tu Micro

programas
lenguajes
ordenadores

AÑO 1/N.º 6
OCTUBRE 1984
250 Ptas.



EL SPECTRUM EN CONCIERTO

DRAGON 64

PASCAL...
¿EL BASIC DE LOS 80?

PROGRAMAS PARA SPECTRUM, SHARP MZ-700, COMMODORE 64, DRAGON, SPECTRAVIDEO

SPECTRAVIDEO

El sistema con el que usted crecerá



Los ordenadores personales de Spectravídeo, gracias a su perfecto diseño y gran facilidad de manejo, se pondrán enteramente en sus manos desde el momento que los desembale. Y, aún más, son suficientemente capaces y ampliables como para cubrir todas las necesidades durante mucho tiempo



SV-318 SV-328

CUANDO SE TRATA DE ORDENADORES PERSONALES SPECTRAVIDEO ESTA MAS ALLA DE CUALQUIER COMPARACION

BASIC extendido de Microsoft, integrado, convierte a Spectravídeo en los primeros ordenadores verdaderamente programables y asequibles del mercado.

Extraordinaria memoria. 32 Kbytes de ROM, ampliables a 96 Kbytes y, 32 u 80 Kbytes de RAM, ampliables a 256 Kbytes.

Expansión diagonal. Un sistema de 14 periféricos plenamente soportado. Incluyendo el adaptador de juegos Colecovisión. Unidad de expansión de 7 slots, controlador de disco flexible, cassette, interface para cartucho, etc.

Gráficos avanzados. El sistema SV ofrece 16 colores en alta resolución y, más importante aún, 32 sprites programables que permiten unas impresionantes posibilidades de control de animación en pantalla.

Otras muchas características atractivas. Tales como un microprocesador Z80A con un rápido (3,6) reloj interno, slot para la conexión de cartuchos, diez teclas de función programables por el usuario, tres canales de sonido (8 octavas por canal), perfil bajo y estilizado diseño.

TABLA DE COMPARACION DE FUNCIONES CON DISTINTAS MARCAS

	SPECTRAVIDEO SV-328	SPECTRAVIDEO SV-318	APPLE IIE	ATARI 800	COMMODORE 64	BBC MODEL B	DRAGON 32	EL MAS VENDIDO HASTA AHORA
CAPACIDAD DEL ORDENADOR								
ROM INCORPORADA	32K	32K	16K	10K	20K	16K	16K	16K
AMPLIABLE A	96K	96K	?	42K	?	64K	?	?
MBASIC INCORPORADO	SI	SI	SI	ADIC.	NO	NO	SI	NO
RAM INCORPORADA	80K*	32K* * *	64K	48K	64K	32K	32K	16K
AMPLIABLE A	256K* *	256K* *	64K	NO	N/A	32K	64K	48K
CARACTERISTICAS DEL TECLADO								
NUMERO DE TECLAS	87	71	63	61	66	73	53	40
TECLAS DEFINIBLES POR EL USUARIO	10	10	?	4	8	10	?	?
PROCESO DE TEXTOS	SI	SI	NO	NO	NO	NO	NO	NO
GRAFICOS (DESDE TECLADO)	SI	SI	NO	SI	SI	SI	SI	
TECLAS MAYUSCULAS Y MINUSCULAS	SI	SI	SI	SI	SI	SI	SI	SI
CARACTERISTICAS DE JUEGO Y SONIDO								
RANURAS DE CARTUCHOS SEPARADAS	SI	SI	NO	SI	NO	NO	SI	NO
JOYSTICK INCORPORADO	NO	SI	NO	NO	NO	NO	NO	NO
COLORES	16	16	15	128	16	16	9	8
RESOLUCION (PUXELK)	256×192	256×192	280×180	320×192	320×200	256×640	256×192	256×192
SPRITES	32	32	N/A	4	8	?	16	?
CANALES DE SONIDO	3	3	1	4	3	1	3	?
OCTAVAS POR CANAL	8	8	4	4	9	3	5	3
ENVOLTURA A.D.S.R.	SI	SI	NO	NO	SI	SI	NO	NO
CARACTERISTICAS DE LOS PERIFERICOS								
CASSETTE	2 CANALES	2 CANALES	1 CANAL	2 CANALES	1 CANAL	2 CANALES	?	?
AUDIO 110	SI	SI	NO	SI	NO	?	?	?
MC INCORPORADO	SI	SI	NO	NO	NO	NO	NO	NO
CAPACIDAD DE LOS DISCOS	256K	256K	143K	92K	170K	100K	?	?
LINEA SENCILLA	SI	SI	NO	NO	NO	NO	NO	NO
COMPATIBILIDAD CON CP/M (programas standard de 80 columnas)								
CP/M 2.2	SI	SI	NO****	NO	NO****	SI	NO	NO
CP/M 3.0	SI	SI	NO	NO	NO	NO	NO	NO
MSX	SI	SI	NO	NO	NO	NO	NO	NO

Microsoft es una marca registrada de Microsoft Corporation
CP/M, es una marca registrada de Digital Research, Inc.

. 64 K disponibles con CP/M 2.2, más 16 K de soporte gráfico
.. 240 K disponibles con CP/M 3.0, más 16 K de soporte gráfico
... 16 K disponibles, más 16 K de soporte gráfico
.... Apple II acepta con una tarjeta de modificación de 40 u 80 columnas CP/M
..... Commodore 64 acepta 40 columnas CP/M
? Dato no confirmado

Las especificaciones están sujetas a cambios sin previo aviso.

DISTRIBUIDOR OFICIAL PARA ESPAÑA

DYNADATA INFORMATICA

● Garantizamos el mantenimiento de nuestros equipos. ● Póliza de seguro, a todo riesgo. ● Programas especiales a necesidades del cliente.

CENTRAL: Sor Angela de la Cruz, 24. Madrid-20
Teléfonos: (91) 279 21 85 - 270 01 93 - 279 28 01

DELEGACION: Aribau, 61, entlo. Barcelona-11
Teléfono: (93) 254 65 48

tuMicro
Año 1, núm. 6
octubre 1984

SUMARIO

lenguajes

Estandarización del BASIC (5) 17
Los bucles o estructuras cíclicas.

PASCAL 62
La arquitectura de un lenguaje estructurado.

programación

El SPECTRUM en concierto 33
La musicalidad de un pentagrama trasladada a un ordenador personal.

equipos

Dragón 64 46
Nunca un heredero siguió tan fielmente los pasos de su antecesor...

programas

Gus 10
SHARP MZ-700

Finos 14
COMMODORE 64

La muralla 21
DRAGON 32/64

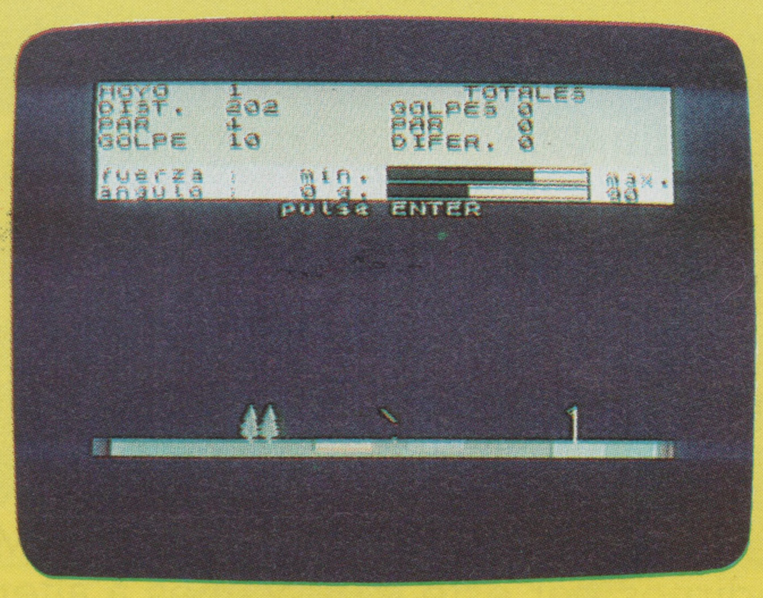
Golf 25
ZX-SPECTRUM

Ruleta rusa 40
SPECTRAVIDEO
SV-318/328

Display 54
ZX-SPECTRUM

Definidor de «sprites» 68
SPECTRAVIDEO
SV-318/328





De DRAGON a DRAGON... del 32 al 64.

Golf: el vencedor de nuestro primer concurso «El programa del lector».



PASCAL... ¿El BASIC de los ochenta?

... y además

TU MICRO noticias 5, 6, 7, 9
Trucos y recetas 13
Bazar Soft 31
REM 39
Concurso «El programa del lector» 45
Glosario micro-informático 59
Tarjetas de servicios 59
TU MICRO diversión 61
Anuncios breves 72
Directorio 73
Relación de anunciantes 73

Con este son ya seis los ejemplares de TU MICRO que han visto la luz. Tiempo aún escaso para que una revista alcance su madurez, pero suficiente para poder evaluar la respuesta que ésta tiene entre sus destinatarios.

Un breve análisis de las opiniones recibidas a través de las tarjetas de encuesta, cartas y llamadas telefónicas, nos permite extraer conclusiones satisfactorias y enriquecedoras. Para empezar, TU MICRO no solo ha conseguido llegar a un elevado número de lectores esporádicos de la publicación, sino que, felizmente, se ha rodeado de un sólido grupo de lectores entusiastas y casi incondicionales. Lectores cuya participación en la revista es sustantiva con sus opiniones, sugerencias e incluso programas.

Una primera opinión generalizada es que TU MICRO debe crecer... ¡Y a ello vamos! En este, su sexto número, TU MICRO empieza a incrementar su peso, con ocho páginas adicionales que van a permitir ampliar el contenido. Desde luego que, en este primer paso, el crecimiento resulta aún liviano, pero, al fin y al cabo, se trata del primer peldaño y por ahí hay que empezar.

En el número anterior, se hizo público el fallo del primer concurso TU MICRO «El programa del lector». Tal como ya anunciamos, el nivel de calidad de los programas era más que notable, y éste es el momento de empezar a comprobarlo. Uno de los programas que aparecen publicados en este número es, precisamente, el vencedor del concurso: «Golf»; un programa que pone de manifiesto hasta qué punto su autor, Alfredo Casaña Bayona, ha logrado «domesticar» al ZX-SPECTRUM.

Antes de despedirnos por este mes, vamos a alternar los papeles, y será TU MICRO quien les haga una sugerencia: ¿qué tal si se arman de paciencia, toman asiento ante su ordenador, y empiezan a elaborar su programa para participar en la segunda convocatoria del concurso «El programa del lector»?...

EDITORIAL

AÑO 1 N.º 6 OCTUBRE 1984

Director:
Antonio M. Ferrer Abelló

Redactor-jefe:
Francisco Lara

Redacción:
Fernando López
Juan Manuel López
Inmaculada de la Torre

Colaboradores:
David Bernabé
Antonio Carvajal
Javier Lázaro
Angel Martínez
Carlos Ruíz
Fernando Seror

Producción gráfica:
Vicente Robles

Ilustración:
José Ochoa

Maquetación:
Carlos González Amezúa

Fotografía:
Equipo Gálata

Publicidad:
Lola González
Fernando Castellanos
Teléf. 457 69 23

Redacción, administración, publicidad y suscriptores:
Avda. Alfonso XIII, 141. 28016 MADRID
Teléf. 250 58 20. Télex 49371 ELOC E

Dirección para correspondencia:
Aptdo. de Correos 61.294.
28016 MADRID

Administración:
María Antonia Buitrago

Distribución:
Santiago Ferrer

Suscripciones:
María Teresa Conde

TU MICRO es una publicación mensual de **Ediciones INGELEK, S. A.**

Presidente:
Ernesto Medina
Copyright 1984
Ediciones INGELEK, S. A.
(MADRID-E)

Reservados todos los derechos. Prohibida la reproducción total o parcial, aún citando su procedencia, de textos, dibujos, fotografías y programas sin autorización escrita de Ediciones INGELEK, S. A. Los programas publicados en TU MICRO sólo pueden ser utilizados para fines privados
Queda prohibido su empleo para fines comerciales.

Imprenta:
Héroes, S. A. MADRID

Distribución:
COEDIS, C/ Valencia, 245. BARCELONA

Precios para España:
Ejemplar: 250 ptas.
Suscripcion anual: 2.500 ptas.
(La suscripción anual incluye 10 números simples más uno doble correspondiente a los meses de julio/agosto.)

Depósito legal:
M. 12.533-1984
ISSN: 0212-9027
Impreso en España

LA PRODUCCION AUTOMATIZADA DEL HP-150

La clave de la producción para el mercado europeo del nuevo ordenador personal de Hewlett-Packard, el HP-150 con pantalla sensible al tacto, es un innovador programa de automatización al estilo japonés.

Las instalaciones que HP tiene en Grenoble asumen la importante responsabilidad de suministrar a cada país una versión en su propio idioma. Esto supone preparar en diez idiomas diferentes el teclado y los mensajes que aparecen en la pantalla sensible al tacto, además de 12 programas de aplicación general. La automatización es clave para la consecución de este objetivo. En la actualidad, el 25 por ciento del trabajo se realiza automáticamente, mientras que, en 1979, esta tasa era de un 1 por ciento. Las previsiones contemplan que esta proporción siga aumentando constantemente.

Ya hay bastantes fases de la cadena en las que las máquinas se encargan del trabajo. Por ejemplo, la mayor parte de los diminutos componentes que forman el circuito de la placa con el «sistema nervioso» del ordenador personal, se clasifican, sitúan e insertan automáticamente. Y, con el tiempo, este proceso se automatizará aún más mediante robots.

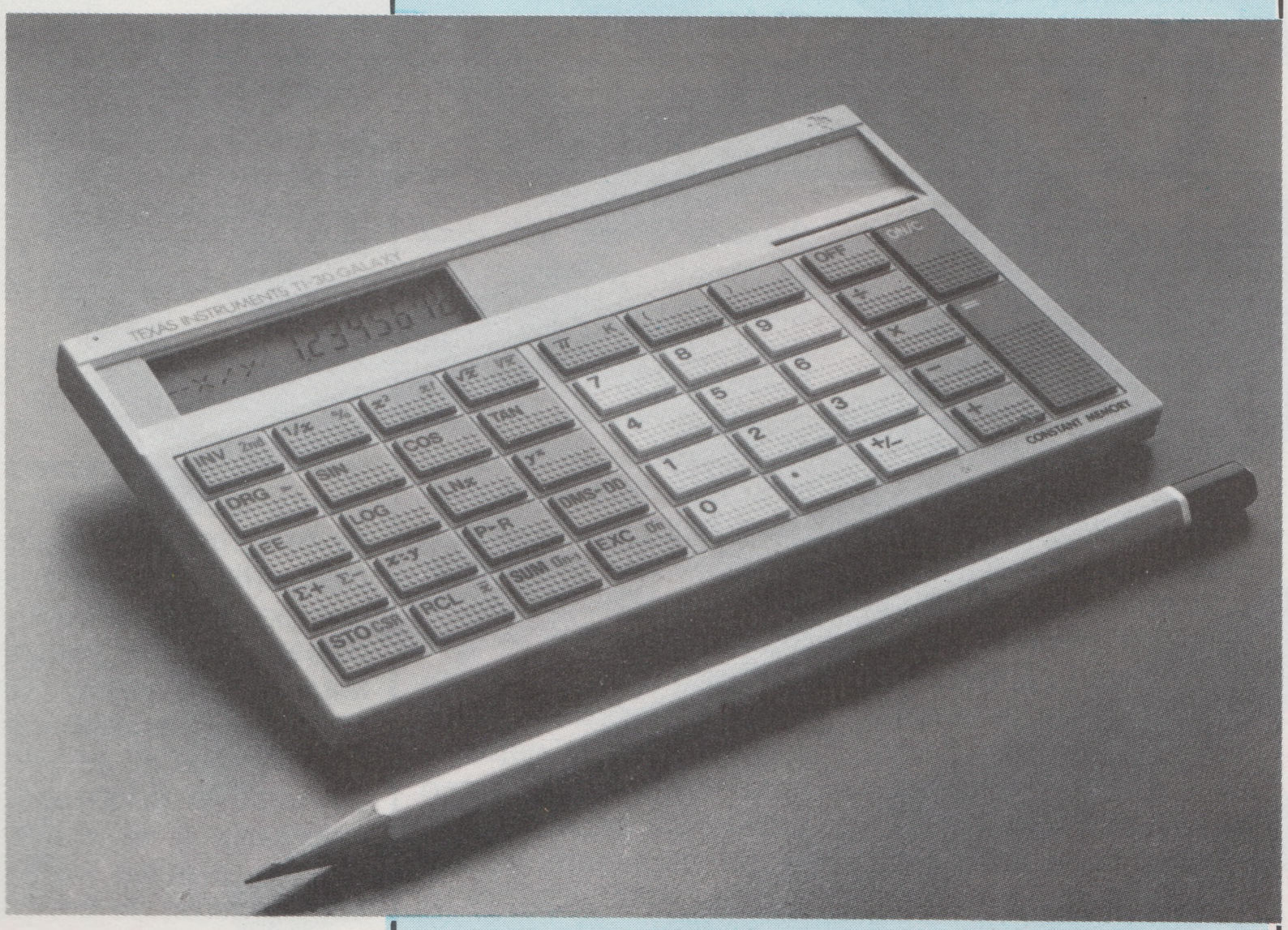
Hasta ahora, los programas «localizados» (adaptados a cada uno de los idiomas) se solían grabar y producir manualmente. Hoy, sin embargo, también esta parte de la producción se ha automatizado con el uso de robots.

DEL ORDENADOR... A LA CALCULADORA

Texas Instruments ha anunciado el lanzamiento en España de una nueva calculadora científica, la TI-30 Galaxy, especialmente diseñada para estudiantes de BUP. La TI-30 Galaxy es un concepto íntegramente desarrollado en Europa, tras una exhaustiva investigación de las tendencias pedagógicas realizadas entre profesores y alumnos de diversos países, España entre ellos.

Esta nueva calculadora realiza 66 funciones, dispuestas en un cómodo teclado codificado por colores, que cubren todas las operaciones científicas y estadísticas, además de conversiones polares-rectangulares. La TI-30 Galaxy ofrece 11 cifras de precisión interna, e incorpora el popular Sistema Operativo de Algebra (AOS), con memoria constante, y 15 niveles de paréntesis, además de 4 indicadores que muestran al estudiante las operaciones pendientes de realizar. Este último dispositivo, especialmente desarrollado por Texas Instruments para la TI-30 Galaxy, hace del sistema AOS un elemento único.

De fácil manejo y elegante diseño, la TI-30 Galaxy viene equipada con un estuche rígido, y es una opción igualmente válida como instrumento de sobremesa o portátil. Funciona con una sola pila que suministra varios años de servicio, y su precio es de 4.990 pesetas. Al igual que todas las calculadoras TI tiene una garantía de dos años.



EL SPECTRUM EN CHINA

La microinformática europea, de la mano del ZX-SPECTRUM, ha abierto una vía de entrada en el sector educativo de la República Popular China.

El departamento de ordenadores y automatización de la Universidad Técnica del Noroeste de China, acaba de dotar a sus investigadores con 600 ordenadores SPECTRUM como suministro inicial. Estos equipos se utilizarán como herramientas para adquirir unos conocimientos prácticos de programación.

La propia Sinclair Research, suministrará también un paquete de programas para SPECTRUM en lengua local.

EL NUEVO PORTATIL SORD IS-11



La gama de microordenadores portátiles presente en el mercado nacional se amplia con el recién llegado SORD IS-11. Un portátil presentado en Europa en la pasada feria de Hannover y que ahora llega a nuestro país de la mano de la firma distribuidora DATA-TRACK.

El IS-11 es el primero de una serie de nuevos ordenadores de SORD creados en base a una nueva idea, el Software Integrado (todos los programas que el usuario de un IS-11 necesita, están incluídos dentro del equipo en memoria ROM).

Destinado a un mercado profesional, el SORD IS-11 es un ordenador personal portátil y compacto que incluye en las dimensiones de un libro normal todo un ordenador, con pantalla de cristal líquido y micro-cassette incluido. Pensando para trabajar fuera de la oficina va alimentado por baterías recargables de Ni-Cd. La CPU del IS-11 es el popular Z80A funcionando a 3,4 Mhz. La memoria utilizada es CMOS, de bajo consumo y la capacidad del equipo es de 64K de ROM y 32 Kbytes de RAM, ampliables hasta un total de 128 Kbytes de ROM y 32 Kbytes de RAM, ampliables hasta un total de 128 Kbytes de ROM y 128 Kbytes de RAM. Los interfaces más importantes que incluye el IS-11, directamente utilizables por el usuario, son: Interface paralelo Centronics, lector de barras y 2 accesos serie RS232, todos ellos incluidos en equipo.

El software incluido en el IS-11 es el nuevo SORD-IS. Se trata de un paquete de programas integrado compuesto por Tratamiento de textos I-WP, hona de planificación/base de datos I-PIPS, gráficos y comunicaciones I-COMM, Calculadora I-CALC. También incluye aplicaciones del PIPS como: Planificación de viajes, agenda, listín telefónico, calendario, etc.

El IS-11 incluye como standart los programas I-COMM de comunicaciones para la conexión a redes como Infonet, GE Mark III, Dow Jones, Lockheed, etc... permitiéndole el acceso a grandes bases de datos.

Como Hardware opcional dispone de un Modem acústico (C-240A), una impresora centronics alimentada por baterías (PT-11), teclado numérico auxiliar (KB-11), lector de código de barras (BR-100), expansión de memoria (EM-11), controlador de pantalla (ID-11), discos de 3" con 1Mb. de capacidad (FD-11) y alimentado para batería de coche.

