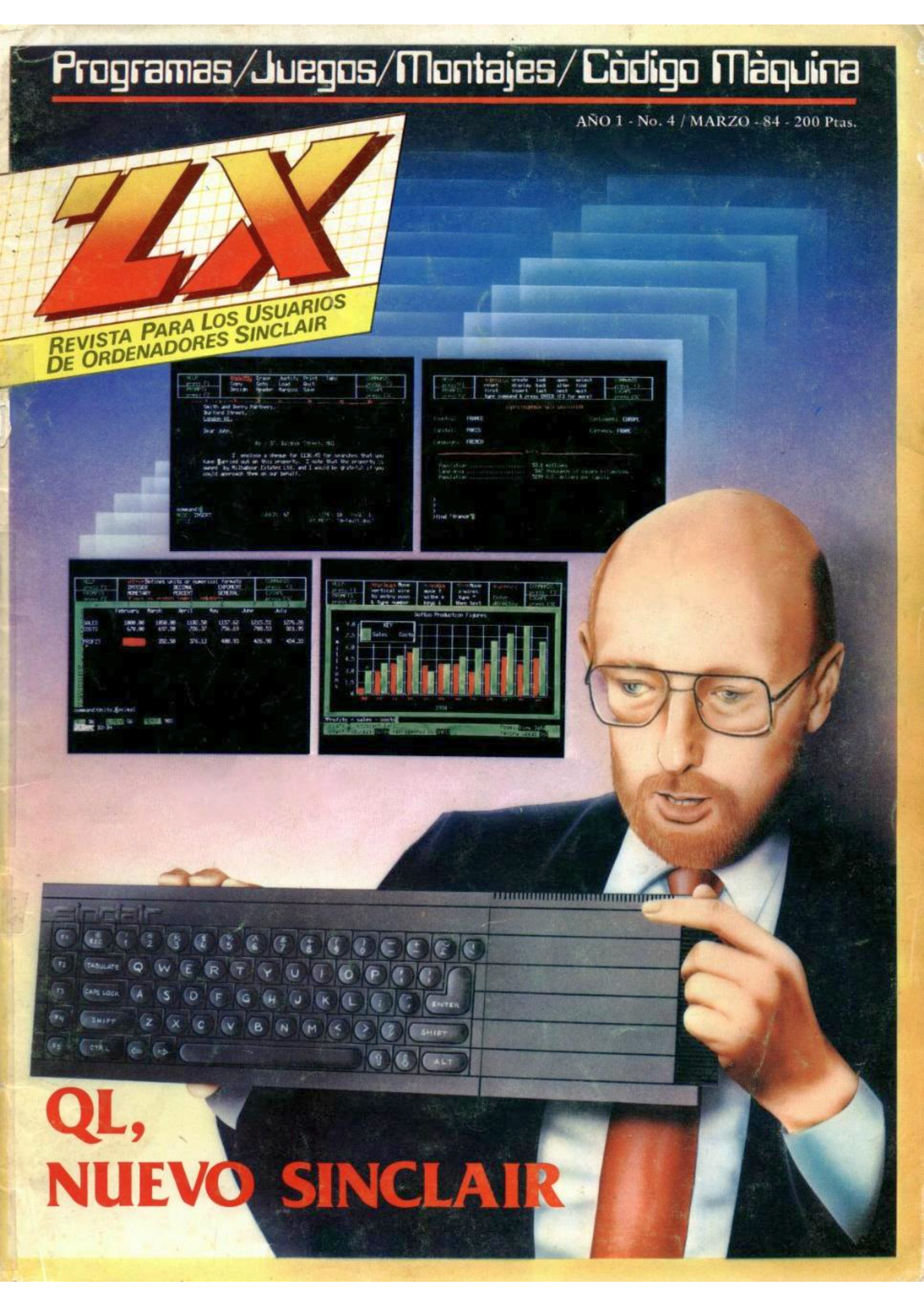
Programas/Juegos/Montajes/Código Máquina

AÑO 1 - No. 4 / MARZO -84 - 200 Ptas.

LX

REVISTA PARA LOS USUARIOS DE ORDENADORES SINCLAIR



QL, NUEVO SINCLAIR


ZX Spectrum
16 K: 39.900 Ptas.
48 K: 52.000 Ptas.
sinclair
ZX Spectrum
El ordenador de todos para todo.
DISTRIBUIDOR EXCLUSIVO:
INVESTRONICA
Central Comercial TOMAS BRETON 60 TELF 468 03 00 TELEX 23399 IYCO E MADRID
Delegación Cataluña MUNTANER 565 TELF 212 68 00 BARCELONA

AÑO I - No. 4

ZX

REVISTA PARA LOS USUARIOS DE ORDENADORES SINCLAIR

En nuestro número anterior, respondíamos a un lector que pedía un programa de bingo, sugiriendo que otros lectores nos hicieran llegar alguno. Pues bien, la reacción ha sido inmediata y sorprendente: hemos recibido más de treinta programas de bingo. Esta respuesta nos halaga y entusiasma. Tanto que hemos decidido hacer de esos programas un tema de concurso en sí mismo. En consecuencia, escogeremos los mejores de esos programas, los publicaremos y premiaremos Hasta el próximo número.

NOVEDADES

Nada más comenzar el año, Clive Sinclair ha vuelto a sorprender al mundo de la informática con un nuevo modelo de ordenador, llamado QL, y que promete ser protagonista de una auténtica revolución. No es un sustituto del Spectrum, sino algo totalmente diferente. Un ordenador destinado al mundo profesional y de negocios, lo que no excluye que sus capacidades gráficas puedan aprovecharse debidamente para programar juegos seguramente máps potentes.

Página 4

SOFTWARE

Continúa este número la serie de artículos que escribe Juan Martínez Velarde sobre Código Máquina en el Spectrum. La segunda entrega de la serie, ya superado el tema de la organización de la memoria, está dedicada a los mnemónicos.

Página 56

PROGRAMAS

El núcleo central de esta revista sigue siendo los listados de programas para correr en los ordenadores Sinclair. Algunos lectores se quejan de la escasa cantidad de programas que publicamos para el ZX81. Vamos a tratar de complacerlos en alguna medida, aunque también es cierto que cada vez son menos los programas originales para un ordenador que, inevitablemente, está dejando lugar a su sucesor. Y, como de costumbre, publicamos los programas seleccionados entre los muchos que nos han enviado los lectores.

Página 26

LIBROS

La bibliografía sobre el Spectrum sigue creciendo. Este mes comentamos dos libros editados por Paraninfo, uno de los cuales es el que obsequiamos junto con la suscripción a ZX. Los dos son de Antonio Bellido.

Página 24

IDEAS

Una nueva sección, destinada a comentar trucos y truquillos para sacar mejor partido de su ordenador. Y esperamos que los lectores nos hagan llegar sus hallazgos.

Página 14

LECTORES

Nada menos que cinco páginas han sido necesarias este mes para dar salida al enorme volumen de correo que nos ha llegado. Quedan todavía muchas por contestar, y siguen llegando.

Página 18

COMENTARIOS

Esta sección, inaugurada el número anterior, continúa pasando revista al *software* disponible en el mercado español. Comentamos esta vez tres programas de juegos.

Página 16

ZX es una publicación de Ediciones y Suscripciones, S. A. • Presidente: Fernando Bolín • Jerez, 3. Telfs.: (91) 250 15 93 - 458 76 02. Madrid-16 • Director Editorial: Norberto Gallego • Redacción: Alejandro Diges, Aníbal Pardo, Simeón Cruz, Gumersindo García, José Luis Durá jr • Diseño: R. Segura • Administración Gerente de Circulación y Ventas: Luis Carrero • Suscripciones: Antonio Zurdo (91) 457 26 17 • Producción: Miguel Onieva • Publicidad Madrid: Telf. (91) 457 45 66 • Publicidad Barcelona: Tallers, 62-64, Barcelona-1. Telf. (93) 302 36 48 • Distribuye: Sociedad General Española de Librería. Avda. Valdelaparra, s/n. Alcobendas, Madrid • Imprime: Heroes, S. A. Torrelara, 8. Madrid-16 • Depósito Legal: M. 37.432-1983.

Esta revista no mantiene relación de dependencia de ningún tipo con respecto de los fabricantes de ordenadores Sinclair ni de sus representantes.

QL el nuevo

Una vez al año, y ya van cuatro, Clive Sinclair sorprende al mundo de la informática introduciendo nuevos productos que marcan la pauta del mercado en cuanto a precio y tecnología. Con su nuevo microordenador QL, lo ha conseguido de nuevo.

Al parecer, Sir Clive (otros prefieren llamarlo "tío Clive") y su empresa, Sinclair Research, tienen un endemoniada habilidad para apreciar las tendencias de los compradores potenciales, y aplican ese buen juicio a la tarea de hacer más ordenadores y más dinero. Cuando la gente tenía poca o ninguna idea de lo que era un ordenador personal o de lo que se podía hacer con uno, Sinclair lanzó al mercado su ZX80, primero, y su ZX81, después, ambos a precios que parecían imbatibles. Cuando esos mismos usuarios comprobaron la versatilidad de los dos sucesivos modelos ZX, Sinclair sacó a relucir el Spectrum en versiones de 16 y 48 Kbytes. Ahora pone en circulación su nuevo QL, orientado hacia el profesional, el universitario y los hombres de negocios. Con un precio de 399 libras (no hay indicación alguna de cuál puede ser el precio en España, pero tenemos varios meses por delante para enterarnos), el QL tendrá en esos sectores del mercado la misma repercusión que la saga de ordenadores ZX ha tenido en la informática doméstica.

Porque, conviene decirlo de entrada, el QL no es un Spectrum mejorado, no viene a sustituirlo ni pretende ocupar su bien ganado lugar en el mercado. Es otra cosa, y el propio Sinclair quiere ponerlo de relieve al bautizar su nuevo modelo con una sigla que significa *Quantum Leap*, salto cuántico. No incurriremos en obsecuencia hacia el inventor británico si decimos que, efectivamente, su cuarto modelo representa un paso gigantesco, una innovación que obligará a otras firmas a reacomodarse.

Como siempre ocurre con los productos de Sinclair Research, el QL es muy poco convencional. En lugar de adaptarse a las tendencias dominantes, que le hubieran llevado a adoptar versiones más baratas de los sistemas operativos CP/M o MS-DOS, o incluso a buscar la compatibilidad con el IBM/PC a nivel de microprocesador, no solamente ha pasado de un microprocesador de 8 bits a uno de 32 bits sino que ha incorporado un sistema operativo propio (Q-DOS) con una impresionante capacidad multitarea, en una máquina cuya memoria RAM es dos o tres veces superior a la de sus más cercanos competidores.

El ordenador que hemos tenido ocasión de examinar era un modelo piloto y, por tanto, no necesariamente respondía a las características que tendrá el QL estándar cuando llegue al mercado español (no antes de setiembre, según estima la empresa Investrónica).

El manual (¿por qué no empezar por este elemento a menudo despreciado por los comentaristas y no siempre cuidado por Sinclair?) es una carpeta de anillas, a la que se pueden añadir o retirar páginas para actualizar la información técnica o de programación. Este detalle lleva a pensar que el fabricante espera introducir novedades lo suficientemente importantes como para alterar el contenido original del manual.

Con el QL que nos facilitó Sinclair Research vinieron cuatro paquetes de aplicación elaborados por Psion para este ordenador, y que se incluyen en su precio. Se llaman QL Quill, QL Abacus, QL Archive y QL Easel. Más adelante hablaremos de ellos. Estos programas vienen alojados en *microdrives* dentro de una caja que se abre como un libro. Y Sinclair ofrece cuatro cartuchos vírgenes para manejar el QL.

Vayamos ahora al ordenador en sí mismo. Mide 138 × 46 × 472 mm, y pesa algo menos de 1,400 kg. Lleva un teclado del tipo QWERTY de 65 teclas, incluyendo barra espaciadora, cinco teclas de función y cuatro más para control de cursor. Para quienes prefieran trabajar con el teclado inclinado, el aparato completo se puede inclinar gracias a unas patas desmontables.

Las características de diseño del *hardware* hacen pensar que el ordenador, pese a su bajo precio, ha sido concebido para evitar fallos atribuibles

Sinclair

a falta de dureza y resistencia. No obstante, habrá que esperar a conocer el porcentaje de reclamaciones que se produzcan durante la primera fase de comercialización (exclusivamente por correspondencia y limitada al Reino Unido) antes de emitir un juicio sobre la fiabilidad física de este modelo, del que esperamos tenga más rudeza que sus predecesores de la misma marca.

En primer lugar, el QL ha abandonado la perenne ranura de expansión típica del diseño Sinclair, a favor de un juego agradable y limpio de conexiones en la parte posterior de la máquina. Podemos destacar la salida para *joysticks*, dos entradas para *interfaces* RS232C (que permiten la conexión de *modem* e impresora profesional), una entrada para televisor y otra para monitor, que permiten dos tipos de resolución: *pixels* de 512 × 256 y de 256 × 256. También posee un enchufe de comunicación de alta velocidad, denominado QLAN, que permite acoplar hasta 64 QL y Spectrum intercambiando datos e información a través de la red a una velocidad de 100 Kbaudios.

Tiene también una conexión a la derecha del aparato para acoplar *microdrives* y una debida entrada para los cartuchos de expansión de ROM y RAM, situada a la izquierda.

En segundo lugar, es una tradición de la casa Sinclair el no colocar un interruptor *on/off*. No obstante, en este modelo, se ha optado por una tecla debidamente colocada a la derecha del aparato, que hace las funciones de *reset*. A ello se puede llegar igualmente desde el teclado pulsante CTRL, ALT y la barra espaciadora a la vez. Las teclas están lo suficientemente separadas como para no pulsarlas por error.

El teclado del QL está diseñado ergonómicamente, como corresponde a estos tiempos, y nos recuerda bastante al del Decisión Mate V, de NCR. Aparentemente, puede parecer un estándar QWERTY, pero tiene ciertos detalles que merecen ser mencionados.

La tecla SHIFT está donde uno espera que esté. Sinclair ha puesto mucho cuidado en diferenciar las teclas CAPS LOCK y SHIFT, para que si se están haciendo programas en SuperBASIC utilizando las funciones que hay encima de las teclas, sus cuatros —4— no se conviertan en dólares —$— debido al cambio de función realizado por el CAPS LOCK. El CAPS LOCK sólo escribe las funciones superiores de las teclas alfabéticas.

Las teclas de función están cuidadosamente separadas del teclado principal para que no se pulsen por equivocación cuando se teclea rápidamente algunas teclas cuya función superior estemos realizando. Las teclas emiten un sonido especial o un *click-click* que hacen la vida más fácil al mecanógrafo.



QL

Finalmente, el QL tiene una hermosa tecla ENTER en forma de L invertida, para que no tenga que buscar alrededor intentando encontrar lo que es la tecla más utilizada e importante en programación y probablemente la más popular de todo el teclado.

Entre las desventajas está la inclusión un tanto desigual del signo C, siendo este el valor cambiado de la tecla ESC. La colocación no estandarizada de las teclas del cursor a ambos lados de la barra espaciadora y el valor ASCII no estandarizado atribuido al signo £ teniéndolo separado como una tecla sin cambio, en la parte superior derecha del teclado.

Pero estas son sutilezas. Puede ser hasta más fácil mover el cursor con las teclas derecha-izquierda situadas a un lado de la barra espaciadora y para moverse de arriba a abajo con las teclas del lado opuesto. En verdad, muchos juegos para microordenadores ya trabajan de esta forma utilizando la Z y la X para ir de izquierda a derecha y las teclas, y . para ir de arriba a abajo.

La posibilidad de variar la inclinación del teclado también facilitará el uso más que otros teclados estándar, que fuerzan la muñeca intentando mantener la actitud correcta hacia el teclado.

El almacenamiento en el QL está constituido por un juego de *microdrives* Sinclair. Este sistema, introducido por primera vez el año pasado (aunque anunciado mucho antes) como accesorio opcional para el Spectrum, pasa ahora a ser un elemento estándar del QL. El *microdrive* utiliza una cinta magnética muy fina de 200 pulgadas enrollada en sí misma. La cinta consta de una bobina única que es alimentada a través de una cabeza de lectura/escritura a una velocidad de 30 pulgadas por segundo, desenrollándose del interior de la bobina y enrollándose en el exterior otra vez. Una vez que se desenrolla totalmente en 7,5 segundos, se rebobina y está preparada otra vez.

Los cartuchos son pequeños pero cuestan 4,95 libras cada uno (no hay indicación alguna de cuanto podrán costar en España). El soporte está situado discretamente a la derecha del tablero. Los *microdrives* del Spectrum no son compatibles con los del Spectrum.

Los *microdrives* del **QL** son versiones mejoradas de los originalmente concebidos para el Spectrum. Su capacidad ha sido aumentada hasta 100 Kbytes (el máximo para el Spectrum era de 85 Kbytes). También es agradable comprobar que Sinclair Research ha conseguido simplificarlos. Para conseguir el catálogo del cartucho basta con teclear CAT MDVI, obteniéndose una lista de los ficheros que contiene el *drive* 1.

El *microdrive* es un dispositivo serie que imita el acceso aleatorio. Su capacidad de 100 Kbytes está dividida en 255 sectores de 512 bytes. Como con un disco, solamente hay que darle el comando para encontrar el fichero deseado, transfiriendo toda la información que encuentre a lo largo de la cinta que se refiera a ese fichero a la memoria. En teoría, el tiempo medio de acceso debe ser la mitad que el tiempo que tarda en rebobinarse, aunque después de largos períodos de utilización, cuando se está cambiando la longitud de los ficheros, añadiendo unos y suprimiendo otros, los sectores de cada fichero tienden a espaciarse a través de la longitud de la cinta. A menos que uno se los prepare concienzudamente, el tiempo de acceso va a ser de cerca de 7 segundos.

Los *microdrives* serán la clave del éxito que pueda obtener el QL. La alternativa de almacenamiento en *diskette* convencional duplicaría el precio del aparato y, a la vez, sería menos cómoda para la comercialización por correspondencia, que es una pieza clave en la estrategia de siempre de Sinclair. Aunque los *microdrives* presentan una innovación considerable, no trabajan como los *floppies*.

Cuando la memoria RAM se abarate lo suficiente, los sistemas como el del QL, basado en el microprocesador 68000, jugarán una importante baza. La velocidad de acceso al almacenamiento será menos problemática con 640 Kbytes de memoria RAM disponible. Y si bien el precio del *pack* de 500 Kbytes de RAM todavía está por anunciarse, se supone que estará alrededor de las 200 libras (en este como en otros precios no conviene hacer una traducción exacta a nuestra moneda). A la larga, podemos esperar que el costo de la memoria pueda abaratarse algo más. Cuando ello ocurra, se podrá cargar toda la información que se pueda necesitar al comienzo de un trabajo. Los ficheros serán accesibles fácilmente, y más rápido que con un Winchester.

Con la ampliación de RAM en 500 Kbytes será posible tener en memoria tantos programas como se necesiten. También será posible tener varios cientos de K de ficheros, todos cargados y preparados para acceder a ellos.

El QL será indudablemente criticado en base al argumento de que los *microdrives* no suministran la velocidad, capacidad ni fiabilidad de las grabadoras de discos, por lo tanto descalificando el producto como máquina seria para aplicaciones de negocios.

En realidad, puede preveerse el éxito del QL en base a un diseño que fue pensado para utilizar memoria masiva interna RAM. Para la mayor parte de las aplicaciones comerciales será una herramienta más rápida y más flexible que cualquier microordenador de los que actualmente se venden para ese uso a precios cuatro veces más caros.

También en el campo del *software* puede pronosticarse que el QL será un éxito instantáneo. Con cuatro paquetes de aplicación incluidos en su precio, seguro que el nuevo modelo de Sinclair no hallará rival. No es exagerado decir que los programas, bajo circunstancias normales, valen por sí mismos el precio de venta del QL. Claro que los productos Sinclair nunca representan circunstancias normales. Y hasta hay quien se atreve a apostar que los precios del *software* para microordenadores de negocios habrán de bajar si los

fabricantes quieren competir con el QL, a condición de que Sinclair Research y sus distribuidores hagan bien su trabajo de *marketing*.

Una vez que se escribe un paquete de aplicaciones es relativamente sencillo hacer una copia. ¿Qué mejor forma de que un micro penetre en el mercado que regalando programas?

La clave del sistema operativo Q-DOS de Sinclair reside en la capacidad multitarea. Multitarea quiere decir tener varios programas ejecutándose simultáneamente. Todos los microordenadores lo hacen hasta cierto punto. Siempre hay un programa que controla la pantalla y otro que controla el teclado al mismo tiempo, mientras el programa principal se está ejecutando.

El número de tareas que un microordenador puede realizar depende directamente de la rapidez con que se procesa la información. Como el *chip* 68000 es muy potente, es posible estar haciendo múltiples tareas al mismo tiempo, sin que el usuario se vea afectado por la pérdida de velocidad. El procesador ejecuta una ínfima parte de un programa y luego de otro y así sucesivamente, lo suficientemente rápido como para parecer que lo está haciendo de forma simultánea.

Por ello, cuando se tiene un procesador muy potente, se pueden diseñar aplicaciones para aprovechar esa capacidad. Los cuatro paquetes diseñados por Psion para el QL (de los que nos ocuparemos en profundidad en otro número de la revista, así como del lenguaje SuperBASIC, que todavía no hemos tenido ocasión de experimentar a fondo) trabajan en multitarea bajo la forma de ventanas. De esta manera se pueden ejecutar dos o más programas y tenerlos en pantalla por separado.

Si a ello añadimos la facilidad de transferir datos de un programa a otro, podremos comprobar la potencia del QL. Los paquetes de Psion permiten sacar por pantalla una página de la base de datos mientras se preparan documentos para el tratamiento de textos, por ejemplo.

Y luego, por qué no, también habrá juegos. Las posibilidades del QL son enormes en este terreno, aunque sus diseñadores se empeñan en subrayar que no ha sido pensado para jugar. Piénsese que, precisamente por sus características que acabamos de reseñar, se pueden tener tantos gráficos ejecutándose simultáneamente en pantalla como se quiera.

En palabras de Clive Sinclair, la compañía no hace previsiones acerca del uso que la gente hará del QL. Ya en su día le sorprendieron las múltiples aplicaciones recreativas que los usuarios encontraron al Spectrum, y está dispuesto a sorprenderse de nuevo con el QL.

Tal vez por ello, hay en el diseño del QL detalles que dan que pensar. Por ejemplo, hay entrada para *joysticks*, para cartuchos de juegos, *microdrives*, *slot* de expansión de RAM, todo lo cual ofrece la suficiente flexibilidad para asegurar que puede responder a las aplicaciones que se le ocurran a los usuarios.

Es difícil no dejarse llevar por la atracción del QL. A no ser que encuentre problemas o rechazos a la hora de su utilización (recordamos que sólo hemos trabajado con un piloto), es muy probable que sea serio candidato al galardón de microordenador del año. Por su parte, el fabricante está seguro de haber identificado una demanda por ordenadores de consumo masivo, de modo que el alto volumen de ventas mantenga los costes bajos y los costes bajos aseguren un alto volumen de ventas. Esta es su estrategia, y hasta ahora le ha dado resultados innegables.





QL

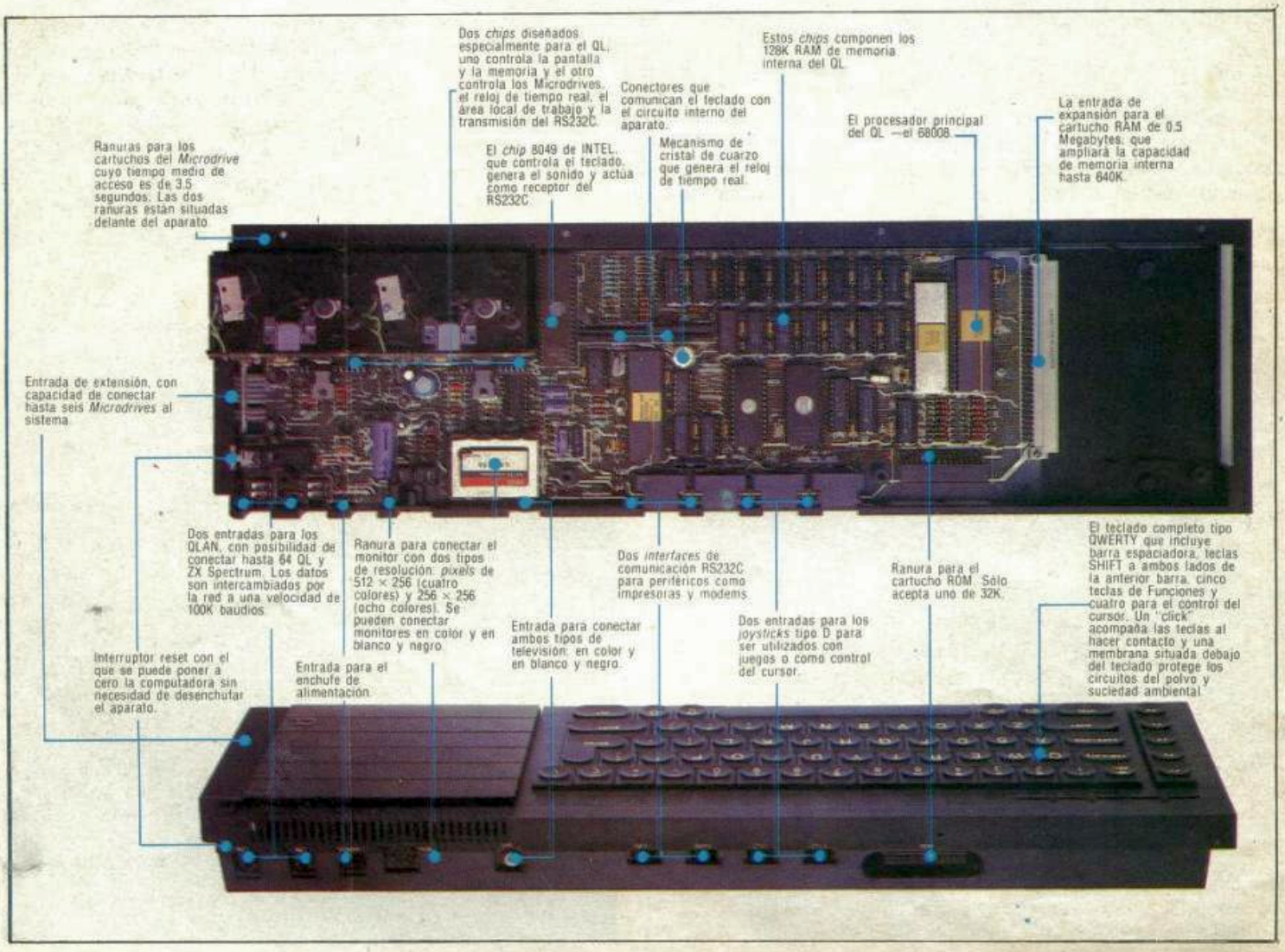
Pros

Teclado. Como muchas máquinas baratas de hoy día, el teclado del QL está basado en el sistema de membrana, que es de alguna manera la versión glorificada del Spectrum. Este sistema es de gran utilidad a la hora de proteger el aparato del polvo y suciedad ambientales. El teclado está bien equipado, con un total de 65 teclas que tienen acceso a la totalidad de los caracteres ASCII. Esto incluye algunos caracteres que parecen extraños pero que serán fundamentales cuando Sinclair Research implemente la versión del lenguaje de programación C en el QL.

Memoria. La ventaja más grande del *chip* 68008 es la memoria. En principio, el QL tendrá 128 Kbytes, pero Sinclair Research ha prometido desarrollar un *chip* para ampliarla hasta 512 Kbytes (medio megabyte). Mucha memoria es la mejor alternativa para el almacenamiento masivo que necesita un ordenador de este tipo.

Aunque los *microdrives* del QL son relativamente lentos en comparación con los *floppies*, se podrá encender el QL y cargar grandes programas con todos sus datos a la memoria, todo con un acceso superior a muchos microordenadores CP/M que ofrecen una memoria de sólo 64 Kbytes.

Lenguaje. El QL viene listo para ser programado en el nada modestamente llamado (la modestina no es virtud de Sinclair) SuperBASIC. Es un lenguaje de programación relativamente nuevo, que tiene poco que ver con el BASIC anterior de Sinclair, que conocen los usuarios de Spectrum. La verdad es que, en comparación con el BASIC clásico tiene todavía menos que ver y, para ir más lejos, creemos que el nombre sólo está para que los novatos sepan que se



trata de un lenguaje de programación. De todos modos, todavía tiene comandos tradicionales como el GO TO y el GO SUB, para que no sea imposible convertir programas ya existentes al QL.

El SuperBASIC es un lenguaje muy estructurado, con nombre de procedimientos y funciones. La capacidad numérica es impresionante, manejando números desde -10^{-615} hasta 10^{615} con una cantidad prácticamente ilimitada de cifras significativas. Las cadenas pueden tener una longitud de 32 Kbytes y Sinclair Research proclama que el lenguaje tendrá capacidad multitarea, permitiendo ejecutar varios programas a la vez. Solamente el tiempo nos dirá si el SuperBASIC es tan bueno como parece. Y nuestros lectores se enterarán lo antes posible de nuestra opinión al respecto.

Software. Como se dice en la nota principal, el QL viene con cuatro clásicos programas de aplicación incluidos en el precio. Hay un procesador de textos, una base de datos, una hoja de cálculo y un paquete de gráficos. Es la primera vez que un ordenador personal se vende con un juego serio de programas. Estas aplicaciones han sido preparadas por la empresa Psion y, a simple vista, todo hace presagiar que tendrán buena acogida.

Según Psion, la hoja de trabajo QL Abacus tiene muchas de las características y velocidad que son propias del Multiplan de Microsoft. Por su parte, el programa QL Archuve ofrece facilidades que no existen en equipos actuales. Es programable, no sólo a través del teclado, utilizando unos cuantos comandos simples, sino que también se pueden hacer programas con su propio lenguaje interno. De esta forma se pueden hacer bases de datos utilizando las facilidades y los comandos preparados para ello. Muchas operaciones complejas se pueden hacer con una simple instrucción. El procesador de textos QL Quill es muy rápido y ofrece una gama entera de facilidades. Esto le permite ver los textos y la segmentación de páginas exactamente igual que en la impresora.

Psion no solamente ha conseguido programas buenos, sino que también los ha hecho fáciles de ejecutar. Los programas conducen al usuario a través de las operaciones, enseñándole lo que ha hecho, lo que hay en la pantalla y los resultados que puede conseguir.



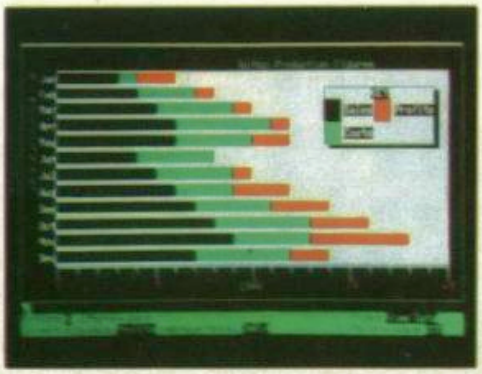


Esta hoja de trabajo permite usar el texto ya existente en vez de un complejo sistema de letras y números, como una referencia para la manipulación de filas enteras, columnas o celdas individuales. También hay un juego de funciones que permiten al usuario juntar páginas de trabajo, utilizar ventanas múltiples, varias la anchura y centrar textos, usar unidades diferentes.



Con la incorporación de unos comandos sencillos y el panel de control en la pantalla, el QL Quill está lleno de opciones, como pueden ser: márgenes de variaciones en tiempo real, formateado de páginas, centrado de textos, etc.





Este paquete interactivo permite al usuario crear gráficos sin necesidad de formatear la pantalla antes de los datos de entrada: por consiguiente el programa puede diseñar y poner la escala de acuerdo con los parámetros de la información de entrada. Hay ocho formatos preestablecidos, permitiendo la creación de diagramas de barras, combinación de gráficos de líneas con los diagramas de barras y diagramas en círculo —pudiéndose colorear para que destaque determinadas zonas.



Este paquete es una base de datos utilizando un lenguaje basado en las técnicas de tratamiento de ficheros, con unos 80 comandos que permiten una búsqueda y selección flexible.

QL

Y, como todos los micros actuales, tiene un comando HELP que visualizará a través de la pantalla un pequeño manual de instrucciones cada vez que se necesita.

Comunicaciones. El *interface* del QL le permite conectar con varios periféricos. No habrá problemas para que el QL "hable". Es más, con el teclado adecuado, un *display* de 80 columnas y dos *microdrives*, el QL puede hacer las veces de terminal. Cabe preguntarse quién será el primero en desarrollar programas para que el QL funcione como terminal.

Contras

El chip de 32 bits. Verdaderamente no lo es. El QL utiliza el *chip* 68008, que es una versión especial de la familia 68000 de Motorola de 16 bits. Concretamente, el 68008 es interiormente de 16 bits pero tiene la característica de poder manejar palabras de 32 bits.

De todos modos, dejando de lado esta cuestión, hay que considerar las ventajas del 68008. Es rápido, permite utilizar mucha memoria directamente y su *software* es compatible con el de su hermano mayor, el 68000. Si se está familiarizado con el código máquina, el 6800 tiene un gran juego de instrucciones y es muy fácil de aprender.

En pocas palabras, ¿a quién le importa el número de bits que tenga?

No posee conexión Centronics. ¡Hasta el Oric la tiene! Una salida paralela Centronics hubiera permitido conectar muchas impresoras profesionales directamente en la parte posterior del QL, utilizando un enchufe estándar. La mayoría de las impresoras tienen *interfaces* Centronics y habrá que pagar más para conseguir una RS 232. El otro problema al utilizar RS 232 es que habrá que controlar la velocidad en baudios y hasta se necesitará un cable especial con su adaptador y cableado adecuado para la impresora.

Sinclair Research está trabajando en una salida Centronics. Pero como muchos usuarios del Spectrum están acostumbrados a enchufar varios periféricos a la vez, el usuario del QL va a encontrar un poco pesada la cantidad de conexiones que lleva el nuevo aparato.

No hay disco. Sorprende que el QL venga con *microdrives*, aunque se nos diga que son mejores que los del Spectrum. Lógicamente, este dispositivo de almacenamiento contribuye a que la máquina sea más barata, pero los *microdrives* son muy lentos, por lo que la velocidad a que pueden correr los programas se verá mermada. El QL no funcionará con el DOS del microprocesador 68000 mientras venga con *microdrives*, y en la actualidad no hay posibilidad de añadir *floppies*, a no ser que se invierta una suma cercana al precio del ordenador para disfrutar de ese privilegio.

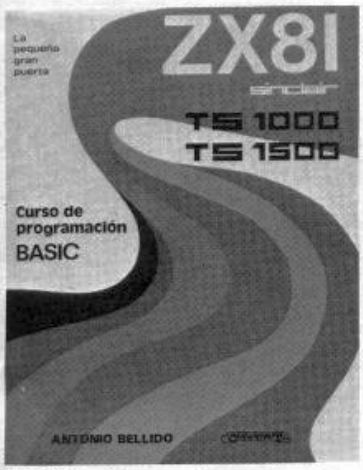
Sinclair está trabajando actualmente en un disco que le dé al QL el potencial necesario para trabajar con el sistema operativo Unix, pero parece ser que no hará ningún anuncio antes de haber investigado a fondo la cuestión. ¿Con qué se le puede respaldar? Un disco Winchester de 5 megabytes tiene 5.000 Kbytes de datos, mientras que cada microdrive posee 500 K, lo que significa que harían falta 50 *microdrives* para respaldar a un disco.

Es una máquina a color. El QL sólo tiene cuatro colores en su modo normal de operación, por lo que se hace obvia la necesidad de utilizarlo con un monitor color. No hay ni un *slot* para conectar un monitor blanco y negro. La única dificultad será que muchos usuarios van a tener problemas a la hora de diferenciar el rojo y el verde en una pantalla monócroma. Esto no significa que no pueda hacerse; simplemente que será difícil distinguirles.

Encerrado en su propia arquitectura. Significa que el QL utiliza su propio BASIC y DOS, lo que dificulta el intercambiar programas y accesorios con otros aparatos. Esto puede ser un *handicap*. Otros fabricantes aseguran que haciendo equipos de acuerdo a un estándar características estándar tendrán todo el *software* necesario, mientras que quien adopte un camino particular acabará acudiendo al estándar.



Su BIBLIOTECA de INFORMATICA



ZX81. Curso de programación BASIC. Bellido. 128 páginas. 2ª edic. 1983. 850,- Ptas.

INDICE EXTRACTADO: Introducción general. Manejando el ZX81. Programando en BASIC. Practicando el BASIC con el ZX81.



