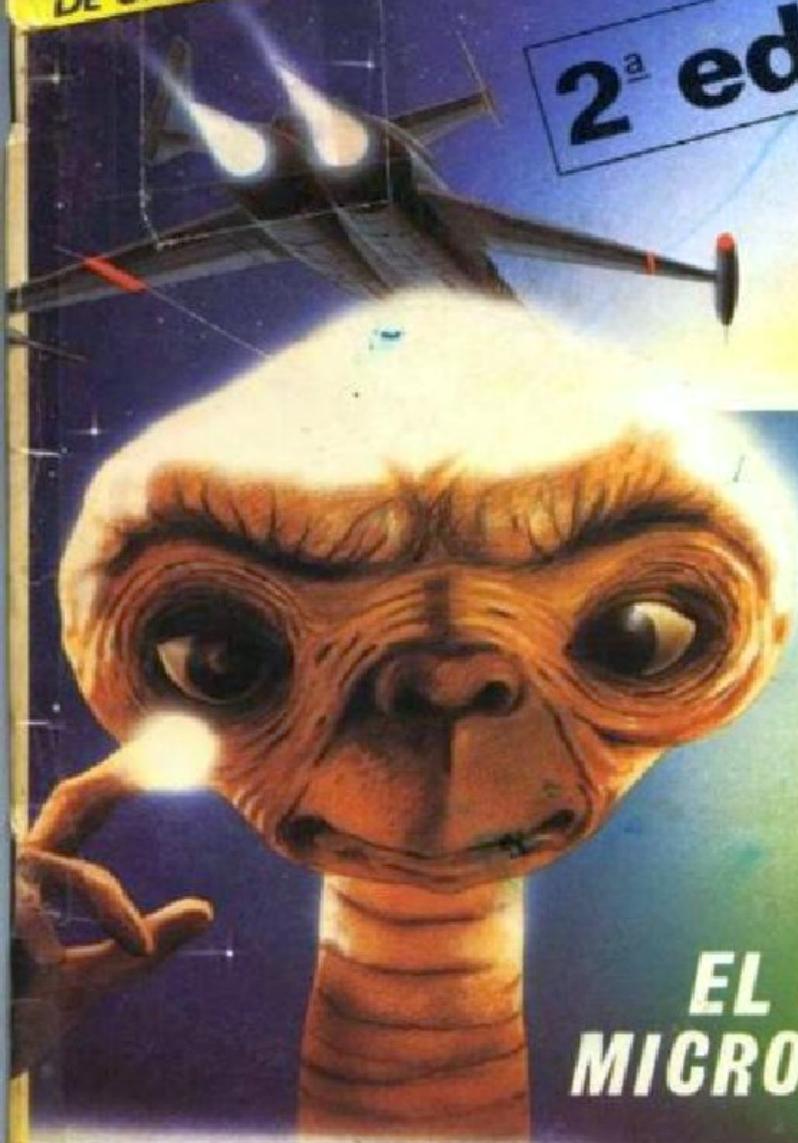


ZX

REVISTA PARA LOS USUARIOS DE ORDENADORES SINCLAIR

2ª edición



EL NUEVO MICRODRIVE



EDIT

RED CAPS LOCK

MAGENTA TRUE VIDEO

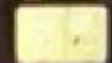
GREEN INV. VIDEO

CYAN

YELLOW

WHITE

GRA



DEL / FN

FN

LINE

OPEN II

CLOSE II

MOVE

ERASE

POINT

CAT

TCN

SIN

COS

TAN

INT

RND

STRS

CHR\$

CODE

PEEK

POKE

ASN

ACS

ATN

VERIFY

MERGE

PRINT

END

IN

OUT

STOP

ALAYA

COMPONENTES ELECTRONICOS GAZTEZ
Domingo Beltran, 58 (Vitona)
DATAI
Aida Gaztez, 51 (Vitona)
DEL CAZ
Aida Gaztez, 58 (Vitona)
VALBUENA
Virgen Blanca, 1 (Vitona)

ALBACETE

ELECTRO MIGUEL
Teodoro Gallego, 27
TECON
Marta Marin, 13

ALICANTE

ASEMCA (Vitona)
Avda. de la Constitucion, 54 (Vitona)
CDI
Roger Llana, 1 (Alcoy)
COMPONENTES ELECTRONICOS LASER
Jaume M + Buch, 7
ELECTRODATA LEVANTE
San Vicente, 26
ELECTRONICA AITANA
Limonero, s/n, Edificio Urquii (Benidorm)
ELECTRONICA OHRMO
Avda. El Hamed, 1
LIBRERIA LLORENS
Alameda, 50 (Alcoy)

AVILA

FELIX ALONSO
San Segundo, 15

BADAJOS

MECANIZACION EXTREMEÑA
Vicente Barantes, 16
SONYTEL
Villanueva, 16

BARCELONA

ARTO
C/ Angl, 43
BERENGUERAS
C/ Diputacion, 219
CATALANA D ORDINADORS
C/ Trafalgar, 70
CECSA
C/ Mallorca, 367
COMPUTERLAND
C/ Infanta Carlota, 89
COMPUTERLAND
Trav. de Dalt, 4
COMADUX
C/ Dos de Mayo, 234
D P 2000
C/ Sabino de Arana, 22-24
DIOTRONIC
C/ Conde Borrel, 108
EL CORTE INGLES
Avda. Diagonal, 617-619
EL CORTE INGLES
Pza. Catalunya, 14
ELECTRONICA H. S.
C/ S. Jose Orot, 9
ELECTRONICA SAUQUET
C/ Guillenas, 10
ELEKTROCOMPUTER
Va. Augusta, 120
EXPOCOM
C/ Vilanova, 68
GUBERNAU
C/ Sepulveda, 104
INSTA-DATA
P + S. Juan, 115
MAGIAL
C/ Sicilia, 253
MANUEL SANCHEZ
Pza. Mayor, 40 (Vic)
MILLIWATTS
C/ Melendez, 55 (Matoni)
ONDA RADIO
Gran Via, 581
RADIO ARGANY
C/ Borrell, 45
RADIO SONDA
Avda. Abad Margal, 77 (Tarrasa)
RAMEL ELECTRONICA
C/ de Vic, 3 (Manresa)
REDDA GESTION
Avda. Serna, 52-54
RFE ELECTRONICA
C/ Anibal, 80, 5 + 1 +
SERVICIOS ELECTRONICOS VALLES
Pza. del Gas, 7 (Sabadell)
STEMA
C/ Baimes, 434
S E SOLE
C/ Muntaner, 10
SUMINISTROS VALLPARADIS
C/ Dr. Ferrer, 172 (Tarrasa)
TECHNIFEI S. A.
C/ La Rambla, 19
VIDEOCOMPUT
P + Pep Ventura, 9, Bl. C. Bpis Bps (Vic)

BURGOS

COMELLECTRIC
Calzada, 7
ELECTROSON
Conde don Sancho, 6
TAGRA
Vitona, 13
ESA
Madrid-4

CACERES

ECO CACERES
Diego Maria Grehuet, 10-12

CADIZ

ALMACENES MARISOL
Canceros, 11 (Ceuta)
INFORMSA
Avda. Fuerzas Armadas, 1 (Algeciras)

ELECTRONICA VALMAR
C/udad de Santander, 8
M R CONSULTORES
Multi Centro Merca 60 (Jerez de la Frontera)
PEDRO VARELA
Pizarra, 36 (Jerez de la Frontera)
LEO COMPUTER
Garcia Escamez, 3
SONYTEL
Quempo de Llano, 17
SONYTEL
Jose Luis Diaz, 7
T L C Y AUTOMATICA
Dr. Herrera Quevedo, 2

CASTELLON

NOU DESPACH'S
Rey D. Jaime, 74

CIUDAD REAL

COMERCIAL R. P.
Travesera de Coso, 2 (Valdepeñas)
ECO CIUDAD REAL
Calatrava, 8
LAGUNA
Gran Capitan, 25 (Puerto Real)

CORDOBA

ANDALUZA DE ELECTRONICA
Falgue R, 15
CONTRON
Calle de Tomas Cabrera, 9
ELECTRONICA PADILLA
Sevilla, 9
MORM
Plaza Colon, 13
SONYTEL
Arta, 3
Avda. de los Mozarabes, 7

CUENCA

SONYTEL
Osmacio Garcia Izcara, 4

GERONA

AUDIFILM
C/ Albarreda, 15
CENTRE DE CALCUL DE CATALUNYA
C/ Barcelona, 35
S E SOLE
C/ Sta. Eugenia, 59

GRANADA

INFORMATICA Y ELECTRONICA
Melchor Almagro, 8
SONYTEL
Manuel de Falla, 3
TEONGAR
Archa de Gracia, 11

GRANOLLERS

COMERCIAL CLAPERIA
C/ Maria Maspons, 4

GUIPUSCOA

ANGEL IGLESIAS
Sancho el Sabio, 7-9
A y C
Una, 3 (Eibar)
BNP NORTE
Ramon M + Lir, 9
ELECTROBON
Reina Regente, 4

GUADALAJARA

RUZA
Mayor, 22

HUELVA

SONYTEL
Ruiz de Alda, 3

HUESCA

ELECTRONICA BARREU
M + Auzadora, 1

IBIZA

IBTEC
C/ Aragon, 76

JAEN

CARMELO MILLA
Coca de la Polera, 3
MARA ILLUMINACION
Avda. Linarios, 13 (Ubelda)
MICROUISA
Garcia Rebut, 6
SONYTEL
Jose Luis Diaz, 7
SONYTEL
Pasaje del Generalismo, 3 (Linares)

LA CORUÑA

DAVINA
República de El Salvador, 29 (Santiago)
PHOTOCOPY
Teresa Herrera, 9
SONYTEL
Avda. de Artejo, 4
SONYTEL
Terra, 37

LAS PALMAS DE GRAN CANARIA

COMPUTERLAND
Carvajal, 4

CHARRAI

Trans, 3
EL CORTE INGLES
Jose Mesa y Lopez, 18

LEON

AUXIFOR
Remota, 36
ELECTROSON
Avda. de la Facultad, 15

MICRO BIERZO

Carlos I, 2 (Ponferrada)

LERIDA

SELEC
C/ Ferrer y Busquet, 14 (Mollerusa)
SEMCO
C/ P y Margal, 47

LUGO

ELECTROSON
Concepcion Arenal, 38
SONYTEL
Primo de Rivera, 30

MADRID

ALFAMICRO
Augusto Figueroa, 16
BELLTON S
Torpedero Tucuman, 8
CHIPS-TIPS
Pto. Rico, 21
CMP
Pto. Santa Maria, 128
COMPUTERLAND
Castro, 89
COESA
Berlúello, 25
DINSA
Gastambide, 4
DISTRIBUIDORA MADRELEÑA
Todos sus centros
ELECTROSON
Duque de Sexto, 15 (y otros centros)
INVESTMICROSTORE
Gambúa, 7
J.P. MICROCOMPUT
Montesa, 44
EL CORTE INGLES
Todos sus centros
ELECTRONICA SANDOVAL
Sandoval, 4
MICROWORLD
Mozas-La Fuente, 63
PENTA
Dr. Cortáez, 12
RADIO QUER
Todos sus centros
SINCLAIR STORE
Bravo Murillo, 2
SONYTEL
Clara del Rey, 24 (y todos sus centros)
SONICAR
Vallehermoso, 19
VICOMUSICA
Ormaiztegui, 25

MALAGA

CANDI
Castelar, 33 (Algeciras)
EL CORTE INGLES
Prolongacion Alameda, s/n
INGESCON
Salicio Galaxia
SONYTEL
Salitre, 13

MELILLA

OFI-TRONIC
Hermanos Cayuela, 11

MENORCA

ELECTRONICA MENORCA
C/ Miguel de Ven, 50 (Mahon)

MURCIA

COMPUTER LIFE
Alameda San Anton, 2 (Cartagena)
EL CORTE INGLES
Libertad, 1
ELECTRONICA COMERCIAL CRUZ
Rio Segura, 2
MICRON
Gran Via, 8

NAVARRA

ENER
Pauino Caballero, 39
GABINETE TECNICO EMPRESARIAL
Juan de Labrit, 3
JOSE LUIS DE MIGUEL
Arreda, 11 bis
MATEO MUÑOZ
Hugarte Gola Maria, 8 (Tudela)

OVIEDO

AUTECA
Valentin Masip, 25
CUADRADO
Conde de Torro, 5 (Oviedo)
EDMAR
Camasa de Oris, 4-6 (Gijón)
ELECTRONICA RATO
Versalles, 45 (Aviles)
LIBRERIA DEL BOSQUE
Palacio Valdes, 9 (Gijón)
RADIO NORTE
Una, 20
RESAM ELECTRONICA
San Agustín, 12 (Gijón)
RETELCO
Cabriles, 31 (Gijón)
SELECTRONIC
Fermín Canelas, 3

ORENSE

COPINSE
Juan XXIII, 15
SONYTEL
Concejo, 11

PONTEVEDRA

EL CORTE INGLES
Gran Via, 25 (Vigo)
ELECTROSON
Santa Clara, 32
ELECTROSON
Venezuela, 32 (Vigo)

SONYTEL

Salvador Moreno, 27
SONYTEL
Gran Via, 52 (Vigo)
TEFASA COMERCIAL
San Salvador, 4 (Vigo)

PALMA DE MALLORCA

GLFT
Va. Alemaná, s/n
IAM
C/ Cecilio Mello, 5
TRON INFORMATICA
C/ Juan Alcover, 54, 8 + C

LA RIOJA

OMESA
Gran Via, 57
YUS COESSA
Ciguñeta, 15

SALAMANCA

DEL AMO
Arco, 5
PRODIESTE
España, 65

SANTANDER

LANZ S A
Reina Victoria, 127
RADIO MARTINEZ
Dr. Jimenez Diaz, 13
VIDEOSON
Solares Escalante, 11 (Torrelavega)
ELECTROKAT
Julian Ceballos, 22 (Torrelavega)

SEGOVIA

ELECTRONICA TORIBO
Obispo Quesada, 6

SEVILLA

A.D.P.
San Vicente, 3
EL CORTE INGLES
Duque de la Victoria, 10
SCI
Acetuno, 8
SONYTEL
Pagos del Campo, 173
Adriano, 32

TARRAGONA

AIA
Rambla Nova, 45, 1 +
CIAL INFORMATICA TARRAGONA
C/ Gasometro, 20
ELECTRONICA REUS
Avda. Prat de la Riba, 5 (Reus)
SEIA
Rambla Vella, 7 B
S E SOLE
C/ Cronista Sesé, 3
T V HUQUEI
Pza. Mayor, 14 (Montblanc)
VRGLI
C/ Dr. Gimbernat, 19 (Reus)

TOLEDO

CENTRO INFORMATICO TOLEDANO
Talamanca, 6
ORTOPEDIA TOLEDANA
Martinez Simancas, 9

STA. CRUZ DE TENERIFE

COMPUTERLAND
Mendez Nuñez, 104 B
TREN CANARIAS
Serrano, 41

VALENCIA

ADISA
San Vicente, 33 (Gandia)
CESPEDES
San Jacinto, 6
COMPUTERLAND
Marques del Tuna, 53
DRAC
Blasco Ibañez, 116
EL CORTE INGLES
Pintor Sorolla, 26
Melendez Pidal, 15
PROMOCION INFORMATICA
Pintor Zanfrena, 12

VALLADOLID

SONYTEL
Leon, 4

VIZCAYA

BLBOMCRO
Aureliano del Valle, 7
DATA SISTEMAS
Henao, 58
DISTRIBUIDORA COM
Gran Via, 19-21 y todos sus centros
EL CORTE INGLES
Gran Via, 9
ELECTROSON
Alameda de Urquijo, 71
San Vicente, 18 (Baracaldo)
GESCO INFORMATICA
Alameda de Reaide, 76
KEYTRON
Hurtado de Ameraga, 20
MUSICAL DEL ABRA
Mayor, 7 (Las Arenas)

ZAMORA

MEZZASA
Victor Gallego, 17

ZARAGOZA

CEMECA
Mendez Nuñez, 31
EL CORTE INGLES
Sagasta, 3
SONYTEL
Va. Pignatelli, 29-31

DISTRIBUIDOR
EXCLUSIVO:
INVESTRONICA

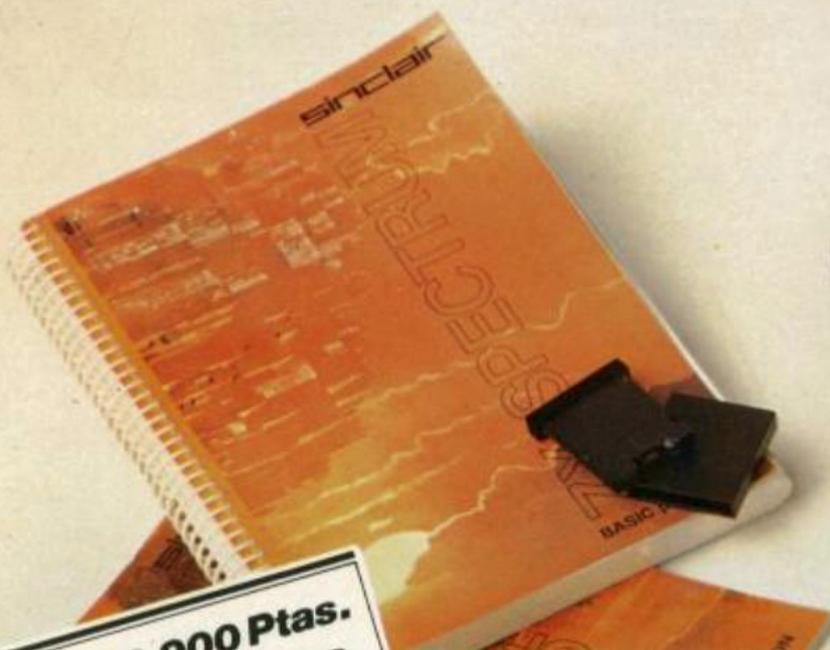
Central Comercial
TOMAS BRETON, 60
TELF. 468 03 00
TELEX 23399 IYCO E
MADRID

Delegación Cataluña
MUNTANER, 565
TELF. 212 68 00
BARCELONA

(15-XI-83)



El ordenador de to



16 K: 39.900 Ptas.
48 K: 52.000 Ptas.

Ayer

El SINCLAIR ZX SPECTRUM ha nacido de la experiencia y técnica adquirida con su hermano pequeño SINCLAIR ZX 81.

Ese pequeño ordenador ha conseguido batir todos los records en lo que a popularidad y ventas se refiere: Más de DOS MILLONES de usuarios en todo el mundo. ¡Parece increíble, verdad!

Hoy

Cuando SINCLAIR decidió poner en el mercado una segunda generación, tenía ante sí, un gran reto. Necesitaba crear un micro-ordenador con el mismo "espíritu" de sencillez de manejo que el ZX 81 pero a la vez con la potencia y las posibilidades de otros ordenadores más grandes, sin perder de vista el precio, con objeto de hacerlo accesible a todos los niveles.

Y SINCLAIR consiguió, una vez más, ganar la batalla al tiempo y a la técnica.

Nació el ordenador de todos... para todo: **SINCLAIR ZX SPECTRUM.**

— Util para los más pequeños, con su amplia variedad de juegos, incluido el aprender a programar en BASIC, como si de otro juego se tratara.

— Para los jóvenes es la más potente calculadora técnico-científica, para la resolución de los más complicados problemas matemáticos, amén de introducirles en el mundo de la informática.

— Para los padres es de la mayor utilidad, tanto en el hogar como en la empresa: fichero de recetas, agenda de amistades, cálculo de menús dietéticos, contabilidad, control de stocks, etc., etc.

Mañana

SINCLAIR está dotando al ZX SPECTRUM de los mayores adelantos técnicos; como por ejemplo el ZX MICRODRIVE.



SINCLAIR ZX 81.



odos... para todo.



El ZX MICRODRIVE es un nuevo concepto de almacenamiento de datos. He aquí algunas características:

- Capacidad de almacenamiento: 85 K
- Tiempo de acceso medio: 3,5 segundos
- Tiempo de carga: 9 segundos (en programa típico de 48 K)
- Conexión de hasta 8 Microdrives en serie (640 K)

También podríamos hablar del ZX INTERFACE 1, preparado para los Microdrives y la creación de la ZX RED... O del ZX INTERFACE 2, creado para los JOYSTICKS y los nuevos ZX CARTUCHOS o también de...

IMPORTANTE:

Al adquirir su ZX SPECTRUM **EXIJA LA TARJETA DE GARANTIA INVESTRONICA**, única válida para todo el territorio nacional y llave para cualquier resolución de duda o reparación. INVESTRONICA no prestará ningún servicio técnico a todos aquellos aparatos que carezcan de la correspondiente garantía.

CARACTERISTICAS TECNICAS: CPU/Memoria

Microprocesador Z80A, RAM de 16K o 48K, ROM de 16K con intérprete BASIC y sistema operativo.

Teclado

Con 40 teclas móviles de agradable tacto. Todas las palabras BASIC se obtienen mediante una sola tecla. Repetición automática.

Representación Visual y Gráficos

32 x 24 caracteres, mayúsculas o minúsculas. Caracteres redefinibles por el usuario. Alta resolución gráfica: 256 píxeles x 192 píxeles.

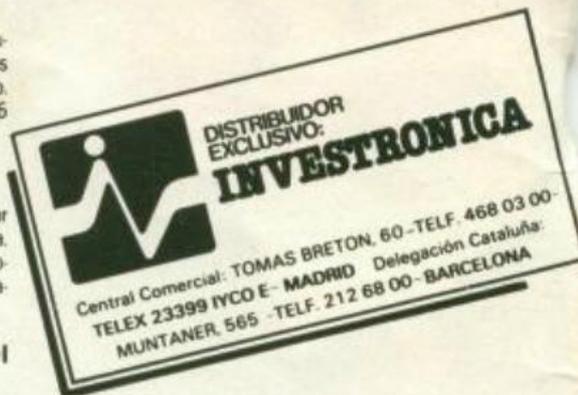
Color y Sonido

Ocho colores, pudiendo estar simultáneamente en pantalla. Altavoz interno: 130 semitonos (10 octavas) con amplificación por toma de micro.

Compatibilidad del ZX-81

El BASIC del ZX-81 es esencialmente un subconjunto del BASIC del ZX Spectrum (consulten las diferencias).

(Escueto resumen de algunas características técnicas. Para total información solicite folleto ilustrativo, a todo color, a su distribuidor habitual o bien, directamente, a INVESTRONICA, sin cargo alguno).



Central Comercial: TOMAS BRETON, 60 - TELF. 468 03 00 -
TELEX 23399 IYCO E - MADRID Delegación Cataluña:
MUNTANER, 565 - TELF. 212 68 00 - BARCELONA

ORDENADOR PERSONAL

SINCLAIR ZX Spectrum

Más que un ordenador... un compañero.

DE VENTA EN CONCESIONARIOS
AUTORIZADOS

AÑO I - No 1

ZX

REVISTA PARA LOS USUARIOS
DE ORDENADORES SINCLAIR

Bienvenido, lector, a esta revista que haremos entre todos, en nuestra común condición de usuarios de los ordenadores ZX81 y ZX Spectrum. Este primer número coincide con la presentación en España del tan esperado Microdrive para los ordenadores Sinclair. De este nuevo dispositivo de almacenamiento ofrecemos un test como apertura de la revista. Presentamos también programas, montajes y un análisis del BASIC específico de los ordenadores que son la razón de existir de esta revista. Aunque, a decir verdad, su razón de ser son los lectores, con los que esperamos iniciar desde hoy mismo un diálogo permanente.

NOVEDADES

Ya está aquí el esperado ZX Microdrive. ¿Valía la pena esperar quince meses? A partir de la página 4 le ofrecemos un test de este nuevo dispositivo de almacenamiento que ensancha notablemente las prestaciones de los ordenadores ZX. Y junto con él, comentamos las nuevas capacidades que aportan las interfaces presentadas simultáneamente por Sinclair Research y que, entre otras, ofrecen la posibilidad de conectar en red hasta 64 Spectrum, con las imaginables ventajas para aplicaciónes educativas.

Página 4

PROGRAMAS

Juegos. De monstruos, de marcianitos y de escapada. Pero también aplicaciones matemáticas y estadísticas. Con diferentes niveles de dificultad. Para todos los gustos.

Página 10

MONTAJES

Para los "manitas", este primer número de ZX presenta dos montajes de extraordinaria utilidad. Por un lado, un dispositivo para proteger la grabación de software. Por otro, una interface —que sabemos será bien recibida— para impresora RS232.

Página 44

SOFTWARE

Una nota de introducción, a la vez que de análisis, al lenguaje BASIC que es específico de los ordenadores Sinclair. Lo que tiene de original, lo que tiene de común con la versión de Microsoft. Sus comandos y rasgos característicos.

Página 49

ZX es una publicación de Ediciones y Suscripciones, S. A. • Presidente: Fernando Bolín • Jerez, 3. Telfs.: (91) 250 15 92 - 458 76 02, Madrid-16 • Director: Norberto Gallego • Redacción: Alejandro Diges, Anibal Pardo, Simeón Cruz, Gumersindo García • Diseño: A. Gordillo • Circulación: Luis Carrero • Suscripciones: Antonio Zurdo • 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.



Quince meses después de su primer anuncio, podemos decir que ya está aquí el Microdrive de Sinclair. Durante ese tiempo, todo han sido rumores acerca de la filosofía y los componentes de este nuevo dispositivo —¿llevaría cinta o disco?, era uno de los interrogantes— sin que Clive Sinclair hiciera nada por desvelar el misterio. Todo lo que el inefable inventor británico admitió se limitaba a este concepto: "será un soporte intercambiable".

Hoy, por fin, las especulaciones han cesado, las preguntas tienen respuesta. El ZX Microdrive y las dos interfaces que lo complementan ya están disponibles. Bueno, no tanto: por el momento sólo se comercializa en el mercado británico... a cuentagotas. El importador de Sinclair en España promete ponerlo a disposición del público usuario en febrero del 1984 a un precio que, en principio, oscilará entre las 15.000 y las 20.000 pesetas.

El Microdrive tiene unas dimensiones externas de $90 \times 85 \times 40$ mm. En la parte delantera presenta la abertura frontal para la carga del cartucho, y en la trasera dos conectores sobre una de las dos tarjetas de circuito impreso que aloja. La otra tarjeta, vertical, lleva la ROM y la cabeza que hace las funciones de lectura y grabación en la cinta. Todos los elementos mecánicos de la unidad son sencillos y robustos. La única preocupación podría ser el posible desgaste de la cabeza debido a la acción abrasiva de la cinta.

El pequeño cartucho ZX Microdrive (compárense sus dimensiones de $45 \times 35 \times 7$ mm. con los 75 mm. de un diskette) contiene una banda continua de alta calidad, que mide aproximadamente 15 metros y tiene una anchura de 1,9 mm. Pese a las prometidas 100 K, su capacidad garantizada es de 85 K, aun-

EL NUEVO ZX MICRODRIVE

que en la práctica suele resultar ligeramente superior (normalmente 89 ó 90 K). Pueden montarse simultáneamente hasta un máximo de 8 unidades, con lo que se dispone de una capacidad total de 680 K o superior. Los Microdrives se alimentan de la fuente del Spectrum, y se conectan a él a través de una pequeña tarjeta que se ha denominado ZX Interface 1. Esta tarjeta entra sencillamente en el conector de la parte de atrás del Microordenador. Dos de los tornillos de la base del Spectrum se sustituyen por otros dos incorporados a la interfase y que sirven para que quede firmemente asegurada. Esta vez, por tanto, no va a pasar como con el famoso módulo de RAM del ZX81, cuya insegura conexión daba lugar a toda clase de problemas. Por el contrario, tiene la ventaja añadida de levantar ligeramente el teclado del Spectrum, dejándolo en un ángulo más cómodo. Quienes hayan sustituido el teclado original por un modelo "propio" juegan en este caso con desventaja.

Gracias a la Interfase I, el Spectrum puede gobernar una impresora RS232. Asimismo, gracias a esta tarjeta, ahora es posible crear una red de hasta 64 Spectrum enlazados. La interfase RS232 opera en un margen de 50 a 19.200 baudios, sin paridad, ocho bits por carácter y dos bits de stop.

Existen dos modos de operación. El canal t se utiliza normalmente para listados; los códigos de control no se envían (a excepción del 13, retorno de carro), y los códigos característicos se expanden. No dispone de capacidad gráfica; todos estos códigos se sustituyen por 63. El canal b envía códigos de ocho bits completos y se utiliza precisamente para mandar caracteres de control a impresoras, etcétera.

La ZX Net (red en área local) promete ser una herramienta muy poderosa. La red se establece simplemente uniendo dos interfases de Spectrum con un cable de 10 pies. El manual habla de una red funcionando a 100 Kbaudios con handshaking entre dos equipos conectados. Pueden enlazarse hasta un máximo de 64 Spectrum. Existe también una función de comunicación para enviar información a cualquiera que esté conectado a la red. Gracias a esta posibilidad, un profesor podría enviar, por ejemplo, un programa simultáneamente a toda una clase. Otro uso podría ser el establecimiento de una estación impresora compartida por todos los Spectrum de la red.

Pero para evaluar seriamente todas

estas posibilidades, es preciso esperar todavía algún tiempo. No obstante, la principal función de la Interface I, y en la que nos encontraremos aquí, es la de controlar los Microdrives.

Cada unidad se conecta a la interfase mediante un cable plano de corta longitud. Las unidades adicionales se conectan cada una a la siguiente mediante conectores rígidos (con cada Microdrive viene uno de estos conectores), fijándose mediante una abrazadera que se atornilla en la base de cada unidad.

La interfase amplía el BASIC de Sinclair para que incluya las necesarias funciones de manejo de ficheros y de comunicaciones, y utiliza el BASIC como un sistema operativo. El manual del Microdrive y la Interface I da todo tipo de detalles sobre las ampliaciones del lenguaje que se han introducido y sobre su utilización. Incluye también los nuevos códigos de error asociados con los Microdrives. Cuando ha sido posible se ha mantenido la sintaxis de las instrucciones precisas para indicar qué es el Microdrive al que se refieren.

Cómo se utiliza el Microdrive *

Una vez conectado un Microdrive con el cable plano a la tarjeta de interfase y ésta al Spectrum, la primera tarea consiste en formatear los cartuchos vírgenes. La manera de hacerlo es introducir sencillamente `FORMAT "m"; 1; "nombre"`, donde "m"; 1 identifica la unidad que se está utilizando (en este caso el Microdrive 1), y "nombre" puede ser cualquiera que tenga 10 caracteres o menos. El formateado tarda unos 30 segundos, durante los cuales la pantalla se ilumina y se borra alternativamente hasta que aparece el OK que indica el fin de la operación. El formateado de un soporte es esencial para que el ordenador identifique las áreas en las que lee, escribe o marca como no utilizables. Una vez hecha esta operación ya no hay que repetirla nunca, a no ser que se quiera borrar por completo un cartucho para dedicarlo a otros fines (como puede hacerse con el cartucho de demostración que viene con el Microdrive). Cuando se introduce CAT 1 (seguimos en el supuesto de que se está utilizando la unidad 1), aparecerá en pantalla el nombre que se ha dado al cartucho, y los nombres de los ficheros que contiene y el espacio no ocupado de que se dispone. La protección contra escritura se logra

quitando una pestaña de plástico, y, al igual que con los cassettes, para poder grabar de nuevo se utiliza el sencillo procedimiento de tapar el hueco con un trocito de cinta adhesiva.