El software disponible como opción es relativamente amplio: lenguaje BASIC, I-TRANS comunicaciones con PC-IBM y compatibles; análisis de bolsa, programas de manejo del lector de barras, estadística y finanzas, etc.

Un dato significativo es que el IS-11 posee la CPU de tipo Z80A, con lo que resulta extremadamente fácil la creación de programas de aplicación utilizando cualquier equipo que permita el desarrollo de programas en assembler para este microprocesador. DATA-TRACK dispone de toda la información necesaria y el soporte técnico preciso para la implantación de programas de utilidades en memoria ROM dentro del IS-11.

SANYO LANZA EN EUROPA SU ORDENADOR PERSONAL MSX

El grupo Sanyo ha iniciado la comercialización en Europa de su ordenador personal de 8 bits, ajustado al estandar MSX. A pesar de la mínima presencia de Sanyo en el mercado informático español, esta firma japonesa tiene en el mercado mundial ordenadores de reconocido prestigio. Según un informe Auerbach, en los diez primeros puestos del «ranking» de ventas de microordenadores en Estados Unidos, figuran cuatro modelos de Sanyo. Un dato significativo que hay que sumar al gran éxito en el mercado USA del ordenador personal MBC 555 de la misma firma. Por lo que respecta al nuevo ordenador personal MSX —que pronto esperamos ver en nuestro país—, el grupo Sanyo Japón exportará inicialmente 10.000 equipos al mercado inglés y 5.000 al francés.

EL SUPERORDENADOR PERSONAL.

MECOMATIC SHARP MZ-700

Ideal y brillante colaborador para la **informática familiar, el profesional, el técnico, la oficina, el estudiante, el radioaficionado, el profesor,...**

La consola con un sólido teclado, genera 512 caracteres distintos, incluye un cassette para leer y grabar programas y datos. La impresora trazadora de gráficos (opcional), es una pequeña joya que imprime y dibuja en 4 colores en 64 tamaños diferentes y hasta 80 caracteres por línea.

Conéctelo a un TV color o B/N y consiga excitantes posibilidades y versatilidad **confeccionando y listando programas, etiquetas y operaciones administrativas y de gestión, diseño gráfico, cuadros estadísticos, videojuegos, música, hobbies,...**

SHARP Mz 721
68 Kbytes RAM, Basic y lenguaje máquina, cassette, cables conexión a TV, manuales en castellano con programas didácticos para principiantes y esquemas técnicos para expertos.

Ptas. . . . . 94.000,—

SHARP Mz 731
Idem. que el Mz 721 incluyendo la impresora-trazadora gráfica de 4 colores y accesorios.

Ptas. . . . . 129.000,—



Hay disponibles varios lenguajes: **BASIC, PASCAL, FORTH, ASSEMBLER, MAQUINA, y otros en preparación.**
Una extensa biblioteca de programas le permitirá desarrollar su creatividad, personalizándolos o diseñando otros nuevos.

Dispone de salidas para conexión de periféricos (disquettes, joys stick, impresora externa, etc.).

MECANIZACION DE OFICINAS, S. A.

BARCELONA-36: Diagonal, 431-bis. Tel. 200 19 22 — MADRID-3: Santa Engracia, 104. Tel. 441 32 11

DIGITAL PRESENTA EL MICRO PDP/11/73



Digital Equipment ha presentado un nuevo microordenador con una capacidad aproximada a la de los miniordenadores de rango medio. El nuevo sistema Micro PDP-11/73 es el segundo miembro de la familia Micro PDP-11 de Digital e incorpora el juego de «chips» J-11 de Digital.

Disponible tanto en rack como en versión autocontenida, está orientado a usuarios finales y a Fabricantes de Equipos Originales (OEMs) que buscan un microordenador económico de alto rendimiento con características multiusuario. Las áreas de aplicación son diseño, fabricación, automatización de oficinas, comunicaciones, medicina, educación, laboratorio y pequeñas empresas.

El nuevo sistema informático es totalmente compatible con la familia PDP-11 de Digital. El Micro PDP-11/73 admite hasta diez sistemas operativos. Este despliegue de sistemas permite al usuario pasar aplicaciones escritas en lenguajes de alto nivel desde otros PDP-11 sin tener que volver a escribirlos.

Con un microcomputador standard PDP-11, Micro PDP-11/73 acepta productos periféricos Q-bus, incluyendo discos, cintas e interfaces de comunicación de datos.

Existen sistemas preconfigurados, incluyendo 31 Mbyte en disco Winchester y unidades de disco flexible duales de 5' 25'' ó bien 60 Mbytes en cinta tipo cartucho.

Para OEM's existe un sistema incluyendo 26 Mbytes en Winchester junto con 26 Mbytes en disco removible.

El nuevo ordenador es complementario del Micro PDP-11. Según un portavoz de la compañía, el Micro PDP-11/73 proporciona otro nivel de crecimiento para los usuarios cuyas necesidades informáticas aumentan continuamente. «Este segundo miembro de familia es un ejemplo más del continuado compromiso y de la dedicación de Digital para ofrecer al mercado un PDP-11 más potente a un precio más reducido. El Micro PDP-11/73 proporciona soporte multiterminal con sistemas operativos de tiempo compartidos. Comparándolo con otros sistemas PDP-11, se caracteriza por un precio más económico por terminal.

Micro PDP-11/73 incorpora el juego de «chips» J-11 de 15 MHZ. El conjunto de chips J-11 incluye el juego de instrucciones básicos del PDP-11, EIS (repertorio ampliado de instrucciones), instrucciones de punto flotante, 8 Kbytes de memoria caché, una unidad de gestión de memoria (MMV), y un microcódigo para el terminal de consola. La tarjeta de la CPU incluye 32 Kbytes, un ROM de diagnóstico/autoarranque, 2 Kbytes EEROM, una línea para consola serie y un reloj de línea controlado por software.

OTRO S.I.M.O. AL CAER

Entre los días 16 y 23 del próximo mes de noviembre, se celebrará en Madrid la vigésimo cuarta edición de S.I.M.O. (Feria Oficial Monográfica Internacional del Equipo de Oficina y de la Informática).

Al igual que en los años anteriores, la manifestación estará emplazada en el Recinto Ferial de la Casa de Campo, ocupando cuatro de los pabellones del recinto de IFEMA (Institución Ferial de Madrid). S.I.M.O. estará abierto al público todos los días de 10,30 a 20 horas, sin interrupción, excepto el domingo día 18 que permanecerá abierto hasta las 15 horas y cerrado por la tarde. El precio de entrada para el público será de 250 ptas.

Las previsiones de los organizadores para la nueva edición de S.I.M.O. superan a los resultados de la edición anterior. En 1983, fueron un total de 1.714 expositores, entre directos e indirectos, los que se dieron cita en el recinto ferial; de ellos 779 españoles. El conjunto de stands, visitados por 157.747 usuarios, ocuparon 24.549 metros cuadrados netos.

Como datos más significativos de la edición 1984, cabe señalar el mantenimiento de la zona SIMOMICRO, dedicada al ordenador personal, y la presentación de dos nuevas zonas especiales: SIMOLOG, dedicada a la exposición de productos «software» y SIMODAT, zona reservada para bases y bancos de datos.

Los días 16, 19, 20, 21, 22 y 23, tendrán la consideración de «jornadas profesionales», de 10,30 a 15 horas.

NOVEDADES DE ABC-ANALOG

La firma ABD-ANALOG ha ampliado su catálogo de programas para microordenadores incorporando las siguientes novedades.

SPECTRUM: MACIACS (DK018), (ZIG ZAG) (DK046) y COSMIC CRUISER (IM010), cuyos respectivos precios de venta al público son: 2.200, 2.200 y 1.800 ptas., respectivamente.

COMMODORE-64: MACIASCS (DK044), 3D TANK (DK043) y COSMIC CRUISER (IM110), cuyos respectivos precios son 2.200, 2.200 y 1.800 ptas.

EL DRAGON YA ES ESPAÑOL

El pasado día 20 de julio se firmó el acuerdo entre Dragon Data Ltd., cuyos principales accionistas son Protec y Agencia Desarrollo Industrial de Gales, y la empresa española Eurohard S. A., para la fabricación en España de los conocidos microordenadores Dragon.

Los menores costes de fabricación así como la próxima integración española en las Comunidades Europeas han inclinado la balanza a favor de Eurohard frente a otras propuestas de diversas nacionalidades. La empresa Eurohard S. A. presidida por Eduardo Merigó, está participada por Sodiex, Diputación de Cáceres y un grupo de accionistas privados.

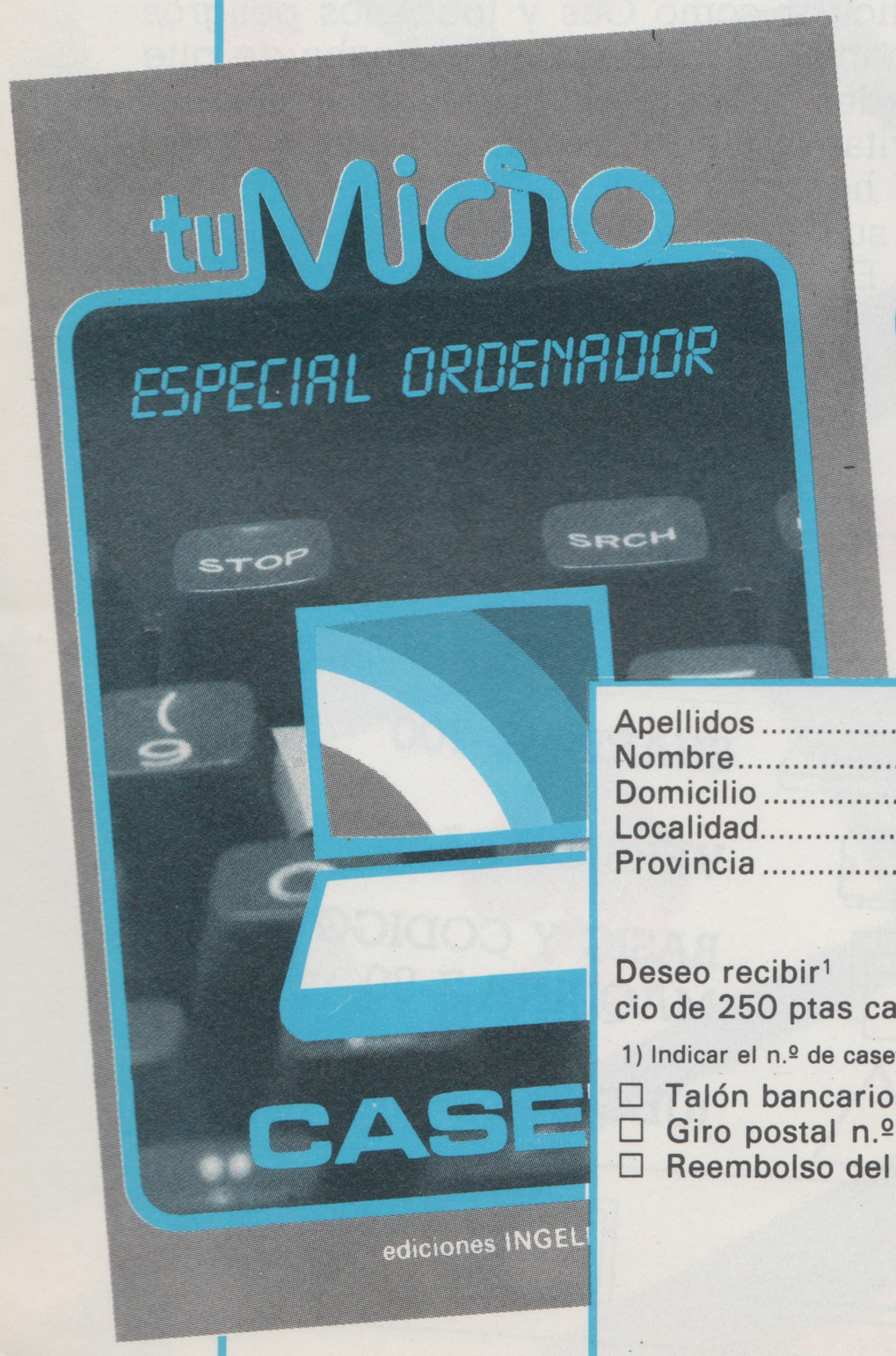
En muy breve plazo, empezarán a salir de la factoría de Cáceres los primeros ordenadores Dragon 32 y Dragon 64. Desde esta factoría se abastecerán los mercados europeos de dichos microordenadores, situados en el tercer puesto del ranking de ventas. Esta posición de Dragon en el mercado convertirán a Eurohard S. A. en el primer exportador de material informático de España.

Las primeras unidades a exportar serán con destino a Venezuela y con SOFTWARE de aplicación español, así como continuará el suministro del importante mercado Británico y de Europa del Norte.

NUEVA IMPRESORA PARA LOS ORDENADORES DE BOLSILLO SHARP

Mecanización de Oficinas, S. A. ha puesto en el mercado nacional una nueva impresora, la CE-126P, destinada a los microordenadores de bolsillo Sharp de las series PC-1245, PC-1251, PC-1401 y PC-1421.

El nuevo periférico, cuyo precio es de 16.800 ptas. permitirá configurar equipos de bolsillo completos y autónomos con una reducida inversión: a partir de las 33.000 pesetas.



DISPONIBLE CASETE VIRGEN

250 ptas

Apellidos .....
Nombre .....
Domicilio .....
Localidad .....Teléfono .....
Provincia .....Código Postal .....

Deseo recibir[1] CASETES VIRGENES «TU MICRO», especiales para ordenador, al precio de 250 ptas cada una. El importe lo haré efectivo mediante:

1) Indicar el n.º de casetes solicitado

☐ Talón bancario a nombre de TU MICRO. n.º de talón .....Banco .....
☐ Giro postal n.º .....Fecha .....Localidad.....
☐ Reembolso del pedido más gastos de envio.

FIRMA

Recortar o copiar este cupón y enviar a:
Ediciones INGELEK Apartado de Correos 61294 28080 MADRID



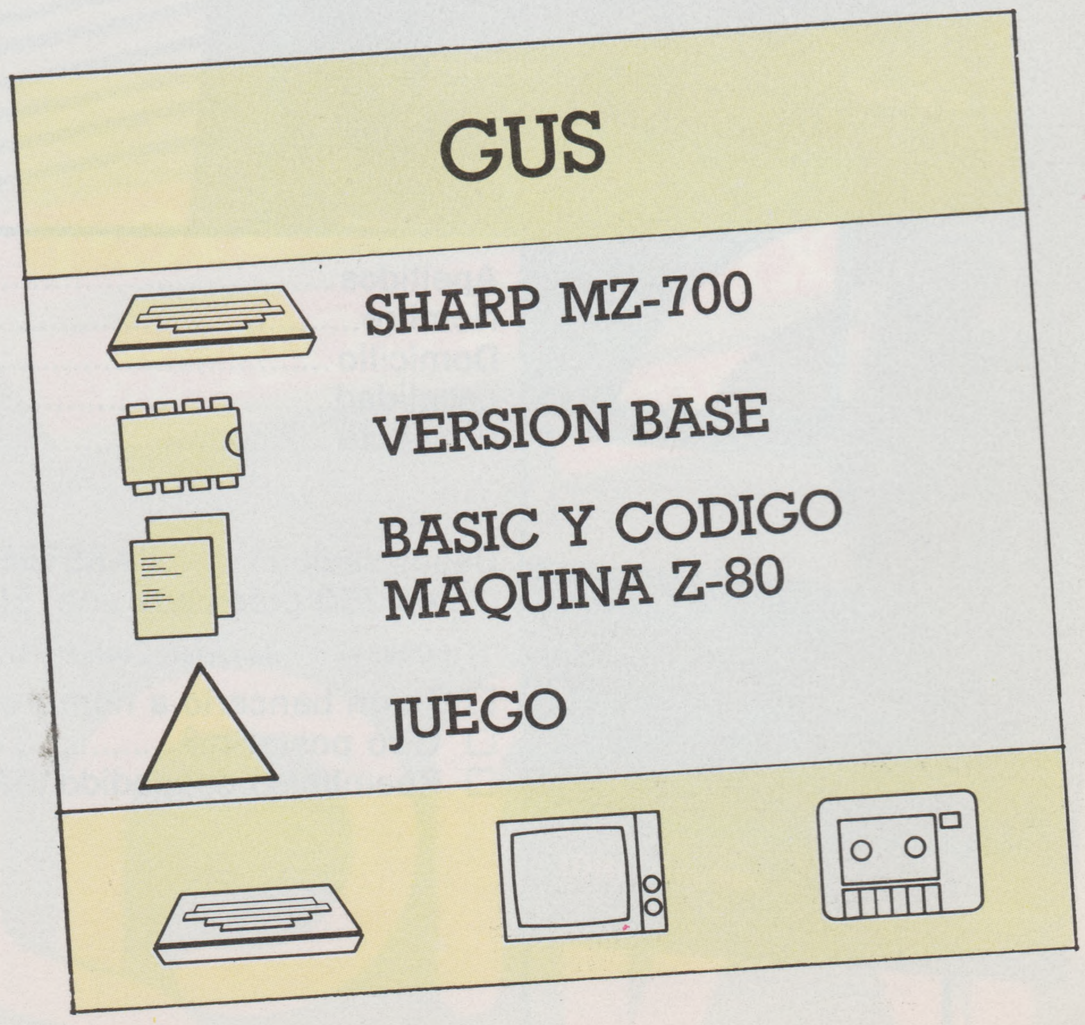
GUS

Aquellos lectores que compartan su interés por TU MICRO con el estudio de nuestra publicación hermana, la «Enciclopedia Práctica de la Informática», habrán tenido ya oportunidad de disfrutar de este simpático programa para el ZX-SPECTRUM. Este mes, el gracioso Gus cambia de domicilio y se viene a habitar una nueva manzana: el Sharp MZ-700.

Gus es un gusano muy especial: se pirra por la fruta y vive en un recinto del parque zoológico, donde es alimentado con uvas, manzanas, y fresas. Aparte del bonito color azul de sus anillos y de su negra cabeza provista de poderosas mandíbulas, algo más hace de él un reptil muy particular: cuando come, su cuerpo se alarga en varios anillos. Una vez que ha tomado su ración diaria (10 frutas), Gus se retira a hacer la digestión, apareciendo de nuevo al día siguiente con su longitud habitual.

Para su desgracia, la envidia ha hecho presa en los directores de otros zoos, e intentan envenenar a tan extraño especimen arrojando en su recinto unos abominables y venenosísimos bichejos de color azul, cuya deglución Gus debe evitar.

Hoy día, la vida no es fácil para un anélido tan particular como Gus y todos los peligros antes mencionados se unen al hecho de que las paredes de su recinto están electrificadas para evitar la huida del curioso animal. Además, ha heredado una característica de algunos de sus camaradas, los reptiles ofídios: el veneno. En efecto, Gus es tan venenoso que si llegara a morderse por accidente, moriría.



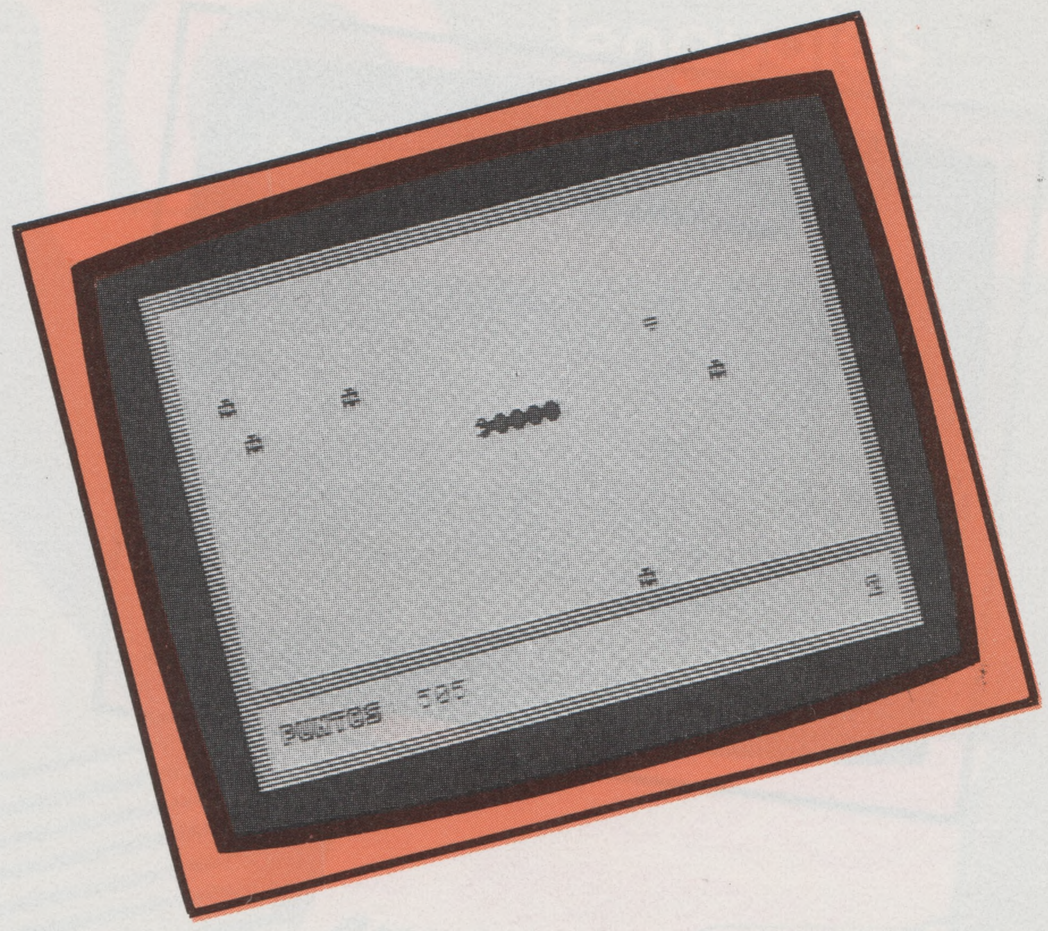
¡A «gusear»!