Cómo programar su SPECTRUM. Bellido. 132 páginas. 2ª edición. 1984. 850,- Ptas.

INDICE EXTRACTADO: Cómo utilizar el Spectrum. Cómo programar el Spectrum. Introducción para principiantes. Introducciones específicas. Expresiones y operadores lógicos.



Cómo usar los colores y los gráficos en el SPECTRUM. Bellido. 96 páginas. 1984. 850,- Ptas.

INDICE EXTRACTADO: Caracteres gráficos. Colores. La casualidad. El movimiento. La resolución de gráficas. Procesos internos. PEEK y POKE. Memoria. Pantalla. Gráficos profesionales.



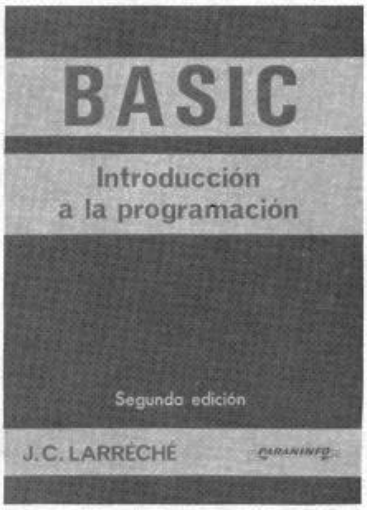
BASIC. Curso acelerado. De Rossi. 224 páginas. 1984. 850,- Ptas.

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



BASIC. Programación de microordenadores. Checroun. 112 páginas. 2ª edición. 1984. 500,- Ptas.

INDICE EXTRACTADO: Los microordenadores. El lenguaje BASIC. Extensiones del BASIC. Los ficheros. Problemas de aplicación. Recapitulación de las instrucciones de BASIC. Las funciones incorporadas o de bibliotecas.



BASIC. Introducción a la Programación. Larreche. 132 páginas. 2ª edic. 1984. 550,- Ptas.

INDICE EXTRACTADO: Introducción a la programación. Definición del lenguaje BASIC. Realización de un programa en BASIC. Programas útiles. Extensión del BASIC. Lenguaje de comandos de los sistemas HP 2000 B, C y F.



Diccionario de informática. Inglés-español. Olivetti. 272 páginas. 4ª edición. 1984. 750,- Ptas.

Facilita la traducción literal y el concepto científico y técnico de cada vocablo. Glosario de términos.



Programación con el lenguaje COBOL. Galán Pascual. 328 páginas. 1983. 975,- Ptas.

INDICE EXTRACTADO: Conceptos previos sobre informática básica. Introducción. "Identification division". "Enviroment division" "Data division". "Procedure division". Instrucciones de entrada/salida. Instrucciones de manipulación de datos, de bifurcación, de cálculo, condicionales y de control. Creación y manejo de tablas. Generación automática de informes. Clasificación de ficheros. Su organización en dispositivos de acceso-directo. Sugerencias de orden práctico. Apéndices.

COPIE O RECORTE ESTE BOLETIN DE PEDIDO

Deseo recibir a reembolso los siguientes libros:

COMO USAR LOS COLORES Y LOS GRAFICOS EN EL SPECTRUM .....

.....

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

Nombre .....
Domicilio .....
Ciudad .....

COMPRO·VENDO·CAMBIO·COMPRO·VENDO·CAMBIO



ZX pone sus páginas a disposición de los lectores que deseen publicar anuncios clasificados para compra-venta de ordenadores y periféricos, intercambiar programas o simplemente tomar contacto con otros usuarios del Sinclair. La publicación de estos anuncios será gratuíta. No serán aceptados los que tengan carácter publicitario. Enviar los textos, acompañados de dirección, identificación y teléfono a:
Revista ZX.
Tablero de anuncios
C/Jerez, 3 Madrid-16.

◆◆◆

— Se vende un ordenador ZX-81 con ampliación de memoria a 16K Ram. Precio a convenir. El precio incluye: transformador, para la televisión, manual en castellano.
Preguntar por: Jorge Alvarez. C/Guzmán el Bueno, 17, 1.ª derecha.
Madrid.
Telf.: (91) 449 59 75.

— Intercambio programas e información.
Preguntar por: Antonio.
Madrid.
Telf.: (91) 457 48 49.

— Vendo para ZX-81 o Spectrum: fuente de alimentación completamente nueva, es la original y amplificador de sonido Spectrum 10 watios, con altavoz y volumen, con caja compacta y pequeña (2.000 Ptas.).
Preguntar por Eduardo Cumella.
Barcelona.
Telf.: (93) 201 56 70 (mañanas o tardes).

— Cambiamos programas de ordenador ZX-Spectrum, tenemos bastantes.
Preguntar por: Miguel.
Madrid.
Telf.: (91) 429 28 13 (a partir de las 22 horas).

— Vendo Spectrum 48 K, 45.000 Ptas., garantía y juegos.
Preguntar por: Carmen. C/ Meléndez Valdés, 18, 1.°, 10.
Madrid.
Telf.: (91) 419 57 34.

— Me gustaría intercambiar programas con usuarios del ZX Spectrum 48 K.
Preguntar por: Gustavo Marín. C/ Joaquín M.ª López, 14.
Madrid-15.

— Nos gustaría tomar contacto con usuarios del Spectrum de Madrid, preferentemente zona Ciudad Lineal, para intercambio de programas de todo tipo. No hay problemas para la duplicación.
Preguntar por: Nacho o Julio.
Telf.: (91) 267 05 42 (Nacho).
Telf.: (91) 407 78 89 (Julio).

— Vendo un ordenador ZX-81, comprado en "Reyes". Está nuevo. A toda prueba: manual caja y conectores.
Madrid.
Telf.: (91) 259 33 85.

— Vendo barato, o cambio por otro programa, VU-FILE a estrenar.
Preguntar por: Agustín.
Madrid.
Telf.: (91) 413 97 28.

— Intercambio programa de 16 K para ZX-81. Amplísimo stock, más de 150 programas de todo tipo. También vendo programas a precios de auténtica ocasión y en grabaciones de calidad. Intercambiaría además programas, listados y copias de libros de programación.
Preguntar por: Iñaki Castillo. C/ Cataluña, 27, 3 B.
Bilbao (Vizcaya).
Telf.: (94) 440 29 99.

— Deseo cartearme con usuarios de ZX-81 de 1K y 16K. Compro amplificador de memoria para ZX81. Precio máximo 5.000 ptas.
Preguntar por: Fernando Navarro. C/ Fuencarral, 9, 2 B.
Madrid-4.
Telf.: (91) 221 06 03.

— Vendo ZX-81 con teclado profesional, ampliación 16 K original Sinclair, manual de instrucciones en español y fuente de alimentación con cables de TV. Todo por 23.000
Preguntar por: Enrique Daroqui Raga. C/ Ramón y Cajal 18, Pta. 6.
Albal (Valencia).
Telf.: (96) 126 52 00.

— Vendo mi ZX Spectrum de 48 K. Garantía hasta junio 84.
Preguntar por: Salvador Hnos.
Barcelona.
Telf.: (93) 207 48 00 ó (93) 427 55 32.

— Vendo ZX-81 con 16K de memoria completo con cables y fuente de alimentación y manual en castellano, fecha de compra 2/4/83 por 17.000 Ptas. También vendo cintas juegos indecon para 16k a 1.000 ptas. cada una.
Preguntar por: José Luis.
Madrid.
Telf.: (91) 473 74 51 (noches).

— Interesado en intercambiar toda clase de programas educativos y de aplicaciones.
Telf.: (91) 739 76 30 (tardes).

— Vendo ordenador ZX-81, con amplificador de memoria a 16 Kbytes incorporada, y 40 programas de juegos, matemáticas, etc., por 16.000 Ptas. Fue comprado hace 4 meses y se encuentra en perfecto estado de uso. Incluye la alimentación y todos los cables, e instrucciones en inglés y español.
Telf.: (91) 448 04 31 (a partir de las 18,30 horas).

COMPRO·VENDO·CAMBIO·COMPRO·VENDO·CAMBIO

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

Cómprela en su kiosco habitual o solicítela a:

ORDENADOR POPULAR
Jerez, 3
Tel. 457 45 66
MADRID-16

En el número anterior informábamos a los usuarios del Spectrum cómo conocer la memoria ocupada por un programa. Los usuarios del ZX81 también disponen de una instrucción similar:

```
10 PRINT (PEEK 16396+256*PEEK
16397)-16565
```

Informa sobre la memoria ocupada por el programa que se encuentre en memoria.

El ordenador, como muchos de vosotros sabéis, trabaja internamente con ceros y unos (en binario). A cada cero o uno se le llama bit y una combinación de 8 bits (un número de ocho cifras compuesto de ceros y unos) se le llama byte u octeto. Las combinaciones posibles van desde la cifra 00000000 a la 11111111, que suman 256 (de la cero a la 255 para ser más exactos). Cada una de estas combinaciones nos da un carácter determinado. Esto es lo que se conoce por el código ASCII.

A la diferente combinación de caracteres podemos acceder a través de la instrucción CHR$. Así PRINT CHR$ 97 nos dará la "a" minúscula, RPINT CHR$ 65 nos dará la "a" mayúscula, o PRINT CHR$ 122 la "z" minúscula.

También podemos conocer el código ASCII de un carácter almacenado en una variable. Para ello utilizamos la instrucción CODE. Así, PRINT CODE "a" nos dará 97, PRINT CODE "A" nos dará 65 ó PRINT CODE "z" nos dará 122.

Conociendo esto podemos preparar una pequeña rutina para cambiar el valor de una variable alfanumérica o *string* a mayúsculas o a minúsculas, según queramos.

Las minúsculas van del código 97 a 122 y las mayúsculas, del 65 al 90. Bastará, pues, comprobar el código para saber si hay una letra (los espacios en blanco responden al código 32) y después cambiarla según deseemos.

Es decir, si tenemos una letra en minúscula podemos obtener su mayúscula como el CODE de la letra minúscula menos 32. Y si la letra es mayúscula, el CODE de dicha letra más 32 nos dará la letra mayúscula.

```
  20 INPUT "Introduzca su nombre
";x$
  22 FOR x=1 TO LEN x$
  23 LET t=CODE x$(x TO x): GO S
UB 100
  30 NEXT x
  40 FOR x=1 TO LEN x$
  45 LET t=CODE x$(x TO x): GO S
UB 200
  60 NEXT x
  70 GO TO 22
 100 REM Mayusculas
 130 IF t>96 AND t<123 THEN LET
x$(x TO x)=CHR$ (t-32)
 140 PRINT AT 5,5+x;x$(x TO x):
RETURN
 200 REM Minusculas
 210 IF t>64 AND t<91 THEN LET x
$(x TO x)=CHR$ (t+32)
 220 PRINT AT 5,5+x;x$(x TO x):
RETURN
```

Lo dicho hasta aquí es válido sólo para el Spectrum, ya que el ZX81 no dispone de letras mayúsculas. Pero podemos hacer el cambio con la inversión de imagen, pues en el ZX81 las letras están situadas entre el código ASCII 38 y 63 y esos mismos en inversión de imagen se sitúan entre los códigos 166 y 191.

Para ello se podría modificar las siguientes instrucciones:

```
 130 IF T>37 AND T<64 THEN LET X
$(X TO X)=CHR$ (T+128)
 210 IF T>165 AND T<192 THEN LET
 X$(X TO X)=CHR$ (T-128)
```

Siguiendo con el ejemplo anterior, podemos añadir una rutina que subraye la información en pantalla. Para ello, podemos utilizar el valor ASCII 95 con OVER 1 para no borrar la información existente en pantalla.

```
  20 INPUT "Introduzca su nombre
";x$
  22 FOR x=1 TO LEN x$
  23 LET t=CODE x$(x TO x): GO S
UB 100
  30 NEXT x
  40 FOR x=1 TO LEN x$
  45 LET t=CODE x$(x TO x): GO S
UB 200
  60 NEXT x
  70 GO SUB 500: PAUSE 50: GO TO
 22
 100 REM Mayusculas
 130 IF t>96 AND t<123 THEN LET
x$(x TO x)=CHR$ (t-32)
 140 PRINT AT 5,5+x;x$(x TO x):
RETURN
 200 REM Minusculas
 210 IF t>64 AND t<91 THEN LET x
$(x TO x)=CHR$ (t+32)
 220 PRINT AT 5,5+x;x$(x TO x):
RETURN
 500 REM Subrayar
 510 FOR x=1 TO LEN x$
 520 PRINT AT 5,5+x; OVER 1;CHR$
 95
 530 NEXT x: RETURN
```

```
JUANITO



juanito
```

Más de una vez habrán visto paréntesis como los de la línea 30 del programa que le ofrecemos a continuación. Podemos llamarlos "paréntesis lógicos", ya que evalúan o comparan el valor de la variable o variables que en él se introducen y devuelven el valor 1 si se cumple la igualdad o 0 si no se cumple. Son muy prácticos y permiten ahorrar muchas instrucciones y, por tanto, memoria. En nuestro programa la línea 30 ha de

leerse como "a" igual a 1, si i = 1; más 20, si i = 2; más 30, si i = 3; más 35, si i = 4; más 50, si i = 5 y m$ = "ZX"; más 100, si i = 1 ó i = 5; más 200, si i = 2 e i = 4. Ejecutando el programa le aparecen los cinco valores que toma la variable "a" para los valores de "i" de 1 a 5. (Al decir, por ejemplo, y = 35 * (1 = 4) puede resultar y = 35 * 0 ó y = 35 * 1, si el valor de "i" es distinto de cuatro o cuatro, respectivamente). ¿Vio cómo funciona? Pueden utilizarse tanto en el ZX81 como en el Spectrum y algún manual lo ha descrito como "Programación avanzada".

Lo más importante de un programa es que funcione bien. Pero si además tiene una buena presentación en pantalla resultará más atractivo. Fíjese en la instrucción 30 del siguiente programa porque sirve para centrar la información contenida en X$. Como la pantalla tiene 32 columnas nos situamos en la parte central (32/2) y retrocedemos LEN(X$)/2 posiciones. Esto mismo puede aplicarse a cualquier impresora, para lo cual sólo habría de cambiarse el valor 32 por el número de caracteres por línea que admitiese la impresora.

```
   10 REM Parentesis logicos.
   15 LET M$="Zx"
   20 FOR i=1 TO 5
   30 LET a=(i=1)+20*(i=2)+30*(i=
3)+35*(i=4)+50*(i=5 AND m$="Zx")
+100*(i=1 OR i=5)+200*(i=2 AND i
=4)
   40 PRINT i,a
   50 NEXT i


1                    101
2                    20
3                    30
4                    35
5                    150
```

```
   10 REM Centrar strings.
   15 LET x=1
   20 INPUT x$
   30 PRINT AT x,32/2-LEN (x$)/2;
x$
   40 LET x=x+1: IF x=20 THEN CLS
  : GO TO 15
   50 GO TO 20


          Nabucodonodor
```




SUSCRIBASE POR TELEFONO

* más fácil,
* más cómodo,
* más rápido

(91) 4572617

Servicio permanente durante las 24 horas del día
"CONTESTADOR AUTOMATICO"

SUSCRIBASE A ZX

Como recordarán los lectores, esta sección permanente está dedicada a comentar y analizar críticamente los programas para ordenadores Sinclair que se ofrecen en nuestro mercado. Ocasionalmente, habremos de incluir también programas que se venden en el extranjero —principalmente en Gran Bretaña— cuando los consideramos de interés para nuestros lectores.

Aún aceptando que siempre habrá una cuota de subjetivismo en nuestros juicios, corresponde que expliquemos, al menos en esta primera publicación, cuáles son los criterios que seguimos a la hora de evaluar los programas.

En el caso de los juegos, tomamos en cuanta, en primer lugar, las cualidades de programa para retener la atención del jugador y, por así decir, para crear adicción. El segundo criterio es la presentación formal, tanto en su aspecto externo como en pantalla. Tercero será la calidad de la resolución gráfica lograda. Y, por último, tomamos en consideración, el movimiento, la acción que genera el juego.

Cuando analizamos programas de aplicaciones, como es el caso de la contabilidad del hogar que comentamos en este número, el primer criterio a observar será su utilidad; el segundo, como en los juegos, la presentación. En reemplazo de la resolución gráfica consideramos la claridad y facilidad de uso y, para concluir, la rapidez de ejecución.

Programa: Stonkers
Tipo: juego
Distribuidor: ABC Analog
Formato: cinta de cassette
ZX Spectrum 48 K

Este juego es una versión reducida de un *Wargame* (simulación de combate entre dos ejércitos), ya que las versiones normales (no para ordenador) de estos juegos revisten una gran complejidad y suelen ir acompañados de un manual que deja pequeño al del Spectrum si no en tamaño, sí en dificultad de comprensión); en Stonkers se han intentado evitar estos problemas simplificando las instrucciones y las unidades de combate.

Cada jugador (la persona y el ordenador) empiezan con el mismo número de unidades y con la misma fuerza, siendo su objetivo la conquista de la fortaleza enemiga o la destrucción de sus tropas. Hay que indicar que no es un juego de acción propiamente dicho, aunque la acción se realice en tiempo real (el jugador y el ordenador se mueven a la vez) no lleva implicado el aumento de adrenalina de estos otros juegos.

Al empezar se nos presenta un plano del campo de batalla y de la disposición de las unidades de combate. Al ser este plano de gran tamaño podemos ampliar una zona específica para poder observarla con atención. Una vez elegida una unidad, nos colocamos encima de ella, y podemos saber su estado (provisiones, fuerza de combate, etcétera) y si nos interesa moverla lo indicamos desplazándonos hasta el punto de destino y una vez dada la orden la unidad se pondrá en marcha (sus movimientos no son instantáneos, lo que le da realismo al juego), estas unidades agotan energías durante los desplazamientos por lo que debemos reaprovisionarla cada cierto tiempo con los camiones previstos a tal efecto, que a su vez se aprovisionan desde un barco. Cuando dos unidades contrarias se encuentran se enzarzan en una lucha hasta que una de las dos muere (o se retira). Naturalmente otras unidades pueden ir a la lucha formándose una gran *melé*. Mientras todo esto sucede en la parte inferior de la pantalla nos van apareciendo mensajes



(en inglés) indicándonos diversos hechos: la llegada del barco de aprovisionamiento, unidades desabastecidas, etc.

El juego es ameno y se acerca más al tipo de juegos de inteligencia (ajedrez, damas) que a los de acción, aunque comparta ideas de ambos. Hemos observado un fallo cuando una unidad viva se coloca junto a una muerta (las unidades muertas no desaparecen, sino que se transforman en calaveras), ya que en esa situación la unidad viva se queda inmovilizada y no hay manera de sacarla de ese estado. Todo lo demás funciona bien y va presentado con preciosas gráficas y otros detalles de muy buen gusto.

PUNTUACION:
■ ■ ■ ■ □

Programa: Android one
Tipo: juego
Distribuidor: Venta-Matic
Formato: cinta de cassette
ZX Spectrum 16 o 48 K

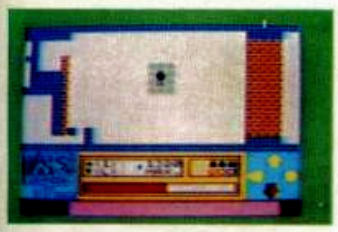
En la mayoría de los juegos el ordenador responde a nuestras órdenes como si fuese un robot. No hay nada más lógico, pues, que un juego en el cual se esté controlando un robot (o androide, si prefiere la terminología moderna) que, inevitablemente, debe destruir a numerosos enemigos para lograr su objetivo.

Las instrucciones de carga son claras (en inglés, por cierto) y al ejecutarlas el programa leído nos indica que apaguemos la cinta. Una vez hecho esto nos pregunta si queremos instrucciones, al responder que si nos las muestra en pantalla (esta vez en castellano) y una vez terminadas nos indica que pongamos en marcha el *cassette* para leer el juego, ya que el programa que hemos cargado sólo eran las instrucciones. Si hubiéramos respondido que no a la pregunta de si queremos las instrucciones, hubiera saltado directamente a este punto. Este proceso de carga en dos partes, aunque útil las primeras veces, a la larga se hace tedioso y sería interesante disponer de un sistema que lo abreviase.

El juego en sí consta de un androide que controlamos haciendo que se mueva y que utilice sus armas (un escudo de defensa y un láser para atacar) con el objetivo de llegar al corazón de un reactor nuclear y destruirlo. Para lograrlo deberemos atravesar innumerables paredes (que hábilmente destruiremos con nuestro láser)

y tendremos que luchar con diversas clases de monstruos y otros obstáculos que intentarán aniquilarnos. Como es evidente, sólo tenemos un número limitado de vidas que deberemos conservar con el mayor cuidado posible.

El manejo se puede realizar por medio del teclado o de un *joystick*, en el primer caso (no teníamos un man-



do de juegos a mano) resulta bastante difícil habituarse al modo de juego, en primer lugar por la colocación de las teclas y en segundo porque el sistema de movimiento es distinto al que se usa habitualmente. En este juego dos teclas hacen girar al androide (cada una para un lado, izquierda o derecha) y otra lo hace desplazarse en la dirección en que esté mirando. Esto último hace que el jugador se desilusione un poco, ya que quiere pelear contra otros robots y no contra un teclado al que no se habitúa. Suponemos que el uso del *joystick* mejorará este punto.

El programa está bien hecho con excepción del pequeño fallo del teclado y posee unas gráficas vistosas un realismo y animación bastante bien hechos.

PUNTUACION:
■ ■ ■ □ □

Programa: Jumping Jack
Distribuidor: ABC Analog
Formato: cinta de cassette
ZX Spectrum 16 o 48 K

Jack el saltarín es un divertido muñeco que tiene que ir subiendo por distintos niveles sin pararse nunca. Además, debe evitar el caerse o que le atrapen los monstruos que hay sueltos por la pantalla.

El juego viene acompañado por un manual en español con instrucciones claras de carga y funcionamiento. Una vez puesto en marcha se ve a Jack en la parte inferior de la pantalla y una serie de pisos con agujeros por los que tiene que



saltar y desplazarse. Los tres movimientos que puede hacer son: correr hacia ambos lados y saltar para arriba. Si salta cuando tiene un agujero encima, habrá logrado subir un piso más; si, en cambio, no logra acertar en el agujero, se dará un golpe, y se quedará atontado y quieto durante algunos segundos; además, si se cae abajo del todo, por uno o varios agujeros, pierde una vida. Al leer las instrucciones y empezar a jugar no parece muy divertido debido a su facilidad. Error enorme; los agujeros de los pisos, que al principio son pocos, empiezan a aumentar, y además hay agujeros desplazándose en las dos direcciones (izquierda y derecha), por lo que nos atrapan fácilmente y caemos quedándonos atontados durante estos segundos que no nos podemos mover; aparece otro agujero (parece que lo hacen a propósito) y nos volvemos a caer al piso inferior y nos volvemos a quedar atontado y... La acción del juego es trepidante sin dar un segundo de descanso. Si logramos subir una pantala, aparecerán versos de un poema que se completa según vayamos subiendo de nivel y además, después de la primera pantalla empezarán a aparecer monstruos que si nos pillan, nos atontan (y ya se sabe qué pasa entonces). Además de los controles que mencionamos antes existe una tecla (la Z para ser exactos) que hace que el juego se detenga (muy útil cuando nos llaman al teléfono en medio de una partida; ojalá todos los juegos tuviesen una tecla similar).

La idea del juego es sencilla y está muy bien desarrollada, evitándose complicadas manipulaciones. Aunque las gráficas no sean excesivamente buenas el juego merece la pena por la animación, pero tenga cuidado, es extremadamente adictivo.

PUNTUACION:
■ ■ ■ □ □

Programa: Video Display
Tipo: Utilidad
Distribuidor: Venta-Matic
Formato: cinta cassette
ZX Spectrum 16 ó 48 K

Este programa es un generador de textos para pantalla que permite realizar programas educativos y de demostración con vistosidad. Dispone de su propio lenguaje de programación (bastante sencillo), lo que le da una gran versatilidad, viniendo la cinta con dos versiones, una para 16 K y la otra (más potente), para 48 K.

Las instrucciones dejan algo que desear en cuanto a claridad, pero una vez comprendidas su uso no plantea mayores problemas. Hay que indicar que en cada cara de la cinta hay dos programas propiamente dicho. Para ejecutar la demostración debemos teclear LOAD "EJEMPLO" y para crear nuestros propios textos hay que poner LOAD "VID.DISP."; hemos aclarado este punto aquí, ya que en las instrucciones no viene explicado y da lugar a confusiones. El programa nos da 11 tipos de letras de distintos tamaños y formas (cuatro en el Spectrum de 16 K) que podemos combinar para imprimir mensajes en pantalla. Los comandos se meten como líneas REM de un programa BASIC y se ejecutan haciendo una llamada del tipo USR, lo que le da una gran potencia, ya que un programa BASIC puede hacer uso de él. Dispone de un amplio conjunto de instrucciones que permite definir ventanas de presentación, nuevos colores, hacer desplazamientos en las cuatro direcciones, etc.

Posee un indudable atractivo, sobre todo para aquellas personas que se dediquen a hacer programas con mucha presentación por pantalla: demostraciones y cosas por el estilo. Su utilidad se ve aumentada por la posibilidad de usarlo en conjunción con un programa escrito por nosotros, aunque tiene el fallo de que sólo imprime textos definidos de antemano y no puede usar otros datos, por ejemplo un dato tecleado por el usuario.

PUNTUACION:
■ ■ ■ □ □

Pregunta: Y... si no es mucho pedir nos manden pegatinas. Posdata: muy amables. Posdata de la posdata: manden muchas pegatinas por favor. Gracias.

Raúl Hernández Galán y
Daniel Cuende Alonso

Respuesta: Está próxima la aparición de pegatinas de la revista ZX para los jóvenes lectores que así lo soliciten. Os mantendremos informados.

Pregunta: Con arreglo al anuncio insertado en su revista por la empresa Ventamatic, pedí por correo 20 cintas vírgenes que venían anunciadas al precio de 1.350 pesetas, y llamé por teléfono preguntando las condiciones del envío. Recibí por correo una fotocopia del anuncio en ZX, pero con los precios corregidos a bolígrafo. Ya no eran 1.350 sino 1.500 pesetas y los gastos de envío subían de 400 a 500 pesetas. Entiendo que cuando se publica un anuncio en una revista, las condiciones deben mantenerse, al menos hasta que aparezca el siguiente número. ¿Qué pueden hacer ustedes?

Miguel Villegas
Zaragoza

Respuesta: Sólo podemos publicar su carta. Una vez más reiteramos que la revista no asume responsabilidad por los anuncios publicitarios que se publican ni sobre las ofertas que ellos puedan contener.

Pregunta:... Por lo demás encuentro la revista altamente interesante, aunque quizás algo demasiado pródiga en programas de juegos. En mi modesta opinión, los microcomputadores personales tienen mayores y mejores poderes en otros campos, pese además, del difícil gobierno de los juegos a través del teclado, que en consecuencia sólo sufre suciedad y deterioro. ¿Publicarán algún montaje para *joysticks* adecuado para el Spectrum, para liberar un poco al teclado?

Otra sugerencia más sería la publicación de un índice completo del contenido de la revista y otro anual resumen de los mensuales. Ello facilitaría enormemente la localización de artículos, rutinas, trucos, programas, etcétera, que se pudieran necesitar en algún momento dado.

Juan B. Guillén Serra
Barcelona

Respuesta: Su primera observación ya ha sido respondida a otros lectores que comparten la misma opinión. Tratamos de lograr un equilibrio entre juegos y programas de aplicación, pero debe usted tener en cuenta que la inmensa mayoría de los usuarios del Spectrum son novatos en la informática, para quienes tales juegos son un eficaz modo de avanzar.

Acerca de la alternativa teclado/*joystick*, diremos que son pocos los Spectrum que se venden en nuestro país con este último dispositivo. Por razones de precio, entre otras, la mayoría de los usuarios se las arregla con el teclado. Tomaremos en cuenta su petición de un montaje.

Y, por último, estamos estudiando la conveniencia de publicar un índice sistemático de todo lo aparecido en ZX. Todavía no hemos decidido si será semestral o anual.

Pregunta: Les escribo, agradeciendo su oportuna aparición, ahora que el Spectrum está en gran auge. Lo bueno que veo a su revista es que se va a dedicar sólo al ordenador ZX, con lo que aumentan las posibilidades para estos usuarios. Aprovecho esta carta para hacerles unas preguntas y sugerencias, para ver si pueden responderme:

1. Me gustaría que repasaran el programa El Frantasma, del primer número, pues yo lo he repasado muchas veces y cuando doy al RUN me sale otra vez (también probé dar al 1 ó 0 antes del RUN, y lógicamente no salió nada), y lo único que hace es llenar la pantalla de cinco líneas numeradas, con las palabras "10 STOP gráficos", y nada más. ¿Qué es lo que puede pasar?
2. Bien a unos cuantos artículos EXHAUSTIVOS sobre la utilización del código máquina en el Spectrum.
3. Artículos mejor desarrollados y planteados sobre el BASIC en general.
4. ¿Va a aparecer algún otro modelo de ZX? ¿Cuándo? ¿Desplazará al Spectrum, como éste ha desplazado al ZX80?
5. ¿Se pueden usar compiladores para el Spectrum para trabajar en Fortran, Pascal y/o ensamblador? ¿Dónde se pueden conseguir? ¿Cuánto cuestan? ¿Cómo funcionan?
6. Una sugerencia: programas del Spectrum para el ZX81 es difícil de adaptar, pero no al revés. ¿Sería mucho pedir si en los números sucesivos, en los programas para ZX81, añadieran unas palabras para ver qué modificaciones hay que hacer para cargarlos en el Spectrum?
7. Que sigan las secciones de periféricos, montajes, ideas y lectores. ¿Cuándo hablan de impresoras?
8. Una sección con críticas y comentarios, además de precios de todas las cintas de juego, aplicaciones y demás que vayan apareciendo, hayan aparecido o no en el mercado. Que no haya que comprarlas a ciegas.
9. Sección de libros y revistas que hable sobre programas para ZX o BASIC en general. Con sus críticas y comentarios, ¿Posibilidad de precios especiales?
10. ¿Podrían darnos una rutina para abrir cintas ya programadas?
11. Sección de anuncios para los usuarios para venta, compra o cambio.

Antonio del Campo
Madrid

Respuesta: Vamos a intentar responder a sus múltiples preguntas, en el mismo orden en que las formula usted.

1. Como en la instrucción 15 hay un *input*, lo que hay que hacer es introducir un cero o un uno y pulsar NEW LINE, en vez de RUN. Después puede jugar con las teclas de movimiento (5, 6, 7, 8) y con la P para limpiar la pantalla.
2. Con el número 3 se ha iniciado una serie de artículos sobre el código, máquina, a un nivel que intentamos sea comprensible para el mayor número de lectores.
3. El BASIC, en general, quizá no sea del interés de la mayoría. Para ello existen otras publicaciones. Nuestro interés está en el BASIC de los ordenadores de Sinclair. No obstante, tal vez sea necesario abordar el tema en la página de principiantes que estamos preparando.
4. Efectivamente, ya se conoce el nuevo ordenador de Sinclair, al que dedicamos la *cover story* de este número. Como puede usted apreciar por la lectura del artículo, es algo muy distinto al Spectrum y, por tanto, no hay parangón posible entre este caso y la sustitución de hecho que vino a protagonizar el Spectrum respecto de sus antecesores. Cabe, sin

embargo, la posibilidad de que ciertos usuarios interesados en determinadas aplicaciones prefieran comprarse un QL en lugar de un Spectrum que les quedaría chico. Pero, en cualquier caso, son dos ramas diferentes de la familia Sinclair.

5. Los compiladores disponibles en España son los de código máquina, Forth y ensamblador - desensamblador. Sobre Fortran no hay nada, y sobre Pascal, sólo tenemos noticias de que se comercializa en Inglaterra.

6. En efecto, no resulta muy difícil. En líneas generales, sólo hay que disponer del mapa de memoria de uno y otro para cambiar los PEEK y POKE y alguna que otra instrucción. Lo que ocurre es que un programa para ZX81 generalmente es bastante pobre (especialmente si ha sido escrito para 1 K), y no merece la pena transcribirlo al Spectrum.

7. Lógicamente, no pretendemos ofrecer sólo juegos. A estas secciones y a otras que iremos incorporando próximamente les espera una larga vida, siempre que los lectores las consideren interesantes. Respecto a las impresoras: pronto nos ocuparemos de ellas.

8. Como habrá visto, antes de recibir su carta ya estábamos trabajando en esa sección, que ha comenzado a rodar en el número 3 de la revista.

9. En el número anterior ya comentamos un libro; en éste va otro, y pretendemos seguir en la misma línea. Como en el caso de los juegos, han comenzado a aparecer muchos libros, no todos de la calidad deseable.

10. No hay rutina para abrir cintas. Los fabricantes protegen sus intereses. Para ello se suele programar en listados fuente y se venden en código objeto, con lo que resulta imposible acceder al listado.

11. La sección Tablero de Anuncios, a la que convocábamos en el número 3, comienza su andadura en el presente ejemplar (y parece que con buen pie).

Esperamos haber aclarado todas sus inquietudes. No vacile en escribir otra vez si le queda algo pendiente.

Pregunta: Desearía me diesen una información más amplia sobre el ordenador T/S 2068, conocido como el Spectrum americano, aparecido en su revista número 2. En concreto, me gustaría saber si es posible el uso de dicho ordenador con un sistema de televisión español. En caso de no ser posible su uso, estaría muy agradecido si me dijesen si existe algún sistema de acople o solución posible. Les agradecería también me contestasen con la mayor brevedad posible, ya que se me presenta un viaje a Estados Unidos y desearía saber si vale la pena traerlo.

José Luis Corbi
Madrid

Respuesta: Como usted sabe, el sistema de televisión que se usa en Estados Unidos es de 60 semiimágenes por segundo y 525 líneas. El sistema de color es el llamado NTSC. Por el contrario, en España utilizamos 50 semiimágenes por segundo y 625 líneas, dentro del sistema PAL de color. Además, la ULA *(Uncommited Logic Array)*, descrito en el análisis del *hardware* que publicamos en el número 3, es distinto. En resumen, no es posible su utilización con el sistema de televisión español y no existe ningún sistema de acople. El interés de haber publicado el artículo sobre el Timex Sinclair 2068 era más bien de curiosidad y anticipo de eventuales (muy eventuales) modificaciones al Spectrum que se comercializa en Europa. Lamentamos haber creado falsas expectativas.

Pregunta: Desearía recibir más información acerca de las condiciones que deben satisfacer los programas que se os envíen, interesándome en especial por la posibilidad de que los programas sean científicos; enviar en un *cassette* varios programas, así como el destino del mismo una vez utilizado. En particular, si es posible la devolución del *cassette* para un posterior reenvío; características de la memoria y del listado; criterio de selección de programas, así como las condicines de pago del mismo; si será todo programa enviado recompensado económicamente; el destino de los no publicados, etc.

Sergi Sugrañes i Palet
Barcelona

Respuesta: No hay ningún inconveniente en que los programas sean científicos ni con que se puedan enviar en un *cassette*. Todo lo que recibimos, incluido el *cassette*, queda en nuestro archivo.

Los programas han de ser para el ZX81, de 1 o 16 K y/o Spectrum, de 16 o 48K, o cualquier otra configuración de memoria. No importa, pues, el tamaño de la memoria para el que estén preparados, a condición de que se haga notar el dato.

El criterio de selección es doble: la originalidad y la simplicidad en el método de progrmación.

El pago, si su programa resulta elegido, se realiza por talón nominativo.

Las únicas condiciones que se requieren son:

— Mandarnos un *cassette* con el programa, el listado y un texto explicativo. Es preferible, pero no imprescindible, que esté grabado dos veces. El listado puede estar hecho en impresora o a mano, y el texto explicativo ha de indicar las instrucciones de manejo del programa: cómo se carga, qué hay que hacer...

— El programa ha de ser original. No valen copias de programas ya publicados previamente en ninguna revista nacional y extranjera (y tenemos una buena colección.

Cada mes se seleccionan varios programas, cuyo número será variable, dependiendo del material que recibamos y de la calidad del mismo. Al final de cada mes se sortea un ZX Microdrive entre todos los programas recibidos, independientemente de que resulten publicados o no.

Y eso es todo. Con la colaboración de los lectores, estamos logrando una revista participativa.

Pregunta: En general, la revista lleva un buen camino, pero se le nota la inexperiencia. Sobre el número 2, quisiera hacerles algunos comentarios, que espero sean de utilidad.

El artículo dedicado a los *joysticks* es bueno y claro, en líneas generales, pero olvida mencionar que para la compra del *Interface* existe también otro tema a considerar, como es la oferta de *software* para dicho *interface*, puesto que no todos los programas funcionan con el mismo *juystick* ni aportan la opción para éstos. También cabría mencionar los más usados (no sólo los españoles, léase Indescomp), sino también los que más se encuentran como opción en los juegos: AGF y el rey KEMPSTON, que, aunque duela, por ser

tecnología extranjera, es el más popular.