Para guardar un programa en el Microdrive hay que dar la orden `SAVE* "m"; 1; "nombre"` (el asterisco indica al microordenador que la instrucción se refiere al Microdrive y no al cassette, y "m"; 1 identifica el número de unidad. La comprobación de que el almacenamiento de un programa ha sido correcto se hace con `VERIFY * "m"; 1; "nombre"`, y la instrucción de carga desde el Microdrive es `LOAD* "m"; 1; "nombre"`.

Se pueden mezclar programas con `MERGE "m"; 1; "nombre"`, así como utilizar la posibilidad de auto ejecución usual del Spectrum añadiendo `LINE x` a la orden `SAVE`. El Microdrive tiene su propia función de carga y ejecución conjuntas, es decir que basta con escribir `RUN` y no `LOAD` y `RUN` cuando se quiere ejecutar un programa residente en el Microdrive.

Para ello es preciso que el programa en cuestión se llame precisamente "run" —tecleándolo letra a letra— y que el cartucho esté en la unidad 1. Esta posibilidad hay que utilizarla inmediatamente después de conectar el aparato o de haber hecho un `NEW`. El programa se guarda con la orden `SAVE "m"; 1; "run" LINE número`. Un `RUN` dado ahora desde el teclado, cargará y ejecutará el programa. Debe observarse que la instrucción `MERGE` no funciona con programas que se hayan guardado utilizando `LINE`.

Aunque lo anterior se refiere a la transferencia de programas en BASIC, las rutinas en lenguaje de máquina se manejan también fácilmente en la manera usual, añadiendo `CODE` al final de la correspondiente instrucción. Un detalle bastante molesto es que la lista de ficheros que proporciona `CAT` no hace distinción entre programas en código de máquina y programas en BASIC. Con el cassette, por el contrario, el BASIC de Sinclair proporciona una serie de prefijos muy prácticos, como "Programa", "Bytes", "Matriz numérica", etcétera. Habría sido de agradecer que esta práctica hubiese continuado con los Microdrives. Pero tal como está, no se da indicación alguna; si se omite el sufijo `CODE` para un programa en lenguaje de máquina, lo único que se obtiene es un mensaje de error.

Borrar un programa o un fichero es lo

NOVEDADES

más sencillo del mundo; basta con introducir ERASE "m"; 1; "nombre". Si se hace un BREAK mientras se está guardando un programa en el cartucho quedará un fichero sin cerrar, y cualquier intento de cargarlo terminará en un poco reconfortante mensaje de "file not found" (no se encuentra el fichero). La instrucción ERASE borra también este tipo de ficheros no cerrados, pero el micro tarda unos 30 segundos en hacer varias comprobaciones del cartucho, en un intento de buscar el inexistente final. En general, hay que tener cuidado de no sacar el cartucho de la unidad mientras el LED testigo esté encendido. Asimismo, tampoco es nada recomendable conectar o desconectar la alimentación sin retirar el cartucho del Microdrive.

Canales de datos y Streams

Todo lo dicho hasta ahora se refiere a la utilización del Microdrive para el almacenamiento de programas. Sin embargo, el aspecto más interesante de la nueva unidad no es éste, sino su uso en el almacenamiento y recuperación de datos desde los programas.

En relación con esta función se introducen los conceptos de canales y de streams. Al hablar de canales, nos estamos refiriendo a los diversos dispositivos componentes de un sistema. Así diremos que los datos pueden enviarse a la pantalla, a la impresora ZX, al Microdrive, a otro Spectrum sin ambas máquinas están conectadas en una red, o a la interfase RS232 y desde aquí a una impresora u otro periférico. De otra parte, los datos pueden provenir de los siguientes canales; el teclado, un fichero

del Microdrive, otro Spectrum, o, a través de la interfase RS232, de un modem o un terminal.

Los datos se envían y se reciben por estos canales por medio de streams (llamémoslas rutas). Este concepto resultará ya familiar a los usuarios del Spectrum. El Manual de Introducción al BASIC da algunos detalles al respecto, y los usuarios quizá hayan utilizado streams para imprimir en las dos últimas líneas de la pantalla, por ejemplo. El Spectrum es capaz de utilizar 16 rutas, numeradas de 0 a 15; (estos números se utilizan siempre precedidos del signo #). Cuatro rutas están permanentemente asociadas a ciertos canales. Así, las rutas #0 y #1 envían datos a la parte inferior de la pantalla y sirven de entrada desde el teclado; la ruta #2 actúa de salida a la parte superior de la pantalla, pero no puede ser entrada; la ruta #3, finalmente, está asociada a la Impresora ZX y tampoco puede ser entrada.

Cada una de las instrucciones de entrada o salida utiliza una de estas rutas. Las sentencias PRINT y LPRINT son, de hecho, expresiones abreviadas de PRINT#2 y PRINT#3 respectivamente. Una instrucción puede hacer uso de una ruta diferente sin más que especificar el número correspondiente; LPRINT #2, por ejemplo, dirigiría la salida a la pantalla en lugar de hacerlo a la impresora.

El usuario no está limitado a utilizar estas cuatro rutas "predefinidas". Los números #4 a #15 están disponibles para cualquier trabajo. Con los especificadores de canal, el usuario determina el periférico que desea. Así, "K" para el teclado, "S" la pantalla, "P" la impresora ZX, "T" para texto (utilizado con el

RS232), "B" para binario (también RS232), "N" para red y "M" para Microdrive.

Para poder utilizar estas rutas y estos periféricos, se precisa de entrada una instrucción OPEN#. OPEN#4, "S" abre, por ejemplo, la ruta #4 y la asocia al canal S (la pantalla). Si se ejecuta ahora PRINT#4; "texto", este texto aparecerá en la pantalla. Obsérvese que K, S y P son canales establecidos y requieren que se utilicen comas como separadores en las instrucciones OPEN#. Los otros dispositivos admiten que el separador sea ";" o ":". El manual previene al usuario contra la apertura de las rutas #0, #1 y #2 advirtiéndole que los resultados serían impredecibles. Repare también en que el símbolo # aparece automáticamente con OPEN y CLOSE sin tener que teclearlo.

Visto esto, pasemos ya a ver cómo se utilizan estas rutas y canales.

Después de abrir un fichero de datos con la instrucción, por ejemplo, OPEN#4; "m"; 1; "números" (habilitando así un nuevo canal "m"; 1; "números" y asociándolo con la ruta #4), ya pueden introducirse los datos. El manual pone como ejemplo un fichero de datos que contiene los números 1 al 10 y sus cuadrados. Este programita

```
10 FOR N =1 TO 10
20 PRINT 4, N, N*N
30 NEXT N
```

produce los datos que se requieran. A primera vista podría suponerse que basta con este programa para grabar en el Microdrive los números generados, pero es preciso decir que el microordenador no transfiere nada hasta que está lleno el buffer del Microdrive (tiene 512

TABLA DE FORMATOS DE LOS COMANDOS DEL MICRODRIVE

LOAD * "M"; 1; "NOMBRE"	El * indica la carga a partir de un microdrive.
SAVE * "M"; 1; "NOMBRE" LINE 10	Establece el nombre para el comienzo automático del mismo. El n.º de línea es opcional.
VERIFY * "M"; 1; "NOMBRE"	
MERGE * "M"; 1; "NOMBRE"	No mezclará programas guardados con un número de línea.
ERASE "M"; 1; "NOMBRE"	

FORMAT "M"; 1; "TITULO" Título se refiere a una cabecera para el cartucho.

TABLA DE LOS COMANDOS DEL BASIC EXTENDIDO

* Significa una función del Microdrive.	
CAT Y	Y es el número del Microdrive de 1 a 8.
CAT # Z; Y	Z es el número de corriente ¹ de 0 a 15.
CLOSE # Z	
ERASE "M"; Y; "NOMBRE"	NOMBRE es el nombre de un fichero.
FORMAT "M"; Y; "TITULO"	TITULO es el nombre de un cartucho.

CON LA NUEVA INTERFACE, PODRAN CONECTARSE HASTA 64 SPECTRUM

bytes), o se ejecuta una instrucción CLOSE. En nuestro caso, pues, la grabación no se consumaría si no añadimos una orden CLOSE#4. Al igual que para OPEN, no deben cerrarse las rutas 0-3.

Para recuperar ahora los datos almacenados en el fichero "números" del Microdrive, se podría utilizar el siguiente programa:

```
10 OPEN#4; "m"; 1; "números"
20 FOR b = TO 10
30 INPUT#4; m, n
40 PRINT "el cuadrado de "; m; "es"; n
50 NEXT b
60 CLOSE #4
```

También se puede utilizar INKEY\$ para leer de un fichero. Esta función devuelve siempre el siguiente carácter:

Un aspecto a resaltar cuando se utiliza INPUT, es que el ordenador espera siempre un ENTER después de un número o una cadena de caracteres. Así pues, a la hora de grabar los datos, cada ítem que luego se quiera leer con INPUT ha de escribirse independiente, esto es:

```
10 PRINT#4;2
20 PRINT#4;3
o separándolos con un apóstrofe como
en
10 PRINT#4;2,3
```

Hay que tener cuidado cuando se trata de introducir una cadena de caracteres que contenga comillas. La manera más segura de manejarlas es utilizando LINE a\$ en lugar de simplemente a\$.

Después de utilizar un canal que no sea la pantalla, puede comprobarse que las órdenes PAPER e INK no funcionan. Esto se arregla introduciendo PRINT; antes de fijar los colores.

Los ficheros quedan automáticamente catalogados (hasta 50 por cartucho) y la lista que se obtiene con CAT puede sacarse por una ruta cualquiera, lo que permite, por ejemplo, imprimir el contenido de un determinado cartucho.

Para "ocultar" un fichero, esto es, para que no salga en la lista que proporciona CAT, basta con añadir CHR\$(80) a su nombre. Así por ejemplo, con la instrucción OPEN#4; "m"; 1; CHR\$(80)+"PCW" nos aseguramos de que el fichero PCW no va a aparecer en el directorio del cartucho que estamos utilizando.

Por otra parte, en las pruebas se ha podido observar que cuando se excede el límite de 50 ficheros por cartucho, los nombres de los ficheros empiezan a desaparecer de una manera aparentemente aleatoria, aunque continúan ocupando espacio y todavía se pueden cargar.

Una importante instrucción dirigida al Microdrive es MOVE. Como su nombre sugiere, permite trasladar los datos de un canal a otro. Para trasladar datos del teclado a la pantalla, por ejemplo, se podría escribir MOVE #1 TO #2. Cualquier cosa que ahora se teclee, aparecerá en la pantalla del televisor.

MOVE puede utilizarse para examinar el contenido de un fichero; MOVE "m"; 1; "números" TO #2, sacará por pantalla el contenido del fichero "números". MOVE también permite copiar ficheros:

```
MOVE "m"; 1; "números" TO "m"; 1;
"números 2".
```

copiará el contenido del fichero "números" en el fichero "número 2". Cambiando simplemente el número de Microdrive, se pueden hacer copias de

seguridad de ficheros de datos en otro cartucho. Por ejemplo:

```
MOVE "m"; 1; "números" TO "m"; 2;
"números 2".
```

Este tipo de copias sólo sirve para ficheros de datos. Para copiar programas es preciso cargarlos en el Spectrum y luego volver a guardarlos con SAVE.

Tiempos

Aunque no se trata de pruebas exhaustivas de laboratorio, las cifras que aportamos pueden muy bien servir de indicación comparativa de prestaciones del cassette y el Microdrive. Tratándose de una cinta magnética, la modalidad de acceso a la información de que puede hablarse es del acceso secuencial, más que del auténticamente aleatorio de los discos. Para los usuarios de Spectrum que previamente han trabajado con cassettes, el aumento de velocidad resultará sencillamente fenomenal, aunque los que estén acostumbrados a los discos quizá no les cause demasiado placer los tiempos de espera para la localización y carga de un fichero. El acceso secuencial supone que los tiempos de búsqueda pueden ser incluso superior a la propia carga. El tiempo máximo de búsqueda es de 3,5 segundos, dependiendo lógicamente en cada caso, de la posición del fichero en el soporte.

Se han utilizado para las pruebas de velocidad una selección de programas en BASIC y en lenguaje de máquina. Primero se cargaron desde el cassette, anotándose el tiempo empleado, se volvieron a guardar en el Microdrive, para, finalmente, pasarlos de nuevo al Spectrum. Como era de esperar, las diferen-

FORMAT "N"; X

N significa red (network) y X estación número 0 a 64.

PRINT # Z

Da salida a una corriente especificada.

FORMAT "T"; S

T indica RS232 texto y S es la velocidad de transferencia en baudies.

SAVE *; CHAN

Guarda (save) programas, datos o codificaciones solamente en canal B, M o N.

FORMAT "B"; S

B es para RS232 para datos binarios.

VERIFY *; CHAN

1 Corriente: La rutina que va de un canal a otro es llamada una corriente. En un sistema basado en Spectrum existen 16 corrientes. Canales (Channels): Los datos pueden ser enviados a diferentes dispositivos, la pantalla, la impresora ZX, fichero en Microdrive, otro Spectrum de la misma red y el interfase RS232. Los datos pueden ser recibidos desde los siguientes dispositivos: el teclado, un fichero en Microdrive, otro Spectrum de la misma red y el interfase RS232. Estos dispositivos son conocidos como canales.

INKEY\$ # Z
INPUT # Z; Var
LOAD * CHAN

Devuelve un carácter o nulo
Introduce una variable.
Carga programas, datos o ficheros de canales B, M o N.

MERGE * CHAN
MOVE desde TO destino
OPEN # Z; CHAN

Solamente desplaza datos.
Enlaza corriente a canal.

NOVEDADES

	Cassette	Microdrive
Programa Basic (35K)	3.25.0	7,8 seg.
Programa en cod. máquina .. (7000 bytes)	40.0	7,5 seg.
Datos (DIM a(4000))	1.21.30	6,8 seg.
SCREEN\$	32.0	3,8 seg.

cias de tiempo más notables se dieron con programas largos; un programa de juegos de 35k, por ejemplo, cuya carga desde cassette toma 3 minutos 25 segundos, tardó algo menos de 8 segundos desde el Microdrive.

Documentación

El manual de los Microdrives y de la Interface 1 es comprensible y fácil de seguir (al menos en su versión inglesa, que hemos consultado). Incluye una completa lista de las ampliaciones del BASIC y de las variables del sistema utilizadas por los Microdrives, la red en área local y la interfase RS232, junto con una descripción de la RAM de usuario que utilizan los nuevos periféricos. Cada vez que se abre un fichero, se reserva una zona designada como CHANS de 595 bytes (que incluyen los 512 del buffer del Microdrive). Para la red, este área de 276 bytes. Es poco probable que esto dé problemas ni siquiera a los programas más largos, pero tendrá efecto sobre las rutinas en código de máquina almacenadas en instrucciones

REM (las rutinas situadas por encima de RAMTOP no se verán afectadas).

Conclusiones

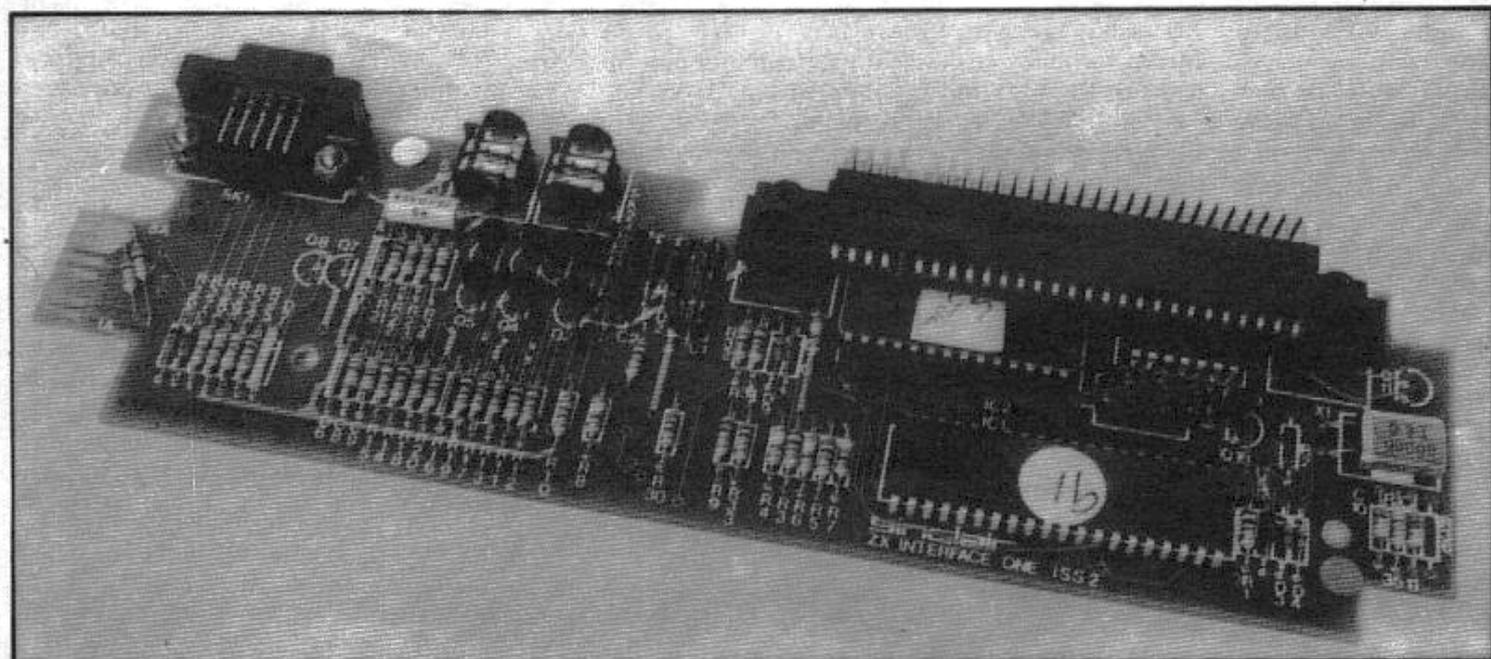
Aunque el nuevo periférico tiene ligeramente menos capacidad, cuesta un poco más caro y es algo más lento que lo esperado, podemos decir que con el Microdrive, Sinclair ha conseguido que por primera vez los propietarios de un home computer tengan acceso a una memoria masiva de bajo coste y con una velocidad de transferencia considerable, comparativamente hablando. Esto supone la apertura de una puerta hacia un montón de aplicaciones "serias" para el Spectrum y va a tener el efecto lateral de alargar la vida de esta máquina en el mercado.

Es una pena que los comandos relativos al Microdrive sean tan retorcidos, pero resulta comprensible si se piensa que se ha querido conservar la ya familiar sintaxis de las instrucciones para el cassette, con la correspondiente amplia-

ción para indicar que se refieren a la nueva unidad.

Por esta razón, las teclas de función serían una auténtica bendición: sería bonito poder disponer de teclas independientes para cargar, guardar y verificar un fichero. También resulta muy incómodo que no haya una manera directa de copiar programas. Los ficheros de datos se pueden copiar utilizando la instrucción MOVE, pero los programas es preciso cargarlos y volverlos a guardar con SAVE para obtener una copia.

Conectado al Microdrive, el Spectrum puede utilizarse como un pequeño equipo de gestión, que dispondrá de aplicaciones incorporadas como pueden ser el tratamiento de textos y la hoja electrónica de datos. Pero, lo que es más importante, los Microdrives van a ser sin duda decisivos a la hora de fijar un standard en almacenamiento de datos para microordenadores pequeños y baratos. Con un coste de 50 libras por unidad y de 30 para la interfase (cuando se compra con una unidad), el Microdrive va a ponérselo muy cuesta arriba a los fabricantes que pretendan cobrar los actuales precios por una sencilla unidad de disco. Sinclair ha supuesto un importante factor en la difusión de la microinformática entre el gran público, y el Microdrive no hace sino continuar en esta línea. Más potencia para sus proyectos. ¡Y recuerde que muy pronto el protagonista será el ZX83!



GANES 5.000 PESETAS

**MENSUALMENTE
PARTICIPANDO EN NUESTRO CONCURSO**

A partir del próximo número, 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



E.T.: EN BICI HASTA LA LUNA

PROGRAMAS

¿Recuerda la película E.T. (El extraterrestre)? Este juego le permite interceptar el vuelo del "bichito" cuando atraviesa la mitad de la superficie lunar. Por supuesto que no necesita haber visto la película para disparar, lo cual hace presionando la tecla "f". También le permite la posibilidad de atraparlo —si lo dió— usando las teclas 5 y 8.

Los disparos están muy bien logrados, lo cual no se puede decir de la imagen de la bicicleta atravesando la luna que es susceptible de mejoras, especialmente en el momento del impacto.

Notas gráficas:
50 - Graphic A y B.
60 - Graphic C y D.

(16 K - Spectrum)

```

10 BORDER 0: PAPER 0: INK 7
15 GO SUB 9000
20 CLS
25 GO SUB 1000
26 PRINT AT 21,0; "PUNTO DE PARTIDA"
27 GO TO 3000
30 LET z=12
40 FOR b=8 TO 15
50 PRINT AT z,b; INVERSE 1;";"
60 PRINT AT z+1,b; INVERSE 1;";"
70 FOR x=1 TO 50
75 NEXT x
80 PRINT AT z,b; INVERSE 1;";"
90 PRINT AT z+1,b; INVERSE 1;";"
95 LET z=z-1
96 IF INKEY$="f" THEN GO SUB 2
100 NEXT b
105 IF INKEY$="f" THEN GO SUB 2
110 FOR b=15 TO 22
120 PRINT AT z,b; INVERSE 1;";"
130 PRINT AT z+1,b; INVERSE 1;";"
140 FOR x=1 TO 50
145 NEXT x
150 PRINT AT z,b; INVERSE 1;";"
160 PRINT AT z+1,b; INVERSE 1;";"
170 LET z=z+1
180 IF INKEY$="f" THEN GO SUB 2
200 NEXT b
210 RETURN
220 PRINT AT 21,0; FLASH 1; "Fatal"
230 STOP
1000 FOR a=1 TO 80
1010 INK 7: CIRCLE 128,68,a
1020 NEXT a
1030 RETURN
2000 LET r=1
2010 LET d=30
2030 FOR s=10 TO 4 STEP -1
2035 INVERSE 1
2040 PRINT AT s,r;"/";AT s,d;"\"
2045 PAUSE 2
2049 INVERSE 0
2050 PRINT AT s,r;"■";AT s,d;"■"
2051 IF s>13 THEN PRINT AT s,r;";"
;AT s,d;" "

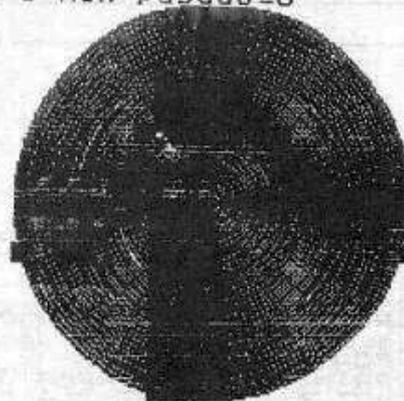
```

```

2055 LET r=r+1
2060 LET d=d-1
2070 BEEP .1,s
2080 NEXT s
2090 IF b=15 AND z=4 THEN GO TO
3500
3000 RETURN
3500 RESTORE 9900
3510 LET score=score+1
3520 FOR k=1 TO 11
3530 READ beep
3540 BEEP 2,beep+15
3550 NEXT k
3560 LET catch=16
3590 NEXT k
3600 LET fall=INT (RND*3)+13
3610 FOR g=4 TO 20
3620 LET catch=catch-(INKEY$="5"
)+(INKEY$="8")
3630 PRINT AT g,fall; INVERSE 1;
";"AT g+1,fall;" ";AT 21,catc
h; INVERSE 0;";"
3640 PRINT AT g,fall;"■";AT g+1
,fall;"■";AT 21,catch;" "
3650 NEXT g
3660 IF catch=fall THEN LET scor
e=score+10
3670 PRINT AT 21,0;";"
3700 RETURN
8000 LET score=0: LET et=0
8020 FOR e=1 TO 25
8030 PRINT AT 0,0;"Punt.=";score
; " Han pasado=";et
8040 GO SUB 30
8050 LET et=et+1
8070 IF score>90 THEN PRINT AT
21,0; FLASH 1;"Ya no quedan mas!"
8100 STOP
8100 NEXT e
9000 FOR v=1 TO 4: READ a$: FOR
w=0 TO 7: READ c: POKE USA a#+w,
c: NEXT w: NEXT v: RETURN
9100 DATA "a",3,3,7,8,20,35,20,8
9110 DATA "b",0,0,102,64,160,16,
160,64
9115 DATA "c",0,0,0,0,0,0,0,1
9120 DATA "d",0,0,0,0,112,112,12
,120
9900 DATA 3,3,3,3,6,4,4,3,3,1,3

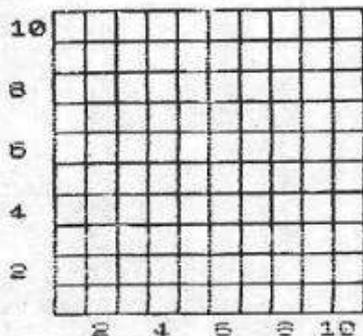
```

Punt.=3 Han pasado=6

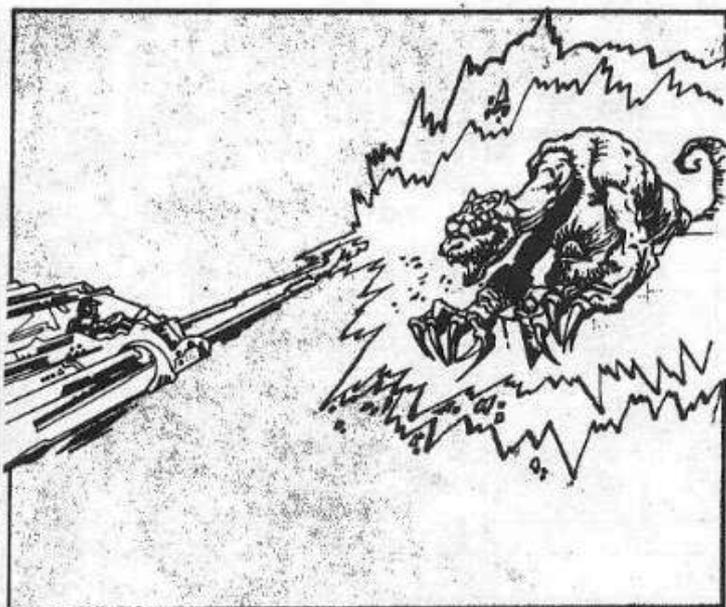


COMBATE CONTRA EL DRAGON

Una vez más tiene ante Vd. un temible monstruo que ha de eliminar. Pero para ello, primero ha de acertar donde está. En la pantalla le aparecen 100 casillas. Al hacer el disparo (introduciendo las coordenadas correspondientes a la casilla), su Spectrum le informa si ha acertado o no y en este último caso le dice además la dirección en la que se encuentra el monstruo respecto a su anterior tiro. ¿Fácil no?



(16 K - Spectrum)



```

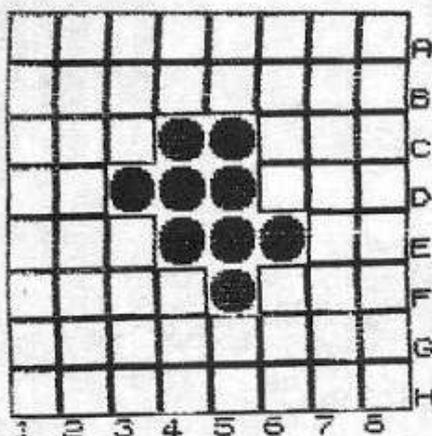
1 DEF FN e(X)=X*12+50
5 BRIGHT 1: INK 6: BORDER 0:
PAPER 0: CLS
6 GO SUB 700
7 LET q=0
8 CLS
10 FOR X=50 TO 170 STEP 12
20 PLOT X,50: DRAW 0,120
30 PLOT 50,X: DRAW 120,0
40 NEXT X
50 PRINT INK 2: PAPER 0: AT 16,
4: " 2 4 6 8 10"
60 FOR z=0 TO 21
90 PRINT INK 2: PAPER 0: AT 1,4
: "10": AT 4,4: "5": AT 7,4: "6": AT 1
0,4: "4": AT 13,4: "2"

```

```

95 RANDOMIZE
100 LET c=INT (RND*11): LET d=I
NT (RND*11): LET d=INT (RND*11):
LET c1=FN e(c): LET d1=FN e(d)
140 LET q=q+1
141 IF q>5 THEN GO TO 220
145 INPUT : INK 5: "Dónde cree q
ue está?": a$, b$: IF a$="c" OR a
$="d" THEN GO TO 146
147 IF q>1 THEN PRINT AT 16,0,"
T AT 19,0: "
157 IF a$="" OR b$="" THEN GO T
O 146
158 LET a=VAL (a$): LET b=VAL (
b$): IF a<0 OR a>10 OR b<0 OR b>
10 THEN GO TO 146
159 LET a=FN e(a): LET b=FN e(b
)
160 PLOT INK 1:a,b
161 BEEP .5,5
170 IF a=c1 AND b=d1 THEN GO TO
500
180 PRINT AT 16,0:"Norte " AND
b<d1: "Sur " AND b>d1: "Oeste " A
ND a>c1: "Este " AND a<c1
210 GO TO 140
220 PRINT AT 16,0:"Ha agotado s
us tiradas. El monstruo estaba e
n (" : c: ", d: ")."
225 PLOT FLASH 1: INK 4:c1,d1
229 PRINT AT 21,0:"Repetimos s/
n)"
230 READ a: IF a=-99 THEN RESTO
RE : READ a
231 READ b: BEEP a,b
239 LET a$=INKEY$: IF a$="" THE
N GO TO 230
240 IF a$="s" THEN GO TO 7
250 PRINT "Otra vez sera!!": ST
OP
500 PRINT AT 16,0:"Bravo! Lo lo
gro en " : q: " tiradas."
510 GO TO 225
600 DATA 1,0,1,2,.5,3,.5,2,1,0,
1,0,1,2,.5,3,.5,2,1,0,1,3,1,5,2,
7,1,3,1,5,0,7,.75,7,.25,6,.5,7,.
5,5,.5,3,.5,2,1,0,.75,7,.25,6,.5
,7,.5,5,.5,3,.5,2,1,0,1,0,1,-5,2
,0,1,0,1,-5,2,0,-99
700 PRINT "No se asuste, pero h
ay un mons- truo dentro de una d
e las casi- llas de dimension 10
*10. Inten- te adivinar donde es
ta."
720 PRINT "Si no acierta le doy
una pista: la direccion en la q
ue se en- contra en su ultim
o intento"
730 PRINT "Buena suerte!!"
740 PRINT : PRINT "Se me olvida
ba decirle que debe introducir c
ada coordenada de forma separa
da (2,2 por ejemplo) y pulsar ENT
ER"
750 INPUT "Pulse ENTER para com
enzar.":a$
760 RETURN

```



OTHELO

Si conoce el juego Othelo y tiene un compañero de juego esta es su oportunidad ya que este programa es para dos, blanco y negro, jugando alternativamente, ocupando un espacio libre cerca de las posiciones del contrario. Cada movimiento, para ser válido, ha de capturar una o más de las piezas del contrario, como consecuencia de atrapar al contrario entre sus piezas, bien horizontal, vertical, diagonal o en cualquiera de estas direcciones al mismo tiempo. Las piezas capturadas pasan a tener el color contrario (se pasan al enemigo!).