De acuerdo a las normas enunciadas hasta ahora, el objetivo del juego es guiar al reptil para que coma la mayor cantidad posible de frutas, evitando siempre los siguientes infortunios:

1. Probar el amargo sabor de los venenosos bichejos azules.
2. Electrocutarse con las paredes del recinto.
3. Caer víctima de la fiereza de sus propias mandíbulas.

GUS se mueve al accionar las flechas de los cursores, y una vez que ha echado a andar, su instinto devorador hace que no se detenga hasta engullir otra fruta.

Por cada bocado alimenticio, el jugador recibirá una cantidad variable de puntos; tal puntuación depende de los pasos que le hayan



CUADRO DE VARIABLES

A	Dirección de comienzo de la pantalla de atributos.
C	SWITCH para indicar a la subrutina 730-790 si se desean seleccionar bichejos o frutas.
F	Uno más del número de frutas engullidas en la jornada.
G	Caracter que representa la cabeza de Gus, según su sentido de movimiento.
I	FOR de diversa utilidad.
L	«Guses» vivos.
N	Diversa utilidad.
P	Puntos.
R	Record.
S	Jornada.
T	Número de pasos dados por Gus hasta comerse una fruta.
V	Dirección de comienzo del archivo de imagen.
X	Diversa utilidad.
CB	Puente en la obtención de CG.
CF	Columna de impresión de la fruta en pantalla.
CG	Columna de impresión de la cabeza de Gus.
FB	Puente en la obtención de FG.
FF	Fila de impresión de la fruta en pantalla.
FG	Fila de impresión de la cabeza de Gus.
FP	Número de fruta en pantalla dentro de la jornada.
PB	Puente en la obtención de P.
VF	Valor de la fruta comida.
M$	Cadena para GET de movimiento de Gus.
X$	Cadena de diversa utilidad.
F(2,1)	Tabla con los códigos de pantalla y atributo de las tres frutas.

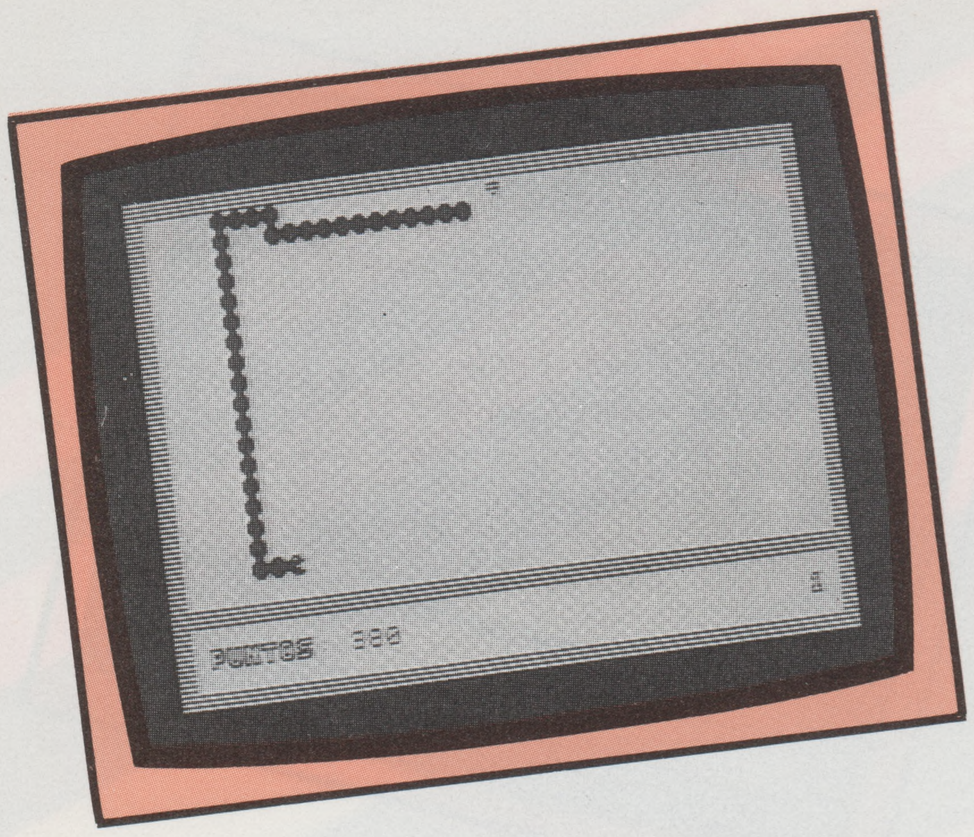
ESTRUCTURA DEL PROGRAMA

10-100	Inicializaciones del programa.
110	Inicializaciones de partida.
120-300	Inicializaciones de jornada.
310-330	Entrada por comienzo de jornada o fruta comida.
340-400	Toma de datos para movimiento.
410-480	Movimiento de Gus.
490-570	Comer fruta.
580-590	Muerte de un Gus.
600-680	Fin de programa.
690-700	Subrutina para creación de las paredes horizontales del recinto de Gus.
710-720	Subrutina para la creación de las paredes verticales del recinto de Gus.
730-790	Subrutina de selección e impresión de bichejos o frutas en pantalla.

sido necesarios a Gus para alcanzar la comida, del número de jornadas que llevemos jugando (recuerden que por cada diez frutas se pasa a una nueva jornada) y de la fruta degustada. El valor nutritivo de éstas sigue el presente orden: fresa más que manzana y esta más que uva.

¡Ah!... se nos quedaba en el tintero un hecho importante: el pantagruélico carácter de Gus le lleva en algún caso a tener alucinaciones y, como es lógico, siguiendo sus preferencias en el tema nutritivo, la mayor probabilidad de estar en presencia de una alucinación ocurre cuando ve una fresa y, a continuación, si ve una manzana; las uvas no son santo de su devoción, por lo que nunca resultan ser alucinaciones.

El jugador dispone de tres anélidos; una vez fenecidos se pasará a una pantalla en la que



se le notificará el número total de puntos, la cantidad de frutas ingeridas y el número de jornadas concluidas, así como si ha conseguido o no batir el récord de puntuación, citándosele, en tal caso, para una nueva partida.

Durante el juego, aparecerá en la parte inferior de la pantalla un marcador con los puntos y otro con el número de jornada en la que nos encontramos; tenga en cuenta que por cada nueva jornada aparecerán cinco bichos venenosos más.

Y para terminar, ahí va un dato para la correcta interpretación del listado: los caracteres subrayados corresponden a los caracteres de control que se obtienen pasando el cursor a GRAFICOS (tecla GRAPH).

LISTADO

```
10 REM Fernando Lopez Martinez # GUS # Sharp MZ-700
20 LIMIT $AFFF
30 DATA 0,0,17,0,176,175,1,10,0,213,33,37,176,229,237,74,26,119,
19,35,26,119
40 DATA 225,209,126,18,19,35,126,18,84,93,27,35,237,176,201
50 DATA 10,21,10,20,10,19,10,18,10,17
60 FORI=45056TO45102:READX:POKEI,X:NEXTI
70 DATA 16,21,14,20,15,19
80 A=55296:V=53248
90 R=0
100 DIMF(2,1):F(0,0)=166:F(0,1)=198:F(1,0)=144:F(1,1)=166:F(2,0)
=157:F(2,1)=182
110 P=0:S=0:L=3
120 S=S+1
130 F=1
140 FG=10:CG=17
150 COLOR,,1,6
160 PRINT"C"
170 N=0:GOSUB690:N=24:GOSUB690:N=20:GOSUB690
180 N=0:GOSUB710:N=39:GOSUB710
190 RESTORE70:FORI=22*40+2TO22*40+7:READX:POKEA+I,150:POKEV+I,X:
NEXTI
200 X$=STR$(S):X=22*40+38-LEN(X$)
210 N=1:FORI=XTOX+LEN(X$)-1
220 POKEA+I,150:POKEV+I,ASC(MID$(X$,N,1))-16:N=N+1
230 NEXTI
240 IFS=1THENGOTO260
250 C=1:FORI=1TO(S-1)*5:GOSUB730:NEXTI:C=0
260 CURSOR8,22:PRINT":"P
270 FORI=10*40+21TO10*40+18STEP-1:POKEA+I,150:POKEV+I,62:NEXTI
280 POKEA+I,134:POKEV+I,74
290 RESTORE50:FORI=45093TO45102:READX:POKEI,X:NEXTI:POKE($B007),
10
300 TEMPO1:MUSIC"AOBOCODOEOFOGO"
310 GOSUB730
320 GETM$:IF(M$<>"↑")*(M$<>"←")*(M$<>"→")*(M$<>"↓")THENG
OTO 320
330 T=0:GOTO350
340 GETX$:IF (X$="↑")+(X$="←")+(X$="→")+(X$="↓")THENM$=
X$
350 IF(T>FP*10+15)*(FP<>0)THENT=0:POKE(V+FF*40+CF),0:GOSUB730
360 FB=FG:CB=CG
370 IFM$="↑"THENFB=FG-1:G=73
380 IFM$="↓"THENFB=FG+1:G=75
390 IFM$="←"THENCB=CG-1:G=74
400 IFM$="→"THENCB=CG+1:G=76
410 POKE$B000,FB:POKE$B001,CB:USR($B002)
420 X=PEEK(V+FB*40+CB)
430 POKEA+40*FG+CG,150:POKEV+40*FG+CG,62
440 FG=FB:CG=CB
450 POKEA+40*FG+CG,134:POKEV+40*FG+CG,G
```

```
460 POKEV+40*PEEK($B000)+PEEK($B001),0
470 T=T+1:TEMPO7:MUSIC"-A0"
480 IFX=0THEN340
490 IF(X=118)+(X=62)+(X=54)THENGOTO580
500 VF=F+1:IFFP=1THENVF=3
510 IFFP=2THENVF=2
520 PB=S+F+5*VF-INT(T/25):IFPB<0THENPB=0
530 P=P+PB:F=F+1:CURSOR9,22:PRINTP
540 TEMPO7:MUSIC"-A0B0C0A0":POKE$B007,10*F
550 FORI=45103+10TO45111*10STEP2:POKEI,22:POKEI+1,1:NEXTI
560 IFF=11THENGOTO120
570 GOTO310
580 TEMPO6:MUSIC"-B2R1-B2R1-B2R1-A4R5-D2R2-D2R2-D2R2-C5"
590 L=L-1:IFL=130
600 S=S-1:COLOR,,6,2:PRINT"C"TAB(13)"- PUNTUACION -"
610 CURSOR14,3:PRINT"PUNTOS:"P
620 CURSOR14,5:PRINT"FRUTAS:"F+S*10-1
630 CURSOR14,7:PRINT"FASES.:"S
640 IFP>RTHENR=P:CURSOR12,9:PRINT"BATIO EL RECORD!!"
650 CURSOR10,13:PRINT"DESEA JUGAR DE NUEVO?"
660 GETX$:IF(X$<>"S")*(X$<>"N")THENGOTO660
670 IFX$="S"THENGOTO110
680 END
690 FORI=N*40TO39+N*40:POKEA+I,240:POKEV+I,118:NEXTI
700 RETURN
710 FORI=40+NTO920+NSTEP40:POKEA+I,240:POKEV+I,118:NEXTI
720 RETURN
730 N=VAL(TI$)
740 FP=INT(RND(N)*3)
750 FF=INT(RND(N)*19)+1:CF=INT(RND(N)*39)+1
760 IFPEEK(V+FF*40+CF)<>0THENGOTO750
770 IFCTHENPOKEA+40*FF+CF,150:POKEV+40*FF+CF,54:GOTO790
780 POKEA+40*FF+CF,F(FP,1):POKEV+40*FF+CF,F(FP,0)
790 RETURN
```

END

trucos y recetas

ZX-SPECTRUM

Receta para el SPECTRUM: liste un programa que ocupe más de una pantalla y el mensaje "SCROLL ?" responda con una acción sobre las teclas CAPS/SHIF y 2 (CAPS LOCK)... ¡La ROM es una fuente inagotable de sorpresas!
Observará que al pulsar las referidas teclas se restablece en las dos líneas inferiores de la pantalla el último contenido del área de trabajo.
A simple vista el truco no tiene utilidad alguna... pero ¿a que resulta chocante?

ZX-SPECTRUM

¿Cómo deshacerse de un programa dejando totalmente limpia la memoria RAM del SPECTRUM... y sin recurrir al apagón? NEW no es la solución, puesto que no borra más allá del RAMTOP ni «resetea» los gráficos definidos. Por otra parte, la alternativa de desconectar la alimentación supone un esfuerzo que puede llegar a deteriorar la clavija a base de «borrados», y desenchufar la toma de red también supone un esfuerzo, aunque esta vez de riñones. La solución cómoda... ¡para expertos!:
PRINT USR Ø
¡Así de fácil!

FINOS

¡Hala... qué caracteres más gordos! Poco más o menos, esta es la primera frase que el flamante propietario de un COMMODORE 64 dirige, de viva voz, a su nueva máquina. Y este es el primer deseo: ¿de dónde saco yo una dieta contra la obesidad de los caracteres?... De las páginas de TU MICRO ¡claro está!

En TU MICRO también nos ocupamos de la estética y desde hace ya algún tiempo andamos preocupados por los 2 pixels de anchura del repertorio de caracteres del C-64. Tras varios días de devaneo, en busca de documentación apropiada, uno de los redactores tropezó su occipital con un antiguo tratado que nos puso en el buen camino. El libro, «Bytes dietéticos y otras hierbas para el riñón», del Doctor A. N. T. Carvajal, profundizaba en el tema con gran detalle, hasta el punto de dedicar todo un capítulo al diagnóstico en pantalla de TV color y a la profilaxis —¡desde luego que por programa!— del denominado pixel lípido del C-64.

La conclusión de nuestro trabajo se plasma en el programa que pasamos a describir. Una dieta en código máquina, de eficacia constatada, que redefine el repertorio de caracteres del equipo, tanto mayúsculas como minúsculas, y modela su «cintura» a un solo pixel de ancho (¡De «finuras» va el programa!).



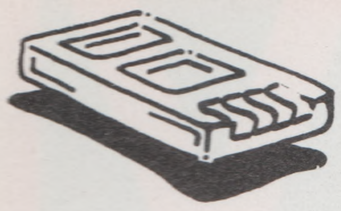
El problema y la solución

La razón que da Commodore para justificar semejante «obesidad» en los caracteres, es que no puede esperarse la misma resolución en un televisor doméstico que en un monitor. Bien es cierto que con los caracteres que incorpora la máquina no plantea problemas ni el peor de los receptores; no obstante, nuestro repertorio de «finuras» da excelentes resultados sin mayores precauciones que la de elegir un buen color de fondo (preferible de tono claro) y un buen color para los caracteres (oscuro; negro, por ejemplo).

La redefinición de caracteres nos brinda la posibilidad de incluir en el nuevo repertorio algunos caracteres no contemplados: la letra «eñe» (en mayúsculas y minúsculas), por ejemplo, que ocupa el lugar del símbolo £. También hemos aprovechado la ocasión para mejorar la presentación de algunas letras; la «eme» minúscula, sin ir más lejos, cuya «belleza» original brillaba por lo atípico.

La dieta programada

El programa utiliza el juego de caracteres ubicado en memoria RAM, a partir de la dirección 12288 y hasta la 16384. Un dato a señalar es que el propio programa «Finos» se encarga de salvaguardar el juego de caracteres, protegiendo su coto en RAM de la posible invasión por parte de cualquier programa BASIC. Para el almacenamiento de estos últimos quedan libres 10 Kbytes, zona que suele ser suficiente para el programador.



La zona de programa en código máquina está dividida en cuatro subrutinas básicas:

1. C000-C0065:

Copia de los caracteres de la ROM a memoria RAM.

2. C066-C08A:

Transformación primaria de caracteres.

3. C08B-C0DE:

Subrutina de transformación.

4. C0DF-C2E5:

Transformación definitiva de caracteres.

Como puede observar, la «reconversión» se desarrolla en dos etapas, una de transformación primaria y otra de acondicionamiento definitivo del repertorio de caracteres.

Y esto es todo; puede ya empezar con la introducción manual del programa, esgrimiendo el dedo de la primera a la última línea, o auxiliado por el truco que se describe en el cuadro adjunto. En cualquier caso, recuerde que antes de ejecutar el programa por vez primera, es conveniente almacenar en casete una copia del mismo; una precaución que le evitará empezar de nuevo la danza dactilar en el caso de que cometa algún error durante la introducción.

tuMicro

La receta final

Para terminar, vamos a contarle un pequeño truco (¡cosa fina!) que simplificará la carga manual del programa, más precisamente del bloque de instrucciones DATA. Veamos cuáles son los pasos a seguir:

1. Introduzca en modo directo la instrucción siguiente: POKE 650,128.
2. A continuación, escriba la línea 1000 como sigue: 1000 FOR I=155 TO 155+105 STEP 5: PRINT I; "DATA": NEXT.
3. Ahora ejecute un RUN 1000, con lo que la pantalla se llenará de números de línea acompañados del comando DATA.
4. Accione la tecla HOME y mantenga la tecla RETURN pulsada hasta el final. A continuación, vaya a la línea 1000 y sustituya los dos 155 que figuran en la misma por el último número aparecido en la pantalla. Ejecute, de nuevo, un RUN 1000 y así hasta terminar en la línea 620.

¿El resultado?... aplicando este método se encontrará con que ya tiene introducidos los números de línea y la palabra comando de todas las instrucciones DATA. ¡Un sustancial ahorro de tiempo! Ahora ya sólo le quedan por introducir los números de las sentencias DATA y las líneas de programa BASIC.

LISTADO



```
100 REM  ********************************
105 REM  * FINOS.POR ANTONIO CARVAJAL *
110 REM  *       PARA    COMMODORE 64 *
115 REM  ********************************
120 PRINT"■■":POKE53280,11:POKE53281,12:POKE53272,21
125 PRINT"Q]]A. CARVAJAL *TU MICRO* A£O 1984"
130 PRINT"QQ]]ESTE PROGRAMA CREA CARACTERES FINOS     ASI COMO LA LETRA:QQ"
135 PRINT"]]ATENCION A LA ££££££££££££QQ"
140 FORI=33TO95:PRINTCHR$(I);" ";:NEXT
145 FORI=49152TO49893:READA:C=C+A:POKEI,A:NEXT
150 IFC<>88347THENPRINT:PRINT:PRINT" ERROR EN DATAS":END
155 DATA 169,48,133,52,133,56,173,14
160 DATA 220,41,254,141,14,220,165,1
165 DATA 41,251,133,1,169,0,133,251
170 DATA 169,16,133,252,169,0,133,253
175 DATA 169,48,133,254,169,0,133,139
180 DATA 169,208,133,140,160,0,166,252
185 DATA 240,14,177,139,145,253,200,208
190 DATA249,230,140,230,254,202,208,242
195 DATA 166,251,240,8,177,139,145,253
200 DATA 200,202,208,248,165,1,9,4
205 DATA 133,1,173,14,220,9,1,141
210 DATA 14,220,173,24,208,41,240,24
215 DATA 105,12,141,24,208,96
220 DATA 160,0
225 DATA 162,2,240,15,177,251,32,139
230 DATA192,145,251,200,208,246,230,252
235 DATA 202,208,241,162,0,240,11,177
240 DATA 251,32,139,192,145,251,200,202
245 DATA 208,245,96,201,102,208,3,169
250 DATA 66,96,201,96,208,3,169,64
255 DATA 96,201,24,208,3,169,16,96
260 DATA 201,6,208,3,169,2,96,201
265 DATA 12,208,3,169,8,96,201,48
270 DATA 208,3,169,32,96,201,99,208
```

```
275 DATA 3,169,65,96,201,119,208,3
280 DATA 169,99,96,201,107,208,3,169
285 DATA 73,96,201,118,208,3,169,49
290 DATA 96,201,108,208,3,169,68,96
295 DATA 201,110,208,2,169,70,96
300 DATA 169,24,141,8,48,141,182,48
305 DATA 141,195,48,141,203,48,141,196
310 DATA 56,141,212,56,169,36,141,9
315 DATA 48,141,181,48,141,194,48,141
320 DATA 196,48,141,202,48,141,16,49
325 DATA 141,17,49,141,18,49,141,16
330 DATA 57,141,17,57,141,18,57,141
335 DATA 195,56,141,197,56,169,56,141
340 DATA 72,48,141,78,48,141,86,48
345 DATA 141,232,48,141,238,48,141,242
350 DATA 48,141,232,56,141,238,56,141
355 DATA 242,56,141,72,56,141,78,56
360 DATA 169,28,141,80,48,169,72,141
365 DATA 85,48,141,89,48,141,93,48
370 DATA 141,148,48,141,34,49,141,34
375 DATA 57,169,70,141,88,48,141,94
380 DATA 48,141,117,48,141,229,48,141
385 DATA 129,49,141,129,57,169,80,141
390 DATA 90,48,141,92,48,141,138,49
395 DATA 141,138,57,169,96,141,91,48
400 DATA 169,98,141,113,48,141,225,48
405 DATA 141,41,49,141,133,49,141,133
410 DATA 57,141,41,57,169,82,141,114
415 DATA 48,141,226,48,141,132,49,141
420 DATA 132,57,169,90,141,115,48,141
425 DATA 227,48,141,131,49,141,131,57
430 DATA 169,74,141,116,48,141,228,48
435 DATA 141,130,49,141,130,57,169,12
440 DATA 141,142,48,169,124;141,160,48
445 DATA 141,243,48,141,91,49,141,142
450 DATA 49,141,224,56,141,226,56,141
455 DATA 226,56,141,91,57,141,142,57
460 DATA 169,32,141,205,48,141,206,48
465 DATA 141,213,48,141,213,56,169,4
470 DATA 141,209,48,141,56,49,141,122
475 DATA 49,141,242,49,141,244,49,141
480 DATA 56,57,141,122,57,141,242,57
485 DATA 141,244,57,169,60,141,211,48
490 DATA 141,205,56,169,16,141,212,48
495 DATA 141,253,48,141,44,49,141,212
500 DATA 56,141,253,56,141,44,57,169
505 DATA 94,141,224,48,169,66,141,230
510 DATA 48,141,227,56,141,228,56,141
515 DATA 229,56,141,230,56,169,48,141
520 DATA 250,48,141,252,48,141,137,49
525 DATA 141,250,56,141,252,56,141,137
530 DATA 57,169,0,141,254,48,141,40
535 DATA 49,141,225,56,141,231,56,141
540 DATA 254,56,141,40,57,141,72,56
545 DATA 169,8,141,32,49,141,38,49
550 DATA 141,43,49,141,241,49,141,32
555 DATA 57,141,38,57,141,245,49,141
560 DATA 241,57,141,245,57,141,43,57
565 DATA 169,10,141,36,49,141,161,49
570 DATA 141,36,57,141,161,57,169,100
575 DATA 141,42,49,141,42,57,169,2
580 DATA 141,121,49,141,121,57,169,38
585 DATA 141,45,49,141,45,57,169,6
590 DATA 141,160,49,141,160,57,169,18
595 DATA 141,162,49,141,162,57,169,34
600 DATA 141,163,49,141,163,57,141,189
605 DATA 56,141,190,56,169,119,141,106
610 DATA 56,169,73,141,107,56,141,108
615 DATA 56,141,110,56,169,85,141,188
620 DATA 56,169,112,141,207,56,96
625 SYS49152
630 POKE251,0:POKE252,48
635 SYS49254
640 POKE251,0:POKE252,56
645 SYS49254
650 SYS49375
655 PRINT:PRINT:PRINT"PARA VER LAS MINUSCULAS PULSE UNA TECLA"
660 GETA$:IFA$=""THEN660
665 POKE251,0:POKE252,2:POKE253,0:POKE254,58:POKE139,0:POKE140,48
670 SYS49196
675 POKE 53272,31
680 FORI=12512TO12512+7:POKEI+2664,PEEK(I):NEXT
685 PRINT:PRINT:PRINT"  CAMBIADAS MINUSCULAS Y MAYUSCULAS"
690 END
```

