de ellas está compuesta por una serie de capítulos, que presentan un nivel de difi-



cultad diferente, pensando en su utilización por niños de distintas edades. Pero esto, lógicamente, no quiere decir que cada parte del libro pueda ser leída de forma aislada, sino que los capítulos "recomendados" a los más pequeños son los que menos dificultad presentan, sin perjuicio de que los mayores tengan que asimilarlos del mismo modo. Las instrucciones están expresadas en inglés pero se recoge también la traducción al español de todas ellas.

Como el libro es bastante extenso el propio autor recomienda a los usuarios comenzar el aprendizaje con calma y de forma ordenada, en el sentido de trazarse una guía del trabajo a realizar. Este sistema, como hemos podido comprobar es bastante práctico. ★

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  
28020 - MADRID

# Descubrimiento de un nuevo lenguaje:

# PASCAL

**E**n este cuarto capítulo dedicado al lenguaje Pascal explicaremos la parte correspondiente a los subprogramas (equivalente a la instrucción GO SUB del BASIC), que en Pascal se dividen en procedimientos y funciones.

Todos conocemos más o menos lo que es un procedimiento. Ocurre muchas veces que dentro de un programa existen una serie de instrucciones que se repiten un número determinado de veces. Sería bastante incómodo tener que escribirlas en el programa cada vez que apareciesen. Lo que hacemos, por tanto, es escribir ese grupo de sentencias "una sola y única vez", bajo la forma de subprograma.

Debemos decir también que incluso las técnicas modernas de programación nos indican que un programa es mucho más legible y fácil de realizar estando bien estructurado. Según este enfoque, si una serie de sentencias realizan una tarea específica y bien definida dentro de un programa, podemos separar este conjunto de sentencias formando un subprograma.

Una definición formal de subprograma sería una parte autónoma del programa que realiza una función o tarea específica, la cual puede ser llamada (utilizada) desde otras partes del programa, si esto se considera necesario para llevar a cabo la perfecta ejecución del mismo.

Visto un poco por encima la definición general de un subprograma, pasamos a comentar los subprogramas dentro del Pascal.

Hemos dicho antes que los subprogramas en Pascal se dividen en procedimientos y funciones. Comencemos con los procedimientos.

Según el esquema general de un programa Pascal, cualquier proce-

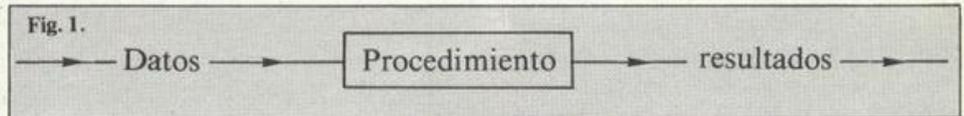
dimiento que utilizemos deberá ser declarado en la parte correspondiente a la declaración de procedimientos y funciones que hemos expuesto en capítulos anteriores.

Tenemos que identificar nuestros procedimientos por un nombre, el cual utilizaremos para hacer la llamada cuando corresponda dentro del programa principal.

El trabajo de un procedimiento es manipular una serie de datos para obtener unos resultados.

Vemos, por tanto, que un procedimiento tiene la misma estructura que un programa.

```
PROCEDURE nombre (lista de parámetros);
Parte de declaración variables procedimiento
BEGIN
Cuerpo del procedimiento
END;
```



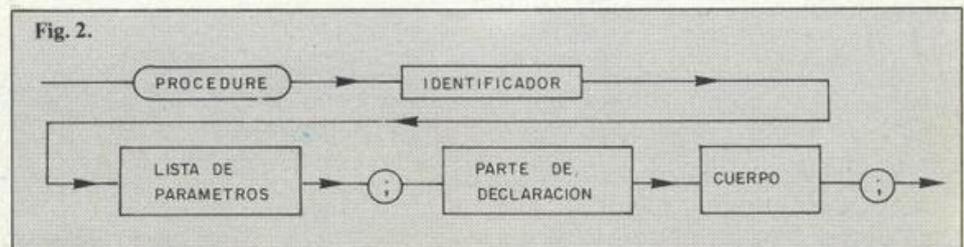
Estos datos y resultados del procedimiento son los llamados parámetros del procedimiento. Podemos clasificar estos parámetros en tres clases bien diferenciadas (ver diagrama sintáctico en figuras 1 y 2):

- a) Parámetros de entrada: serán los datos que suministraremos al procedimiento.
- b) Parámetros de salida: son los resultados que obtenemos al ejecutar el procedimiento.
- c) Parámetros de entrada/salida: son datos de entrada que son utilizados y modificados por el procedimiento, volviendo a darlos como resultados.

Un procedimiento, como podemos observar en este esquema, puede utilizar variables propias, y también variables del programa principal.

Tenemos que distinguir entonces entre estos dos tipos de variables. A las variables del programa principal y que pueden ser usadas también dentro del propio procedimiento, las llamaremos variables Globales. A las variables del procedimiento en cuestión las llamaremos variables Locales. Estas variables serán usadas siempre dentro del procedimiento, y jamás se podrán usar fuera de él.

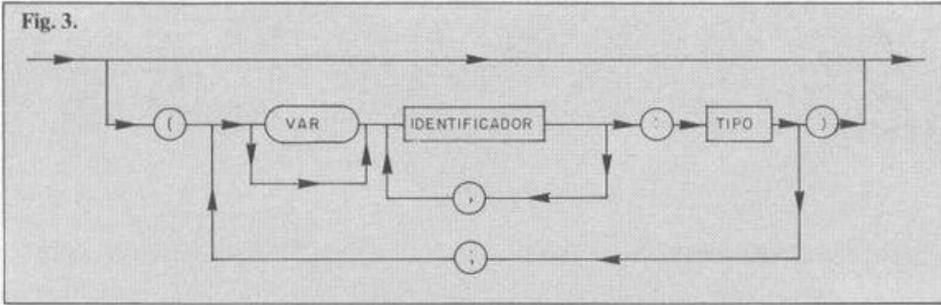
Otra parte importante dentro de



véase el diagrama de la lista de parámetros (figura 3)

los procedimientos, es que una vez concluido éste deberá devolver el

Fig. 3.



control al programa principal para que éste siga ejecutando. Como muchos de vosotros sabréis, en BASIC esto se consigue usando la palabra RETURN, al final de un GO SUB. En Pascal indicaremos que un procedimiento ha finalizado poniendo al final del mismo la instrucción END;.

Mediante esta instrucción indicamos que devolvemos el control al programa principal indicando que el subprograma ha terminado.

Nos queda por explicar dentro del diagrama de la figura 3 la palabra VAR, que como vemos va incluida en la cabecera del procedimiento. Todos los parámetros de salida o de entrada/salida de un procedimiento tienen que ir precedidos por la palabra reservada VAR, dentro de la lista de parámetros (cabecera del procedimiento). Esto quiere decir que los valores obtenidos en esas variables van a ser utilizados fuera del propio procedimiento, por el programa principal o por cualquier otro procedimiento del mismo.

A estos parámetros de entrega de resultados se les llama parámetros-variables, precisamente por ir en la cabecera del procedimiento precedidos de la palabra VAR.

Los parámetros de entrada al procedimiento (datos para calcular los resultados) no van precedidos de la palabra VAR, y se les llama parámetros-valor.

Una vez explicada la estructura sintáctica de un procedimiento, explicaremos cómo se hará la llamada al procedimiento desde el programa principal (ver figura 4).

Para comprender mejor lo que hemos visto hasta ahora, vamos a ver un ejemplo de un procedimiento.

Tenemos que realizar un proce-

dimiento al cual le vamos a pasar como datos de entrada dos números, y nos va a devolver como resultado cuál de los dos números es el mayor.

La programación de este procedimiento será:

```
PROCEDURE mayor (N1, N2:
REAL: VAR Z: REAL);
BEGIN
IF N1 > N2 THEN Z:= N1
ELSE Z:= N2
END;
```

Como vemos, este procedimiento tiene dos parámetros de entrada, que serán N1 y N2, y un parámetro de salida Z, siendo los tres de tipo REAL.

Vemos también que este procedimiento no tiene variables locales, y que lo único que realiza es mediante una sentencia condicional comparar los dos números que le hemos dado, asignando el mayor de ellos a nuestro parámetro de salida Z.

El programa principal tendrá que leer dos números y pasárselos al procedimiento, el cual le devolverá en Z cuál de los dos es mayor.

El esquema del programa principal será:

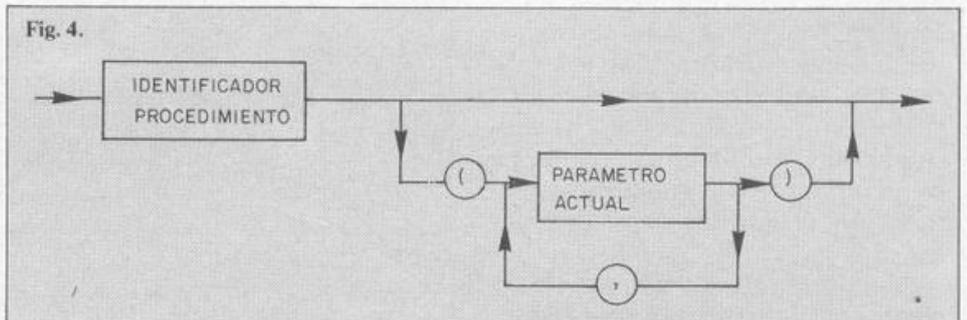
```
PROGRAM EJEMPLO;
.....
PROCEDURE MAYOR (...);
.....
BEGIN (*PROGRAMA PRINCIPAL*)
.....
READ (A, B);
MAYOR (A, B); (*llamada al procedimiento mayor*)
WRITELN ('El mayor es: ', Z)
END;
```

Comentemos en líneas generales lo que hace el programa: lee dos números A y B. Se hace una llamada al procedimiento mayor, pasándole como parámetros los números anteriormente leídos A y B (parámetros actuales con los que va a trabajar el procedimiento). Este procedimiento asignará a N1 el n.º A y a N2 el n.º B, calculando cuál de los dos es el mayor, guardando este valor en su parámetro de salida Z.

Como habréis podido observar, los números en el programa principal se llaman A y B, mientras que en el procedimiento se llaman N1 y N2, pero siguen siendo los mismos números.

Por último, reseñar que el Pascal tiene ya procedimientos incorporados dentro del propio lenguaje y que hemos usado ya muchas ve-

Fig. 4.



ces, como son: READ, READLN, WRITE, WRITELN, etc.

Hemos visto ya una clase de subprogramas que son los procedimientos. Pasemos a ver a continuación la otra clase de subprogramas, que son las llamadas FUNCIONES.

Una función es un procedimiento que, a partir de una lista de parámetros, permite obtener un resultado "único" como salida, a diferencia de las procedure, en las cuales podíamos tener uno, varios o ningún parámetro de salida.

Hasta ahora ya hemos visto y utilizado algunas funciones incorporadas del Pascal, tales como:

ABS: nos daba el valor absoluto de un número,  
SIN, COS, EXP, SQR, etc.

El diagrama sintáctico de una función puede verse en la figura 4 y la lista de parámetros en la figura 5.

Pongamos un ejemplo para

comprender mejor el funcionamiento de una función. Tenemos un programa en el que vamos a calcular para 10 valores de X la expresión  $\sqrt{1-X}$ .

Como va a ser un cálculo que vamos a repetir 10 veces, separamos esta parte del programa (subprograma), y como necesitamos un único valor de salida, emplearemos una función.