El juego acaba cuando ningún jugador puede mover o cuando un jugador ha perdi-

do todas sus piezas. El ganador es el jugador con mayor número de piezas al finalizar la partida.

Para introducir sus movimientos sólo ha de pulsar la letra y número correspondiente. No es necesario pulsar la tecla "Enter". Después con la opción cambio Vd. puede cambiar las "posiciones atrapadas" y/o pulsar 0 al terminar. Lástima que esto no lo haga una subrutina de forma automática. ¿Se atreve Vd. con ella?

Notas gráficas:

- 100 - Graphic E, F, E...
- 110 - Graphic H, G, H...
- 410 - Graphic, A, B.
- 420 - Graphic C, D.

(16 K - Spectrum)

```

5 BORDER 7: PAPER 7: CLS
6 LET q=0
7 FOR x=USR "a" TO USR "h"+7
8   FOR y=USR "a" TO USR "h"+7
9     DATA 0,7,31,63,127,127,1
10    DATA 0,7,31,63,127,127,1
11    DATA 0,7,31,63,127,127,1
12    DATA 0,7,31,63,127,127,1
13    DATA 0,7,31,63,127,127,1
14    DATA 0,7,31,63,127,127,1
15    DATA 0,7,31,63,127,127,1
16    DATA 0,7,31,63,127,127,1
17    DATA 0,7,31,63,127,127,1
18    DATA 0,7,31,63,127,127,1
19    DATA 0,7,31,63,127,127,1
20    DATA 0,7,31,63,127,127,1
21    DATA 0,7,31,63,127,127,1
22    DATA 0,7,31,63,127,127,1
23    DATA 0,7,31,63,127,127,1
24    DATA 0,7,31,63,127,127,1
25    DATA 0,7,31,63,127,127,1
26    DATA 0,7,31,63,127,127,1
27    DATA 0,7,31,63,127,127,1
28    DATA 0,7,31,63,127,127,1
29    DATA 0,7,31,63,127,127,1
30    DATA 0,7,31,63,127,127,1
31    DATA 0,7,31,63,127,127,1
32    DATA 0,7,31,63,127,127,1
33    DATA 0,7,31,63,127,127,1
34    DATA 0,7,31,63,127,127,1
35    DATA 0,7,31,63,127,127,1
36    DATA 0,7,31,63,127,127,1
37    DATA 0,7,31,63,127,127,1
38    DATA 0,7,31,63,127,127,1
39    DATA 0,7,31,63,127,127,1
40    DATA 0,7,31,63,127,127,1
41    DATA 0,7,31,63,127,127,1
42    DATA 0,7,31,63,127,127,1
43    DATA 0,7,31,63,127,127,1
44    DATA 0,7,31,63,127,127,1
45    DATA 0,7,31,63,127,127,1
46    DATA 0,7,31,63,127,127,1
47    DATA 0,7,31,63,127,127,1
48    DATA 0,7,31,63,127,127,1
49    DATA 0,7,31,63,127,127,1
50    DATA 0,7,31,63,127,127,1
51    DATA 0,7,31,63,127,127,1
52    DATA 0,7,31,63,127,127,1
53    DATA 0,7,31,63,127,127,1
54    DATA 0,7,31,63,127,127,1
55    DATA 0,7,31,63,127,127,1
56    DATA 0,7,31,63,127,127,1
57    DATA 0,7,31,63,127,127,1
58    DATA 0,7,31,63,127,127,1
59    DATA 0,7,31,63,127,127,1
60    DATA 0,7,31,63,127,127,1
61    DATA 0,7,31,63,127,127,1
62    DATA 0,7,31,63,127,127,1
63    DATA 0,7,31,63,127,127,1
64    DATA 0,7,31,63,127,127,1
65    DATA 0,7,31,63,127,127,1
66    DATA 0,7,31,63,127,127,1
67    DATA 0,7,31,63,127,127,1
68    DATA 0,7,31,63,127,127,1
69    DATA 0,7,31,63,127,127,1
70    DATA 0,7,31,63,127,127,1
71    DATA 0,7,31,63,127,127,1
72    DATA 0,7,31,63,127,127,1
73    DATA 0,7,31,63,127,127,1
74    DATA 0,7,31,63,127,127,1
75    DATA 0,7,31,63,127,127,1
76    DATA 0,7,31,63,127,127,1
77    DATA 0,7,31,63,127,127,1
78    DATA 0,7,31,63,127,127,1
79    DATA 0,7,31,63,127,127,1
80    DATA 0,7,31,63,127,127,1
81    DATA 0,7,31,63,127,127,1
82    DATA 0,7,31,63,127,127,1
83    DATA 0,7,31,63,127,127,1
84    DATA 0,7,31,63,127,127,1
85    DATA 0,7,31,63,127,127,1
86    DATA 0,7,31,63,127,127,1
87    DATA 0,7,31,63,127,127,1
88    DATA 0,7,31,63,127,127,1
89    DATA 0,7,31,63,127,127,1
90    DATA 0,7,31,63,127,127,1
91    DATA 0,7,31,63,127,127,1
92    DATA 0,7,31,63,127,127,1
93    DATA 0,7,31,63,127,127,1
94    DATA 0,7,31,63,127,127,1
95    DATA 0,7,31,63,127,127,1
96    DATA 0,7,31,63,127,127,1
97    DATA 0,7,31,63,127,127,1
98    DATA 0,7,31,63,127,127,1
99    DATA 0,7,31,63,127,127,1
100 DATA 0,7,31,63,127,127,1
101 DATA 0,7,31,63,127,127,1
102 DATA 0,7,31,63,127,127,1
103 DATA 0,7,31,63,127,127,1
104 DATA 0,7,31,63,127,127,1
105 DATA 0,7,31,63,127,127,1
106 DATA 0,7,31,63,127,127,1
107 DATA 0,7,31,63,127,127,1
108 DATA 0,7,31,63,127,127,1
109 DATA 0,7,31,63,127,127,1
110 DATA 0,7,31,63,127,127,1
111 DATA 0,7,31,63,127,127,1
112 DATA 0,7,31,63,127,127,1
113 DATA 0,7,31,63,127,127,1
114 DATA 0,7,31,63,127,127,1
115 DATA 0,7,31,63,127,127,1
116 DATA 0,7,31,63,127,127,1
117 DATA 0,7,31,63,127,127,1
118 DATA 0,7,31,63,127,127,1
119 DATA 0,7,31,63,127,127,1
120 DATA 0,7,31,63,127,127,1
121 DATA 0,7,31,63,127,127,1
122 DATA 0,7,31,63,127,127,1
123 DATA 0,7,31,63,127,127,1
124 DATA 0,7,31,63,127,127,1
125 DATA 0,7,31,63,127,127,1
126 DATA 0,7,31,63,127,127,1
127 DATA 0,7,31,63,127,127,1
128 DATA 0,7,31,63,127,127,1
129 DATA 0,7,31,63,127,127,1
130 DATA 0,7,31,63,127,127,1
131 DATA 0,7,31,63,127,127,1
132 DATA 0,7,31,63,127,127,1
133 DATA 0,7,31,63,127,127,1
134 DATA 0,7,31,63,127,127,1
135 DATA 0,7,31,63,127,127,1
136 DATA 0,7,31,63,127,127,1
137 DATA 0,7,31,63,127,127,1
138 DATA 0,7,31,63,127,127,1
139 DATA 0,7,31,63,127,127,1
140 DATA 0,7,31,63,127,127,1
141 DATA 0,7,31,63,127,127,1
142 DATA 0,7,31,63,127,127,1
143 DATA 0,7,31,63,127,127,1
144 DATA 0,7,31,63,127,127,1
145 DATA 0,7,31,63,127,127,1
146 DATA 0,7,31,63,127,127,1
147 DATA 0,7,31,63,127,127,1
148 DATA 0,7,31,63,127,127,1
149 DATA 0,7,31,63,127,127,1
150 DATA 0,7,31,63,127,127,1
151 DATA 0,7,31,63,127,127,1
152 DATA 0,7,31,63,127,127,1
153 DATA 0,7,31,63,127,127,1
154 DATA 0,7,31,63,127,127,1
155 DATA 0,7,31,63,127,127,1
156 DATA 0,7,31,63,127,127,1
157 DATA 0,7,31,63,127,127,1
158 DATA 0,7,31,63,127,127,1
159 DATA 0,7,31,63,127,127,1
160 DATA 0,7,31,63,127,127,1
161 DATA 0,7,31,63,127,127,1
162 DATA 0,7,31,63,127,127,1
163 DATA 0,7,31,63,127,127,1
164 DATA 0,7,31,63,127,127,1
165 DATA 0,7,31,63,127,127,1
166 DATA 0,7,31,63,127,127,1
167 DATA 0,7,31,63,127,127,1
168 DATA 0,7,31,63,127,127,1
169 DATA 0,7,31,63,127,127,1
170 DATA 0,7,31,63,127,127,1
171 DATA 0,7,31,63,127,127,1
172 DATA 0,7,31,63,127,127,1
173 DATA 0,7,31,63,127,127,1
174 DATA 0,7,31,63,127,127,1
175 DATA 0,7,31,63,127,127,1
176 DATA 0,7,31,63,127,127,1
177 DATA 0,7,31,63,127,127,1
178 DATA 0,7,31,63,127,127,1
179 DATA 0,7,31,63,127,127,1
180 DATA 0,7,31,63,127,127,1
181 DATA 0,7,31,63,127,127,1
182 DATA 0,7,31,63,127,127,1
183 DATA 0,7,31,63,127,127,1
184 DATA 0,7,31,63,127,127,1
185 DATA 0,7,31,63,127,127,1
186 DATA 0,7,31,63,127,127,1
187 DATA 0,7,31,63,127,127,1
188 DATA 0,7,31,63,127,127,1
189 DATA 0,7,31,63,127,127,1
190 DATA 0,7,31,63,127,127,1
191 DATA 0,7,31,63,127,127,1
192 DATA 0,7,31,63,127,127,1
193 DATA 0,7,31,63,127,127,1
194 DATA 0,7,31,63,127,127,1
195 DATA 0,7,31,63,127,127,1
196 DATA 0,7,31,63,127,127,1
197 DATA 0,7,31,63,127,127,1
198 DATA 0,7,31,63,127,127,1
199 DATA 0,7,31,63,127,127,1
200 DATA 0,7,31,63,127,127,1

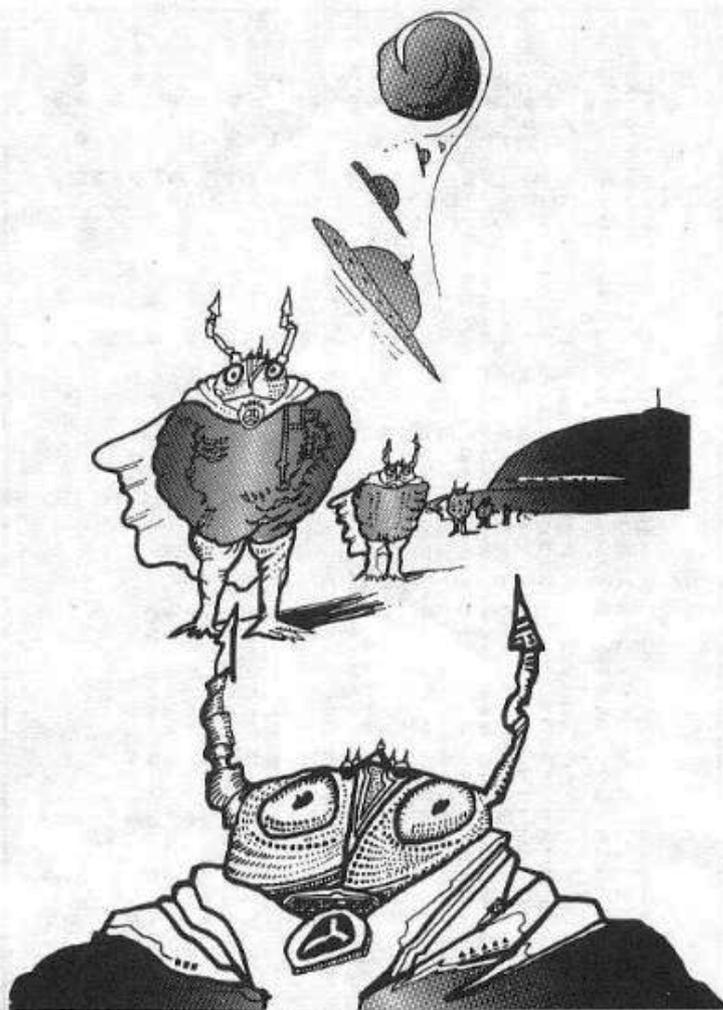
```

```

120 PRINT INK 0; PAPER 4; a$; b$;
a$; b$; a$; b$; a$; b$; a$; b$; a$;
b$; a$; b$
130 PRINT AT 16,0; "1 2 3 4 5 6
7 8"
140 FOR a=1 TO 8: PRINT AT a*2-
1,16; CHR$(64+a): NEXT a
150 GO TO 630
160 REM Entrada
170 BEEP .05,35
180 LET a$=INKEY$
190 IF q=0 THEN PRINT AT 4,16; "
Negro "
200 IF q=7 THEN PRINT AT 4,16; "
Blanco "
210 PRINT AT 5,16; "Posicion: "
220 IF a$<"a" OR a$>"h" THEN GO
TO 160
230 PRINT AT 5,27; a$
240 LET b$=a$
250 LET a$=INKEY$
260 IF a$>"8" OR a$<"1" THEN GO
TO 250
270 PRINT AT 5,28; a$
280 LET b$=b$+a$
290 REM Movimiento
300 LET c$=b$(1)
310 IF c$="a" THEN LET z=0
320 IF c$="b" THEN LET z=2
330 IF c$="c" THEN LET z=4
340 IF c$="d" THEN LET z=6
350 IF c$="e" THEN LET z=8
360 IF c$="f" THEN LET z=10
370 IF c$="g" THEN LET z=12
380 IF c$="h" THEN LET z=14
390 LET c$=b$(z)
400 LET x=VAL c$*2-2
410 PRINT AT z,x; INK q; "●"
420 PRINT AT z+1,x; INK q; "●"
430 RETURN
440 LET p=q: LET e=e+1: IF e>65
THEN GO TO 610
450 IF p=7 THEN LET q=0: LET bl
ack=black+1
460 IF p=0 THEN LET q=7: LET wh
ite=white+1
470 GO SUB 160
480 FOR q=1 TO 20
490 PRINT AT 21,0; "Introduzca s
u cambio"
500 IF q=0 THEN LET black=black
+1: LET white=white-1
510 IF q=7 THEN LET white=white
+1: LET black=black-1
520 LET a$=INKEY$
530 IF a$="0" THEN GO TO 610
540 IF a$<"a" OR a$>"h" THEN GO
TO 520
550 LET b$=a$
560 LET a$=INKEY$
570 IF a$<"1" OR a$>"8" THEN GO
TO 560
580 LET b$=b$+a$
590 GO SUB 290
600 NEXT q
610 PRINT AT 21,0; "
620 GO TO 440
625 REM Dibujo posicion inicial
630 LET z=6: LET x=6: LET q=7
640 GO SUB 410
650 LET z=8: LET x=8: LET q=0
660 GO SUB 410
670 LET z=8: LET q=0: LET x=6
680 GO SUB 410
690 LET z=8: LET x=8: LET q=7
700 GO SUB 410
710 LET white=2
720 LET black=2
730 LET e=4
740 GO TO 440
750 PRINT AT 21,0; "Blancas="; wh
ites; " Negras="; black

```

Ver nº 10, pág 47
para modificar línea 240



OTRO DE INVASORES

Un juego más de las típicas luchas contra el invasor que pretende conquistar la tierra y que, a fuego de laser y de lo que su imaginación quiera, hay que impedir por el bien de la humanidad.

Aunque un poco tedioso para pasar a la memoria de su Spectrum por su longitud, el trabajo vale la pena. La vistosidad de sus gráficos le convierten en uno de los mejores "juegos galácticos".

Requiere además, gran habilidad en el manejo de la nave para poder destruir al enemigo.

En cuanto a las sentencias utilizadas, ha de destacarse la creación del ovni a través del modo gráfico (rutina 5000). El carácter gráfico de las sentencias 300 y 1.020 se obtiene con Graphic + shift y P.

Se lo recomendamos efusivamente. Disfrute y protéjanos de este "feroz ataque".

(16 K - Spectrum)

```
3 GO SUB 5000
4 GO SUB 4000
5 LET l=16: LET k=11: LET pos
-10 6 LET m=3: LET n=INT (RND*26)
+2 8 LET score=0
9 LET lan=0
10 PLOT 0,0: DRAW 97,50
20 PLOT 255,0: DRAW -97,50
30 PLOT 137,50: DRAW 50,-50
40 PLOT 117,50: DRAW -50,-50
50 PLOT 70,50: DRAW -70,-15
60 PLOT 185,50: DRAW 70,-15
70 PLOT 0,50: DRAW 255,0
80 PLOT 127,50: DRAW 0,-50
90 PLOT 0,40: DRAW 255,0
100 PLOT 0,15: DRAW 255,0
110 PLOT 0,14: DRAW 255,0
120 INK 7: PLOT 0,0: DRAW 0,175
DRAW 255,0: DRAW 0,-175: DRAW
-255,0
125 PRINT AT k,l-1;" "
126 PRINT AT k,l+1;" "
127 PRINT AT k-1,l;" "
128 PRINT AT k+1,l;" "
130 IF INKEY$="8" AND l<29 THEN
LET l=l+1
140 IF INKEY$="5" AND l>2 THEN
LET l=l-1
150 IF INKEY$="6" AND k<13 THEN
LET k=k+1
160 IF INKEY$="7" AND k>2 THEN
LET k=k-1
180 PRINT AT INT m,INT n;" "
200 PRINT AT k,l-1;" "
210 PRINT AT k-1,l;" "
220 PRINT AT k+1,l;" "
230 PRINT AT k,l+1;" "
240 IF INKEY$="0" THEN FOR t=0
TO 3: BEEP .02,40: NEXT t: IF k=
INT m AND l=INT n THEN GO SUB 10
00
245 LET n=n+INT (RND*3)-1
246 IF n<3 THEN LET n=n+1
247 IF n>29 THEN LET n=n-1
248 BEEP .005,(RND*35)+15
250 LET m=m+.2
252 IF score>1000 THEN LET m=m+
.2: IF m>2000 THEN LET m=m+.2
255 IF n<1 OR n>31 THEN PLOT 0,
0: DRAW 0,175: DRAW 255,0: DRAW
0,-175: DRAW -255,0
260 IF m>15 THEN GO SUB 2000
```

Han conquistado la tierra
y construido sus ciudades

Su puntuacion fue 200

Ha perdido la batalla
Pulse RUN para jugar de nuevo



Nunca es tarde para intentar sa-
lvar la tierra!!!

CALCULO DE INTERESES

```

300 INK INT (RND*7)+2: PRINT AT
INT #,INT n;"*": INK 7
310 PRINT AT 1,1;"TANTOS:";score
e;"
999 GO TO 125
1020 PRINT AT k,l;"*": PRINT ; O
UER 1;AT k,l;"*
1030 FOR h=50 TO 45 STEP -.5: BE
EP .01,h: NEXT h
1040 PRINT AT k,l;" "
1050 LET #=3: LET n=INT (RND*25)
+2
1060 LET score=score+100
1080 RETURN
2000 LET lan=lan+1
2010 IF lan>10 THEN GO TO 7000
2020 RETURN
3005 FOR n=0 TO 100: BEEP .1,40:
NEXT n
4000 PAPER 1: INK 7: BORDER 1: C
LS
4010 PRINT AT 0,10; FLASH 1;"OVN
I"; FLASH 0
4020 INK 6: PRINT : PRINT "Salva
la tierra. Pretenden inva-dirta
": PRINT : PRINT "Atacalos ante
s de que la con- quisten"
4030 INK 7: PRINT : PRINT : PRIN
T " 5=Izquierda": PRINT : PRINT
" 6=Abajo": PRINT : PRINT " 7=Ar
riba": PRINT : PRINT " 8=Derecha
": PRINT : PRINT " 0=Disparo"
4040 PRINT AT 21,1; FLASH 1;"Bue
na suerte.....": FLASH 0
4045 PAUSE 9999
4050 BORDER 0: PAPER 0: INK 6: C
LS : FOR n=0 TO 50: BEEP .005,(R
ND*35)+15: INK INT (RND*7)+1: PL
OT INT (RND*250)+5,INT (RND*120)
+50: NEXT n: INK 4: GO TO 5
5000 DATA BIN 0,BIN 00011000,BIN
0111110,BIN 11011011,BIN 111111
11,BIN 01100110,BIN 0,BIN 0,BIN
0
5010 FOR n=0 TO 7: READ J
5030 POKE USR "P"+N,J: NEXT n
5040 RETURN
7000 BEEP .7,1: BEEP .5,1: BEEP
1,6
7010 PRINT AT 1,1;" Han conquist
ado la tierra"
7020 PRINT AT 2,0;"y construido
sus ciudades"
7025 PAUSE 100
7030 INK INT 2: PRINT AT 12,10;"
"
7035 BEEP .5,5
7040 PRINT AT 13,9;"■■■■"
7045 BEEP .5,4
7050 PRINT AT 14,9;"■■■■"
7055 BEEP .5,3
7060 PRINT AT 15,9;"■■■■"
7065 BEEP .5,2
7070 PRINT AT 16,8;"■■■■"
7075 BEEP .5,1
7076 INK 6
7078 PRINT AT 8,1;"Ha perdido la
batalla": PRINT "Pulse RUN para
jugar de nuevo"
7120 PRINT AT 19,1;"Nunca es tar
de para intentar salvar la tierra
■■■■"
7140 FLASH 1: PRINT AT 5,1;"Su p
ontuacion fue ";score: FLASH 0

```

Este breve programa nos permite hacer un cálculo de intereses. El ordenador va interesándose sucesivamente por una serie de datos, a saber: la cantidad implicada en el cálculo, el porcentaje anual, la cantidad de tiempo (que automáticamente es convertida en meses) y el período de actualización. Y, eso es todo.

A continuación el ZX entrará en trance, pasando a

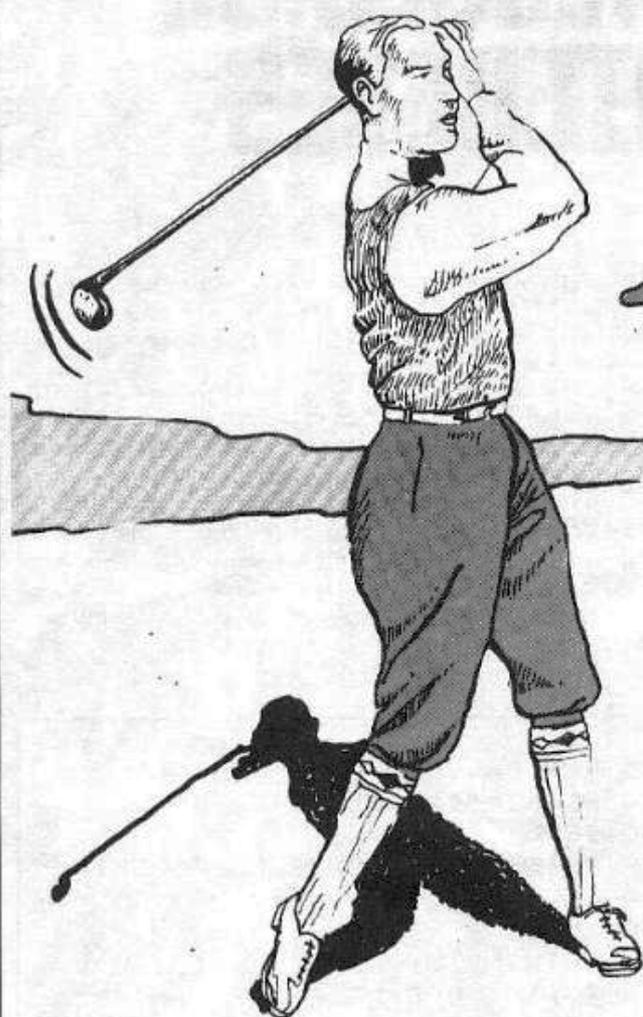
mostrarnos lo ricos que vamos a ser con un poco de tesón y paciencia.

El cálculo en sí es extremadamente simple, pero también muy útil si deseamos establecernos por nuestra cuenta y abrir un pequeño banco local. Mire usted por donde, un inocente Sinclair podría revelarse en el elemento clave de nuestro éxito en los negocios.

```

5 REM CALCULO DEL INTERES
10 PRINT "INTRODUCE EL TOTAL "
30 INPUT A
40 PRINT "INTRODUCE EL PORCETA
JE ANUAL"
50 INPUT B
60 PRINT "INTRODUCE EL TOTAL D
E ANOS, Y LOS MESES RESTANTES"
70 INPUT C
80 INPUT E
90 LET D=C*12+E
100 PRINT "INTRODUCE EL PERIODO
ENTRE CADA PLAZO EN MESES"
110 INPUT F
120 LET DD=D/F
130 FOR G=1 TO INT DD
140 LET H=INT (A*(B*F/12))/100
150 PRINT "PER. ";G; " : ";A; "+";B;
"/=";A+H
160 LET A=A+H
170 IF PEEK 16442<=2 THEN SCROL
L
180 NEXT G
190 PRINT "PULSA UNA TECLA PARA
SEGUIR"
200 IF INKEY$="" THEN GOTO 200
210 CLS
220 GOTO 10

```



GOLF EN CASA

¿Le apetece echar una partidita de Golf sin moverse de casa y sin tener que elegir el palo? El hoyo lo tendrá siempre a la vista y no importa que no sea Vd. un Severiano Ballesteros: es sólo cuestión de práctica y siempre podrá repetir el tiro cuantas veces desee.

Para lanzar la pelota coló-

quese en la posición adecuada, concéntrese e introduzca los datos: el ángulo (nunca mayor de 85) y la potencia que desea dar a su tiro.

Pruebe a mejorar su representación gráfica a fin de sentirse como si realmente estuviese en un campo de golf y... ¡Buen tiro!

(16 K - Spectrum)

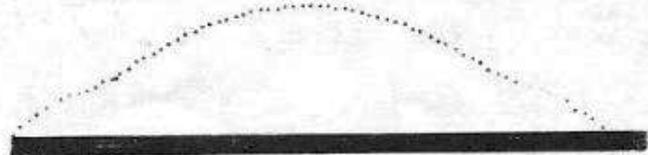
```

1 REM GOLF
3 PAPER 5
4 BORDER 2
10 PRINT TAB 13; INK 2; "GOLF@"
20 LET t=15*(1+RND)
30 PRINT AT 21.0; INK 4; "
40 PRINT AT 21,t-1; INK 5; "■"
50 INPUT "Angulo =";s
51 IF s>=85 THEN GO TO 1
60 PRINT AT 3,0;"Angulo ";s
70 INPUT "Velocidad =";d
80 PRINT "Velocidad ";d
90 LET a=d*CO5 (PI*s/180)
100 LET b=d*SIN (PI*s/180)
110 FOR J=0 TO b/16 STEP .3
115 LET c=.01*(b*J-16*J*J)
120 IF a*J>6200 THEN GO TO 190
130 IF c>40 THEN GO TO 170
140 INK 7
150 PLOT .04*a*J,4*c+8
160 BEEP .005,c+10
170 NEXT J
180 IF ABS (a*b/3200-t) < 1 THEN
GO TO 210
190 PRINT AT 10,20;"PERDIDA"
200 PAUSE 200
210 CLS
220 GO TO 1
220 PRINT AT 20-J,T-2;"HOYO":
BEEP .3,6-3*J
230 NEXT J
235 PAUSE 200
240 CLS
245 GO TO 1

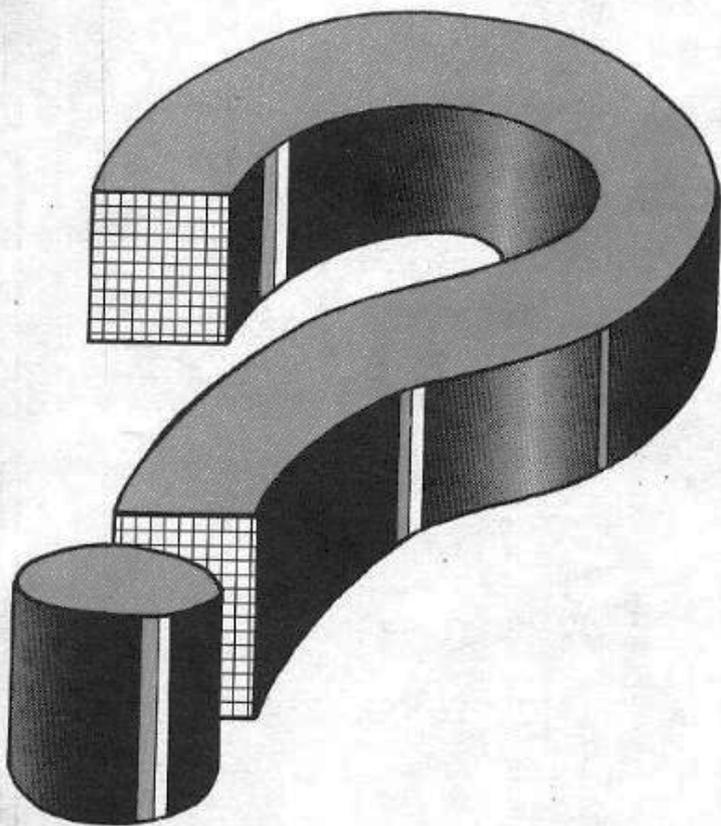
```

Angulo 50
Velocidad 440

PERDIDA



PROGRAMAS



E.S.P

ESP es el nombre de un test inventado por J. B. Rhine en 1934. Cinco símbolos aparecen en la pantalla y Ud. debe adivinar cuál es el seleccionado. Tiene 25 oportunidades para acertar, conociendo sólo que todos los

símbolos tienen la misma probabilidad de ocurrencia (20 %).