ESTRUCTURA DEL PROGRAMA

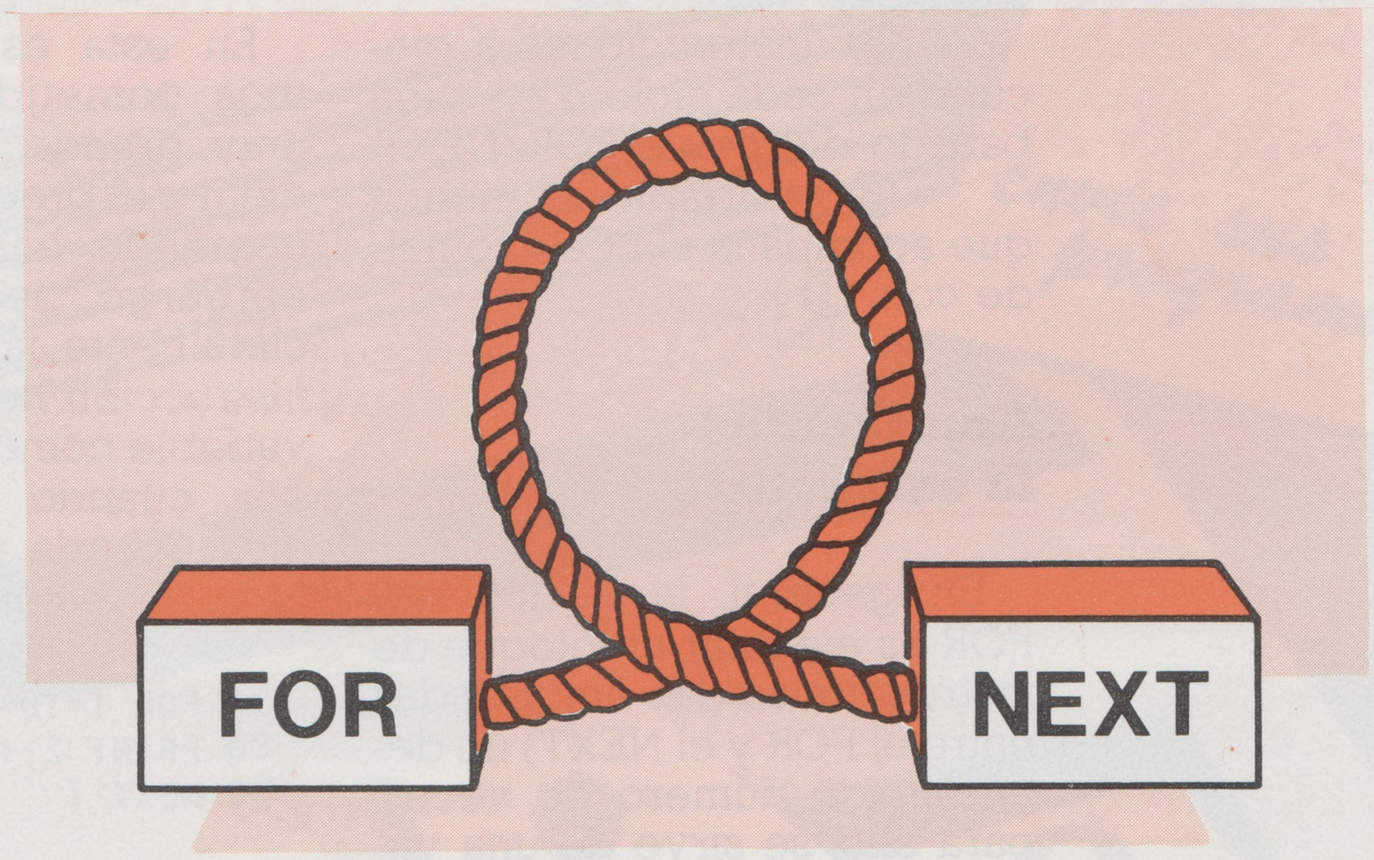
120-135	Cabecera del programa.
140	Caracteres a cambiar.
145	Lectura de DATAs y colocación de éstos en memoria.
150	Checksum de datos y mensaje de error.
155-620	DATAs con el código máquina.
625-650	Saltos a subrutinas de código máquina.
655-660	Espera una tecla para ver minúsculas.
665-670	Saltos a subrutinas de código máquina.
680	Cambia «eñe» mayúscula para el juego de minúsculas.
685-690	Mensaje final.

CUADRO DE VARIABLES

A	Variable para la lectura de datos en código máquina.
A$	Variable GET para ver las minúsculas.
C	Variable para «checksum».
I	Variable auxiliar.

END

Estandarización del BASIC (5)



Los bucles o estructuras cíclicas

Con gran frecuencia, se encomiendan al ordenador trabajos basados en monótonas operaciones repetitivas; para ello, el BASIC dispone de las llamadas estructuras cíclicas, sin duda de enorme utilidad para el programador.

Rizando el rizo

Existen lenguajes, como el Pascal, que destinan una parte importante de su filosofía al tratamiento de los bucles (estructuras cíclicas). El BASIC, no obstante, sólo dispone, habitualmente, de una instrucción a tal efecto, aunque de considerable potencia: el FOR.

Más adelante veremos que algunos intérpretes, como el del Oric, han considerado necesaria la inclusión de alguna estructura de bucle adicional y, rizando el rizo, incorporan el REPEAT-UNTIL.

Empezaremos hablando del FOR: una de las instrucciones clásicas del BASIC que aparece en prácticamente todas las versiones conocidas. La estructura general de esta instrucción es la siguiente:

```
FOR ① = ② TO ③ STEP ④
...
... ⑤
...
NEXT ⑥
```

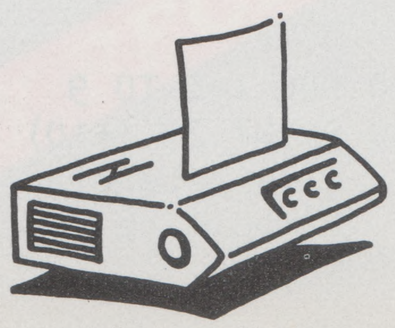
en donde:

① Variable numérica

② Valor inicial de la variable

③ Valor final de la variable

④ Incremento del valor de la variable en cada paso por NEXT





```
10 FOR I=0 TO 9
20 FOR J=1 TO 5
30 PRINT I*J
40 NEXT I
50 NEXT J
 .  .  .
 .  .  .
```

Estructuras cíclicas mal anidadas. Ambos bucles se entrecruzan.

```
10 FOR I=0 TO 9
20 FOR J=1 TO 5
30 PRINT I*J
40 NEXT J
50 NEXT I
 .  .  .
 .  .  .
```

Estructuras cíclicas correctamente anidadas.

⑤ Instrucciones a ejecutar en cada ciclo o en cada barrido del bucle FOR/NEXT.

⑥ Variable numérica que acompaña al FOR (coincide con ①).

Analicemos la estructura

En general, la misión del FOR es ejecutar un bloque de instrucciones (las emplazadas entre el FOR y el NEXT) un determinado número de veces; para ello, se sirve de una variable que actuará fundamentalmente como contador, pero que puede tener otra función más en el ciclo. Aclarémoslo con un ejemplo. Supongamos que se trata de hacer descender nuestra posición de PRINT 10 líneas más abajo, para conseguirlo podemos ejecutar las siguientes instrucciones:

```
10 FOR I=0 TO 9
20 PRINT
30 NEXT I
```

Sin embargo, es frecuente que además de esa finalidad, la variable de FOR tenga otra misión dentro del bucle: la de argumento de una función, parámetro de una instrucción, índice de una variable suscrita, etc... Así, por ejemplo, si quisiéramos escribir las potencias de dos, entre la décima y la decimonovena, podríamos hacerlo como sigue:

```
10 FOR I=0 TO 9
20 PRINT 2↑(I+10) : REM El simbolo ↑ corresponde
   a la potenciacion; en algunos ordenadores este
   simbolo se sustituye por (**) o por (^)
30 NEXT I
```

En esta estructura habríamos cometido un error, no muy grande, puesto que, en efecto, el programa cumple el cometido encomendado; sin embargo, se desaprovecha claramente la potencia de la instrucción: hemos utilizado su variable como un mero contador, cuando también podíamos haberla empleado en la función matemática. Veamos:

```
10 FOR I=10 TO 19
20 PRINT 2↑I
30 NEXT I
```

Ambos programas funcionarán por igual, pero el segundo de ellos ocupará menos memoria y tendrá un menor tiempo de ejecución, y, en todo caso, será más «correcto».

Respecto a la «elegancia» en la ejecución de un FOR podríamos hacer 2 recomendaciones, que si bien no aportan modificaciones sustanciales a un programa, si contribuyen a un mejor estilo. Por una parte, la variable que se utilice para el FOR puede tener un cierto significado «analógico»; por ejemplo, si se utilizara para un marcador de *puntos*, podríamos usar P. Sin embargo, en ocasiones, el sentido de la variable no es más que el de un simple índice, en cuyo caso podríamos utilizar cualquier nombre variable. En este aspecto es donde recomendamos conservar «el orden mental» que siempre debe imperar en un buen informático, y acostumbrarnos a utilizar siempre las mismas variables, (decimos variables, en plural, pues más adelante estudiaremos los anidamientos, en los que se precisa más de una va-

riable). Como costumbre —reminiscencias de la utilización del lenguaje FORTRAN— suelen utilizarse como variables FOR, con la única función de índice, las letras I, J y K, principalmente.

El segundo punto a destacar, que puede evidenciar cómo el programador de un «micro» pasa de ser un simple aficionado y empieza a pensar con la filosofía de un informático, aparece a la hora de utilizar las variables FOR como simples contadores. Supongamos que sólo desea retardar la acción en un programa, durante el tiempo que éste tarda en contar 10. El siguiente par de instrucciones servirá al efecto:

```
10 FOR I=1 TO 10
20 NEXT I
```

Algo hay fuera de lógica en este bucle de retardo. ¿Por qué hemos empezado a contar desde 1 y no desde 0? Por alguna razón oculta, en el mundo «no informático» el número cero es una especie de proscrito: la numeración de las calles de una ciudad comienza por 1 y no por 0, etc... Sin embargo, el sistema de numeración comienza por cero (ya sea decimal, binario, hexadecimal...). Por lo tanto, la forma correcta de realizar el programa anterior sería:

```
10 FOR I=0 TO 9
20 NEXT I
```

A aquél que todavía albergue alguna duda sobre lo dicho, puede resultarle útil la siguiente reflexión: el programa que aparece a continuación realiza exactamente la misma función...

```
10 FOR I=7 TO 16
20 NEXT I
```

e incluso ocupa la misma cantidad de memoria, sin embargo, se nos antoja totalmente absurdo ¿verdad? (el que diga que no... ¡que se borre de la informática!). Pues bien, tan absurdo como le parece al aficionado esta última solución, le parece al informático la de empezar a contar por el 1.

Aclaraciones sobre el FOR

Existen una serie de aclaraciones que es conveniente realizar sobre esta estructura:

- Los intérpretes de algunos ordenadores, averiguan, en el momento de hacer el FOR, la posibilidad de efectuar el ciclo, pasando directamente al NEXT en caso negativo. Otros, sin embargo, no se dan cuenta de esta circunstancia hasta que no acceden por primera vez al NEXT. Así, por ejemplo, una estructura como la siguiente:

```
10 FOR I=5 TO 4
20 PRINT I
30 NEXT I
```

se traducirá en la escritura de un 5 en los intérpretes menos «perspicaces»; pero otros, más desarrollados, pasarán directamente el control a partir del NEXT.

- El anidamiento (bucles dentro de bucles) sigue unas reglas estándar, señaladas en la figura adjunta, por las que el bucle de menor orden (el que comienza más tarde) siempre debe cerrarse antes que el de mayor orden.
- Algunos intérpretes no precisan (aunque si admiten) la especificación en el NEXT de la variable de FOR correspondiente, sino que asumen la debida variable según la norma de anidamiento antes mencionada. De todas formas, cuando se anidan bastantes bucles y de considerable longitud, es conveniente especificar la variable correspondiente, aunque sólo sea para facilitarnos a nosotros mismos la lectura y entendimiento posterior del programa.
- En algunos aparatos no es recomendable abandonar un bucle por un camino distinto al del NEXT. El ejemplo siguiente realizará la toma de 10 números, abandonando inmediatamente el ciclo si encontrara un cero.

```
10 FOR I=0 TO 9
20 INPUT A
30 IF A=0 THEN GOTO 60
40 PRINT A
50 NEXT I
60 ...
```



En algunos intérpretes BASIC, este método conducirá a un error en la interpretación de los siguientes bucles. Por ello, suele recurrirse a la solución de forzar la variable de FOR a un valor superior al límite y enviarla al NEXT:

```
10 FOR I=0 TO 9
20 INPUT A
30 IF A=0 THEN LET I=10 : GOTO 50
40 PRINT A
50 NEXT I
60 ...
```

```
10 LET A=9
20 FOR I=0 TO A
30 INPUT A
40 PRINT A
50 NEXT I
60 ...
```

- De manera estándar (salvo en los pocos equipos que sólo contemplan la aritmética entera), los valores iniciales y finales de la variable de FOR pueden ser negativos o positivos y decimales o enteros. Esto mismo ocurre con el STEP, que contiene la cantidad que se incrementa o decrementa a la variable de FOR para compararla al llegar al NEXT de fin de bucle.
- Como caso general, se asume que el STEP de FOR es 1, siempre que no se defina un valor contrario.
- Por último, aclararemos que, mientras no finalice el bucle, el NEXT realiza un salto automático a la instrucción *siguiente al FOR;* por lo tanto, aunque los valores de los límites del bucle o del STEP se hayan prefijado en el FOR mediante variables, éstos no podrán ver alteradas sus condiciones durante el ciclo mediante la modificación del valor de dichas variables. Así, por ejemplo, el siguiente bucle se realizará 10 veces, sean cuales fueren los valores que se introduzcan para la variable A con posterioridad al FOR.

Otro bucle más

Además del FOR-NEXT, el BASIC de Oric soporta otra estructura de ciclo, la REPEAT-UNTIL; estructura que se asemeja mucho a la de FOR, con la salvedad de que el número de veces que se recorre el bucle no depende del valor de una variable, sino de una determinada condición. Su sintaxis general es la siguiente:

—El bucle comienza con una instrucción REPEAT

—Su final se marca mediante UNTIL (condición).

El bucle que va desde REPEAT hasta UNTIL se ejecutará hasta que se cumpla la condición especificada en UNTIL. En el siguiente ejemplo, se tomarán y escribirán los números ingresados en la variable A hasta que se introduzca un número menor o igual cero (éste último también se escribirá):

```
10 REPEAT
20 INPUT A
30 PRINT A
40 UNTIL A<0
50 ...
```

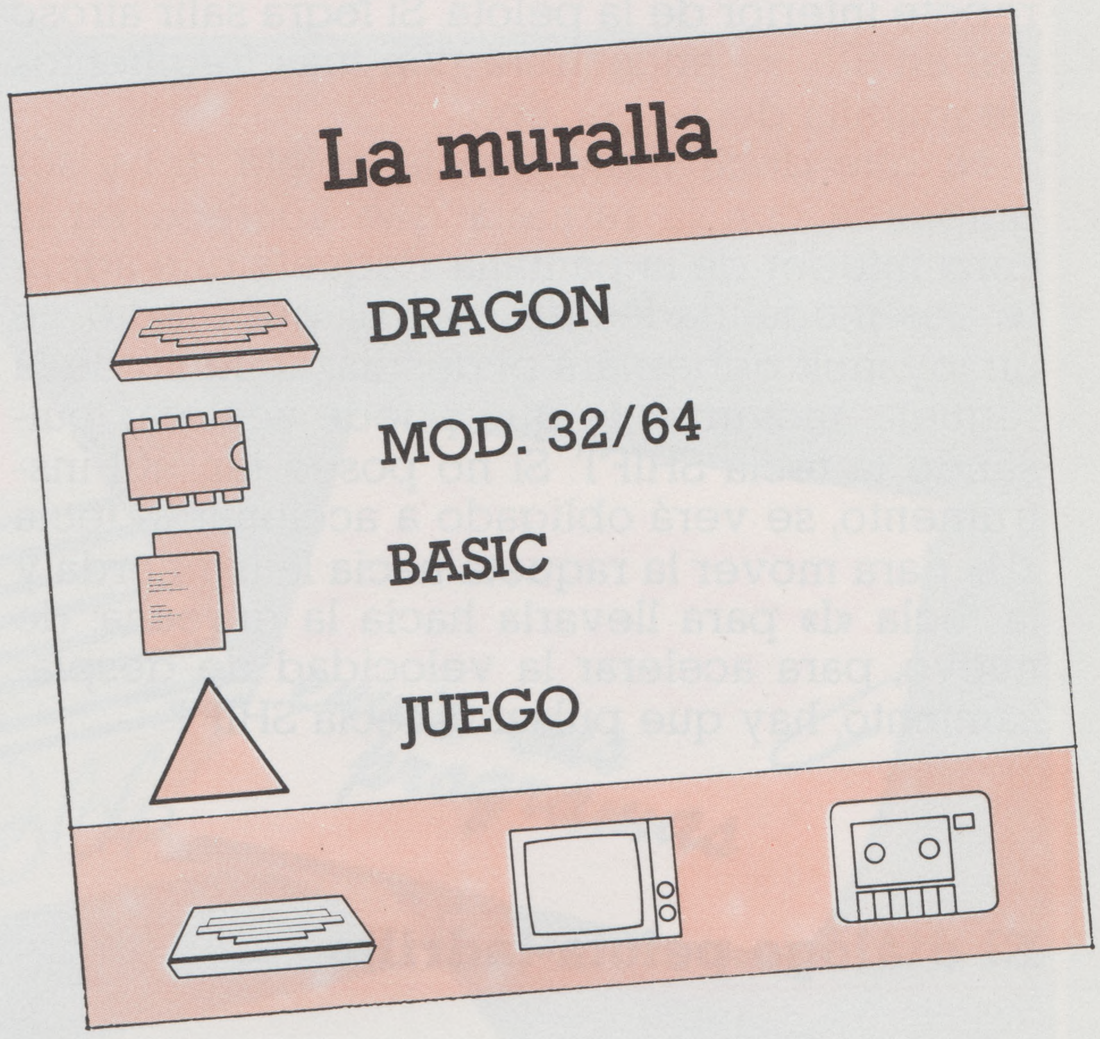
Una última utilidad del bucle la encontrarán en las despedidas afectuosas, ahí va un ejemplo claramente ilustrativo...

```
10 FOR I=0 TO 999
20 PRINT ''MUCHAS GRACIAS POR SU ATENCION''
30 NEXT I
40 PRINT
50 PRINT ''HASTA EL MES QUE VIENE''
```

TABLA DE COMPATIBILIZACION

Símbolo	ORIC	COMMODORE	SINCLAIR	NEW BRAIN	DRAGON	ATARI	Descripción
FORT-NEXT	×	×	×	×	×	×	Ejecuta el bloque de instrucciones comprendido entre FOR y NEXT un determinado número de veces.
REPEAT-UNTIL	×						Ejecuta el bloque de instrucciones comprendido entre REPEAT y UNTIL hasta que se cumple determinada condición.