Respecto al número 2 me gustaría aclarar otras dos cosas:

— Sobre la sustitución del ON X GOTO, el sistema más versátil y claro es GOTO 100—(X*10), para altar (por ejemplo) a las líneas 110, 120, etc., en función del valor de x. El hecho de permtir expresiones tras los comandos de salto es la razón por la cual no incorpora el Spectrum la instrucción ON X GOTO.

— Sobre el intercambio entre usuarios del ZX Spectrum y del ZX81 existe un interesante artículo en la revista *Your Computer* (diciembre de 1983), pág. 178, sobre cómo realizarlo vía audio (con el *cassette)*. Puedo proporcionar fotocopias.

En cuanto al apartado de *software,* en mi opinión, después de haber hecho una buena entrada, debería dirigirse a probar y explicar "claramente" los programas existentes en el mercado, principalmente juegos, pero no exclusivamente. En este sentido, y en todos los demás, podéis contar con mi colaboración (poseo más de 50 títulos del mercado español e inglés). Además, creo interesante para muchos la publicación de artículos sobre las curiosidades y peculiaridades no detalladas en el Manual (Stream, pokes, etc.).

Vicente Mendoz
Barcelona

Respuesta: Gracias por sus sugerencias e ideas. Tenemos a mano el ejemplar de *Your Computer,* que usted menciona, pero no podemos reproducir el artículo porque no disponemos del copyright correspondiente. Ya encontraremos la forma de tratar el tema del intercambio entre usuarios del ZX Spectrum y del ZX81.

En cuanto a la instrucción GOTO 100 (X*10), efectivamente es una buena sustitución de la instrucción ON X GOTO, mejor la que proponíamos con el IF-THEN.

Pregunta: Me dirijo a vosotros para rogaros que en los próximos números de la revista, a poder ser, publicáseis más programas ilustrativos, que no tanto juego.

D. Vázquez
Madrid

Respuesta: Pensamos como usted. No todo han de ser juegos. Sin embargo, hay muchos lectores que nos reclaman más juegos. Intentaremos combinar ambos gustos e incorporar pequeños programas ilustrativos.

Pregunta: Tengo un ZX81 y, a pesar de que para él no dais muchos programas, me divierto adaptando algunos del Spectrum y corrigiendo otros. Aquí os mando dos sugerencias:

1. Golf para ZX81 de 1 K. He modificado el que pusísteis para el Spectrum de la siguiente forma:

) El programa empezaría a partir de la línea 20.

b) En las líneas 30 y 40 se eliminan las sentencias INK.

c) Supongo que los lectores sabrán adaptar los INPUT y PRINT de las líneas 50 a 80.

d) Se eliminan las líneas 140 y 160.

e) En la línea 150 PLOT quedará: 150 PLOT (4* A *J)/4, (4*C18)/4.

f) 203 obviamente se convertirá en 203 GOTO 20.

g) En la línea 220 se eliminará la sentencia BEEP.

A propósito: las líneas 230-245 están repetidas.

2. Laser. Combate a muerte. Propongo la introducción de 105 IF INKEY $="" THEN GOTO 150. Con lo cual las naves enemigas se mueven sin que nos movamos nosotros. ¿Qué se gana con esto? Más emoción, ya que al moverse, aunque estemos quietos, nos obliga a tener más reflejos. Por último, quisiera hacerlos una consulta: ¿Con mi ZX81 puedo definir caracteres?

Miguel Ortiz Lombardía
Barcelona

Respuesta: Gracias por su colaboración y por detectar un error en el programa Golf en casa, que publicaremos en el número 1. Como bien dice usted, las instrucciones 230 a 245 aparecieron duplicadas y deben eliminarse. Respecto a su pregunta final: en el ZX81 los gráficos están en ROM, por lo que no se puede acceder a ellos, como en el Spectrum, que los lleva en RAM.

Pregunta: Quisiera que me informaran si existe la posibilidad de conectar mi Spectrum de 48 K a un *modem* (accesorio que permite comunicarse con otros ordenadores a través del teléfono) o si la única manera de comunicar con otro ordenador es mediante la ZX Net.

Enrique Marco
San Sebastián

Respuesta: Suponemos que está usted enterado que el prometido *modem* de Sinclair todavía no ha salido al mercado británico, de modo que en España "va para largo". No obstante, sí es posible la utilización del Spectrum a través de la ZX Net, ya que permite la conexión RS-232, que es la que necesita un *modem* de otras marcas. Pero ya en nuestro número anterior hacíamos referencia a los problemas surgidos con el *interface* del Spectrum de Sinclair, con el que es posible mandar información sin mayor problema, pero la recepción dista de ser perfecta.

Pregunta: Les envío la tarjeta de suscripción y, como el regalo que hacen ustedes con la suscripción está relacionada con el ZX Spectrum, les pediría que me mandaran otra publicación similar, pero para el ZX81, que es el que yo tengo.

Aprovecho la presente carta para decirles que he metido varios programas cortos de los números 1 y 2 de **ZX** y los veo un poco malos o mal explicados, pues algunos no hacen lo que parece, o dice, que tiene que hacer.

Para acabar, quería decirles que soy estudiante de cuarto curso de empresariales. A ver si tenéis algún programa relacionado con estos estudios.

José Luis García Ruiz
Cádiz

Respuesta: Respecto al libro, lamentamos decirle que no podemos cambiar por el momento la oferta de suscripción. Hay otras ofertas en vista, pero siempre con libros sobre el Spectrum.

En cuanto a los programas para el ZX81, procuramos ofrecerlos para las dos versiones (1 K y 16K). Lógicamente, los dedicados a la versión más baja son forzosamente cortos, ya que el ordenador no da para más. La explicación puede mejorarse, y tomamos buena nota de ello.

Respecto a la solicitud de "programas empresariales", dejamos abierta la puerta a que algunos otros lectores le

ayuden y nos ayuden. ¿Alguien se atreve a hacer programas de modelos de inversiones, gestión de *stocks*, programación lineal...?

Como adelanto, aquí le ofrecemos un pequeño programa para calcular el valor actual neto o valor capital de una inversión. ¿Satisfecho?

```
   Analisis de inversiones

Cash-flows                    3
Interes                       0.1%
Valor actual          -223.1405

Inversion nefasta!!
```

```
  10 BORDER 7: PAPER 6: INK 0: C
LS : CLEAR
  30 PRINT AT 2,3; BRIGHT 1;"Ana
lisis de inversiones": PAUSE 200
  40 INPUT "Numero de cash-flows
";n
  50 INPUT "Interes (tanto por c
ien)";k: LET k=k/100
  60 LET s=0
  70 FOR i=0 TO n-1
  80 INPUT "Cash-flow ",c
  90 LET s=s+c/(1+k)↑i
 100 NEXT i
 110 PRINT '"Cash-flows";TAB 29-
LEN (STR$ n);n
 120 PRINT "Interes";TAB 29-LEN
(STR$ i);k;"%"
 130 PRINT "Valor actual";TAB 29
-LEN (STR$ s);s
 140 GO TO 140+(((SGN s)+2)*10)
 150 PRINT FLASH 1;''"Inversion
nefasta!!"; FLASH 0: GO TO 990
 160 PRINT FLASH 1;''"Inversion
indiferente!!"; FLASH 0: GO TO 9
90
 170 PRINT FLASH 1;''"Inversion
interesante!!"; FLASH 0
 990 STOP
```

Pregunta: ¿Cómo puedo mezclar colores en mi Spectrum? También les pediría si tienen ocasión de editar algún artículo sobre el *modem* que anunciaron en su revista como de próxima aparición.

Emilio Porcar de Sousa
Cádiz

Respuesta: Si se refiere a la posibilidad de mezclar colores en un mismo punto, pensando quizá en obtener nuevos colores, ello no es posible. Lo que sí es posible, aunque suponemos que ya lo sabrá, es combinar colores en una línea, siempre que se varíe la instrucción INK antes de cada impresión. Respecto al *modem*, sólo podemos decirle que Sinclair Research viene hablando de él como "de próxima aparición". Suponemos que, cuando finalmente salga, habrá un desfase entre el lanzamiento para el mercado británico y su comercialización efectiva en España. Entonces hablaremos de él.

Pregunta: Me dirijo a ustedes para pedirles información sobre el Sinclair Spectrum, concretamente sobre sus gráficos definibles. Quisiera saber si trae algunos de estos gráficos de fábrica y, si los trajera, les ruego me digan cómo se obtienen.

Luis Constenla López
La Coruña

Pregunta: He confeccionado el programa que aparece en el número 2 de la revista bajo el título Bazooka. ¿Por qué en la línea 40, la nota gráfica A no aparece al instante?

Paco Cleandre
Barcelona

Respuesta: Como veíamos en los números 1 y 2 de esta revista, el tema de la generación de gráficos definibles es uno de los puntos "más oscuros" del manual del Spectrum.

Algunos gráficos sí "vienen de fábrica". Son los que están incorporados en el Spectrum, y los que se forman con ellos reciben el nombre de baja resolución. Son los que están ubicados en las teclas numéricas. Pero, al margen de ellos, el Spectrum tiene ya la posibilidad de incorporar gráficos definibles por el usuario, los llamados de alta resolución. Para esto existen unas localizaciones de memoria donde se pueden guardar. A dichas posiciones se accede cambiando al modo gráfico e indicando en qué posición.

Por ello, al margen de los juegos, verán ustedes Graphic A, Graphic B..., quiere esto decir que se accede a la información gráfica contenida en dichas teclas. Inicialmente, no habrá nada o, mejor dicho, estará la misma letra (Graphic A produce una A). Una vez que se ejecuta un programa que contiene generación de caracteres gráficos, las sentencias del programa que hacen referencia a tales caracteres se verán modificadas automáticamente (ver ZX números 1 y 2).

Por ello, no aparecía la nota gráfica A al instante (respondemos aquí específicamente al lector Clandre). Al no haberse ejecutado el programa (RUN y ENTER), no se ha generado el carácter gráfico. Sólo cuando el programa se ejecute y encuentre las instrucciones POKE necesarias aparecerá instantáneamente. Haga RUN y mire el listado. ¿Verdad que ahora sí están? Pulse Graphic y la A y verá cómo sale al instante.

Pregunta: ¿Cómo puedo meter un programa en memoria continua en mi Spectrum? ¿A qué equivale la función RIGHT en mi Spectrum? ¿Podría aumentarse la cantidad de gráficos definibles por el usuario en mi Spectrum?

Antonio Francisco
de Sousa
Cádiz

Respuesta: Para contestar a la primera pregunta necesitamos que nos explique qué entiende por memoria continua. Respecto a la función RIGHT, lo primero que ha de saber es que siempre va acompañada del signo dólar, es decir, RIGH$. Viene completada con la LEFT$ y MID$. El Spectrum las incorpora a través de la instrucción TO. Veamos cómo:

LEFT$: Devuelve una subcadena formada por los N primeros caracteres de otra. L$=LEFT$ (A$,N) asigna a L$ los N primeros caracteres de A$. En el Spectrum L$=A$ (1 TO N).

RIGHT$: Devuelve una subcadena formada por las N últimos caracteres de otra. L$=RIGHT$ (A$,N) asigna a L$ los N últimos caracteres de A$. En el Spectrum

LECTORES

L$=A$ (LEN A$-(N-1) TO LEN A$)

MID$: Devuelve una subcadena empezando por el carácter A y con una longitud L. L$=MID$ (A$,A,L) asigna a L$ L caracteres del String A$, empezando por el que se encuentra en la posición A. En el Spectrum L$=A$ (A TO B) asigna a L$ los caracteres de A$ entre A y B.

Respecto a los gráficos definibles..., hay un límite dado por la cantidad de memoria, sean 16 o 48 K. Si el ordenador no acepta más, no hay más remedio que borrar y meter nuevos.

Pregunta: En el interior de la portada de vuestro número 2 se ofrece un "folletto ilustrativo a todo color" sobre el ordenador Spectrum. He llamado a la empresa Investrónica para solicitarlo, pero la señorita que me atendió dijo no saber nada del asunto. ¿Por qué publican ustedes falsas promesas?

Jesús Martínez Vázquez
Valencia

Respuesta: Ignoramos las circunstancias de su peripecia con la telefonista de Investrónica. No le será difícil comprender que no somos responsables por los anuncios publicitarios que se publican en nuestra revista.

Su queja nos da una buena ocasión para reiterar nuestra absoluta independencia respecto de Sinclair Research, de sus representantes en España o de cualquier otra firma que comercializa *hardware* o *software*. Se ha dado el caso de algunos lectores que nos han hecho llegar por correo la tarjeta de garantía de los Spectrum que acababan de comprar (naturalmente, las giramos inmediatamente a Investrónica).

De todos modos, para que no quede usted insatisfecho, le hacemos llegar por correo un "folleto ilustrativo a todo color" que extraemos de nuestro archivo.

Pregunta: El motivo de nuestra carta es presentar nuestras quejas ante la falta de programas para el ZX81.

(Francisco García.—Pontevedra. Pablo Rodríguez.—Madrid. Antonio Delgado.—Madrid. Martín Saguer.—Las Palmas. Miguel Gray.—Alicante. Y otros).

Respuesta: El ZX81 ha sido relegado un poco a segundo término en el mercado de ordenadores. No sólo en cuanto a programación, sino también en cuanto a avances técnicos. Por eso el Spectrum es algo así como nuestra "Bandera" y todo lo que actualmente hay en el mercado de Sinclair está orientado en un 90 % a mejorarlo.

Prometemos no olvidarnos. Estamos seleccionando material sobre el ZX81.



SUSCRIBASE POR TELEFONO

* más fácil,
* más cómodo,
* más rápido

(91) 4572617

7 días por semana, 24 horas a su servicio

SUSCRIBASE A CIRCUITO IMPRESO

sinclair ZX Spectrum

- 16K: 34.950 ptas.
- 48K: 43.950 ptas.



■ EL PRECIO INCLUYE: ALIMENTADOR, CABLES PARA CASSETTE NORMAL Y TV (COLOR O B/N), CASSETTE DE DEMOSTRACIÓN, MANUALES EN INGLÉS, MANUAL EN CASTELLANO Y CASSETTE DE PROGRAMAS.

■ MICROPROCESADOR Z80A ■ 8 COLORES ■ 2 INTENSIDADES ■ SONIDO POR ALTAVOZ INTERNO ■ 40 TECLAS MÓVILES CON AUTO-REPETICIÓN Y SONIDO ■ MAYÚSCULAS, MINÚSCULAS, CARACTERES GRÁFICOS, INVERSOS Y DEFINIBLES ■ CÓDIGO ASCII ■ PANTALLA DE 24×32 CARACTERES ■ GRÁFICOS DE ALTA RESOLUCIÓN (256×192 PUNTOS) ■ BASIC SINCLAIR AMPLIADO EN 16K ROM ■ ALMACENAMIENTO DE DATOS Y PROGRAMAS EN CASSETTE (1.500 BAUDIOS) ■ CONECTOR DE EXPANSIONES.

- KIT AMPLIACIÓN A 48K RAM: 7.950 ptas.
- ADAPTADOR MEMORIAS 16K ZX81: 2.250 ptas.
- AMPLIFICADOR DE SONIDO + RESET: 4.990 ptas.
- IMPRESORA ZX: 13.950 ptas.
- CAJA 5 ROLLOS PAPEL: 2.625 ptas.
- INTERFACE CENTRONICS + CABLE: 11.450 ptas.
- INTERFACE PALANCA JUEGOS: 3.550 ptas.
- PALANCA JUEGOS "QUICK-SHOT": 2.900 ptas.
- CONECTOR HEMBRA: 990 ptas.

PRONTO DISPONIBLE: ■ ZX INTERFACE 1 ■ ZX INTERFACE 2 ■ ZX MICRO-DRIVE ■ MODEM ■ TRAZADOR DIGITAL ■ TECLADO PROFESIONAL ■ INTERFACE PROGRAMABLE PALANCA JUEGOS ■ PALANCA JUEGOS ALTA SENSIBILIDAD ■ FLOPPY DISK

Superprogramas ZX-Spectrum

EN ESPAÑOL

VIDEO-JUEGOS 16K/48K
- SUPER-COMECOCOS: 1.190,–
- CYBOTRON: 1.190,–
- GULPMAN: 1.390,–
- PHEENIX: 1.390,–
- WRECKAGE (DESTRUCCIÓN): 1.390,–
- LA RANA + Z-MAN: 1.690,–
- ANDROIDE UNO: 1.390,–
- BEDLAM: 1.390,–
- GALAXIANS + SPYNADS: 1.490,–
- CIENPIÉS + STORM-FIGHTERS: 1.490,–
- ARMAGGEDON + AMENAZA: 1.690,–

VIDEO-JUEGOS 48K
- EL DETECTIVE: 1.390,–
- TUNELES MARCIANOS: 1.390,–
- ESCALADOR LOCO: 1.390,–
- GRAND PRIX: 1.490,–
- STOMPING STAN: 1.390,–
- KILLER KONG: 1.390,–
- BARMY BURGER'S: 1.390,–
- MINA MALDITA: 1.390,–
- AUTOSTOPISTA GALÁCTICO: 1.390,–
- ULTIMO AMANECER EN LATTICA: 1.390,–

AVENTURAS (48K)
- THE QUILL (DISEÑO DE AVENTURAS) + EJEMPLO: 2.490,–

JUEGOS INTELIGENTES 48K
- AJEDREZ 2002: 2.490,–
- DALLAS: 1.390,–

JUEGOS INTELIGENTES 16K
- GUERRA DE BARCOS: 1.390,–

EDUCATIVOS 16K
- TUTOR MORSE: 1.390,–

GESTIÓN 48K
- MASTERFILE (BASE DE DATOS): 2.990,–
- CONTABILIDAD PERSONAL: 2.500,–
- SIMPLEX (PROGRAMACIÓN LINEAL): 2.500,–
- CONTEXT (PROCESADOR TEXTOS 64 COLUMNAS): 2.990,–

UTILIDADES 16K/48K
- ENSAMBL./DESENSAMBL.: 2.490,–
- COMPILADOR: 1.990,–
- 3D VISIÓN: 1.990,–
- VIDEO-DISPLAY: 1.990,–
- FORTH: 2.990,–
- SUPERGRAFICS + RUIDO: 1.990,–
- 64 PRINT: 1.490,–
- ADAPT. PROGRAMAS BASIC ZX81: 1.490,–

UTILIDADES 48K
- MASTER-DISEÑO: 1.990,–

AMPLIA SELECCIÓN DE LOS MEJORES PROGRAMAS DE IMPORTACIÓN EN INGLÉS: SINCLAIR, ULTIMATE, IMAGINE, BUG-BYTE, ETC. DISPONIBLES

ORIC-1

48.950 ptas.

sinclair ZX81

NUEVOS MICRO-PRECIOS AHORA SÓLO: 13.450 ptas.



■ EL PRECIO INCLUYE: ■ ALIMENTADOR, CABLES PARA CASSETTE NORMAL Y TV, MANUAL EN INGLÉS, MANUAL AMPLIADO EN CASTELLANO Y CASSETTE DEMOSTRACIÓN ■ IDEAL PARA INICIACIÓN A LA MICRO-INFORMÁTICA Y PROGRAMACIÓN, JUEGOS, GESTIÓN DOMÉSTICA Y PERSONAL, EDUCACIÓN, ETC. ■ 1K RAM ■ BASIC EN 8K ROM ■ MICROPROCESADOR Z80 A ■ ALMACENAMIENTO DE DATOS Y PROGRAMAS EN CASSETTE (250 BAUDIOS) ■ GRÁFICOS DE 44×64 PUNTOS ■ PANTALLA DE 24×32 CARACTERES ■ CONECTOR DE EXPANSIONES ■ 40 TECLAS SENSITIVAS.

SUPER OFERTA ESPECIAL: ZX81 + 16K RAM PACK SÓLO 17.950 ptas.

- CONECTOR MACHO: 300 ptas.
- INVERSOR DE VIDEO: 1.790 ptas.

MEMOTECH + ZX81 = LA ESTÉTICA DEL CONJUNTO

NO MÁS BORRADOS ACCIDENTALES DE MEMORIA



- MEMOPAK 16K (AMPLIABLE): 7.950 ptas.
- MEMOPAK 32K (AMPLIABLE): 14.950 ptas.
- MEMOPAK 64K (56K ÚTILES): 17.950 ptas.
- MEMOPAK INTERFACE RS232: 12.950 ptas.
- MEMOPAK INTERFACE CENTRONICS + CABLE PARA IMPRESORA NORMAL 80 COLUMNAS (MAYÚSCULAS Y MINÚSCULAS): 13.950 ptas.
- MEMOPAK ALTA RESOLUCIÓN GRÁFICA (192×256 PUNTOS) CON GRAN NÚMERO DE INSTRUCCIONES GRÁFICAS INCORPORADAS: 11.950 ptas.
- TECLADO PROFESIONAL MEMOTECH CON BUFFER: 14.950 ptas.



MICRO PRECIOS

IMPORTADOR EXCLUSIVO PARA ESPAÑA

- MEMOPAK EPROM: ENSAMBLADOR Z80: 8.950 ptas.
- MEMOPAK EPROM: MEMOCALC (HOJA DE CÁLCULO): 8.950 ptas.
- MEMOPAK EPROM: MEMOTEXT (PROCESADO TEXTOS): 8.950 ptas.

Superprogramas ZX81

VIDEO JUEGOS
- SUPER COMECOCOS: 1.190,–
- SUPER GULP: 990,–
- FROGGER: 1.190,–
- ALUNIZAJE: 1.190,–
- BATALLA ESPACIAL 3D: 1.190,–
- ASTEROIDES: 990,–
- DANGER TRACK: 990,–
- SCRAMBLE: 990,–
- CRASHBOOT + COMECOCOS: 990,–
- SUPER DEFENDER: 990,–
- SUPER JUEGOS (9 DE 1K): 990,–
- CASSETTE UNO (11 DE 1K): 990,–
- CASSETTE 2 (9 DE 16K): 1.590,–
- EL ACORRALADO: 990,–

MÚSICA
- ORQUESTA: 990,–

JUEGOS INTELIGENTES
- ZX AJEDREZ II: 1.990,–
- GUERRA DE BARCOS: 990,–
- MISIÓN GALÁCTICA: 990,–

EDUCATIVOS
- GEOGRAFÍA ESPAÑA: 1.390,–

UTILIDADES
- SUPERGRAFICS: 1.490,–
- VIDEOGRAFIC: 1.890,–
- ESCAPARATES: 990,–
- COMPILADOR: 1.890,–
- ENSAMB./DESENSAMB.: 1.890,–
- RAPID SAVER: 1.490,–
- ALTA RESOLUCIÓN: 1.490,–

GESTIÓN
- BASE DE DATOS: 2.790,–
- S. CONTROL STOCKS: 2.790,–
- VISI-PLAN: 1.890,–

ACCESORIOS

- CAJA 15 CINTAS VÍRGENES C-15: 1.350 ptas.
- CAJA 15 CINTAS VÍRGENES C-30: 1.800 ptas.
- MONITOR FÓSFORO VERDE 12": 24.950 ptas.
- MONITOR FÓSFORO VERDE 9": 20.450 ptas.
- MONITOR COLOR RGB 14": 69.950 ptas.

SEIKOSHA IMPRESORAS GRÁFICAS

SIMPLEMENTE LA MEJOR RELACIÓN CALIDAD/PRECIO ■ INTERFACE CENTRONICS DE ORIGEN ■ IMPRESIÓN AGUJAS UNIHAMMER



49.900 ptas.

GP100 ■ 80 COLUMNAS ■ 30 CARACT/SEG ■ MAYÚSCULAS, MINÚSCULAS (CÓDIGO ASCII) ■ CARACTERES EXPANDIDOS ■ PAPEL 10"

GP250 ■ 50 CARACT./SEG. ■ INTERFACE RS232 INCORPORADO ■ CARACTERES DOBLE ALTO/DOBLE ANCHO ■ RESTO COMO GP100: 59.900 ptas.

GP700 ■ TODOS LOS COLORES ■ ARRASTRE FRICCIÓN/TRACCIÓN ■ PAPEL HASTA 10": 98.500 ptas.

LIBROS

- 20 SIMPLE ELECTRONIC PROJECTS FOR THE ZX81: 1.590,–
- THE ZX81 POCKET BOOK: 1.660,–
- 49 EXPLOSIVE GAMES FOR THE ZX81: 1.490,–
- MASTERING MACHINE CODE ON YOUR ZX81: 1.890,–
- GAMES ZX COMPUTERS PLAY (30 GAMES FOR ZX81 & SPECTRUM): 990,–
- 60 GAMES & APPLICATIONS FOR SPECTRUM: 1.490,–
- CREATING ARCADE GAMES ON SPECTRUM: 1.190,–
- SPECTRUM GRAPHICS: 1.890,–
- SPECTRUM PROGRAMMES: 1.890,–
- GUÍA PRINCIPIANTE NEWBRAIN (C/CASSETTE): 1.000,–
- NEWBRAIN TECHNICAL MANUAL: 6.000,–
- BOLETINES CLUB NACIONAL USUARIOS ZX 1982 (1-4): 1.200,–
- BOLETINES CLUB NACIONAL USUARIOS ZX 1983 (5-10): 2.500,–
- MANUAL AMPLIADO ZX81
- MANUAL AMPLIADO ZX-SPECTRUM
- MANUAL CÓDIGO MÁQUINA ZX81
- MANUAL CÓDIGO MÁQUINA ZX-SPECTRUM
- LIBROS PROGRAMAS ZX

EN PREPARACIÓN EN CASTELLANO

NewBrain 74.950 ptas.

- MÓDULO BATERÍAS: 18.950 ptas.

PROGRAMAS: 1.000 ptas. c/u

■ BASE DE DATOS ■ CONTABILIDAD PERSONAL ■ ENTRETENIMIENTOS I ■ ENTRETENIMIENTOS II ■ UTILIDADES I ■ UTILIDADES II ■ VOLPLOT ■ FUENTES

SORD M-5 54.950 ptas.

ENVÍENME: FECHA .....

.....

.....

.....

.....

ENVÍO GIRO/TALÓN CONFORMADO PTAS. .....

PARA ENVÍOS C/REEMBOLSO MANDAR 20% A CTA.

NOMBRE .....

APELLIDOS .....

DOMICILIO .....

POBLACIÓN ..... D.P. .....

PROVINCIA .....

■ TARJETA VISA/MASTERCARD N.º .....

CADUCA ____ ____ FIRMA

■ GIRO POSTAL N.º ..... FECHA .....

GASTOS ENVÍO: 400 PTAS. REEMBOLSO O TARJETA DE CRÉDITO 200 PTAS. CUALQUIER OTRA FORMA

Ventamatic micro-informática

Avda. de Rhode, 253 - Apartado 168

ROSAS (GERONA) - Tel. (972) 255616

EXPOSICIÓN, VENTA Y CURSOS DE BASIC Y CÓDIGO MAQUINA CON ZX81 EN BARCELONA:

C/. Rocafort, 241, entlo. (DILVIS)

ESPECIALISTAS EN VENTA POR CORREO - ENVÍOS INMEDIATOS A TODA ESPAÑA - TODO EN STOCK - 6 MESES GARANTÍA

LIBROS

Cómo programar su Spectrum
Antonio Bellido
Ed. Paraninfo
132 páginas
850 pesetas

Muchos de vosotros ya le conoceréis, ya que ha sido el obsequio que la revista ZX viene haciendo a sus nuevos suscriptores. Por ello, y porque puede ser de interés para otros que no lo tengan, en este número analizaremos "nuestro obsequio".

Comienza el libro diciéndonos algo con lo que estamos completamente en desacuerdo: "Es el caso que, para un inglés sin conocimientos de programación, es fácil la interpretación de un programa sencillo en BASIC, mientras que para uno de nosotros, sin saber inglés, sería imposible". Esto es doblemente falso: ni un inglés puede programar sin saber BASIC ni el desconocimiento del inglés imposibilita programar. Basta con saber qué instrucción hay que darle al ordenador en cada momento, si bien conociendo inglés se facilita la memorización de tales instrucciones.

Comenzando a ver este libro, varias cosas llaman la atención del lector. En primer lugar, destaca que no es una traducción, sino que es un libro *made in Spain*. Pero lo más curioso es la forma en que su autor nos cuenta las cosas. Antonio Bellido ha "personalizado" nuestro querido Spectrum y nos hace ver que es el propio ordenador el que nos va contando qué podemos hacer y cómo hemos de hacerlo.

Con la frase "Hola. Soy tu Spectrum", se inicia una larga charla, en la que nos cuenta las instrucciones para principiantes, instrucciones específicas y expresiones y operadores lógicos.

La primera parte se titula Instrucciones para Principiantes, y ciertamente es difícil lograr mayor claridad y sencillez en la explicación de cómo dar los primeros pasos con el BASIC y con el Spectrum. Pero sólo para eso, pues si usted ya tiene conocimientos del BASIC le prometemos que se aburrirá. Es la parte principal del libro, con lo que ya se ve que está destinado a los que no tienen ningún conocimiento del BASIC y desean que se lo cuenten con mucha tranquilidad.



La parte de instrucciones específicas trata de estudiar algunas instrucciones BASIC especiales del Spectrum, tales como el MERGE, PAPER, BORDER, etc., y otras más generales, como el LOAD, READ, DEF FN.

La tercera y última parte se dedica a las expresiones y operadores lógicos, describiéndose el uso del AND, OR y NOT.

Si en la parte primera el Spectrum nos habla con todo lujo de detalles y ejemplos, en las partes dos y tres deja de hablarnos, y la información que se nos ofrece es bastante concisa y sin apenas ejemplos.

Finalmente, acaba el libro con el "modelo de impreso" para desarrollar los programas. No está demasiado logrado, ya que el espacio reservado para la intrucción es muy pequeño, habida cuenta de que el Spectrum admite múltiples instrucciones por línea. Asimismo, el espacio reservado a comentarios resulta excesivo. No obstante, ello obliga a sistematizar los programas, y esto siempre es positivo.

En resumen, un libro aconsejado sólo para los "muy principiantes".



Cómo usar los colores y los gráficos en el Spectrum
Antonio Bellido
Ed. Paraninfo
96 páginas
850 páginas

Por pura casualidad, le ha tocado a Antonio Bellido el ver dos de sus libros comentados en este mismo número de ZX. Pero no es nada extraño, si bien se mira, porque Bellido parece haber escogido como tema de su actividad bibliográfica la divulgación de las peculiaridades del Spectrum. Seguro que en lo sucesivo nuestros lectores volverán a encontrar su nombre en estas páginas, si cumple su promesa de lanzar dos títulos ya anunciados: "Del Microdrive al Diskdrive" y "Los trucos del Spectrum".

En los primeros libros publicados por este autor, su preocupación principal fue popularizar el BASIC. El título que ahora nos toca comentar —y los que están por venir— será bien recibido por todos aquellos que han pasado la etapa de iniciación y desean sacarle más provecho a su Spectrum.

El título es por sí mismo bastante claro sobre las intenciones del texto: el tema principal son los gráficos. Pero al margen de ellos encontramos una buena descripción de la memoria del Spectrum a través del uso de las instrucciones PEEK y POKE, y de diversas direcciones del mapa de memoria.

La descripción de las posibilidades gráficas del Spectrum es francamente buena. Las instrucciones en baja y alta resolución son explicadas con gran lujo de detalles y ejemplos.

Con el nombre de "Procesos internos del Spectrum", Bellido nos introduce someramente en el lenguaje binario a fin de lograr una fácil comprensión de las instrucciones de acceso directo a memoria (PEEK y POKE). A continuación analiza las "variables del sistema" que, aunque no tengan mucha relación con los gráficos, resulta ser una información interesante, que nos permitirá, por ejemplo, hacer desaparecer el mensaje de SCROLL y hacer que éste se produzca automáticamente.

Finaliza el libro con el estudio de aplicaciones para el GRAFKIT, cuadernillo de 30 hojas que reproduce la pantalla del Spectrum en alta y baja resolución y que ayudará a todos aquellos que deseen una utilización "seria" de las posibilidades de graficación de su Spectrum.

Todos los programas que se desarrollan en el libro se incluyen en un *cassette* que puede adquirirse opcionalmente.

S. C.

LA REVISTA PARA LOS USUARIOS DE COMMODORE



YA ESTA A LA VENTA

EL NUMERO 1

Pídala en su quiosco

Han sido muchos los lectores que escribieron o llamaron por teléfono a la redacción de ZX para señalar errores en programas publicados en los 1 y 2 de la revista. Tras una revisión de nuestros listados, les dijimos en su momento que los programas estaban perfectamente bien. Pero, ante la insistencia, hemos verificado en algunos programas diferencias entre los listados originales y la versión que de ellos aparece impresa. Pedimos disculpas por estos fallos (que no son atribuibles al célebre "duende de las imprentas") y agradecemos la amabilidad de los lectores que nos hicieron notar el error y, al mismo tiempo, han ratificado su confianza en la revista. Quede claro, en cualquier caso, que estos errores han sido corregidos en la tercera reimpresión del número 1 y en la segunda del número 2, que son las que circulan actualmente.