Al finalizar, le indica el porcentaje de aciertos alcanzados y su variación respecto al normal.

(16 K - Spectrum)

```

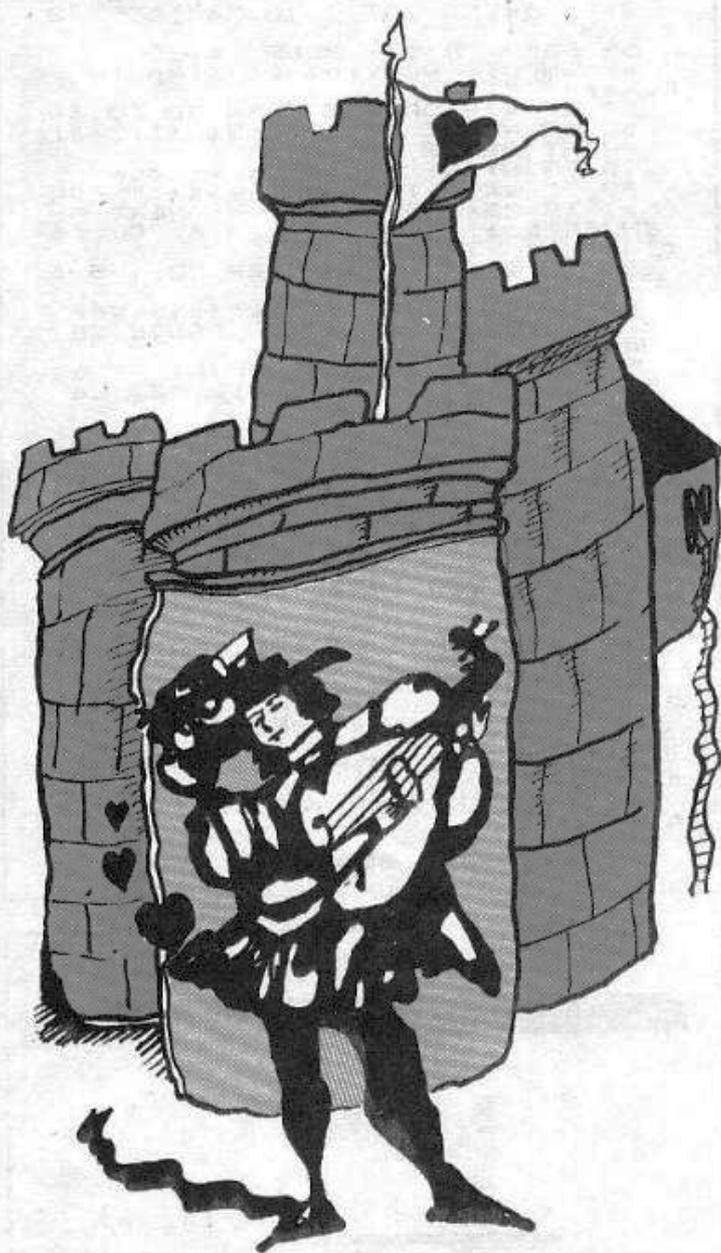
1 BORDER 7: PAPER 7: INK 0
2 INVERSE 1: INK 4: PRINT AT
1,0;"El test E.S.P. fue inventado
por J.B.Rhine en 1934": INK 0:
INVERSE 0
10 LET z=0
20 FOR t=1 TO 25
30 LET rp=INT (8*RND): IF rp=0
THEN GO TO 30
32 PRINT AT 5,6;CHR$ 144;AT 5,
10;CHR$ 145;AT 5,14;CHR$ 146;AT
5,18;CHR$ 147;AT 5,22;CHR$ 148
34 PRINT AT 7,6;"1";AT 7,10;"2
";AT 7,14;"3";AT 7,18;"4";AT 7,2
2;"5"
35 BEEP .2,14: BEEP .4,11
40 INPUT "Adivine el simbolo.
Numero? ";a
50 IF a<0 OR a>5 THEN GO TO 40
60 PRINT AT 10,10;CHR$ (143+a)
: PRINT AT 12,10;"Ud."
70 PRINT AT 12,24;"Spectrum"
80 IF a=rp THEN LET z=z+1: FOR
n=1 TO 25: BEEP .1,a+n: NEXT n:
INVERSE 1: PRINT AT 13,6;"Corre
cto=";z: INVERSE 0
81 IF a>rp THEN BEEP .3,(-5-a
)
86 PRINT AT 20,5;"Tiradas que
le quedan=";25-t;" ": PAUSE 25:
NEXT t
90 LET p=z/25*100: PRINT AT 12
,5;"Su puntuacion es ";p;"%. Lo
normal es 20%"
100 IF p>20 THEN PRINT AT 17,5;
"Su habilidad es ";(p-20);" %sob
re lo normal"
105 IF t=0 THEN STOP
110 FOR n=144 TO 148
120 FOR f=0 TO 7
125 READ X
130 POKE USR CHR$ (n)+f,x
140 NEXT f
150 NEXT n
160 DATA 24,36,66,129,129,66,36
,24
170 DATA 82,137,82,137,82,137,8
2,24
180 DATA 24,24,24,255,255,24,24
,24
190 DATA 153,90,60,255,60,90,15
3,0
200 DATA 255,129,129,129,129,12
9,129,255
210 STOP
    
```

```

El test E.S.P. fue inventado por
J.B.Rhine en 1934.

1 2 3 4 5
6
Ud.
Correcto

Tiradas que le quedan= 17
    
```



Con este juego, podrá rescatar a su amada que se encuentra prisionera en la torre. Para ello dispone de un trampolín y ha de calcular bien el impulso ya que si se equivoca se estrellará contra las duras piedras de la torre. No es difícil y después de varios golpes se alcanza el objetivo: llegar al balcón

donde ella le espera. ¡Apresúrese... ella le espera!

Notas gráficas:
 15, 40, 46 - Graphic A.
 18 - Graphic B.
 23 - Graphic C.
 240 - Graphic D.
 60, 210 - Graphic E.

(16 K - Spectrum)

```

1 PAPER 0: INK 7: BORDER 0: C
LS
2 DATA 56,56,146,254,16,124,6
  8,198
3 DATA 0,255,171,171,171,171,
  255,255
4 DATA 0,0,28,20,62,62,127,12
  7
5 DATA 0,0,0,0,24,126,255,255
6 DATA 192,224,224,240,240,22
  4,224,192
7 FOR n=144 TO 148
8 FOR f=0 TO 7: READ a: POKE
  USR CHR$ n+f,a: NEXT f
9 NEXT n
10 PRINT AT 4,0: INK 6;"█"
  "█":AT 5,0;"█":AT 6,0;"█"
11 FOR n=7 TO 21: PRINT AT n,1
  : INK 6:"█":NEXT n
12 PRINT AT 11,13: INK 4:"█"
  :AT 12,15:"█":AT 13,15:"█":AT 14
  :AT 15,15:"█":AT 16,15:"█":AT 17,15:"█"
  :AT 18,15:"█":AT 19,15:"█"
  :AT 20,15:"█":AT 21,15:"█"
13 FOR n=11 TO 21: PRINT AT n,
  15: INK 4:"█":NEXT n
14 LET x=21: LET y=6: PRINT AT
  21,10:"O":PLOT 64,0: DRAW 37,1
5
15 PRINT AT x,y: INK 5:"▲"
17 LET h=INT (RND*11)+8
18 PRINT AT h+1,6:"■"
20 PRINT AT 4,11:"peso?":AT 5,
  11:"(Mx.14)"
22 INPUT w: IF w<=0 OR w>=15 T
  HEN GO TO 22
23 FOR n=10 TO 20: PRINT AT n-
  1,12:"█":AT n,12:"▲":BEEP .1,n:
  NEXT n
30 PLOT 64,0: DRAW OVER 1,37,1
6: PLOT 64,16: DRAW 37,-16
35 FOR n=1 TO w
36 LET x=x-1
40 PRINT AT x,y:"▲":AT x+1,y:"
  "
45 NEXT n
46 FOR n=1 TO 2: LET y=y-1: PR
  INT AT x,y:"▲":NEXT n
50 IF x=h THEN GO TO 100
55 IF x>h THEN GO TO 200
60 IF x<=h THEN PRINT AT x,y:
  INK 5:"█":BEEP 1,-10:GO TO 200
100 PRINT AT x,y+1:"BRAVO!":G
  O TO 300
  
```

ROMEO RESCATA A JULIETA...

PROGRAMAS

ARCO IRIS

Si le gustan los dibujos geométricos, este es un sencillo programa con el que

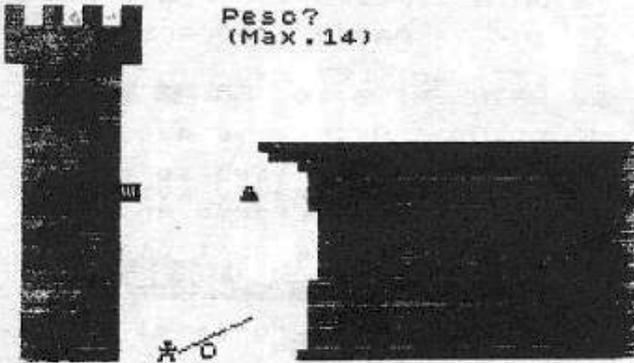
podrá generar gráficos geométricos de forma aleatoria y con bonitos colores.

(16 K - Spectrum)

```

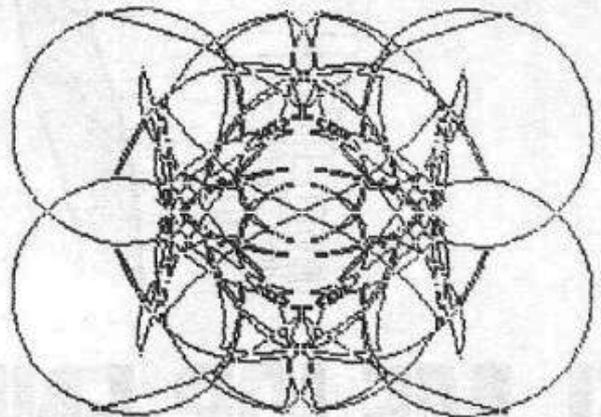
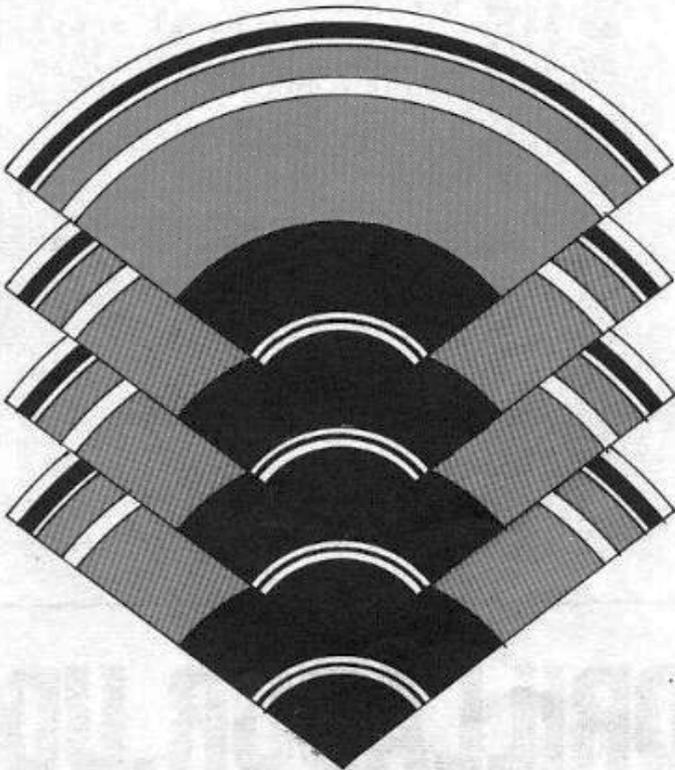
110 FOR n=0 TO 40 STEP 10: BEEP
.1,n: NEXT n
120 CLS
130 RUN
200 FOR n=x TO 21
210 PRINT AT n,y; INK 5;">";AT
n-1,y;""
220 BEEP .05,n
230 NEXT n
240 PRINT AT 21,y; INK 5;"▲"
250 BEEP 1,-10
300 PRINT AT 21,0; BRIGHT 1; FL
RSH 1;"PRESIONE 's' PARA REPETIR"
310 LET r#=INKEY#
320 IF r#="s" OR R#="S" THEN RU
N
330 GO TO 310
    
```

Peso?
(Max.14)



```

10 RANDOMIZE : BORDER 0: PAPER
0: INK 7: OVER 1
20 LET d=15: DIM a(d,3)
30 LET y1=84: LET x1=123: PRIN
T AT 10,12;"PENSANDO"
40 FOR b=1 TO d
50 LET x=213-INT (RAND*426): LE
T y=175-INT (RAND*350)
70 IF x1+x>213 OR y1+y>168 OR
x1+x<38 OR y1+y<=0 THEN GO TO 60
80 LET c=6*RND*(1-2*(y1>84 AND
x<0 OR y1<84 AND x>0)): LET c=c
*2/(2+(ABS x>50 OR ABS y>50))
90 LET x1=x1+x: LET y1=y1+y: L
ET a(b,1)=x: LET a(b,2)=y: LET a
(b,3)=c
100 NEXT b
120 CLS : FOR a=-1 TO 1 STEP 2:
FOR c=-1 TO 1 STEP 2
130 PLOT 123,84
140 FOR b=1 TO d
150 DRAW a*a(b,1),c*a(b,2),a*c
a(b,3)
160 NEXT b
170 NEXT c: NEXT a
190 LET p=0: LET c=22895: FOR a
=0 TO 352 STEP 32: FOR b=0 TO 13
200 LET at=INT (8*RND)
210 LET at=at+5+(at<2)+64
220 POKE c+b+a,at: POKE c+b-a,a
t: POKE c-b+a,at: POKE c-b-a,at
230 NEXT b: NEXT a
    
```



Seguro que conoce este juego de verlo en los videojuegos comerciales. Ahora puede tenerlo en su casa sin perder calidad y a todo color, a través de este programa que es uno de los mejores juegos por sus estupendos gráficos y la posibilidad de luchar inteligentemente contra la máquina.

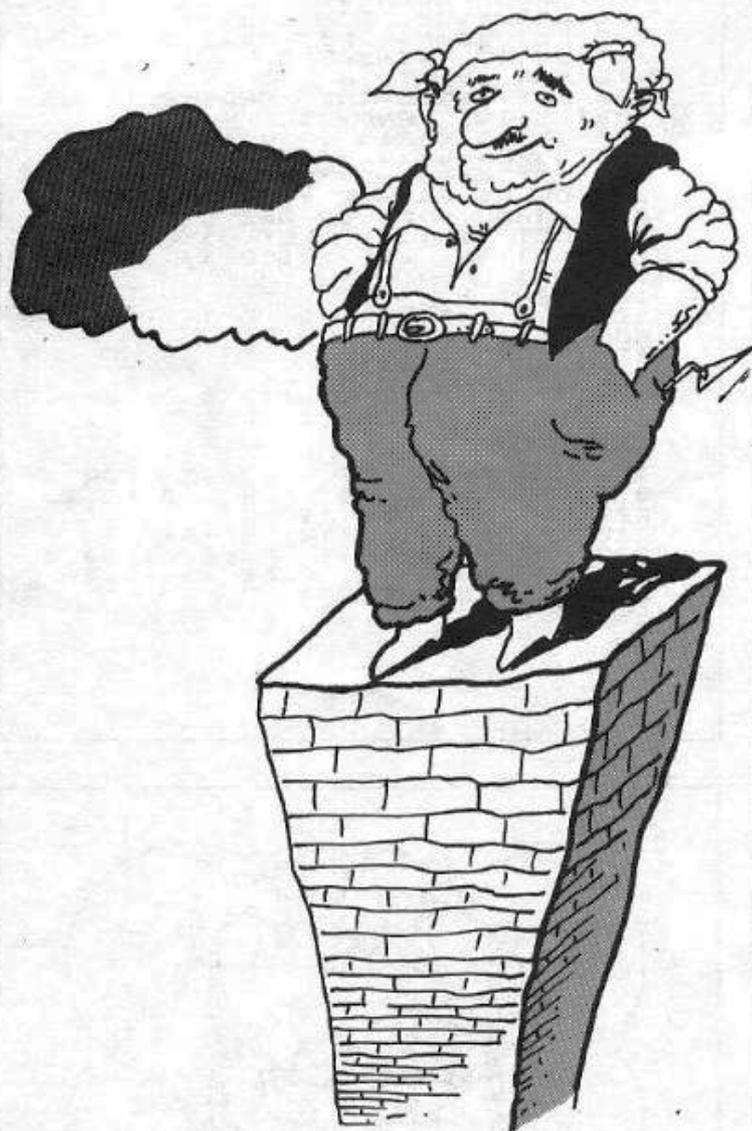
Por si no lo recuerda, ha de construir un muro, al mis-

mo tiempo que su Spectrum construye otro. El que quede encerrado es el que pierde. El movimiento se realiza a través de las teclas:

Q = Arriba.
Z = Abajo.
I = Izquierda.
P = Derecha.

Notas gráficas:
2120, 20 - Graphic A.

(16 K - Spectrum)



```

1 REM Wall race
4 FOR e=0 TO 7: READ r: POKE
5 "a"+e,r: NEXT e
6 DATA 126,126,126,0,231,231,
10 INK 7: PAPER 1: BORDER 1: C
10 LET s=0: LET o=0
20 PRINT AT 0,10: "EL MURC "
30 PRINT "Ud. tiene que const
40 PRINT "A su lado su rival
41 PRINT "El primero en choca
50 PRINT "puede controlar su
movimienti con 'Q' (ARRIBA),
Z' (ABAJO), 'I' (IZQUIERDA), 'P'
(DERECHA)
55 PRINT "Su muro es el verde
60 PRINT "Su Spectrum los colo
ca amarillos "; INK 6; "
65 PRINT TAB 20; FLASH 1; "Puls
e cualquier tecla
70 PAUSE 0: CLS: PRINT "Cuida
do con los agujeros negros. Si c
ae pierde 5 puntos."
80 PRINT "Si derriba al Spect
rum obtiene 5 puntos."
90 INPUT "Presione ENTER si se
atreve a jugar conmigo"; LINE U
100 PRINT AT 0,0: FOR r=1 TO 2
110 PRINT "..... NEXT r
120 LET a=0: LET b=0
130 LET c=12: LET d=22
140 LET p=INT (RAND*20+1): LET q
=INT (RAND*30+1): IF p=0 OR p=12
150 GO TO 140
160 PRINT AT p,q: PAPER 0; "
170 POKE 23560,112
180 LET c0=SGN (RAND-.5)
190 LET d0=SGN (RAND-.5)
200 LET c1=c0*(RAND(.5))
210 LET d1=d0*(RAND(.5))
220 LET l=PEEK 23560
230 IF l=105 THEN LET b=b-1+32*
240 IF l=112 THEN LET b=b+1-32*
250 IF l=122 THEN LET a=a+1-22*
260 IF l=113 THEN LET a=a-1+22*
270

```

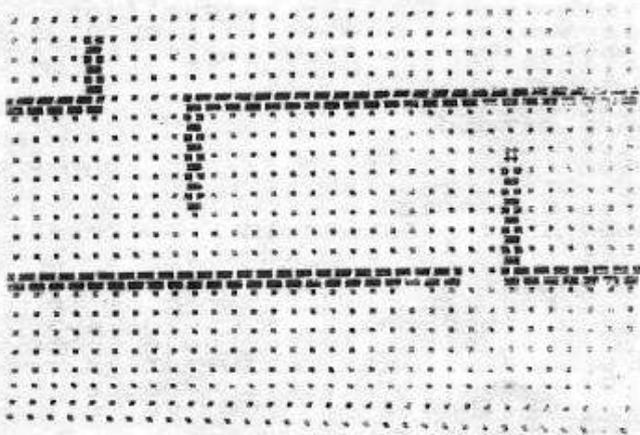
EL MEJOR ENLADRILLADOR, UD.

MUSICA

```

110 IF SCREEN$ (a,b) <> "." THEN
120 TO 5100
130 PRINT AT a,b; INK 4; "#"; AT
140 INK 6; "."
150 LET dd=d; LET cc=c
160 IF RND>.9 THEN GO TO 4000
170 LET c=c+c1; LET d=d+d1
180 GO SUB 8000
190 IF SCREEN$ (c,d) <> "." THEN
200 TO 4000
210 GO TO 2000
220 LET c2=c1; LET d2=d1
230 LET c1=c3*(c1=0); LET d1=d3
240 LET c=cc+c1; LET d=dd+d1
250 LET c=c+22*((c<0)-(c>21))
260 LET d=d+32*((d<0)-(d>31))
270 IF SCREEN$ (c,d) = "." THEN G
280 TO 2000
290 LET c=cc-c1; LET d=dd-d1
300 LET c=c+22*((c<0)-(c>21))
310 LET d=d+32*((d<0)-(d>31))
320 IF SCREEN$ (c,d) = "." THEN L
330 LET c1=-c1; LET d1=-d1; LET c3=c1
340 LET d3=d1+d2; GO TO 2000
350 LET c1=c2; LET d1=d2
360 LET c=cc+c1; LET d=dd+d1
370 GO SUB 8000
380 IF SCREEN$ (c,d) = "." THEN G
390 TO 2000
400 GO TO 7000
410 IF ATTR (a,b)=7 THEN BEEP ;
420 BEEP .4, -20; LET s=s+4; G
430 TO 5120
440 PRINT AT a,b; FLASH 1; "#"
450 LET s=s+1
460 FOR r=20 TO -20 STEP -1: BE
470 .01, r; NEXT r
480 PRINT AT 0,0; INK 0; PAPER
490 Cd, "o; " Spectrum "s; "
500 GO TO 90
510 LET o=o+1
520 IF (p=cc OR q=dd) AND (ABS
530 (d-q)=1 OR ABS (cc-p)=1) THEN L
540 LET o=o+4; BEEP .3,0; BEEP .4,8;
550 TO 7040
560 PRINT AT c,d; FLASH 1; "*"
570 FOR r=-20 TO 20: BEEP .01, r
580 NEXT r
590 GO TO 5000
600 LET c=c+22*((c<0)-(c>21))
610 LET d=d+32*((d<0)-(d>31))
620 RETURN

```



¿Le habían dicho alguna vez que con un ZX-81 se puede aprender música? Usted dirá que vaya tontería, teniendo en cuenta que este microordenador no tiene altavoz ni generador musical interno que envíe notas al altavoz del televisor. Hasta aquí su razonamiento sería más que correcto. Pero, ¿ha pensado alguna vez que Beethoven era sordo? Eso no fue

obstáculo para que su producción musical fuera tan excelente como si realmente oyera. El truco era fácil, se conocía las notas de memoria, tal como su ZX-81.

Bromas aparte, en la pantalla aparecen en posición aleatoria un pentagrama y una nota, de forma que tendrá que adivinar de qué nota se trata.

```

1 LET A=7
2 PRINT
3 PRINT
4 FOR B=1 TO 5
5 FOR N=0 TO 10
6 PRINT "-";
7 NEXT N
8 PRINT
9 PRINT
10 NEXT B
11 PRINT AT 0,0; "###"
12 PRINT " # #"
13 PRINT "-#-#"
14 PRINT " # #"
15 PRINT "-#-#"
16 PRINT " # #"
17 PRINT "-#-#"
18 PRINT "###"
19 PRINT "-#-#"
20 PRINT "###"
21 PRINT "-#"
22 PRINT " #"
23 PRINT "###"
24 LET B=INT (RND*11)+1
25 PRINT AT B,A; "#"
26 IF B>=8 THEN LET B=B-8
27 LET B=(11-B)+34
28 PRINT AT 15,10; "NOTA?"
29 INPUT A$
30 IF CODE A$<>B THEN PRINT AT
31 15,10; " mal "
32 IF CODE A$=B THEN PRINT AT
33 15,10; " bien "
34 PAUSE 4E4
35 CLS
36 RUN

```



BOMBARDEO

Aquí tiene la oportunidad de realizar su obra buena del día. Boomer es un fanático que disfruta lanzando bombas. Su misión consiste no en esquivarlas (que sería bastante sencillo), sino en atraparlas antes de que alcancen su objetivo. No se preocupe, Vd. está perfectamente equipado. Para correr hacia la

izquierda pulse la tecla 5 y para ir hacia la derecha pulse el 8.

Notas gráficas:
 170 Graphic A.
 180 Graphic S.
 190 Graphic D.
 200 Graphic G.
 221 Graphic F.

(16 K - Spectrum)

```

30 POKE USR "a",BIN 00011000
31 POKE USR "a"+1,BIN 001000100
32 POKE USR "a"+2,BIN 010000010
33 POKE USR "a"+3,BIN 101000101
34 POKE USR "a"+4,BIN 101000101
35 POKE USR "a"+5,BIN 100000001
36 POKE USR "a"+6,BIN 10111101
37 POKE USR "a"+7,BIN 010000010
40 POKE USR "s",BIN 00111100
41 POKE USR "s"+1,BIN 00111100
42 POKE USR "s"+2,BIN 01111110
43 POKE USR "s"+3,BIN 11111111
44 POKE USR "s"+4,BIN 11011011
45 POKE USR "s"+5,BIN 11011011
46 POKE USR "s"+6,BIN 11011011
47 POKE USR "s"+7,BIN 00011000
50 POKE USR "d",BIN 00011000
51 POKE USR "d"+1,BIN 00011000
52 POKE USR "d"+2,BIN 01111110
53 POKE USR "d"+3,BIN 01100110
54 POKE USR "d"+4,BIN 01100110
55 POKE USR "d"+5,BIN 0110010
56 POKE USR "d"+6,BIN 11100111
57 POKE USR "d"+7,BIN 11100111
60 POKE USR "f",BIN 00000000
61 POKE USR "f"+1,BIN 00001000
62 POKE USR "f"+2,BIN 00010000
63 POKE USR "f"+3,BIN 00011000
64 POKE USR "f"+4,BIN 00111100
65 POKE USR "f"+5,BIN 01111110
66 POKE USR "f"+6,BIN 01111110
67 POKE USR "f"+7,BIN 01111100
70 POKE USR "g",BIN 11011011
71 POKE USR "g"+1,BIN 10111101
72 POKE USR "g"+2,BIN 10011001
73 POKE USR "g"+3,BIN 11111111
74 POKE USR "g"+4,BIN 00011000
75 POKE USR "g"+5,BIN 00111100
76 POKE USR "g"+6,BIN 00100100
  
```

```

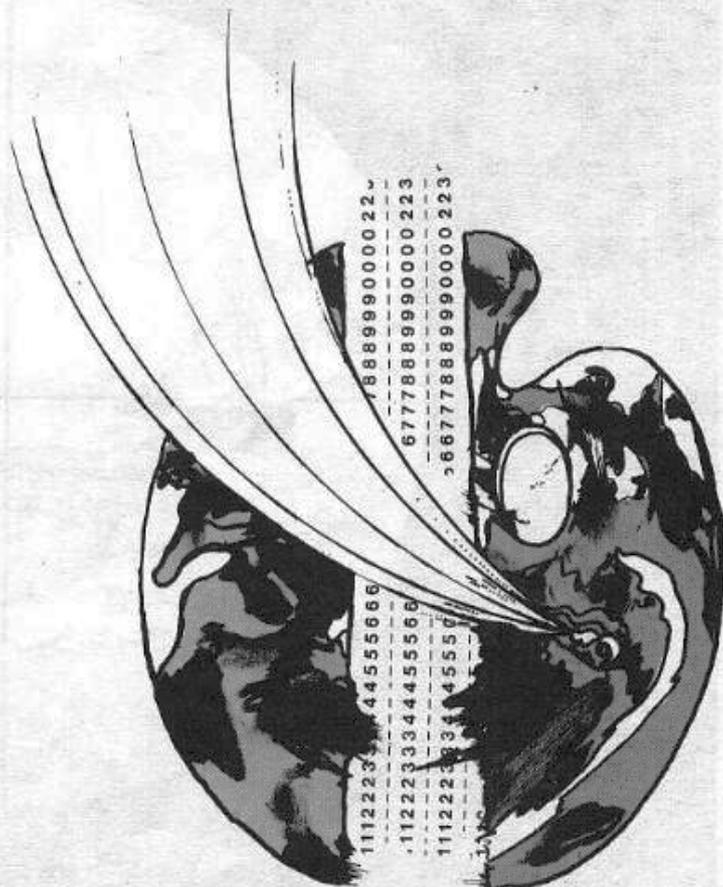
77 POKE USR "g"+7,BIN 0110010
80 BORDER 2: PAPER 6: INK 0: C
L5: PRINT AT 10,7;"Instruccion
s(s/n)": PAUSE 0: IF INKEY$="s"
THEN GO SUB 9000
96 LET high=0
101 BRIGHT 1: FLASH 0: INK 0: P
APER 6: BORDER 1: CLS
102 LET bombs=0: LET lives=4
103 LET y=0
110 LET cx=15: LET cy=20
120 LET lives=lives+2
135 LET p=0
140 LET by=2
150 LET bx=INT (RND*25)+1
155 IF p=0 THEN LET by=y+3
160 FOR x=1 TO 30
167 PRINT AT 21,11;"Vidas=";liv
es
169 IF lives<=0 THEN GO TO 9999
170 PRINT INK 0;AT y,x;" "
180 PRINT INK 2;AT y+1,x;" "
190 PRINT INK 1;AT y+2,x;" "
200 PRINT AT cy,cx-1;" "
205 IF INKEY$="5" THEN LET cx=c
x-1
206 IF INKEY$="8" THEN LET cx=c
x+1
207 IF cx>30 THEN PRINT AT 20,3
0;" " : LET cx=1
208 IF cx<1 THEN PRINT AT cy,cx
-1;" " : LET cx=30
210 IF x=bx THEN LET p=1
213 IF INT (by)<>20 THEN GO TO
216
214 IF cx=bx THEN BEEP .02,20:
BEEP .05,30: LET bombs=bombs+10:
GO TO 215
215 IF NOT cx=bx THEN PRINT AT
by,bx; INK 2;" " : BEEP .5,-40: P
RINT AT by,bx;" " : LET lives=li
ves-1
216 IF by>19 THEN LET p=0: CLS
: GO TO 300
221 PRINT AT by,bx;" " : IF p=1
THEN LET by=by+1: PRINT INK 2;AT
by,bx;" " : BEEP .009,-by+10
230 NEXT x
232 LET by=by+1: LET y=y+1: IF
y>15 THEN GO TO 9100
240 CLS
250 GO TO 160
300 IF p=0 THEN LET by=y+2: LET
p=0: LET bx=INT (RND*30)+1
310 IF x<30 THEN NEXT x
320 GO TO 160
9000 BORDER 5: INK 0: PAPER 7: C
L5
9010 PRINT " " "Hola, me llamo Boo
mer!"
9020 PRINT INK 0;" " " Este"
9030 PRINT INK 2;" " " soy"
9040 PRINT INK 1;" " " yo"
9050 PRINT " " "Algunos creen que
estoy loco, solo porque lanzo
mis bombas sobre los tontos.Y
ahora las voy a lanzar sobre
ti. !PREPARETE!"; FLAS
H 1: FLASH 0
9055 PAUSE 0: CLS
9060 RETURN
9100 CLS
9200 FOR n=1 TO 20 STEP 2: PAUSE
2: BEEP .05,n: NEXT n
9300 GO TO 103
9998 IF bombs>high THEN LET high
=bombs
9999 FLASH 1: PAPER 2: INK 7: CL
S: PRINT AT 10,4;"Boomer te pil
lo": FOR n=0 TO -255 STEP -1: PA
USE 2: BEEP .02,n: BEEP .01,n+5:
NEXT n: PAUSE 10: FLASH 0: PAPE
R 6: INK 0: CLS: PRINT AT 10,10
;"Probamos otra vez": PAUSE 0: C
LS: IF INKEY$="s" THEN GO TO 10
0: STOP
  