El ladrillo es uno de los inventos más útiles del cerebro humano. Ahí están los albañiles, sin ir más lejos, fervientes admiradores de tan importante personaje, no afectado por la reconversión, y al que se aferran para no caer en la cola del paro.
No está bien, pues, que alguien piense en atentar contra un auténtico garante de la paz social y del equilibrio arquitectónico. Aquel que sienta deseos de saldar alguna causa pendiente (¡... un ladrillazo mal encajado, por ejemplo!) contra tan digno objeto, puede utilizar este programa para descargar su ira en la pantalla del televisor (... ¡y deje en paz al ladrillo, hombre!)





El programa les resultará familiar, sin lugar a dudas. Hace algunos años —antes de la era del tragaperras—, éste era uno de los video-juegos más populares en cafeterías y salones de billar. Recuerde que el objetivo del entretenimiento era derribar una muralla, destruyendo los ladrillos a pelotazo limpio.

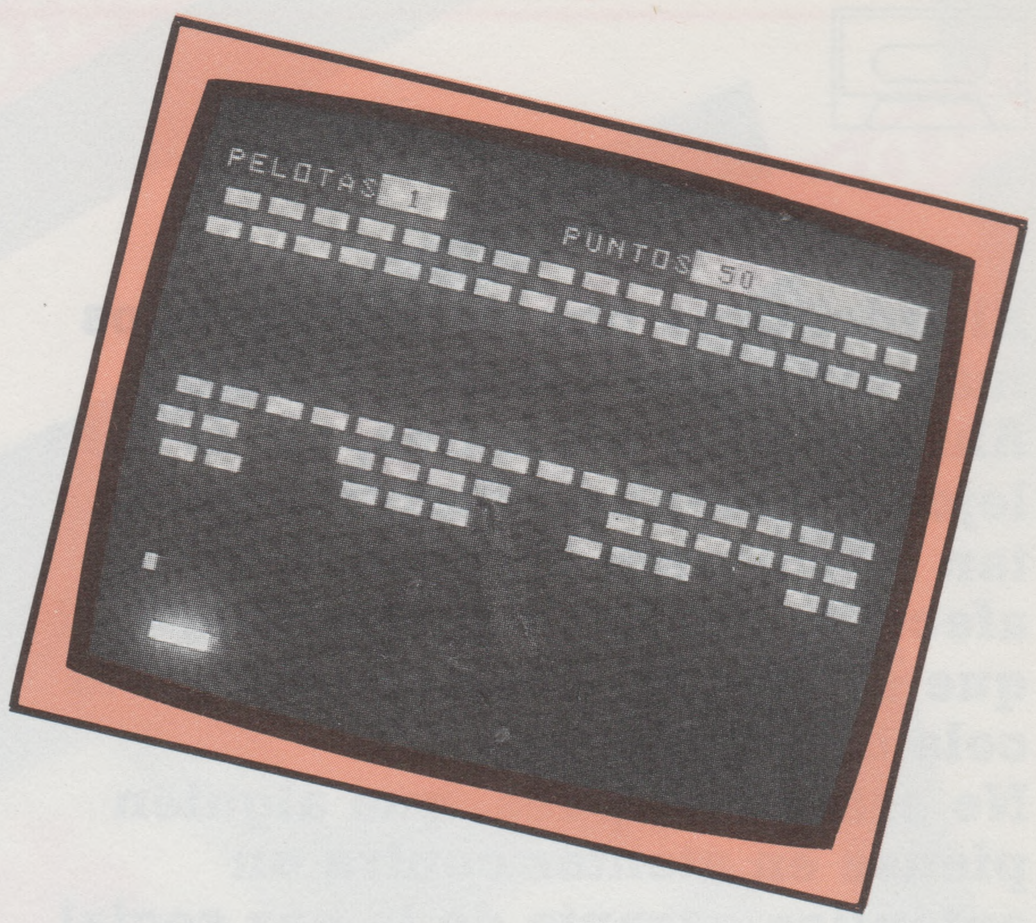
Nuestro programa lleva el popular juego a la pantalla de un receptor de TV conectado al DRAGON 32 ó 64.

Una vez cargado el programa cuyo listado adjuntamos, y obtenida la pertinente copia de seguridad, puede ya aprestarse a demoler la muralla. La primera tarea que acomete el ordenador es la de mostrarle en pantalla las reglas del juego. Al leerlas, observará algunas diferencias respecto al tradicional video-juego. Para empezar, son diez las pelotas de que dispone por partida para derribar la pared. Para ir complicando el juego a medida que su pericia aumenta, el programa le presentará tres tipos de murallas.

Al iniciarse el juego, aparece en escena una muralla del primer tipo, integrada por un total de 80 ladrillos, distribuidos en dos muros separados por una franja libre. Al terminar con su demolición, surgirá una segunda muralla, esta vez de 112 ladrillos, compuesta por dos muros separados por una delgada franja intermedia. Si logra superar este segundo obstáculo, ayu-

dado por los rebotes de la pelota, pasará al tercer tipo de muralla. El nivel de dificultad llega al punto álgido: ¡son 144 los ladrillos que se verá obligado a demoler, uno a uno! Y ya no hay franja libre entre dos muros que facilite el rebote interior de la pelota. Si logra salir airoso del envite, se encontrará con más fragmentos de muralla del tercer tipo.

Ni que decir tiene que la pelota debe ser golpeada con la raqueta que aparece en la zona inferior de la pantalla. Raqueta que admite dos modalidades de control: por medio de un joystick gobernará el desplazamiento de la raqueta, movimiento que puede acelerar pulsando la tecla SHIFT. Si no posee tan útil instrumento, se verá obligado a accionar la tecla «U» para mover la raqueta hacia la izquierda y la tecla «I» para llevarla hacia la derecha; de nuevo, para acelerar la velocidad de desplazamiento, hay que pulsar la tecla SHIFT.



El diálogo pelota-ladrillo

Al impulsar la pelota hacia arriba, ésta rebotará en el ladrillo que encuentra a su paso, destruyéndolo. El impacto en el ladrillo provocará, a su vez, el rebote de la pelota. Si ésta se encuentra en la franja libre abierta entre los dos muros, puede dar lugar a una serie de rebotes sucesivos entre ambos muros, lo que acelerará la demolición.

En todo momento, debe permanecer atento a las evoluciones de la pelota, con el fin de golpearla en el instante adecuado. Si al bajar, ésta rebasa la posición de la raqueta, el instrumento de derribo se perderá del escenario del juego. Recuerde que el juego termina al perder las 10 pelotas disponibles.

Hablemos de técnicas de juego. En las dos primeras murallas, es conviente empezar abriendo un boquete en el muro anterior. La introducción de la pelota a través del agujero facilitará el derribo, puesto que la pelota empezará a rebotar y a destruir los ladrillos que delimitan la franja interior.

Una cierta habilidad en el golpe le permitirá modificar la trayectoria de la pelota. Veamos. Si golpea con la zona central de la raqueta, la pelota seguirá en la orientación del rebote: esto es: si viene por la izquierda, saldrá impulsada hacia la derecha, y viceversa. No obstante, si golpea con uno de ambos extremos de la misma, le pelota invertirá el sentido de rebote y «se irá por donde ha venido», retomando la trayectoria de caída.

Otro punto a señalar es que el final de la pantalla no coincide, en todos los casos, con la destrucción completa de la muralla. Así, las primeras dos pantallas terminan cuando quedan aún indemnes 7 y 5 ladrillos, respectivamente. La tercera y cuarta murallas se complican, puesto que su demolición no se da por consumada hasta que permanezca intactos 3 y 1 ladrillos, respectivamente. A partir de la quinta muralla no hay rebaja: hay que romper todos los ladrillos para pasar a la siguiente pantalla.

La zona superior de la pantalla está ocupada por dos marcadores que informan de la puntuación obtenida y de las pelotas que aún quedan en juego. Cuantas más pantallas se logren superar, mayor será la puntuación que acumulará cada ladrillo destruido.

CUADRO DE VARIABLES

A, B	Nuevas coordenadas de la pelota.
A1, B1	Antiguas coordenadas de la pelota.
A$, B$	Variables para control de los ladrillos.
C	Color de los ladrillos.
C$	Variable para INKEY$.
DD	Código ASCII de la tecla pulsada.
E	Número de pasos para la raqueta.
F	Situación de la raqueta.
F1	Anterior posición de la raqueta.
F$	Indicador de uso de joystick.
FF	Indicador de si «tropiezo» con un ladrillo.
G	Código ASCII de la posición donde se busca el ladrillo.
HH	Posición en la que hay que buscar y borrar el ladrillo.
I	Pulsación del botón de disparo del joystick.
J	Número de ladrillos por pantalla.
K	Variable para RANDOM de aparición de la pelota.
O	Puntuación actual.
P	Incremento de izquierda a derecha.
P2	Incremento de arriba hacia abajo.
PA	Número de la pantalla en juego.
PE	Número de pelotas que quedan.
U	Número de ladrillos que pueden quedar por pantalla.
X	Variables para bucles FOR.
X$	Espacio en negro para rótulos.

El programa

El programa consta de las ocho zonas básicas que se relacionan en la tabla «Estructura del programa». Las primeras líneas de instrucción están destinadas a exponer las normas que rigen el juego; un recordatorio que nunca está de más para los olvidadizos. Tras los preámbulos «educativos», empieza el cuerpo del programa a partir de la línea 230.

En la línea 250 se realizan algunas asignaciones que puede modificar el usuario a voluntad. Por ejemplo, la definición del color de los ladrillos (asignación de la variable C), que puede ser alterada sin más que utilizar cualquier otro de los parámetros de color indicados en la línea 240. Observe que en el listado adjunto se opta por el color naranja (C=112). También es admisible una modificación en el número de pelotas disponibles en cada partida; una simple sustitución del valor 10, asignado a la variable PA, bastará para satisfacer sus deseos de incrementar o reducir la reserva de pelotas. Retocando los valores que intervienen en las funciones STRING$ de las líneas 330 y 340, puede cambiar la anchura de la franja libre que separa a ambos muros en las dos primeras pantallas.

Una advertencia, destinada a los aficionados al retoque de programas, es que no resulta conveniente poner más filas de ladrillos por debajo de la última ya existente. La razón es que el programa está construido para «detectar» si hay ladrillos por encima de la línea 20. Este extremo se comprueba al analizar la línea de programa, en donde interviene la variable B, representativa de la posición de la pelota, variable que puede adoptar valores comprendidos entre 2 y 30.

Y una observación final: los caracteres que aparecen en «minúsculas» dentro del listado, corresponden a la representación en mayúsculas y vídeo inverso (en negativo) de los caracteres en cuestión. Así, pues, no se trata de caracteres de control, sino simplemente, de literales que hay que introducir en mayúsculas y en vídeo inverso.

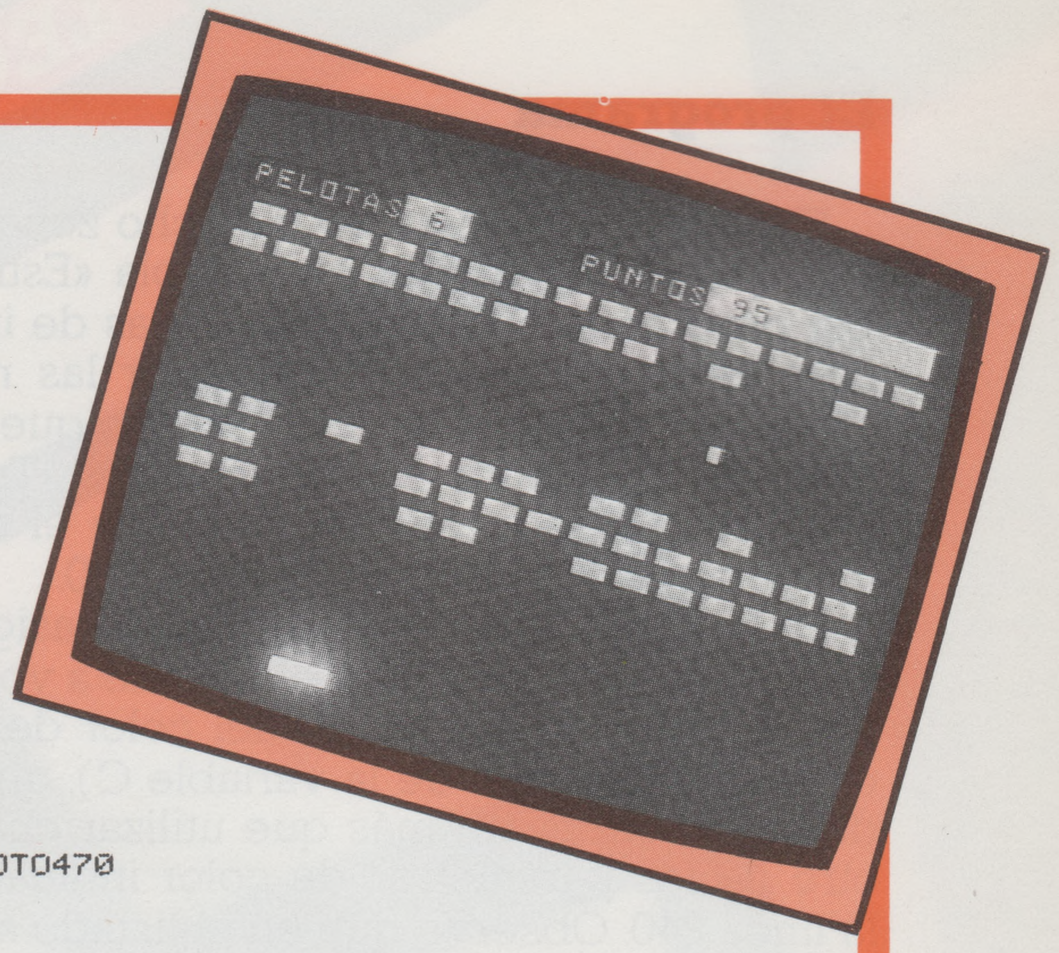
¡Suerte... y al ladrillo!

ESTRUCTURA DEL PROGRAMA

10- 220	Instrucciones relativas al juego.
230- 450	Dibujo de la muralla y cálculo de los ladrillos por pantalla.
460- 530	Zona principal del programa. Cálculo y desplazamiento de la pelota a la nueva posición.
540- 750	Subrutina para mover la raqueta.
760- 820	Subrutina para golpear la pelota.
830- 940	Comentarios finales y conclusión del programa.
950-1030	Subrutina para borrar ladrillos.
1040-1080	Paso de una a otra muralla.

LISTADO

```
10 REM **muralla**dragon(32-64)*******david bernabe alvaro******
20 X$=CHR$(128):PA=1
30 CLS6
40 PRINT@32+6,X$;"juego";X$;"de";X$;"la";X$;"muralla";X$;
50 PRINT@32*5,"VAS A JUGAR CON EL JOYSTK?(S/N)";
60 F$=INKEY$:IF F$="" GOTO60
70 IF F$<>"S" AND F$<>"N" THEN SOUND13,3:GOTO60
80 IF F$="N" GOTO120
90 PRINT@32*5,"":PRINT@32*3,"UTILIZA EL JOYSTK DERECHO PARA  MOVER LA RAQUETA Y
PARA IR MAS  RAPIDO PULSA EL BOTON DEL MANDO";
100 PRINT@32*14,"PULSA 'enter' PARA CONTINUAR";
110 C$=INKEY$:IF C$<>CHR$(13) THEN 110 ELSE140
120 PRINT@32*5,"":PRINT@32*3,"UTILIZA LA TECLA 'u' PARA MOVER LA RAQUETA DE DERE
CHA A IZQUIER-DA, Y LA 'i' PARA IR DE IZQUIER-DA A DERECHA. PULSA 'shift' (Y  LA
 TECLA PARA IR MAS DEPRISA)";
130 GOTO100
140 CLS7:PRINT@32,"AL EMPEZAR EL JUEGO DISPONES DE 10 PELOTAS QUE IRAS PERDIENDO
 ALNO GOLPEARLAS CON LA RAQUETA    CUANDO LLEGUEN ESTAS A LA PARTE INFERIOR DE L
A PANTALLA"
150 PRINT@7*32,"DEBERAS ROMPER LOS LADRILLOS DE CADA PANTALLA PARA OBTENER MAS
PUNTOS Y PASAR A OTRAS PANTALLAS(EN LA PRIMERA PANTALLA HAY 80  LADRILLOS,EN LA
SEGUNDA 112,Y ENLAS SIGUIENTES 144)"
160 PRINT@32*14,"PULSA 'enter' PARA CONTINUAR";
170 C$=INKEY$:IF C$<>CHR$(13) GOTO170
180 CLS7
190 PRINT@32,"CUANTAS MAS PANTALLAS CONSIGAS  HACER MAYOR SERA EL NUMERO DE   LA
DRILLOS QUE DEBERAS ROMPER ASIEN LA PRIMERA PANTALLA TERMINA- RAS CUANDO TE FALT
EN 7 LADRILLOS,EN LA SEGUNDA A 5, EN LA TERCE-RA A 3,EN LA CUARTA A 1 Y EN EL RE
STO A 0.";
200 PRINT@10*32,"EN LA PARTE SUPERIOR TIENES DOS INDICADORES QUE TE INFORMAN DE
 LAS PELOTAS QUE TE QUEDAN Y LOS PUNTOS OBTENIDOS"
210 PRINT@32*15,"PULSA 'enter' PARA EMPEZAR";
220 A$=INKEY$:IF A$<>CHR$(13) GOTO220
230 CLS0
240 REM C=COLOR=16-AMARILLO;      AZUL-32;48-ROJO;64-BEIGE;       80-TURQUESA;96-
MAGENTA;          112-NARANJA
250 C=112 :J=144:PE=9
260 A$=CHR$(131+C)+CHR$(130+C)
```