La declaración de la función será:

```
FUNCTION F(X:REAL): REAL
;
(*Esto se interpretaría de la forma siguiente: a la función F le pasa un valor de tipo real y esta me devuelva otro valor de tipo real.*)
BEGIN
  F:=SQRT (1-X)
END;
```

Supongamos que en el programa principal queremos calcular la expresión en cuestión para diez valores distintos de la variable X. El programa quedaría:

```
PROGRAM expresión:
.....
FUNCTION F(X:REAL): REAL
;
.....
BEGIN (*PROGRAMA PRINCIPAL*)
.....
FOR I = 1 TO 10 DO
BEGIN
  READ (A);
  Z:=F(A);
  WRITE ('El resultado es:',Z)
END;
.....
END. (*FIN PROGRAMA*)
```

## SUSCRIBASE A

# Todospectrum

(12 NÚMEROS)

### TARIFA DE PRECIOS DE SUSCRIPCION

	CORREO ORDINARIO		CORREO CERTIFICADO		CORREO AEREO		CORREO AEREO-CERTIF.	
	PTAS.	\$	PTAS.	\$	PTAS.	\$	PTAS.	\$
ESPAÑA .....	3.000	21	3.273	23	3.055	22	3.333	24
EUROPA, MARRUECOS, TUNEZ, TURQUIA, ARGELIA Y CHIPRE .....	3.456	25	4.272	31	3.600	26	4.418	31
COSTA RICA, CUBA, CHILE, PARAGUAY Y REP. DOMINICANA .....	3.396	24	4.212	30	4.164	30	4.980	36
GIBRALTAR Y PORTUGAL .....	3.264	23	4.080	29	3.149	22	3.965	28
FILIPINAS .....	3.264	23	3.540	25	3.775	27	4.050	29
RESTO DEL MUNDO .....	3.456	25	4.272	31	4.224	30	5.040	36

## CUPON DE PEDIDO

### Todospectrum

Recorte y envíe este cupón a: EDISA. Lopez de Hoyos, 141 - 28002 - MADRID

El importe lo abonaré: POR CHEQUE  CONTRA REEMBOLSO

CON TARJETA DE CREDITO  American Express  Visa  Interbank

Número de mi Tarjeta:

Fecha de caducidad: \_\_\_\_\_

NOMBRE \_\_\_\_\_

DIRECCION \_\_\_\_\_

CIUDAD \_\_\_\_\_ D.P. \_\_\_\_\_ PROVINCIA \_\_\_\_\_

Por supuesto, la variable del programa principal A, para que el programa se ejecute sin error, deberá ser compatible (del mismo tipo) que la que admite como entrada nuestra función. Por tanto, A deberá ser de tipo REAL.

La llamada a la función desde el programa principal es, como hemos podido observar, similar a la de una procedure (ver figura 7).

Para finalizar con el tema de las funciones, hemos de indicar que también debemos diferenciar las variables propias de la función (va-

riables locales) y las variables del programa principal (variables globales). En nuestro ejemplo la variable local de la función es el parámetro de entrada X, mientras que las variables globales son A y Z.

### Analogías y diferencias entre funciones y procedimientos

La principal analogía que encontramos es que ambas son sub-

programas, es decir, partes autónomas del programa que son llamadas durante la ejecución del programa principal.

La principal diferencia es que emplearemos una función cuando deseemos obtener un único resultado de salida, mientras que emplearemos un procedimiento cuando no se obtienen resultados o cuando obtenemos más de un resultado.

Por último, para finalizar con este artículo sobre los subprogramas, vamos a escribir un programa

```

AB28      10 PROGRAM GRANDE;
AB2D      20 VAR
AB34      30   A,B,Z:REAL;
AB34      40 PROCEDURE MAYOR(N1,N2:
REAL;VAR Z:REAL);
AB37      50 BEGIN
AB4F      60   IF N1>N2 THEN Z:=N1
AB67      70   ELSE Z:=N2
AB67      80 END;
AB67      90 BEGIN (*PP*)
ABD0     100   READ (A,B);
ABE4     110   MAYOR (A,B,Z);
ABF4     120   WRITE (CHR(16));
AC06     130   WRITELN('EL MAYOR DE
A:4:1, Y B:4:1);
AC56     140   WRITELN('ES:',Z:4:1)
AC78     150 END.
End Address: AC7D

```

Ordenación usando la función mayor.

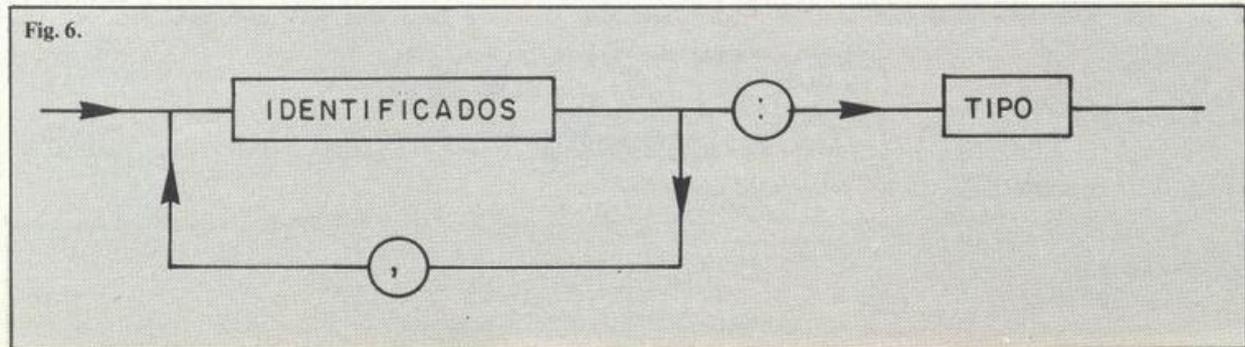
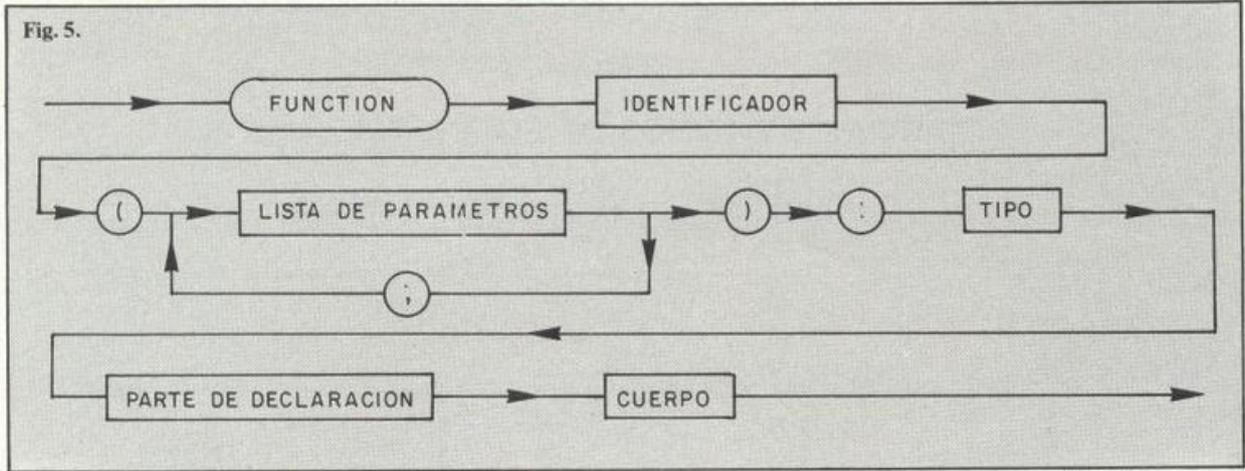
```

Run?
EL MAYOR DE 12.0 Y 34.0
ES: 34.0

EL MAYOR DE 2.57000E+02 Y 4.58
000E+02
ES: 4.58000E+02

EL MAYOR DE 1.0 Y 3.0
ES: 3.0

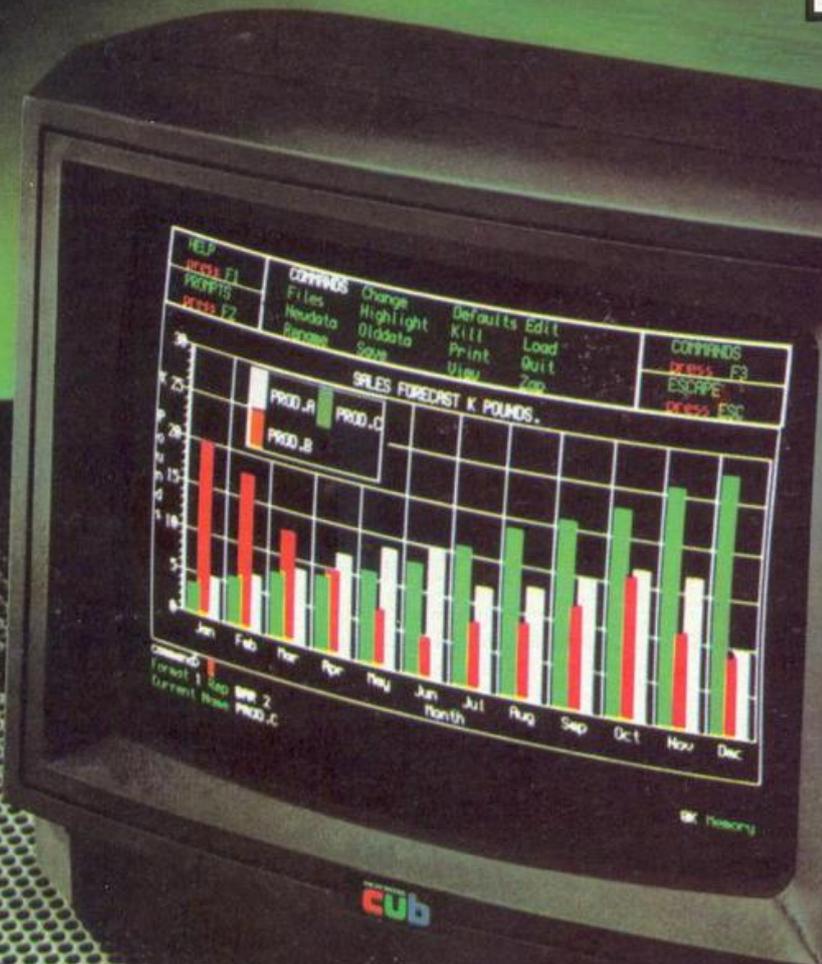
```



# ESPECIFICAMENTE PARA QL



INMEDIATAMENTE  
DISPONIBLE



## ESPECIFICACIONES

MODELO:  
CUB 1451/DQ3  
14" QL MONITOR  
Entrada RGB-TTL  
Resolución (PIXELS)  
653 (H) x 585 (V)  
DOT PITCH 0.43 mm  
Bandwidth 18 MHz  
Específicamente diseñado  
para el QL que  
aprovecha su facilidad  
única de 85 columnas.

Mod. 1451/DQ3  
P.V.P. 96.000 Ptas.



MICROVITEC 653  
**CUB**  
COLOUR DISPLAYS

DISTRIBUIDO EN EXCLUSIVA POR MULTILOGIC, Ramón de Santillán, n.º 15 28016 MADRID Tel. 458 74 75

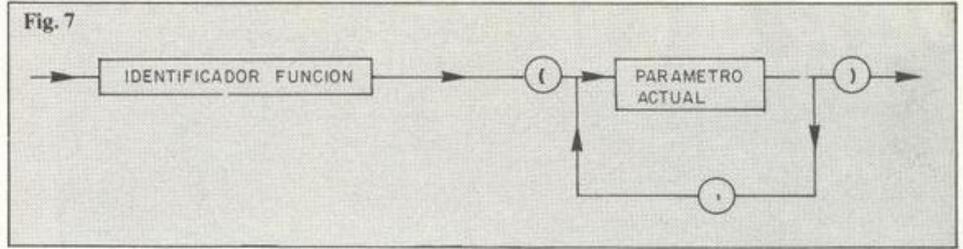


para ordenar dos números mediante funciones y procedimientos. En este caso el uso de una función o un procedimiento es indistinto.

- 1) Usando la función mayor y
- 2) Mediante el procedimiento mayor.