```

PROGRAMAS

```

100 LET b#="7"
105 LET HS#0
200 GO SUB 600
300 LET k=10
400 LET b=1
1100 FOR f=1 TO 10
1110 LET b=b+1: DIM z(b)
1200 FOR g=1 TO b
1400 LET z(g)=INT (RND*9)
1500 IF z(g)>7 THEN GO TO 140
1600 NEXT g
170 FOR h=1 TO b
200 PAPER z(h): BORDER z(h): CLS
220 BEEP 1,z(h)
230 NEXT h
300 FOR i=1 TO b
310 LET a#=INKEY$
320 IF a#="" THEN GO TO 310
330 IF a#<"0" OR a#>"7" THEN GO
TO 310
340 LET l=(0 AND a#="0")+ (1 AND
a#="1")+ (2 AND a#="2")+ (3 AND a
#="3")+ (4 AND a#="4")+ (5 AND a#="
5")+ (6 AND a#="6")+ (7 AND a#="7
")
345 IF l<z(i) THEN GO TO 500
350 PAPER l: BORDER l: CLS
360 NEXT i
390 NEXT f
400 PAPER 1: BORDER 1: INK 7: C
LS
410 PRINT AT 5,2;"su puntuacion
es ";k
420 IF k=10 THEN PRINT AT 7,7;
FLASH 1;"FELICIDADES"
425 IF k>HS THEN INPUT "Alta pu
ntuacion. Introduzca su nombre";
hs: LET HS=k
440 PRINT AT 12,4;"puntuacion m
as alta";HS;" ";h$
470 INPUT "Continua";y$
480 IF y#="y" THEN GO TO 20
490 PRINT AT 15,10;"D.K. Usted
anda ";STOP
500 PAPER 1: INK 7: CLS : FOR n
TO 10: PRINT FLASH 1, BRIGHT
"ERROR"; NEXT n
510 PLOT 125,85: DRAW 70,40: PL
OT 125,85: DRAW -70,-40: PLOT 12
5,85: DRAW -70,-40: PLOT 125,85:
DRAW 70,-40
510 LET k=k-1
520 BEEP 3,-20: BEEP 3,-35
530 PRINT PAPER 1: INK 7: AT 20,
7;"Para continuar pulse cualquier
tecla"
540 PAUSE 0
550 LET b=1
560 GO TO 390
580 PAPER 1: BORDER 1: INK 7: C
LS
610 PRINT AT 0,10;"SECUENCIA";A
T 0,10: OVER 1;"
620 PRINT "El objetivo de este
juego es seguir mi secuencia de
colores y sonidos, pulsando la
s teclas de 0 a 7, correspondien
te a mi se-
cuencia de colores."
650 PRINT "Por ejemplo, si los
colores son verde, rojo...se puls
a 4, 2... y asi sucesivamente"
670 PRINT AT 20,7;"Pulse cualqu
ier tecla para comenzar"
680 PAUSE 0: CLS : RETURN

```

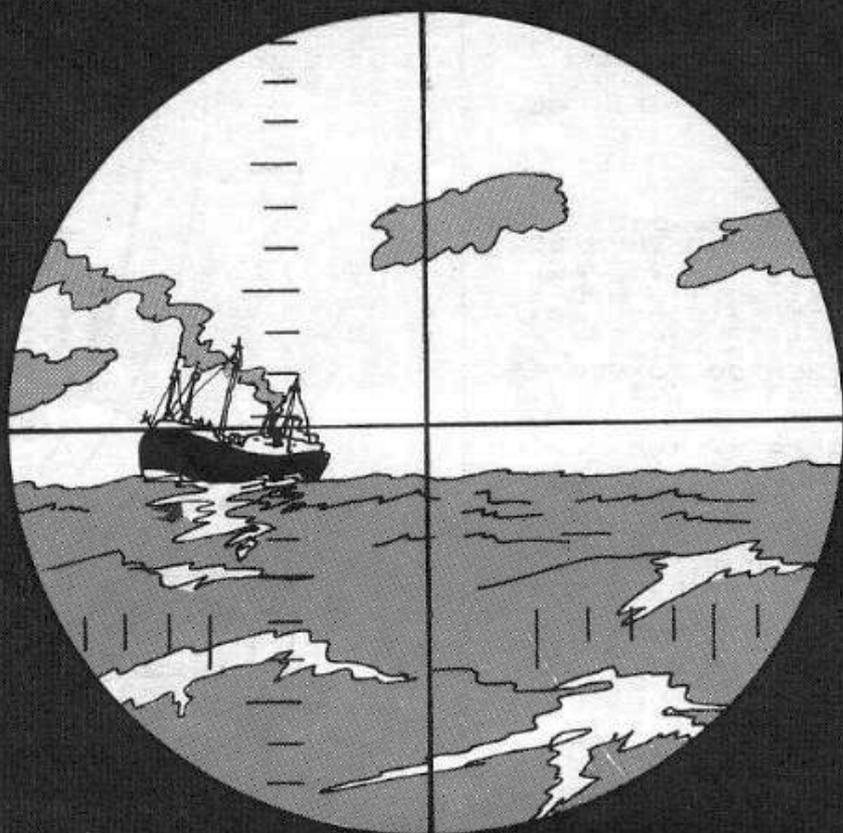


Este es un sencillo juego que podrá introducir en su Spectrum, siempre que lo tenga conectado a un televisor de color, de otra manera le podrá resultar bastante difícil adivinar el color ya que de eso se trata. En la pantalla de su televisor irán apareciendo una serie de colores, acompañados de "beeps", para rápidamente desaparecer y darle la oportunidad de recordar dichos colores. Es

una buena oportunidad para que practique con las distintas opciones de color de su Spectrum (pruebe a cambiar las instrucciones de PAPER, BORDER, INK... etc.), al mismo tiempo que puede competir con sus amigos para ver quien alcanza mayor puntuación. El número de "tiradas" es de 10 y para variarlo sólo ha de modificar la sentencia 30.

(16 K - Spectrum)

¿RECUERDA UD. LOS COLORES?



GUERRA SUBMARINA

"Era el día D y se acercaba la hora H. Nuestros hombres estaban en sus puestos y los torpedos listos para recibir a la flota japonesa, que en breves momentos pasaría delante del periscopio...". Echele un poco de imaginación y disfrute con este ma-

ravilloso juego. Los gráficos son francamente buenos como puede ver en el listado. El submarino es fácil de maniobrar: se presiona 5 para ir a la izquierda y 8 para la derecha. Los torpedos los lanza con la tecla espaciadora.

(16 K - Spectrum)

Notas gráficas:

70 - Graphic A, B.

190 - Graphic K, L.

210 - Graphic C, D, E.

230 - Graphic P, Q, R.

250 - Graphic M, N, O.

270 - Graphic F, G, H, I, J.

1030 - Graphic U.

2510 - Graphic S.

2520 - Graphic T.

PROGRAMAS

```

10 GO SUB 3000. 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
20 REM submarino
50 PAUSE 50
70 PRINT AT 10,11;"submarino";
AT 12,14;" "
100 PAUSE 50
110 REM *****
120 REM Instrucciones
130 CLS
140 PRINT TAB 11;"SUBMARINO"
150 PRINT TAB 11;" "
160 PRINT "El juego consiste en
destruir el barco enemigo evi
tando las cargas de profundidad
d de los barcos torpederos
lo que significa": INK 2:
FLASH 1: PRINT "MUERTE": INK 2: FLA
SH 0: INK 1
165 PRINT "(5) Izquierda - (8) De
screa "
170 PRINT "(Space) Fuego": INK 1
190 PRINT " " "1000 puntos"
200 PRINT " " "500 puntos"
220 PRINT " " "250 puntos"
240 PRINT " " "50 puntos"
260 PRINT " " "50 puntos"
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$,val
540 NEXT a
545 LET step=.25+((INT (RND*5) +
1)/6)
550 RESTORE
560 INK 5: PRINT AT 9,0;" "
580 PRINT AT 0,0;"Tantos:";score
e: PRINT AT 0,15;"Torpedos:";tor
p: IF torp<10 THEN PRINT AT 0,25
700 INK 5: PRINT AT 9,shx+5;" "
705 INK 1: PRINT AT 20,sux;" "
710 LET sux=sux+(INKEY$=" " AND
sux<30)-(INKEY$="5" AND sux>0)
715 PRINT OVER 1;AT 20,sux;" "
720 PRINT OVER 0;AT 9,shx;a$
725 IF INKEY$=" " AND state=0 T
HEN LET state=1: LET torp=torp-1
726 IF state=1 THEN GO SUB 1000
727 IF state=0 THEN GO SUB 900
728 IF sdc=1 THEN GO TO 3000
729 IF sdc=1 THEN GO SUB 2500
730 IF sdc=0 AND val=1000 AND R
ND<.4 THEN GO SUB 2000
740 LET shx=shx-step
745 IF (shx<1 AND torp<1) THEN
GO TO 3000
750 IF shx<1 THEN GO TO 500
755 GO TO 500
900 LET pos t=sux
905 LET tpy=10
910 RETURN
1000 INK 1: PRINT AT tpy+1,pos t
;" "
1030 PRINT OVER 1;AT tpy,post;" "
1034 IF tpy>10 THEN GO TO 1040
1035 IF tpy=10 THEN LET state=0
1036 IF tpy=10 THEN PRINT AT tpy
,post;" "
1037 IF tpy=10 AND SCREEN$ (tpy-

```

```

1,post)<)" " THEN LET score=score
+val: PRINT OVER 1;AT tpy-1,pos
t;" " : PAUSE 50: FOR y=1 TO 8: B
EEP .05,10*RND+20: NEXT y: PRINT
AT tpy-1,0;" " : GO TO 500
1040 LET tpy=tpy-.5: BEEP .08,-.1
0: RETURN
2000 REM ***** Soltar
cargas profundidad
2010 LET pdc=shx
2020 LET ddc=10
2030 LET sdc=10
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;" " : L
ET sdc=0
2530 IF ddc=20 AND ABS (pdc-sux)
<5 THEN PRINT AT 20,sux;" " : B
EEP .25,-30: PAUSE 25: PRINT AT 2
0,sux;" " : GO TO 4000
2540 LET ddc=ddc+1
2550 RETURN
3000 CLS
3100 PRINT "Fin. su puntuacion e
s " : score
3200 GO TO 5000
4000 INK 2: CLS : PRINT "BOOOMM:
"
4010 INK 0: PRINT "supuntuacion
era " : score
5000 PRINT "otra vez (s/n)"
5010 INPUT a$
5020 IF a$="s" OR a$="S" THEN RU
N
5030 IF a$="n" OR a$="N" THEN ST
OP
5040 GO TO 5010
6000 RESTORE 9000
6010 FOR a=144 TO 164
6020 FOR b=0 TO 7
6030 READ c
6040 POKE (USR CHR$ a)+b,c
6050 NEXT b
6060 NEXT a
6070 RESTORE
6080 RETURN
9000 DATA 1,3,3,127,255,255,127,
0
9010 DATA 0,192,192,248,255,255,
248,0
9020 DATA 0,0,3,3,255,127,63,0
9030 DATA 0,60,60,250,255,255,55
,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
4,252
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

```



ESCARABAJO

Conviértase por un momento en un escarabajo e introdúzcase en la pantalla de su televisor. El juego consiste en escapar en cualquier dirección antes de que su Spectrum le cerque, lo cual implica su muerte, al igual que si intenta pasar por uno de los obstáculos que le irán apareciendo. El movimiento se efectúa a través de las teclas del cursor.

El programa es francamente bueno y la representación gráfica es formidable. Conseguir una alta puntuación está reservado sólo a los muy habilidosos.

Notas gráficas:
160, 3000 - Graphic A.
105, 106, 108, 100 - Graphic B.

(16 K - Spectrum)

```

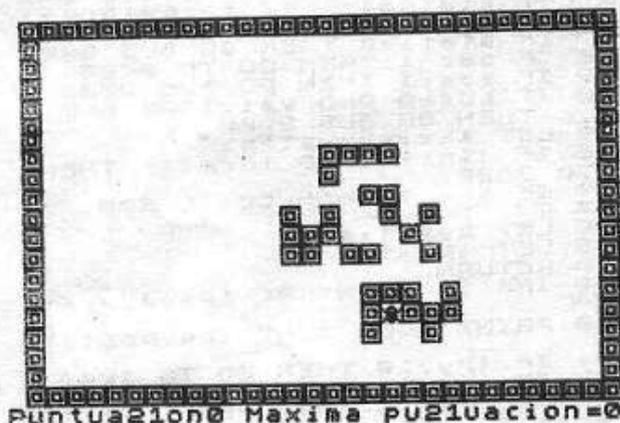
4 DATA 0,24,165,126,60,60,125
,150
5 FOR t=0 TO 7
6 READ X
7 POKE USR "a"+t,x
8 NEXT t
10 DATA 255,129,189,165,165,16
9,129,255
200 FOR t=0 TO 7
30 READ X
40 POKE USR "b"+t,x
50 NEXT t
55 BORDER 0
60 LET hs=0
65 CLS
70 LET x=10
75 LET y=15
80 LET s=0: PRINT AT 21,0;"PUN
tu,":s
90 PRINT AT 21,12;"Maxi,=";hs
100 PRINT INK 0; PAPER 2;AT 1,1
...
125 PRINT INK 0; PAPER 2;AT 20,
...
106 FOR f=1 TO 20: PRINT INK 0;
PAPER 2;AT f,1;"0": NEXT f
120 FOR f=1 TO 20: PRINT INK 0;
PAPER 2;AT f,30;"0": NEXT f
150 PRINT INK 0;AT x,y;"*": FOR
f=1 TO 5: BEEP .005,f: NEXT f
165 LET s=s+1: PRINT AT 21,6;s

```

```

167 IF hs<s THEN LET hs=s: PRIN
T AT 21,21;hs
170 LET a=y+INT (RND*3)+2
180 LET b=x+INT (RND*3)+2
185 IF b=x AND a=y THEN GO TO 1
70
190 PRINT INK 0; PAPER 2;AT b,a
...
191 IF SCREEN$ (x-1,y)<>" " THE
N GO TO 193
192 GO TO 200
193 IF SCREEN$ (x+1,y)<>" " THE
N GO TO 196
194 GO TO 200
195 IF SCREEN$ (x,y+1)<>" " THE
N GO TO 198
196 GO TO 200
197 IF SCREEN$ (x,y-1)<>" " THE
N GO TO 200
200 LET d=1
210 IF INKEY$="5" AND y>2 THEN
GO TO 1000
220 IF INKEY$="6" AND y<29 THEN
GO TO 1100
230 IF INKEY$="7" AND x>2 THEN
GO TO 1200
240 IF INKEY$="8" AND x<19 THEN
GO TO 1300
250 LET d=d+1: IF d=10 THEN GO
TO 170
260 GO TO 210
1000 PRINT AT x,y;" ": LET y=y-1
: IF SCREEN$ (x,y)<>" " THEN GO
TO 2000
1001 GO TO 160
1100 PRINT AT x,y;" ": LET y=y+1
: IF SCREEN$ (x,y)<>" " THEN GO
TO 2000
1101 GO TO 160
1200 PRINT AT x,y;" ": LET x=x-1
: IF SCREEN$ (x,y)<>" " THEN GO
TO 2000
1201 GO TO 160
1300 PRINT AT x,y;" ": LET x=x+1
: IF SCREEN$ (x,y)<>" " THEN GO
TO 2000
1301 GO TO 160
3000 PRINT ; FLASH 1;AT x,y;"*":
FOR f=1 TO 100: NEXT f
3001 BEEP 1,-12: BEEP .6,-12: BE
EP .3,-12: BEEP .9,-12: BEEP .6,
-9: BEEP .3,-10: BEEP .57,-10: B
EEP .3,-12: BEEP .6,-12
3002 BEEP .4,-10: BEEP 1,-12
3004 PRINT INK 0; PAPER 2;AT x,y
...
3006 IF s>hs THEN LET hs=s
3010 PRINT FLASH 1;AT 0,10;"Otra
...
3015 IF INKEY$<>" " THEN GO TO 60
3020 GO TO 3015

```



PROGRAMAS

En este juego, como su nombre indica, tendrá que escapar de los obstáculos que le pone su ordenador. Pero, a diferencia de otros, en éste se requiere una gran habilidad para no ser "atrapado" rápidamente. Es un excelente

programa con pocas sentencias y una buena presencia en pantalla.

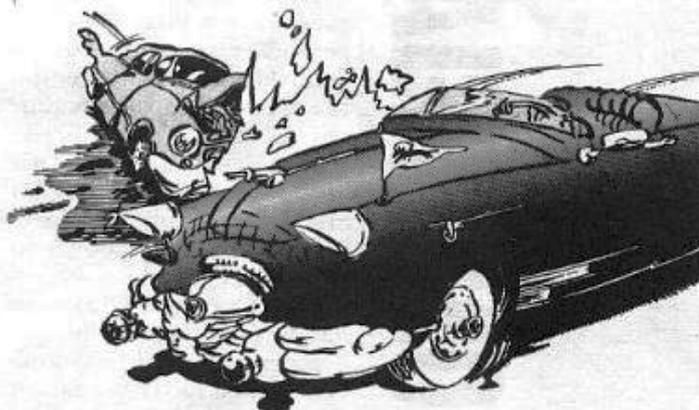
Notas gráficas:
2020 - Graphic P.

(16 K - Espectrum)

```

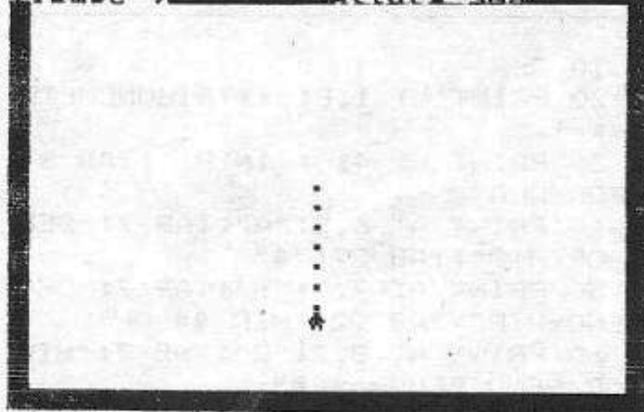
5 LET hs=355: GO SUB 2000
10 INK 0: PAPER 6: BORDER 1: C
LS: INPUT "Grado de dificultad
(1-6)";d: IF d<1 OR d>6 OR d>INT
d THEN BEEP 1,0: GO TO 10
20 LET t=0: LET l=5: LET q=d+5
LET d=d/100: FOR f=0 TO 31: PR
INT AT 0,f: INK 3;"■";AT 21,f;"■
": IF f<22 THEN PRINT INK 3;AT f
,0;"■";AT f,31;"■"
30 NEXT f: FOR f=1 TO 45: PRIN
T PAPER RND*3+1;AT RND*19+1,RND*
29+1;"": NEXT f: PRINT PAPER 3;
INK 7;AT 0,16;"Altura ";hs
40 BEEP 1,0: LET a=1: LET b=0:
LET x=15: LET y=10: FOR f=9 TO
13: PRINT AT f,14;"": NEXT f
50 LET i$=INKEY$: IF i$>"4" AN
D i$<"9" THEN LET a=(i$="6")-(i$
="7"): LET b=(i$="8")-(i$="5"):
BEEP .005,x
60 PRINT AT 0,0: INK 7: PAPER
3: t=empo
70 FOR f=1 TO q: NEXT f
90 BEEP d,x: LET t=t+1: LET y=
y+a: LET x=x+b: LET c=ATTR(y,x)
PRINT AT y,x;"#";AT y-a,x-b;"
": IF c=48 THEN GO TO 50
100 PRINT AT 21,16; FLASH 1; IN
K 2: PAPER 6;"Ja,Ja!!": FOR y=1
TO 20: PRINT AT y,1: PAPER 0; IN
K 0;TAB 31: NEXT y: FOR y=1 TO
15: PRINT AT RND*19+1,RND*29+1;
PAPER RND*5;"": NEXT y
110 LET l=l-1: PRINT PAPER 3;AT
y,x;"":AT 21,0: PAPER 3; INK 7
;"Vidas";l: IF l THEN GO TO 1000
120 PRINT #0; FLASH 1; INK 2; P
APER 6;"Se quedo sin vida": IF t
>hs THEN PRINT AT 21,0; FLASH 1;
PAPER 3; INK 7;"Alta puntuacio
n": LET hs=t
130 IF INKEY$<>" " THEN GO TO 13
0
140 IF INKEY$="" THEN GO TO 140
150 GO TO 10
1010 PRINT AT 21,16; PAPER 3;"
": GO TO 40
2000 RESTORE: FOR f=USR "p" TO
USR "p"+7: READ a: POKE f,VAL ("
BIN "+STR$ a): NEXT f: DATA 1100
0,a,111100,1011010,1011000,10010
0,a,0
2010 BORDER 0: PAPER 0: INK 6: C
LS
2020 PRINT TAB 9;"Hola" "Ud. (#)
debe intentar salir de este la
berinto sin chocar con los obs
táculos. Su movimiento se real
iza con las teclas de movimie
nto de cursor."
2025 PRINT "Pulse cualquier tec
la para comenzar"
2030 IF INKEY$<>" " THEN GO TO 20
30
2040 IF INKEY$="" THEN GO TO 204
0
2050 RETURN

```



ESCAPA COMO PUEDES

Tiempo 7 Altura 355



TRIGONOMETRIA FACIL

Probablemente la primera vez que usted escuchó la palabra trigonometría pensó que se trataba de un trabalenguas. Si no fue así, enhorabuena. De cualquier forma, lo que sí es seguro es que todavía no había posibilidades de comprarse un ZX para ayudarle a deshacer el enredo.

Puede usted preguntarse que por qué le hablamos a estas alturas de aquel tedioso capítulo de su libro de matemáticas. No es cierto que queramos que vuelva a tener aquellas pesadillas. Pero, ¿ha pensado en lo que ocurrirá dentro de no mucho tiempo cuando sus niños le pregunten sobre el tema? Elija lo que prefiere, si hacer el ridículo o echar mano de su Sinclair y seguir siendo el héroe que todo lo sabe y todo lo puede. Piénselo.

Por si a estas alturas ya se ha decidido, recuerde que una vez pasado el programa va a ser más fácil recordar los conceptos de seno, coseno y tangente, así como sus relaciones inversas.

El listado que presentamos ha sido realizado en una impresora diferente a la ZX Print, lo que significa que quienes utilicen el dispositivo propio de Sinclair deberán sustituir# por Q Gráfico en las líneas 40 a 130. En la línea 130 deberán escribir STOP en inverso.

```
10 CLS
20 PRINT AT 1,8: "**TRIGONOMETRIA**"
30 PRINT AT 4,0: "INTR. "; TAB 8; "FORMULA"
40 PRINT AT 6,2: "A"; TAB 7; "SEN O=OP/HIP"; TAB 27; "$"
50 PRINT AT 7,2: "B"; TAB 7; "OP= SEND*HIP"; TAB 22; "HIP $$ OP"
60 PRINT AT 8,2: "C"; TAB 7; "HIP =OP/SEND SEND->###"
```

```
70 PRINT AT 10,2: "D"; TAB 7; "COS=AY/HIP"; TAB 27; "$"
80 PRINT AT 11,2: "E"; TAB 7; "AY =COS*HIP"; TAB 22; "HIP $$"
90 PRINT AT 12,2: "F"; TAB 7; "HIP=ADY/COS"; TAB 20; "COS->###"
100 PRINT TAB 25; "ADY"
110 PRINT AT 15,2: "G"; TAB 7; "TAN=OP/ADY"; TAB 27; "$"
120 PRINT AT 16,2: "H"; TAB 7; "OP =TAN*ADY"; TAB 26; "##"
130 PRINT AT 17,2: "I"; TAB 7; "ADY=OP/TAN"; TAB 20; "TAN->###"
140 PRINT TAB 25; "ADY"; AT 19,2; "J"; TAB 7; "stop"
150 PRINT AT 21,0: "QUE FORMULA NECESITAS (A A J)"
800 INPUT B$
810 IF B$="A" THEN GOTO 1100
820 IF B$="B" THEN GOTO 1300
830 IF B$="C" THEN GOTO 1500
840 IF B$="D" THEN GOTO 1700
850 IF B$="E" THEN GOTO 1900
860 IF B$="F" THEN GOTO 2100
870 IF B$="G" THEN GOTO 2300
880 IF B$="H" THEN GOTO 2500
890 IF B$="I" THEN GOTO 2700
900 IF B$="J" THEN STOP
910 IF B$<"A" OR B$>"J" THEN GOTO 800
1100 CLS
1110 PRINT AT 2,2: "OP=?"
1120 INPUT O
1130 PRINT AT 2,15: "OP = "; O
1140 PRINT AT 4,2: "HIP=?"
1150 INPUT H
1160 PRINT AT 4,15: "HIP= "; H
1170 LET S=O/H
1180 LET X=ASN S*180/PI
1190 PRINT AT 6,5: "ANGULO="; X; "GRADOS"
1200 PRINT "PULSA NL PARA CONTINUAR..."
1210 INPUT A$
1220 GOTO 10
1300 CLS
1310 PRINT AT 2,2: "SEN=?"
1320 INPUT S
1330 PRINT AT 2,15: "SEN="; S; " GRADOS"
1340 PRINT AT 4,2: "HIP=?"
1350 INPUT H
1360 PRINT AT 4,15: "HIP= "; H
```

PROGRAMAS

```
1370 LET O=(SIN (S*PI/180))*H
1380 PRINT AT 6,5;"OP= ";O
1390 PRINT AT 10,3;"PULSA NL PAR
A CONTINUAR..."
1400 INPUT A$
1410 GOTO 10
1500 CLS
1510 PRINT AT 2,2;"OP=?"
1520 INPUT O
1530 PRINT AT 2,15;"OP= ";O
1540 PRINT AT 4,2;"SENO=?"
1550 INPUT S
1560 PRINT AT 4,15;"SENO=";S;" G
RADOS"
1570 LET H=O/(SIN (S*PI/180))
1580 PRINT AT 6,5;"HIP= ";H
1590 PRINT AT 10,3;"PULSA NL PAR
A CONTINUAR..."
1600 INPUT A$
1610 GOTO 10
1700 CLS
1710 PRINT AT 2,2;"ADY=?"
1720 INPUT A
1730 PRINT AT 2,15;"ADY= ";A
1740 PRINT AT 4,2;"HIP=?"
1750 INPUT H
1760 PRINT AT 4,15;"HIP= ";H
1770 LET C=A/H
1780 LET X=ACS C*180/PI
1790 PRINT AT 6,5;"ANGULO=";X;" G
RADOS"
1800 PRINT AT 10,3;"PULSA NL PAR
A CONTINUAR..."
1810 INPUT A$
1820 GOTO 10
1900 CLS
1910 PRINT AT 2,2;"COS=?"
1920 INPUT C
1930 PRINT AT 2,15;"COS= ";C;" G
RADOS"
1940 PRINT AT 4,2;"HIP=?"
1950 INPUT H
1960 PRINT AT 4,15;"HIP= ";H
1970 LET A=(COS (C*PI/180))*H
1980 PRINT AT 6,5;"AY= ";A
1990 PRINT AT 10,3;"PULSA NL PAR
A CONTINUAR..."
2000 INPUT A$
2010 GOTO 10
2100 CLS
2110 PRINT AT 2,2;"ADY=?"
2120 INPUT A
2130 PRINT AT 2,1;"ADY= ";A
```

```
2140 PRINT AT 4,2;"COS=?"
2150 INPUT C
2160 PRINT AT 4,15;"COS= ";C;" G
RADOS"
2170 LET H=A/(COS (C*PI/180))
2180 PRINT AT 6,5;"HIP= ";H
2190 PRINT AT 10,3;"PULSA NL PAR
A CONTINUAR..."
2200 INPUT A$
2210 GOTO 10
2300 CLS
2310 PRINT AT 2,2;"OP=?"
2320 INPUT O
2330 PRINT AT 2,15;"OP= ";O
2340 PRINT AT 4,2;"ADY=?"
2350 INPUT A
2360 PRINT AT 4,15;"ADY= ";A
2370 LET T=O/A
2380 LET X=ATN T*180/PI
2390 PRINT AT 6,5;"ANGULO=";X;"
GRADOS"
2400 PRINT AT 10,3;"PULSA NL PAR
A CONTINUAR..."
2410 INPUT A$
2420 GOTO 10
2500 CLS
2510 PRINT AT 2,2;"TAN=?"
2520 INPUT T
2530 PRINT AT 2,15;"TAN= ";T;" G
RADOS"
2540 PRINT AT 4,2;"ADY=?"
2550 INPUT A
2560 PRINT AT 4,15;"ADY= ";A
2570 LET O=(TAN (T*PI/180))*A
2580 PRINT AT 6,5;"OP= ";O
2590 PRINT AT 10,3;"PULSA NL PAR
A CONTINUAR..."
2600 INPUT A$
2610 GOTO 10
2700 CLS
2710 PRINT AT 2,2;"OP=?"
2720 INPUT O
2730 PRINT AT 2,15;"OP= ";O
2740 PRINT AT 4,2;"TAN=?"
2745 INPUT T
2750 PRINT AT 4,15;"TAN= ";T;" G
RADOS"
2770 LET A=O/(TAN (T*PI/180))
2780 PRINT AT 6,5;"ADY= ";A
2790 PRINT AT 10,3;"PULSA NL PAR
A CONTINUAR..."
2800 INPUT A$
2810 GOTO 10
```

UN CASO DE PROBABILIDAD

Los aficionados a los juegos y la estadística encontrarán en este programa un valioso aliado. Con él podremos conocer las oportunidades teóricas que puede tener la yegua Manuela de entrar la primera en la meta, seguida a medio cuerpo por su eterno rival. También podría calcularse cuál va a ser la probabilidad teórica de que salga una determinada carta de un mazo de ellas o que el número en que caiga la bolita de la ruleta sea éste o aquél. O si los dados tienen más preferencias por caer en una posición o en otra. Quién sabe, a lo mejor sirve para darnos cuenta de que Alfonso siempre nos gana gracias a sus dados trucados.