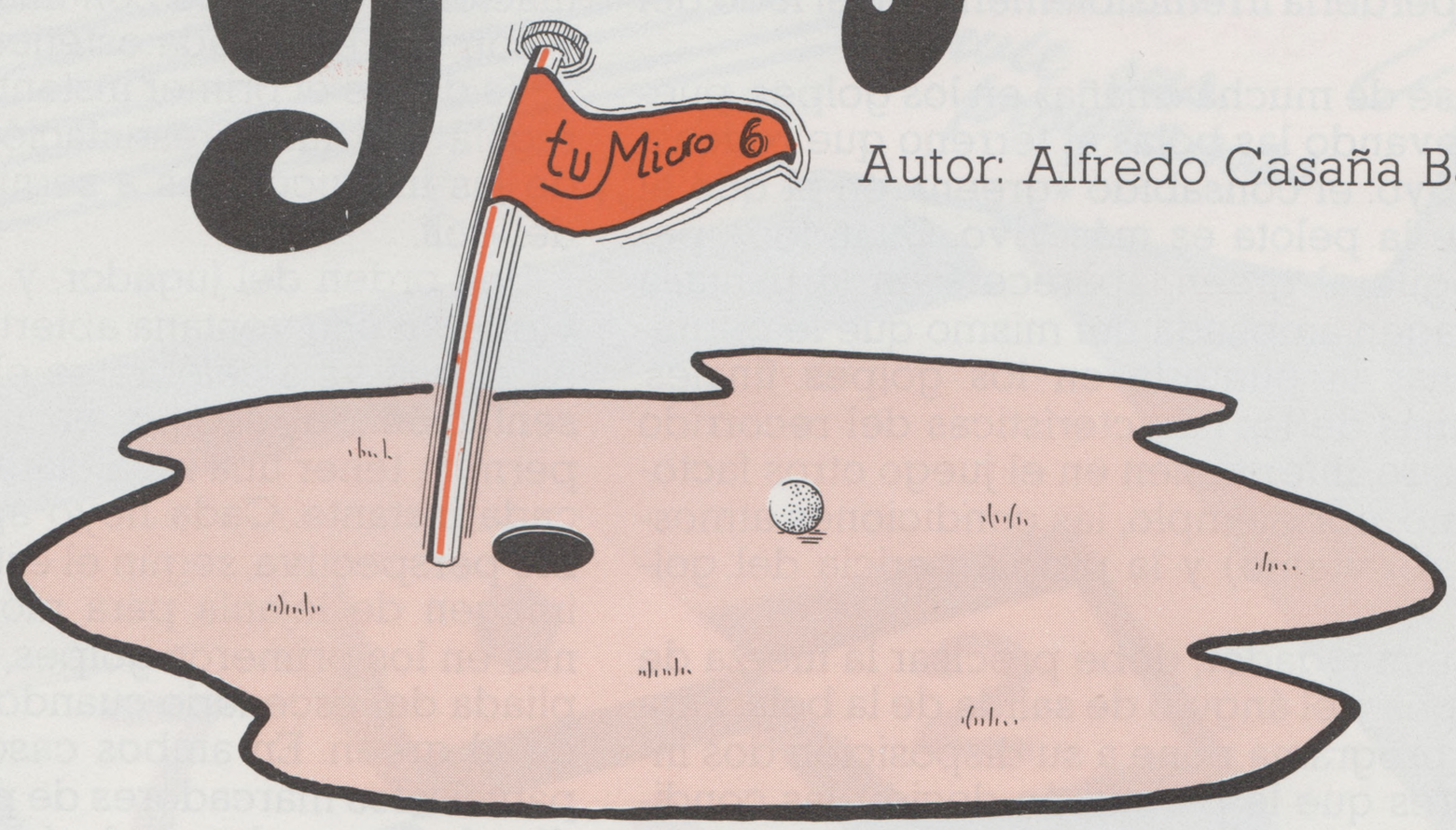

```
270 B$=CHR$(129+C)+CHR$(131+C)
280 FORP=1TO5
290 FORX=1TO16:PRINTA$;:NEXT
300 FORX=1TO16:PRINTB$;:NEXT
310 NEXT
320 PRINT@0,"pelotas";PE;STRING$(4,X$);"puntos";O
330 IF PA=1 THEN J=80:PRINT@96,STRING$(128,X$);
340 IF PA=2 THEN J=112:PRINT@128,STRING$(64,X$);
350 JJ=JJ+PA*5*J:IF PA=1 THEN U=35: GOTO390
360 IF PA=2 THEN U=50:GOTO390
370 IF PA=3 THEN U=45: GOTO390
380 IF PA=4 THEN U=20 ELSE U=0
390 K=RND(7):IF K=1 THEN A1=2:GOTO450
400 IF K=2 THEN A1=3:GOTO450
410 IF K=3 THEN A1=6:GOTO450
420 IF K=4 THEN A1=7:GOTO450
430 IF K=5 THEN A1=10:GOTO450
440 IF K=6 THEN A1=11 ELSE A1=14
450 :B1=30:P=2:P2=-2
460 A=A1+P:IFA>63ORA<0THENP=-P:SOUND189,1:GOTO460
470 B=B1+P2:IFB<1 OR B>31 THENP2=-P2:SOUND189,1:GOTO470
480 GOSUB550
490 IF B<20 THEN GOSUB950
500 IF FF=1 THEN FF=0 :GOTO460
510 SET(A,B,0):RESET(A1,B1):A1=A:B1=B
520 IF B=28 ANDP2<0THEN FORX=0 TO 4:SET(F+X,30,5):NEXT
530 GOTO460
540 REM SUBRUTINA PARA MOVER LA RAQUETA
550 IF F$="S"GOTO620
560 DD=PEEK(337):IF DD<>247 AND DD<>239 AND DD<>183 AND DD<>175 THEN 740
570 POKE135,0
580 IF DD=239 THEN E=-2:GOTO680
590 IF DD=247 THEN E=2:GOTO680
600 IF DD=175 THEN E=-8:GOTO680
610 E=8:GOTO680
620 E=JOYSTK(1):E=JOYSTK(0)
630 IF E<4 THEN E=-1:GOTO660
640 IF E>60 THEN E=1:GOTO660
650 E=0
660 I=PEEK(65280):IF I=254 OR I=252 OR I=126 OR I=124 THEN E=E*4
670 E=E*2
680 F=F+E:IFF<0THENF=0
690 IF F>59 THEN F=59
700 IF F =F1 THEN 740
710 FORX=0TO4:RESET(F1+X,30):NEXT
720 FORX=0TO4:SET(F+X,30,5):NEXT
730 F1=F
740 IF B=30 THEN GOSUB760
750 RETURN
760 REM SUBRUTINA PARA GOLPEAR  LA PELOTA
770 IF POINT(A,30)=5THENP2=-P2:SOUND189,2 ELSE 830
780 IF A=63GOTO810
790 IF POINT(A+1,30)=0 THEN P=-P
800 IF A=0THEN RETURN
810 IF POINT(A-1,30)=0 THEN P=-P
820 RETURN
830 PE=PE-1:IF PE>-1 THEN SOUND15,3:PRINT@0,"pelotas";PE;STRING$(5,X$);"puntos";
O:RETURN
840 CLS8:PRINT@11,X$;"perdiste";X$;:SOUND1,5
850 IF O<200 THEN PRINT@32*3,"ERES UNA VERDADERA CALAMIDAD TE HAS HECHO SOLAMENT
E";O;"PUNTOS";:PRINT@6*32,"NO HAS PASADO DE LA PRIMERA PAN-TALLA Y ENCIMA TE HAS
 DEJADO";J-O/5;"LADRILLOS";:GOTO910
860 IFO<365 THENPRINT@32*3,"SIGUES SIN PROSPERAR DEMASIADO  TE HAS DEJADO";J-O/5
;"LADRILLOS Y     HAS OBTENIDO";O;"PUNTOS";:GOTO910
870 IF O<720 THENPRINT@32*3,"NO ESTA MAL MUCHACHO PERO AUN   TE FALTAN";J-(O-365
)/10;"LADRILLOS PARA      PASAR LA SEGUNDA PANTALLA";:PRINT@32*8,"TU PUNTUACION
HA SIDO DE";O;"   PUNTOS";:GOTO910
880 IF O<900 THEN PRINT@32*3,"MUY BIEN TE HAN FALTADO";J-(O-365)/10;"    LADRILL
OS PARA TERMINAR LA      SEGUNDA PATALLA"::PRINT@8*32,"TU PUNTUACION ES:";O;"PUN
TOS";:GOTO910
890 IF O<2160 THEN PRINT@32*3,"FENOMENO,TE QUEDA POCO PARA SER CAPAZ DE ROMPER:
                'la";X$;"gran";X$;"muralla";X$;"china'":PRINT@32*8,"HAS OBTEN
IDO";O;"PUNTOS":GOTO910
900 PRINT@32*3,"PERFECTO,ERES UN GRAN JUGADOR   HAS OBTENIDO";O;"PUNTOS";
910 PRINT@32*12,"QUIERES VOLVER A JUGAR?(S/N)";
920 A$=INKEY$:IF A$=""GOTO920
930 IF A$<>"S" AND A$<>"N" THEN SOUND1,4:GOTO920
940 IF A$="S" THEN RUN ELSE CLS:END
950 REM SUBRUTINA PARA BORRAR    LADRILLOS
960 H=1024+INT(A/2)-1+INT(B/2)*32
970 IF INT(B/4)=B/4 THEN HH=H+1:GOTO990
980 HH=H
990 G=PEEK(HH):IF G=128 THEN RETURN
1000 O=O+PA*5:IFO =JJ-U THEN 1040
1010 PRINT@0,"pelotas";PE;STRING$(5,X$);"puntos";O
1020 SOUND200,1
1030 POKEHH,128:POKEHH+1,128:FF=1:P2=-P2:RETURN
1040 CLS1
1050 PRINT@32+2,X$;"felicidades";X$;"lo";X$;"conseguiste";X$
1060 PRINT@32*3+8,"ERES FANTASTICO";
1070 PA=PA+1
1080 PRINT@32*5,"TIENES";O;"PUNTOS Y TE QUEDAN";PE;"PELOTAS";:PRINT:PRINT"A VER
HASTA DONDE LLEGAS EN LA  SIGUIENTE PANTALLA":FORX=1 TO 6000:NEXT:CLS0:GOTO260
```

END

VENCEDOR DE LA PRIMERA EDICION DEL CONCURSO

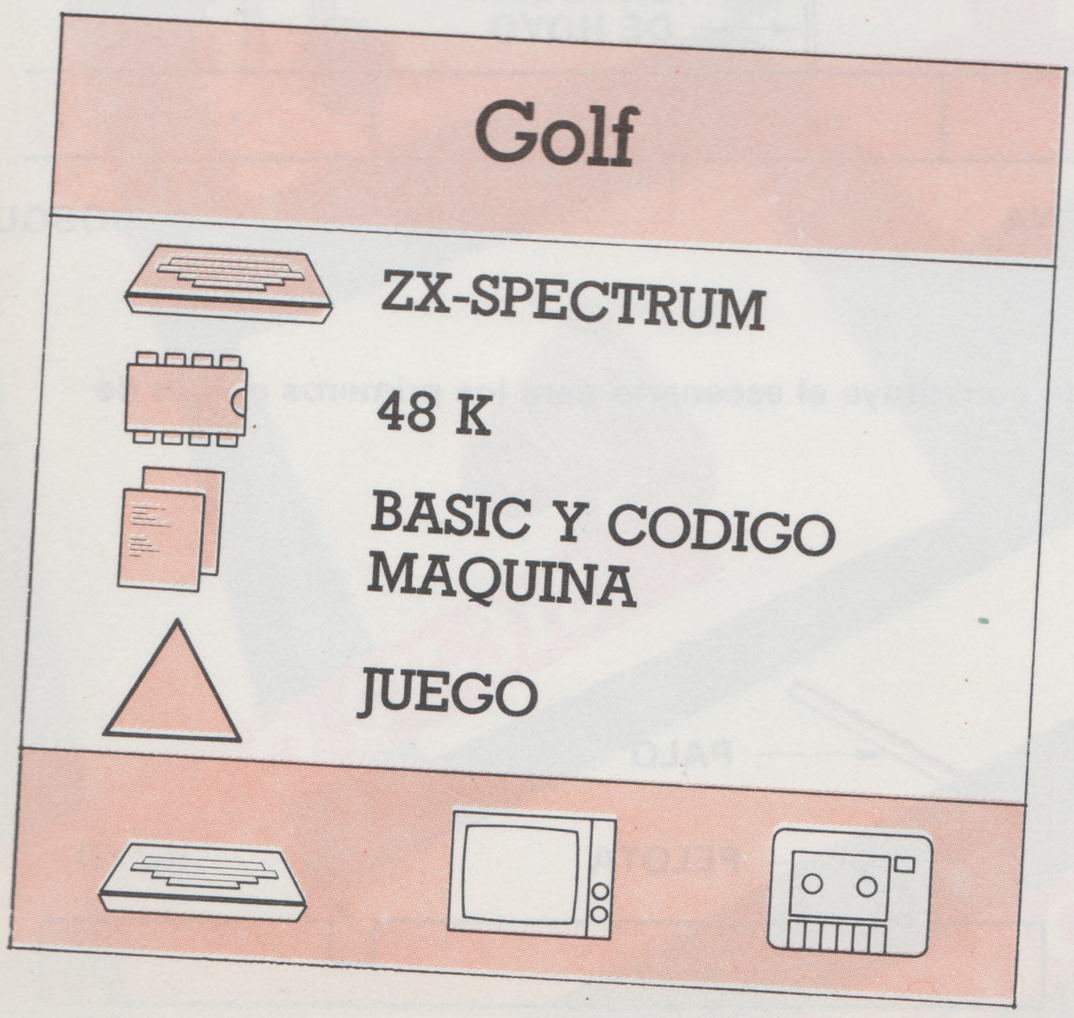
EL PROGRAMA DEL LECTOR

golf



Autor: Alfredo Casaña Bayona

Un digno y contundente vencedor de la primera edición de nuestro concurso «El programa del lector». Los lectores que posean un SPECTRUM tendrán ocasión de disfrutar con la ejecución de este programa; un programa cuidado hasta el más mínimo detalle y que evidencia un esfuerzo más que destacable por parte de su autor.



El programa reproduce en la pantalla todas y cada una de las circunstancias que rodean a un partido de Golf, con una verosimilitud nada frecuente en los programas de juego.

Las instrucciones, extraídas del más estricto reglamento golfista, aparecen sobre la pantalla a solicitud del usuario. Los más ajenos a las hazañas del popular «Seve», van a tener que familiarizarse con unas reglas de juego muy distintas de las que imperan en el mundo del balón.

Las reglas del golf

Un partido de golf viene a ser algo semejante a un entretenido paseo campestre. Un recorrido en el que el jugador se tropezará con un total de 18 hoyos, en los que debe introducir

la bola a base de sucesivos golpes de «stick». Desde luego que la cosa no es tan simple como ir paseando al tiempo que se atiza a la bola con el bastón. Son varios los obstáculos y dificultades que hay que sortear.

Para empezar, y como es de ley, el terreno está salpicado de árboles. Si la suerte les es favorable, la bola pasará a través de sus ramas; no obstante, si el jugador anda con el biorritmo bajo, es muy posible que ésta se empeñe en rebotar en el tronco o, simplemente, que se pierda entre el follaje.

De nuevo, tal como ocurre en la realidad, hay algún que otro banco de arena en el lugar más inapropiado; su efecto es reducir el bote de las bolas y amortiguar la fuerza de los golpes que se dan desde esta zona. Y, cómo no, también hay charcos de agua, dispuestos a recibir la bola y perderla irremisiblemente en el lodo del fondo.

A base de mucha «maña» en los golpes, puede ir llevando las bolas al terreno que rodea a cada hoyo: el consabido «green», en el que el bote de la pelota es más vivo. Cuando la pelota llegue al green, aparecerá en la pantalla una imagen ampliada del mismo que le permitirá afinar la puntería en los golpes finales.

Además de las características del recorrido campestre, intervienen en el juego otros factores como, por ejemplo, las condiciones atmosféricas (el viento) y la propia pericia del golfista.

Un buen jugador, debe precisar la fuerza de los golpes y el ángulo de salida de la bola. Para ello, el programa pone a su disposición dos indicadores que le permitirán decidir las condiciones óptimas del disparo. Así, pues, antes de atizar el golpe a la bola, hay que seleccionar la fuerza, accionando la tecla ENTER cuando el correspondiente indicador se encuentre en el nivel adecuado. Otro tanto habrá que hacer, a continuación, para determinar el ángulo de salida: de nuevo se pulsará ENTER cuando el nivel sea el apetecido... ¡Pero cuidado! no es conveniente disparar sin antes echar una ojeada al banderín del green; recuerde que el viento puede llevar la bola hacia el lugar menos indicado. Compruebe antes cuál es la dirección y fuerza del viento (la figura adjunta muestra cuál es la posición del banderín en cada caso), y en el instante propicio, pulse de nuevo la tecla ENTER para ordenar el golpe.

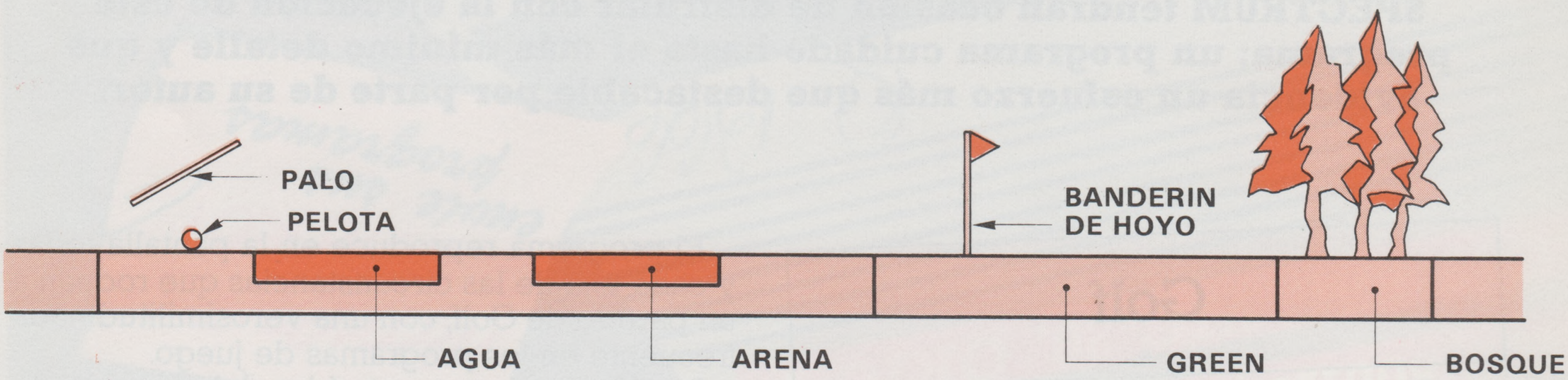
El «Open» TU MICRO

El escenario del partido que aparece sobre la pantalla, refleja todas y cada una de las situaciones del juego con una notable precisión y con una esmerada estética.

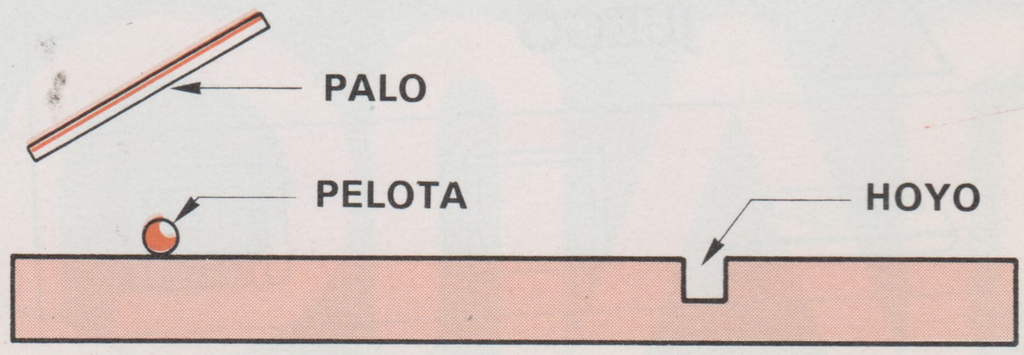
Ya desde el primer instante, el programa nos regala con una presentación, clara y concisa, de las instrucciones a seguir para la práctica del golf.

Una orden del jugador, y la pantalla se convierte en una ventana abierta al campo de golf en el que va a celebrarse el partido. La representación del campo, en una vista de perfil, permite tener una idea clara de la situación en cada instante. Cada hoyo aparece en una doble perspectiva, según el estado del juego: una imagen de lejanía para mostrar las evoluciones en los primeros golpes, y una imagen ampliada del escenario cuando la pelota ha llegado al green. En ambos casos, aparecen en la pantalla los marcadores de puntuación y los indicadores que le ayudarán a precisar los golpes.

Arboles, bancos de arena, lagunas, banderines... todos los ingredientes que dan cuerpo a

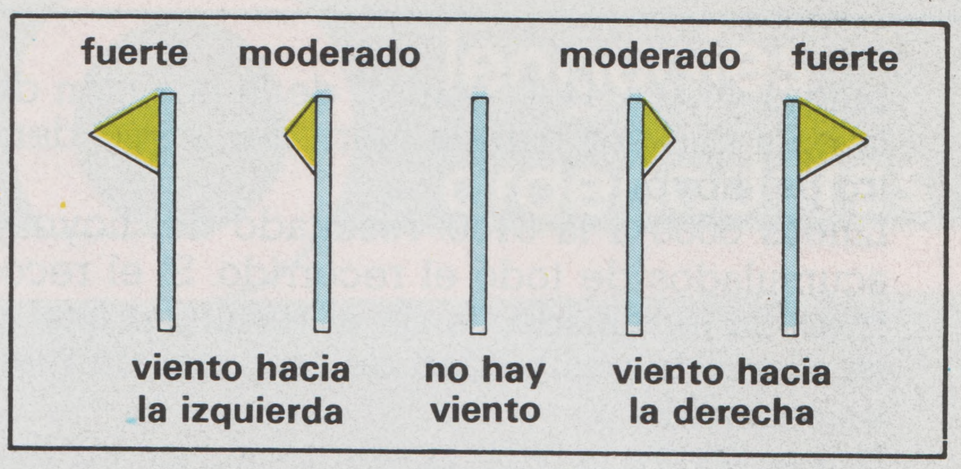


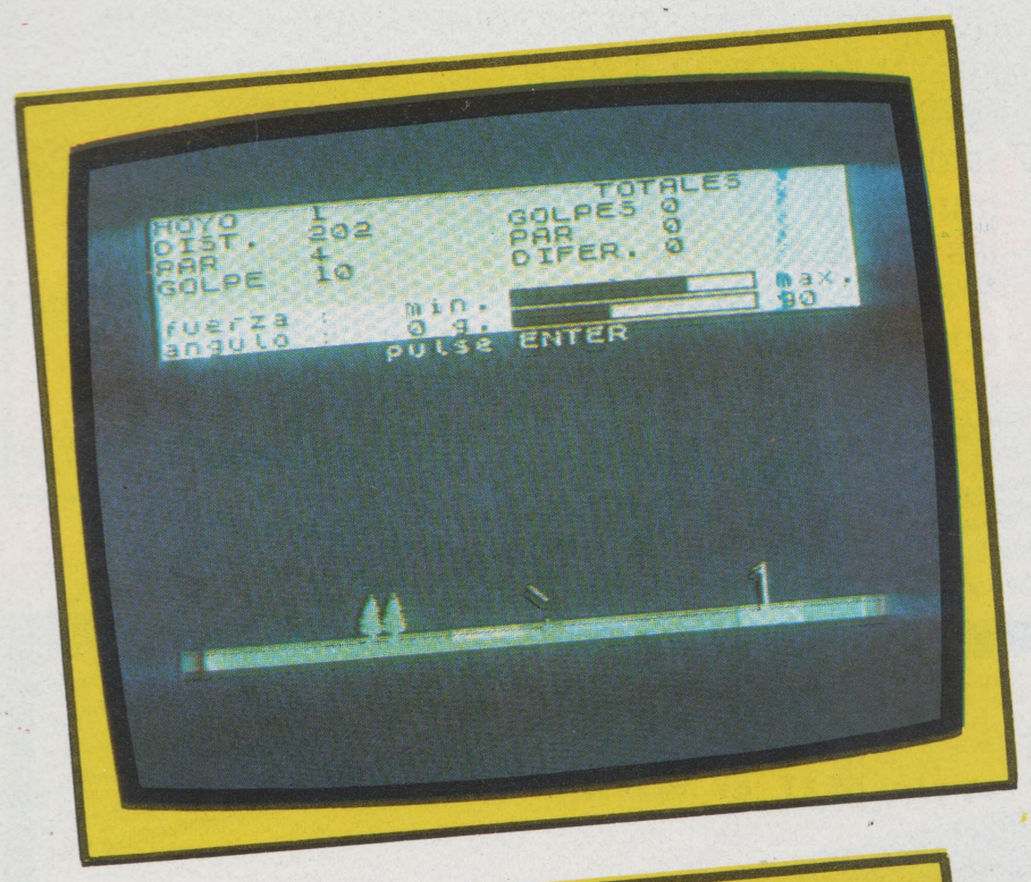
Representación del campo de folf en perspectiva lejana. Este constituye el escenario para los primeros golpes de aproximación al «green».