En el número 1, hay dos programas que contienen errores:

- **GOLF EN CASA.** Deben suprimirse las instrucciones números 230, 235, 240 y 245.
- **GUERRA SUBMARINA.** Se habían omitido algunas instrucciones, por lo que ofrecemos de nuevo este buen programa, ahora con todas las instrucciones necesarias.

```
   9 GO SUB 6000
  10 INK 0: BORDER 0: PAPER 5: LET score=0: LET sux=15: LET state=0: LET sdc=0: LET pdc=0: LET ddc=0: LET torp=20: LET tpy=0: CLS
  15 PRINT "Instrucciones (s/n) ?"
  16 LET a$=INKEY$: IF a$="" THEN GO TO 16
  17 IF a$="n" THEN GO TO 310
  20 REM Submarino
  50 PAUSE 50
  70 PRINT AT 10,11;"Submarino";AT 12,14;"▄▄"
  80 PAUSE 50
 100 REM *************
 110 REM Instrucciones
 120 CLS
 130 PRINT TAB 11;"SUBMARINO"
 140 PRINT TAB 11;"▀▀▀▀▀▀▀▀▀"
 150 PRINT
 160 PRINT "El juego consiste en destruir    el barco enemigo evitando las     cargas de profundidad de los     barcos torpederos ▄▄ lo que      significa";: INK 2: FLASH 1: PRINT "█ MUERTE █": FLASH 0: INK 1
 165 PRINT "(5)Izquierda - (8)Derecha"
 170 PRINT "(Space)Fuego": INK 1
 190 PRINT "▄▄ ","1000 puntos"
 200 PRINT
 210 PRINT "▄▄▄ ","250 puntos"
 220 PRINT
 230 PRINT "▄▄▄ ","250 puntos"
 240 PRINT
 250 PRINT "▄▄▄ ","500 puntos"
 260 PRINT
 270 PRINT "▄▄▄▄ ","50 puntos"
 280 INK 0: PRINT
 290 PRINT "     Pulse cualquier tecla para comenzar"
 300 PAUSE 0
 310 CLS
 365 INK 1
 370 FOR a=10 TO 21
 390 PRINT AT a,0;"████████████████████████████████"
 410 NEXT a
 500 LET shx=25
 510 DATA "▄▄▄▄ ",50,"▄▄▄ ",250,"▄▄ ",1000,"▄▄▄ ",500,"▄▄▄ ",250
 520 FOR a=1 TO (5*RND)+1
 530 READ a$,val
 540 NEXT a
 545 LET step=.25+((INT (RND*6)+1)/8)
 550 RESTORE
 560 INK 5: PRINT AT 9,0;"███████████"
 600 PRINT AT 0,0;"Tantos:";score: PRINT AT 0,16;"Torpedos:";torp: IF torp<10 THEN PRINT AT 0,26;" "
 700 INK 5: PRINT AT 9,shx+5;"█"
 705 INK 1: PRINT AT 20,sux;"██"
 710 LET sux=sux+(INKEY$="8" AND sux<30)-(INKEY$="5" AND sux>0)
 715 PRINT OVER 1;AT 20,sux;"▄▄"
 720 PRINT INK 0;AT 9,shx;a$
 725 IF INKEY$=" " AND state=0 THEN LET state=1: LET torp=torp-1
 726 IF state=1 THEN GO SUB 1000
 727 IF state=0 THEN GO SUB 900
 728 IF torp<0 THEN GO TO 3000
 729 IF sdc=1 THEN GO SUB 2500
 730 IF sdc=0 AND RND<.4 THEN GO SUB 2000
 740 LET shx=shx-step
 745 IF shx<1 AND torp<1 THEN GO TO 3000
 750 IF shx<1 THEN GO TO 500
 760 GO TO 600
 900 LET post=sux
 905 LET tpy=18
 910 RETURN
1000 INK 1: PRINT AT tpy+1,post;"█"
1030 PRINT OVER 1;AT tpy,post;"λ"
1034 IF tpy>10 THEN GO TO 1040
1035 IF tpy=10 THEN LET state=0
1036 IF tpy=10 THEN PRINT AT tpy,post;"█"
1037 IF tpy=10 AND SCREEN$ (tpy-1,post)<>" " THEN LET score=score+val: PRINT OVER 1;AT tpy-1,post;"✱": PAUSE 50: FOR y=1 TO 8: BEEP .05,10*RND+20: NEXT y: PRINT AT tpy-1,0;"                                ": GO TO 500
1040 LET tpy=tpy-.5: BEEP .08,-10: RETURN
2000 REM ****************** Soltar cargas profundidad
2010 LET pdc=shx
2020 LET ddc=10
2030 LET sdc=1
2040 BEEP .1,-20
2050 RETURN
2500 INK 1: PRINT AT ddc-1,pdc;"█"
2510 PRINT OVER 1;AT ddc,pdc;"▪"
2520 IF ddc=20 THEN PRINT OVER 1;AT ddc,pdc;"✱": PAUSE 25: BEEP .25,-20: PRINT AT ddc,pdc;"█": LET sdc=0
2530 IF ddc=20 AND ABS (pdc-sux)<5 THEN PRINT AT 20,sux;"✱✱": BEEP .25,-30: PAUSE 25: PRINT AT 20,sux;"██": GO TO 4000
2540 LET ddc=ddc+1
2550 RETURN
3000 CLS
3100 PRINT "Fin. Su puntuacion es ";score
3200 GO TO 5000
4000 INK 2: CLS : PRINT "BOOOMM"
4010 INK 0: PRINT "Su puntuacion era ";score
5000 PRINT "Otra vez (s/n) ?"
5010 LET a$=INKEY$: IF a$="" THEN GO TO 5010
```

```
5020 IF a$="s" OR a$="S" THEN GO
 TO 10
5030 IF a$="n" OR a$="N" THEN ST
OP
5040 GO TO 5010
8000 RESTORE 9000
8010 FOR a=144 TO 164
8020 FOR b=0 TO 7
8030 READ c
8040 POKE (USR CHR$ a)+b,c
8050 NEXT b
8060 NEXT a
8070 RESTORE
8080 RETURN
9000 DATA 1,3,3,127,255,255,127,
0
9010 DATA 0,192,192,248,255,255,
248,0
9020 DATA 0,0,3,3,255,237,63,0
9030 DATA 8,60,60,253,255,255,55
,0
9040 DATA 0,0,0,192,255,255,254,
0
9050 DATA 0,0,0,24,255,127,63,31
9070 DATA 0,0,1,15,255,255,255,2
55
9080 DATA 1,255,249,255,255,255,
255,255
9090 DATA 0,3,243,255,255,255,25
5,255
9100 DATA 0,0,192,236,255,255,25
5,252
9110 DATA 0,0,0,3,255,127,63,0
9120 DATA 0,0,192,224,255,255,25
4,0
9130 DATA 0,0,1,3,255,127,63,0
9140 DATA 4,30,62,255,255,255,25
5,0
9150 DATA 0,0,192,224,255,255,25
4,0
9160 DATA 0,0,3,55,255,127,63,0
9170 DATA 4,28,30,254,255,255,25
5,0
9180 DATA 0,0,192,230,255,255,25
4,0
9190 DATA 0,0,120,120,120,0,0,0
9200 DATA 137,74,42,0,255,255,12
6,0
9210 DATA 24,24,24,24,24,24,36,6
6
```

También en el número 2 hemos localizado errores en tres programas:

• **DEFENSA DE LA CIUDAD.** Debe modificarse la instrucción

```
160 LET t$="00"( TO 2-LEN (STR$
t))+STR$ t
```

Sobra la línea 900

• **OVNIS.** Debe insertarse la instrucción

```
 445 IF y=20 AND y1=1 THEN BEEP
2,-20: GO TO 420
```

• **GALAXIA 2000.** El programa estaba preparado para su autoejecución en línea 670, ya que por efecto de la línea 690 se grababa con la orden de autoejecutarse partiendo de la línea 670. Por tanto, si introduce el programa, lo graba de acuerdo con la instrucción 690 y después lo carga de cinta funcionará correctamente. Si lo introduce directamente no le aparecerán los caracteres gráficos. Para evitar esto haga dos cosas. Introduzca la línea 40 GOSUB 670 y cambie la instrucción RUN de la línea 670 por RETURN. Verá como ahora funciona correctamente.

También en el número 3 hemos detectado un error.

• En el programa **JUEGO DE PALABRAS** (página 30), eliminar línea 659 e insertar:

```
4040 PRINT AT 20,20; INK 0;"El m
ovimiento debe ser entre"
4140 GO SUB 3000
5580 GO TO 5550
```

• **TRES EN RAYA** (pág. 33):

En la línea 6070 no se ve claro que hay que dejar 25 espacios. Eliminar 8004. Insertar:

```
8040 PLOT 104,135
```

• **CARRERA DE CABALLOS** (pág. 34). Introducir:

```
5030 PRINT AT 15,i;" h"
5035 PAUSE 2
```

• En **SOBRE EL ABISMO** (pág. 38).

```
3080 IF j>10 AND j<20 THEN GO TO
 3090
7560 PRINT AT 15,0;"          ";
 REM 10 espacios
7570 PRINT AT 16,0;"          ";
 REM 10 espacios
```

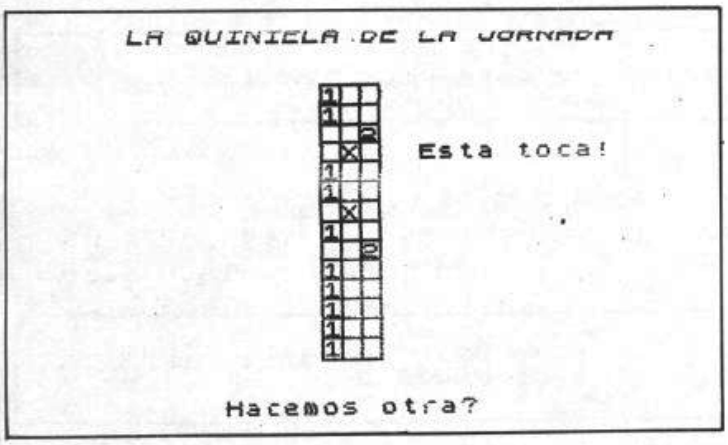
En las líneas 7560 y 7570, no han salido impresas comillas al final de los 10 espacios y los dos puntos.

Gracias de nuevo a quienes nos pusieron en la pista de estos fallos.

QUINIELAS

Con este sencillo programa y un poco de suerte podemos conseguir un buen "pellizco" jugando a las quinielas, si bien su autor confiesa que todavía no lo ha conseguido. Su funcionamiento es muy sencillo, hasta con poner en marcha el programa para que aparezca en la pantalla la quiniela de la fortuna, a continuación pregunta si se quiere hacer otra y si se dice que no el programa se autodestruye. Como puede observarse, el programa únicamente trabaja con valores aleatorios y sin datos sobre el pasado para realizar buenas predicciones, de otra forma no creemos que quisiese compartir su programa ¿verdad?

(16K-Spectrum)




```
   1 REM Antonio Verdugo Ltd.
   2 PRINT AT 1,3; FLASH 1;"LA Q
UINIELA DE LA JORNADA"
   4 PRINT : PRINT
   6 FOR n=1 TO 14
  10 LET a=INT (RND*7)+1
  15 BORDER a: PAUSE 7
  20 IF a=1 OR a=2 OR a=3 OR a=7
 THEN PRINT TAB 13;1
  30 IF a=4 OR a=5 THEN PRINT TA
B 14;"X"
  40 IF a=6 THEN PRINT TAB 15;2
  50 NEXT n
  60 IF a=7 THEN PRINT AT 7,18;"
Esta toca!"
  70 PLOT 104,144: DRAW 0,-113:
DRAW 24,0: DRAW 0,113: DRAW -24,
0
  80 PLOT 112,144: DRAW 0,-113
  90 PLOT 120,144: DRAW 0,-113
 100 FOR n=0 TO 96 STEP 8
 110 PLOT 104,136-n: DRAW 24,0
 120 NEXT n
 130 PRINT AT 20,8;"Hacemos otra
?"
 140 IF INKEY$="s" THEN RUN
 150 IF INKEY$<>"n" THEN GO TO 1
40
 160 PRINT AT 19,1;"Este program
a se autodestruira          en
  segundos"
 170 FOR n=1 TO 22: BEEP .1,10:
PRINT AT 20,12;22-n: PAUSE 40
 175 IF n>10 THEN PRINT AT 20,13
;" "
 180 NEXT n
 190 NEW
```

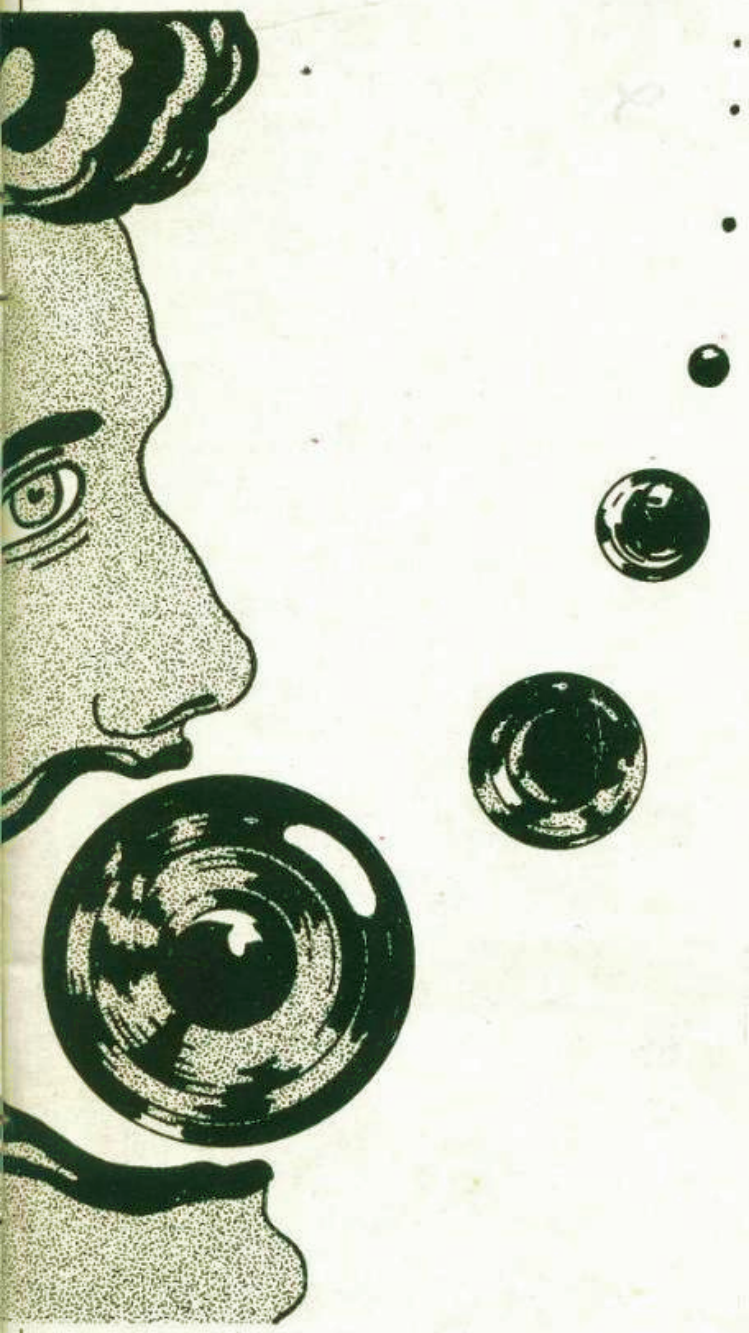


PROGRAMA GANADOR DE 5.000 PTAS

Antonio Verdugo de Osa, de Madrid, es el autor de esta quiniela que no da garantía alguna de premio. Salvo para Antonio, que se ganado 5.000 pesetas gracias a este programa.

TRAGABOLAS

En este juego habrá de luchar contra el tiempo haciendo alarde de su habilidad. Todo lo que tiene que hacer es guiar al "tragabolas" para que atrape el mayor número posible de bolas. Como las bolas caen desde la parte superior de la pantalla, todo lo que debe hacer es conducirle de izquierda a derecha para que no se le escape ninguna o las menos posibles; para lo cual deberá presionar la "o" para desplazarse a la izquierda, y la "p" para ir hacia la derecha. Así de sencillo y así



de original cuando lo vea en pantalla.

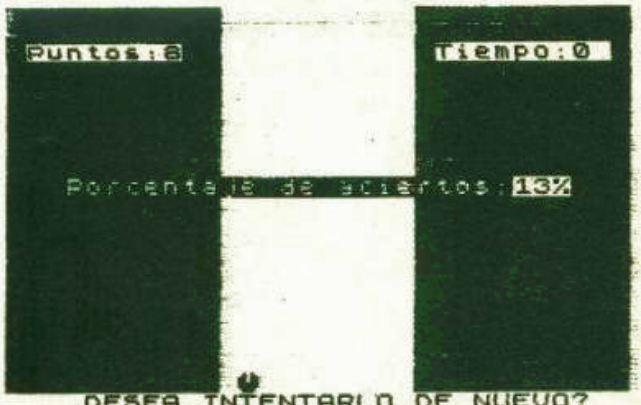
El gráfico del "tragabolas" está definido en la tecla "a", por lo que en las instrucciones 60 y 300 teclee GRAPHIC A.

(16K-Spectrum)

PROGRAMA GANADOR DE

5.000 PTAS

Con este simpático programa el lector Francisco Hernaz Costa, de Barcelona, se ha ganado 5.000 pesetas que recibirá por correo.




```
    DESEA INTENTARLO DE NUEVO?
   1 GO SUB 200
   2 BORDER 7: PAPER 7: INK 0: C
LS : FOR i=0 TO 20: PRINT INK 4;
AT i,0;"                    ";AT i,21;"
                    ": NEXT i
   6 LET p=0
   7 LET t=60
   8 LET s=13
   9 LET b=15
  10 LET a=INT (RND*10)+11
  15 LET t=t-1: IF t=-1 THEN GO
TO 101
  20 FOR i=1 TO 21
  40 PRINT AT i,a;"."
  45 PRINT AT i-1,a;" "
  50 IF INKEY$="o" THEN LET b=b-
1: LET s=b+1
  55 IF INKEY$="p" THEN LET b=b+
1: LET s=b-1
  56 IF b<=11 THEN LET b=11: LET
 s=12
  57 IF b>=20 THEN LET b=20: LET
 s=19
  60 PRINT ;AT 20,b;"Ψ"
  61 PRINT AT 20,s;" "
  65 IF i=20 AND a=b THEN LET p=
p+1: BEEP .2,-20: GO TO 80
  70 NEXT i
  80 PRINT AT i-1,a;" "
  85 PRINT INK 2;AT 3,1;"Puntos:
";p
  87 PRINT AT 3,30;" "
  90 PRINT INK 1;AT 3,22;"Tiempo
:";t
 100 GO TO 10
 101 FOR i=20 TO 1 STEP -2: BEEP
 .2,i: NEXT i
 110 LET w=INT ((100*p)/60)
 115 PRINT AT 10,3;"Porcentaje d
e aciertos:";w;"%"
 120 PRINT AT 21,4;"DESEA INTENT
ARLO DE NUEVO? "
 125 IF INKEY$="" THEN GO TO 125
 130 IF INKEY$<>"s" THEN STOP
 140 GO TO 2
 200 POKE USR "a"+0,BIN 00100100
 210 POKE USR "a"+1,BIN 01100110
 220 POKE USR "a"+2,BIN 11100111
 230 POKE USR "a"+3,BIN 11100111
 240 POKE USR "a"+4,BIN 11111111
 250 POKE USR "a"+5,BIN 11111111
 260 POKE USR "a"+6,BIN 01111110
 270 POKE USR "a"+7,BIN 00111100
 300 PAPER 0: BORDER 0: CLS : PR
INT INK 6;AT 8,12;"TRAGABOLAS";A
T 10,0;"Ud.(Ψ) debe,en un tiempo
 limite,tragarse el mayor numero
 de      bolitas posibles.";AT 17
,8;"CONTROL:  'O' Izquierda
              'P' Derecha";AT 21,
3;"Pulse ENTER para comenzar"
 310 IF INKEY$="" THEN RETURN
 320 GO TO 310
```

EN BUSCA DEL WC

PROGRAMA GANADOR DE 5.000 PTAS

Juan Manuel Prieto, residente en Madrid, nos ha enviado este En busca del WC que, si tuviéramos que elegir, probablemente mereciera la calificación de mejor programa entre los participantes del concurso de este mes. Se ha ganado, por tanto, muy cabalmente el premio de 5.000 pesetas.

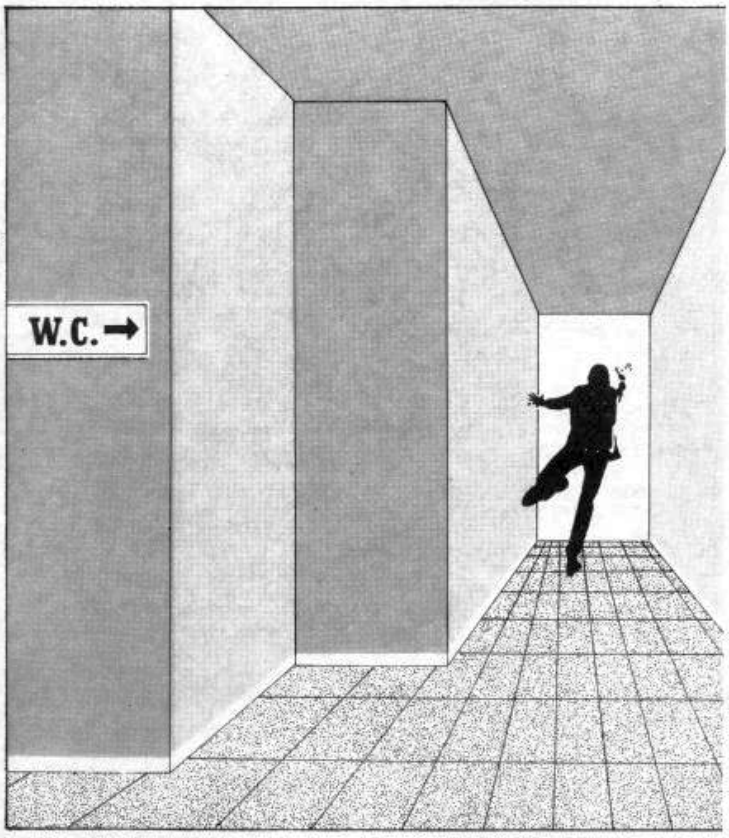
Bajo este título realmente cachondo el lector Juan Manuel Prieto nos ha hecho llegar un programa lleno de interés para quienes pretendan explorar las posibilidades del Spectrum. El autor define así el argumento de su juego:

"Los hangares de una ciudad del año 2020 han sido invadidos por robots extraterrestres. Estos impiden que los pacíficos robots de la ciudad evacúen las sustancias corrosivas producidas por sus circuitos electromecánicos. Por suerte, desde el ordenador central aún nos queda el control de los elevadores del hangar. Es decir, lo que usted maneja es el elevador, con lo que posibilitará, si es hábil, que el robot llegue a "su destino". Para ello ha de pulsar la 'm' para bajar y la 'i' para subir."

Utiliza siete gráficos definidos, pero usted sólo ha de hacer referencia a cinco por teclado:

Graphic ADG
Graphic J
Graphic K
Línea 70
Línea 1070
Línea 420, 3020, 3050, 3130, 3140, 5010, 5050

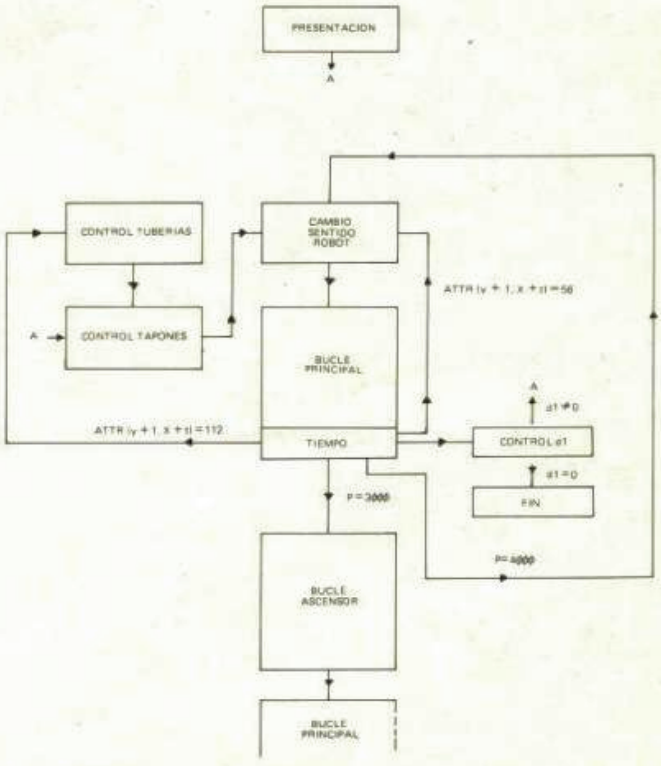
(16K-Spectrum)




```
  5 REM En busca del WC
  por Juan Prieto  (ZX 1984)
 10 BORDER 1: PAPER 1: INK 7: F
LASH 0: BRIGHT 0: OVER 0: POKE 2
3658,0: CLS
 20 GO SUB 9000
 30 REM Presentacion
 40 PRINT AT 2,9; PAPER 5; INK
0;"EN BUSCA DEL"
 50 PLOT 90,120: DRAW 10,-20: D
```

```
RAW 5,15: DRAW 5,-15: DRAW 10,20
: PLOT 145,120: DRAW 0,-20,58/45
*PI: CIRCLE 122,99,2: CIRCLE 152
,99,2
 60 FOR n=28 TO 1 STEP -3
 70 PRINT AT 15,n;"ÄÄ": NEXT n
 80 PRINT AT 18,5;"I_____
_____ARRIBA";AT 20,5;"M_____
_____ABAJO"
 90 FOR n=.001 TO .13 STEP .001
: BEEP .13-n/1.2,50*n: NEXT n
100 PAPER 4: INK 0: CLS : PAPER
 7
110 FOR n=0 TO 25: PRINT PAPER
2;AT RND*18+3,RND*31;" ": NEXT n
115 PRINT AT 0,0;
120 FOR n=0 TO 2: PRINT PAPER 5
;"
  ": NEXT n
130 FOR n=7 TO 11: PRINT PAPER
6; BRIGHT 1;AT n,12;" ";AT n+5,2
3;" ";AT n+5,8;" ";AT n,19;" ":
NEXT n
140 RESTORE 9100: FOR n=0 TO 4
150 READ a$,a,b
160 PRINT AT a,b;a$
170 NEXT n
180 INK 7: PLOT 16,128: DRAW 21
0,0: PLOT 16,40: DRAW 208,0: PLO
T 46,80: DRAW 201,0: PLOT 8,24:
DRAW 31,0: INK 0
190 FOR n=3 TO 20: PRINT AT n,5
;" ";AT n,16;" ";AT n,28;" ": NE
XT n
400 LET d1=3: PRINT AT 0,10; PA
PER 1; INK 7;d1
410 LET d2=0: PRINT AT 0,20; PA
PER 2; INK 7;d2
420 PRINT PAPER 8;AT 0,12;"$";A
T 0,22;"WC"
480 LET z=15
490 LET g=200
500 LET t=-1: LET y=19: LET x=1
: LET b=50
510 GO TO 5080
990 REM Bucle principal
1000 PRINT AT y,x;" ": LET x=x+t
1010 PRINT AT y,x;a$
1020 LET q=(INKEY$="m" AND z<20)
-(INKEY$="i" AND z>5)
1030 IF ATTR (y,x+t)<>56 THEN LE
T p=4000
1040 PRINT AT y,x;b$
```

```
1050 PRINT AT z,5;" ";AT z,16;"
";AT z,28;" "
1060 LET z=z+q
1070 PRINT INK 2;AT z,5;"■";AT z
,16;"■";AT z,28;"■"
1080 PRINT AT y,x;c$
1090 IF ATTR (y+1,x+t)>b THEN LE
T p=3000
1100 LET g=g-1: PRINT AT 0,1; BR
IGHT 1;g;" ": IF g=0 THEN GO TO
8500
1110 GO TO p
2990 REM Bucle ascensor
3000 IF ATTR (y+1,x+t)=56 THEN G
O TO 4010
3010 IF ATTR (y+1,x+t)=112 THEN
GO TO 5000
3020 PRINT AT y,x;" ": LET x=x+t
: PRINT AT y,x;a$
3030 BEEP .09,20: BEEP .01,40
3040 LET b=100: LET v=t: LET t=0
: LET p=3060
3050 LET a$="↑": LET b$="↑": LET
 c$="": GO TO 1010
3060 LET y=y+q: PRINT AT z-2,x;"
"
3070 IF ATTR (y,x-v)=56 THEN LET
 p=3100
3080 IF ATTR (y,x+v)=56 THEN LET
 p=3110
3090 GO TO 1010
3100 LET t=v: GO TO 3120
3110 LET t=-v
3120 BEEP .05,-15: IF RND>=.5 TH
EN GO TO 3140
3130 LET x$=CHR$ 16+CHR$ 0+" ":
LET y$=CHR$ 16+CHR$ 2+"↑": GO TO
 3150
3140 LET x$=CHR$ 16+CHR$ 2+"↑":
LET y$=CHR$ 16+CHR$ 0+"
3150 PRINT AT 12,10;x$;AT 17,21;
x$;AT 17,10;y$;AT 6,21;y$
3160 LET b=50: GO TO 4020
3990 REM Control robot
4000 IF ATTR (y,x+t)=31 THEN GO
TO 8000
4010 BEEP .01,50
4020 LET a$=CHR$ (145-t): LET b$
=CHR$ (148-t): LET c$=CHR$ (151-
t)
4030 LET t=-t: LET p=1000: GO TO
 1020
4990 REM Bucle tuberias
5000 PRINT AT y,x;" ": LET x=x+t
: LET t=-t
5010 PRINT AT y,x; PAPER 8; BRIG
HT 8;" "
5020 LET y=y+1
5030 BEEP .01,25
5040 IF ATTR (y,x)=56 THEN BEEP
.05,-20: GO TO 5070+e
5050 PRINT AT y,x; PAPER 8; BRIG
HT 8;"↑"
5060 GO TO 5010
5070 PRINT PAPER 4;AT 13,8;" ";A
T 7,19;" "; PAPER 6; BRIGHT 1;AT
 7,12;" ";AT 13,23;" ": LET e=10
: GO TO 4020
5080 PRINT PAPER 4;AT 7,12;" ";A
T 13,23;" "; PAPER 6; BRIGHT 1;A
T 13,8;" ";AT 7,19;" ": LET e=0:
 GO TO 4020
7990 REM Contador WC
8000 PRINT AT 4,29;" "
8010 LET d2=d2+1: PRINT AT 0,20;
 PAPER 2; INK 7;d2
8020 FOR n=0 TO 2: FOR m=10 TO 4
0 STEP 5: BEEP .02,m: NEXT m: NE
XT n
8030 IF d2>1 THEN LET g=150: GO
TO 500
8040 GO TO 490
8490 REM Destruccion robot
8500 RESTORE 9110: FOR m=30 TO -
10 STEP -10
8510 READ w$: BEEP .1,m
8520 PRINT AT y,x; OVER 1;w$
8530 NEXT m
8540 LET d1=d1-1: PRINT AT 0,10;
 PAPER 1; INK 7;d1
8550 PRINT AT y,x;" ": IF d1=0 T
HEN GO TO 8700
8560 GO TO 8030
8600 DATA "$","*","+","-"
8690 REM Fin
8700 FOR n=0 TO 2: FOR m=10 TO -
30 STEP -5: BEEP .02,m: NEXT m:
NEXT n
8720 INPUT FLASH 1;"Otra partida
 (s/n)";w$
8730 IF w$="s" THEN GO TO 100
8740 CLS : STOP
9000 RESTORE 9050
9010 FOR n=0 TO 7: READ y$
9015 FOR m=0 TO 7
9020 READ p: POKE USR y$+m,p
9030 NEXT m: NEXT n
9040 RETURN
9045 REM Robot izquierda
9050 DATA "a",12,12,6,13,20,12,1
9,32,"d",48,48,24,120,16,40,44,3
2,"g",192,192,96,208,64,192,96,6
4
9055 REM Robot derecha
9060 DATA "c",48,48,96,176,40,48
,200,4,"f",12,12,24,30,8,20,52,4
,"i",3,3,6,11,2,3,6,2
9065 REM Elevador
9070 DATA "j",255,255,153,165,19
5,0,0,0
9075 REM Robot en el elevador
9080 DATA "k",24,24,60,90,60,24,
24,24
9100 DATA "
      ",6,2,"
      ",12,6,"
        ",17,2,"     ",19,1," "+CH
R$ 16+CHR$ 7+CHR$ 17+CHR$ 3+"WC"
,4,29
9110 DATA "$","*","+","-","↑"
```



GRAFICOS

El programa realiza la representación espacial de funciones de dos variables. Además, la función es representada en perspectiva con lo que se le intenta dar la visión tridimensional que tanto apasiona a nuestros lectores.

Para probarlo, Manuel Roig Riu nos envía cuatro ejemplos:

Z = f(x,y) = 1/(ABS (x * y) * .5)
Valor inicial de x = —3
Valor final de x = 3
Valor inicial de y = —3
Valor final de y = 3

Z = f(x,y) = SEN x + COS y
Valor inicial de X = 0
Valor final de x = 8
Valor inicial de y = 0
Valor final de Y = 8

Z = f(x,y) = SIN x
Valor inicial de x = —8
Valor final de x = 8
Valor inicial de y = 0
Valor final de y = 1

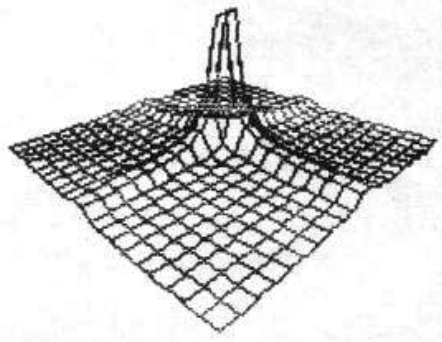
Z = f(x,y) = EXP (—x * x) EXP (—y * y/4)
Valor inicial de x = —4
Valor final de x = 4
Valor inicial de y = —4
Valor final de y = 4

(16K — Spectrum)

PROGRAMA GANADOR DE 5.000 PTAS

Manuel Roig Riu, de Tabernes de Valldigna, Valencia, es el autor de este programa que, por su publicación, le ha hecho acreedor a uno de nuestros premios de 5.000 pesetas. Esperamos que este lector tenga otros programas para enviarnos.

```
  10 BORDER 1: PAPER 1: INK 7: C
LEAR : PAPER 1: INK 7
  40 PRINT AT 20,3;"Espera unos
instantes";AT 13,8; FLASH 1;" PA
RA LA CINTA ": DIM x(21,21): FOR
 n=0 TO 20: FOR m=0 TO 20: LET x
(m+1,n+1)=(45568-850*m+5202*n)/(
356+17*(m+n)): NEXT m: NEXT n: G
O SUB 9910
  50 CLS : PRINT "  ESTE ES UN P
ROGRAMA QUE PERMI-TE LA  REPRESE
NTACION  TRIDIMEN-SIONAL Y EN PE
RSPECTIVA DE  FUN-CIONES DE DOS
VARIABLES.": PRINT : PRINT "  DE
BES SIEMPRE DENOMINAR A LAS VARI
ABLES INDEPENDIENTES ""x"" E  ""
y""."
  60 PRINT AT 21,3;" PULSA UNA T
ECLA "
  70 IF INKEY$="" THEN GO TO 70
  80 CLS
  90 INPUT "Introduzca z=f(x,y)
"; LINE a$
 100 DEF FN f()=VAL a$
 110 DIM z(22,22)
 120 INPUT "Valor inicial de x=
";xi: INPUT "Valor final de x =
";xf
 130 IF xi=xf THEN PRINT FLASH 1
;"ERROR: El valor inicial es igu
al al del valor final: FIJATE BI
EN": GO TO 120
 140 INPUT "Valor inicial de y=
";yi: INPUT "Valor final de y =
";yf
 150 IF yi=yf THEN PRINT FLASH 1
;"ERROR: El valor inicial es igu
al al del valor final: FIJATE BI
EN": GO TO 140
 170 POKE 23674,0: POKE 23673,0:
 POKE 23672,0
 180 LET m=1: LET n=1: LET x=xi:
 LET y=yi: LET zi=FN f(): LET zf
=FN f()
 190 FOR x=xi TO xf STEP (xf-xi)
/20.9999: FOR y=yi TO yf STEP (y
f-yi)/20.9999
 200 IF m=2 AND n=1 THEN GO SUB
9800
 205 LET z(m,n)=FN f()*(x(m,n)+5
0)/(178+17*(m+n-2))
 210 IF z(m,n)>zf THEN LET zf=z(
m,n)
 220 IF z(m,n)<zi THEN LET zi=z(
m,n)
 230 LET n=n+1: NEXT y: LET n=1:
 LET m=m+1: NEXT x
 240 LET a=55/(zf-zi)
 300 CLS : FOR m=1 TO 21: FOR n=
1 TO 21
 310 LET x=x(m,n): LET y=178*(17
*m-128+x(m,n))/(178+17*m)+a*(z(m
,n)-zi): PLOT x,y: IF n>1 THEN D
RAW x(m,n-1)-x,178*(17*m-128+x(m
,n-1))/(178+17*m)+a*(z(m,n-1)-zi
)-y: PLOT x,y
 320 IF m>1 THEN DRAW x(m-1,n)-x
,178*(17*(m-1)-128+x(m-1,n))/(17
8+17*(m-1))+a*(z(m-1,n)-zi)-y
 340 NEXT n: NEXT m
 400 INPUT "Parar?: (s/n) ";r$
 410 IF r$="s" OR r$="S" THEN ST
OP
 420 IF r$<>"n" AND r$<>"N" THEN
 GO TO 400
 430 INPUT "Nueva funcion?: (s/n
) ";r$
 440 IF r$="s" OR r$="S" THEN GO
 TO 90
 450 IF r$="n" OR r$="N" THEN GO
 TO 120
 460 GO TO 430
9800 REM Control del tiempo
9810 LET seg=INT (.4*PEEK 23672+
102.4*PEEK 23673+26214.4*PEEK 23
674-5)
9820 LET min=INT (seg/60): LET s
eg=seg-60*min: LET hor=INT (min/
60): LET min=min-60*hor
9830 PRINT AT 0,2;"TARDARE APROX
IMADAMENTE"
9840 PRINT AT 2,2;hor;" Horas  "
;min;" min.  ";seg;" seg."
9850 RETURN
9900 REM Musica
9910 FOR n=-20 TO 30: BEEP .01,n
: NEXT n: RETURN
```



NUMEROS PRIMOS

El cálculo de números primos puede convertirse en una labor tediosa, siempre que no tenga un Spectrum a mano. Si no es ése su caso, puede introducir el programa que nos ha sido remitido por Fernando, con el que obtener todos los números primos comprendidos entre unos intervalos previamente introducidos po r teclado.