José R. Herreros

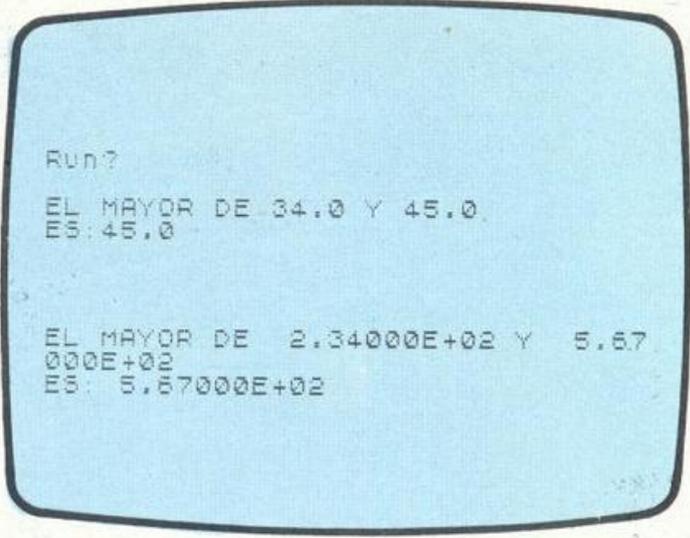
Fig. 7



```

AB004 10 PROGRAM GRANDE;
AB004 20 VAR
AB004 30 A,B:REAL;
AB004 40 FUNCION MAYOR(N1,N2:P
AB004 REAL;
AB004 50 BEGIN
AB004 60 IF N1>N2 THEN MAYOR:
AB004 70 ELSE MAYOR:
AB004 80 END;
AB004 90 BEGIN (*PP*)
AB004 100 READ(A,B);
AB004 105 WRITE(CHR(15));
AB004 110 WRITELN('EL MAYOR DE
AB004 A:4:1 Y B:4:1);
AB004 115 WRITELN('ES:',MAYO
AB004 R(A,B):4:1);
AB004 120 END.
End Address: AC65
  
```

Ordenación mediante el procedimiento mayor.



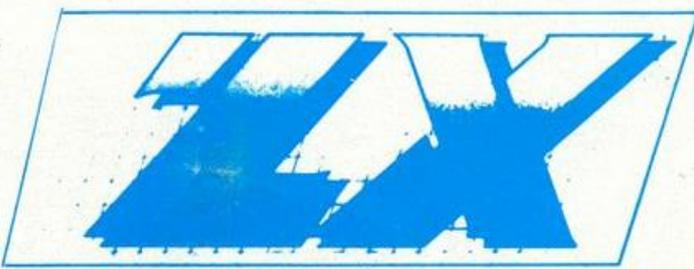
## SUSCRIBASE POR TELEFONO

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

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

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

SUSCRIBASE A



# Microtodo

## Algo más que una tienda de ordenadores.

### **Algo más en Servicio.**

Personal altamente cualificado le asesorará en todo lo relacionado con el mundo de la microinformática y la robótica, asesoramiento que continuará aún después de haberle instalado su ordenador, en su propio domicilio. Garantía total en todos sus productos.

### **Algo más en Ordenadores.**

Más de 30 marcas de ordenadores, familiares, profesionales y superprofesionales, donde poder elegir el más adecuado a sus necesidades.

### **Algo más en Complementos.**

La más completa gama de complementos imaginables: interfaces, cassettes, floppy disk, diskettes... compatibles con Apple e IBM. Telefonía sin hilos. y además disponemos de la más completa bibliografía sobre microinformática y robótica con más de 500 libros y revistas editados en varios idiomas. También podemos suscribirle en cualquier revista nacional o extranjera.

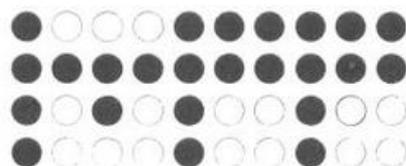
### **Algo más en Robótica.**

Somos la primera tienda en Madrid especializada en robótica. Le ofrecemos desde el más divertido Robot-juguete de 13.800 pts. hasta el más sofisticado de 1.000.000.

### **Algo más en Facilidades de Pago.**

Plazos especiales en ordenadores familiares y Leasing en ordenadores profesionales.

Todo en Microinformática



C/ Orense, 3. Tfno.: 253 21 19. 28020 - MADRID. (Entrada por jardines)



Para los amantes de los juegos ya se sabe que un *cassette* y la instrucción `LOAD ""` es todo lo que necesitan. Quienes quieren realizar trabajos más serios, especialmente de cara al manejo de archivos, se ven en la necesidad de incorporar una unidad de *microdrive* o disco, decisión que vendrá motivada fundamentalmente por el tipo de trabajo que se quiera realizar y el dinero que se desee invertir. En uno u otro caso, hay que tener siempre en cuenta que todos los programas existentes se dan en cintas y opcionalmente, para aplicaciones profesionales, en *microdrive*, pero no hay por el momento programas para disco, si bien alguna casa está estudiando esta posibilidad.

No es probable que en España llegue a darse una oferta de discos tan variada como en el Reino Unido, por lo que seguramente su elección pasará forzosamente por alguno de estos cuatro sistemas analizados. Y puestos a elegir, veamos los puntos principales a comparar:

● **Capacidad.** Este es uno de los aspectos más importantes. Los *microdrives* Sinclair tienen 90 K o

tres opciones en el caso del *Wafadrive*: 16, 64 y 128 K. En el caso del disco de *Technology Research Limited* se puede trabajar con capacidades mayores: de 100 a 640 K, y de 160 ó 640 en el caso del *Investdisk 200*. Lógicamente para grandes volúmenes de datos una unidad de disco puede evitar muchos problemas, amén de la sencillez de trabajar con un solo disco en vez de diversas cintas de *microdrive*.

Un aspecto a tener en cuenta es la posibilidad de disponer de un archivo de datos único o fraccionado en distintos soportes. Si ha de trabajar con archivos grandes, siempre será conveniente tenerlos en un solo archivo (es decir, en una sola cinta o disco) a fin de facilitar el trabajo del programador y del usuario, que de otra forma tendrá que estar pendiente del ordenador para intercambiar las cintas.

● **Rapidez.** Junto con la capacidad, es otro de los elementos decisivos. Aquí vuelven a ganar los discos: mayor superficie de grabación a la que se puede acceder de forma rápida. Para las aplicaciones domésticas resultaría ridículo utilizar una potente unidad de disco, cuan-

do un simple *microdrive* puede resolverlo en tiempos de acceso nada despreciables. Sin embargo, llevar una contabilidad, y en general cualquier aplicación que consuma bastante tiempo de ordenador, será preferible hacerlo con discos.

Las diferencias entre los distintos sistemas son notables, y una deficiente programación puede complicarlo innecesariamente. Así, si utiliza *microdrives* será más útil trabajar en memoria y realizar la grabación al final de la sesión, mientras que con disco se pueden realizar las operaciones de lectura/grabación directamente.

Otro aspecto crucial que marca una delimitación clara entre cintas y discos es la posibilidad de trabajar con los llamados archivos aleatorios. En *microdrive*, como ocurre con las cintas de *cassette*, la grabación es únicamente secuencial, por lo que el proceso es necesariamente lento y la actualización o cambio de datos de estos ficheros requiere la utilización de todo el fichero. Los discos, por el contrario, permiten la utilización de archivos aleatorios, mediante la utilización de registros (como si de los surcos de un disco de su cantante favorito

# DISKETTES PARA SPECTRUM

*Los asiduos de TODOSPECTRUM ya conocen el OPUS, o mejor dicho, el interface de Technology Research Limited que permite al Spectrum trabajar con unidades de disco compatibles IBM. A este sistema se une ahora el nuevo Invesdisk 200. Este artículo pretende analizar estos dos sistemas, en comparación con las unidades de microdrive.*

se tratase), pudiendo acceder única y exclusivamente a la información que se precise, sin tener que utilizar la restante información de un fichero. En otras palabras, se disminuyen considerablemente los tiempos de acceso a la información.

● **Precio.** Paradójicamente, el abaratamiento de los ordenadores domésticos ha dado lugar a que los periféricos, como es el caso de los

discos, tienen un precio superior al propio ordenador. Es lamentable que el abaratamiento típico de Sinclair no se haya podido ver acompañado en el caso de las unidades de discos, lo que hace que una configuración profesional de Spectrum con impresora de calidad y

potente unidad de discos, tenga un precio "excesivo" para lo que se considera un ordenador doméstico.

Las unidades de disco son más caras que las de *microdrive* o un simple *cassette*. Por contra, son más fiables y permiten mayores capacidades y menores tiempos de





# DISKETTES

precio sensiblemente inferior, ofertadas al precio de 2 libras (400 pts.) en una feria del pasado mes de febrero en Londres.

● **Software.** Las unidades de *microdrive* y discos incorporan ventajas notables, pero no se engañe: el programa se lo tendrá que hacer usted, al menos por el momento.

Diversas aplicaciones profesionales utilizan el *microdrive*, en muchos casos se vende en cintas y se ofrece la compatibilidad con *microdrive*, pero hasta el momento nadie ha utilizado el disco. Tenga presente este aspecto si usted es de los que quiere "que se lo den todo hecho".

Finalmente, si quiere nuestro consejo, procure olvidarse del *cassette*. Los cuatro sistemas analizados presentan diversas ventajas e inconvenientes y no siempre el más caro es el mejor: depende de la utilización que se quiera hacer de ellos. ★

acceso a la información. Otro aspecto a tener en cuenta es el elevado precio de los *microdrives*: 1.500

pesetas en el caso de Sinclair, frente a las 500 de un disco convencional. Afortunadamente, Sinclair lleva tiempo trabajando en unas nuevas cintas de *microdrive* de

## SISTEMAS DE ALMACENAMIENTO DE DATOS

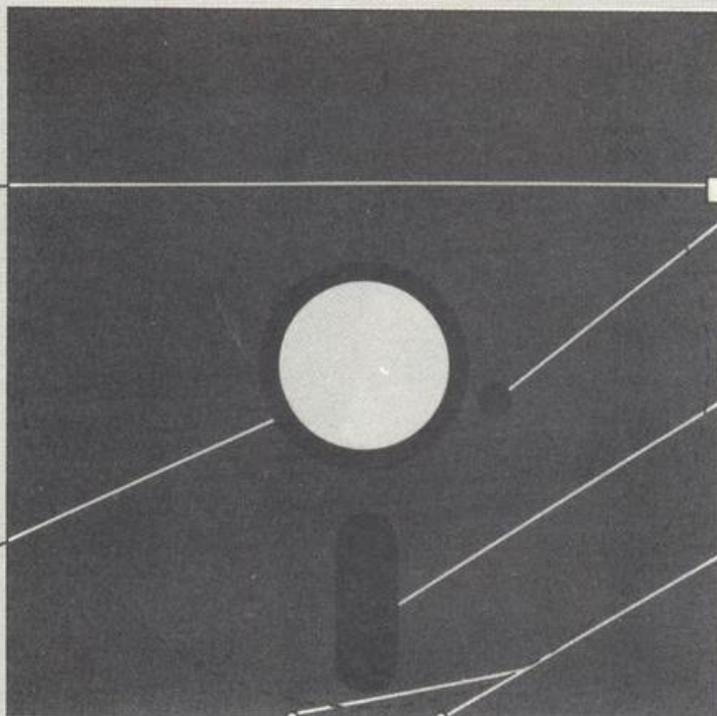
CARACTERISTICAS	ZX MICRODRIVE	WAFADRIFE	TECNOLOGY RESEARCH	INVEDISK 200
Capacidad	90 K (Format.)	16, 64, 128 (formateado)	100, 160, 400, 640 K.	160 K (simple densidad) 640 K (doble densidad)
Tipo acceso	Secuencial	Secuencial	Secuencial/aleatorio	Secuencial/aleatorio
Sistema almacenamiento	Cinta "sin fin"	Cinta "sin fin"	Disco 5 pulgadas	Disco 3 pulgadas
Comunicaciones	RS232 y red local	RS232 y Centronic	No dispone	2 conectores RS232
Ampliaciones	Hasta 8	Ninguna	Hasta 4	Hasta 4
Software	Aplicaciones profesionales: textos, contabilidad, etc.	Procesador textos	Utilidades	Utilidades
Tamaño	Interface: 23 mm × 7 mm × 2 mm Microdrive: 8 mm × 9 mm × 3,5 mm	230 mm × 110 mm × 80 mm	Interface: 12 mm × 16 mm × 25 mm Unidad disco: 15 mm × 23 mm × 4,5 mm	Interface: 30 mm × 95 mm × 100 mm Unidad disco: 67 mm × 122 mm × 157 mm
Peso	Interface: 210 gr Microdrive: 190 gr	900 gr	Interface: 210 gr Unidad: 2 Kg	Controlador: 550 gr Disco: 800 gr Alimentador: 1.350 gr
Precio	Interface 1: 15.650 ptas. Microdrive: 15.650 ptas.	48.500 ptas.	Interface: 29.850 ptas. Unidad 160 K: 55.200 ptas. Unidad 640 K: 69.500 ptas.	Sin determinar
Fabricante	Sinclair	Retronics	Tecnology Research	Timex
Distribuidor	Investrónica	Microbyte	Silog	Investrónica

# PARA SPECTRUM

## Qué es un disco flexible

**Protección antiescritura.** Cerrando la abertura se protege la información ante el borrado o escritura accidental sobre el disco. Se puede leer, pero no modificar la información. Cualquier intento de grabar información con el disco protegido genera un mensaje de error.