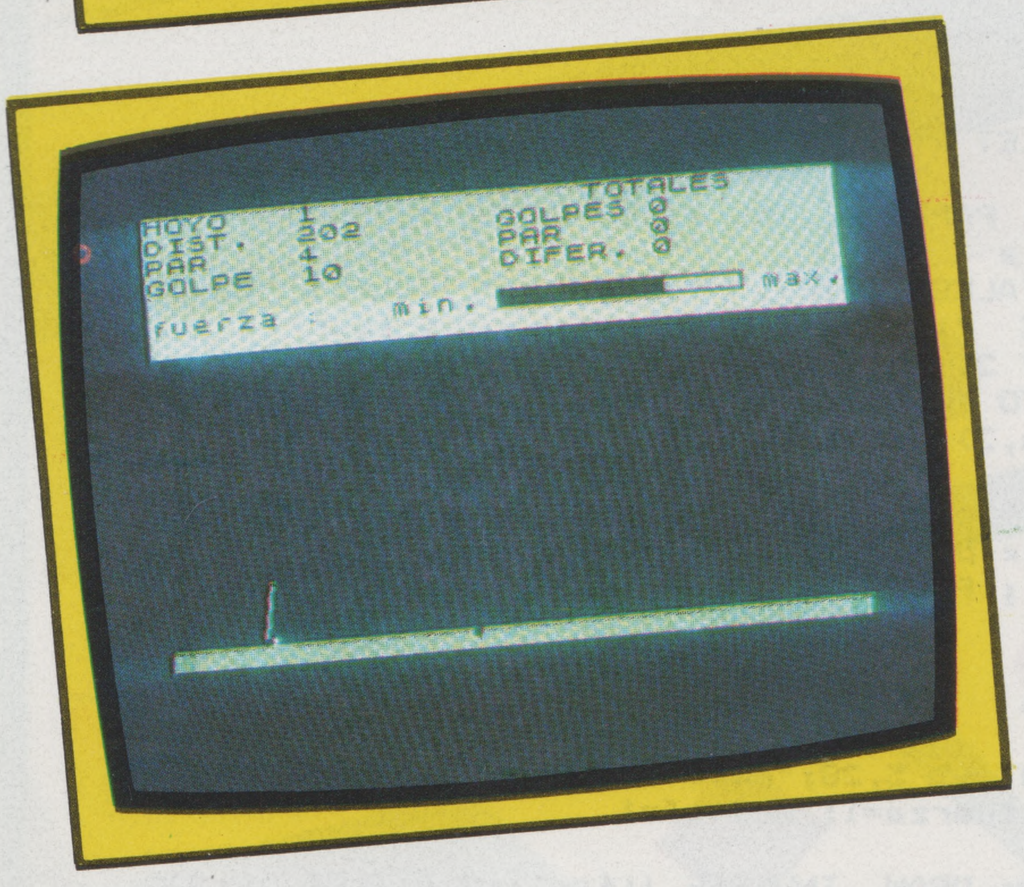


Vista de la zona próxima al hoyo que aparece en la pantalla al entrar la pelota en el «green». Al jugar en el green, sólo interviene el indicador de fuerza de la pelota.

Indicaciones de dirección y fuerza del viento aportadas por el banderín de hoyo.







un partido de golf y que condicionan su desarrollo, aparecen representados en la pantalla del televisor.

Detalles reseñables los encontramos en cualquier punto de la ejecución. Por ejemplo, la trayectoria de la bola es perfectamente visible: el espectador ve cómo golpea el stick a la bola y cómo se eleva hasta caer sobre el terreno. Y no termina aquí el detalle: para dar una pincelada de realidad de la caída de la bola, ésta va rebotando en el suelo, a saltos cada vez más cortos, hasta que se detiene.

También hay que mencionar la originalidad de los indicadores para la elección de las condiciones del golpe. Tanto la fuerza del golpe como el ángulo de subida de la bola, se representan en marcadores de tipo analógico; la indicación la proporciona una barra, similar a la de un termómetro, que oscila entre los niveles máximo y mínimo. Una vez con la pelota en el green, sólo interviene el marcador de fuerza, puesto que, tal como sucede en la realidad, los golpes son rasantes.

Y terminemos las loanzas con una alusión al banderín, cuyas indicaciones sólo entran en consideración durante los golpes largos, fuera del green. La posición de la banderola sujeta al mástil revela, en todo momento, la dirección y fuerza del viento.

tuMicro

Bytes en el agujero

A pesar de la aparente complejidad del programa, su análisis puede acometerse con una relativa comodidad de la mano de las sentencias REM.

Estas delimitan cada una de las secciones de que consta.

Un viaje a través del listado nos revela los siguientes bloques:

Líneas 5 a la 20: inicialización de variables y carga de los gráficos redefinidos en los DATA localizados a partir de la línea 9000.

Líneas 500 a 1050: lectura de los datos del hoyo a preparar y presentación en pantalla del diseño del mismo.

Líneas 1500 a la 1550: presentación del indicador de fuerza de la bola. En el instante en el que se pulse la tecla ENTER, queda fijado el valor de la fuerza sin posibilidad de corrección.

Líneas 1800 a 1840: selección del ángulo de salida de la bola.

Líneas 2000 a la 2150: creación de un efecto de viento aleatorio que refleja el banderín. Al pulsar la tecla ENTER se fija el valor del viento que se mantendrá constante a lo largo de toda la trayectoria de la bola.

Líneas 2500 a 2520: cálculo de los valores iniciales de velocidad vertical y horizontal de la bola.

Líneas 2600 a 2700: cálculo y visualización de la trayectoria de la bola, controlando el rebote según el terreno y verificando los márgenes.

Líneas 3000 a la 3030: control de la situación de la bola una vez parada, según haya caído en el green, en la arena, en el hoyo...

Líneas 4000 a la 4100: representación ampliada del green sobre la pantalla al caer la bola en el mismo.

Líneas 4100 a la 4140: petición de la fuerza del golpe en el interior del green.

Líneas 4200 a 4240: cálculo de la velocidad inicial de la bola en el green.

Líneas 4300 a 4360: cálculo y visualización de la trayectoria de la bola en el green.

Líneas 4400 a 4430: control de la posición de la bola una vez parada fuera del green, dentro del hoyo...
Líneas 5000 a la 5130: resultado del hoyo y acumulados de todo el recorrido. Si el recorrido ha terminado, se da el resultado final y se ofrece la posibilidad de empezar un nuevo recorrido.
Líneas 6000 a la 6280: presentación de las instrucciones del juego.
Líneas 8000 a 8080: subrutinas y efectos de sonido.
Línea 9000: DATA con los datos generales y con la definición de gráficos.
Líneas 9010 a la 9180: características de los 18 hoyos, uno por línea. Cada DATA contiene la posición de green, posición del primer y segundo bancos de arena, posición de la trampa de agua, distancia al hoyo y par del hoyo. La modificación de estos datos permite variar fácilmente las características del hoyo. En el caso de poner el valor O, se omite el obstáculo correspondiente.
Tal como ya es costumbre en los programas que publicamos, los caracteres que aparecen subrayados en el listado (por ejemplo, en las líneas 500, 1020, 1050, etc.), corresponden a caracteres gráficos; en consecuencia, habrá que introducir los caracteres gráficos asociados a las respectivas teclas.
Y esto es todo; una relación sumarial, aunque completa, de la estructura de todo un vencedor de nuestro primer concurso de programación «El programa del lector.»

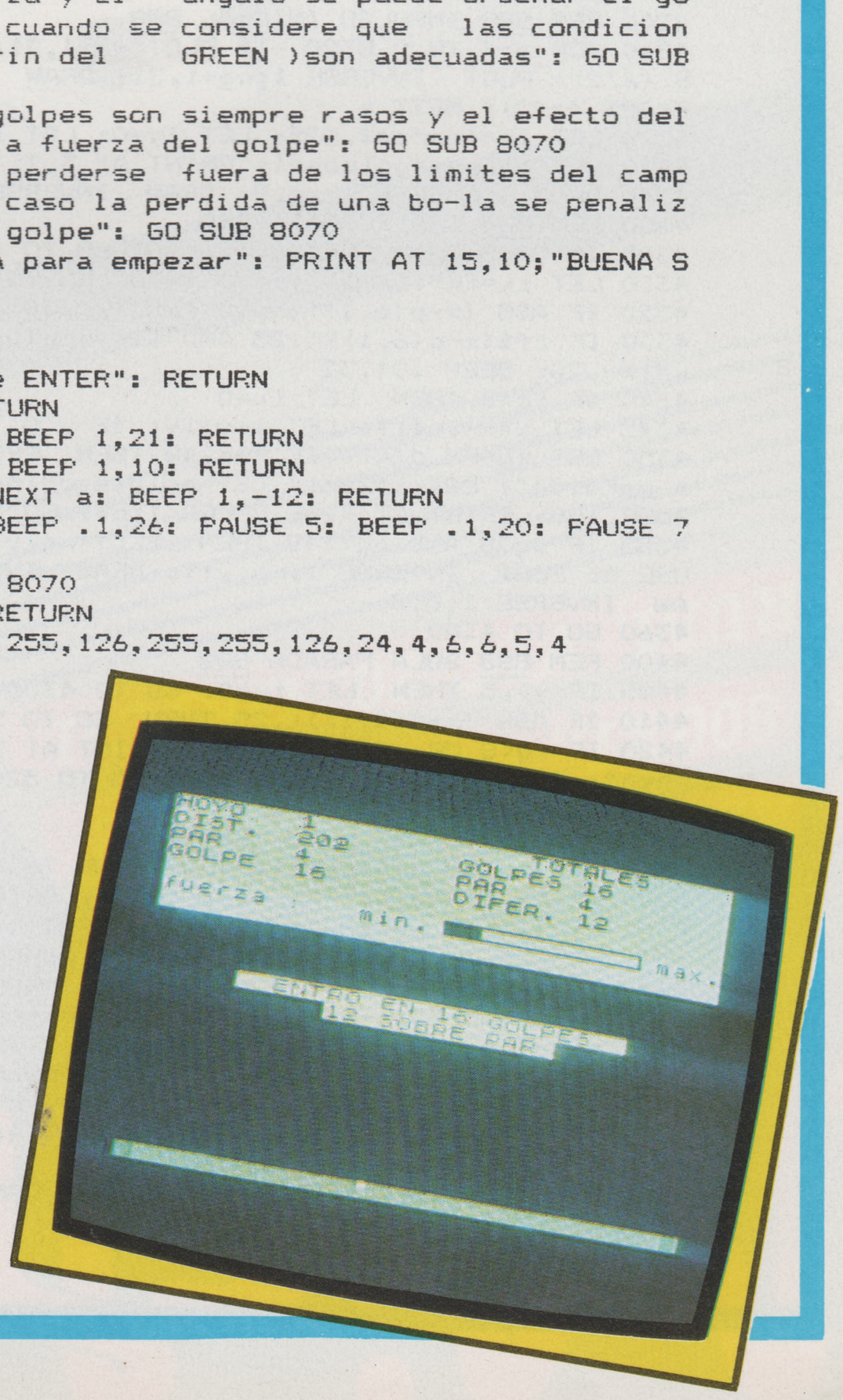
tuMicro

LISTADO

```
  5 DIM p(7,2): RESTORE : LET b$="
 10 FOR a=0 TO 15: READ a1: POKE USR "A"+a,a1: NEXT a: FOR a=1 TO 5: READ p(a,2
): NEXT a
 15 GO TO 6000
 20 LET tgolpe=0: LET tpar=0: LET hoyo=1
500 REM 888 INICIO HOYO 888
510 FOR a=1 TO 7: READ p(a,1): NEXT a: LET x=8: LET y=0: LET xant=8: LET golpes
=0: LET arena=0
520 PAPER 1: INK 7: CLS : PRINT AT 21,0; PAPER 4; INK 4;b$: PAPER 6: INK 0
530 FOR a=0 TO 4: PRINT AT a,0;b$: NEXT a
540 PRINT AT 0,0;"HOYO   ";hoyo,"    TOTALES","DIST.  ";p(6,1)-8,"GOLPES ";tgol
pe,"PAR    ";p(7,1),"PAR    ";tpar,"GOLPE  ";golpes,"DIFER. ";tgolpe-tpar
550 PRINT AT 21,0; PAPER 2;" ": PRINT AT 21,31; PAPER 2;" "
1010 FOR a=1 TO 5: IF p(a,1)=0 THEN  GO TO 1030
1020 PRINT AT 21,p(a,1); PAPER 4; INK p(a,2); BRIGHT (a=1);"333"
1030 NEXT a
1040 PAPER 1: INK 7: PLOT x,8: PRINT AT 19,p(6,1)/8-1;"   ": PLOT p(6,1),23: DRA
W 0,-15: PLOT x,16: DRAW 5*(x>p(6,1))-5*(x<p(6,1)),5
1050 IF p(5,1)>0 THEN  PRINT AT 19,p(5,1); PAPER 1; INK 4;"AA": PRINT AT 20,p(5,
1); PAPER 1; INK 4;"BB"
1500 REM 888 DAR FUERZA 888
1505 FOR a=1 TO 25: LET o$=INKEY$: NEXT a
1510 PAPER 7: INK 0: PRINT AT 5,0;b$;b$
1520 PRINT AT 5,0;"fuerza :   min.": PRINT AT 5,28;"max.": PLOT 126,129: DRAW 90
,0: DRAW 0,6: DRAW -90,0: DRAW 0,-6: LET fuerza=1: LET if=1
1525 GO SUB 8010
1530 PLOT  INVERSE (if=-1);fuerza+126,130: DRAW  INVERSE (if=-1);0,4: LET o$=INK
EY$: IF o$=CHR$ (13) THEN  GO SUB 8020: GO TO 1800
1540 LET fuerza=fuerza+if: IF fuerza=89 OR fuerza=1 THEN  LET if=if*-1
1550 GO TO 1530
1800 REM 888 DAR ANGULO 888
1805 FOR a=1 TO 25: LET o$=INKEY$: NEXT a
1810 PRINT AT 6,0;"angulo :   0 g.": PRINT AT 6,28;"90": PLOT 126,121: DRAW 90,0
: DRAW 0,6: DRAW -90,0: DRAW 0,-6: LET angulo=1: LET if=1
1815 GO SUB 8010
1820 PLOT  INVERSE (if=-1);angulo+126,122: DRAW  INVERSE (if=-1);0,4: LET o$=INK
EY$: IF o$=CHR$ (13) THEN  GO SUB 8020: GO TO 2000
1830 LET angulo=angulo+if: IF angulo=89 OR angulo=1 THEN  LET if=if*-1
1840 GO TO 1820
2000 REM 888 DAR GOLPE 888
2010 FOR a=1 TO 25: LET o$=INKEY$: NEXT a: LET viento=0: PAPER 8: INK 8
2015 GO SUB 8010
2020 LET v=INT (RND*21)-10: LET viento=viento-1*(v<viento AND viento>-7)+1*(v>vi
ento AND viento<7): PLOT p(6,1),23: DRAW viento,-2: DRAW -viento,-2: DRAW 0,4
2030 PAUSE 2: LET o$=INKEY$: IF o$=CHR$ (13) THEN  GO SUB 8020: GO TO 2100
2040 PLOT p(6,1),23: DRAW  INVERSE 1;viento,-2: DRAW  INVERSE 1;-viento,-2: GO T
O 2020
2100 IF x<p(6,1) THEN  LET a1=-120: LET a2=120: LET a3=30
2110 IF x>p(6,1) THEN  LET a1=120: LET a2=-120: LET a3=-30
2120 PLOT x,16: DRAW  INVERSE 1;5*a1/120,5
2130 FOR a=a1 TO a2 STEP a3: LET px=8*SIN (a/180*PI): LET py=-8*COS (a/180*PI)
2140 PLOT x,16: DRAW px,py: PLOT  INVERSE 1;x,16: DRAW  INVERSE 1;px,py: NEXT a
2150 LET golpes=golpes+1: PRINT AT 3,7; PAPER 6; INK 0;golpes
2500 REM 888 IMPULSO INICIAL 888
2510 LET ix=fuerza*COS (angulo/180*PI)*.15: LET iy=fuerza*SIN (angulo/180*PI)*.1
5: IF x>p(6,1) THEN  LET ix=ix*-1
2520 LET xant=x: IF arena=1 THEN  LET ix=ix*.4: LET iy=iy*.8
2600 REM 888 TRAYECTORIA 888
2610 LET x=x+ix+viento*ABS (y)*.015: LET y=y+iy: IF y<0 THEN  LET y=0
```

```
2613 IF POINT (x,y+8)=1 AND x/8>p(5,1) AND x/8<p(5,1)+2 THEN  LET ix=-ix*.4
2615 IF x<8 OR x>248 THEN  LET x=xant: LET y=0: PRINT AT 7,10; FLASH 1;"bola per
dida": LET golpes=golpes+1: PRINT AT 3,7; PAPER 6; INK 0;golpes: GO SUB 8050: PA
USE 300: PRINT AT 7,0;b$: GO TO 1040
2620 PLOT x,y+8: PAUSE 8: PLOT  INVERSE 1;x,y+8: IF y>0 THEN  LET iy=iy-2: GO TO
 2600
2630 LET car=INT (x/8): LET vcar=ATTR (21,car)
2640 IF y=0 AND ABS (iy)<3 AND ABS (ix)<.5 THEN  GO TO 3000
2660 IF vcar>=64 THEN  LET iy=iy*-.5: LET ix=ix*.5: BEEP .01,36: GO TO 2600
2670 IF vcar=36 THEN  LET iy=iy*-.4: LET ix=ix*.4: BEEP .01,36: GO TO 2600
2680 IF vcar=38 THEN  LET iy=iy*-.25: LET ix=ix*.25: BEEP .01,0: GO TO 2600
2690 IF vcar=37 THEN  INK 5: PLOT x,y+8: DRAW -3,3: PLOT x,y+8: DRAW 0,3: PLOT x
,y+8: DRAW 3,3: BEEP .2,-12
2700 IF vcar=37 THEN  PLOT x,y+8: DRAW  INVERSE 1;-3,3: PLOT x,y+8: DRAW  INVERS
E 1;0,3: PLOT  INVERSE 1;x,y+8: DRAW  INVERSE 1;3,3: LET x=0: GO TO 2615
3000 REM 888 BOLA PARADA 888
3005 IF vcar=38 THEN  LET arena=1: PRINT AT 8,8; PAPER 6; INK 0;"bola en la aren
a"
3006 IF vcar<>38 THEN  LET arena=0: PRINT AT 8,0; PAPER 1;b$
3008 IF ABS (x-p(6,1))<.25 THEN  GO TO 5020
3010 IF vcar>=64 THEN  PLOT x,y+8: GO TO 4000
3020 IF x/8>p(5,1) AND x/8<p(5,1)+2 THEN  LET x=0: GO TO 2615
3030 GO TO 1040
4000 REM 888 BOLA EN GREEN 888
4010 PRINT AT 7,10; PAPER 2; INK 6;"  EN GREEN  ": GO SUB 8030: PAUSE 100: PRINT
 AT 7,0; PAPER 1; INK 7;b$: PRINT AT 19,0; INK 7;b$;b$
4015 PRINT  PAPER 4; INK 1; BRIGHT 1;b$: PRINT AT 21,0; PAPER 4;" ": PRINT AT 21
,31; PAPER 4;" "
4020 LET xg=FN d(p(6,1)): FOR a=xg TO xg+2: PLOT a,7: PLOT a,6: PLOT a,5: NEXT a
4030 LET xg=FN d(x): PLOT xg,9: DRAW 2,0: PLOT xg+1,8: DRAW 0,2
4040 PRINT AT 5,0; PAPER 7; INK 0;b$;b$
4050 PLOT xg+1,31: DRAW 7*(x>p(6,1))-(7*(x<p(6,1))),-19.5: LET xant=x
4100 REM 888 DAR FUERZA 888
4105 FOR a=1 TO 25: LET o$=INKEY$: NEXT a
4110 PRINT AT 5,0;"fuerza :   min.": PRINT AT 5,28;"max.": PLOT 126,129: DRAW 90
,0: DRAW 0,6: DRAW -90,0: DRAW 0,-6: LET fuerza=1: LET if=1
4115 GO SUB 8010
4120 PLOT  INVERSE (if=-1);fuerza+126,130: DRAW  INVERSE (if=-1);0,4: LET o$=INK
EY$: IF o$=CHR$ (13) THEN  GO SUB 8020: GO TO 4200
4130 LET fuerza=fuerza+if: IF fuerza=89 OR fuerza=1 THEN  LET if=if*-1
4140 GO TO 4120
4200 REM 888 IMPULSO INICIAL 888
4210 FOR a=7 TO 0 STEP -1: PLOT xg+1,31: DRAW a*(x>p(6,1))-(a*(x<p(6,1))),-23+AB
S (a/2): PLOT  INVERSE 1;xg+1,31: DRAW  INVERSE 1;a*(x>p(6,1))-(a*(x<p(6,1))),-2
3+ABS (a/2): NEXT a
4220 LET ix=fuerza*.025: LET iy=0: LET if=1: IF x>p(6,1) THEN  LET if=-1
4230 LET golpes=golpes+1: PRINT AT 3,7; PAPER 6; INK 0;golpes
4240 PLOT  INVERSE 1;xg,9: DRAW  INVERSE 1;2,0
4300 REM 888 TRAYECTORIA 888
4305 IF y>.5 THEN  LET iy=iy-10: GO TO 4312
4310 LET ix=ix-.1: IF ix<=0 THEN  GO TO 4400
4320 IF ABS (x-p(6,1))<=.20 AND ix<.20 AND y<.5 THEN  LET ix=0: GO TO 4400
4330 IF if*(x-p(6,1))<.25 AND if*(x-p(6,1))>0 AND y<.5 THEN  LET iy=ix*20: LET i
x=ix-.30: BEEP .01,32
4340 IF ix<0 THEN  LET ix=0
4345 LET x=x+ix*if: LET y=y+iy: IF y<0 THEN  LET y=0: LET iy=0
4350 LET xg=FN d(x): IF x>=248 THEN  PRINT AT 7,10; PAPER 7; INK 0; FLASH 1;"bol
a perdida": LET x=xant: LET golpes=golpes+1: PRINT AT 5,7;golpes: GO SUB 8050: P
AUSE 100: PRINT AT 7,0; PAPER 1;b$: GO TO 4030
4355 IF xg>8 AND xg<248 THEN  PLOT xg,y+9: DRAW 2,0: PLOT xg+1,y+8: DRAW 0,2: PA
USE 5: PLOT  INVERSE 1;xg,y+9: DRAW  INVERSE 1;2,0: PLOT  INVERSE 1;xg+1,y+8: DR
AW  INVERSE 1;0,2
4360 GO TO 4300
4400 REM 888 BOLA PARADA 888
4405 IF y>.5 THEN  LET ix=0: GO TO 4305
4410 IF ABS (x-p(6,1))<.25 THEN  GO TO 5000
4420 IF xg<8 OR xg>=248 THEN  PRINT AT 7,10; PAPER 7; INK 0; FLASH 1;"fuera de g
reen": GO SUB 8040: PAUSE 100: GO TO 520
4430 GO TO 4030
5000 REM 888 FINAL HOYO 888
5010 LET xg=FN d(p(6,1)): PLOT  INK 7;xg,6: DRAW  INK 7;2,0
5020 PRINT AT 10,5; PAPER 6; INK 1; BRIGHT 1;"  ENTRO EN ";golpes;" GOLPES  ": P
RINT AT 11,10; PAPER 6; INK 1; BRIGHT 1;ABS (golpes-p(7,1));
5030 IF golpes<p(7,1) THEN  PRINT  PAPER 6; INK 1; BRIGHT 1;" BAJO PAR "
5040 IF golpes>=p(7,1) THEN  PRINT  PAPER 6; INK 1; BRIGHT 1;" SOBRE PAR "
5050 LET tpar=tpar+p(7,1): LET tgolpe=tgolpe+golpes
5060 PRINT AT 0,0;"HOYO   ";hoyo,"    TOTALES","DIST.  ";p(6,1)-8,"GOLPES ";tgol
pe,"PAR    ";p(7,1),"PAR    ";tpar,"GOLPE  ";golpes,"DIFER. ";tgolpe-tpar
5070 GO SUB 8060: PAUSE 200: LET hoyo=hoyo+1: IF hoyo<19 THEN  GO TO 500
5080 PAPER 1: INK 7: CLS : PRINT AT 9,4;"TERMINO EL RECORRIDO CON": PRINT AT 11,
11;ABS (tgolpe-tpar);" golpes"
5090 PRINT AT 13,5;"SOBRE EL PAR DEL CAMPO": IF tgolpe<tpar THEN  PRINT AT 13,5;
" BAJO"
```

```
5100  PRINT AT 20,1;"desea empezar de nuevo ?   s/n"
5110 LET o$=INKEY$: IF o$="s" THEN  GO TO 5
5120 IF o$="n" THEN  GO TO 9999
5130 GO TO 5110
6000 REM 888 PRESENTACION 888
6010 BORDER 6: PAPER 6: INK 1: CLS : PRINT AT 5,0;
6020 PRINT " 8888888 8888888 8       8888888"
6030 PRINT " 8     8 8     8 8       8"
6040 PRINT " 8     8 8     8 8       8"
6050 PRINT " 8       8     8 8       8"
6060 PRINT " 8       8     8 8       88888"
6070 PRINT " 8   888 8     8 8       8"
6080 PRINT " 8     8 8     8 8       8"
6090 PRINT " 8     8 8     8 8       8"
6100 PRINT " 8888888 8888888 8888888 8"
6110 PRINT AT 16,3;"(C) ALFREDO CASANA BAYONA": PLOT 137,48: DRAW 5,0: PRINT AT
20,0;"  DESEA INSTRUCCIONES ?  s/n"
6120 LET o$=INKEY$: IF o$<>"s" AND o$<>"n" THEN  GO TO 6120
6130 IF o$="n" THEN  BORDER 1: GO TO 20
6140 BORDER 1: PAPER 1: INK 7: CLS : PRINT AT 21,0; PAPER 4;b$: PRINT AT 21,0; P
APER 2;" ": PRINT AT 21,31; PAPER 2;" "
6150 PRINT AT 21,14; PAPER 4; INK 6;"333": PRINT AT 21,19; PAPER 4; INK 5;"333":
 PRINT AT 21,25; INK 4; BRIGHT 1;"888"
6160 PRINT AT 19,8; INK 4;"AA": PRINT AT 20,8; INK 4;"BB"
6170 PLOT 216,8: DRAW 0,15: DRAW 8,-3: DRAW -8,-3: PLOT 8,8
6180 PRINT AT 17,5;"arboles";TAB (24);"green": PRINT AT 19,13;"arena agua"
6190 PRINT AT 0,0;"El recorrido completo consta de","18 hoyos en los cuales pued
en","aparecer los siguientes obstacu-","los y dificultades :": GO SUB 8070
6200 PRINT AT 0,0;"ARBOLES : Pueden, segun las cir-cunstancias dejar pasar la bo
la a su traves, rechazarla o causar su perdida entre las ramas": GO SUB 8070
6205 PRINT AT 0,0;"ARENA : reduce el bote de las  bolas y amortigua la fuerza d
e  los golpes que se realizan sobreella": GO SUB 8070
6210 PRINT AT 0,0;"AGUA : Las bolas que caen al a- gua se pierden irremisiblemen
te": GO SUB 8070
6220 PRINT AT 0,0;"GREEN : Es la zona que rodea el hoyo, en su interior el bote
de la bola es mas vivo. Cuando la  bola ha entrado en el GREEN     este se ampli
ara en la pantalla para afinar mejor el tiro final.": GO SUB 8070
6230 PRINT AT 0,0;"La fuerza deseada en los golpes se seleccionara pulsando la
   tecla ENTER cuando el indicador de la fuerza se halle en el     nivel adecuad
o. Lo mismo ocurri-ra con el angulo de salida de labola.": GO SUB 8070
6240 PRINT AT 0,0;"Una vez fijada la fuerza y el   angulo se puede ordenar el go
lpepulsando de nuevo la tecla      ENTER cuando se considere que   las condicion
es del viento( ob- servables en el banderin del    GREEN )son adecuadas": GO SUB
 8070
6250 PRINT AT 0,0;"Dentro del GREEN los golpes son siempre rasos y el efecto del
   viento nulo por lo que solo se  pide la fuerza del golpe": GO SUB 8070
6260 PRINT AT 0,0;"La bola tambien puede perderse  fuera de los limites del camp
o  ( zona de color rojo ). En cual-quier caso la perdida de una bo-la se penaliz
a con un golpe mas debiendose repetir el golpe": GO SUB 8070
6270 CLS : PRINT AT 12,2;"pulse una tecla para empezar": PRINT AT 15,10;"BUENA S
UERTE": GO SUB 8070
6280 GO TO 20
8000 REM 888 SUBRUTINAS 888
8010 PRINT AT 7,10; PAPER 1; INK 7;"pulse ENTER": RETURN
8020 PRINT AT 7,0; PAPER 1; INK 7;b$: RETURN
8030 BEEP .3,10: BEEP .5,15: BEEP .1,17: BEEP 1,21: RETURN
8040 BEEP .3,21: BEEP .5,15: BEEP .1,17: BEEP 1,10: RETURN
8050 FOR a=0 TO -11 STEP -1: BEEP .1,a: NEXT a: BEEP 1,-12: RETURN
8060 FOR a=15 TO 25: BEEP .1,a: NEXT a: BEEP .1,26: PAUSE 5: BEEP .1,20: PAUSE 7
: BEEP .1,26: RETURN
8070 LET o$=INKEY$: IF o$="" THEN  GO TO 8070
8080 PRINT AT 0,0;b$;b$;b$;b$;b$;b$;b$: RETURN
9000 DATA 8,8,28,28,60,62,126,60,126,254,255,126,255,255,126,24,4,6,6,5,4
9010 DATA 25,12,0,17,8,210,4
9020 DATA 10,5,15,0,0,100,3
9030 DATA 15,8,0,0,19,123,3
9040 DATA 16,20,0,10,25,140,3
9050 DATA 25,17,21,0,0,210,4
9060 DATA 20,10,0,0,17,165,4
9070 DATA 17,12,0,25,0,153,3
9080 DATA 19,5,23,0,9,160,3
9090 DATA 11,0,0,15,8,95,3
9100 DATA 22,12,0,16,0,182,4
9110 DATA 27,10,22,0,16,225,4
9120 DATA 18,22,0,13,0,148,3
9130 DATA 12,16,0,20,8,100,3
9140 DATA 23,10,0,17,14,200,4
9150 DATA 28,6,24,10,18,229,5
9160 DATA 16,10,0,0,20,140,3
9170 DATA 21,13,17,0,10,172,4
9180 DATA 28,10,14,18,21,244,5
9500 DEF FN d(x)=(x-p(1,1)*8)*10+8
9999 BORDER 7: PAPER 7: INK 0: CLS
```