(16K-Spectrum)

PROGRAMA GANADOR DE 5.000 PTAS

Es un programa corto, pero tiene su mérito. Su autor es Fernando Sánchez Gabinez, de Cartagena. Cinco mil pesetas de premio para él.



```
    5 REM "NUM-PRIMOS"
   10 PRINT "Programa para invest
igar los nu-meros primos compren
didos entre dos enteros impares
positivos.": PRINT
   20 DIM n(5)
   30 INPUT "Ponga numero menor a
 investigar",n
   40 INPUT "Ponga numero mayor a
 investigar",num: CLS
   50 LET c=0: LET num=num+1
   60 LET L=0
   70 LET J=INT (SQR n)
   80 FOR K=3 TO J STEP 2
   90 LET L=INT (n/K)
  100 IF L*K-n>=0 THEN GO TO 140
  110 NEXT K
  120 LET c=c+1
  130 LET n(c)=n
  140 LET n=n+2
  150 IF c-5=0 THEN PRINT TAB 2;n
(1);TAB 8;n(2);TAB 14;n(3);TAB 2
0;n(4);TAB 26;n(5): LET c=0: FOR
 i=1 TO 5: LET n(i)=0: NEXT i
  160 IF num-n>0 THEN GO TO 70
  170 IF c=0 THEN GO TO 190
  180 PRINT TAB 2;n(1);TAB 8;n(2)
;TAB 14;n(3);TAB 20;n(4);TAB 26;
n(5)
  190 PRINT
  200 PRINT "Fin de programa"
  210 STOP
```

```
 1      3      5      7      11
 13     17     19     23     29
 31     37     41     43     47
 53     59     61     67     71
 73     79     83     89     97
 101    103    107    109    113
 127    131    137    139    149
 151    157    163    167    173
 179    181    191    193    197
 199    0      0      0      0

Fin de programa
```

MASTER MIND

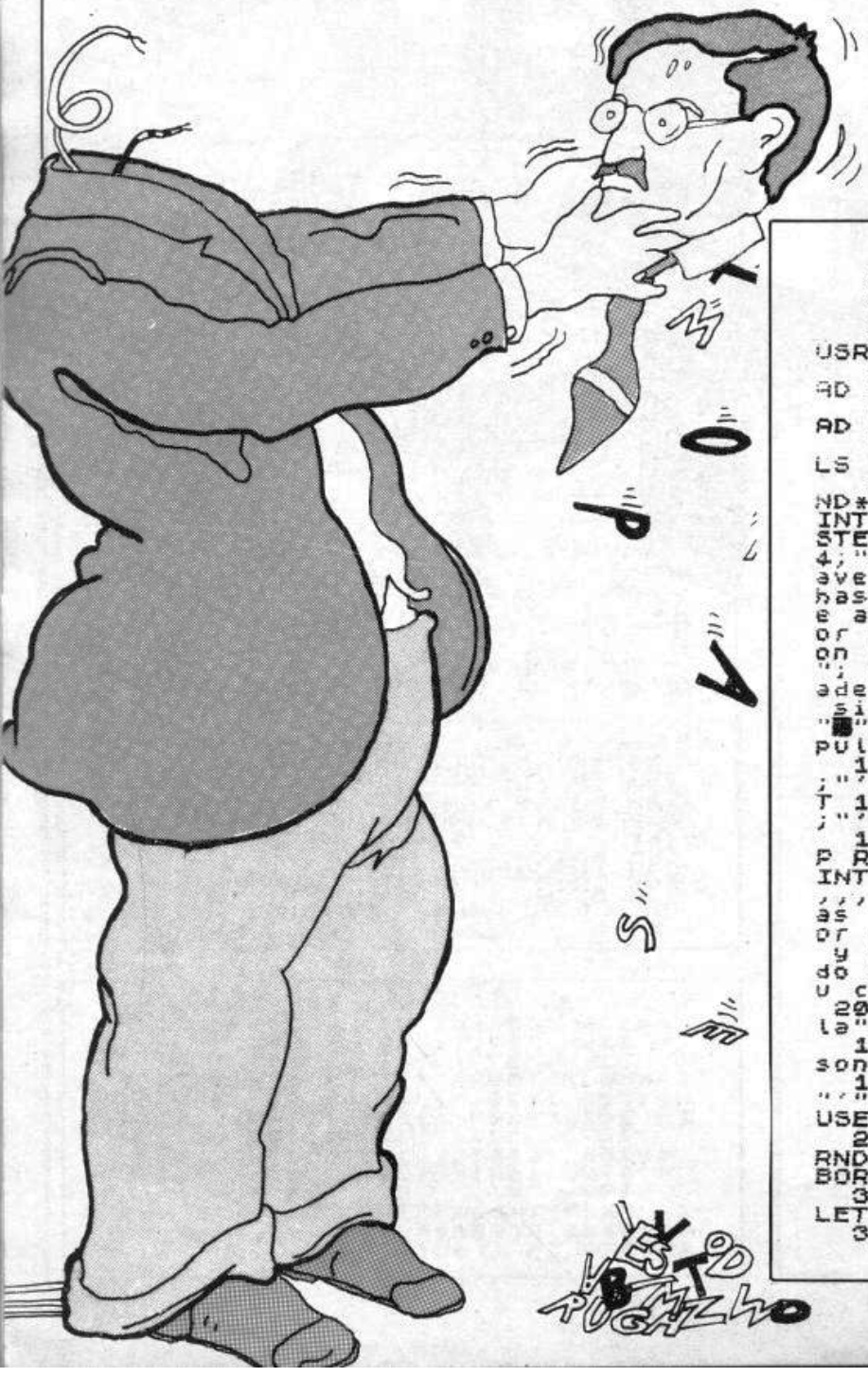
En el número anterior ya incluimos una versión del Master Mind, pero ésta es mejor, especialmente por los gráficos. Apenas hacen falta explicaciones, pues se dice todo por pantalla. Como usted sabe ya, el juego consiste en descubrir una clave de colores que el ordenador genera. Cuando usted hace su intento el Spectrum le indicará si ha utilizado los colores de que consta la clave y si éstos están en su sitio. Con esa información habrá de preparar su siguiente intento a fin de acertar con la "combinación secreta".

En este programa se permite dar un número variable de colores e incluso posibilita la repetición de colores, con los que se puede alcanzar niveles de dificultad extraordinarios.

(16K-Spectrum)

PROGRAMA GANADOR DE 5.000 PTAS

Este Master Mid nos ha sido remitido por José Ignacio de Angel Martín, de Madrid. Como corresponde, le enviamos por correo el premio de 5.000 pesetas.



```
    1 DATA 0,16,0,16,32,66,120,0
    2 DATA 0,0,4,2,127,2,4,0
    3 DATA 0,120,0,120,66,66,68,0
    4 FOR n=0 TO 7: READ a: POKE
USR "c"+n,a: NEXT n
    5 RESTORE 2: FOR n=0 TO 7: RE
AD a: POKE USR "h"+n,a: NEXT n
    6 RESTORE 3: FOR n=0 TO 7: RE
AD a: POKE USR "n"+n,a: NEXT n
    8 BORDER 5: PAPER 5: INK 1: C
LS
    9 FOR n=0 TO 10: BEEP RND/5,R
ND*6: NEXT n: BRIGHT 0: CLS : PR
INT ; FLASH 1; INK 0;AT 5,10;"MA
STER MIND"; FLASH 0; INK 1;AT 8,
4;"En mi memoria guardo una","cl
ave secreta de colores que","tu
has de averiguar.";AT 12,4;"Yo t
e ayudare";AT 14,4;"Por cada col
or de tu clave",,"que coincida c
on la mia te lo","indicare asi:
"; INK 0;"■";AT 18,4; INK 1;"Si
ademas de coincidir esta","en su
 sitio te lo indicare: "; INK 0;
"■";AT 21,2; INK 1;"Para seguir
pulsa una tecla"
   10 PRINT OVER 1; INK 1;AT 10,1
;"'";AT 12,16;"'";AT 15,21;"'";A
T 16,7;"'";AT 16,11;"'";AT 18,11
;"'";AT 18,30;"'";AT 19,25;"'"
   11 PAUSE 0: FOR n=0 TO 10: BEE
P RND/3,RND*20: NEXT n: CLS : PR
INT AT 5,4;"Cuando te pregunte:"
,,,"""¿Conforme? (s/n)"" y escoj
as ""N""";"podras cambiar el col
or que","señale la flecha ""↑"",
 y esta la","podras mover pulsan
do SPACE,,,,"  Cuando tengas t
u combinación","pulsa ""S"".";AT
 20,2;"Para seguir pulsa una tec
la"
   12 PRINT AT 15,4;"Los colores
son del 1 al 6"
   14 PRINT OVER 1; INK 1;AT 8,4;
"'";AT 10,4;"'";AT 12,29;"'": PA
USE 0
   25 FOR n=1 TO 10: BEEP RND/10,
RND*40: NEXT n: CLS : BRIGHT 0:
BORDER 6: PAPER 6: INK 0: CLS
   30 LET x=2: LET y=2: LET d=0:
LET e=0
   35 PRINT AT 6,3;"Para instrucc
```

```
iones pulsa I";AT 10,3;"¿Cuantos
 colores pongo?:"
  40 PAUSE 0
  42 IF INKEY$="2" THEN LET c=2
  43 IF INKEY$="3" THEN LET c=3
  44 IF INKEY$="4" THEN LET c=4
  45 IF INKEY$="5" THEN LET c=5
  46 IF INKEY$="6" THEN LET c=6
  47 IF INKEY$="i" OR INKEY$="I"
 THEN GO TO 8
  48 IF INKEY$<"2" OR INKEY$>"6"
 THEN GO TO 40
  49 BEEP .3,40: PRINT AT 10,28;
c
  50 DIM a(c): DIM b(c): DIM i(c
)
  51 PRINT AT 12,3;"¿Quieres rep
atidos? (s/n)"
  52 IF INKEY$="s" OR INKEY$="S"
 THEN LET g=5: GO TO 57
  53 IF INKEY$="n" OR INKEY$="N"
 THEN LET g=2: GO TO 57
  54 IF INKEY$="i" OR INKEY$="I"
 THEN GO TO 8
  56 IF INKEY$<>"i" OR INKEY$<>"
I" OR INKEY$<>"s" OR INKEY$<>"S"
 OR INKEY$<>"n" OR INKEY$<>"N" T
HEN GO TO 52
  57 BEEP .3,40: CLS : BORDER 5:
 PAPER 7: BRIGHT 1: GO TO (g*100
)-100
  100 CLS : PRINT FLASH 1;AT x,0;
"→"; FLASH 0;AT x,(c*2)+4;"Pon t
us colores"
  110 FOR n=1 TO c
  120 LET a(n)=INT (RND*6)+1: FOR
 m=1 TO c: IF n=m THEN GO TO 140
  130 IF a(n)=a(m) THEN GO TO 120
  140 NEXT m: NEXT n
  200 FOR n=1 TO c
  201 PAUSE 0: IF INKEY$="1" THEN
 LET b(n)=1: GO TO 230
  202 IF INKEY$="2" THEN LET b(n)
=2: GO TO 230
  203 IF INKEY$="3" THEN LET b(n)
=3: GO TO 230
  204 IF INKEY$="4" THEN LET b(n)
=4: GO TO 230
  205 IF INKEY$="5" THEN LET b(n)
=5: GO TO 230
  206 IF INKEY$="6" THEN LET b(n)
=6: GO TO 230
  207 IF INKEY$<"1" OR INKEY$>"6"
 THEN GO TO 200
  230 BEEP .05,30
  231 FOR m=1 TO c: IF n=m THEN P
RINT INK b(n);AT x,y;"■": LET y=
y+2: NEXT n: GO TO 800
  240 IF b(n)=b(m) THEN BEEP .5,-
20: PRINT AT x,(c*2)+4;"No valen
 repetidos": PAUSE 100: PRINT AT
 x,(c*2)+4;"                ":
 GO TO 201
  250 NEXT m
  400 CLS : PRINT FLASH 1;AT x,0;
"→"; FLASH 0;AT x,(c*2)+4;"Pon t
us colores"
  410 FOR n=1 TO c: LET a(n)=INT
(RND*6)+1: NEXT n
  500 FOR n=1 TO c
  501 PAUSE 0: IF INKEY$="1" THEN
 LET b(n)=1: GO TO 520
  502 IF INKEY$="2" THEN LET b(n)
=2: GO TO 520
  503 IF INKEY$="3" THEN LET b(n)
=3: GO TO 520
  504 IF INKEY$="4" THEN LET b(n)
=4: GO TO 520
  505 IF INKEY$="5" THEN LET b(n)
=5: GO TO 520
  506 IF INKEY$="6" THEN LET b(n)
=6: GO TO 520
  507 IF INKEY$<"1" OR INKEY$>"6"
 THEN GO TO 501
  520 BEEP .02,30
  530 PRINT INK b(n);AT x,y;"■":
 LET y=y+2: NEXT n: GO TO 800
  800 PRINT AT x,(c*2)+4;"¿Confor
me? (s/n)"
  830 IF INKEY$="s" OR INKEY$="S"
 THEN GO TO 1000
  840 IF INKEY$="n" OR INKEY$="N"
 THEN PRINT AT x,y+2;"Pon otros
      ": GO TO 3000
  850 IF INKEY$<>"s" OR INKEY$<>"
S" OR INKEY$<>"n" OR INKEY$<>"N"
 THEN GO TO 830
1000 BEEP .5,30: LET y=(c*2)+4:
PRINT AT x,y;"
": LET d=0: LET e=0: LET h=0
1005 FOR n=1 TO c: LET i(n)=a(n)
: NEXT n
1006 FOR n=1 TO c
1007 IF i(n)=b(n) THEN LET d=d+1
: LET i(n)=0: LET b(n)=0
1008 NEXT n
1010 FOR n=1 TO c
1015 IF b(n)=0 AND n=c THEN GO T
O 1060
1020 IF b(n)=0 THEN NEXT n
1030 FOR f=1 TO c
1040 IF i(f)=b(n) AND i(f)<>0 TH
EN LET e=e+1: LET i(f)=0: LET b(
n)=0
1050 NEXT f: NEXT n
1060 GO SUB 2000
1070 LET y=2: LET x=x+2: GO TO g
*100
1500 IF n=f THEN LET d=d+1: RETU
RN
1510 LET e=e+1
1530 RETURN
2000 IF e>0 THEN LET e=e-1: BEEP
 .03,10: PRINT AT x,y+h;"▬": LET
 h=h+2: GO TO 2000
2005 IF d=c THEN GO TO 2500
2010 IF d>0 THEN LET d=d-1: BEEP
 .05,50: PRINT AT x,y+h;"■": LET
 h=h+2: GO TO 2010
2013 IF x>18 THEN PRINT AT 21,2;
"Muy mal": LET v=10: FOR n=1 TO
 c: PRINT FLASH 1; INK a(n);AT 21
,v;"■": LET v=v+2: NEXT n; GO TO
 3500
2015 PRINT AT x,0;" "; FLASH 1;A
T x+2,0;"→"; FLASH 0;AT x+2,2;"P
rueba otra vez": FOR n=0 TO 100:
 NEXT n: PRINT AT x+2,2;"
                "
2020 RETURN
2500 LET y=(c*2)+4: FOR n=1 TO c
: PRINT FLASH 1;;AT x,y;"■": LET
 y=y+2: NEXT n
2520 GO TO 3400
3000 BEEP .5,0
3005 LET y=0
3010 LET y=y+2: IF y=(c*2)+2 THE
N GO TO 3005
3020 PRINT AT x+1,y;"↑"
3030 PAUSE 0: IF INKEY$=CHR$ 32
THEN BEEP .08,10: PRINT AT x+1,y
;" ": GO TO 3010
3031 IF INKEY$="1" THEN LET b(y/
2)=1: GO TO 3050
3032 IF INKEY$="2" THEN LET b(y/
2)=2: GO TO 3050
3033 IF INKEY$="3" THEN LET b(y/
2)=3: GO TO 3050
3034 IF INKEY$="4" THEN LET b(y/
2)=4: GO TO 3050
3035 IF INKEY$="5" THEN LET b(y/
2)=5: GO TO 3050
3036 IF INKEY$="6" THEN LET b(y/
2)=6: GO TO 3050
```

```
3037 IF INKEY$="s" OR INKEY$="S"
 THEN GO TO 3100
3040 IF INKEY$<>CHR$ 32 OR INKEY
$<"1" OR INKEY$>"6" THEN GO TO 3
030
3050 BEEP .04,45: PRINT INK b(y/
2);AT x,y;"■": GO TO 3020
3100 PRINT AT x+1,y;" ": IF g=5
THEN GO TO 1000
3110 FOR f=1 TO c: FOR n=1 TO c:
 IF n=f THEN GO TO 3130
3120 IF b(n)=b(f) THEN GO TO 320
0
3130 NEXT n: NEXT f
3140 GO TO 1000
3200 PRINT AT x,(c*2)+4;"No vale
n repetidos": BEEP .5,-20: PAUSE
 125: BEEP .3,40: PRINT AT x,(c*
2)+4;"Pon otros          ": GO TO
 3000
3400 BEEP .1,10: BEEP .1,10: BEE
P .1,10: BEEP .3,15: BEEP .3,15:
 BEEP .3,17: BEEP .3,17: BEEP .5
,22: BEEP .1,19: BEEP .15,15: PA
USE 2: BEEP .12,15: BEEP .12,19:
 BEEP .15,15: BEEP .2,12: BEEP .
5,20: BEEP .2,17: PAUSE 1: BEEP
.1,14: BEEP .1,15
3500 PRINT FLASH 1; BRIGHT 1; PA
PER 6; INK 1;AT 21,26;"¿OTRA?"
3510 IF INKEY$<>"" THEN GO TO 25
3520 GO TO 3510
```

```
MASTER MIND

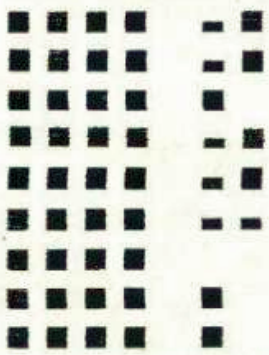
   En mi memoria guardo una
clave secreta de colores que
tú has de averiguar.

   Yo te ayudaré

   Por cada color de tu clave
que coincida con la mía te lo
indicaré así: ▬

   Si además de coincidir está
en su sitio te lo indicaré: ■

  Para seguir pulsa una tecla
```



(12 NUMEROS)

TARIFA DE PRECIOS DE SUSCRIPCION

	CORREO ORDINARIO		CORREO CERTIFICADO		CORREO AEREO		CORREO AEREO-CERTIF.	
	PTAS.	$	PTAS.	$	PTAS.	$	PTAS.	$
ESPAÑA	2.400	17	2.676	19	2.448	17	2.724	19
EUROPA, MARRUECOS, TUNEZ, TURQUIA, ARGELIA Y CHIPRE	2.856	20	3.672	26	2.964	21	3.780	27
COSTA RICA, CUBA, CHILE, PARAGUAY Y REP. DOMINICANA	2.796	20	3.612	26	3.372	24	4.188	30
GIBRALTAR Y PORTUGAL	2.664	19	3.480	25	2.515	18	3.331	24
FILIPINAS	2.664	19	2.940	21	2.983	21	3.259	23
RESTO DEL MUNDO	2.856	20	3.672	26	3.432	25	4.248	30

CUPON DE PEDIDO

Recorte y envíe este cupón a: **ZX**, Jerez, 3 - Madrid-16

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 _____

DIBUJO DE UNA FUNCION

Son muchos los programas de representaciones gráficas que recibimos. Sin duda es el primer paso de todos aquellos que desde temprana edad pierden el miedo al Spectrum. Este programa enviado por Eduardo Orts Miralles es bastante simple, con pocas instrucciones, pero con buena resolución gráfica.

El programa nos pide que introduzcamos una función f(x), que se almacena como un *string*. Después se dan valores a "x" desde —15,8 a 16. Los valores de "y" se convierten en coordenadas del *pixel* y se dibuja el punto. Se ha tomado la posición de cada carácter como una unidad, por lo que el eje de las "y" va desde —11 a 10,875.

Si se quiere variar la separación entre los puntos dibujados, se puede modificar la línea 70 cambiando el STEP.1 por STEP.2 ó STEP05. Como ejemplos se puede probar:

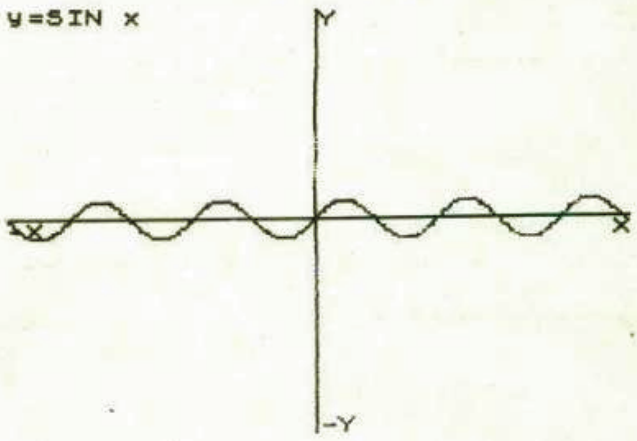
```
3 * x * x  + 10 * x —27
x * x
x * SIN x
7/x
```

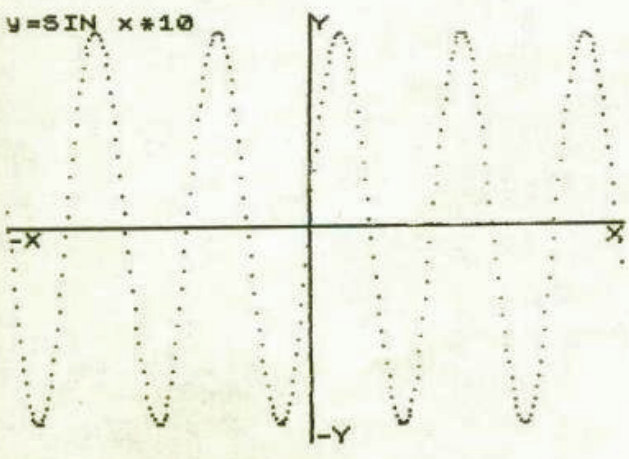
(16K-Spectrum)

PROGRAMA GANADOR DE 5.000 PTAS

Como decimos en el texto, este programa es bastante simple, pero muy didáctico. Por eso lo publicamos. Y, por haber sido publicado, Eduardo Orts Miralles, que vive en Alicante, se ha ganado 5.000 pesetas.

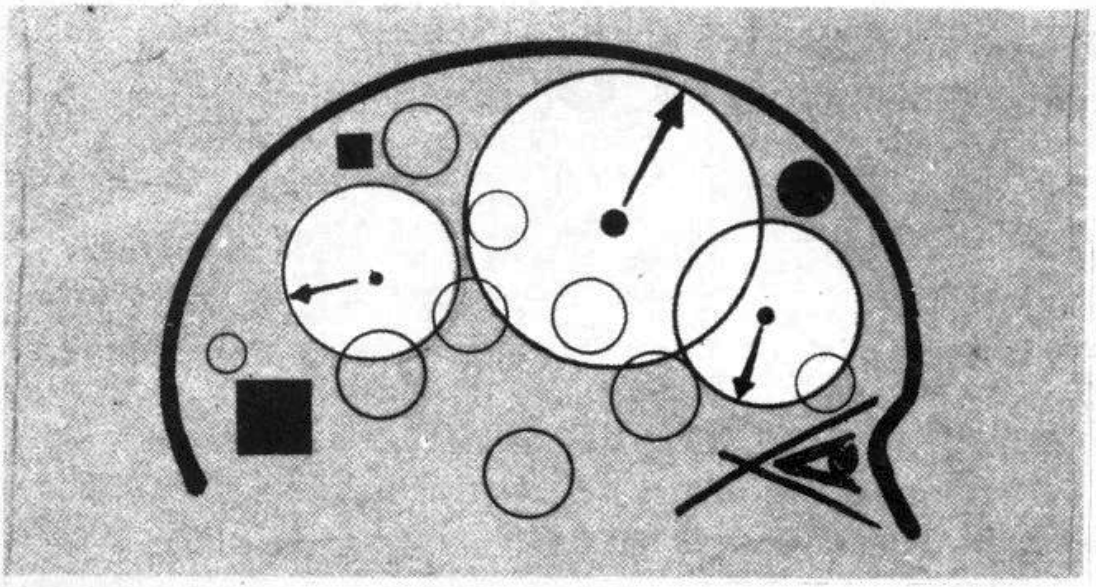
```
  10 REM Dibuja una funcion
  20 REM @ Eduardo Orts,28-12-82
  30 BORDER 5: PAPER 5: BRIGHT 1
: INK 0
  35 CLS
  40 BEEP .2,-5: INPUT AT 0,0;"T
ecleee una funcion ","y= ";y$
  45 PRINT AT 0,0;"y=";y$
  50 PLOT 127,0: DRAW 0,175: PLO
T 0,88: DRAW 255,0
  60 PRINT AT 11,0;"-X";AT 11,31
;"X";AT 0,16;"Y";AT 21,16;"-Y"
  70 FOR x=-15.8 TO 16 STEP .1
  80 LET y=VAL y$
  90 IF y>10.875 OR y<-11 THEN G
O TO 130
 100 LET a=INT (127+x*8)
 110 LET b=INT (88+y*8)
 120 PLOT a,b
 130 NEXT x
 140 BEEP .2,-5: INPUT AT 1,0;"D
esea volver a empezar?(s/n)";r$
 150 IF r$="s" THEN GO TO 35
```





GANADOR DEL ZX DE ESTE MES
MICRODRIVE

Este mes, el sorteo de un ZX Microdrive ha favorecido a un programa que no publicamos. Se trata del Golf que nos enviara el lector Vicente Díaz, de Alcalá de Henares, para el que no hemos encontrado manera de cargar el *cassette* que nos hizo llegar. No obstante esta circunstancia, que impide su publicación, la suerte ha querido que fuera el ganador de nuestro premio especial. Naturalmente, como ya advertíamos en la convocatoria del concurso, este lector deberá esperar hasta que el microdrive se distribuya oficialmente en España. Un poco de paciencia y mucha suerte.



PROGRAMA GANADOR DE 5.000 PTAS

Este programa nos ha sido remitido por Julián Cervera González, un lector de Málaga, a quien enviamos por correo un talón nominativo por valor de 5.000 pesetas. Felicitaciones.

Añadir:
277 GO TO 300

BIORRITMOS

Como es sabido la teoría de los biorritmos afirma que desde la fecha de nacimiento de cada persona existen unos ciclos FISICO (de 23 días), EMOCIONAL (de 28 días) e INTELECTUAL (de 33 días) que condicionan la conducta de las personas. Existen estadísticas, por ejemplo de accidentes de automóviles, que parecen confirmar la existencia de unos días "críticos" que se producen cuando cambia la fase del biorritmo, bien por pasar de la zona positiva a la negativa o viceversa.

El programa pregunta la fecha de nacimiento y el mes del que se desea el biorritmo. Calcula el número de días transcurridos y dibuja en distintos colores las curvas sinoidales correspondientes a cada uno de los biorritmos físico, emocional e intelectual antes señalados.

(Spectrum 16K)

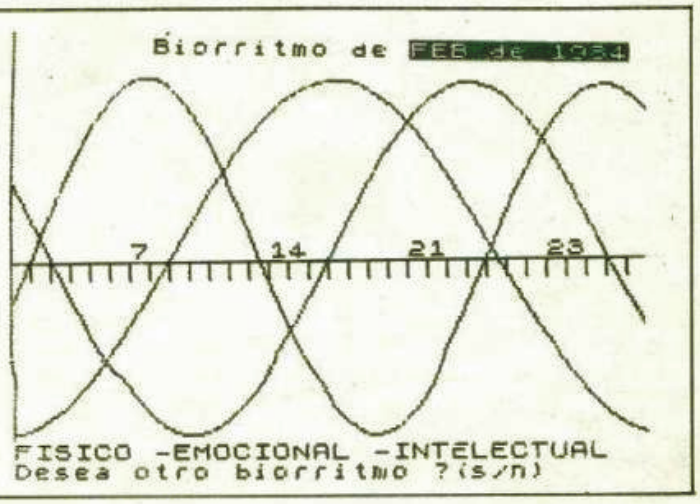
Notas gráficas

Línea 950: Graphic "a".

```
  20 DIM m$(31,3): DIM m(12): DI
M Y(3)
  25 GO SUB 900
  70 CLS
  80 BORDER 6: PAPER 6: INK 0
  90 LET n1=0
 100 GO SUB 872
 109 REM Calculo de dias
 110 IF a2-a1=1 THEN GO TO 170
 120 IF a2-a1=0 THEN GO TO 280
 130 FOR j=a1+1 TO a2-1
 140 LET n1=n1+365
 150 IF INT (j/4)=j/4 THEN LET n
1=n1+1
 160 NEXT j
 170 IF m1=12 THEN GO TO 210
 180 FOR j=m1+1 TO 12
 190 LET n1=n1+m(j)
 200 NEXT j
 210 LET n1=n1+m(m1)-d1
 220 IF m2=1 THEN GO TO 260
 230 FOR j=1 TO m2-1
 240 LET n1=n1+m(j)
 250 NEXT j
 260 LET n1=n1+d2
 270 IF INT (a1/4)=a1/4 AND m1<=
2 THEN LET n1=n1+1
 275 IF INT (a2/4)=a2/4 AND m2>2
 THEN LET n1=n1+1
 280 IF m1=12 THEN GO TO 300
 290 FOR j=m1+1 TO m2: LET n1=n1
+m(j): NEXT j
 300 LET n1=n1+m(m1)-d1-m(m2)+d2
 310 IF INT (a1/4)=a1/4 AND m1<=
2 AND m2>2 THEN LET n1=n1+1
 330 PRINT '''';"NUMERO DE DIAS T
RANSCURRIDOS :": PRINT 'TAB 25;n
1
 335 PRINT ''' FLASH 1;"teclee p
ara continuar"
 340 PAUSE 0: CLS
 345 LET S1=N1/23-INT (N1/23): L
ET S2=N1/28-INT (N1/28): LET S3=
N1/33-INT (N1/33)
 350 FOR A=1 TO 31: PRINT AT 11,
A;"|": IF A=7 OR A=14 OR A=21 OR
 A=28 THEN PRINT AT 10,A-1;A
 355 NEXT A
 357 REM Dibujo del biorritmo
 358 PRINT INK 2;AT 0,7;"Biorrit
mo de "; INVERSE 1;m$(m2);" de "
;a2
 360 PLOT 0,87: DRAW 255,0
 370 PLOT 0,0: DRAW 0,175
 375 PRINT AT 20,0; INK 3;"FISIC
O "; INK 4;"-EMOCIONAL "; INK 5;
"-INTELECTUAL"
 385 FOR A=1 TO 3
 387 LET t=0: LET OLD Y=0
 390 FOR F=0 TO 255 STEP 3
 392 LET Y(1)=70*SIN ((F+184*S1)
*PI/92): LET Y(2)=70*SIN ((F+224
*S2)*PI/112): LET Y(3)=70*SIN ((
F+264*S3)*PI/132)
 400 IF ABS y(a)>87 THEN GO TO 4
40
 420 IF NOT t THEN PLOT F,Y(A)+8
8: LET t=1: GO TO 440
 430 INK A+2: DRAW 3,Y(A)-OLD Y
 440 LET OLD Y=INT (Y(A)+.5)
 450 NEXT F: NEXT A
 460 PRINT FLASH 1; INK 2;AT 21,
0;"Desea otro biorritmo ?(s/n)"
 480 PAUSE 0: IF INKEY$="" THEN
GO TO 480
 500 IF INKEY$="s" OR INKEY$="S"
 THEN GO TO 70
 510 STOP
```

```
  872 PRINT INK 2;AT 1,10; INVERS
E 1;"BIORRITMOS": PRINT '''"FECHA
 NACIMIENTO ?"
  874 INPUT "Dia del mes ?";d1: I
F d1<1 OR d1>31 THEN GO TO 874
  876 INPUT "mes  (1 a 12)  ?";m1:
 IF m1>12 OR m1<1 THEN GO TO 876
  878 INPUT "ano ?(1900/2050)";a1
: IF a1<1900 OR a1>2050 THEN GO
TO 878
  880 PRINT '';TAB 20;d1;TAB 23;m$
(m1);TAB 27;a1
  882 PRINT '''"MES DEL BIORRITMO
 ?"
  884 INPUT "mes ?(1/12)";m2: IF
m2<1 OR m2>12 THEN GO TO 884
  886 INPUT "ano ?(1900/2050)";a2
: IF a2>2050 OR a2<1900 THEN GO
TO 886
  888 IF a1>a2 THEN PRINT "Imposi
ble : la fecha de nacimiento es
posterior": GO TO 884
  890 LET d2=1
  892 PRINT '';TAB 20;m$(m2);TAB 2
7;a2
  895 RETURN
  900 RESTORE 960
  910 FOR A=1 TO 12: READ M(A): N
EXT A
  920 RESTORE 980
  930 FOR a=1 TO 12: READ m$(a):
NEXT a
  940 RESTORE 970
  950 FOR a=0 TO 7: READ n: POKE
```

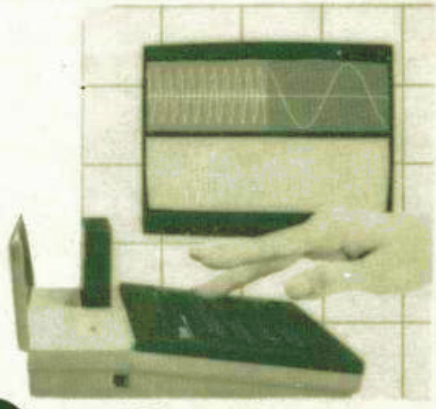
```
USR "a"+a,n: NEXT a
  960 DATA 31,28,31,30,31,30,31,3
1,30,31,30,31
  970 DATA 128,128,128,128,128,12
8,128,128
  980 DATA "ENE","FEB","MAR","ABR
","MAY","JUN","JUL","AGO","SEP",
"OCT","NOV","DIC"
  990 RETURN
```



electrónica
LUGO
ORDENADORES .



DRAGON 32



ATARI

SORD



ZX
sinclair

CABLES ESPECIALES
CONECTORES
PROGRAMAS

BARQUILLO. 40
MADRID-4

FUTBOL

Con este programa podrá retar a su Spectrum a un partido de fútbol. Tras dibujar el campo en verde, y poner el marcador a cero entre su equipo y el del Spectrum F.C. empieza el partido.

En las jugadas impares es usted quien ataca, mientras que en las pares habrá de defenderse del feroz ataque de su Spectrum. Cuando ataque se encontrará con el portero y diez defensas a los que no verá inicialmente. Cuando llegue a una posición inmediatamente cercana a la de algún defensa, el ordenador le avisará con un zumbido que no cesará hasta que se cambie de posición o sea interceptado. Si ocurre esto último, el turno pasa automáticamente al Spectrum.

Cuando avanza el último fichaje del Spectrum F.C. habrá de defenderse solamente con el portero quien además, no puede abandonar la portería. Resulta bastante difícil ganar y su autor confiesa haber ganado un par de veces por 2 a 1 y con muchos apuros. ¿Se atreve usted a dar unas pataditas? Aquí vale todo... ¡no hay árbitros! El movimiento se realiza con las teclas del cursor (5, 6, 7 y 8).

(16K-Spectrum)

Notas gráficas:

Graphic A
Graphic B
Graphic C
Graphic E
Graphic F
Graphic CA

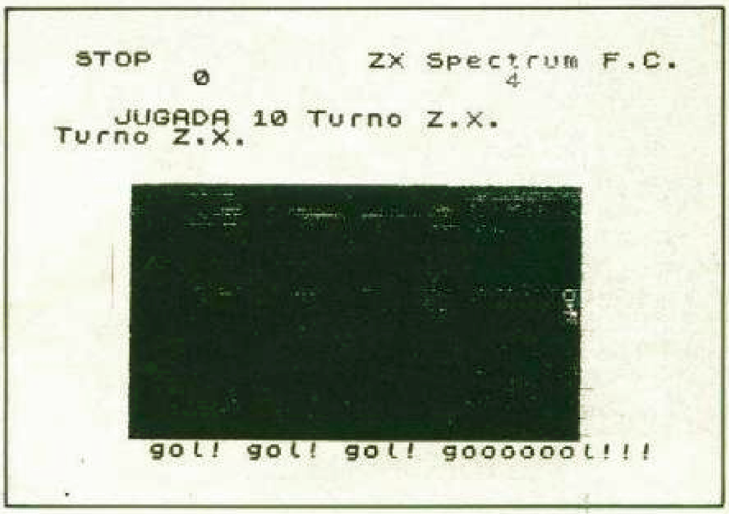
Línea 200, 219, 230, 240,
Línea 200, 219, 230, 240,
Línea 610,
Línea 219,
Línea 620, 640, 650, 1960,
Línea 1905
Línea 1600

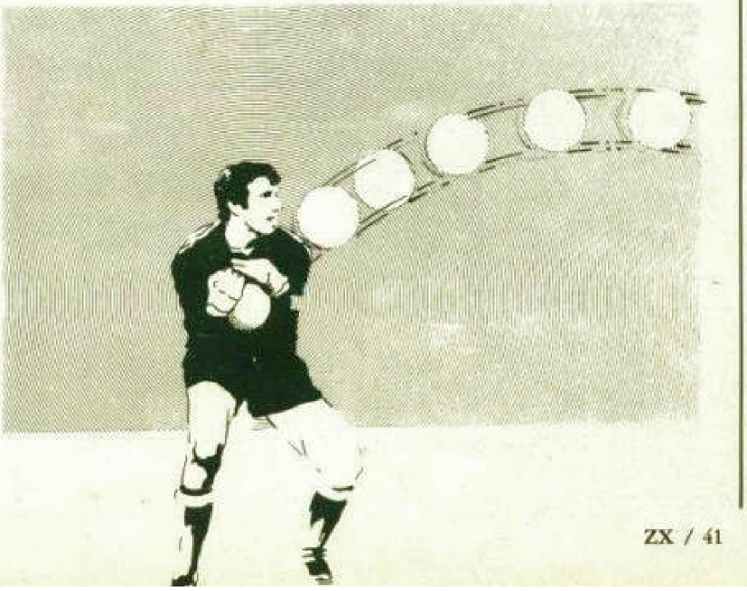
PROGRAMA GANADOR DE
5.000 PTAS

No sabemos de qué equipo es forofo nuestro lector Luis Vinagre Solans, de Lérida. Pero debe ser un buen aficionado al fútbol, a juzgar por el programa que nos ha enviado. Y que le ha permitido ganarse 5.000 pesetas.