**Abertura central.** El disco es "atrapado" por el centro, para su giro generalmente a una velocidad de 300 revoluciones por minuto. Algunas casas refuerzan esta parte del disco para alargar su vida.



**Orificio de sincronismo.** Permite conocer el comienzo de los surcos, sirviendo de guía a la cabeza de grabación/lectura. Al desplazarse el disco, un haz luminoso pasa por los distintos orificios, proporcionando el sincronismo adecuado.

**Ventana de acceso de la cabeza de grabación/lectura.** La cabeza se desplaza sobre la superficie magnética para acceder al surco a trabajar.

**Orificios de alineamiento.** Se utilizan para asegurar la correcta colocación del disco dentro de la funda.

# ...MI ORDENADOR ES SINCLAIR, MI SERVICIO TECNICO ES HISSA...

Y es lo lógico. Si has elegido el mejor microordenador del mercado, no vas a repararlo con cualquiera.

Y recuerda que no tendrás sobresaltos con el precio.

## "COSTE ESTANDAR POR REPARACION"

ZX 81:	3.150 Ptas.
Spectrum 16K:	5.250 Ptas.
Spectrum 48K:	6.300 Ptas.



Sólo Hissa te puede garantizar la utilización de piezas originales SINCLAIR y expertos técnicos en reparación.

Acude a la delegación **HISSA** más cercana.

C/. Aribau, n.º 80, piso 5.º 1.º  
Telfs.: (93) 323 41 65 - 323 44 04  
08036 BARCELONA

C/. San Sotero, n.º 3  
Telfs.: 754 31 97 - 754 32 34  
28037 MADRID

C/. Avda. de la Libertad, n.º 6. Bloq. 1.º Entl. Izq. D.  
Telf. (968) 23 18 34  
30009 MURCIA

P.º de Ronda, n.º 82, 1.º E  
Telf.: (958) 26 15 94  
18006 GRANADA

C/. 19 de Julio, n.º 10 - 2.º local 3  
Telf.: (985) 21 88 95  
33002 OVIEDO

C/. Hermanos del Río Rodríguez, n.º 7 bis  
Telf.: (954) 36 17 08  
41009 SEVILLA

C/. Universidad, n.º 4 - 2.º 1.º  
Telf.: (96) 352 48 82  
46002 VALENCIA

Avda. de Gasteiz, n.º 19 A - 1.º D  
Telf.: (945) 22 52 05  
01008 VITORIA

C/. Travesía de Vigo, n.º 32 - 1.º  
Telf. (986) 37 78 87  
6 VIGO

C/. Atares, n.º 4 - 5.º D  
Telf.: (976) 22 47 09  
50003 ZARAGOZA

# Palos: El último es el ganador



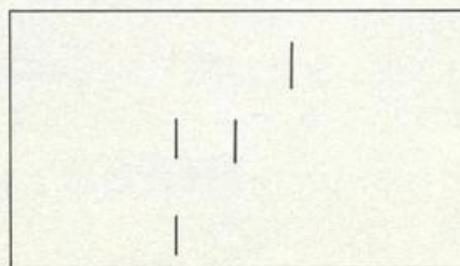
*Seguro que alguna vez ha oído hablar de este antiguo juego que ahora le ofrecemos para el Spectrum, ampliamente comentado. En esta versión, el que logra tachar el último palo es el ganador.*

Este juego para dos personas o una persona y ordenador, consiste en ir tachando sucesivamente y alternativamente palos hasta tachar el último. Los palos se disponen de tal modo que en la primera línea hay  $n$  palos en la segunda  $n-1$  en la tercera  $n-2$  y así hasta la última en la que hay un sólo palo (en el ordenador  $n$  varía aleatoriamente entre 6 y 10 en cada partida).

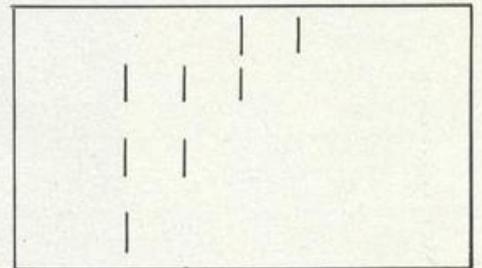
Para tachar los palos, se comienza por la esquina superior izquierda y se van tachando en número máximo de tres alternativamente por cada uno de los jugadores. El número de palos máximo que se puede tachar cada vez varía aleatoriamente a lo largo de la partida, no obstante para fijar ideas y ver la estrategia a adoptar mantendremos

fijo en tres el número máximo de palos a tachar y en 6 el número de palos de la fila superior.

Con las reglas anteriormente fijadas, el problema se reduce a calcular en cada momento el número de palos totales, y tachar los necesarios para que en cada momento los palos restantes sean siempre múltiplos de 4. Veamos esto en un ejemplo particularmente sencillo. Supóngase el siguiente caso:



Evidentemente aquí estamos en una situación en que los palos que quedan son múltiplos de cuatro y ocurre que el primero que tacha pierde. Examinemos esto mismo en una situación ligeramente más compleja.



Volvemos a tener un número de palos múltiplos de 4, tache los que tache el primero pierde pues el segundo en tachar siempre podrá ponerse en el caso primeramente analizado. Este, es el punto esencial de la cuestión, como el número máximo de palos a tachar es tres, una vez se ha conseguido llegar a una situación en que los palos que quedan son múltiplos de 4 siempre se podrá continuar en ella para lo cual bastará tachar en cada momento  $4 - m$  palos, siendo  $m$  el número de palos tachados por el adversario. Cuando el número de palos de la primera línea es 6 el número total de palos será:  $(6+1) * 6/2 = 21$ .

Dado que de una línea a otra se disminuye en un palo con lo que deberíamos sumar:  $6 + 5 + 4 + 3 + 2 + 1$ .

Que se puede agrupar de la siguiente forma:

$$(6+1) + (5+2) + (4+3) = 7 \times 3 = 21.$$

Y que es justamente como se calcula el número de palos totales en las líneas 12 y 15 (el número de palos de la primera línea se introduce en la variable FIN en la línea 10).

# Programas

36

Con 6 palos en la primera línea, si el primero en jugar tacha un palo gana, pues deja por tachar 20 palos que es múltiplo de 4. Por el contrario, si en la primera línea hay 7, el número total de palos es 28 como este número es múltiplo de 4 tache lo que tache el primer jugador, éste pierde, pues jamás podrá dejar esta situación llegando indefectiblemente de esta manera a la primera posición considerada en la que ya se ha demostrado que el primero que juega pierde.

Veamos ahora cómo "decírselo" al Spectrum. Entre las líneas 20 y 60 se dibujan los palos en la pantalla mediante un bucle FOR i = 1 TO FIN1 donde FIN1 es una variable que inicialmente tiene el valor FIN disminuyendo de 1 en 1 hasta llegar a 1 en la línea 60.

A continuación encontramos dos bloques bien diferenciados. El que va de la línea 1000 a la 1510 y que denominamos "JUGADOR"

es la subrutina donde se introduce el número de palos a tachar por parte del jugador. Entre las líneas 2003 y 2530 está la subrutina "ORDENADOR" en la que se calcula el número de palos que debe tachar el ordenador, con lo dicho anteriormente, este cálculo es trivial; en la línea 2500 se calculan los palos que quedan por tachar. Para ello previamente calculamos el número total de palos y se introducen en la variable TOT, mientras que en la variable TA se introduce sucesivamente el número total de palos tachados. En la línea 2510, se calcula el número de grupos de 4 palos que se pueden formar con los que quedan (STE es el número máximo de palos a tachar; varía entre dos y cinco siempre que se cumpla la condición  $\text{INT}(\text{RND} \times 10) + 1 = 9$  de la línea 9070). En la misma línea 2510 se calcula el número de palos a tachar NUM como la diferencia en-

tre el número de palos que quedan y el número de palos con los que se pueden formar grupos de 4. Para finalizar puede ocurrir que este número sea 0 lo que significaría que no es posible tachar el número de palos idóneos por lo que entonces se define NUM como un número entre 1 y STE aleatoriamente.

Supongamos ahora que o bien porque se introducen directamente o bien porque el ordenador lo calcula, el número de palos a tachar es NUM. Para hacerlo, recurrimos al subprograma "TACHAR" que se encuentra entre las líneas 9010 y 9100. Además, para distinguir los palos tachados entre el ordenador y el jugador se define una variable COL a la que se asigna el valor 4 en la línea 1010 que corresponde al bloque "JUGADOR" y 6 en la 2010 que pertenece a su vez al bloque "ORDENADOR" (siempre hablamos de la opción en

```
1 PRINT AT 10,0:"JUGAR CON OR
DENADOR: 1"
2 PRINT AT 15,0:"JUGAR CON OT
RA PERSONA: 2"
3 INPUT "OPCION: ";OF:CLS
4 IF OF=1 THEN INPUT "NIVEL
(1 - 3): ";NI
5 RANDOMIZE : LET STE=INT (RN
D*4)+2
6 PRINT AT 20,0:"SE FUEDEN TA
CHAR HASTA: ";STE: PRINT AT 0,30
;" "
10 LET COL=4: LET FRA=0: LET f
in=INT (RND*5)+6: LET mun=0: LET
FIN1=FIN: LET P$="FIN/2-INT (FI
N/2)": LET CO=0: LET LIN=1: LET
GAN=0: LET ta=1: LET ju=0: LET a
$=""
12 IF VAL P$=0 THEN LET TOT=(
FIN+1)*FIN/2
15 IF VAL P$<>0 THEN LET TOT=
(FIN+1)*(FIN-1)/2+INT (FIN/2)+1
20 FOR i=1 TO fin1
30 PRINT " ■ ";
40 NEXT i
50 PRINT : PRINT
60 IF fin1>=2 THEN LET fin1=f
in1-1: GO TO 20
70 LET INI=INT (RND*10)
999 REM *****
*****JUGADOR*****
```

```
*****
1000 INPUT "numero?: ";num
1001 PRINT AT 20,0:"SE PUEDEN TA
CHAR HASTA: ";STE
1002 IF NUM>ste OR NUM=0 THEN P
RINT #1:"TRAMPOSO": BEEP 1,7: P
AUSE 50: GO TO 1000
1010 IF OF<>2 THEN LET COL=4
1020 GO SUB 9000
1022 IF COL=4 AND OF=2 THEN LET
COL=6: GO TO 1026
1023 IF COL=6 AND OF=2 THEN LET
COL=4
1026 IF TA-1=TOT THEN BEEP 1,3:
PRINT #1:"TU GANAS": PAUSE 0: S
TOP
1500 IF OF=1 THEN GO TO 2000
1510 IF OF=2 THEN GO TO 1000
2000 REM *****
*****ORDENADOR*****
*****
2003 IF NI=1 AND INT (RND*10)+1>
=3 AND TOT-TA>=8 THEN BEEP .1,1
: BEEP .1,3: LET NUM=INT (RND*ST
E)+1: GO TO 2010
2004 IF NI=2 AND INT (RND*10)+1>
=6 AND TOT-TA>9 THEN LET NUM=IN
T (RND*STE)+1: GO TO 2010
2005 GO SUB 2500
2010 LET COL=6: GO SUB 9000
2030 IF TA-1=TOT THEN BEEP 1,1:
```

```
PRINT #1:"YO GAND": PAUSE 0: ST
OP
2100 GO TO 1000
2500 LET var=TOT-(TA-1)
2510 LET PAR=INT (var/(STE+1)):
LET NUM=var-par*(STE+1)
2520 IF NUM<=0 OR NUM>STE THEN
BEEP .1,-3: LET NUM=INT (RND*STE
)+1
2530 RETURN
9000 REM *****
*****TACHAR*****
*****
9010 FOR J=1 TO num
9020 LET CON=1: FOR I=FIN TO 1 S
TEP -1
9030 LET CON=CON+I
9035 IF CON=TA THEN LET LIN=LIN
+2: LET CO=0: GO TO 9045
9040 NEXT I
9050 PRINT INK COL, BRIGHT 1:AT
LIN,CO:" ■ "
9055 LET CO=CO+2: LET TA=TA+1
9060 NEXT J
9070 IF INT (RND*10)+1>=9 THEN
BEEP 1,7: LET STE=INT (RND*4)+2:
PRINT AT 20,0:"SE FUEDEN TACHAR
HASTA: "; FLASH 1:STE
9100 RETURN
```

# Programas

36



Una última aclaración, se refiere a la forma de establecer los niveles en las líneas 2003 y 2004. Como se puede ver lo único que se hace es dar una cierta probabilidad de que

el ordenador se olvide de la estrategia y de un número aleatorio volviendo inmediatamente a "JUGADOR".