Para calcular todo esto, y mucho más, se recurre a la conocida fórmula de Poisson, que proporciona la probabilidad de que ocurra un

determinado evento basándose en el número de veces que ha ocurrido anteriormente.

Una vez que conocemos el porcentaje de las posibilidades de que ocurra lo que queremos o no queremos, pasamos a estar en disposición de evaluar mejor nuestra decisión.

Así no será de extrañar que en nuestra próxima partida de cartas aparezcan el resto de compañeros de juegos con una misteriosa caja, que pondrán celosamente su lado en la mesa, y que una vez abierta será el Sinclair programado como aquí se indica. Por si acaso, familiarícese con este programa. Se lo hemos advertido. Después no se queje. El que avisa no es traidor. El programa está escrito para un ZX81 con 1 Kbyte.

```

9 CLEAR
10 PRINT "ESTUDIO DE PROB. A
LARGO PLAZO"
11 LET S=0
12 LET T=0
13 LET U=0
14 LET V=0
15 LET W=0
16 LET X=0
17 LET Y=0
18 LET Z=0
20 LET AA=0
22 LET BB=0
25 PRINT
30 PRINT "INTRODUCE EL NUMERO
MAXIMO DE SUCEOS EN CADA UNID
AD DE TIEMPO"

```

```

31 PRINT
32 PRINT "INTERVALO DE 3 A 12"
40 INPUT A
45 CLS
50~PRINT "INTRODUCE EL NUMERO
TOTAL DE CASOS"
60 INPUT B
70 PRINT
80 PRINT "INTRODUCE EL TIEMPO
TOTAL"
90 INPUT C
95 CLS
100 LET D=B/C
110 LET E=(1+(1/B))*B
120 LET F=1/(E**D)
130 LET P=F
140 LET Q=D*F
150 LET R=D**2/2*F
160 IF A>=3 THEN LET S=D**3/6*F
170 IF A>=4 THEN LET T=D**4/24*
F
180 IF A>=5 THEN LET U=D**5/120
*F
190 IF A>=6 THEN LET V=D**6/720
*F
200~IF A>=7 THEN LET W=D**7/504
0*F
210 IF A>=8 THEN LET X=D**8/403
20*F
220 IF A>=9 THEN LET Y=D**9/362
880*F
230 IF A>=10 THEN LET Z=D**10/3
628800*F
233 IF A>=11 THEN LET AA=D**11/
39916800*F
237 IF A=12 THEN LET BB=D**12/4
79001600*F
240 PRINT "PROB. DE", "PORCENTAJ
E"
250 PRINT
253 REM FACTOR DE CORRECCION
255 LET N=P+Q+R+S+T+U+V+W+X+Y+Z
+AA+BB
260 LET PP=(INT (P*10000/N+0.5)
)/100
270 LET QQ=(INT (Q*10000/N+0.5)
)/100
280 LET RR=(INT (R*10000/N+0.5)
)/100
290 LET SS=(INT (S*10000/N+0.5)
)/100

```

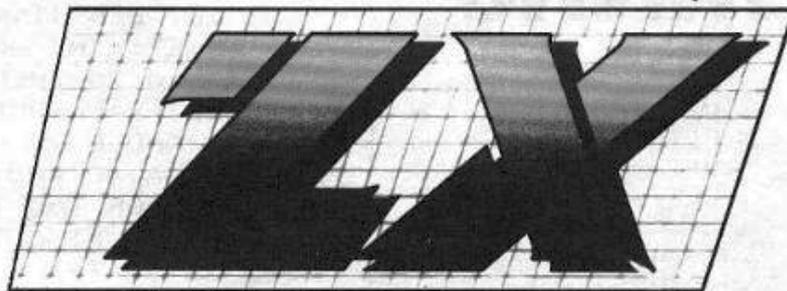
PROGRAMAS

```

300 LET TT=(INT (T*10000/N+0.5)
)/100
310 LET UU=(INT (U*10000/N+0.5)
)/100
320 LET VV=(INT (V*10000/N+0.5)
)/100
330 LET WW=(INT (W*10000/N+.5)
)/100
340 LET XX=(INT (X*10000/N+0.5)
)/100
350 LET YY=(INT (Y*10000/N+0.5)
)/100
360 LET ZZ=(INT (Z*10000/N+0.5)
)/100
363 LET AAA=(INT (AA*10000/N+0.
5))/100
366 LET BBB=(INT (BB*10000/N+0.
5))/100
    
```

```

370 LET A$="0 SUCESOS"
380 LET B$="1 ""
390 LET C$="2 ""
400 LET D$="3 ""
410 LET E$="4 ""
420 LET F$="5 ""
430 LET G$="6 ""
440 LET H$="7 ""
450 LET I$="8 ""
451 LET J$="9 ""
452 LET K$="10 ""
455 LET L$="11 ""
457 LET M$="12 ""
570 PRINT A$,PP,B$,QQ,C$,RR,D$,
SS,E$,TT,F$,UU,G$,VV,H$,WW,I$,XX
,J$,YY,K$,ZZ,L$,AAA,M$,BBB,, ,PP+
QQ+RR+SS+TT+UU+VV+WW+XX+YY+ZZ+AA
A+BBB
    
```



TARIFA DE SUSCRIPCION

	CORREO ORDINARIO		CORREO CERTIFICADO		CORREO AEREO		CORREO AEREO-CERTIF.	
	PTAS.	\$	PTAS.	\$	PTAS.	\$	PTAS.	\$
ESPAÑA	2.200	16	2.453	18	2.244	16	2.497	18
EUROPA, MARRUECOS, TUNEZ, TURQUIA, ARGELIA Y CHIPRE .	2.618	19	3.366	24	2.717	19	3.465	25
COSTA RICA, CUBA, CHILE, PA- RAGUAY Y REP. DOMINICANA .	2.563	18	3.311	24	3.091	22	3.839	27
GIBRALTAR Y PORTUGAL	2.442	17	3.190	23	2.305	16	3.053	22
FILIPINAS	2.442	17	2.695	19	2.734	20	2.987	21
RESTO DEL MUNDO	2.618	19	3.366	24	3.146	22	3.894	28

CUPON DE PEDIDO

Recorte y envíe este CUPON DE PEDIDO a: **ZX** Jerez, 3 Madrid 16

NOMBRE _____

CALLE _____ N.º _____

CIUDAD _____ D.P. _____ PROVINCIA _____



EL LIMPIADOR DE VENTANAS

Todo lo que se necesita para poner en práctica este juego es saber subirse a una escalera. Después hay que intentar hacer equilibrio en la parte superior de la misma, con la intención de alcanzar una ventana. Las teclas M y N sirven para el movimiento a derecha o izquierda, el stop proporciona el movimiento hacia arriba. Si se golpea el lateral de la escalera, la muerte es instantánea, pero no importa demasiado, el juego nos concede tres vidas (cuatro menos que a los gatos), si es que no se acaba antes el tiempo.

Una vez alcanzada la ventana se comienza otra nueva escalera, pero se cuenta con menos tiempo para subirla. Con la tercera escalera subida, se obtiene un limpiaventanas más de recompensa. Las notas gráficas son: 40 = E gráfica con SIHT, 7 gráfico

con SHIFT, R gráfica con SHIFT. 60 = 5 gráfico con SHIFT. 70 = 5 gráfico con SHIFT. 80 = 5 gráfico con SHIFT, espacio, 2 espacios, 5 gráficos con SHIFT. 90 = W gráfica con SHIFT, 12 gráficos con 6 SHIFT, 5 gráficos con SHIFT.

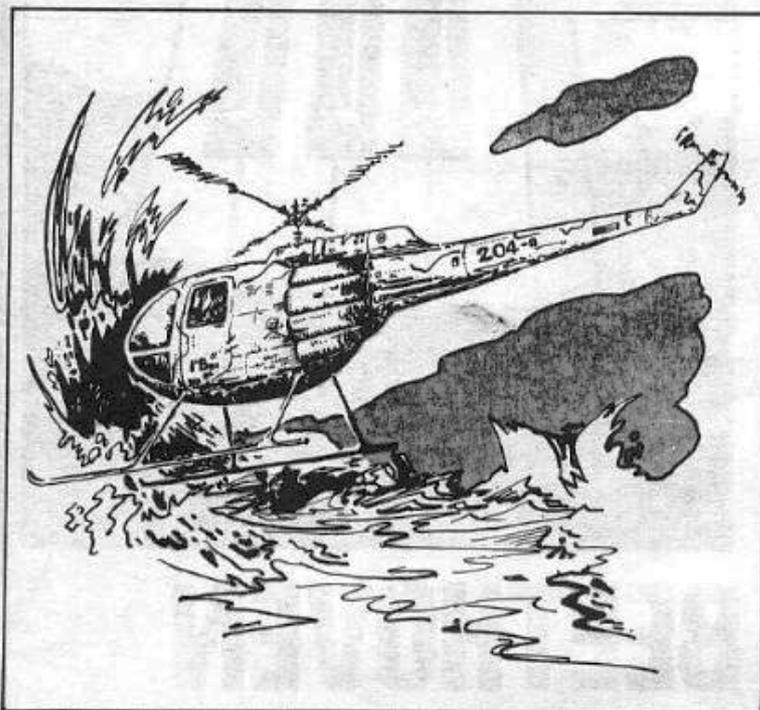
Ahora bien, como este programa ha sido en una impresora que no es la propia de Sinclair (con lo que hemos ganado en calidad de reproducción), deberá usted tener en consideración lo siguiente:

- en la línea 40 sustituir # por E gráfico, * por 7 gráfico y # por R gráfico.
- en las líneas 60, 70 y 80 sustituir* por 5 gráficos.
- en la línea 90 sustituir \$ por W gráfico, * por espacio gráfico y # por 5 gráficos.
- en la línea 160 sustituir \$ por E gráfico, * por 7 gráfico y # por R gráfico.

```
1 REM LIMPIA VENTANAS
2 LET HS=0
5 LET V=3
```

```
6 LET S=0
7 LET X=500
10 LET W=0
15 LET C=21
20 LET D=15
25 LET T=X
30 LET A=5
35 LET B=5
40 PRINT AT 21,14; "###"
45 FOR N=0 TO V
46 PRINT AT 21,N; "X"
47 NEXT N
48 PRINT AT 21,V; " "
50 PRINT AT 0,0; "PUNT.= ";S
60 PRINT AT 0,9; "*P.MAX. = ";HS
;
70 PRINT AT 0,21; "*TIEMPO=";T;
" "
80 PRINT AT 1,9; "*VENTANAS=";W
90 PRINT AT 2,9; "*****"
"
100 PRINT AT A,B-1; " "
110 IF INKEY$="." AND A>3 THEN
LET A=A-2
120 LET B=B+(INKEY$="M" AND B<2
B)-(INKEY$="N" AND B>2)
130 IF A<20 THEN LET A=A+1
140 LET B=B+(RND>.5 AND B<28)-(
RND<.6 AND B>2)
150 IF A>=C AND B=D-1 OR A>=C A
ND B=D+1 OR T=0 THEN GOTO 300
160 PRINT AT A,B-1; "###"
170 LET T=T-1
180 PRINT AT 0,29;T
190 IF A=C-1 AND B=D THEN GOTO
200
195 GOTO 100
200 LET S=S+1
210 LET C=C-1
220 IF C>4 THEN GOTO 30
230 CLS
240 LET C=21
250 LET W=W+1
260 LET X=X-25
270 IF W=3 THEN LET V=V+1
280 GOTO 25
300 LET V=V-1
310 IF V>0 THEN GOTO 25
320 LET S=S+(W*2)
330 IF HS<S THEN LET HS=S
340 INPUT A$
350 IF A$="N" THEN STOP
360 CLS
370 GOTO 5
```

PROGRAMAS



RESCATE EN EL MAR

Lograr aterrizar un helicóptero en un barco de reducidas dimensiones no es tarea fácil, pero con este juego podrá volverse un experto. No tema, su Spectrum no se lo va a poner muy difícil.

Como siempre, el movimiento lo logra a través de

las teclas 5, 6, 7, 8 y puede ver el tiempo que tarda en lograr su aterrizaje.

Notas gráficas:
1000 - Graphic C.
2050 - Graphic A y B.



Para maniobrar el helicóptero utilice las flechas.

```

5 BORDER 7: INK 0: PAPER 5: C
Ls 5 GO SUB 5000
10 LET b=0
1000 LET s=0: LET h$="✈": LET x=
1: LET sub=20: LET y=10
1020 PRINT PAPER 1: INK 7: AT 14,
0: "Para maniobrar el helicóptero
utilice las flechas."
1100 PRINT AT x,y:h$
1101 LET s=s+1
1105 IF x=9 THEN GO SUB 5000
1110 BEEP .125,-30: PRINT AT x,y
:"
1120 IF INKEY$="5" THEN LET y=y-
1
1130 IF INKEY$="8" THEN LET y=y+
1
1140 IF INKEY$="6" THEN LET x=x+
1
1150 IF INKEY$="7" THEN LET x=x-
1
1155 IF sub=20 THEN GO SUB 2000
1160 IF y<=1 THEN LET y=1
1165 IF y>=28 THEN LET y=28
1170 IF x>=9 THEN LET x=9
1180 IF y>=28 THEN LET y=28
1190 LET sub=sub+1
1200 GO TO 1100
200000 REM Barco
200005 LET sub=0
200010 PRINT AT 10,b:" "
200015 PRINT AT 9,b:" "
200020 LET b=INT (RND*20)+5
200025 PRINT INK 7: AT 9,b:
200030 PRINT INK 2: AT 10,b:
"
200035 RETURN
200040 IF ATTR (x+1,y+1)=42 THEN G
O TO 5100
200045 RETURN
200050 PRINT AT 10,1: "Tiempo: ";s
200055 PRINT AT 20,1: "Presione 0
para reiniciar el juego."
200060 IF INKEY$<>"0" THEN GO TO 5
1000
200065 GO TO 1
200070 REM Definición de caractere
s gráficos
200075 POKE USA "a"+0,BIN 11111111
200080 POKE USA "a"+1,BIN 01111111
200085 POKE USA "a"+2,BIN 00111111
200090 POKE USA "a"+3,BIN 00011111
200095 POKE USA "a"+4,BIN 00001111
200100 POKE USA "a"+5,BIN 00000111
200105 POKE USA "a"+6,BIN 00000011
200110 POKE USA "b"+0,BIN 11111111
200115 POKE USA "b"+1,BIN 11111110
200120 POKE USA "b"+2,BIN 11111110
200125 POKE USA "b"+3,BIN 11111100
200130 POKE USA "b"+4,BIN 11111000
200135 POKE USA "b"+5,BIN 11110000
200140 POKE USA "b"+6,BIN 11000000
200145 POKE USA "b"+7,BIN 10000000
200150 POKE USA "c"+0,BIN 11111111
200155 POKE USA "c"+1,BIN 00111100
200160 POKE USA "c"+2,BIN 00011100
200165 POKE USA "c"+3,BIN 01111110
200170 POKE USA "c"+4,BIN 11111111
200175 POKE USA "c"+5,BIN 11111111
200180 POKE USA "c"+6,BIN 00011000
200185 POKE USA "c"+7,BIN 00011000
200190 RETURN
    
```

```

1 REM 12345678901234567890123
456789012345678901234544
2 REM .....
.....
.....
.....
.....

```

```

10 LET A$="... "
20 LET A$=A$+"FFF"
25 LET O=16567
30 FOR I=1 TO LEN A$-1 STEP 2
40 POKE O, (CODE A$(I)-28)*16+C
ODE A$(I+1)-28
50 LET O=O+1
60 NEXT I
70 FAST
80 LET K=USR 16546
99 STOP

```

```

100 SAVE "BEETHOVEN"
110~PRINT "*****
*****BEETHOVE
*****
*****"

```

```

130 PRINT "A CADA NOTA CORRESPONDE UN VALOR HEXADECIMAL. PARA OBTENER LA MELODIA DESEADA INTRODUCIR SUCESIVAMENTE EN LA LINEA 10 (LET A$=""... ..") EL VALOR HEXADECIMAL CORRESPONDIENTE A LA NOTA EN EL ORDEN DE LA MELODIA."

```

```

140 PRINT "LAS NOTAS TIENEN UNA MISMA LONGITUD. PARA CONSEGUIR UNA DURACION MAYOR BASTA CON INTRODUCIR LAS VARIAS VECES. LOS INTERVALOS SE OBTIENEN INTRODUCIENDO EL VALOR ""00""."

```

```

145~PRINT AT 21,0;"PULSA UNA TECLA PARA CONTINUAR."

```

```

150 IF INKEY$="" THEN GOTO 150
155 CLS

```

```

160 PRINT "NOTA      1A ESCALA
2A ESCALA-----"

```

```

165 DIM B$(2,24)
170 LET B$(1)="FOE5D8CFC3B8ADA5
9C948B83"

```

```

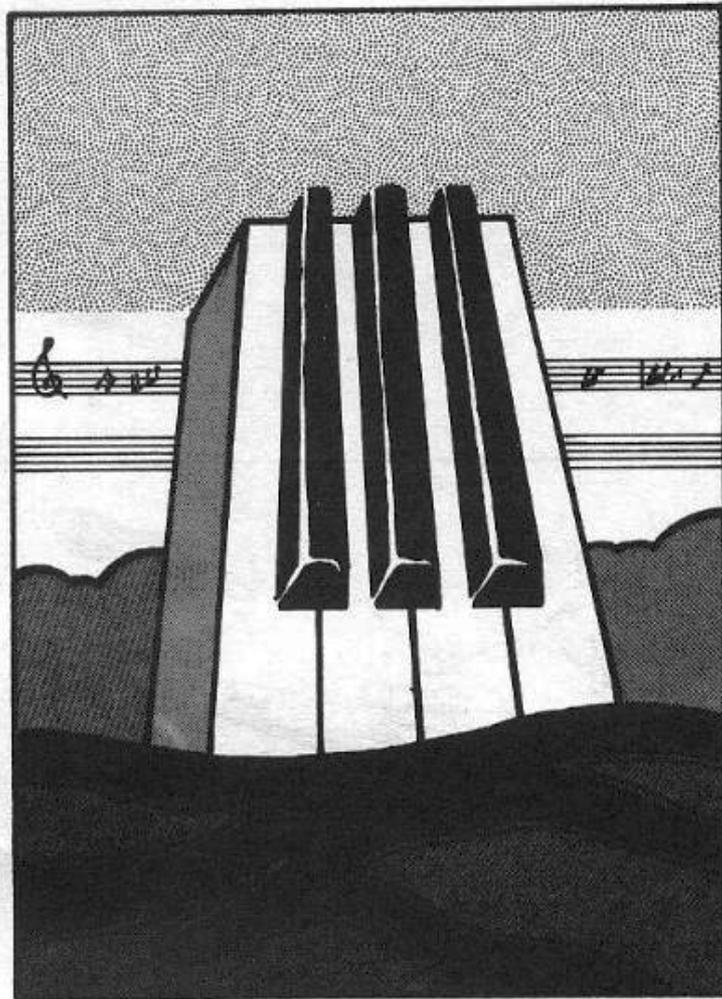
175 LET B$(2)="7A736B67605A5651
4B484440"

```

```

180 LET C$="C      C+;D-D      D+;E
-E      F      F+;G-G      G+;A-A      A

```



BEETHOVEN

Si durante el funcionamiento de su ZX-81 se ha fijado en los extraños ruidos que se producen a través del altavoz del televisor, le sorprenderá que con ellos se puede producir música. Con ellos, asociándole valores hexadecimales, se puede producir una tabla de correspondencia con los tonos de la escala musical.

Lo primero que se debe hacer es teclear GOTO 110, de manera que se puedan leer las instrucciones, la tabla de escalas y los códigos hexadecimales. Después habrá que pasar la melodía a hexadecimal. Introduciendo LIST 10, EDIT y alternando A\$ para convertirlo en la

lista de valores hexadecimales, se ha introducido la melodía. Subiendo el volumen del televisor y tecleando GOTO 1, se escuchará la melodía. Las letras que aparecen en minúscula en el listado deberán teclearse en video inverso.

Una aclaración necesaria: como en otros programas para ZX81 que presentamos en este número, Beethoven no ha sido listado con la impresora de Sinclair. Por tanto, el lector deberá traducir las normas en inglés, ya que si se empieza a ejecutar el programa sin leerlas puede perder los datos de la memoria.

EL FANTASMA

```

+;B-B      "
190 FOR I=1 TO 12
200 PRINT C$(I*5-4 TO I*5);TAB
24;B$(2,I*2-1 TO I*2)
210 NEXT I
215 STOP
220~PRINT "COMO EJEMPLO,ALGUNAS
MELODIAS ES-TAN CODIFICADAS EN
LAS LINEAS 1000... SI QUIERES
PROBARLAS PULSA run 1000,D 10
10.... sube el volumen del
tv Y pulsa return."
230 PRINT AT 21,0;"PULSA UNA TE
CLA PARA CONTINUAR."
240 IF INKEY#="" THEN GOTO 240
245 CLS
247 PRINT "ESCUSA,ESTOY TRABAJA
NDO..."
248 FOR I=1 TO 30
249 NEXT I
250 FAST
255 RUN 9990
1000 REM happy birthday
1005~LET A#="B8B800B800A5A500B8B
B00B8B800949494940000B8B800B800A
5A500B8B800B8B8009494949400B8B80
0B80060606073737300B8B8B80094940
0A5A5A5006B6B006B737300949400B3B
300949494"
1009 GOTO 20
1010 REM sinfonia n 40
1011 LET A#="B8C300C3C300B8C300C
3C300B8C300C3C37A7A7A007A8394940
094A5B8B800B8C3D8D800D8D8D800C3D
800D8D8C3D800D8D800C3D800D8D883B
3B30083949C9C009CB8C3C300C3DBF0F
000F0F0007A830083836B6B9C9C83839
494C3C3007A830083836B6B9C9C83839
4947A7A007A8394A5B8B8C3C3B8B8"
1019 GOTO 20
9989 STOP
9990 LET A#="7B3D20FDC906505EAFB
B2B0CDBFECD8240D3FFCD82401B050E0
50D20FD10E8C921B740CD8740237EFEF
F20F7C9"
9991~LET O=16514
9992 FOR I=1 TO LEN A#-1 STEP 2
9993 POKE O,(CODE A$(I)-28)*16+C
ODE A$(I+1)-28
9994 LET O=O+1
9996 NEXT I
9997 SLOW
9998 CLS

```

Este programa pone en juego a un fantasma inteligente. El nivel de dificultad se establece mediante las teclas 1 ó 0, el 1 es el más sencillo. Una vez hecho esto, se le aproximará un fantasma.

Para huir de él se pulsarán las teclas de movimiento del cursor, comiendo los puntos, para obtener una puntuación ascendente.

Si se limpia la pantalla,

habrá que presionar P y establecer el nivel de dificultad, pudiéndose continuar con la misma puntuación obtenida.

Estos juegos cruentos son difíciles si se desea salir con bien de ellos. Le advertimos que cuando lo juegue en una noche tormentosa acuértese de mirar de vez en cuando a su espalda, podría encontrarse con alguna desagradable sorpresa.

```

10 LET S=0
14 CLS
15 INPUT A
20 LET Q=1
30 LET W=1
40 LET R=INT (RND*7)+1
50 LET E=INT (RND*7)+1
60 FOR F=1 TO 10
110 PRINT "10 stops graficos"
120 NEXT F
130 IF INKEY#="P" THEN GOTO 14
190 PRINT AT Q,W;"< inverso"
200 IF PEEK (PEEK 16398+256*PEEK
K 16399)=155 THEN GOSUB 1000
300 LET Q=Q+(INKEY#="6" AND Q<B
)-(INKEY#="7" AND Q>1)
310 LET W=W+(INKEY#="8" AND W<B
)-(INKEY#="5" AND W>1)
314 IF Q=R AND W=E THEN GOTO 70
00
315 IF RND<A THEN GOTO 130
320 PRINT AT R,E;CHR$ 11
322 IF PEEK (PEEK 16398+256*PEEK
K 16399)-128 OR 155 THEN PRINT A
T R,E;"espacio inverso"
330 LET R=R+(Q>R)-(Q<R)
340 LET E=E+(W>E)-(W<E)
900 GOTO 130
1100 LET S=S+1
1150 PRINT AT Q,W;"espacio inver
so"
1200 RETURN
7000 PRINT S

```



SALTANDO ESTABA LA RANA

Lo que se pretende con este juego es ayudar a una rana a encontrar su camino.

En principio, la rana debe salir de un pequeño estanque de nenúfares, el cual debe cruzar. Después debe sortear a los veloces automóviles que cruzan una autopista para llegar a su casa situada en el otro lado de la misma.

El programa va dando instrucciones a medida que se las reclama para simplificar la tarea.

El movimiento de la rana se logra mediante las teclas 5, 7 y 8. Las notas gráficas que aparecen son como siguen: 107 = F inversa, 510 = 5 gráfico, dos A gráficas, 8

gráfico, 512 = C inversa, 519 = L inversa, 551 = F inversa y 2521 = F inversa.

Cuando ayude a la rana recuerde que podría tratarse de un príncipe encantado que quizá le proporcione los tres deseos si logra desencantarlo.

Para listar este programa con la impresora ZX tome en cuenta que en las líneas 107, 512, 513, 515, 519, 520, 551, 2512, 9001 y 9002 las letras minúsculas deben escribirse en inverso. En la línea 511 sustituir por el espacio inverso. En la línea 510 sustituir # por 5 gráficos, \$ por A gráfico, \$ por A gráfico y # por 8 gráficos.

```

3 PRINT AT 11,4;"SALTO DE LA
RANA"
4 PAUSE 100
5 CLS
6 PRINT "NECESITAS INSTRUCCIO
NES ?"
7 INPUT P$
8 IF P$="S" THEN GOSUB 8999
9 CLS
10 LET H=0
12 LET H=H
14 LET F=1
15 LET S=0
16 LET E=20
17 LET F=F
18 PRINT AT 21,2;"FRANJA ";F

```

PROGRAMAS

```
19 PRINT AT 21,15;"PUNTOS ";S
20 LET S=S
95 LET C=INT (RND*25)+3
100 FOR A=0 TO 3
105 LET E=E-2*(INKEY$="7")
106 LET C=C-2*(INKEY$="5")+2*(I
NKEY$="8")
107 PRINT AT E,C;"f"
108 IF C>28 THEN LET C=28
109 IF C<5 THEN LET C=5
110 GOSUB 2500
115 GOSUB 5000
118 IF E<=0 AND E=CODE " " THEN
GOTO 2000
119 IF INKEY$="7" THEN LET S=S+
10
120 PRINT AT 21,15;"PUNTOS ";S
125 IF E<14 THEN GOTO 509
130 IF E>16 THEN GOTO 509
140 IF E=16 THEN GOSUB 3000
150 IF E=14 THEN GOSUB 3120
400 GOSUB 4000
510 PRINT AT 0,A;" #### ####
#### #### "
511 PRINT AT 2,0;"#####
#####";AT 3,0;"###
#####";A
T 12,0;"#####
#####";AT 13,0;"#####
#####";AT 18,0;"###
#####";
AT 19,6;"PUNT.MAXIMA: ";H;AT 20,
0;"#####
###"
512 PRINT AT 4,A;" c c
c c"
513 PRINT AT 6,A;" c c c
c"
514 PRINT AT 8,A;" c c
c c c"
515 PRINT AT 10,A;" c c
c c"
519 PRINT AT 14,6;" 1 1 1 1
1 1 1 1"
520 PRINT AT 16,6;" 1 1 1 1
1 1 1 1"
530 NEXT A
540 IF A+23>=31 THEN GOTO 550
545 IF A+23>=0 THEN GOTO 100
550 FOR A=8 TO 0 STEP -1
551 PRINT AT E,C;"f"
```

```
552 IF C>28 THEN LET C=28
553 IF C<5 THEN LET C=5
555 LET E=E-2*(INKEY$="7")
556 LET C=C-2*(INKEY$="5")+2*(I
NKEY$="8")
558 GOSUB 2500
566 IF E<=0 AND E=CODE " " THEN
GOTO 2000
567 GOSUB 5000
568 IF INKEY$="7" THEN LET S=S+
10
569 PRINT AT 21,15;"PUNTOS ";S
570 IF E<14 THEN GOTO 1600
575 IF E>16 THEN GOTO 1600
580 GOSUB 3000
590 GOSUB 4000
1600 GOTO 509
2000 PRINT AT E,C;"FALLASTE"
2004 PRINT AT 21,15;"PUNTOS ";S
2006 GOTO 9500
2500 IF E<=0 AND C=A+2 THEN GOTO
2510
2501 IF E<=0 AND C=A+3 THEN GOTO
2510
2502 IF E<=0 AND C=A+8 THEN GOTO
2510
2503 IF E<=0 AND C=A+9 THEN GOTO
2510
2504 IF E<=0 AND C=A+14 THEN GOT
O 2510
2505 IF E<=0 AND C=A+15 THEN GOT
O 2510
2506 IF E<=0 AND C=A+20 THEN GOT
O 2510
2507 IF E<=0 AND C=A+21 THEN GOT
O 2510
2508 RETURN
2510 LET F=F+1
2512 PRINT AT E,C;"f"
2515 PRINT AT 21,15;"PUNTOS ";S
2520 PAUSE 75
2525 LET S=S+100
2530 CLS
2535 GOTO 16
3000 IF E=16 AND C=6 THEN GOTO 5
09
3001 IF E=16 AND C=9 THEN GOTO 5
09
3002 IF E=16 AND C=12 THEN GOTO
509
3003 IF E=16 AND C=15 THEN GOTO
```

```

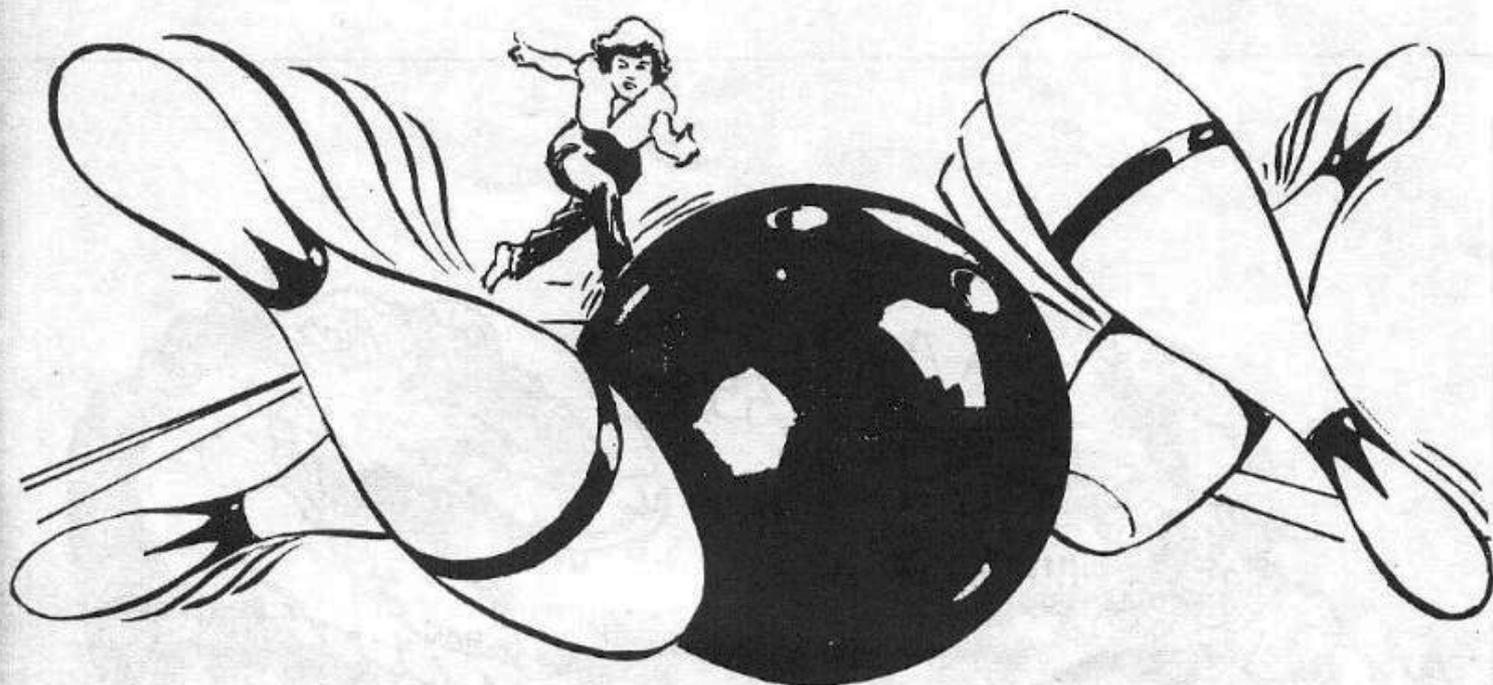
509
3004 IF E=16 AND C=18 THEN GOTO
509
3005 IF E=16 AND C=21 THEN GOTO
509
3006 IF E=16 AND C=24 THEN GOTO
509
3007 IF E=16 AND C=27 THEN GOTO
509
3120 IF E=14 AND C=6 THEN GOTO 5
09
3121 IF E=14 AND C=9 THEN GOTO 5
09
3122 IF E=14 AND C=12 THEN GOTO
509
3123 IF E=14 AND C=15 THEN GOTO
509
3124 IF E=14 AND C=18 THEN GOTO
509
3125 IF E=14 AND C=21 THEN GOTO
509
3126 IF E=14 AND C=24 THEN GOTO
509
3127 IF E=14 AND C=27 THEN GOTO
509
4000 PRINT AT E,C;"PLOP"
4001 PRINT AT 21,15;"PUNTOS ";S
4002 GOTO 9500
5000 IF E=4 AND C=A+1 THEN GOTO
5400
5010 IF E=4 AND C=A+8 THEN GOTO
5020 IF E=4 AND C=A+15 THEN GOTO
5400
5030 IF E=4 AND C=A+22 THEN GOTO
5400
5040 IF E=6 AND C=A+1 THEN GOTO
5400
5050 IF E=6 AND C=A+5 THEN GOTO
5400
5060 IF E=6 AND C=A+9 THEN GOTO
5400
5070 IF E=6 AND C=A+13 THEN GOTO
5400
5080 IF E=8 AND C=A+10 THEN GOTO
5400
5090 IF E=8 AND C=A+14 THEN GOTO
5400
5100 IF E=8 AND C=A+18 THEN GOTO
5400
5110 IF E=8 AND C=A+22 THEN GOTO
5400