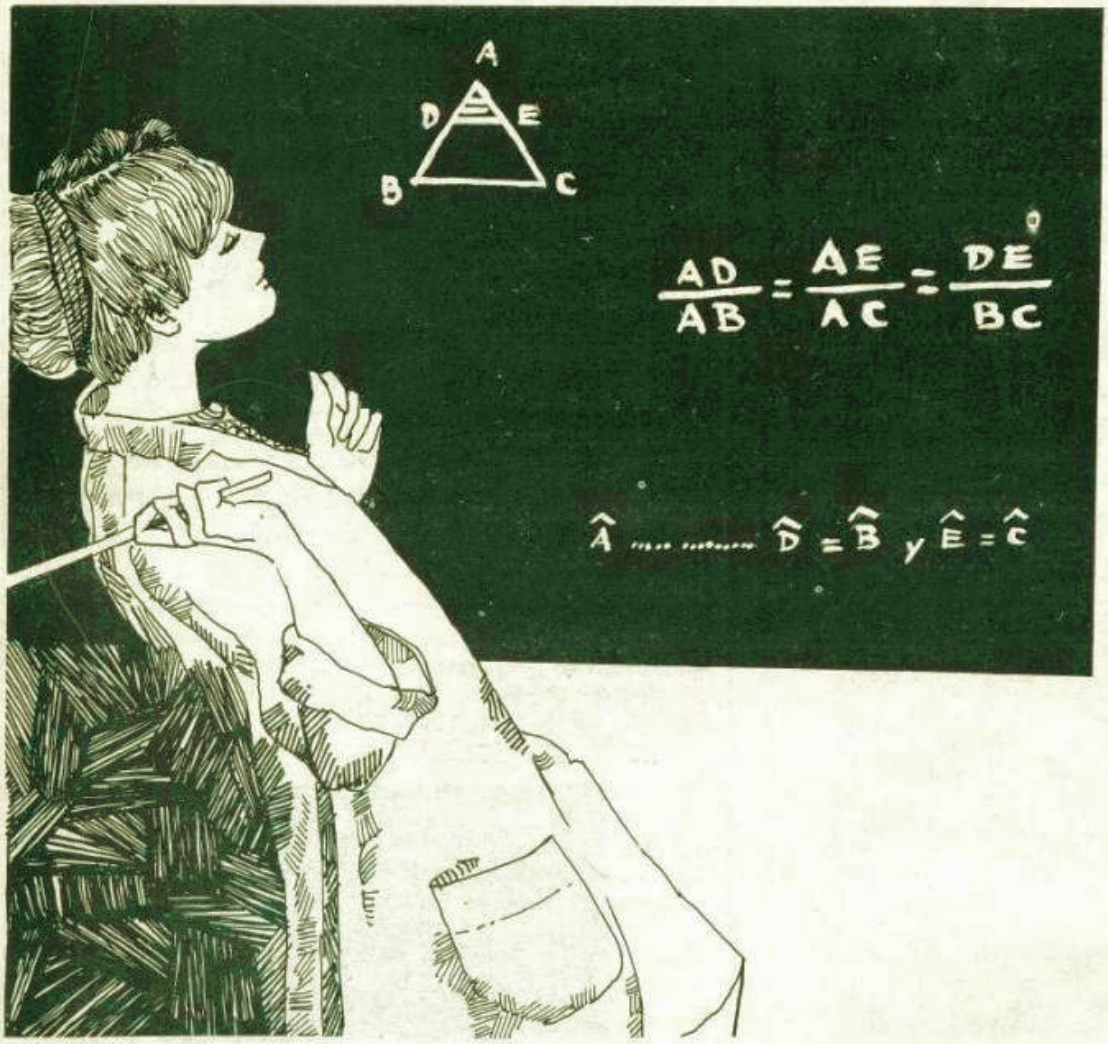
```
   2 CLS
   3 GO SUB 7000
   4 DIM k(2)
   5 DIM z(15)
   6 DIM f(10): DIM v(12)
  10 INPUT "Da el nombre a tu seleccion       "; LINE a$
  12 CLS
  20 LET b$="ZX Spectrum F.C."
  30 LET a=0
  40 LET b=0
  50 FOR n=1 TO 11
  55 IF N>10 THEN GO TO 3000
  60 GO SUB 1000
  80 IF n/2<>INT (n/2) THEN GO SUB 1200: GO TO 200
  90 IF n/2=INT (n/2) THEN GO SUB 1200: GO TO 600
 200 INK 0: PRINT AT 13,14;"♟"; OVER 1; INK 4;CHR$ 8;"█"
 201 LET f=13: LET c=14: LET g=14
 205 FOR x=12 TO 2 STEP -1: LET v=x-1
 206 IF c=4 AND f=k THEN GO TO 1900
 207 IF c=4 AND f=k(1) THEN GO TO 2000: IF c=4 AND f=k(2) THEN GO TO 2000
 208 IF x=1 THEN GO TO 210
 209 IF z(x)=f THEN GO TO 1600
 210 IF f<=10 AND c=5 THEN GO TO 1800: IF f>=16 AND c=5 THEN GO TO 1800
 211 IF f<8 THEN GO TO 1800
 212 IF f>18 THEN GO TO 1800
 215 IF C=4 THEN GO TO 1900
 216 IF f=z(x)-1 THEN GO SUB 9000
 217 IF f=z(x)+1 THEN GO SUB 9000
 218 IF f=z(v) THEN GO SUB 9000
 219 IF INKEY$="5" THEN LET c=c-1: INK 0: PRINT AT f,c;"♟"; OVER 1; INK 4;CHR$ 8;"█": BEEP .1,40: INK 4: PRINT AT f,c+1;"█": INK 0: PRINT AT z(x),g;"♟"; OVER 1; INK 4;CHR$ 8;"█": INK 0: LET g=g-1: NEXT x
 220 IF INKEY$="8" THEN GO TO 1500
 230 IF INKEY$="6" THEN LET f=f+1: PRINT AT f,c;"♟"; OVER 1; INK 4;CHR$ 8;"█": BEEP .1,40: INK 4: PRINT AT f-1,c;"█"
 240 FLASH 0: IF INKEY$="7" THEN LET f=f-1: PRINT AT f,c;"♟"; OVER 1; INK 4;CHR$ 8;"█": BEEP .1,40: INK 4: PRINT AT f+1,c;"█"
 250 GO TO 206
 600 LET q=13: LET k=0
 605 FOR x=16 TO 26: LET k=k+1
 606 LET i=INT (RND*3)+12
 607 LET v(k)=i
 608 INK 5
 610 PRINT AT v(k),x; INK 3;"♟"; OVER 1; INK 4;CHR$ 8;"█": PAUSE 20: PRINT INK 4;AT v(k),x;"█"
 620 INK 0: PRINT AT q,26;"♟"; OVER 1; INK 4;CHR$ 8;"█"
 630 BEEP .01,40
 631 IF q<=11 OR q>=16 THEN LET o=0: GO TO 635
 633 LET o=4
 635 IF q<12 OR q>15 THEN LET ti=0: GO TO 640
 636 LET ti=4
 640 IF INKEY$="6" THEN PRINT INK o;AT q,26;"█": LET q=q+1: PRINT AT q,26;"♟"; OVER 1; INK ti;CHR$ 8;"█": INK 0
 650 IF INKEY$="7" THEN PRINT INK o;AT q,26;"█": LET q=q-1: PRINT AT q,26;"♟"; OVER 1; INK ti;CHR$ 8;"█": INK 0
 660 IF v(k)<>q AND x=26 THEN GO TO 2100
 670 IF v(k)=q AND x=26 THEN GO TO 1950
 680 NEXT x
1000 CLS
1005 INK 0
1010 PRINT AT 0,0;a$,
1011 PRINT AT 0,16;b$
1020 PRINT AT 1,7;a: PRINT AT 1,23;b
1023 IF n/2=INT (n/2) THEN LET c$=" Turno Z.X.        "
1024 IF n/2<>INT (n/2) THEN LET c$=" Turno humano"
1025 LET m=8: PRINT AT 3,3;"JUGADA ";n;c$: IF n/2=INT (n/2) THEN PRINT "Turno Z.X.": IF n/2<>INT (n/2) THEN PRINT "Turno humano"
1049 INK 0
1050 INK 0: PRINT AT 7,4;"████████████████████████"
1060 PRINT AT 8,2;"  █                  █"
1070 PRINT AT 9,2;"  █                  █"
```

```
1080 PRINT AT 10,2;"   █
   █"
1090 PRINT AT 11,2;"   █
   █"
1100 PRINT AT 12,2;" "
1110 PRINT AT 13,2;" "
1120 PRINT AT 14,2;" "
1130 PRINT AT 15,2;"   █
   █ "
1140 PRINT AT 16,2;"   █
   █ "
1150 PRINT AT 17,2;"   █
   █ "
1160 PRINT AT 18,2;"   █
   █ "
1170 PRINT AT 19,4;"████████████
█████████"
1175 FOR e=5 TO 25
1176 INK 4: PRINT AT m,e;"█"
1177 NEXT e
1178 IF m>17 THEN GO TO 1185
1179 LET m=m+1: GO TO 1175
1185 FOR e=12 TO 14
1187 INK 4: PRINT AT e,4;"█";AT
e,26;"█"
1188 NEXT e
1190 FOR e=7 TO 18
1192 PRINT AT e,15; INK 0;"█"
1194 NEXT e
1200 FOR x=12 TO 1 STEP -1
1210 LET z(x)=INT (10*RND)+8
1211 IF x=12 AND z(x)=13 THEN GO
 TO 1210
1220 NEXT x
1230 LET k=INT (3*RND)+1
1240 IF k=1 THEN LET K=12: LET k
(1)=13: LET k(2)=14
1420 IF k=2 THEN LET K=13: LET k
(1)=12: LET k(2)=14
1430 IF k=3 THEN LET K=14: LET k
(1)=12: LET k(2)=13
1450 RETURN
1451 PRINT
1500 PRINT AT 20,5;"falta"
1505 BEEP 2,-3
1520 NEXT n
1600 INK 0: PRINT AT f,c;"‡‡"; O
VER 1; INK 4;CHR$ 8;CHR$ 8;"██":
 INK 0
1610 PRINT AT 20,5;"intercepcion
 del balon"
1615 BEEP 2,0
1630 NEXT n
1800 PRINT AT 20,5;"saque de pue
rta"
1805 BEEP 2,-1
1820 NEXT n
1900 PRINT AT 20,5;"gran parada
del portero"
1905 INK 2: PRINT AT f,c;"‡"; OV
ER 1; INK 4;CHR$ 8;"█": INK 0
1906 BEEP 2,8
1920 NEXT n
1950 PRINT AT 20,5;"gran parada
del portero"
1960 INK 0: PRINT AT v(k),x;"‡";
 OVER 1; INK 4;CHR$ 8;"█": INK 0
: BEEP 2,2: NEXT n
2000 INK 0: PRINT AT f,c;"o"; OV
ER 1; INK 4;CHR$ 8;"█": INK 0
2010 LET a=a+1
2020 PRINT AT 20,5;"gol! gol! go
l! gooooool!!!"
2025 BEEP 4,16
2040 NEXT n
2100 INK 0: PRINT AT v(k),x;"o";
 OVER 1; INK 4;CHR$ 8;"█": INK 0
2105 LET b=b+1
2110 PRINT AT 20,5;"gol! gol! go
l! gooooool!!!"
2115 BEEP 4,16
2130 NEXT n
3000 CLS : BORDER 6
3010 PRINT AT 10,2;a$;"-";a;AT 1
1,2;b$;"-";b
3030 IF a=b THEN PRINT FLASH 1;A
T 0,10;"EMPATE": PRINT AT 21,0;"
JO TIO, ESTAS HECHO UN MARADONA"
3040 IF a>b THEN PRINT FLASH 1;A
T 0,4;"VENCEDOR  ";a$: PRINT AT
20,0;"EL SELECCIONADOR NACIONAL
AL    TELEFONO"
3050 IF a<b THEN PRINT FLASH 1;A
T 0,4;"VENCEDOR  ";b$: PRINT AT
20,0;"DEDICATE A OTRA COSA.LO TU
YO    NO ES EL FUTBOL"
3060 PAUSE 300: BORDER 7: GO TO
12
7000 FOR r=0 TO 7: READ m: POKE
USR "a"+r,m: NEXT r
7010 FOR r=0 TO 7: READ m: POKE
USR "b"+r,m: NEXT r
7020 FOR r=0 TO 7: READ m: POKE
USR "c"+r,m: NEXT r
7030 FOR r=0 TO 7: READ m: POKE
USR "f"+r,m: NEXT r
7040 FOR r=0 TO 7: READ m: POKE
USR "e"+r,m: NEXT r
7050 RETURN
7100 DATA 0,12,12,4,62,13,212,22
6
7110 DATA 0,48,48,32,124,176,43,
71
7120 DATA 0,24,24,16,127,40,80,4
0
7130 DATA 24,24,32,126,112,80,40
,80
7140 DATA 24,24,8,126,14,10,20,1
0
9000 BEEP .01,0: RETURN
```





REPASO DE GEOMETRIA



¿Qué tal están sus conocimientos de geometría? Con este programa se generan múltiples preguntas aleatorias con los que poner a prueba su habilidad matemática. Una ayuda, si le preguntan por el área de un círculo de radio 2 tiene dos posibilidades: decir 12, 75 ó 4 PI, que es más cómodo. El programa es muy sencillo aunque a Vd. no le sea tan sencillo responder. ¿Se atreve a ampliarlo?

(16K-Spectrum).

```
Cual es el area de un circulo
de radio 14
615.75216
O.K. La respuesta es 615.75216
```

```
  100 CLS
  110 LET q=1+INT (4*RND)
  120 LET m=2+INT (20*RND)
  130 GO SUB q*1000
  200 INPUT r: PRINT r
  210 IF ABS (r-a)>.5 THEN GO TO
240
```

```
  220 PRINT "O.K. La respuesta es
";a
  230 GO TO 250
  240 PRINT "Lo siento, la respue
sta era ";a
  250 INPUT "Presione <ENTER>";y$
  260 RUN
 1000 PRINT "Cual es la circunfer
encia de un circulo de radio ";m
 1010 LET a=2*PI*m
 1020 RETURN
 2000 PRINT "Cual es el area de u
n circulo   de radio ";m
 2010 LET a=PI*m*m
 2020 RETURN
 3000 PRINT "Cual es el area de u
na esfera   de radio ";m
 3010 LET a=(4/3)*PI*m*m
 3020 RETURN
 4000 PRINT "Cual es el volumen d
e una esferade radio ";m
 4010 LET a=PI*m*m*m/3
 4020 RETURN
```

GANE 5.000 PESETAS

MENSUALMENTE
PARTICIPANDO EN NUESTRO CONCURSO

ZX premiará mensualmente los programas que hagan llegar los lectores.

Para participar en este concurso abierto, todo aficionado a los ordenadores ZX81 y ZX Spectrum, deberá hacer llegar a la redacción de la revista el listado, un cassette y un texto explicativo.

Entre todos los programas que recibamos cada mes, serán seleccionados para su publicación aquellos que reúnan los siguientes criterios:
— Originalidad de la aplicación.
— Simplicidad del método de programación.

La única condición para participar en el concurso será que los programas no hayan sido publicados previamente en ninguna revista.



Y TAMBIEN...



UN ZX MICRODRIVE*
será sorteado cada mes entre todos los programas que recibamos, con independencia de que sean publicados o no.

***El ZX MICRODRIVE estará disponible en España, en principio, a partir de marzo de este año**

ATERRIZA COMO PUEDAS

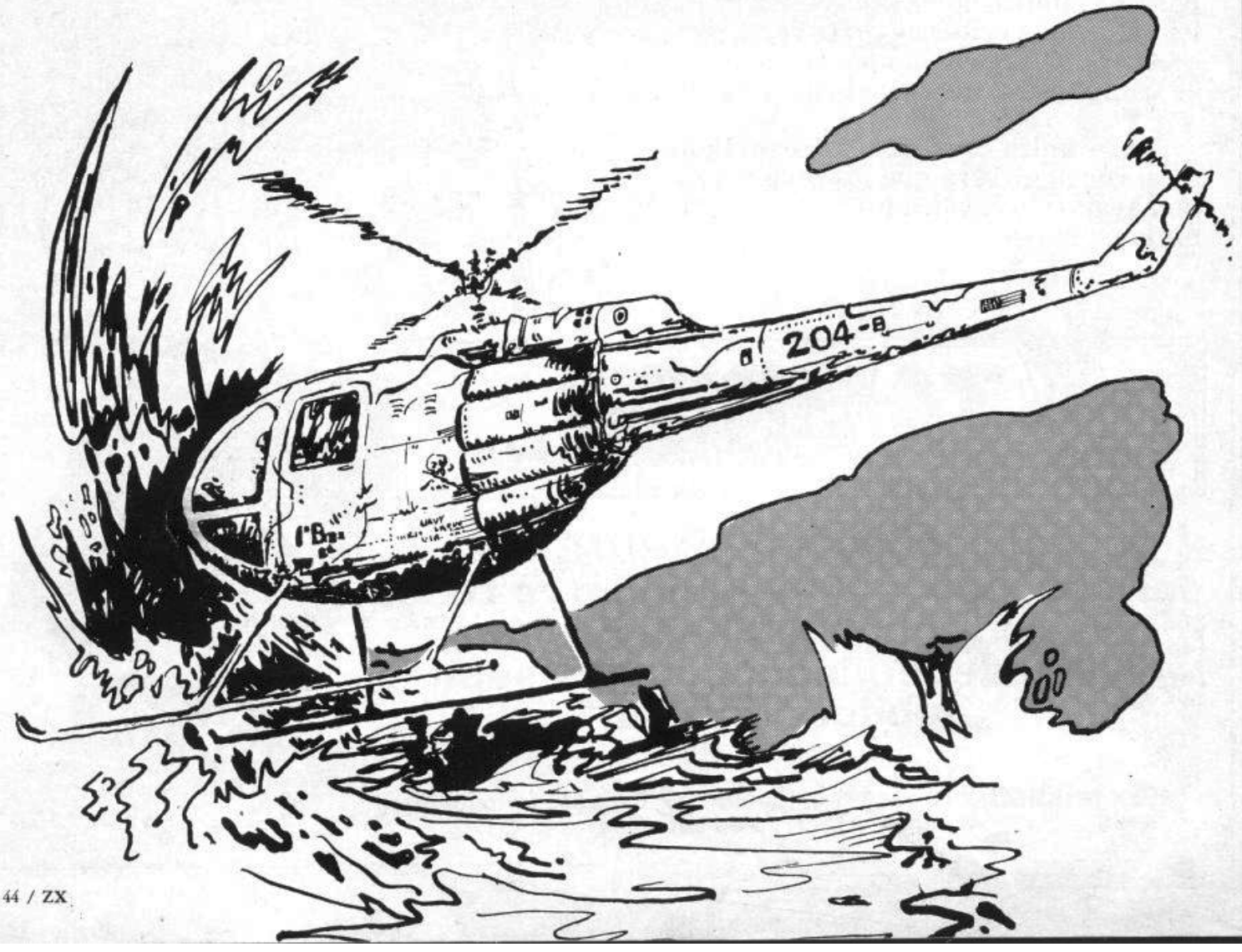
Lograr aterrizar con un helicóptero en un barco de pequeñas dimensiones es tarea difícil, más cuando queda poco combustible. No contará con la ayuda del radar y nadie le ayudará lo más mínimo. ¿Se atreve? ¡No le descontaremos el precio del helicóptero de su sueldo!

(1K-ZX81)

```
  10 LET L=VAL "5"
  20 LET C=VAL "25"
  30 LET A$=""
  40 FOR T=VAL "0" TO VAL "50"
  50 CLS
  55 LET D=INT (RND*4)+6
  60 PRINT AT 18,5;"----------
---";AT L+2,C+1;A$;AT L,C;"▄▄";
AT L-1,C-1,"--+--"
  70 IF A$="" THEN PRINT AT 18,D
;"▄▄▄▄"
  90 IF L=VAL "17" AND C=D+3 THE
N LET A$="0"
  95 IF L=VAL "18" THEN GOTO 300
```

```
 100 IF A$="0" AND L=VAL "17" TH
EN GOTO 500
 120 LET C=C+(INKEY$="8")-(INKEY
$="5")
 130 LET L=L+(INKEY$="6")-(INKEY
$="7")
 140 NEXT T
 200 PRINT "NO HAY MAS FUEL"
 300 PRINT "CRASH"
 310 STOP
 500 PRINT "ATERRIZAJE ";T
```

```
      --+--
        ▄▄
   ---▄▄▄▄▄▄--------
NO HAY MAS FUEL
CRASH
```



SOPA DE LETRAS

Seguro que alguna vez ha resuelto una "sopa de letras" cuando estaba aburrido. Normalmente se hacen en el lugar más imprevisto, y no siempre se tiene un Spectrum a mano, pero si ese no es su caso, ¡adelante! Puede jugarlo con su chico(a) favorito(a) cuando no se le ocurra otra cosa.

Su Spectrum le guiará al principio. Le preguntará cuántas palabras desea buscar y cuáles son éstas. Después se crea el puzle, que tarda un poco, dependiendo de cuántas palabras haya introducido. Cuando finalice, le preguntará si desea crear usted el puzle, lo cual es más sencillo a la hora de la búsqueda. Cuando aparecen las letras verá un cuadrado brillante, con el que se señala la posición inicial. Con las teclas de desplazamiento (5, 6, 7, 8) ha de colocarlo donde esté el inicio de una palabra, de las introducidas previamente. Después pulse "w" y conteste la palabra localizada (horizontal, vertical o diagonal). Si es correcto, obtendrá un punto; si no, una señal acústica de error. Si quiere retirarse, pulse "r".

Es un juego francamente bonito y bien realizado. Un único detalle le falta: señalar en la "sopa de letras" las palabras encontradas para simplificar la búsqueda.

(16K Spectrum).

```
  1 PRINT (PEEK 23730+256*PEEK
23731)-(PEEK 23653+256*PEEK 2365
4)
 10 REM Crucigrama

 20 GO SUB 1000
 30 GO SUB 3000
 40 GO SUB 5000
 50 GO SUB 6000
 60 GO SUB 7000
1000 DIM l$(10,10)
1005 DIM c(10)
1006 LET list=0
1010 INPUT "Cuantas palabras ";w
1020 IF w<2 OR w>10 THEN BEEP .1
,10: GO TO 1010
1040 FOR i=1 TO w
1050 PRINT AT 5+i,5;"palabra num
ero ";i;"=";
1060 INPUT w$
1070 IF LEN w$>10 THEN INPUT "Ma
ximo 10 letras!";w$: GO TO 1070
1080 IF LEN w$<1 THEN GO TO 1060
1090 PRINT w$
1100 LET l$(i)=w$
1110 LET c(i)=LEN w$
1120 NEXT i
1130 RETURN
2000 LET m=0: LET j=0
2010 FOR z=1 TO w
2020 IF m<c(z) THEN LET m=c(z):
LET j=z
2030 NEXT z
2040 RETURN
3000 DIM d$(15,15)
3010 CLS
3080 FOR i=1 TO w
3090 GO SUB 2000
3100 LET l=c(j)
3110 LET c(j)=0
3120 LET x=INT (RND*(15-l))+1
3130 LET y=INT (RND*(15-l))+1
3140 LET v=INT (RND*3)-1
3150 LET u=INT (RND*3)-1
3155 IF u=0 AND v=0 THEN GO TO 3
140
3160 LET a=x: LET b=y
3165 IF v<0 THEN LET a=a+l
3166 IF u<0 THEN LET b=b+l
3170 FOR k=1 TO l
3180 IF d$(a,b)<>" " AND d$(a,b)
<>l$(j,k) THEN GO TO 3120
3190 LET a=a+v
3200 LET b=b+u
3210 NEXT k
3300 PRINT "En proceso"
3310 LET a=x: LET b=y
3315 IF v<0 THEN LET a=a+l
3316 IF u<0 THEN LET b=b+l
3320 FOR k=1 TO l
3330 LET d$(a,b)=l$(j,k)
3340 LET a=a+v
3350 LET b=b+u
3360 NEXT k
3370 NEXT i
3380 RETURN
5000 CLS
5005 FOR m=1 TO 15
5010 FOR n=1 TO 15
5020 IF d$(m,n)=" " THEN PRINT "
.";
5030 IF d$(m,n)<>" " THEN PRINT
d$(m,n);
5040 NEXT n
5050 IF list=0 OR m>10 THEN PRIN
T
5055 IF list=1 AND m<=10 THEN PR
INT TAB 20;l$(m)
5060 NEXT m
5070 RETURN
6000 INPUT "Organiza el puzle ";
a$
6010 IF LEN a$=0 THEN GO TO 6000
6020 IF a$(1)="s" THEN GO SUB 50
00
6030 FOR i=1 TO 15
6040 FOR j=1 TO 15
6050 IF d$(i,j)=" " THEN LET d$(
i,j)=CHR$ (INT (RND*26)+97)
6060 NEXT j
6070 NEXT i
6080 INPUT "Palabras al lado del
 puzle ";a$
6090 IF LEN a$=0 THEN GO TO 6080
6100 IF a$(1)="s" THEN LET list=
1
6110 GO SUB 5000
6120 RETURN
7000 LET x=0
7010 LET y=0
7020 PRINT AT y,x; BRIGHT 1;SCRE
EN$ (y,x);
7040 LET a$=INKEY$
7045 IF INKEY$="" THEN GO TO 704
0
7050 IF a$="w" THEN GO TO 7500
7055 PRINT AT y,x; BRIGHT 0;SCRE
EN$ (y,x);
7060 IF a$="5" AND x>0 THEN LET
x=x-1
7070 IF a$="8" AND x<14 THEN LET
 x=x+1
7080 IF a$="6" AND y<14 THEN LET
 y=y+1
7090 IF a$="7" AND y>0 THEN LET
y=y-1
7095 IF a$="r" THEN GO SUB 9000
7100 PRINT AT y,x; BRIGHT 1;SCRE
EN$ (y,x);
7110 GO TO 7040
7500 INPUT "Cual es la palabra "
;w$
7510 IF LEN w$=0 OR LEN w$>15 TH
EN BEEP .5,20: GO TO 7500
```

EL PISTOLERO

MALO

¡Afina tu puntería forastero! ZX City es una ciudad sin ley donde sólo el más rápido puede sobrevivir. Todo lo que tienes que hacer es disparar a las cajas sin fallar, pues ello te ocasionaría perder puntos. Tu máxima puntuación es 100 y el premio es la vida, ¿está claro forastero?

(1 K-ZX81)



```
7515 GO SUB 7800
7516 IF match=0 THEN BEEP .5,10:
 GO TO 7040
7520 FOR u=-1 TO 1
7530 FOR v=-1 TO 1
7540 IF u=0 AND v=0 THEN GO TO 7
570
7550 GO SUB 7900
7560 IF match=1 THEN GO TO 7600
7570 NEXT v
7580 NEXT u
7590 BEEP .5,-20
7595 GO TO 7040
7600 LET score=score+1
7605 PRINT AT 19,0;"Puntos= ";sc
ore;" "
7610 BEEP .2,20
7620 LET l$(word)=" "
7630 IF score=w THEN GO TO 9500
7640 GO TO 7040
7800 LET match=0
7810 FOR i=1 TO w
7820 IF w$+"          "( TO 10-L
EN w$)=l$(i) THEN LET match=1: L
ET word=i: RETURN
7830 NEXT i
7840 RETURN
7900 LET match=0
7910 LET a=x+1
7920 LET b=y+1
7930 FOR i=1 TO LEN w$
7940 IF w$(i)<>d$(b,a) THEN RETU
RN
7950 LET a=a+u
7960 LET b=b+v
7970 IF a<1 OR a>15 THEN RETURN
7980 IF b<1 OR b>15 THEN RETURN
7990 NEXT i
7995 LET match=1
7996 RETURN
8000 POKE USR "a"+0,BIN 00000000
8010 POKE USR "a"+1,BIN 00000000
8020 POKE USR "a"+2,BIN 00000000
8030 POKE USR "a"+3,BIN 00011000
8040 POKE USR "a"+4,BIN 00011000
8050 POKE USR "a"+5,BIN 00000000
8060 POKE USR "a"+6,BIN 00000000
8070 POKE USR "a"+7,BIN 00000000
8080 LET score=0
8090 RETURN
9000 INPUT "Seguro que no puede"
;"encontrar mas palabras ";a$
9010 IF a$(1)<>"s" THEN RETURN
9020 CLS
9030 PRINT AT 8,10;"Puntos= ";sc
ore
9040 INPUT "Otra partida (s/n) "
;a$
9050 IF a$="s" THEN RUN
9060 IF a$<>"n" THEN GO TO 9040
9070 STOP
9500 PRINT AT 18,0;"Tomo todas!!
"
9510 GO TO 9040
```

```
      xauyhhecrkdnttq      ce
      xlninjlmevwnecc      vq
      otwbmidntvvcuep
      mdudmreaekfhzom
      giuwvrwqiqedsap
      lnkyesraafgtbdh
      qochvonzehndejj
      wxrgppwkwqhgaqy
      dhgplzuigwzqmqg
      cdbkuwzehrxddiy
      kngfnvjbhyvlunj
      uzmsaoyoykiqhaf
      aqzjqupjgnjvfel
      byoqaitahuaprvu
      tticcgfbnslvlmd


      Puntos= 1
```

```
 10 LET Z=100
 20 FOR B=1 TO 10
 30 PRINT "
 40 NEXT B
 50 FOR B=0 TO 28
 60 PRINT AT 6,B;"  ,="
 70 IF INKEY$<>"" THEN GOTO 110
 80 NEXT B
 90 LET Z=Z-5
100 GOTO 150
110 FOR D=5 TO 0 STEP -1
120 PRINT AT D,B+1;"=";AT D,B+1
;" "
130 NEXT D
140 LET Z=Z-2
150 PRINT AT 6,B;"     "
160 FOR B=0 TO 29
170 PRINT AT 0,B;
180 IF PEEK (PEEK 16398+256*PEE
K 16399)<>0 THEN GOTO 50
190 NEXT B
```

```
195 PRINT AT 0,0;"PUNTOS ";Z;
200 PRINT AT 10,1;"FIN#OTRO JUE
GO? PULSA UNA TECLA"
210 PAUSE 40000
220 CLS
230 RUN
```



DIAMANTES

El sueño de su vida: su gran diamante. Todos lo codician y las expediciones ya han partido. Sólo tiene que adelantarse, ascender a través de los obstáculos, guardianes, fantasmas... y todo lo que su ZX le pondrá por medio. Puede ir a la izquierda (5), saltar a la izquierda (1), a la derecha (8), saltar a la derecha (0) y ascender (7).

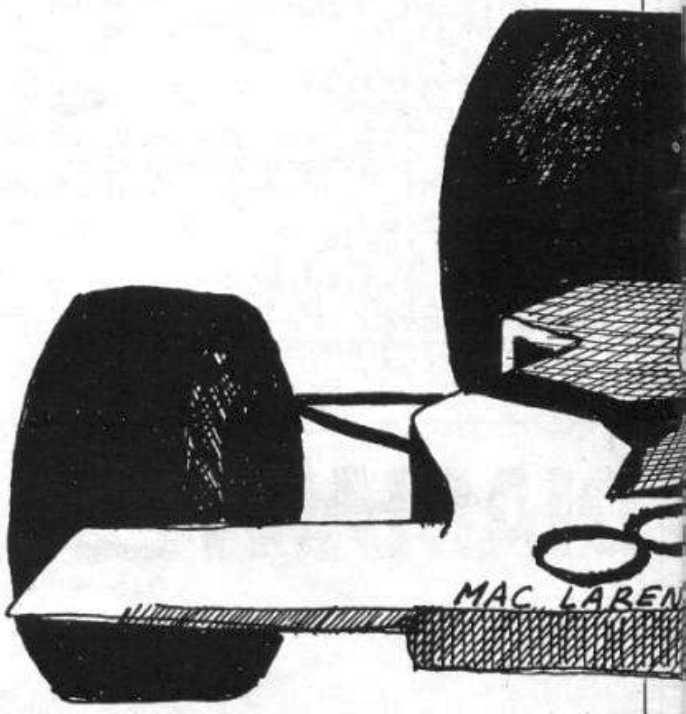
Apresúrese o no llegará a tiempo.