*Autor: Julio Gómez Herrera*

la que el ordenador interviene inteligentemente y por tanto  $OP = 1$ ). Además, también en este bloque se define la variable TA que informa sobre el número de palos tachados y que se utiliza, aparte de los casos ya mencionados, en la determinación de la jugada final en la línea 1026 para "JUGADOR" y en la 2030 para "ORDENADOR".



## ESTAREMOS en Informat85

**Del 16 al 20  
de Abril de 1985**

**Informat 85**

Barcelona del 16 al 20 de Abril de 1985  
Hardware Equipos • Software Servicios • Telecomunicaciones Equipos • Servicios

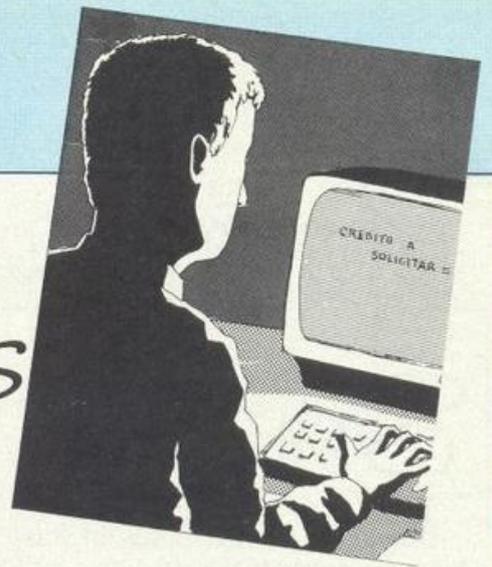
Feria de Barcelona

**Palacio y Ferial • Nivel 7 • Stand 709  
Barcelona del 16 al 20 de Abril de 1985**

# Cálculo de amortizaciones

Realizar un correcto cálculo de amortizaciones es una tarea fácil pero tediosa, especialmente si se quieren ver todas las posibilidades: ¿Cuánto será el importe de la letra? ¿Qué ocurriría si se obtiene un

tipo de interés menor?... Con este programa introduce los datos disponibles y la barra "/" en el concepto que desea calcular. Del resto se encarga su ordenador.



```

10 GO TO 130
20 FAUSE 0: IF CODE INKEY$>=46
AND CODE INKEY$<=57 THEN LET x
$=x$+INKEY$: PRINT AT 1,16;x$:
GO TO 20
30 IF CODE INKEY#=12 THEN LET
x$="": PRINT AT 1,16;"
"
40 IF x$="" THEN GO TO 20
50 RETURN
60 LET e=d/(1-(1+d)^-c): RETUR
N
70 GO SUB 60: LET a=e*b: PRINT
AT 5,16; PAPER 5; INK 0;a;" pts
": RETURN
80 GO SUB 60: LET b=a/e: PRINT
AT 7,16; PAPER 5; INK 0;b;" pts
": RETURN
90 LET g=1-b*d/a: IF g<0 THEN
PRINT AT 20,2;"No lo puedo calc
ular.Repité.": FAUSE 0: RUN
95 LET c=ABS LN g/LN (1+d): LE
T c=INT (c+.5): PRINT AT 9,16; P
APER 5; INK 0;c: RETURN
100 LET f=1e-8: LET g=a/b-b/c^2
/a
110 LET h=g: LET i=a/h: LET j=1
+h: LET k=j^-c: LET m=1-k: LET g
=g-(b-i*m)/(b-i*(c*k/j-m/h)): LE
T w=ABS (g-h): IF w>=f THEN GO
TO 110
120 LET d=g*1200: PRINT AT 11,1
6; PAPER 5; INK 0;d;" %": RETURN

130 BORDER 1: PAPER 1: INK 7: C
LEAR : PRINT AT 1,3: INVERSE 1;"
CALCULO DE AMORTIZACIONES":AT 16
,0: INVERSE 0: PAPER 2;" Tecllea
la cifra de los datos y pulsa EN
TER.Escribe ~/~ si es el apartado
que quieres calcular. "

```

```

140 LET x$="": LET a$="Letra me
nsual=": LET b$="Prestamo ="
: LET c$="Mensualidades=": LET d
$="Interes anual="
150 PRINT AT 5,0: PAPER 4; INK
0;a$: PRINT AT 7,0;b$' 'c$' 'd$
160 LET l=5: GO SUB 20: IF x$=""
/" THEN LET p=70: GO TO 180
170 PRINT " pts": LET a=VAL x$
180 LET x$="": PRINT AT 5,0;a$'
' PAPER 4; INK 0;b$: LET l=7: GO
SUB 20: IF x$="/" THEN LET p=8
0: GO TO 200
190 PRINT " pts": LET b=VAL x$
200 LET x$="": PRINT AT 7,0;b$'
' PAPER 4; INK 0;c$: LET l=9: GO
SUB 20: IF x$="/" THEN LET p=9
0: GO TO 220
210 LET c=VAL x$
220 LET x$="": PRINT AT 9,0;c$'
' PAPER 4; INK 0;d$: LET l=11: G
O SUB 20: IF x$="/" THEN LET p=
100: GO TO 240
230 PRINT " %": LET d=VAL x$: L
ET d=d/1200
240 PRINT AT 11,0;d$: GO SUB p:
FAUSE 0: CLS : GO TO 130
250 REM Alfonso Aguirre,1984

```

## CALCULO DE AMORTIZACIONES

```

Letra mensual= 725 pts
Prestamo = 58000 pts
Mensualidades= 5000
Interes anual= 15 %

```

Tecllea la cifra de los datos y pulsa ENTER.Escribe ~/~ si es el apartado que quieres calcular.

# Programas

38

## Sonamor

¿Recuerda el programa en Pascal que convertía sus números arábigos en romanos? Este también está en Pascal y hace lo contrario, por ello y con gran sentido de humor, su autor le dio este título: ¡léalo al revés!

El programa pide la introducción de un número romano (acabado en un espacio). Mediante llamadas a una procedure ANALIZAR se comprueba cada orden de magnitud. Esta procedure llama a REPETICION y LEER, que reali-

zan tareas auxiliares. El programa trabajará correctamente siempre que el número introducido sea menor que 4000.

*Autor: Carlos Rodriguez* 16K

```
B21E 2 PROGRAM SONAMOR:
B21E 3
B21E 4 (*****
B21E 5 (*
B21E 6 (*PROGRAMA TRADUCTOR*)
B21E 7 (*DE NUMEROS ROMANOS*)
B21E 8 (* A CARACTERES *)
B21E 9 (* ARABIGOS *)
B21E 10 (*
B21E 11 (*****
B21E 12 (*
B21E 13 (* CARLOS LUIS *)
B21E 14 (* RODRIGUEZ *)
B21E 15 (* RODRIGUEZ *)
B21E 16 (*
B21E 17 (*****
B21E 18
B21E 19 LABEL
B221 20 1;
B224 21
B224 22 VAR
B22D 23 LETRA: CHAR;
B22D 24 ERROR: BOOLEAN;
B22D 25 TOTAL, CIFRA: INTEGER;
B22D 26
B22D 27 PROCEDURE LEER;
B230 28
B230 29 BEGIN
B248 30 READ (LETRA);
B24E 31 IF NOT (LETRA IN ['M', 'D', 'C', 'L', 'X', 'V', 'I', ' ']) THEN
B2A8 32 BEGIN
B2A8 33 ERROR := TRUE;
B2AD 34 WRITELN ('SIMBOLO ILEGAL:', LETRA)
B2CD 35 END
B2D0 36 END;
B2D6 37
B2D6 38 PROCEDURE REPETICION (LALETRA: CHAR; MAX: INTEGER; VAR RESULTADO: INTEGER);
B2D9 39
B2D9 40 BEGIN
B2F1 41 RESULTADO := 0;
```

# ZX

REVISTA PARA LOS USUARIOS  
DE ORDENADORES SINCLAIR

**CASSETTES:**  
solución  
a los problemas  
de grabación

Animación  
gráfica

Cinemática  
con Spectrum

Tests  
psicológicos

**YA ESTA A LA VENTA**



**¡GANE ESTE  
MONITOR!**

# Programas

38

```
B300 42 WHILE LETRA=LALETRA DO
B315 43 BEGIN
B315 44 RESULTADO:=RESULTADO+1;
B32C 45 LEER
B32C 46 END;
B338 47 IF RESULTADO>MAX THEN
B355 48 BEGIN
B355 49 ERROR:=TRUE;
B35A 50 WRITELN('ERROR: MAS DE 3 ',LALETRA,' CONSECUTIVAS')
B393 51 END
B396 52 END;
B3A0 53
B3A0 54 PROCEDURE ANALIZAR(MENOR,MEDIO,MAYOR:CHAR;VAR RESULTADO:INTEGER);
B3A3 55
B3A3 56 BEGIN
B3BB 57 RESULTADO:=0;
B3CA 58 IF LETRA=MENOR THEN
B3DC 59 BEGIN
B3DC 60 LEER;
B3E5 61 IF LETRA=MEDIO THEN
B3F7 62 BEGIN
B3F7 63 RESULTADO:=4;
B406 64 LEER
B406 65 END
B40F 66 ELSE IF LETRA=MAYOR THEN
B424 67 BEGIN
B424 68 RESULTADO:=9;
B433 69 LEER
B433 70 END
B43C 71 ELSE
B43F 72 BEGIN
B43F 73 REPETICION(MENOR,2,RESULTADO);
B458 74 RESULTADO:=RESULTADO+1
B46E 75 END
B46F 76 END
B46F 77 ELSE IF LETRA=MEDIO THEN
B484 78 BEGIN
B484 79 LEER;
B48D 80 REPETICION(MENOR,3,RESULTADO);
B4A6 81 RESULTADO:=RESULTADO+5
B4BC 82 END
B4C5 83 END;
B4CF 84
B4CF 85 (*****
B4CF 86 (* PROGRAMA PRINCIPAL *)
B4CF 87 (*****
B4CF 88
B4CF 89 BEGIN
B4D8 90 I:ERROR:=FALSE;
B4DC 91 WRITELN;
B4DF 92 WRITELN('INTRODUCE NUMERO ROMANO');
B504 93 WRITELN(' (MENOR QUE 4000) ');
B522 94 WRITELN;WRITELN('EL ULTIMO CARACTER');
B545 95 WRITELN('DEBE SER UN BLANCO');
B565 96 WRITELN;
```

```

B568  97  READLN;
B56B  98  LEER;
B570  99  REPETICION('M',3,CIFRA);
B581  100 TOTAL:=CIFRA*1000;
B58D  101 ANALIZAR('C','D','M',CIFRA);
B5A2  102 TDOTAL:=TOTAL+CIFRA*100;
B5B9  103 ANALIZAR('X','L','C',CIFRA);
B5CE  104 TOTAL:=TOTAL+CIFRA*10;
B5E5  105 ANALIZAR('I','V','X',CIFRA);
B5FA  106 TOTAL:=TOTAL+CIFRA;
B60B  107 IF NOT ERROR THEN
B614  108   IF (LETRA=' ') OR (LETRA=CHR(13)) THEN
B637  109   BEGIN
B637  110   WRITELN('EL NUMERO ROMANO');
B655  111   WRITELN('ES IGUAL A :',TOTAL)
B675  112   END
B67B  113   ELSE WRITELN('SECUENCIA DE LETRAS INCORRECTA');
B6A7  114 GOTO 1
B6AA  115 END.
End Address: B6AC
Run?C-P

```



TU TIENDA

P.º GRACIA, 11, ESC. C, 2.º, 4.ª  
08007 BARCELONA  
☎ 318 24 53

DURANTE LOS MESES  
DE MARZO Y ABRIL, LOS  
AUTORES DE S.I.T.I. Y CONTEXT  
ATENDERAN PERSONALMENTE TODAS LAS  
CONSULTAS RELACIONADAS CON SUS  
PROGRAMAS DE FORMA TOTALMENTE  
GRATUITA.