```

```

5120 IF E=10 AND C=A+1 THEN GOTO
5400
5130 IF E=10 AND C=A+8 THEN GOTO
5400
5140 IF E=10 AND C=A+15 THEN GOT
0 5400
5150 IF E=10 AND C=A+22 THEN GOT
0 5400
5300 RETURN
5400 PRINT AT E,C;"PAFF"
5401 PRINT AT 21,15;"PUNTOS ";S
5402 GOTO 9500
8999 CLS
9000 PRINT AT 0,11;"SALTO DE LA
RANA"
9001 PRINT AT 3,3;"DEBES HACER S
ATAR A LA RANA f"
9002 PRINT AT 4,0;"A TRAVES DE L
AS FLORES ""1""
9003 PRINT AT 5,0;"HACIA LA CARR
ETERA. LA TIENES"
9004 PRINT AT 6,0;"QUE ATRAVESAR
ESQUIVANDO COCHES"
9005 PRINT AT 7,0;"(QUE PUEDEN C
AMBIAR DE SENTIDO)"
9006 PRINT AT 8,0;"Y METER A LA
RANA EN SU CASA"
9007 PRINT AT 10,0;"SOLO PUNTUAN
LOS MOVIMIENTOS"
9008 PRINT AT 11,0;"HACIA ADELAN
TE,PERO GANAS "
9009 PRINT AT 12,0;"100 PUNTOS A
L ATRAVESAR UNA "
9010 PRINT AT 13,0;"FRANJA."
9011PRINT AT 15,0;"LOS MOVMIENT
OS SE HACEN CON LAS"
9012 PRINT AT 16,0;"TECLAS DEL C
URSOR (5,6,7 Y 8)"
9013 PRINT AT 18,0;"PULSA NL PAR
A CONTINUAR..."
9014 PAUSE 4E4
9015 IF INKEY$="" THEN CLS
9016 RETURN
9500 PAUSE 200
9501 IF H<S THEN LET H=S
9510 CLS
9519 PRINT "QUIERES JUGAR DE NUE
VO?"
9520 INPUT Y$
9530 IF Y$="S" THEN GOTO 12
9540 STOP

```



BOWLING PARA 1 KBYTE

Si le apetece jugar a los bolos durante una lluviosa tarde de invierno, no se preocupe en buscar el paraguas si dispone de un ZX-81 con la mínima memoria de 1 Kbyte.

En la parte superior de la pantalla aparecerán 16 bolos. En la parte inferior se muestra la bola, que puede ser alineada utilizando las teclas 5 y 8. Para lanzarla, bastará con solo presionar la

tecla O. Cuando impacta con los bolos se obtendrá un efecto muy real, la trayectoria cambia con cada golpe a un bolo, como en el juego habitual.

Cada bolo derribado pro-

porcionará un punto más. El juego termina cuando se lanzan hasta 8 bolas, apareciendo la puntuación en la pantalla. Así que a entrenarse para obtener plenos en su próxima competición en la bolera.

```

1 LET Q=CODE ""
2 LET S=Q
3 LET C=Q
4 LET M=CODE "1 grafico"
5 FOR B=M TO CODE "4 grafico"
6 FOR A=M+C TO CODE "E grafic
o"-C
7 PRINT AT B,A;"T"
8 NEXT A
9 LET C=C+M
10 NEXT B
11 LET X=CODE "7 grafico"
20 LET F=M
25 LET Y=CODE "?"
30 LET X=X+(INKEY$="B" AND X<C
ODE "T grafico")-(INKEY$="5" AND
X>CODE "")
35 PRINT AT Y,X;" O "
40 IF F>CODE "E grafico" THEN
GOTO CODE "- inverso"

```

```

45 IF INKEY$="O" THEN GOSUB CO
DE "W"
50~GOTO CODE ";"
60 PRINT AT Y,X+M;
65 IF PEEK (PEEK 16398+VAL "25
6"*PEEK 16399)=CODE "T" THEN LET
Q=Q+M
70 PRINT AT Y,X+M;"O";AT Y,X+M
;"O inverso";AT Y,X+M;" "
75 LET Y=Y-M
80 IF Y=M-M THEN LET F=F+M
90 IF Q>S AND RND<VAL ".5" THE
N LET X=X-M
95 IF Q>S AND RND>VAL ".6" THE
N LET X=X+M
100 LET S=Q
110 IF Y=M-M THEN RETURN
120 GOTO CODE "W"
150 PRINT Q

```



EL ESPANTOSO ALIENIGENA

Con este juego ya se pueden matar alienígenas de modo aficionado (A). Ahora bien, si usted es un consumado cazador puede optar por el modo profesional (P).

La acción consiste en lo siguiente: en la parte inferior izquierda de la pantalla aparece una pequeña rejilla que contiene 4 marcianos, que tienen la forma de \$ y X. Un

soldado debe cavar un hoyo para introduciendo a los invasores. Por cada marciano de la clase X se consigue una bonificación de 100 puntos y por cada \$ serán 200.

El hoyo se cava presionando el 2 y la tecla de movimiento de cursor indica a qué lado el soldado va a cavar. El soldado se puede mover con los 0 1 y también con las teclas de cursor.

El soldado también corre sus riesgos en la lucha, tiene dos modos. El primero consiste en caer en su propia trampa, los hoyos, o ser devorado por el marciano.

De todas formas, el juego acaba cuando desaparecen tres soldados sucesivamente. Cuando se obtiene una puntuación mayor de 3.000 puntos aparecerá un cuarto soldado de regalo.

El hoyo debe ser cavado en un máximo tiempo determinado y tampoco se puede cavar bajo los pies del marciano.

Cuando pase este programa a su ZX-81, con 16 Kbytes de memoria, recuerde que no es conveniente cazar mañana los marcianos que pueda destruir hoy.

PROGRAMAS

```
1 LET BEST=0
2 PRINT "DIFICULTAD?(A/P)"
3 INPUT A$
5 CLS
10 LET SO=2
15 LET S=0
20 PRINT "" "INFOME""
21 PRINT AT 14,14;"NIVEL:";A$
22 PRINT AT 16,10;"PUNT.MAX.:"
;BEST
23 PRINT AT 18,0;"TU PUNTUACION ES :";S
30 PRINT AT 20,0;"HOMBRES DISPONIBLES:";SO
40 LET X=17
50 LET Y=0
60 LET A=17
70 LET B=8
80 LET U=21
90 LET V=4
100 LET G=1
110 LET HX=15
120 LET HY=0
130 LET C=17
140 LET D=4
150 LET E=19
160 LET F=2
165 LET GG=19
170 LET HH=6
175 PAUSE 200
180 CLS
195 PRINT AT U,V;"*"
200 PRINT AT X,Y;"$"
210 PRINT AT A,B;"$"
220 PRINT AT C,D;"X"
230 PRINT AT E,F;"X"
240 PRINT AT GG,HH;"X"
300 LET Q=141
305 LET M=X
310 LET N=Y
320 GOSUB 2000
322 LET W=345
325 GOTO 6000
335 LET X=M
340 LET Y=N
345 PRINT AT X,Y;"$"
347 GOSUB 2430
```

```
350 LET M=A
360 LET N=B
370 GOSUB 2000
375 LET W=410
380 GOTO 6000
400 LET B=N
405 LET A=M
410 PRINT AT A,B;"$"
415 GOSUB 2430
417 LET Q=189
420 LET M=C
440 LET N=D
460 GOSUB 2000
470 LET W=510
475 GOTO 6000
500 LET D=N
505 LET C=M
510 PRINT AT C,D;"X"
515 GOSUB 2430
520 LET M=E
540 LET N=F
560 GOSUB 2000
570 LET W=610
580 GOTO 6000
600 LET F=N
605 LET E=M
610 PRINT AT E,F;"X"
615 GOSUB 2430
620 LET M=GG
640 LET N=HH
660 GOSUB 2000
670 LET W=710
680 GOTO 6000
700 LET HH=N
705 LET GG=M
710 PRINT AT GG,HH;"X"
720 GOSUB 2430
770 GOTO 300
2003 IF INKEY$="1" THEN LET G=1
2004 IF INKEY$="2" THEN LET G=2
2005 LET R=RND
2010 LET Z=RND
2020 PRINT AT M,N;" "
2090 IF M=18 OR M=20 OR N=V AND A$="P" AND (N=0 OR N=2 OR N=4 OR N=6 OR N=8) THEN GOTO 2120
2100 IF R<=.5 OR M=U OR N=1 OR N
```

PROGRAMAS

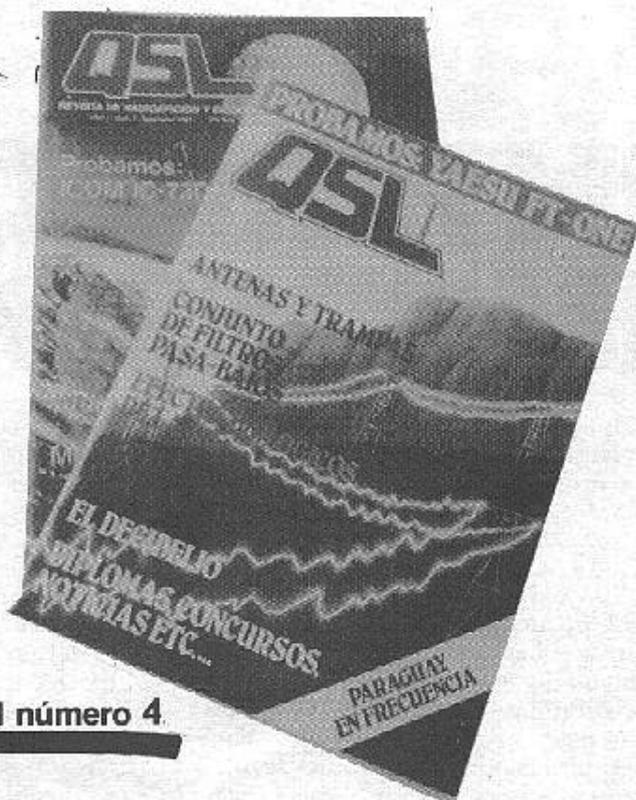
```
=3 OR N=5 OR N=7 THEN GOTO 2300
2120 IF M=21 OR U<M AND A$="P" T
HEN LET Z=.7
2122 IF M=17 OR U>M AND A$="P" T
HEN LET Z=.5
2125 IF Z<=.5 THEN LET M=M+1
2140 IF Z>.5 THEN LET M=M-1
2160 GOTO 2340
2300 IF N=0 OR N<V THEN LET Z=.5
2305 IF N=8 OR N>V THEN LET Z=.7
2310 IF Z<=.5 THEN LET N=N+1
2320 IF Z>.5 AND N>0 THEN LET N=
N-1
2340 IF M=U AND N=V THEN GOTO 70
00
2400 IF INKEY$="2" THEN LET G=2
2410 IF INKEY$="1" THEN LET G=1
2420 RETURN
2430 IF INKEY$<>"5" AND INKEY$<>
"6" AND INKEY$<>"7" AND INKEY$<>
"8" THEN RETURN
2440 IF G=2 THEN GOTO 2590
2450 PRINT AT U,V;" "
2520 IF INKEY$="5" AND V>0 AND (
U=17 OR U=19 OR U=21) THEN LET V
=V-1
2540 IF INKEY$="8" AND V<8 AND (
U=17 OR U=19 OR U=21) THEN LET V
=V+1
2560 IF INKEY$="6" AND U<21 AND
(V=0 OR V=2 OR V=4 OR V=6 OR V=8
) THEN LET U=U+1
2580 IF INKEY$="7" AND U>17 AND
(V=0 OR V=2 OR V=4 OR V=6 OR V=8
) THEN LET U=U-1
2585 GOTO 3005
2590 PRINT AT HX,HY;" "
2597 IF INKEY$="5" AND V>0 AND (
U=21 OR U=19 OR U=17) THEN GOTO
2700
2600 IF INKEY$="8" AND V<8 AND (
U=21 OR U=19 OR U=17) THEN GOTO
2750
2620 IF INKEY$="6" AND U<21 AND
(V=0 OR V=2 OR V=4 OR V=6 OR V=8
) THEN GOTO 2800
2640 IF INKEY$="7" AND U>17 AND
(V=0 OR V=2 OR V=4 OR V=6 OR V=8
) THEN GOTO 2850
2660 GOTO 3005
2700 LET HX=U
```

```
2710 LET HY=V-1
2720 GOTO 3000
2750 LET HX=U
2760 LET HY=V+1
2770 GOTO 3000
2800 LET HX=U+1
2810 LET HY=V
2820 GOTO 3000
2850 LET HX=U-1
2860 LET HY=V
3000 IF HX=X AND HY=Y OR HX=A AN
D HY=B OR HX=C AND HY=D OR HX=E
AND HY=F OR HX=GG AND HY=HH THEN
LET HX=15
3003 IF HX<>15 THEN PRINT AT HX,
HY;"0"
3020 IF HX=U AND HY=V OR U=X AND
V=Y OR U=A AND V=B OR U=C AND V
=D OR U=E AND V=F OR U=GG AND V=
HH THEN GOTO 7000
3030 PRINT AT U,V;"*"
3040 RETURN
6000 IF M=X AND N=Y OR M=A AND N
=B OR M=C AND N=D OR M=E AND N=F
OR M=GG AND N=HH THEN GOTO W
6005 IF M=HX AND N=HY THEN GOTO
6020
6010 GOTO W-10
6020 IF Q=141 THEN LET S=S+100
6035 LET S=S+100
6040 PRINT AT M,N;CHR$ Q
6044 IF S/3000=INT (S/3000) OR (
S/100)/3000=INT ((S-100)/3000) T
HEN LET S0=S0+1
6045 PRINT AT 18,21;S
6047 PRINT AT 20,21;S0
6050 LET HX=15
6060 GOTO W-10
7000 PRINT AT U,V;"*"
7010 LET S0=S0-1
7030 PAUSE 100
7032 IF S0=-1 THEN GOTO 8000
7035 CLS
7040 GOTO 20
8000 FOR N=0 TO 21
8010 SCROLL
8020 NEXT N
8030 IF S>BEST THEN LET BEST=S
8040 PRINT AT 0,0;"FIN DEL JUEGO
,TODOS TUS HOMBRES HAN MUERTO"
8050 GOTO 2
```

QSL

REVISTA MENSUAL
DE RADIOAFICION Y DIEXISMO

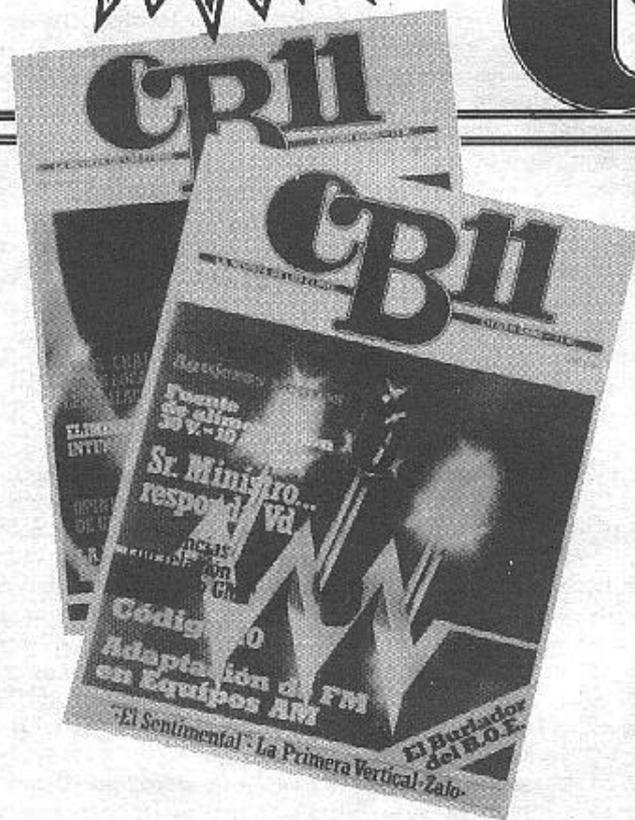
Publicación dirigida a los
RADIOAFICIONADOS
y DIEXISTAS. Su contenido refleja
los avances en este campo,
como así también sirve de enlace
escrito entre la afición
de España y América.



COMPRELAS
EN SU
KIOSCO

Ya está a la venta el número 4.

CB11



LA REVISTA MENSUAL DE LOS 27 MHZ.
CITIZEN BAND-11 M.

La nueva generación
de radioaficionados, usuarios
de la denominada
BANDA CIUDADANA
O CITIZEN BAND,
encuentran en esta
publicación mensual
un vehículo útil, ameno
y debidamente documentado
sobre este campo de la
Radioafición.

Ya está a la venta el número 11

INTERFACE PARA IMPRESORA RS-232

Aunque la mayoría de los lectores ya deben saberlo, no está de más comenzar este artículo diciendo que de los varios métodos existentes para conectar impresoras a los ordenadores, dos son los más corrientes: se les conoce como RS-232 y Centronics.

El primero utiliza básicamente un simple cable por el cual se envían los códigos de los caracteres en serie, es decir un bit después de otro, hasta completar el código. En cambio, cuando se emplea la norma Centronics, la información correspondiente a cada carácter es enviada de una sola vez, mediante siete cables en paralelo. Otros cables son utilizados para controlar cómo se realiza la transferencia de información.

Ambos sistemas son bastante incompatibles. En este artículo sólo hablaremos de un *interface* RS-232.

El término RS-232 define un *interface* serie que tiene una amplia gama de usos que van más allá de las impresoras. Entre ellos se incluyen los *modems*, los monitores de video y los enlaces entre sistemas. En tanto que los niveles de tensión, los tipos de conector y la distribución de patillas vienen especificados, se proporcionan distintas opciones para ajustarse a las necesidades de los diferentes instrumentos. Para asegurar la compatibilidad con todos los dispositivos que pueden encontrarse, el ordenador debe ir provisto de cierto grado de versatilidad. Las mayores variaciones se encontrarán en los códigos estándar que se utilizan. Afortunadamente, existen muchos circuitos integrados disponibles que se adaptan a toda la gama de códigos que probablemente puedan necesitarse. Uno de ellos es el que posibilita que el diseño que presentamos pueda realizarse de manera relativamente sencilla.

Las interconexiones en un sistema RS-232 se efectúan por medio de un conector "D" de 25 contactos, con el zócalo dispuesto en el ordenador. Las únicas patillas normalmente utilizadas se muestran en la figura 1 junto con sus descripciones funcionales, según las "ve" la impresora. Cuando se las lea, debe tenerse en cuenta que lo que para la

impresora es recepción de datos, es para el ordenador transmisión de datos. Si se olvida esto, aparecerán enseguida las confusiones. El carácter que se está enviando a la impresora en un instante determinado se hace por la patilla 3, en forma de bits en serie de acuerdo con el código ASCII, que requiere de 7 bits para definir cada carácter.

La frecuencia de transmisión de los bits por separado es conocida como velocidad de transferencia en Baudios. La fig. 2 muestra las velocidades más comúnmente empleadas, la velocidad real de transmisión de los caracteres es aproximadamente 1/10 de la velocidad en Baudios. En la fig. 3 se muestra la forma en que se envía el código. El primer bit transmitido es siempre un bit inicial de polaridad.

Tras él es enviado el código del carácter, que puede ser seguido por un bit de paridad. La paridad puede ser par o impar, dependiendo del estándar de paridad utilizado, y su misión es posibilitar que puedan detectarse errores haciendo que el número total de unos lógicos sea par o impar. Hay también uno o a veces dos bits de parada al final de cada carácter. La línea estará entonces preparada para pasar el próximo carácter, que comienza con su bit inicial.

El apretón de manos (*handshake*) se emplea muy a menudo entre un ordenador y una impresora. Esto capacita a la impresora para decirle al ordenador que deje de enviar más datos si ya tiene tantos como puede ir imprimiendo. Esto podría ocurrir por ejemplo después de que se envía un orden de vuelta del carro al principio de la línea, donde muchas impresoras necesitan demasiado tiempo para ejecutar esta función mecánica, más del que emplea para imprimir un carácter. El apretón de manos hace que se detenga el envío de más caracteres, los cuales se perderían mientras el carro está a medio camino de su vuelta. Un sistema puede tener "apretón de manos" por *hardware*, por *software* o ninguno. Si no existe ninguno, la velocidad en Baudios deberá ser establecida bastante lenta, para que la

información pueda ser impresa tan rápidamente como se recibe.

El "apretón de manos" por *hardware* utiliza la patilla 4, que forma un conector llamado "Request To Send", que posibilita la transmisión cuando su estado lógico es alto y la impide cuando está bajo. Esta es la única forma de "apretón de manos" que se proporciona en este montaje, aunque es posible que el usuario escriba su propia rutina de *software* si lo requiere. El "apretón de manos" por *software* es el sistema en el cual determinados códigos son enviados en ambas direcciones entre el ordenador y la impresora para iniciar y parar la transmisión de caracteres. Para utilizar este método se requiere que tanto las vías de recepción como el de transmisión existan. Para la comunicación de datos en el ordenador se utilizan las patillas 2 y 5. La patilla 2 lleva la entrada de datos y la 5 lleva la salida de "apretón de manos" que dice al extremo que envía cuando puede empezar a hacerlo. Se verá que las patillas 2 y 5 son complementarias a las patillas 3 y 4. Si dos ordenadores tienen que comunicarse a través de una línea RS-232, entonces se requiere un cable especial que conecte la patilla 2 de un extremo a la patilla 2 del otro extremo del enlace y viceversa. se actúa de manera similar con las patillas 4 y 5. Esto se muestra en la fig. 4.

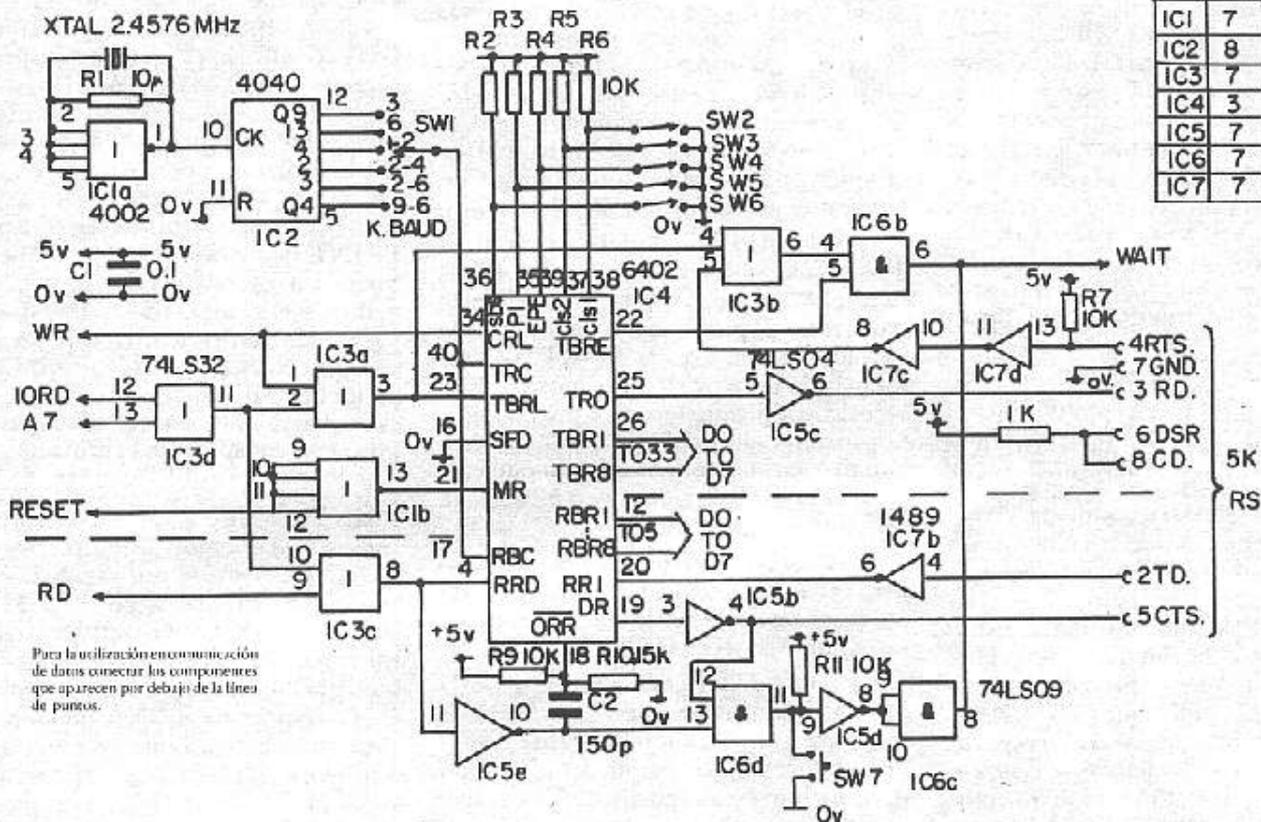
Un problema que se encuentra a veces cuando se conecta la impresora es la presencia o ausencia de autoalimentación de línea. La alimentación de línea es una instrucción que pasa la cabeza de la impresora a la línea siguiente, es decir sube el papel una línea hacia arriba.

Algunos ordenadores como el Spectrum terminan sus líneas con sólo un retorno de carro (código OD); se pide a la impresora que haga la alimentación de línea y que se vaya a la línea siguiente. A esto se le llama autoalimentación de línea. Algunas impresoras, sin embargo, requieren una alimentación de línea por separado (código OA); de otra forma toda la impresión se efectuará en la misma línea. Este diseño proporciona, por la adición de un alimentador de

MONTAJES

Número de las patillas para la alimentación.

	0V	5V
IC1	7	14
IC2	8	16
IC3	7	14
IC4	3	1
IC5	7	14
IC6	7	14
IC7	7	14



línea óptico, tras cada retorno del carro una localización de memoria durante el procedimiento de inicialización con método POKE.

Otro problema común lo constituyen los diferentes juegos de caracteres. Los juegos de caracteres definen las formas de los caracteres que van a ser visualizados o impresos en respuesta a los códigos de caracteres. No existe un estándar utilizado universalmente; aunque la mayoría de las máquinas conforman aproximadamente el ASCII y proporcionarán correctamente los números 0 a 9, las letras A a Z y los símbolos aritméticos, el resto de los caracteres varían con mucha frecuencia. Por ejemplo, sólo las impresoras para idioma inglés utilizan el carácter ® . Hay poca cosa que pueda hacer el usuario excepto recordar cuál es el verdadero significado de los signos diferentes que le proporciona la impresora. Este problema no surge con la Impresora ZX, porque su juego de caracteres está den-

Fig. 1

Velocidades de transferencia estándar en baudios

50
100
300
600
1200
2400
4800
9600

tro del ordenador, que le va diciendo a la impresora lo que debe imprimir punto por punto.