(16 K-ZX81)

```
 1 LET C=2
 2 LET F=9
 3 LET I=6
 4 LET H=22
 5 LET G=16
 6 LET SC=0
 7 CLS
 8 PRINT AT 0,0;"
 9 PRINT AT 21,0;"
10 FOR A=1 TO 20
11 PRINT AT A,0;"";AT A,31;"
"
12 NEXT A
13 PRINT AT 4,24;"
14 PRINT AT 17,1;"
"
15 PRINT AT 20,1;"
"
16 PRINT AT 18,1;"
"
17 PRINT AT 13,4;"
"
18 PRINT AT 14,4;"
"
19 PRINT AT 16,27;"H";AT 13,27
;"H";AT 15,27;"H";AT 14,27;"H"
20 PRINT AT 9,6;"
21 FOR A=9 TO 12
22 PRINT AT A,6;"H"
23 NEXT A
24 PRINT AT 9,20;"
25 FOR A=5 TO 8
26 PRINT AT A,21;"H"
27 NEXT A
28 PRINT AT 5,22;"";AT
6,22;"
29 FOR A=5 TO 16
30 PRINT AT A,29;"H"
31 NEXT A
32 LET A$=INKEY$
33 LET C=C+(A$="8")-(A$="5")
34 LET SC=SC+1
35 PRINT AT G,C;"";AT G,C-1;"
```

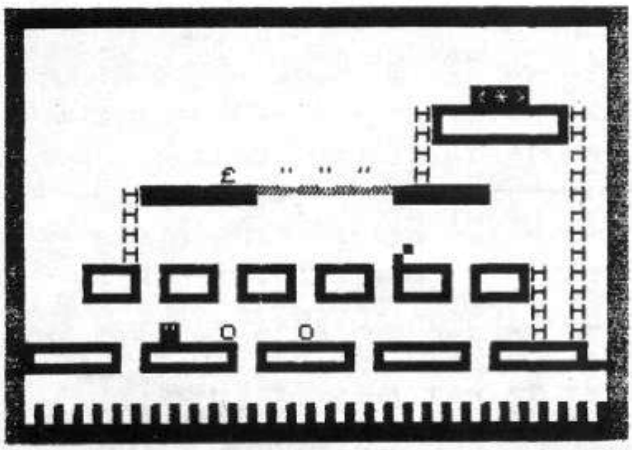
```
";AT G,C+1;" ";AT 16,27;"H"
36 PRINT AT 9,13;"";AT
8,14;"" "" ""
37 IF A$="6" AND C=29 AND G=4
THEN GOTO 7000
38 LET F=F+1
39 IF A$="7" AND C=27 AND C=16
THEN GOTO 87
40 IF A$="7" AND G=12 AND C=6
THEN GOTO 90.
41 LET H=H-1
42 IF G=4 AND C=24 THEN PRINT
AT 4,23;"  "
43 LET I=I+1
44 IF A$="1" THEN LET C=C-2
45 IF G=4 AND C=29 THEN GOTO 9
8
46 IF A$="1" THEN PRINT AT G,C
+2;"
"
47 IF A$="0" THEN LET C=C+2
48 IF A$="0" THEN PRINT AT G,C
-2;"
"
49 PRINT AT 8,I;"£";AT 8,I-1;"
"
50 IF I=12 THEN LET I=6
51 PRINT AT 16,H;"0   0";AT 16
,H+5;"
"
52 PRINT AT 12,7;"
"
53 PRINT AT 12,F;"";AT 12,F-1
;"
"
54 IF F> 24 THEN LET F=9
55 IF F=C AND G=12 THEN GOTO 7
7
56 IF G=8 AND C=14 OR C=16 AND
G=8 OR C=18 AND G=8 THEN GOTO 7
7
57 IF H=1 THEN LET H=22
59 IF C=5 AND G=8 THEN GOTO 77
60 IF C=12 AND G=16 THEN GOTO
77
61 IF C=18 AND G=16 THEN GOTO
77
62 IF A$="7" AND C=21 AND G=8
THEN GOTO 94
63 IF C=24 AND G=16 THEN GOTO
77
```

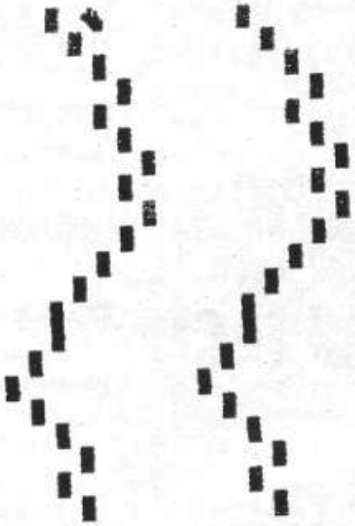
```
  64 IF C=7 AND G=12 THEN GOTO 7
7
  65 IF C=11 AND G=12 THEN GOTO
77
  66 IF C=15 AND G=12 THEN GOTO
77
  67 IF C=19 AND G=12 THEN GOTO
77
  68 IF C=6 AND G=16 THEN GOTO 7
7
  69 IF G=12 AND C=23 OR C=28 AN
D G=12 THEN GOTO 77
  70 IF C=H+4 AND G=16 THEN GOTO
 77
  71 IF C=H AND G=16 THEN GOTO 7
7
  73 IF C=I AND G=8 THEN GOTO 77
  74 PRINT AT 12,F-2;" ";AT 8,12
;" ";AT 16,5;" ";AT 16,1;" "
  75 PRINT AT 12,24;" "
  76 GOTO 32
  77 FOR F=G TO 19
  78 PRINT AT F,C;"■";AT F,C;" "
  79 NEXT F
  80 PRINT AT F,C;""""
  81 PRINT AT 0,0;"MAL-PULSA UNA
 TECLA PARA EMPEZAR"
  82 PRINT AT 21,0;"  PULSA S PA
RA TERMINAR"
  83 INPUT B$
  84 IF B$="S" THEN STOP
  85 IF B$<>"S" THEN GOTO 1
  86 STOP
  87 PRINT AT 12,27;"■"
  88 LET G=12
  89 GOTO 32
  90 PRINT AT 8,6;"■"
  91 PRINT AT 12,6;"H"
  92 LET G=8
  93 GOTO 32
  94 PRINT AT 4,21;"■"
  95 PRINT AT 8,21;"H"
  96 LET G=4
  97 GOTO 32
  98 PRINT AT 16,C;"■";AT 4,29;"
"
 101 PRINT AT 0,0;" BIEN,MISION
CUMPLIDA"
 103 PRINT AT 21,0;"TIEMPO INVER
TIDO  ";SC;" PUNTOS ";100-SC
 104 INPUT B$
 105 IF B$="" THEN GOTO 1
 106 IF B$<>"" THEN STOP
```

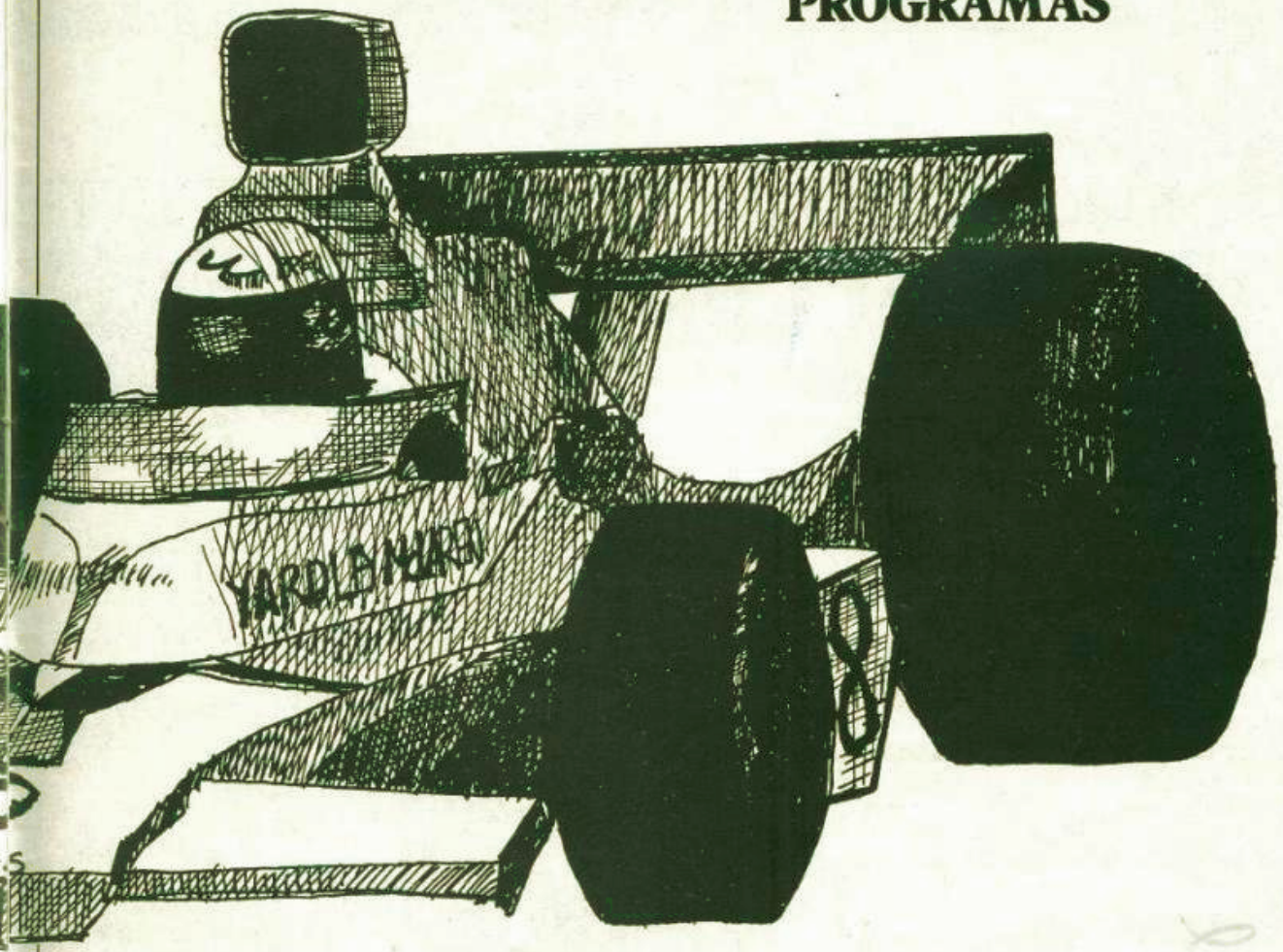


GRAND PRIX

No se preocupe por el coche (lo pone su Spectrum), o por el carnet de conducir (nadie se lo pedirá). Dispone de cinco tipos de circuitos, no demasiado largos, pero, según la opción elegida, puede resultar demasiado estrechos. En vez de volante, tiene dos mandos: uno, para la izquierda ("5"), y otro, para la derecha ("8"). No necesita







más para "pegársela". No se preocupe por su adversario, su único problema es llegar al final, cosa bastante improbable si selecciona el nivel cuatro o cinco.

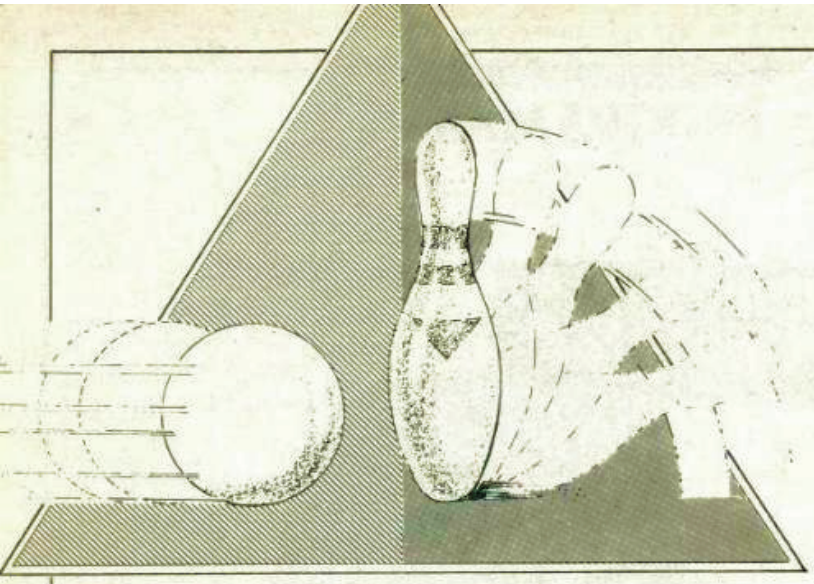
(16K Spectrum)

Notas gráficas

(320,1140 Graphic 5
(330,1220 Graphic B

```
  5 REM Carreras.
 20 GO SUB 7000
500 POKE USR "b"+0,BIN 00100000
510 POKE USR "b"+1,BIN 00101000
520 POKE USR "b"+2,BIN 11101000
530 POKE USR "b"+3,BIN 11111100
540 POKE USR "b"+4,BIN 01111110
550 POKE USR "b"+5,BIN 00111110
560 POKE USR "b"+6,BIN 00011110
570 POKE USR "b"+7,BIN 00001110
800 LET yb=1
1000 REM Pista.
1010 PAPER 7: INK 7
1020 CLS
1050 LET x=(RND*10)+5
1060 LET xb=x+2
1100 FOR y=1 TO 20
1140 PRINT AT y,x; INK 0;" ";AT
y,x+d;" "
1150 LET x=x+(SGN (RND-.5))
1160 IF x>31 THEN LET x=31
1170 IF x<0 THEN LET x=0
1200 NEXT y
1220 PRINT AT yb,xb; INK 2;"b"
1230 PAUSE 50
1250 FOR z=1 TO 20
1260 LET x=x+(SGN (RND-.5))
1270 IF x>31 THEN LET x=31
1280 IF x<0 THEN LET x=0
```

```
1310 POKE 23692,41
1320 PRINT AT 21,x; INK 0;" ";AT
 21,x+d;" ";
2000 GO SUB 5000
2400 NEXT z
3000 FOR z=1 TO 20
3100 PRINT AT 21,1;
3150 GO SUB 5000
3200 NEXT z
3500 GO SUB 8000
5000 REM Movimiento.
5100 LET a$=INKEY$
5150 PRINT AT yb-1,xb;" "
5200 IF a$="5" THEN LET xb=xb-1
5210 IF a$="8" THEN LET xb=xb+1
5320 IF ATTR (yb,xb)=56 THEN GO
TO 8500
5330 PRINT AT yb,xb; INK 2;"b"
5500 RETURN
7000 INK 0
7010 PAPER 7
7020 PRINT AT 1,5;"C A R R E R A
 S"
7030 PRINT AT 5,0;"Conduzca su c
oche a traves"
7040 PRINT "de un curso peligros
o."
7050 INPUT "Nivel dificultad ";
De 1 (facil) a 5 (Dificil) ";d
7060 IF d<1 OR d>5 THEN GO TO 70
50
7070 LET d=9-d
7080 RETURN
8000 PRINT AT 20,1; INK 0;"Felic
idades, lo logro!!"
8100 GO TO 9000
8500 PRINT AT 20,1; INK 0;"Choco
!!!"
8510 BEEP 1,-10
9000 INPUT "Otra vez (s/n) ";b$
9020 IF b$(1)="s" THEN RUN
9500 INK 0
9510 PAPER 7
9520 CLS
```



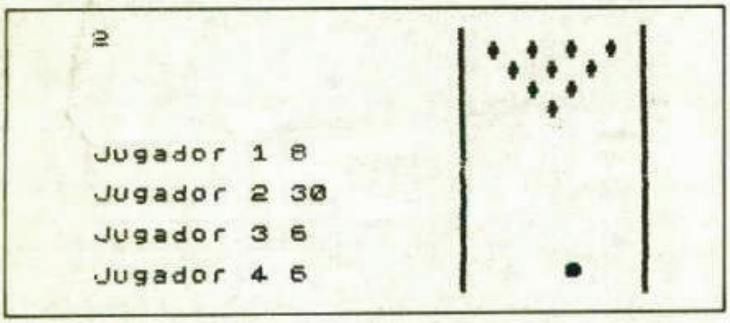
BOLERA

Quizás ya haya visto otros juegos de bolos, pero le aseguramos que éste es mejor. Le daremos unas cuantas razones: tiene una estupenda resolución gráfica; pueda dirigir la bola una vez lanzada; puede competir con sus amigos... Para ello, sólo ha de introducir este programa que no resulta excesivamente largo y merece la pena.

El número de jugadores puede ser de 1, 2, 3 ó 4. En el margen superior izquierdo se muestra el número de la partida y debajo la puntuación de cada jugador. Para lanzar la bola presione el "1" y para dirigirlo el "5" y el "8". No se preocupe, no necesita tomar carrera ni anotar su puntuación. ¿Se puede pedir más?

(16K-Spectrum).

Notas gráficas:
Línea 50 - Graphic b.
Línea 60 - Graphic b.
Línea 80 - Graphic b.
Línea 200 - Graphic b.
Línea 8020 - Graphic a.
Línea 8030 - Graphic b.




```
  1 REM Bolos
  3 DIM s(5)
  5 FOR f=0 TO 7: READ a: POKE
USR "a"+f,a: NEXT f
 10 FOR f=0 TO 7: READ a: POKE
USR "b"+f,a: NEXT f
 15 LET x=20: LET k=19
 20 GO SUB 7500
 25 FOR q=1 TO 10
 30 FOR p=1 TO pl
 32 RESTORE 9010+p: READ b,c: B
ORDER b: PAPER c: CLS
 35 GO SUB 8010
 40 FOR t=1 TO 2
 50 FOR k=18 TO 26: PRINT AT x,
k;" ●": FOR f=1 TO 3: GO SUB 70:
 NEXT f: NEXT k
 60 FOR k=27 TO 19 STEP -1: FOR
 f=1 TO 3: GO SUB 70: NEXT f: PR
INT AT x,k;"●  ": NEXT k: GO TO 5
0
 70 IF INKEY$<>"1" THEN RETURN
 80 FOR f=x TO 0 STEP -1: PRINT
 AT f,k;"●": IF f>=12 THEN PRINT
 AT f+1,19;"
                "
 85 IF f=5 AND k=22 AND t=1 AND
 RND<.5 THEN GO TO 200
 95 PRINT AT f+1,k-1;"    ": IF
f>=12 THEN LET k=k-(INKEY$="5")+
(INKEY$="8"): IF k=18 OR k=28 TH
EN GO TO 110: NEXT f
100 FOR n=1 TO 4: NEXT n: NEXT
f
105 GO TO 130
120 PRINT AT 16,4;"Fuera": FOR
d=1 TO 130: NEXT d: PRINT AT 16,
4;"
"
130 NEXT t
135 GO SUB 6000
140 FOR d=1 TO 130: NEXT d
145 NEXT p
150 NEXT q
160 CLS : FOR d=2 TO pl*2 STEP
2: PRINT AT d+5,6;"Jugador ";d/2
;" Puntos= ";s(d/2): NEXT d: PRI
NT AT 1,10;"Fin de juego"
170 INPUT "Seguimos (s/n) ";a$
180 IF a$="s" OR a$="S" THEN RU
N
190 STOP
200 FOR f=5 TO 0 STEP -1: PRINT
 AT f,19;"  ";"●";"  ";AT f+1
,22;"  ": BEEP .5,f: NEXT f: LET
 s(p)=s(p)+30: GO TO 140
6000 REM ****************
6005 LET s(5)=0
6010 RESTORE 9010
6020 FOR u=1 TO 10: READ a,b: LE
T s(5)=s(5)+(ATTR (a,b)<90): NEX
T u
6025 IF s(5)=10 THEN LET s(5)=15
6030 LET s(p)=s(p)+s(5)
6040 FOR u=2 TO pl*2 STEP 2: PRI
NT AT u+5,10;s(u/2): BEEP .01,pl
*u: NEXT u
6050 RETURN
7500 PRINT AT 6,8;"10 Jugadas":
PLOT 64,119: DRAW 78,0: PRINT AT
 8,3;"Para uno, dos, tres o cuat
ro     jugadores."
7510 PAUSE 40: PRINT AT 10,3;"Si
 derriba todos los bolos      co
nsigue un pleno que son 30     pu
ntos, sino un punto por bo-    lo
 derribado."
7515 PRINT TAB 3
7520 PRINT TAB 3;"Presione ""1""
 para disparar";TAB 3;"Puede gui
ar su disparo con el  ""5 y 8"""
7530 PAUSE 30: INPUT "Cuantos ju
gadores 1-4 ";pl: IF pl<1 OR pl>
4 THEN GO TO 7530
7540 BORDER 1: PAPER 5: INK 0: C
LS : RETURN
8000 REM ****************
8010 RESTORE 9010: PLOT 150,15:
DRAW 0,150: PLOT 151,15: DRAW 0,
150: PLOT 224,15: DRAW 0,150: PL
OT 225,15: DRAW 0,150
8020 FOR f=1 TO 10: READ a,b: PR
INT BRIGHT 1;AT a,b;"♦": NEXT f
8030 PRINT ;AT x,k;"●"
8040 PRINT AT 1,0; FLASH 1;q; FL
ASH 0: FOR f=2 TO pl*2 STEP 2: P
RINT AT f+5,0;"Jugador ";f/2;" "
;s(f/2): NEXT f
8050 RETURN
9000 DATA BIN 00011000,BIN 00011
000,BIN 00111100,BIN 00111100,BI
N 00111100,BIN 00011000,BIN 0001
1000,BIN 00000000
9005 DATA BIN 00000000,BIN 00111
100,BIN 01111110,BIN 01111110,BI
N 01111110,BIN 01111110,BIN 0011
1100,BIN 00000000
9010 DATA 2,20,2,22,2,24,2,26,3,
21,3,23,3,25,4,22,4,24,5,23
9020 DATA 1,5
9021 DATA 2,6
9022 DATA 6,4
9023 DATA 0,7
```

OFERTA DE INTRODUCCION

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

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

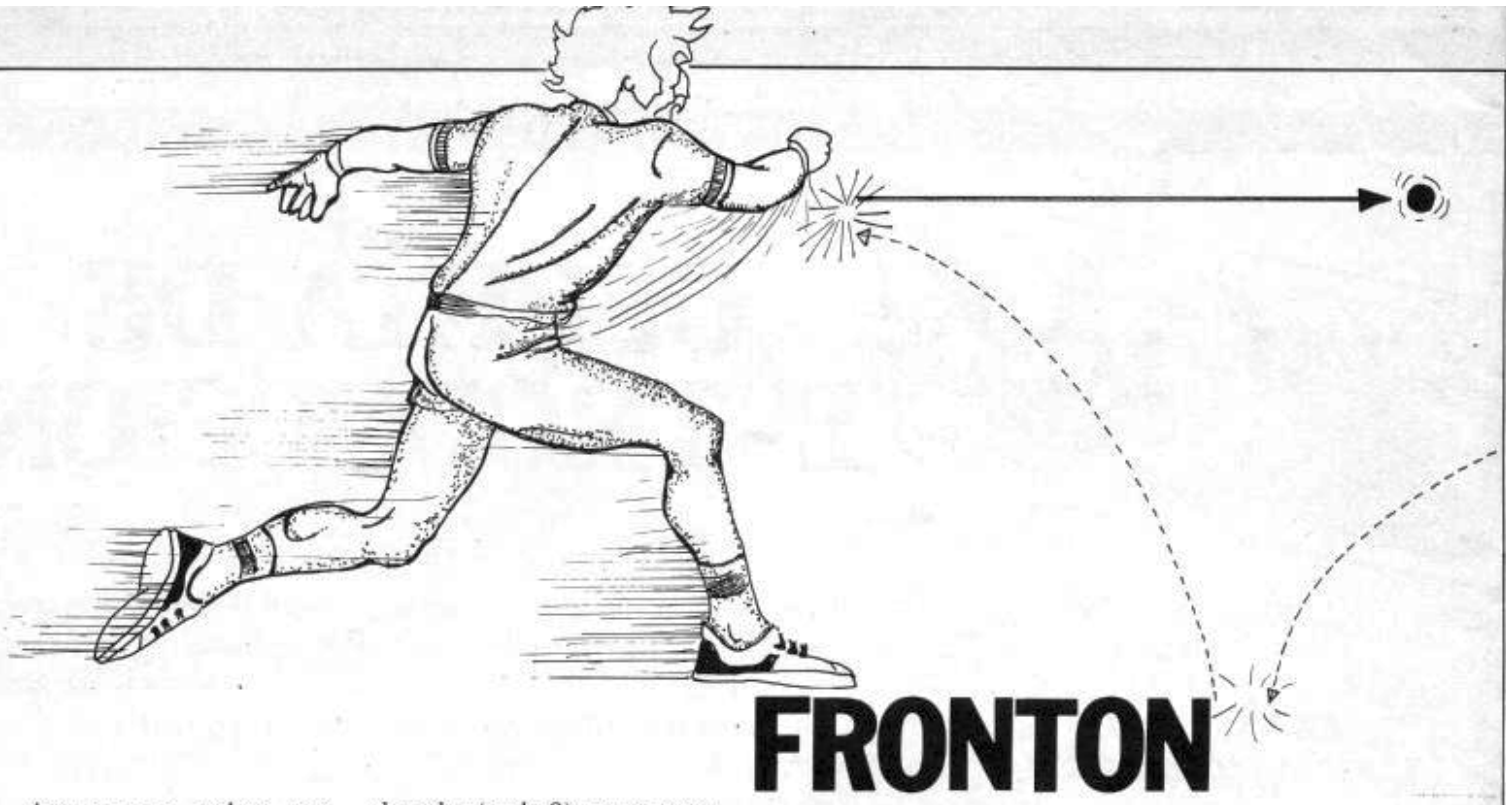
GRATIS PARA USTED si se suscribe a ZX.

Una obra imprescindible en la biblioteca de todo poseedor de un ordenador personal SINCLAIR Spectrum. **Cómo programar su Spectrum,** un regalo de 132 páginas, tamaño 210 × 270 mm., cuyo precio de venta es de 850 Pts.



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

Jerez, 3
Tel. 457 26 17
MADRID-16



FRONTON

Acertar una pelota que viene desde el fondo de su televisor puede parecer fácil cuando usted se encuentra al otro extremo. Así empieza este juego de pelota a tres bandas, pero, a medida que transcurre el tiempo, la distancia entre usted y la pelota se acortará, y su Spectrum le pondrá en apuros si desea acertarlas todas. Puede correr de izquierda (tecla 5) a derecha (tecla 8), pero no se canse demasiado. Tiene diez bolas, y en pantalla se le informa constantemente del número de bola y puntuación obtenida. ¿No le tienta echar una partidita?

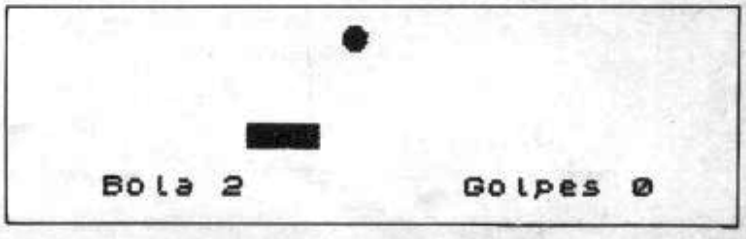
(16K Spectrum).

Notas gráficas

Línea 360 Graphic A
" 110 Shift 888

```
    1 REM Fronton.
    5 BORDER 1: PAPER 7: INK 0
   10 LET h=0: LET ht=0: LET d=19
: LET ball=0
   15 GO SUB 800: GO SUB 500
   20 LET ball=ball+1
   30 IF ball>10 THEN GO TO 2000
   40 LET a=5
   50 LET b=5
   60 LET v=1
   70 LET w=1
   80 LET x=10: LET y=d
   90 INPUT " "

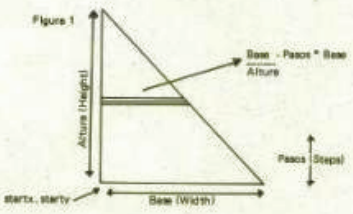
  100 PRINT AT 21,5;"Bola ";ball;
  110 PRINT AT y,x; PAPER 8; INK
0;"   "
  120 PRINT AT 21,20;"Golpes ";ht
  130 GO SUB 300
  140 GO SUB 200
  150 IF b+w<>y THEN LET y=d: GO
TO 110
  160 BEEP 1,0
  170 PRINT AT y,x; PAPER 8;"
": REM 5 espacios.
  180 PRINT AT b,a; PAPER 8;" "
  190 IF d<19 THEN LET d=d+1
  194 LET h=0
  195 GO TO 20
  200 LET a$=INKEY$
  210 IF a$="5" AND x>0 THEN LET
x=x-1
  220 IF a$="8" AND x<27 THEN LET
 x=x+1
  230 RETURN
  300 PRINT AT b,a; PAPER 8; INK
8;"  "
  310 LET a=a+v
  320 LET b=b+w
  330 IF a=30 OR a=1 THEN LET v=-
v: BEEP .1,15
  340 IF b=1 THEN LET w=-w: BEEP
.1,15
  350 IF b+w=y THEN GO TO 400
  360 PRINT AT b,a; PAPER 8;"●"
  370 RETURN
  400 LET r=a-x
  410 IF r<1 OR r>3 THEN GO TO 36
0
  420 LET w=-w
  430 BEEP .1,15
  440 LET h=h+1: LET ht=ht+1
  450 IF h<>2 THEN GO TO 360
  460 LET h=0: LET d=d-1
  470 PRINT AT y,x; PAPER 8;"
": REM 5 espaciosm
  490 GO TO 300
  500 FOR i=0 TO 31
  510 PRINT AT 0,i; PAPER 0;" "
  520 NEXT i
  530 FOR i=0 TO 20
  540 PRINT AT i,0; PAPER 0;" ";A
T i,31;" "
  550 NEXT i
  560 RETURN
  800 PRINT AT 7,0;
  810 LET c=2: GO SUB 850
  820 LET c=6: GO SUB 850
  830 GO SUB 930
  840 RETURN
  850 FOR i=1 TO 7*32
  860 PRINT PAPER c;" ";
  870 NEXT i
  880 RETURN
  930 POKE USR "a"+0,BIN 00111100
  940 POKE USR "a"+1,BIN 01111110
  950 POKE USR "a"+2,BIN 11111111
  960 POKE USR "a"+3,BIN 11111111
  970 POKE USR "a"+4,BIN 11111111
  980 POKE USR "a"+5,BIN 11111111
  990 POKE USR "a"+6,BIN 01111110
 1000 POKE USR "a"+7,BIN 00111100
 1010 PAPER 7
 1020 RETURN
 2000 INK 0
 2010 PAPER 7
 2030 CLS
 2040 PRINT AT 10,10;"Puntos "
 2050 PRINT AT 12,14;ht
 2060 INPUT "Otra (s/n) ";a$
 2070 IF a$(1)="s" THEN RUN
 2080 CLS : BORDER 7
 2090 PRINT "O.K."
```



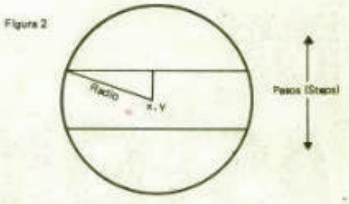
¡PINTELO DE NEGRO!

Como Vd. sabe, su Spectrum tiene potentes instrucciones de graficación, como el PLOT, DRAW y CIRCLE. Las tres se utilizan en los programas que le ofrecemos a continuación. Lo que no existe es una instrucción que sirva para "rellenar" el área de las figuras geométricas que es lo que hace este programa.

La figura 1 muestra un triángulo. Su origen está en el punto x, y a partir del cual se construyen los catetos conociendo la altura y la base. A partir de ahí se puede generar la hipotenusa. El trazado se realiza en múltiples líneas horizontales de acuerdo con la fórmula que en dicha figura aparece.



La figura 2 muestra una circunferencia y los datos que se requieren para su dibujo, tal y como hace el segundo programa que, igualmente, lo "rellena" de negro. Parte también de un punto inicial x, y requiere la información sobre el radio para su dibujo.
(16K-Spectrum).





```
  10 REM Triangulos.
  20 INPUT "Coordenadas de x e y
";startx,starty
  30 INPUT "Anchura y altura";Wi
dth,height
  40 LET steps=height
  50 PLOT startx,starty+steps
  60 GO TO 100
  70 REM Trazado lineas.
  80 PLOT startx,starty+steps
  90 DRAW Width-steps/height*Wid
th,0
 100 IF NOT steps THEN STOP
 110 LET steps=steps-(steps>0)+(
steps<0)
 120 GO TO 80
```



```
  10 REM Circulos.
  20 INPUT "Coordenadas de x e y
";x,y
  30 INPUT "radio ";s
  40 LET s=ABS INT s
  50 IF NOT s THEN STOP
  60 LET rad2=s*s
  80 LET s=s-1
  90 IF s<0 THEN STOP
 100 LET xl=EXP (1/2*LN (rad2-s*
s))-1
 110 LET line=xl+xl
 120 LET sl=x-xl
 130 IF sl<0 THEN LET line=line+
sl: LET sl=0
 150 PLOT sl,y+s: DRAW line,0
 170 IF y-s<0 THEN GO TO 80
 180 PLOT sl,y-s: DRAW line,0
 190 GO TO 80
```

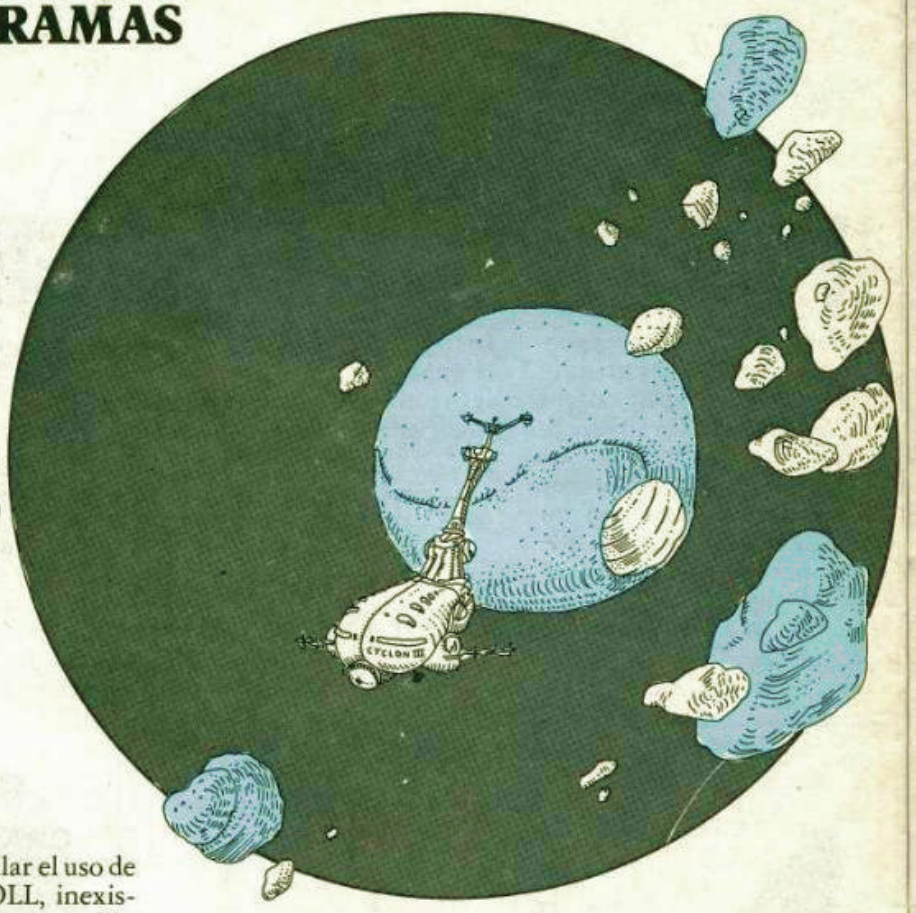



CIRCUITO IMPRESO & COMPUTER

* * *

ya está a la venta el nº 51

OTRO JUEGO DE ASTEROIDES



Transcurría el año 3027 cuando apareció la primera tormenta de asteroides. Desde entonces, lo que pareció una novedad se tornó en pesadilla y los tripulantes de la nave "CICLON-III" fueron condenados a un destino implacable luchar sin tregua para no morir aplastados. ¡No les deje a su suerte! Sólo tiene que presionar la tecla "5" y la "8" para indicarles su camino hacia la izquierda o derecha respectivamente.

Tiene una particularidad, cual la de es simular el uso de la función SCROLL, inexistente en el Spectrum. No obstante, hay que reconocer que en juegos espaciales existe una gran competencia.

(16K-Spectrum).

Notas gráficas:

Línea 240 - Graphic 6, Graphic Shift 6.

Línea 270 - Graphic Shift 6, Graphic Shift 7.

Línea 350 - Graphic 6, Graphic Shift 6.

```
A S T E R O I D E S
```

```
  10 REM ** ASTEROIDES **
  20 FOR k=1 TO 5: PAUSE 5: FOR
i=1 TO 9: PRINT AT 5,5; INK i;"A
S T E R O I D E S": BEEP K/10,1
0: NEXT i: NEXT k
 100 LET a$=" *"
 110 LET n=0
 120 LET a=0
 130 LET b=0
 140 LET c=0
 150 LET d=0
 160 LET t=1
 170 LET x=12
 180 LET r=INT (27*RND)
 190 PRINT AT 21,r; INK (4*RND);
a$
 200 POKE 23692,255
 210 PRINT AT 21,31;"  ": REM 2
espacios.
 220 BEEP .3,0
 230 PRINT AT 10,x-2;"      ": R
EM 6 espacios.
 240 PRINT AT 11,x;"▄▀"
 250 PRINT AT 21,31;"  ": REM 2
espacios.
 260 LET n=n+t
 270 IF n=100 THEN
     LET a$="▀▄"
 280 IF n=104 THEN LET t=2
 290 LET e=d
 300 LET d=c
 310 LET c=b
 320 LET b=a
 330 LET a=r
 340 PRINT AT 10,x-2;"      ": R
EM 6 espacios.
 350 PRINT AT 11,x;"▄▀"
 360 IF x>=e-2 AND x<=e+t THEN G
O TO 410
 370 BEEP .2,5
 380 IF INKEY$="5" THEN LET x=x-
t
 390 IF INKEY$="8" THEN LET x=x+
t
 400 GO TO 180
 410 PRINT AT 11,x-1;"BOOOM!!"
 420 PRINT AT 0,0;"Puntuacion =
";n
 430 BEEP 2,-12
 440 PRINT AT 21,0;"Presione cua
lquier tecla para   continuar."
 450 IF INKEY$="" THEN GO TO 450
 460 RUN
```

EPSILON
micro-informática

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

ECHEGARAY, 9 – Telf. 22 20 27 - 27 00 73
MALAGA - 15

VALMAR DIV. INFORMATICA
SU TIENDA DE INFORMATICA EN CADIZ LE OFRECE:
* **Ordenadores:**
– Sinclair – Sharp – Toshiba
– Honeywell Bull – Dragon – Apple
* **Gran biblioteca de programas y libros**
* **Programas conformes a sus necesidades**
* **Periféricos**

C. CIUDAD DE SANTANDER, 8
Tfno. 28 10 69 27 60 42 – TX - 76171 VLMA E

MICRO WORLD
ORDENADORES PERSONALES Y MICROORDENADORES DE GESTION
- SPECTRUM
- KATSON
- ORIC-1
- APPLE
- NEW BRAIN
- ALTOS

SOFTWARE STANDARD Y A MEDIDA
CURSILLOS Y FORMACION
En Madrid:
MODESTO LAFUENTE, 36. TEL. 253 94 54

electrónica
LUVI
ORDENADORES PERSONALES
Vizcaya, 6 – Tfno. 230 44 84/ 227 89 62
MADRID

SINCLAIR / ZX - SPECTRUM
TU DISTRIBUIDOR EN VALENCIA
CESPEDES
COMPONENTES ELECTRONICOS
C/ San Jerónimo, 6
Tfno. 330 35 81/370 17 24
LIBROS - PROGRAMAS ACCESORIOS

Anúnciese por Módulos
Telf: (91) 4574566
(93) 3023648

AHORRE
CASI UN 25 POR CIEN
ZX SPECTRUM
48 K. . . . . 39.900,- Ptas.
SEIS MESES DE GARANTIA
SERVICIO DE REPARACION
(Para todo usuario de un ordenador SINCLAIR)
ENVIOS CONTRA REEMBOLSO
COMPUTER DISKONT
Barcelona - Tfno. (93) 241 55 18

ZX SPECTRUM
en BILBAO
Programas, libros, información...
gi **gesco-informática, s.a.**
C/ Telesforo Aramzadi, 1
Tfno. (94) 431 87 60

CLUB sinclair de
photo copy
Hazte socio del 1er Club de Informática de Galicia
Si quieres formar parte del CLUB SINCLAIR, rellena el cuestionario y envíalo o tráelo personalmente a PHOTO COPY. c/. Teresa Herrera, 9. La Coruña
Pronto recibirás noticias nuestras.
Apellidos:
Nombre:
Dirección:
Teléfono:
Plaza:
MODELO DE SINCLAIR ZX-81 ☐ ZX-Spectrum ☐
Recibido ☐☐☐ N.º entrada ☐
photo copy DISTRIBUIDOR OFICIAL sinclair
Teresa Herrera, 9 Telef. 213421 LA CORUÑA

COMPUTEST
REPARAMOS SU
SPECTRUM ZX 81
CON GARANTIA
Llame tfnos. (91) 415 95 88
416 73 85

micro Bierzo
Carlos I nº 2 - Telef. 417421 - Apartado 141 - PONFERRADA
CENTRO DE MICROINFORMATICA Y ELECTRONICA
ORDENADORES PERSONALES ZX-81 SPECTRUM
VIC-20 COMMODORE 64
Ordenadores de Gestión
Programas Profesionales - Docentes
de Gestión y de Juegos
CLUB DE USUARIOS – FORMACION

APRENDIENDO EL

SEGUNDA PARTE: LOS MNEMÓNICOS

16

BASIC versus código máquina

Una vez que estamos bien introducidos en el conocimiento de la organización de la memoria que es propia del ZX Spectrum, el autor nos conduce al siguiente paso, entrando propiamente en el código máquina que es la razón de ser de esta serie de artículos.