**PIN SOFT, S.A.** / Paseo Gracia, 11, Esc. C 2.º 4.ª • BARCELONA

## Impresión vertical

Todas las impresoras tienen una lógica limitación: ¡el ancho de papel! Este pequeño programa elimina este "problema" al escribir en vertical, de forma simultánea a la impresión horizontal en la pantalla.

Pueden escribirse simultánea-

mente cuatro líneas, con tres opciones distintas de impresión: normal, cuadrados negros o cualquier carácter a elegir (si utiliza caracteres definidos, el mensaje sólo aparecerá en pantalla).

Para "acortar distancias" entre los caracteres, puede modificar el

valor del bucle FOR-NEXT de la línea 410. Asimismo, el número de caracteres por línea se puede alterar modificando el valor de PS en la línea 100. Y para los amantes del código máquina, incluimos el listado en assembler.

¡Que usted lo lea bien!

```

5 REM *****
10 REM *** Impresion ***
15 REM *** Vertical ***
20 REM *****
25 CLEAR 31699
30 LET mcode=31700
35 LET cset=mcode+100
40 FOR i=mcode TO mcode+61
45 READ n: POKE i,n
50 NEXT i
55 DATA 42,178,92,17,91,0,25,2
29,1,10,0,9,229,221,225,225,6,96
60 DATA 237,91,54,92,20,197,22
9,6,8,26,119,35,19,16,250,225,22
9,6
65 DATA 8,197,229,6,8,175,203,
6,31,35,16,250,221,119,0,221,35,
225
70 DATA 193,16,236,225,193,16,
218,201
75 RANDOMIZE USR mcode
80 REMCodigo maquina de caract
eres a partir de la direccion m
code + 100.
90 REM *****
95 BORDER 1: PAPER 5: INK 0: C
LS
100 DIM o$(4): DIM p$(4,100): D
IM a(4)
105 REM Para imprimir mas de 10
0 caracteres por linea, aumentar
la dimension de p$.
110 PRINT "Este programa permit
e la impre- sion vertical en la
impresora ZX Printer o GP-50S,
con 4 li- neas de texto y una
longitud maxima de 100 caract
eres por linea."

```

```

120 PRINT "Se imprimen todos lo
s caracteres con CODE comprendido
entre los valores 32 y 127. Lo
s graficos definidos no salen."
140 PRINT "Existen tres opcione
s de impre- sion:"
145 PRINT "1) Normal."
150 PRINT "2) Cuadrados negros."
"
155 PRINT "3) A elegir."
160 PRINT "Cual desea: 1,2 o 3
": INPUT option
165 IF option<1 OR option>3 THE
N GO TO 160
166 PRINT option
170 IF option<>3 THEN GO TO 19
0
175 PRINT "Seleccione el caract
er elegido?": INPUT c$
180 IF LEN c$<>1 THEN GO TO 17
5
185 PRINT c$
190 PRINT "OK. Introduzca el me
nsaje."
200 LET max=0
210 FOR i=1 TO 4
220 INPUT ("Linea ";i;"?");a$
230 IF LEN a$>max THEN LET max
=LEN a$
240 LET p$(5-i)=a$
250 PRINT a$
260 NEXT i
270 LET b2=INT (cset/256)-1
280 LET b1=cset-256*(b2+1)
290 POKE 23606,b1: POKE 23607,b
2
300 REM Nueva direccion del jue
go de caracteres en CHARS.

```

```

310 FOR j=1 TO 4
320 IF option=2 THEN LET o$(j)
=" "
330 IF option=3 THEN LET o$(j)
=c$
340 NEXT j
350 REM comienzo impresion.
360 FOR n=1 TO max
370 FOR j=1 TO 4
380 LET a(j)=cset+8*(CODE p$(j,
1)-32)
390 IF option=1 THEN LET o$(j)
=p$(j,1)
400 NEXT j
410 FOR j=0 TO 7
420 FOR k=1 TO 4
430 LET w=PEEK (a(k)+j)
440 IF w=0 THEN GO TO 540
450 LET t=8*(k-1)
460 LET c=128
470 IF w<c THEN GO TO 510
480 PRINT TAB t;o$(k);
490 LPRINT TAB t;o$(k);
500 LET w=w-c
510 LET t=t+1
520 LET c=c/2
530 IF w>0 THEN GO TO 470
540 NEXT k
550 PRINT : LPRINT
560 NEXT j
570 FOR j=1 TO 4
580 LET p$(j)=p$(j,2 TO )
590 NEXT j

```

```

600 POKE 23692,55: REM Para evi
tar el SCROLL?.
610 NEXT n
620 POKE 23606,0: POKE 23607,60
630 REM Vuelta al juego de cara
cteres normales.
640 STOP

```

```

LD HL,(23730)
LD DE,91
ADD HL,DE
PUSH HL
LD BC,10
ADD HL,BC
PUSH HL
POP IX
POP HL
LD B,96
LD DE,(23606)
INC D
ROTAT PUSH BC
PUSH HL
LD B,B
RT1 LD A,(DE)
LD (HL),A
INC HL
INC DE
DJNZ RT1
POP HL
PUSH HL
LD B,B
RT2 PUSH BC
PUSH HL
LD B,B
XOR A
RT3 RLC (HL)
RRA
INC HL
DJNZ RT3
LD (IX+0),A
INC IX
POP HL
POP BC
DJNZ RT2
POP HL
POP BC
DJNZ ROTAT
RET
END

```

```

j
j
j
jjjj j
uuuu
u
u
uuuu
a
a a a
a a a
a a a
aaaa
nnnnn
n
n
nnnn
i i
iiii i
i
t
tttt
t t
t
ooo
o o
o o
o o
ooo
e
j
e
m
p
l
o
d
e
s
a
l
i
d
a
m
e
n
s
a
j
e
p
o
r
i
m
p
r
e
s
o
r
a

```



**microgesa**

**ESPECIALISTAS EN SINCLAIR  
SAQUE RENTABILIDAD AL SPECTRUM**

**PROGRAMAS EN MICRODRIVE ZX  
GESTION:**

	Ptas.
Contabilidad (P. Nacional C.)	12.000
Base de Datos	6.000
Proceso de Textos (Español)	6.500
Calc (hoja electrónica)	4.000
Control Stock - Facturación	8.500

**PROGRAMAS TECNICOS**

Agente de Bolsa	6.500
Mediciones y Presupuestos	24.000

**PROGRAMAS UTILIDADES**

	Ptas.
SUPERDESARROLLOS 1X2 .....	3.900
Declaración renta 84 .....	2.400

**PROGRAMAS EN CASSETTE  
EDUCATIVOS:**

	Ptas.
Geografía de España I	1.900
Geografía de España II	1.900
Geografía de Europa I	1.900
Curso de Contabilidad I	2.200
Curso de Contabilidad II	2.200
Curso de Contabilidad III	2.200
Curso de Contabilidad IV	2.200
Geometría y Trigonometría	1.900

(Imprime boletos con Impresora ADMATE)  
(Impresión carta de pago)

**ORDENADORES, IMPRESORAS, MONITORES, ACCESORIOS Y AMPLIACIONES DE MEMORIA**

ENVIOS CONTRA-REEMBOLSO, GIRO O TALON CONFORMADO C/Silva, 5 - 4.º - Tel. 242 24 71 - 28013 MADRID

# Preguntas y respuestas

**P** He leído detenidamente el artículo "64 caracteres por línea" y lo encuentro muy interesante y práctico, pero en él se dice que se adjunta una tabla de número en decimal que constituyen el nuevo juego de caracteres. Esta tabla deberá introducirse..." (pág. 23 del número 5). La duda que tengo es que no encuentro esta tabla y quisiera saber si es que no la habéis incluido en el artículo, dejándola al libre albedrío de los lectores.

**R** Es bueno que los lectores tengáis iniciativa y que la podáis canalizar a través de esta revista, pero fue ese el motivo por el que no aparecía el juego de caracteres: icreemos que fue sabotaje de Gusanez por el castigo del número anterior!

```

0 0 0 0 0 0 64 0
0 0 0 32 64 128 0 0
0 64 160 224 160 160 64 0
0 192 64 64 64 64 224 0
0 224 32 32 64 128 224 0
0 192 32 224 224 32 192 0
0 32 96 160 224 32 32 0
0 224 128 224 32 32 224 0
0 224 128 224 160 160 224 0
0 224 32 32 64 128 128 0
0 224 160 64 224 160 224 0
0 224 160 224 32 32 224 0
0 0 0 64 0 0 64 0
0 0 64 0 64 64 128 0
0 0 32 64 128 64 32 0
0 0 0 224 0 224 0 0
0 0 128 64 32 64 128 0
0 224 32 224 128 0 128 0
0 32 0 32 224 128 224 0
0 224 160 160 224 160 160 0
0 192 160 192 192 160 192 0
0 224 128 128 128 128 224 0
0 192 160 160 160 160 192 0
0 224 128 192 192 128 224 0
0 224 128 192 192 128 128 0
0 224 128 128 160 160 224 0
0 160 160 224 224 160 160 0
0 224 64 64 64 64 224 0
0 32 32 32 160 160 224 0
0 160 160 192 192 160 160 0
0 128 128 128 128 128 224 0
0 160 224 160 160 160 160 0
0 192 160 160 160 160 160 0
0 224 160 160 160 160 224 0
0 224 160 224 128 128 128 0
0 224 160 160 160 224 64 0
0 224 160 160 224 192 160 0
0 224 128 128 224 32 224 0
0 224 64 64 64 64 64 0
0 160 160 160 160 160 224 0

```

```

0 160 160 160 160 160 64 0
0 160 160 160 224 224 160 0
0 160 160 160 64 160 160 0
0 160 160 160 64 64 64 0
0 224 32 64 128 128 224 0
0 192 128 128 128 128 192 0
0 0 128 64 32 0 0 0
0 96 32 32 32 32 96 0
0 64 224 64 64 64 64 0
0 0 0 0 0 0 0 240
0 96 96 64 224 64 96 0
0 0 224 32 224 160 224 0
0 128 128 128 224 160 224 0
0 0 224 128 128 128 224 0
0 32 32 32 224 160 224 0
0 0 224 160 224 128 224 0
0 96 64 64 224 64 64 0
0 0 224 160 224 32 96 0
0 128 128 128 224 160 160 0
0 64 0 64 64 64 96 0
0 64 0 64 64 64 192 0
0 0 160 192 192 160 160 0
0 64 64 64 64 64 96 0
0 0 160 224 160 160 160 0
0 0 192 160 160 160 160 0
0 0 224 160 160 160 224 0
0 0 224 160 224 128 128 0
0 0 224 160 224 32 32 0
0 0 160 192 128 128 128 0
0 0 224 128 224 32 224 0
0 64 64 224 64 64 96 0
0 0 160 160 160 160 224 0
0 0 160 160 160 160 64 0
0 0 160 160 160 224 160 0
0 0 160 160 64 160 160 0
0 0 160 160 64 64 128 0
0 0 224 32 64 128 224 0
0 96 64 192 192 64 96 0
0 64 0 64 64 64 64 0
0 192 64 96 96 64 192 0

```

```

0 0 0 0 0 0 0 0
0 64 64 64 64 0 64 0
0 0 160 0 0 0 0 0
224 0 160 160 224 160 160 0
64 224 128 224 32 224 64 0
0 160 32 64 128 160 0 0
224 0 192 160 160 160 160 0
0 0 0 0 0 0 0 0
0 64 128 128 128 128 64 0
0 64 32 32 32 32 64 0
0 0 0 160 64 160 0 0
0 0 0 64 224 64 0 0
0 0 0 0 64 64 128 0
0 0 0 0 224 0 0 0

```

## GUSANEZ por Jose C. Tomás



**P** Me podrían decir si están bien los tiempos sobre la carga de programas correspondientes a su artículo OPUS, O LA RAPIDEZ DE LOS FLOPPYS. La pregunta viene ya que si el tiempo de carga del floppy es igual que el del microdrive me gustaría que me aclararan las ventajas de este floppy, sobre todo si no es compatible con el interface 1.