El diseño del *interface* está basado en un circuito integrado en Intersil, un receptor/transmisor universal asíncro-

Fig. 2 Conector RS-232

Referente a la impresora.		
Número de patilla	Descripción	Dirección de la señal.
1	Protección a masa.	
2	Datos transmitidos.	Salida
3	Datos recibidos.	Entrada
4	Petición de envío.	Salida
5	Poner a cero para enviar.	Entrada
6	Datos listos.	Entrada
7	Señal de masa.	
8	Detector de portadora.	Entrada

no (más conocido por UART), que realiza todo el trabajo de codificar los datos en forma serial. La única circuitería extra necesaria es un oscilador patrón, un simple decodificador de direcciones y los *buffers* de línea RS-232. El circuito aparece en la fig. 5 y para

utilizar impresora sólo se requiere la parte por encima de la línea de puntos. A continuación explicamos brevemente el funcionamiento del circuito. El decodificador de direcciones IC3a enviará un pulso de escritura a la UART siempre que haya una instrucción OUT que actúe en el *port* número 65407. Esto se distingue de la dirección de memoria del mismo valor en que la línea IORQ está activada en baja. Aunque el circuito sólo requiere que el número del *port* tenga la dirección A7 en baja, el resto de las líneas de dirección deben estar en alta para evitar que ningún otro dispositivo entre en funcionamiento en el Spectrum.

El código del carácter está tomado desde el bus por la UART y almacenado en su registro *buffer* para transmisión desde el cual será enviado juntamente con los bits de inicio, parada y paridad por la salida en serie, situada en la patilla 22. Debido a que la transmisión en serie es mucho más lenta que la velocidad normal de trabajo del ordenador, es probable que los datos estén disponibles con mayor frecuencia de la que la UART puede enviarlos. Para evitar la pérdida de datos, la UART producirá que la línea WAIT (espera) sea activada, en baja, cuando su registro *buffer* para transmisión esté lleno, indicado porque la patilla 22, TRBE, esté en baja. Si el "apretón de manos" (*handshake*) RTS, está en baja y aparece otro carácter para ser escrito en la UART, entonces IC3b pasará a estar en baja, causando un WAIT hasta que RTS se ponga en alta. El *buffer* de entrada RS-232, IC7, funcionará con entradas TTL (tensiones de 0 y 5 voltios), pero también aceptará niveles de hasta ± 15 voltios sin peligro de daños. Los datos en serie son transmitidos a la línea RS-232 por medio de IC5c. Esta es una puerta

TTL normal y su entrada no conforma totalmente los niveles de tensión RS-232 en términos del nivel lógico cero. Este debería ser -3 voltios, sin embargo, el *driver* proporciona 0 voltios. En la práctica, nos encontraremos con que casi todas las impresoras RS-232 trabajarán correctamente con los niveles TTL proporcionados. Este atajo se hizo en el diseño debido a la dificultad de obtener una línea de alimentación con tensión negativa a partir del Spectrum.

La velocidad de transferencia en Baudios es seleccionada mediante el conmutador SW1 que conecta la frecuencia requerida a partir del contador a la entrada de reloj de la UART. La frecuencia de reloj es 16 veces mayor que la velocidad de transferencia en Baudios. Los conmutadores SW2 a SW6 seleccionan el estándar de transmisión que va a utilizar la UART. La figura 6 muestra la combinación disponible.

Si el estándar requerido para la impresora es desconocido, la mayoría de las veces será suficiente con poner todos los conmutadores en OFF.

La parte del circuito situada bajo la línea de puntos era necesaria, sólo se requiere comunicación de datos con el ordenador. Se ha ampliado el diseño para incluir esta capacidad, puesto que no utiliza circuitos integrados adicionales y requiere sólo 3 resistencias, un condensador y un pulsador. Es controlado a partir de una instrucción IN también desde el *port* 65407. Cuando se hace esto, la línea WAIT pasará a baja y parará todo el procesamiento hasta que la UART indique por medio de su patilla 19, de datos recibidos, que se ha recibido algún código en la entrada serie. El código pasará entonces al bus de datos. El pulsador SW7 actúa como retirada para permitir al usuario pueda

interrumpir la operación IN si no hay datos disponibles.

En aplicaciones sencillas, el *port* de salida puede manejar un carácter a la vez a partir del BASIC. El comando OUT 65407 hará que sea impreso el carácter cuyo código ASCII es X. A pesar de que esto permite un control total sobre la impresora, dista mucho de ser la forma más conveniente de imprimir tablas, texto o listados. Sería mucho mejor el poder utilizar las sentencias PRINT normales. Afortunadamente el Spectrum es muy versátil en sus capacidades de entrada/salida y está diseñado para poder utilizar otros canales aparte de los estándar y está diseñado para poder utilizar otros canales aparte de los estándar de la pantalla, el teclado o la impresora ZX.

Cuando se ejecuta una sentencia PRINT o INPUT, el microprocesador leerá en la memoria la dirección de la rutina que hay que utilizar. Estas direcciones son guardadas en la RAM entre las 5CB6 y 5CC9 en un área llamada Información sobre los canales. Estas localizaciones de RAM son establecidas en el momento de la conexión de la alimentación para que se refieran a las direcciones de las rutinas apropiadas de la ROM. Todo lo que se tiene que hacer es utilizar la instrucción POKE con el puntero de la rutina de salida por impresora para que sea la dirección utilizada. Cada vez que se efectúe un LPRINT, LLIST o PRINT 3, la salida irá automáticamente al RS-232. Las direcciones del POKE son 23749 y 23750 (5CC5 y 5CC6 en hexadecimal) que van a contener la dirección de 16 bits de la rutina del usuario, escrita de la forma corriente para el Z-80, el byte menos significativo en primer lugar.

En su forma elemental, la rutina de

Fig. 3 Formato de una transmisión en serie.

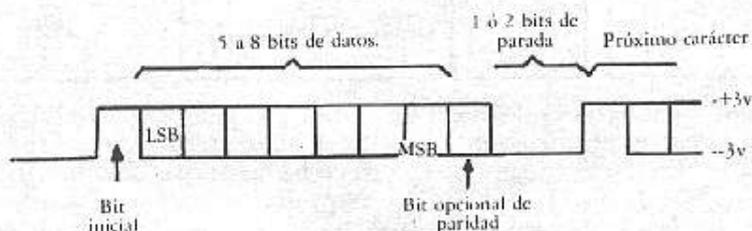


Fig. 4. Diagrama de cableado para la interconexión de ordenadores.



MONTAJES

salida tiene que tomar el código del carácter, que está en el registro A, y hacer un OUT con él a la dirección del port 7F. Esto no tiene ningún problema con los caracteres alfanuméricos, pero no descifrará palabras clave o responderá a funciones de control tales como TAB. El programa dado hace ambas cosas e insertará automáticamente el retorno del carro, pasando a la línea siguiente si una línea excede una longitud predeterminada. La longitud de la

línea es almacenada como un número en la localización de memoria 23681 (5C81) y debe hacerse con ella un POKE antes de comenzar. Adicionalmente, el programa monitoriza la dirección 23728 (5CB0) y si con el POKE se ha hecho que valga 2, se añadirá una orden de paso a otra línea tras cada retorno del carro. Para ahorrar el tener que hacer tantos POKEs, se proporciona un programa en código máquina que haga la inicialización, que puede ser guardado

junto con la propia rutina de salida. A esto se le llama una sola vez mediante una instrucción PRINT USER.

La rutina de salida ha sido hecha relocable y por tanto puede ser localizada en cualquier parte de la memoria. Sólo la rutina de inicialización tiene que ser alterada para apuntarle la dirección correcta de comienzo. Para mostrar que dos direcciones cambian la rutina de inicialización, se observará la figura 8. La cifra de la longitud de línea también puede ser cambiabile si es necesario. En la figura 9 se muestra un cargador del programa, que colocará el código en la parte superior de la RAM del Spectrum de 48 K por encima de la dirección 65024. Una vez que se ha pasado todo el código, se le podrá almacenar en cinta; su longitud es de 284 bytes. La sentencia CLEAR 65000 debería ser ejecutada antes de volver a cargar el código. Para el caso del Spectrum de 16 K, hay que sustraer 32768 de los números escritos entre paréntesis y alterar el duodécimo byte del código pasando a ser 7E en lugar de FE. Si dispone de una tarjeta para grabar EPROMs, una de ellas será

```

5 CLEAR 65000
10 LET a=65024
15 RESTORE
20 READ c$
30 LET c=CODE c$-48-(7*(c$(1)>
"@"))
40 LET d=CODE c$(2)-48-(7*(c$(
2)>"@"))
50 POKE a,1E*c+d
55 LET a=a+1
70 IF LEN c#=2 THEN GO TO 20
80 LET c#=c$(3 TO )
90 IF c$(1)<>" " THEN GO TO 30
100 LET c#=c$(2 TO ) GO TO 90
150 REM en las sentencias de da
tos se deben utilizar mayusculas
200 DATA "0E 00 10 02 0E 02 02
B0 5C 01 10 FE ED 43 05 5C"
210 DATA "0E 48 32 81 5C 09 FD CB
C8 76 4E CE 7F 20 30 FD CB
C8 76 96 C9 FD CB 76 D6 C9 FD 7E
FE40 77 16 0F C8 3E 20 CD 7D 1E FD 34 77 18 F0 FD CB
FE50 76 86 FD BE 77 38 07 FD 96 77 47 04 18 19 47 04
FE60 FD 36 77 FF 3E 0D CD 7D 1E FL CB 76 4E 28 05 3E
FE70 0A CD 7D 1E FD 34 77 3E 20 10 F6 C9 FE A5 30 48
FE80 FE 0D 28 32 FE 20 D8 FE 90 38 02 D6 2F FE 80 38
FE90 02 D6 1F FD 34 77 47 3A 81 5C FD BE 77 30 14 3F
FEA0 0D CD 7D 1E FD 36 77 01 FD CB 76 4E 18 05 3E 0A
FEB0 CD 7D 1E 78 18 0F FD 36 77 00 FD CB 76 4E 28 05
FEC0 CD 7D 1E 3E 0A C3 7D 1E 21 95 00 D6 A4 47 CD 30
FED0 03 23 E6 80 28 F8 05 20 F5 3E 20 CD 7D 1E FE 34
FEE0 77 FD 34 77 3A 81 5C FD BE 77 30 14 3E 0D CD 7D
F1FC 1E FD 36 77 00 FD CB 76 4E 28 05 3E 0A CD 7D 1E
FF00 CD 30 03 23 CB 17 38 07 CB 3F CD 7D 1E 18 D2 CB
FF10 3F CD 7D 1E FD 34 77 3E 20 C3 7D 1E 00 00 00

```

```

FE00 3E 00 18 02 3E 02 32 B0 5C 01 10 FE ED 43 05 5C
FE10 3E 48 32 81 5C 09 FD CB 76 4E CE 7F 20 30 FD CB
FE20 76 56 20 0C FE 06 28 16 FE 16 28 0D FE 17 20 4C
FE30 FD CB 76 C6 FD CB 76 96 C9 FD CB 76 D6 C9 FD 7E
FE40 77 16 0F C8 3E 20 CD 7D 1E FD 34 77 18 F0 FD CB
FE50 76 86 FD BE 77 38 07 FD 96 77 47 04 18 19 47 04
FE60 FD 36 77 FF 3E 0D CD 7D 1E FL CB 76 4E 28 05 3E
FE70 0A CD 7D 1E FD 34 77 3E 20 10 F6 C9 FE A5 30 48
FE80 FE 0D 28 32 FE 20 D8 FE 90 38 02 D6 2F FE 80 38
FE90 02 D6 1F FD 34 77 47 3A 81 5C FD BE 77 30 14 3F
FEA0 0D CD 7D 1E FD 36 77 01 FD CB 76 4E 18 05 3E 0A
FEB0 CD 7D 1E 78 18 0F FD 36 77 00 FD CB 76 4E 28 05
FEC0 CD 7D 1E 3E 0A C3 7D 1E 21 95 00 D6 A4 47 CD 30
FED0 03 23 E6 80 28 F8 05 20 F5 3E 20 CD 7D 1E FE 34
FEE0 77 FD 34 77 3A 81 5C FD BE 77 30 14 3E 0D CD 7D
F1FC 1E FD 36 77 00 FD CB 76 4E 28 05 3E 0A CD 7D 1E
FF00 CD 30 03 23 CB 17 38 07 CB 3F CD 7D 1E 18 D2 CB
FF10 3F CD 7D 1E FD 34 77 3E 20 C3 7D 1E 00 00 00

```

COMPONENTES

- IC1 4002 CMOS
- IC2 4040 CMOS
- IC3 74LS32
- IC4 1M6402
- IC5 74LS04
- IC6 74LS09
- IC7 MCI489
- C1 0.1 µF
- R1 10 M ω
- R2 to R7 10 K ω
- R8 1 K ω
- R9, R11 * 10 K ω
- R10 * 15 K ω
- SW1 conmutador DIL de 1 circuito y 6 posiciones.
- SW2 a SW6 conmutadores DIL.
- SW7 * pulsador miniatura normalmente abierto.
- Cristal de cuarzo de 2,4576 Mhz.
- SKT1 conector de 28 pistas y doble cara.
- STK2 conector "D" de 25 pistas.
- * Sólo se necesita para entrada de datos.

MONTAJES

el mejor lugar para guardar todo este código, pues estará instantáneamente disponible, se puede colocar en la parte superior a la 3C40 en la EPROM. Cuando se cargue la rutina de inicialización, esta correrá mediante PRINT USER

65024 si no se necesita la opción de alimentación de nueva línea, o PRINT USER 65028 si lo es. No se necesitará nada más y todo lo que pueda ser escrito normalmente por la Impresora ZX irá normalmente al port RS 232.

No aparece ningún *software* para una rutina de entrada. Sin embargo, la instrucción IN 65407 puede ser utilizada para leer los códigos de caracteres de entrada individualmente.

Fig. 5 Rutina de inicialización en Mnemonico.

Dir	Datos en Hexa	Op	Operandos
Rutina de inicialización			
FE00	3E00	LD	A,00h
FE02	1802	JR	+02h
FE04	3E02	LD	A,02h
FE06	32B05C	LD	(5CB0h), A
FE09	C116FE	LD	BC, Salida R (Output R)
FE0C	ED43C55C	LD	(5CC5h), BC
FE10	3E48	LD	A, Longitud de la línea
FE12	32815C	LD	(5C81h), A
FE15	C9	RET	
Rutina de salida			
FE16	FDCB7656	BIT	0, (1Y + 76 h)

Fig. 6 Selección del estándar de transmisión.

Longitud del carácter en bits.	5	6	7	8
SW2	ON	OFF	ON	OFF
SW3	ON	ON	OFF	OFF
Paridad.	Par.	Impar.	Par.	Impar.
SW4	ON	OFF	ON	OFF
SW5	ON	ON	OFF	OFF
N.º de bits de parada.	1	2		
SW6	ON	OFF		



SUSCRIBASE POR TELEFONO

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

(91) 250 15 93

7 días por semana, 24 horas a su servicio

SUSCRIBASE A

**CIRCUITO
IMPRESO**

EL BASIC DE SINCLAIR

Los manuales que acompañan a los ordenadores ZX 81 y ZX Spectrum suelen recibir de los programadores novatos críticas sólo comparables a las que usuarios más experimentados formulan a propósito de la impresora térmica ZX Print. Lo que ha dado lugar, naturalmente, a la aparición de una nutrida literatura destinada a ilustrar al usuario sobre el modo de sacar mejor partido de la máquina que acaba de adquirir. En cualquier caso, corresponde decir que las posibilidades que ofrece el BASIC específico de los ordenadores Sinclair merecen una exploración sistemática, que el manual apenas deja abierta del usuario. ¿Cuáles son esas características del lenguaje propio de la marca británica que han hecho de estos pequeños ordenadores herramientas difícilmente mejorables para el aprendizaje de la programación? Vamos por partes.

El BASIC fue desarrollado en el Dartmouth College (Estados Unidos) a mediados de los años 60. Su principal finalidad consistía en que los estudiantes pudiesen programar ordenadores sin tener que preocuparse de los aspectos formales, tan importantes en los lenguajes de programación existentes a esa fecha. Pero, lo que es más importante, el lenguaje podía ser interactivo, es decir, que se podía conseguir una respuesta casi inmediata del ordenador, a medida que se escribían los programas. Ya empezaba a no ser necesario esperar a que un programa fuese compilado para poder correrlo en el ordenador. Las instrucciones eran traducidas (interpretadas) a medida que el programador las introducía. Actualmente se pueden encontrar numerosas versiones del BASIC, tanto interpretado como compilado.

Como lenguaje de programación de alto nivel, que en un principio fue desarrollado para fines exclusivamente educativos, el BASIC ha alcanzado elevadas cotas de universalidad, que nunca fueron siquiera soñadas por Kemeny y Kurtz, sus autores.

Hoy en día prácticamente todos los ordenadores personales, salvo pocas excepciones, llevan incorporada alguna versión de aquel BASIC, formando parte del *firmware* de la máquina, es decir, en

una memoria ROM grabada en fábrica. Parte de esta ROM contiene el intérprete del BASIC, que va traduciendo todo lo que se introduce desde el teclado a una secuencia de instrucciones más sencillas que puede entender el microprocesador. Con este método es mucho más fácil detectar los errores de sintaxis a medida que se van produciendo, de tal manera que también los corrige simultáneamente. La ventaja, entonces, consiste en que los programas se pueden escribir de una manera algo más rápida. Sin embargo, el tributo que se paga lo es en términos de velocidad cuando se hacen correr los programas. El ordenador lee una vez más, siempre como si se tratara de la primera, el programa línea por línea, siendo traducido de igual manera; esto hace que el lenguaje interpretado sea más lento que el compilado, que se traduce una sola vez en forma de instrucciones en código máquina almacenadas normalmente en algún tipo de soporte magnético (cinta, *diskette*, etcétera) y el ordenador las va asimilando directamente.

En realidad, el BASIC tiene el aspecto de ser una versión simplificada del FROTRAN, coin menos reglas minuciosas a la hora de escribir programas. Ambos son lenguajes altamente iterativos, que tienen la posibilidad de repetir un número indeterminado de veces —fijado por el programador— un conjunto de instrucciones, sin necesidad de escribirlas más que una vez. En el FORTRAN se utiliza el DO y en el BASIC el FOR... NEXT. Pero hay muchas más similitudes entre ambos.

Antes de abordar específicamente el BASIC de Sinclair, conviene que nos derengamos un poco más en algunos antecedentes de este lenguaje. Durante sus primeros años, que hemos descrito someramente, no apareció ninguna versión del BASIC específicamente representativa; muchos fabricantes disponían de una propia para sus miniordenadores y grandes ordenadores, pero que raras veces eran parecidas en su totalidad. A ninguno le preocupaba demasiado la normalización del lenguaje, que servía para vender ordenadores pero se revelaba inútil a la hora de transportar los lenguajes escritos en una

máquina para su funcionamiento en otra distinta. La mayoría de las veces el lenguaje se adecuaba totalmente a las características del *hardware*.

Fue con la aparición de los microordenadores, más tarde conocidos también como ordenadores personales, cuando una firma llamada Microsoft —hoy mundialmente famosa— desarrolló un BASIC bastante neutro, disponible para los primeros equipos que aparecieron en el mercado. Esta versión gustó a los fabricantes que se aproximaban al naciente mercado. El lenguaje, aunque dispone de la potencia suficiente para escribir programas que funcionan de manera más que decente, es lo suficientemente compacto como para no absorber una excesiva cantidad de direcciones de memoria accesibles por el ordenador, condicionante clave en máquinas con arquitecturas de 8 bits. De esta forma, Microsoft estableció lo que puede considerarse como estándar para un BASIC mínimo que funcione. Posteriormente, la misma empresa, en coordinación con los fabricantes de ordenadores, ha ido desarrollando versiones diferentes a la medida de cada *hardware*, haciendo especial hincapié en características tales como las capacidades gráficas, de color o incluso las musicales, pero siempre respetando el espíritu de aquel antiguo estándar. De tal forma, se conservan los aciertos de aquellas primeras versiones... y también se perpetúan sus defectos.

Por el contrario, en ZX BASIC ha ido sufriendo sucesivos cambios destinados a su mejora. En algunos casos se le ha provisto de comandos presentes en el BASIC de Microsoft, que han demostrado ser particularmente útiles. La evolución del BASIC de Sinclair también ha venido acompañada de nuevas versiones de máquinas. Esto se debe en gran parte a la forma que tiene la empresa británica de diseñar sus microordenadores. Sinclair siempre ha querido conseguir máquinas compactas y de bajo coste. Pare ello es fundamental el ahorro en el número de componentes internos. Para conseguir ordenadores más potentes necesitaba versiones del BASIC también más potentes, es decir, más memoria ROM y más componen-

SOFTWARE

tes electrónicos. Había que llegar a un compromiso. En su primer microordenador, el legendario ZX80 —que no llegó a venderse en España— se utilizaron gran cantidad de componentes, entre los que se incluían 22 circuitos integrados. Su BASIC estaba contenido en parte de una ROM de 4 Kbytes. El siguiente modelado lanzado al mercado, el ZX81, aun disponiendo de una versión más potente con un BASIC contenido en la memoria de 8 Kbytes, lleva sólo 4 circuitos integrados. Lo que es más, el ZX80 podía ser convertido en un ZX81 con sólo cambiarle la ROM y el diagrama externo del teclado mediante una carátula adhesiva.

El ZX Spectrum, último modelo —por ahora— de la saga de Sinclair, utiliza doble cantidad de memoria ROM, 16 Kbytes, y cada una de las teclas alfabéticas puede ejecutar hasta seis operaciones diferentes, y las numéricas hasta ocho.

El BASIC, al fin y al cabo, fue desarrollado originalmente para que fuese fácil de aprender por alumnos angloparlantes de manera interactiva. Por ello se le dotó de una estructura gramatical bastante semejante a la del inglés coloquial. Algunas otras palabras están formadas por abreviaturas de lo que se pretende que el ordenador realice, pero siempre valiéndose de palabras inglesas.

Esto viene a cuento porque una de las principales diferencias entre el BASIC de Sinclair y otras versiones es la posibilidad de escribir las palabras clave presionando sólo una tecla, a la que corresponde tal palabra; entonces, el significado concreto de la tecla es buscando en la ROM y escrito en la pantalla con todas sus letras, aunque en la memoria RAM, donde se guarda el programa que se está escribiendo, se almacena como la pulsación de una sola tecla. Por el contrario, en otras versiones del BASIC, las palabras (y todo) deben escribirse letra por letra, y una vez escrita la línea completa se introduce (mediante la tecla Enter o Return) y el intérprete va localizando y traduciendo las palabras una por una, encargándose el editor de los problemas de sintaxis. Como vemos, el método empleado por el ZX BASIC es más rápido y menos dado a los errores.

A algunas de las palabras descritas en el párrafo anterior, Sinclair las llama palabras clave. Cuando se comienza a escribir una nueva línea de programa, en la pantalla aparece una K dentro del cursor; esto quiere decir que el ordena-

dor espera que se le introduzca una palabra clave, y que no admitirá ninguna otra. Esto, indudablemente, representa una ventaja para el programador novato. Cada palabra clave debe ser seguida por una forma particular de gramática, que es esperada por el ordenador. Por ejemplo, el ZX81 dispone de un juego de 26 de estas palabras clave del BASIC.

Cuando apareció el FORTRAN, lenguaje orientado a las aplicaciones científicas y técnicas, se habló de expresiones aritméticas, que es lo que se esperaba que el ordeador ayudase a manejar. Tales expresiones aritméticas deberían ser posteriormente evaluadas por el ordenador para producir una respuesta en forma de un valor concreto, tal como se haría con un lápiz y un papel. Para ello habrá que proceder a analizar y evaluar la expresión y manejar datos dentro de ella. Por tanto, expresiones y datos son componentes fundamentales para poder realizar tanta más potencia de cálculo cuanto mayores sean las variedades de datos y expresiones que puede manejar y compatibilizar el lenguaje.

El ZX BASIC puede aceptar varios tipos de expresiones, a saber: matemáticas ya aritméticas, lógicas, relaciones y cadenas.

Las expresiones matemáticas y aritméticas son las que implican la utilización de determinadas reglas, que van desde la suma, resta, producto y división hasta los logaritmos, relaciones trigonométricas o la raíz cuadrada.

Las expresiones lógicas implican la utilización de los llamados operadores lógicos, tales como OR, AND.

Las expresiones relacionales establecen relaciones entre elementos e implican la utilización de @: @ =, =, etc.

Las expresiones en cadena son conjuntos de elementos alfanuméricos o caracteres. Las cadenas se pueden representar por un nombre al que se le añade como último elemento el carácter \$ o bien por una expresión escrita entre comillas.

Por otro lado, es de fundamental importancia la forma en la cual pueden ser mezcladas las diferentes expresiones entre sí.

El tipo de datos utilizados por el BASIC de Sinclair también es de importancia capital. En el ZX80, las expresiones aritméticas sólo podían utilizar números enteros como datos. Pero en el ZX81 ya se introdujeron también los reales.

En el BASIC empleado por los mode-

los ZX, la forma de manejar cadenas ha evolucionado de una manera importante. Con el ZX81, aparecían las instrucciones LEN, VAL y STR\$. La primera proporcionaba como respuesta un número que era la longitud total de la cadena evaluada, la segunda trataba el contenido de la cadena como si de una expresión aritmética se tratase, hallando el resultado numérico de la misma. La última convertía un n La última convertía un número en una cadena.

; El ZX BASIC trae una ampliación de las posibilidades de manejo de cadenas. Una cadena seguida de la expresión (número TO número) proporciona los mismos resultados que las instrucciones LEFT\$, RIGHT\$, MID\$ en otros BASICs, que respectivamente fragmentan la cadena desde el lado izquierdo, derecho o en su parte central, de acuerdo con otros valores numéricos que acompañan a la expresión. Indudablemente, el método adoptado por Sinclair es más intuitivo y compacto.

En el ZX BASIC, la utilización de las funciones LEN, VAL y STR\$ permite interesantes manipulaciones. LEN permite hacer operaciones con las mismas, tales como colocarlas en orden creciente o decreciente de longitud, por ejemplo.

STR\$ permite que puedan mezclarse las expresiones de cadena con expresiones aritméticas, que convierte un valor numérico en una cadena numérica y VAL que hace lo contrario, mezclando expresiones numéricas con expresiones de cadena, convirtiendo una cadena que contenga números en un número.

Volviendo a las facilidades que proporciona el ZX BASIC, las indicaciones del cursor son útiles a la hora de saber qué es lo que espera el ordenador que se introduzca a continuación. En el ZX81 los cuatro modos indicados por la letra en inverso contenida dentro del cursor son L (letra), G (gráfico), K (clave), F (función).

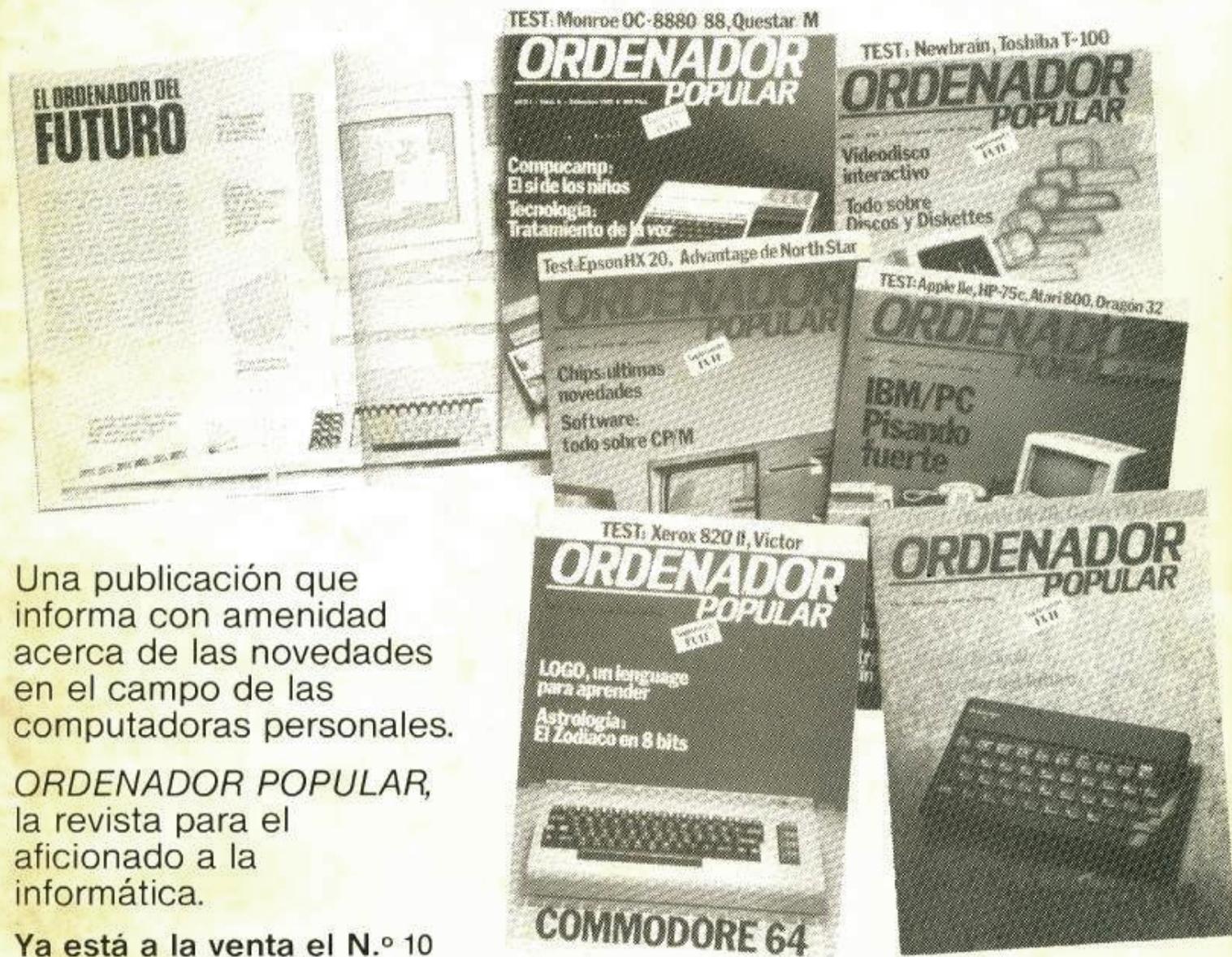
En el ZX Spectrum aparece un nuevo cursor, el E, que tiene como fin indicar la posibilidad de que otras leyendas, escritas en la parte superior de las teclas, puedan ser introducidas, aumentando así la versatilidad del teclado. A este nuevo cursor se le hace aparecer presionando las teclas CAPS SHIFT (mayúsculas); el Spectrum viene con mayúsculas y minúsculas) y Symbol SHIFT (viene a ser el equivalente de la tecla SHIFT del ZX81).

(Continuará en el próximo número)

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 el N.º 10

Cómprela en su kiosco habitual o solicítela a:

**ORDENADOR
POPULAR**

Jerez, 3
Tel. 457 45 66
MADRID-16

sinclair store

Todo para **sinclair**

Te esperamos en *sinclair store*, tu tienda informática, donde encontrarás la mas extensa variedad en hardware y software para **sinclair**.



SINCLAIR STORE · Bravo Murillo, 2 - Tél. - 4 46 62 31 - Madrid