Instrucciones BASIC tan sencillas para nosotros como PRINT, GOTO o BEEP resultan muy complicadas para la CPU del Spectrum. El microprocesador Z80A es incapaz de comprender estas órdenes. Para hacerlo, debe traducirlas previamente mediante un intérprete BASIC que ocupa 7 Kbytes de memoria ROM.

La CPU trabaja con otro lenguaje, el código máquina. Al programar en éste, su lenguaje original, cada byte (8 bits) puede tomar un valor que en el sistema binario va del 00000000 al 11111111 (del 0 al 255 en base decimal).

Cada número, entre estos dos márgenes, corresponderá a una orden que será entendida de forma instantánea por la CPU y por lo tanto será ejecutada consecuentemente. Hay órdenes que ocupan dos, tres, cuatro y hasta cinco bytes.

Al ser humano le resultará imposible memorizar la correspondencia entre el valor binario y su comando. Para ello existe un lenguaje próximo al código máquina. En este lenguaje, llamado Assembler, se reemplaza cada número binario por una instrucción Assembler, también llamada mnemónica, que facilita la programación de la unidad de proceso central.

El microprocesador no se ayuda de variables numéricas o alfanuméricas, como nos tiene acostumbrado el BASIC. La CPU hace uso de los denominados REGISTROS.

Un registro simple, cuya longitud es de un byte, puede tomar un valor entre 0 y 255. Los registros dobles están formados por agrupaciones de dos registros simples, que al ocupar dos bytes pueden tomar valores que van del 0 al 65535, pudiendo ser tratados separadamente como registros simples.

REGISTROS DEL USUARIO

A-F
B-C
D-E
H-L

IX
IY

Registros dobles

Registros simples: A, F, B, C, D, E, H, L

Agrupación de simples en dobles: BC, DE, HL, AF.

La CPU del ZX Spectrum parece caprichosa por usar, para determinadas órdenes, sólo el registro A y para otras sólo el registro doble HL.

Los registros A y HL son los preferidos por el microprocesador.

Los lenguajes de programación que más se aproximan al código máquina de

CODIGO MAQUINA

cada ordenador son llamados "lenguajes de bajo nivel"; cuanto más evolucionan del código máquina van tomando niveles mayores, hasta llegar al BASIC o al COBOL, lenguajes de programación de alto nivel.

BASIC-COBOL

FORTRAN

ASSEMBLER
CODIGO MAQUINA

Ventajas y desventajas del BASIC frente al código máquina

Ventajas: BASIC
facilidad de programación
posibilidad de adaptar el mismo programa a otros ordenadores
programas más cortos en instrucciones
cálculos aritméticos fáciles

Desventajas:
ejecución y velocidad lenta
gran ocupación de memoria
obligatoriedad de trabajar con el sistema operativo
BASIC
LET A = A - L

Desventajas: C/M
difícil de programar

imposibilidad de adaptar el mismo programa a otros ordenadores
programas más largos en instrucciones
cálculos aritméticos difíciles

Ventajas: C/M
alta ejecución y velocidad
uso eficiente de la memoria
libertad del sistema operativo
C/M
código binario: 10010101
mnemónico: SUB L

El mnemónico SUB L corresponde al número hexadecimal 95.
El número decimal correspondiente al hexadecimal 95 es: 9*16 + 5 = 149

SUB L	95h	149	10010101

La instrucción mnemónica SUB L, o lo que es lo mismo, restar del registro A el valor L y almacenar el resultado en el registro A, corresponderá en código máquina al código binario 10010101.

17

Hexadecimal, decimal y binario

Por todo lo visto, hasta el momento, se hace imprescindible a la persona que quiera continuar trabajando en el código máquina, un conocimiento preciso de los sistemas de numeración decimal, hexadecimal y binario.

Del sistema de numeración decimal poco cabe decir, por ser el sistema de uso y conocimiento normales. El sistema hexadecimal, de una aplicación directa en microprocesadores de bytes de 8 bits —recordemos que en un byte de 8 bits caben 256 números diferentes que van del 0 al 255— siendo 256 igual a 16 **2, al crear un sistema de numeración en base 16 con dos caracteres dígitos o dígitos y letras, cubre toda la gama de decimales que puede contener un byte.

A continuación y dado que será de gran uso, incluimos la tabla de conversión de hexadecimal a decimal y viceversa:

TABLA DE CONVERSION DE HEXADECIMALES

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
2	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
3	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
4	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
5	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
6	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
7	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
8	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
9	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
A	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
B	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
C	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
D	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
E	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
F	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

TABLA DE CONVERSION DECIMAL A HEXADECIMAL

HEX	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	—128	—127	—126	—125	—124	—123	—122	—121	—120	—119	—118	—117	—116	—115	—114	—113
9	—112	—111	—110	—109	—108	—107	—106	—105	—104	—103	—102	—101	—100	— 99	— 98	— 97
A	— 96	— 95	— 94	— 93	— 92	— 91	— 90	— 89	— 88	— 87	— 86	— 85	— 84	— 83	— 82	— 81
B	— 80	— 79	— 78	— 77	— 76	— 75	— 74	— 73	— 72	— 71	— 70	— 69	— 68	— 67	— 66	— 65
C	—64	— 63	— 62	— 61	— 60	— 59	— 58	— 57	— 56	— 55	— 54	— 53	— 52	— 51	— 50	— 49
D	—48	— 47	— 46	— 45	— 44	— 43	— 42	— 41	— 40	— 39	— 38	— 37	— 36	— 35	— 34	— 33
E	— 32	— 31	— 30	— 29	— 28	— 27	— 26	— 25	— 24	— 23	— 22	— 21	— 20	— 19	— 18	— 17
F	— 16	— 15	— 14	— 13	— 12	— 11	— 10	— 9	— 8	— 7	— 6	— 5	— 4	— 3	— 2	— 1

Para valores superiores a 255, que no vienen contenidos en la tabla, la transformación de hexadecimal a decimal se efectúa de la forma siguiente:

5CCA = 5*4096 + 12*256 + 12*16 + 10 = 23754

Naturalmente, decimales superiores a 255 no pueden ser contenidos en un byte y necesitan para ello 2 bytes. Resulta claro que en dos bytes la información que pueden contener es de 16 bits y por tanto de 0 a 65535, es decir de 65536 números diferentes.

La CPU Z80A no maneja números superiores a éste. Al ser superiores los da como exponenciales de 10, mantisa y en coma flotante; el número máximo almacenable por el microprocesador es de ± 1e ± 38.

Recordemos lo dicho a este respecto en el número 3 de esta revista (pág. 56). Para almacenar números superiores a 255 en hexadecimal, tenemos que hacer uso de 2 bytes. Uno de los bytes almacena números entre 0 y 255; a este byte se le llama byte *"low"*, "inferior" o "menos significativo". El resto queda almacenado en el byte *"high"*, "alto" o "más significativo".

El sistema de numeración binario está directamente relacionado con el funcionamiento de un bit. Un bit que contiene una información elemental puede almacenar dos valores: el valor 0 o el valor 1. Un bit sólo podría contar en un sistema cuyo dígito más alto fuera el 1. Este sistema es el sistema binario. Números compuestos de unos y ceros representarán un número binario que tendrá una correspondencia con un número decimal. También dado la gran aplicación de este sistema de numeración parece interesante incluir el listado de equivalencias decimal y binarios entre 0 y 255.

LISTADO DE CONVERSION

0	00000000	128	10000000	43	00101011	171	10101011	86	01010110	214	11010110
1	00000001	129	10000001	44	00101100	172	10101100	87	01010111	215	11010111
2	00000010	130	10000010	45	00101101	173	10101101	88	01011000	216	11011000
3	00000011	131	10000011	46	00101110	174	10101110	89	01011001	217	11011001
4	00000100	132	10000100	47	00101111	175	10101111	90	01011010	218	11011010
5	00000101	133	10000101	48	00110000	176	10110000	91	01011011	219	11011011
6	00000110	134	10000110	48	00110001	177	10110001	92	01011100	220	11011100
7	00000111	135	10000111	50	00110010	178	10110010	93	01011101	221	11011101
8	00001000	136	10001000	51	00110011	179	10110011	94	01011110	222	11011110
9	00001001	137	10001001	52	00110100	180	10110100	95	01011111	223	11011111
10	00001010	138	10001010	53	00110101	181	10110101	96	01100000	224	11100000
11	00001011	139	10001011	54	00110110	182	10110110	97	01100001	225	11100001
12	00001100	140	10001100	55	00110111	183	10110111	98	01100010	226	11100010
13	00001101	141	10001101	56	00111000	184	10111000	99	01100011	227	11100011
14	00001110	142	10001110	57	00111001	185	10111001	100	01100100	228	11100100
15	00001111	143	10001111	58	00111010	186	10111010	101	01100101	229	11100101
16	00010000	144	10010000	59	00111011	187	10111011	102	01100110	230	11100110
17	00010001	145	10010001	60	00111100	188	10111100	103	01100111	231	11100111
18	00010010	146	10010010	61	00111101	189	10111101	104	01101000	232	11101000
19	00010011	147	10010011	62	00111110	190	10111110	105	01101001	233	11101001
20	00010100	148	10010100	63	00111111	191	10111111	106	01101010	234	11101010
21	00010101	149	10010101	64	01000000	192	11000000	107	01101011	235	11101011
22	00010110	150	10010110	65	01000001	193	11000001	108	01101100	236	11101100
23	00010111	151	10010111	66	01000010	194	11000010	109	01101101	237	11101101
24	00011000	152	10011000	67	01000011	195	11000011	110	01101110	238	11101110
25	00011001	153	10011001	68	01000100	196	11000100	111	01101111	239	11101111
26	00011010	154	10011010	69	01000101	197	11000101	112	01110000	240	11110000
27	00011011	155	10011011	70	01000110	198	11000110	113	01110001	241	11110001
28	00011100	156	10011100	71	01000111	199	11000111	114	01110010	242	11110010
29	00011101	157	10011101	72	01001000	200	11001000	115	01110011	243	11110011
30	00011110	158	10011110	73	01001001	201	11001001	116	01110100	244	11110100
31	00011111	159	10011111	74	01001010	202	11001010	117	01110101	245	11110101
32	00100000	160	10100000	75	01001011	203	11001011	118	01110110	246	11110110
33	00100001	161	10100001	76	01001100	204	11001100	119	01110111	247	11110111
34	00100010	162	10100010	77	01001101	205	11001101	120	01111000	248	11111000
35	00100011	163	10100011	78	01001110	206	11001110	121	01111001	249	11111001
36	00100100	164	10100100	79	01001111	207	11001111	122	01111010	250	11111010
37	00100100	165	10100101	80	01010000	208	11010000	123	01111011	251	11111011
38	00100110	166	10100110	81	01010001	209	11010001	124	01111100	252	11111100
39	00100111	167	10100111	82	01010010	210	11010010	125	01111101	253	11111101
40	00101000	168	10101000	83	01010011	211	11010011	126	01111110	254	11111110
41	00101001	169	10101001	84	01010100	212	11010100	127	01111111	255	11111111
42	00101010	170	10101010	85	01010101	213	11010101				

Si quiere calcular el valor decimal de cualquier número binario, que tenga 16 dígitos o bits, puede usar la función BIN, de esta manera:

PRINT BIN número binario.

Por todo lo dicho anteriormente, no se podrá calcular con el ZX Spectrum el equivalente decimal de un número binario con más de 16 bits. Recordemos que la CPU del ZX Spectrum no puede manejar números binarios con más de 16 dígitos o bits.

18

La pantalla

Vamos a introducirnos en un programa en código máquina, mediante un estudio de la pantalla. Recordemos cómo se estructura la pantalla:

La pantalla consta de 24 líneas por 32 columnas, lo que da un total de 24 × 32 = 768 caracteres. Cada caracter consta de 8 bytes, que contiene 8 bits. Un caracter consta en total de 8 × 8 = 64 bits.

La pantalla se almacena en memoria RAM, en el archivo de pantalla.

La longitud del archivo de pantalla es de 768 caracteres × 8 bytes/caracter = 6144 bytes ó 6 K de memoria. El número total de bits o *pixels* es de 6144 bytes × 8 bits/byte = 49152 bits o pixels.

El comienzo del archivo de pantalla en memoria RAM se fija en la dirección de memoria 16384. El archivo de pantalla se prolonga 6144 bytes, es decir hasta la dirección de memoria 22528.

Planteémonos el problema de imprimir toda la pantalla en negro, byte a byte.

Esquema del programa:

1. Inicialización una variable que contenga la primera dirección de memoria del archivo de pantalla (ver revista ZX n.° 3, pág. 54).
2. Introducir en la dirección de memoria correspondiente a esta variable inicializada el valor de 8 bits puestos a 1, ocho *pixels* en negro, que corresponden al valor decimal 255.
3. Incrementar en 1 la variable que almacena la dirección de memoria.
4. Comparar esta variable con el final del archivo de pantalla.
5. Si el valor de la variable es igual al final del archivo de pantalla, finalizar el programa.
6. Si el valor de la variable es inferior al final del archivo de pantalla, volver a INCREMENTAR (Punto 3).

Esta sería la estructura lógica del programa. Ver el organigrama para la representación gráfica del esquema anteriormente mencionado.

PROGRAMA BASIC EQUIVALENTE:

10 FOR F = 16384 TO 22527 Inicialización de la variable que almacena las direcciones de memoria del archivo de pantalla.

20 POKE F, 255 Se introduce en la dirección de memoria contenida en la variable F el valor 255 (11111111 en binario).

30 NEXT F La variable F se aumenta en 1. La función NEXT compara el valor de f con el número seguido después de la función TO (22527).

El programa termina si la variable F alcanza este valor.

El programa vuelve a introducir en la variable F el valor 255, que vuelve a ser comparada y aumentada si la variable F no ha alcanzado este valor.

La ocupación de memoria de este programa en BASIC es de 55 bytes.

PROGRAMA CODIGO MAQUINA EQUIVALENTE

Por fin, después de esta larga pero necesaria introducción, construimos el primer programa en código máquina, traducción al lenguaje del microprocesador.

1. Inicialización de la variable: LD HL, 16384.

La instrucción mnemónica LD registro, valor, introduce un valor en un registro.

El valor debe estar contenido en un byte, es decir debe tener valor entre 0 y 255 para un registro simple.

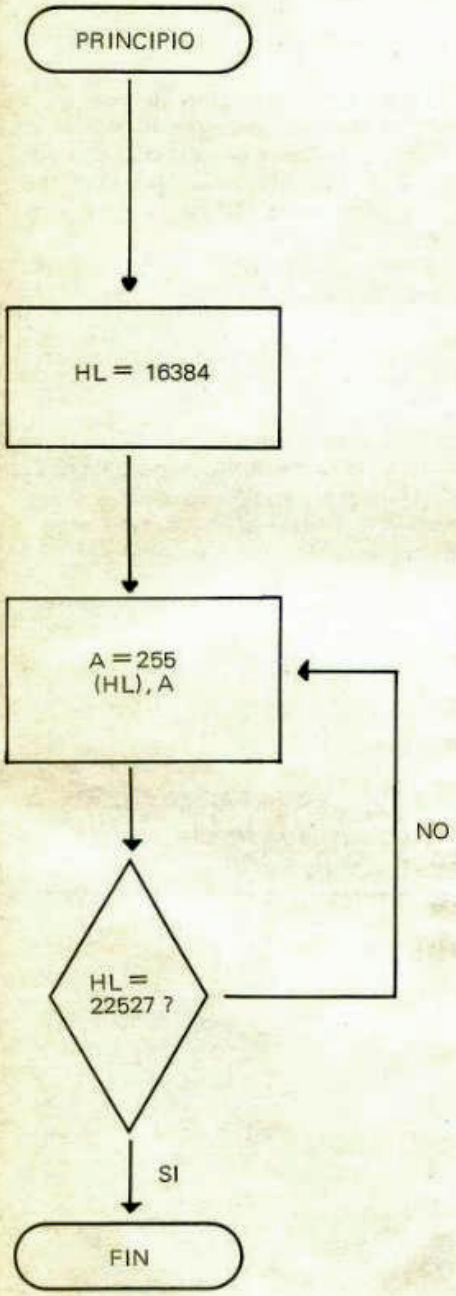
Si trabajamos con registros dobles, los valores deben estar contenidos en dos bytes, con un margen entre 0 y 65535.

LD A, 5 carga el registro A con el valor 5.

LD L, C8 carga el registro L con el valor 200 * = C8 hex.

LD HL, 5CCA carga en el registro doble HL el valor 5CCA h que equivale al 23754 en decimal.

El registro simple H toma el valor 5C. El registro simple L toma el valor CA.



Los registros H y L pueden ser tratados por separado o juntos en cuyo caso se forma el registro doble HL.

Cada orden "LD registro, valor" tiene un número de código que la identifica, dependiendo del registro en cuestión.

Así:

MNEMONICO
LD A, valor
LD H, dd
LDL, dd
LDHL, dd dd

INSTRUCCION HEXAD.
3E, valor
26 dd
2E, dd
21 dd dd

MSB LSB

LSB MSB

Existen códigos de instrucciones hexadecimales que cambian los valores de los registros entre sí, introducen el valor de un registro simple en una dirección de memoria expresada por un registro doble.

En este caso la instrucción

LD HL, 4000 (16384)

carga el registro doble HL con el valor hexadecimal 4000, decimal 16384. El registro H se carga con el valor 40 h; el registro L se carga con el valor 00 h. La instrucción mnemónica para cargar un número hexadecimal de dos bytes en el registro doble HL es 21 h (33 d). Los valores a cargar se introducirán después de este código, primero el del byte menos significativo (L) y después el del más significativo (cargado en el registro H). Así tendremos:

210040 LD HL, 4000 h

Seguidamente se inicializa un nuevo registro, el registro A. Cargamos en el registro A el valor 255.

El número 255 d corresponde en binario, al byte con la forma; 11111111; ocho unos. La representación en pantalla de un byte con esta forma será de ocho pixels en negro. Un bit con el valor 0 en el archivo de pantalla representa un pixels en blanco, un punto en blanco de una determinada posición de la pantalla. Un bit con valor 1 en el archivo de pantalla representa un pixel negro, un punto en negro, de una determinada posición de memoria. Si los ocho bits de un byte tienen valor uno (11111111), la posición de pantalla, cuyos pixels almacena ese byte, contendrá ocho pixels en negro. Si introducimos en todas las posiciones de memoria del archivo de pantalla el valor 255 (ocho bits con valor uno), toda la pantalla tornará a negro

3EFF LD A, FF h

Tenemos ahora un registro doble (HL) que almacena las direcciones del archivo de pantalla, comenzando por la 16384 d. También tenemos otro registro, el registro simple A que contiene el byte a introducir en las direcciones de memoria del archivo de pantalla.

2. Introducir en la dirección de memoria correspondiente a esta variable inicializada el valor de ocho bits puestos a unos, ocho pixels en negro, que corresponden al valor 255 d.

La instrucción LD tiene una forma especial:

LD (HL), A código 77 h, 119 d

El registro HL aparece entre paréntesis. Los paréntesis tienen aquí una gran importancia:

( ) = Contenido de

Los paréntesis aparecen, en instrucciones mnemónicas, sólo en registros dobles e implican aquí lo siguiente:

"La introducción del valor A en el valor contenido en el registro doble HL".
HL contiene el valor 4000 h A contiene el valor FF h

La instrucción mnemónica LD (HL), A se carga en el contenido de HL (4000 h) el valor almacenado por el registro A: "LD 4000 h, FF h".

Se introduce en la dirección de memoria 4000 h, 16384 d el valor FF h, 255 d.

Al ser esta dirección de memoria parte del archivo de pantalla tras esta instrucción vemos representado en pantalla ocho pixels en negro que corresponden al número 255 d y FF h. Ya tenemos parte del programa que queríamos realizar en código máquina:

```
210040    LD HL, 4000 h
3EFF      LD A, FF h
77        LD (HL), A
```

3. Incrementar en uno la variable que almacena la dirección de memoria. Las instrucciones mnemónicas INC aumentan en uno el valor de un registro. La instrucción mnemónica INC B representada por el código hexadecimal 04, aumenta en uno el valor del registro B.

```
0605    LD B,5      LET B = 5
04      INC B       LET B = B + 1
78      LD A, B     LET A = B
```

En nuestro caso particular aumentamos un uno el registro que almacena las direcciones de la memoria pantalla, el registro doble HL. El registro HL comienza con el valor 4000 h. En esta posición de pantalla se introdujo el byte FF h. Una instrucción INC HL aumenta el registro doble HL en uno para dar paso a la siguiente dirección de memoria en la pantalla.

4. Comparar esta variable con el final del archivo de pantalla.

La comparación en el lenguaje ASSEMBLER se realiza sólo a través del registro simple A, uno de los preferidos por la CPU.

La instrucción mnemónica CP, abreviatura para Compare o Comparar, comparará el registro A con, bien, un número, bien el valor del registro, o bien el valor almacenado en una dirección de memoria.

La CPU utiliza el registro F para almacenar las llamadas Flags o indicadores. Los indicadores tienen la función de transmitir información al programador acerca del registro simple preferido por la CPU, el registro A o acerca del último cálculo realizado.

Estos indicadores se almacenan en el registro F. Aunque el registro F

Fe de erratas

En la primera entrega de esta serie sobre código máquina se han deslizado algunos errores tipográficos que pueden dificultar la comprensión del texto. A continuación, damos las claves para reemplazar los textos erróneos y cifras erróneas por los correctos.

Página	donde dice	debe decir
52	ocupadas por 4986 bytes	ocupadas por 4096 bytes
56	22529	22528
56	23937	23938
56	23938-23942	23939-23943
56	23943	23944
56	23944-23948	23945-23949
59	primer buffer	printer buffer
64	concadevación	concatenación

Por otra parte, el programa número 4, que debía aparecer en la página 62, fue impreso en la página 64. Para mayor claridad, lo reproducimos.

```
  10 LET f=PEEK 23641+256*PEEK 2
3642
 100 PRINT f;TAB 10;PEEK f;TAB 2
0;CHR$ PEEK f AND PEEK f>32
 110 LET f=f+1: GO TO 100
```

```
23873     0
23874     0
23875     67          C
23876     93          ]
23877     0
23878     128
23879     247         RUN
23880     13
23881     128
23882     0
23883     0
23884     76          L
23885     93          ]
23886     0
23887     0
23888     51
23889     0
23890     0           ?
23891     0           ?
23892     0
```

contiene ocho bits, sólo se utilizan seis de éstos para almacenar los valores de los indicadores. Un indicador *(flag)* sólo puede tomar dos valores: cero o no. Cuando el bit correspondiente a un indicador toma el valor cero, el indicador está en posición "bajada" u *"off"*.

Cuando el bit correspondiente a un indicador toma el valor uno, el indicador está en posición "subida" u *"on"*.

Excepcionalmente, hay un bit que contiene dos indicadores: el indicador de paridad/*overflow*.

Algunas instrucciones del código máquina actuarán de una manera u otra de acuerdo con el estado de algún indicador.

En nuestro programa en código máquina sólo haremos uso del indicador de cero *(zero flag)*. Si el último cálculo realizado da como resultado cero el indicador de cero se "levanta" (estado *"on"*).

La instrucción mnemónica CP compara, como ya dijimos, el registro A. Se substrae del valor contenido en el registro A, el valor que estamos comparando. El resultado no se almacena en ninguna parte. La instrucción mnemónica CP sólo afecta a los indicadores.

En el caso que ambos valores sean iguales, el indicador de cero se subirá, el bit que almacena ese indicador tomará el valor uno.

```
LD A,5    LET A = 5
LD B,5    LET B = 5
CP B      Comparar: si A es igual a B indicador cero = 1
```

En nuestro caso particular debemos comparar si el valor del registro HL pertenece al archivo de pantalla o no. El archivo de pantalla comienza en la



Microshop,s.a.

BOUTIQUE INFORMATICA

Juan Sebastián Elcano, 6
Teléfono 27 20 65. SEVILLA - 11

OFERTA
CON LA COMPRA DE UN ORDENADOR UNA SUSCRIPCION POR UN AÑO

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

TODOS LOS JUEGOS DEL ZX 81 Y SPECTRUM
Amplia librería en Español, Inglés y Francés.

ORIC 16 K	38.000
ORIC 48K	49.500
IMPRESORA PLOTER ORIC .	45.000
DRAGON 32K	67.800
FLOPPY 170 K	95.000
COMODORE 64	80.000
MONITORES B.M.C.	29.000
MAGNETOFON SONYESPECIAL MICRO	9.750
VIC-20	43.890
SOFTWARE ORIC	1.700

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

OFERTA ESPECIAL DE PRIMAVERA:
CURSO INTENSIVO BASIC DE 30 DIAS, 24.000 Ptas. y REGALAMOS UN ORDENADOR PERSONAL.

dirección de memoria 4000 h (16384 d), y termina 6144 bytes más adelante, en la dirección de memoria 57FF h (22527 d).

La instrucción mnemónica CP debe comparar el registro A con el registro simple H, el byte más significativo del registro doble HL.

Cuando el registro doble HL, que almacena las direcciones de memoria del archivo de pantalla, llegue al final del mismo, pasará a tener el valor 5800 h, es decir, el byte menos significativo (L) contendrá el valor cero y el byte más significativo (H), el valor 58 h.

Debemos dar al registro A el valor 58 h (88 d) y compararlo después con el registro H. En el caso que ambos sean iguales se levantará el indicador de cero.

```
210040    LD HL, 4000 h
3EFF      LD A, FF h
77        LD (HL), A
3E58      LD A, 58 h
BC        CP H
```

5. Si el valor de la variable es igual al final del archivo de pantalla, finalizar el programa.

6. Si el valor de la variable es inferior al final del archivo de pantalla, volver al punto 2.

El comando análogo en Assembler del GOTO del BASIC es la instrucción JP dirección. La dirección está definida en hexadecimal por dos bytes. El primero será el byte menos significativo, el segundo será el byte más significativo. La instrucción mnemónica JP dirección ocupa 3 bytes en memoria. Es una "instrucción de bifurcación".

La instrucción JP es una abreviatura para JUMP o "salto absoluto", a la dirección definida por los dos bytes siguientes.

Por ocupar tres bytes en memoria y por otros motivos, es más frecuente usar la instrucción mnemónica JR desplazamiento. JR es una abreviatura para Jump Relative o "salto relativo". El desplazamiento viene definido por el así llamado "2.° complemento" (1). El 2.° complemento es un número que varía de —128 a +127. Es el 7.° bit el que decide si el número es positivo o negativo. Los 2.°s complementos positivos tienen el 7.° bit "off" o con valor 0 (números decimales entre 0 y 127). Los 2.°s complementos negativos tiene el 7.° bit con valor 1 y representan números entre —128 y —1.

```
comienzo...

3E05 comienzo    LD A, 5 h
77               LD (HL), A
3EOA             LD A, OAh
18F8             JR comienzo
```

Los saltos condicionales o relativos transfieren el control del programa a direcciones absolutas o relativas, cuando se ejecuta la instrucción...

Ya mencioné que existían ciertas órdenes en ASSEMBLER que funcionaban de acuerdo con el estado de los indicadores.

Unas de ellas son las instrucciones de salto condicionales.

Puede ser un salto absoluto o relativo.

absoluto:

CA dirección — JPZ dirección
Salto absoluto a la dirección definida por los bytes siguientes, si el indicador de cero tiene el valor 1

C2 dirección — JP NZ dirección
Salto absoluto a la dirección definida por los bytes siguientes, si el indicador de cero tiene el valor 0

relativo:

28 desplazamiento — JR Z desplazamiento
Salto relativo a la dirección actual más el desplazamiento, si el indicador de cero tiene el valor 1

20 desplazamiento — JR NZ desplazamiento
Salto relativo a la dirección actual más el desplazamiento, si el indicador de cero tiene el valor 0

Estas instrucciones de saltos relativos y absolutos condicionales del indicador de cero, conjuntamente con las instrucciones de comparación CP, son usadas mucho en programas en código máquina.

```
OC  comienzo    INC C
3E14            LD A, 14h
B9              CP C
20 FA           JR NZ comienzo
C9              RET
```

Este pequeño programa nos puede servir de ejemplo. Imaginando desconocer el valor del registro C, le aumentamos en uno. Este valor se compara con el registro A, cuyo valor es 14h (20d). Si los valores de C y del registro A no coinciden, el indicador de cero toma el valor 0, y la instrucción siguiente vuelve a la dirección de comienzo. Solamente en el caso que coincidan los valores de los registros c y A, el indicador de cero tendrá el valor 1, y la instrucción RET pasará a ejecutarse.

Esta última instrucción, RET, es una abreviatura para RETURN o RETORNO, que devuelve el control del programa a BASIC. Mientras el programa en código máquina estaba funcionando, ninguna orden ni función BASIC podía ser ejecutada: no se podía introducir ninguna orden, la tecla BREAK no respondía.

La falta de esta instrucción en c/m tendría como consecuencia la no aparición del cursor, no poder volver a introducir una orden BASIC.

RET también se utiliza como instrucción de retorno en subrutinas de código máquina.

MICRO WORLD

PRESENTA

Los mejores microordenadores del mundo a los mejores precios

SINCLAIR ZX81	14.975 ptas. desde	784 ptas. mes
SINCLAIR SPECTRUM 16K	39.900 ptas. desde	2.089 ptas. mes
SINCLAIR SPECTRUM 48K	52.000 ptas. desde	2.723 ptas. mes
ORIC-1 16K	38.000 ptas. desde	1.900 ptas. mes
NEW BRAIN	75.000 ptas. desde	3.928 ptas. mes
KATSON (Comp. APPLE II)	105.000 ptas. desde	5.499 ptas. mes
APPLE IIe	280.859 ptas. desde	14.709 ptas. mes

TODOS NUESTROS ORDENADORES CON 12 MESES DE GARANTIA Y UNA CINTA CON PROGRAMAS DE REGALO

Deseo información sobre .....

.....

.....

Nombre .....

Domicilio .....

Localidad .....

Ven a tu tienda:
Modesto Lafuente, 63 - MADRID-3 - Tel. 253 94 54

El código para la instrucción RET es el C9 h ó 201d.
Así sería por fin el programa completo en código máquina:

```
210040    LD HL, 4000h
3EFF      LD A, FFh
77        LD (HL), A
23        INC HL
3E58      LD A, 58h
BC        CP H
20F7      JR NZ siguiente
C9        RET
```

Este es todo el programa en código máquina. Si, resultado de la comparación, el indicador de cero tiene el valor (, el control del programa vuelve a la posición de memoria "etiquetada" con el distintivo "siguiente". Estas "etiquetas" *("labels")* sólo tienen la función de ayuda a la programación.

En caso que el indicador de cero tenga el valor 1, la siguiente orden mnemónica a realizar será el comando RET, que vuelva al BASIC. El informe 0: OK aparece.

Dado que el programa tiene un salto condicional relativo puede ser introducido en cualquier parte de la memoria. Si existiera en su lugar un salto absoluto, sólo podría estar en un sitio determinado, para que la dirección tras la orden de salto condicional absoluto encajara con el programa.

COMO INTRODUCIR LOS CODIGOS HEXADECIMALES

```
10 FOR F = 32000 TO 32012
20 INPUT A: POKE F, A
30 NEXT F
```

Debe introducir los siguientes códigos decimales, que corresponden con los hexadecimales:

```
33,0,64
62,255
119
35
62,88
188
32,247
201
```

(ocupación de memoria 13 bytes)

Siempre podrá encontrar en su manual del ZX Spectrum, Apéndice A, los códigos hexadecimales, sus correspondientes decimales y las órdenes mnemónicas.

Para hacer funcionar su programa en código máquina dar el comando: LET L = USR 32000.

Juan Martínez Velarde

(Continuaremos en el próximo número).

SU TIENDA INFORMATICA
EN CEUTA

CON LA GARANTIA
DE LA PENINSULA

sinclair

VIC 20

NewBrain

OSBORNE



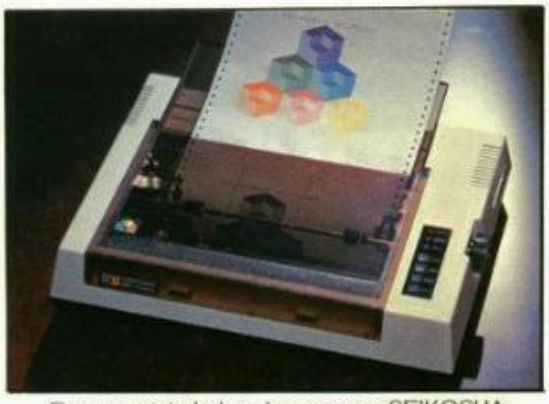
Toda la gama
SINCLAIR: ZX 81
ZX SPECTRUM, ZX IMPRESORA
(concesionario autorizado INVESTRONICA)

DRAGON 32

HEWLETT PACKARD



VIC-20
COMMODORE 64 y
todos los accesorios
COMMODORE...



Extensa variedad en impresoras SEIKOSHA:
GP - 100 GP - 100 VC (para Commodore con interf. Centronics incluido)
GP - 250 X y muy pronto la nueva gama: GP-50/GP-550/GP-700 color

SEIKOSHA

commodore 64

EPSON

SEIKOSHA

LAPIZ OPTICO PARA: ZX SPECTRUM VIC-20 COMMODORE 64

SOFTWARE DE INDESCOMP

EXTENSA VARIEDAD DE LIBROS PARA ZX SPECTRUM, ZX 81, VIC 20, COMMODORE 64

¡SOMOS LOS PRIMEROS EN DISPONER DE TODAS LAS NOVEDADES EN EL CAMPO DE LA INFORMATICA!

¡TENEMOS LOS PRECIOS MAS COMPETITIVOS DEL MERCADO!

LA MAS EXTENSA GAMA EN ORDENADORES PERSONALES:

- ZX SPECTRUM
- ZX 81
- OSBORNE 1
- COMMODORE 64
- VIC20
- DRAGON 32
- NEW BRAIN
- ATOM ACCORN
- ORIC 1
- etc, etc...

¡PEGA EL SALTO Y VEN A VERNOS A CEUTA!

¡¡YA DISPONIBLE
ZX MICRODRIVE!!

almacenes marisol

Casa Navalrai, Calle Camoens, nº 11 - CEUTA Teléfonos: 516840 - 516841 - 516842



TODO. ABSOLUTAMENTE TODO PARA SINCLAIR.

Y lo decimos así. De una manera rotunda y clara.

No en vano, somos la **única tienda profesional** de informática dedicada **exclusivamente** a SINCLAIR.

Ven a vernos. Podrás ver, a tu anchas, todas las novedades SINCLAIR, nacionales y extranjeras , todas las revistas y libros...

Tenemos la más extensa variedad en software (con todos los números 1 en Inglaterra). Los periféricos para SINCLAIR no tienen secretos para nosotros y somos los primeros en recibirlos.

Si no encuentras algo que buscas, nosotros lo compramos en tú nombre, en cualquier parte del mundo, gracias a nuestra extensa red comercial.

Regalamos un curso de BASIC para el SPECTRUM a todos nuestros clientes. Ven a vernos. Te esperamos.

sinclair store

Bravo Murillo, 2 - Telf. 446 62 31 - Madrid