**Emiliano Bartolomé  
Madrid**

**R** Aprovechamos su pregunta para hacer algunas puntualizaciones al respecto:

1. Lo más destacable no es la unidad de discos OPUS, sino el interface Technology Research, que contiene el sistema operativo DOS y el controlador de la unidad de disco.

2. El test de velocidad efectuado se refería a operaciones de carga, no de grabación, resultando tiempos sensiblemente inferiores a los de microdrive.

3. Ya existe la versión doble densidad para el interface, con lo cual los discos hasta ahora de 100K se convierten en 160K, y los de 400 pasan a 640K.

4. El interface de Technology Research es totalmente compatible con el Interface 1. La incompatibilidad a que hacíamos alusión era debida al interface Kempston para impresoras.

**P** Tengo una impresora Seikosha GP-100 conectada al micro por medio del Interface I, pero no consigo copia de la pantalla. Esto es lógico, dado que la función COPY de Sinclair no es estándar y no puede ser usada más que con impresoras fabricadas para conectarse directamente, como ZX-PRINT. Sin embargo con una salida RS-232, teóricamente sería posible hacer una copia de la pantalla por medio del siguiente programa:

10 FORMAT "t"; 4800: REM AJUSTAR BAUDIOS.

20 OPEN #3; "t": REM ABRIR CANAL DE IMPRESORA.

30 MOVE #2 TO #3: REM VOLCAR EL CONTENIDO DE LA PANTALLA EN LA IMPRESORA.

40 CLOSE #3: REM CERRAR CANAL DE IMPRESORA.

**Miguel Angel  
Oviedo**

**R** El programa que propone el lector no funciona ya que la pantalla es una unidad que sólo puede utilizarse como destino de la información, nunca como fuente. El programa no funciona del mismo modo que no funcionaría uno que intentara copiar la impresora en pantalla.

En cuanto a su intención de usar el Interface I para imprimir caracteres gráficos con la SEIKOSHA GP-100 requiere un notable esfuerzo de programación. Cuando detectemos un carácter gráfico o definido por el usuario debemos informar a la impresora que vamos a pasar a modo gráfico. Para este paso y el siguiente, será necesario pasar la salida RS-232 del canal "t" al canal "b". A continuación se buscará en la zona de memoria reservada para los gráficos la información referente al que nos ocupa.

Pero tenga cuidado, porque los datos se obtendrán cortando el carácter en "rebanadas" verticales y no horizontales como las encuentra en RAM.

De nuevo volverá a modo texto "t" para continuar con la impresión. Pero como habrá deducido, el uso de este método implica que tendrá que escribir, no sólo la rutina comentada sino TODO el programa que gestiona la impresión. Encontrará ideas interesantes en los programas para el interface paralelo Centronics (TODOSPECTRUM número 2)

# VENTAMATIC

## TECLADO PROFESIONAL PARA ZX-SPECTRUM MODELO LO-PROFILE

- Diseño ultra-moderno y compacto.
- Con barra espaciadora y teclado numérico separado.
- Ergonómicamente diseñado e inclinado hacia adelante para facilitar su uso.
- 53 teclas SERIGRAFIADAS IMBORRABLES (sin etiquetas adhesivas), de altura perfectamente ajustada y comprobadas para 20 millones de operaciones.
- Sencilla instalación del ZX-SPECTRUM en su interior.
- Compatible con ZX-INTERFACE 1, ZX-MICRODRIVE y demás accesorios.



DISPONIBLE YA. SOLO 14.500,— PTAS.

### OFERTAS ESPECIALES

- 1) SPECTRUM PLUS + AJEDREZ + VU-3D + BANDERA A CUADROS + SCRABBLE + MAKE-A-CHIP + TASWORD TWO + SPECTRUMANIA. Precio normal: 67.000,— ptas. **Oferta: 49.000,— ptas.**
- 2) GESTION 48K: BASE DE DATOS S.I.T.I. + PROCESADOR DE TEXTOS CONTEXT V.6 (ambos 64 caracteres/linea). Precio normal: 8.000,— ptas. **Oferta: 6.400,— ptas.**
- 3) TECLADO LO-PROFILE + S.I.T.I. + CONTEXT V.6. Precio normal: 22.500,— ptas. **Oferta: 19.900,— ptas.**
- 4) JUEGOS 48K 3D: FULL THROTTLE + ANDROID TWO + DEATHCHASE + TORNADO LOW LEVEL + CODENAME MAT + 3D INTERCEPTOR. Precio normal: 10.400,— ptas. **Oferta: 7.800,— ptas.**
- 5) UTILIDADES 48K: HISOFT DEVPAC + HISOFT PASCAL + BETABASIC. Precio normal: 12.500,— ptas. **Oferta: 10.000,— ptas.**
- 6) ZX-INTERFACE 1 + ZX-MICRODRIVE + S.I.T.I. + CONTEXT V.6 + 2 CARTUCHOS VIRGENES ZX-MICRODRIVE. Precio normal: 46.150,— ptas. **Oferta: 39.900,— ptas.**

TAMBIEN TENEMOS OFERTAS CON IMPRESORAS DE 80 COLUMNAS, CENTRONICS, RS232, FRICCION, TRACCION, ETC.

VEN A CONOCERNOS. Somos los SUPER-ESPECIALISTAS DEL SPECTRUM y lo tenemos TODO para TU SPECTRUM. SOLICITA CATALOGO COMPLETO

VENTAMATIC - C/ Córcega, 89, entlo.  
08029-BARCELONA - Tel.: (93) 230 97 90. Metro Estenya  
(línea 5). Bus: 41, 27, 15, 54, 66.

BOLETIN DE PEDIDO  
Enviar a: VENTAMATIC - Avda. de Rhode, 253  
ROSES (Girona) - Tel.: (972) 25 79 20.

Fecha: \_\_\_\_\_  
Nombre: \_\_\_\_\_  
Apellidos: \_\_\_\_\_  
Dirección: \_\_\_\_\_  
Población: \_\_\_\_\_  
Provincia: \_\_\_\_\_  
D.P.: \_\_\_\_\_  
Deseo recibir los siguientes artículos:

\_\_\_\_\_

GASTOS DE ENVIO: \_\_\_\_\_  
TOTAL: \_\_\_\_\_

Señalar con una cruz la forma de pago:  
( ) Talón adjunto (sin gastos de envío).  
( ) Contra-reembolso (500,— ptas. gastos envío).  
( ) Giro postal n.º \_\_\_\_\_ (sin gastos de envío)  
( ) Tarjeta VISA/MASTERCARD n.º \_\_\_\_\_  
Cada: \_\_\_\_\_  
Firma: \_\_\_\_\_ (500,— ptas. gastos envío).



# Preguntas y respuestas

**P** Estoy haciendo un juego en código máquina para el Spectrum de 48K, cuyo organigrama les detallo. El problema surge en el momento de hacer el salto (JUMP) de nuevo a la rutina de lectura del teclado. Creo que la respuesta esté en la instrucción JP NN — C3h — 195, pero el problema surge cuando direcciono el valor NN ¿cómo lo hago? ¿Ha de ser Nn la dirección de comienzo de rutina de lectura de teclado? Supongamos que la rutina de teclado empieza en la dirección 62000, ¿quedaría C3DOF3? Además, como podría hacer para que atacaran a la vez dos o más naves enemigas? El problema surge cuando una vez des-

truida una de las naves, ataque otra sin que la primera pierda su posición

**Jesús Gómez Madrid**

**R** La instrucción de ensamblador Jp b1 b2 realiza un salto (Jump) a la dirección de memoria  $256 * b2 + b1$ , donde b1 y b2 son números de 8 bits.

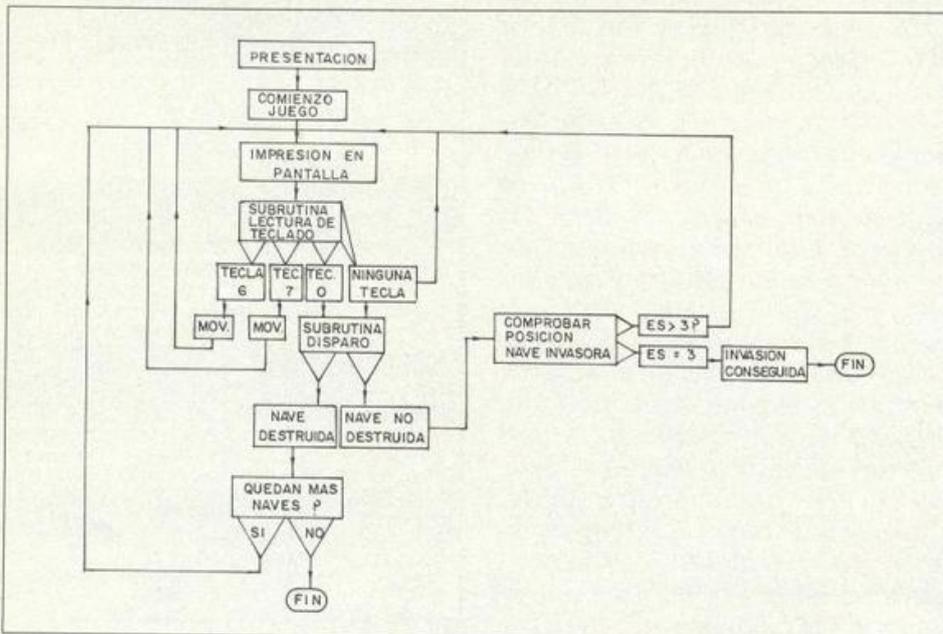
Consecuentemente si queremos ejecutar una rutina en la dirección 62000 (F230H) la instrucción correspondiente en código máquina será: C3 30 F2.

Existe también la posibilidad de usar la instrucción JR N (Jump

Relative, Salto Relativo), donde N tomará un valor que depende, no de la posición absoluta de la rutina, como en el caso anterior, sino de la posición relativa entre el origen y el destino del salto. Esto hace que la instrucción no necesite modificarse si el programa se localiza en una dirección diferente de la prevista.

Pero ¡cuidado! Si en la dirección 62000 tenemos una subrutina, la instrucción a emplear ser la CALL F2 30, (CD 30 F2).

De la carta deducimos que el lector hace sus programas en código máquina. Por ello le recomendamos que adquiera un buen programa ensamblador. Esto le ahorrará tiempo y disgustos, y le conducirá al estimulante mundo de la programación en lenguaje ensamblador.



**P** Les mando un dibujo realizado con el Spectrum, en respuesta al reto que planteaban al final del artículo del mes de enero sobre cómo realizan sus portadas. La idea del dibujo parte de otras dos, muy de actualidad, por un lado la piratería de programas que nos afecta desgraciadamente a nosotros los usuarios de Spectrum, ya que te pueden vender una copia en lugar del original, y por otro el fundamento del dibujo es el símbolo, tan de moda, de los cazafantasmas.

Desde mi status de usuario, no

Para abril nos esperan grandes novedades Sinclair y la comercialización definitiva del QL por Investrónica. Este será nuestro tema principal, pero no el único:

- AMPLIE SU MEMORIA. Nos referimos al Spectrum.
- PREYME. Un programa muy serio para realizar sus presupuestos y mediciones de obra.
- CONOZCA EXTREMADURA. Un original programa se la enseña.
- JUEGUE AL MONOPOLY CON SU SPECTRUM. No necesitará el tablero.





puedo hacer nada ya que no sé si el programa que me venden es pirata o no, ante esto envió esta cassette en la que está grabado el dibujo, que creo que sería bien visto en la portada de vuestra revista o en cualquier artículo referente a la piratería de programas.

El dibujo está localizado a partir de la dirección 32768 hasta la 32768+6912 ya que los realicé con el programa Supergráficos y he intentado grabarlo como una pantalla normal, es decir, con SAVE "x" SCREEN\$, después de reubicarlo en las direcciones 16384 en adelante y me encuentro con el problema del mensaje "Start tape, then press any key" que me borra parte del dibujo.

**Francisco Carrión  
Sevilla**

**R** Efectivamente estamos contigo. La piratería cada vez se vuelve más importante y ello también repercute en el usuario, a quien fácilmente le pueden dar "gato por liebre", ello sin contar el incalculable daño que se hace en todos aquellos que quieren tener un futuro como programador.

**P** Soy un estudiante de telecomunicaciones y como estoy empezando no tengo aún mucha idea. He intentado realizar el montaje del Lápiz Optico del número seis y algo va mal. ¿Están bien los componentes?

**R** Efectivamente, aunque principiante en telecomunicaciones, sus «tiros» iban por buen camino. En la lista de componentes se hacía referencia a un condensador de 10  $\mu F$  en vez de nF que es el correcto. ¿Funcional ahora?

## ACLARACION

El artículo Control de evaluaciones y su correspondiente programa del número de enero, nos fue remitido por Juan Carlos Fernández Romero de Las Palmas de Gran Canarias.

VENDO ZX-81 en perfecto estado con ampliación a 16 K así como todos los cables necesarios para la conexión a TV, a cassette y fuente de alimentación; regalo dos cintas de juegos y listados de programas. Interesados llamar al 246 24 80, preguntar por Gonzalo, preferentemente por las tardes



VENDO impresora Sikosha GP-50-S, conectable directamente al Spectrum, muy nueva, con embalaje original y garantía de seis meses a partir del día en que la venda, por sólo 22.000 (su precio: 28.900). Utiliza papel normal. Llamar al (96) 340 72 28, preguntando por José (mediodía o noche).



Interface CENTRONICS VENTAMATIC, para SPECTRUM con cable y cinta (8.000 pts). Tel. 254 39 55 (BARCELONA). Julián NAJERA

CAMBIO pistola de aire comprimido marca El gamo, con poco uso y papeles. Interface programable para joystick. Interesados llamar al (94) 463 47 91, de 19,00. Preguntar por Inaki.



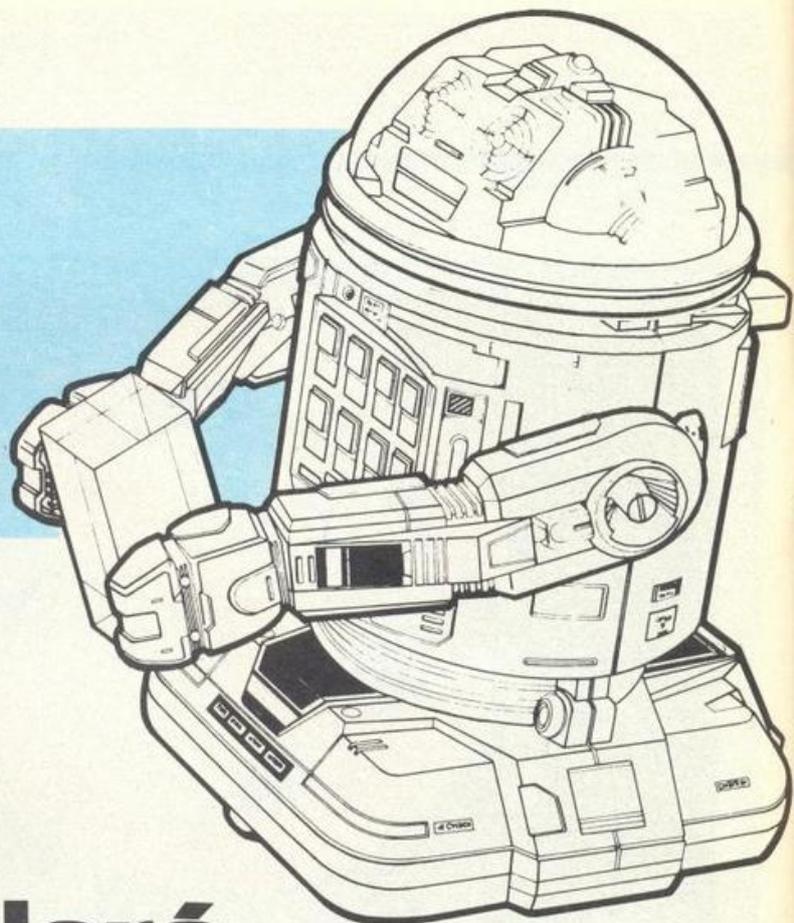
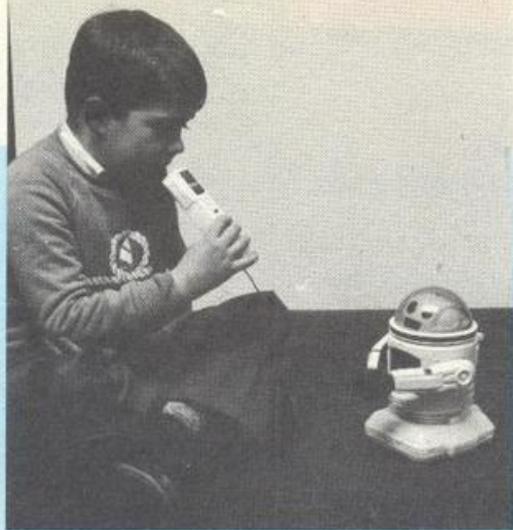
COMPRO Impresora conectable a serie RS-232, de 80 o más columnas, de impresión matricial, en buen estado, y que no cueste más de 35.000; imprescindible manual de operación. Llamar al (96) 340 72 28, preguntando por José (mediodía o noche).



Vendo interface y microdrive. Todo 29.000 pts. Regalo 4 cartuchos para el microdrive y cassette con el trans-express, ghostbusters y gift from the gods. Alberto Tf F38-97-01 de Madrid.

VENDO IMPRESORA SEIKOSHA GP250-X. Permite entrada. IFCENTRONICS y RS 232. 64 caracteres programables. 4 tipos letra. Ideal para SPECTRUM 48 K. (42.000 pts.).





# Dígasele en voz alta: Verbot le entenderá

**H**ace algunos años cuando fue estrenada la película «la Guerra de las Galaxias» todos quedamos gratamente sorprendidos por la presentación del robot R2D2, que hacía las delicias de grandes y pequeños. La fama de éste simpático robot fue extendiéndose y, desde entonces, es rara la película de ciencia ficción en la que los protagonistas no están acompañados por un robot parecido. A partir de éste momento, como era de esperar, los fabricantes de juguetes deciden aprovechar el éxito de estos divertidos robots-mascotas para comenzar a fabricar en serie un juguete que los imitara. Así, en las últimas navidades los japoneses nos han presentado a **Verbot**, un pequeño y curioso robot controlado a través de un transmisor de voz.

**Verbot** es el segundo en tamaño de la saga de los robots fabricados por la firma **Tomy Corp** y distribuido por **Investrónica**. Su aspecto exterior nos recuerda al de R2D2, aunque **Verbot** es bastante más pequeño (mide solamente 23 centímetros de alto por 22 de ancho) y tiene dos brazos que le permiten

levantar objetos. Su cabeza es una bola de cristal oscuro con dos ojos y una boca que se iluminan con una luz roja; en el tronco (de color blanco) están los dos pequeños brazos, un panel con las ocho teclas de memoria y el interruptor de OFF/ON con luz indicadora y una luz de memoria que indica cuándo son registradas las órdenes; por último, en la base del robot hay dos ruedas que facilitan su desplazamiento, un espacio en el que se introducirán las pilas (las instrucciones recomiendan que sean alcalinas) y la antena receptora.

Antes de comenzar a programarlo es conveniente leerse las instrucciones y, después, introducir cada pila en su lugar correspon-

diente. Para funcionar utiliza 1 pila de 9 voltios para el transmisor de voz, 2 medianas de 1,5 voltios y 4 pequeñas de 1,5 voltios que se introducen en la base del robot. Una vez realizada esta operación podemos comenzar a programarlo. Para ello tenemos que transmitirle las órdenes por el micrófono, a la vez que pulsamos la tecla de memoria que corresponda (cada tecla tiene un dibujo que representa la orden), mientras que la luz indicadora de memoria destellee. De éste modo, las órdenes (ondas sonoras) se convierten en señales digitales que el robot memoriza y compara. Para que **Verbot** entienda sin dificultad, las órdenes deben ser cortas. Es conveniente también que las palabras sean diferentes en sonido y tono para que las comprenda. Una vez programado, podrá realizar ocho tareas: pararse, adelante, atrás, girar a la izquierda, girar a la derecha, sonreír, levantar objetos y soltarlos. En las instrucciones vienen una serie de palabras que pueden utilizarse para dar las órdenes.

El único defecto es su precio: 13.750 ptas. ★

## ¡DEMASIADO HUMANO!

Este es el caso del robot instalado en la entrada de la Cincinnati Milacron Inc., que juega al black-jack. Reparte cartas para él y otros dos jugadores, y está programado para decidir si quiere cartas.

Cuando alcanza los 16 puntos, señala al techo y dice, en una computarizada voz: "¿Qué es eso?" momento que aprovecha para coger un cinco escondido en la manga y así alcanzar el 21.

# No se caliente la "CABEZA"

## SEIKOSHA

DIRAC S.L.



Nuestra calidad es "SEIKO";  
nuestros precios, únicos.  
Si desea más información,  
consulte con nuestro distribuidor  
más cercano, o llame o escriba a:

**DIRAC S.L.**

Dirección comercial:  
Av. Blasco Ibáñez, 114-116.  
46022-Valencia.  
Tel. (96) 372 88 89.  
Télex 62220

Delegación en Cataluña:  
C/ Muntaner, 60, 4, 1.  
08011-Barcelona.  
Tel. (93) 323 32 19.

### ESTOS SON NUESTROS MODELOS:

Modelo	Velocidad	Columnas	Tipos de letra	Interface	P.V.P.
GP-50	40 cps	46	2	A-Paralelo AS-Serial S-Spectrum	A-25.900 AS-29.900 S-28.900
GP-500	50 cps	80	2	A-Paralelo AS-Serial	A-47.900 AS-49.900
GP-550	86 cps	80-136	18	A-Paralelo	A-59.900
GP-700	50 cps	80-106	3	A-Paralelo	A-89.900
BP-5200	200 cps	136-272	18	Paralelo y serial	199.000
BP-5420	420 cps	136-272	18	Paralelo y serial I-IBM PC	299.000 I-299.000

Disponemos de interfaces opcionales para todos los modelos: IBM PC, COMMODORE 64, ZX SPECTRUM, ATARI, DRAGON 64, SHRAP MZ 700, SPECTRAVIDEO, NEW BRAIN, APPLE, ETC...

# QL



**PROGRAMAS DISPONIBLES  
EN MICRODRIVE  
PARA EL QL**

SUPERCONT  
Adaptada al plan general contable  
PASCAL  
FORTH  
ENSAMBLADOR  
DESENSAMBLADOR

PROGRAMAS DE GESTION POR ENCARGO

DEMOSTRACIONES SEMANALES

**SOBRE EL QL  
NOS LAS SABEMOS TODAS.  
¡¡PREGUNTANOS!!**

**Muy pronto... ¡con unidades de disco!**

**sinclair store**  
SOMOS PROFESIONALES

BRAVO MURILLO, 2 (aparc. gratuito en C/. Magallanes, 1). Tel.: 446 62 31  
DIEGO DE LEON, 25 (aparc. gratuito en C/. Núñez de Balboa, 114). Tel.: 261 88 01 MADRID