END

bazar SOFT

Android one

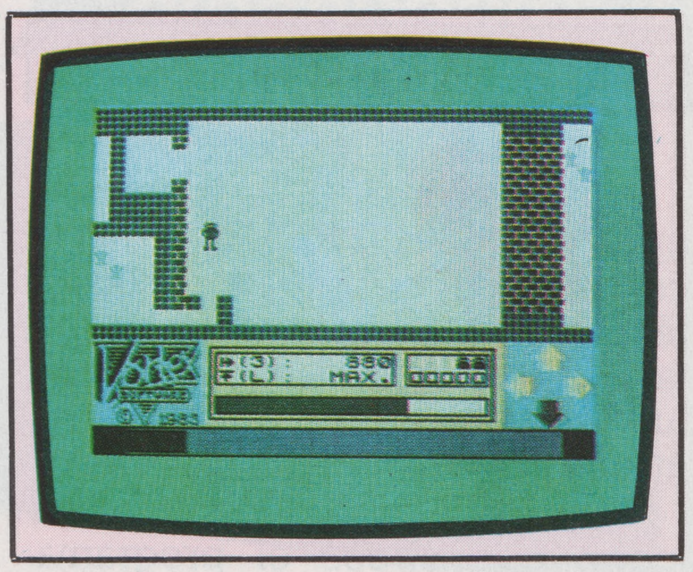
El objetivo es infiltrarse en el complejo de un reactor, evitar sus sistemas defensivos, destruirlo y volver a la base. Para conseguirlo, se dispone de una poderosa arma, lo último en tecnología androide: el ANDROIDE 1, dotado de rayo laser, barrera de fuerza y potencia ilimitada.

El complejo del reactor está vigilado por los mutantes defensores. Estos no se detienen ante nada para impedir el éxito de la misión. Los mutantes son: Saltarín, Groupie, Patinador y Asustador.

Como ayuda para llevar a feliz término la peligrosa aventura se cuenta con el concurso de los indicadores de dirección, vidas y barreras utilizadas, de tiempo transcurrido desde el comienzo de la misión y de situación de la sala, además del de nivel de dificultad y del marcador con la puntuación en curso y record.

El juego termina cuando ANDROIDE 1 ha sido alcanzado por cinco disparos, o cuando el tiempo disponible para la destrucción del reactor se ha agotado.

El juego, que admite cinco niveles de dificultad, puede manejarse desde el teclado o por medio de un JOYSTICK KEMPSTON.



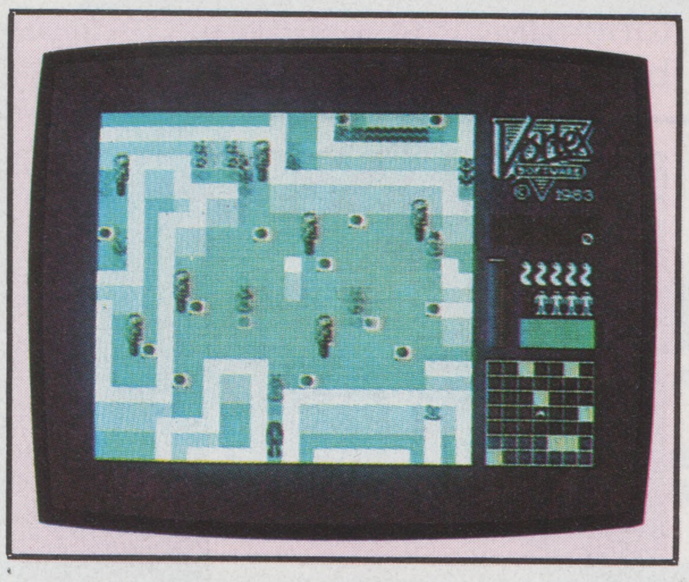
Ordenador	ZX-SPECTRUM
Soporte	Casete
Tipo	Juego
Origen	Vortex Software
Distribuidor	Ventamatic
Precio	1.600 ptas.

Gráficos/estética	**
Sonido	**
Facilidad de control	*
Originalidad	**
Adicción	**
Resumen	**

Android two

Este programa se presenta como continuación al ANDROIDE 1, y tiene la novedad de que, esta vez, la acción se desarrolla a través de un laberinto en 3 dimensiones, infectado de mutantes. Ahora la misión consiste en destruir 5 MEGATOIDES en 3 zonas diferentes: el Laberinto de la Muerte, la Zona Paradójica y las Tierras Bajas.

ANDROIDE 2 es transportado a cada una de estas zonas, debiendo actuar rápidamente, ya que la cápsula de transporte desaparece tras un corto periodo de tiempo, impidiéndole el regreso a la nave. El ingenio está equipado con un disparador de rayos fotónicos que destruye a los MEGATOIDES al conseguir tres blancos en su cabeza. Otros compañeros de aventura son: el OVERDROIDE, que es destruido con un solo disparo, el SALTADOR que es indestructible, aunque predecible, y las minas, que deben esquivarse a toda costa.



Existen indicadores de puntuación, de MEGATOIDES destruidos, vidas, minas y de tiempo para la desaparición de la cápsula, además del indicador de dirección y del radar. El juego termina cuando ANDROIDE 2 ha sido alcanzado 5 veces o se ha agotado el tiempo disponible para la misión. El control se realiza desde el teclado o a través del JOYSTICK KEMPSTON.

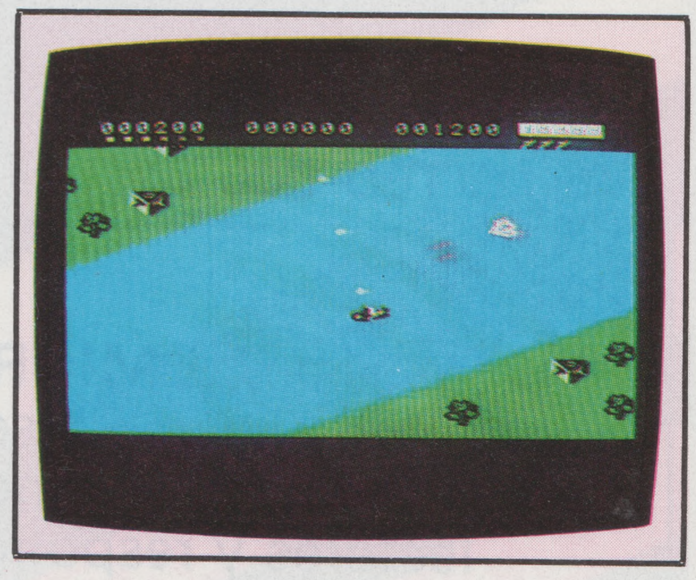
Ordenador	ZX-SPECTRUM
Soporte	Casete
Tipo	Juego
Origen	Vortex Software
Distribuidor	Ventamatic
Precio	1.700 ptas.

Gráficos/estética	***
Sonido	**
Facilidad de control	***
Originalidad	**
Adicción	**
Resumen	**

Turboat

En este caso se trata, básicamente, de un juego de acción, para un solo jugador, que se desarrolla en tres fases. La misión del protagonista, TURBOAT, es capturar los radares secretos del enemigo para que los científicos los analicen. En la primera etapa, TURBOAT dispone de un número limitado de disparos para penetrar en la zona enemiga. Para impedir el paso de nuestra intrépida nave, el cuartel general enemigo dispone de Jet-Foils y puestos de tiro en tierra, armados con misiles. TURBOAT debe destruir 10 Jet-Foils para pasar a la siguiente etapa.

En la segunda fase, nuestro ingenio penetra en las líneas enemigas, perdiendo todo su poder ofensivo. Debe capturar los radares evitando las bombas y torpedos de los aeroplanos. Cuando ha conseguido 5 radares se completa la segunda etapa, accediendo a la tercera y última de la misión. En esta última fase, el enemigo dispone de misiles y torpedos para evitar la huida de TURBOAT, que debe intentar permanecer intacto hasta volver a sus líneas.



Ordenador	SPECTRAVIDEO
Soporte	Casete
Tipo	Juego
Origen	Mass Tael Ltd.
Distribuidor	Dynadata
Precio	—

Gráficos/estética	***
Sonido	**
Facilidad de control	**
Originalidad	***
Adicción	***
Resumen	***

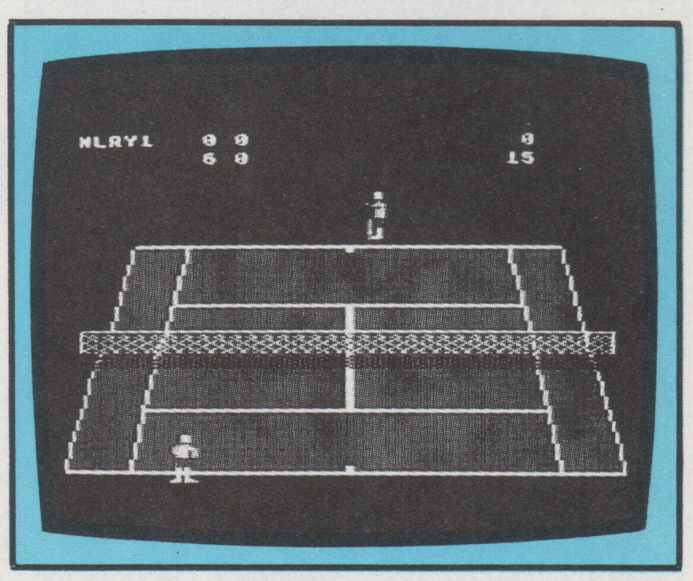
Tennis

Al accionar el interruptor de encendido de su ATARI, con este cartucho enchufado en el receptáculo al efecto, convertirá la pantalla del TV en el escenario de una apasionante competición tenística.

Varias son las posibilidades de juego que ofrece el programa: partido individual de un jugador contra la máquina, individual con dos jugadores como contendientes, o partido de dobles controlando cada jugador a uno de los tenistas y quedando el otro componente de cada equipo en manos del ordenador.

El juego se desarrolla sobre la pista de tenis que ocupa la totalidad de la pantalla durante el tiempo de juego. Las reglas coinciden exactamente con las que rigen los partidos de tenis de alta competición, hasta el punto de que incluso se aplica el «tiebreaker» (muerte súbita) para resolver un set empatado a seis juegos.

El éxito depende de la habilidad del jugador empuñando el joystick. Las posibilidades del programa permiten al jugador experimentado orientar sus golpes hacia 8 zonas del terreno contrario, al tiempo que ponen a disposición del jugador todo el repertorio de golpes tenísticos: lobs, boleas, vaselinas...



Ordenador	ATARI
Soporte	Cartucho
Tipo	Juego
Origen	ATARI Inc.
Distribuidor	Unimport
Precio	8.450 ptas.

Gráficos/estética	***
Sonido	**
Facilidad de control	***
Originalidad	***
Adicción	***
Resumen	***

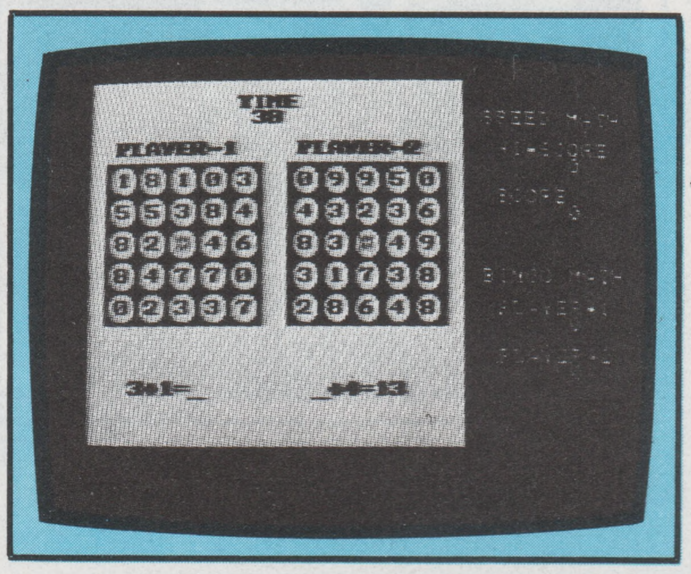
Bingo Math

BINGO MATH es una aplicación educativa, disponible en cartucho para el COMMODORE 64, que permite tanto el aprendizaje como el ejercicio de la agilidad mental del usuario, a través de cálculos numéricos simples, basados en las cuatro reglas aritméticas.

El sistema elegido para el desarrollo de esta habilidad parece de lo más apropiado, al recurrir a una especie de BINGO en el que pueden participar dos jugadores.

Dispone de diferentes modalidades de operaciones. En principio las cuatro básicas, más una especial que propone la resolución de ejercicios, en cuyo cálculo intervienen varias operaciones dentro de una fórmula aleatoria.

El juego se controla fácilmente a través del teclado, permitiendo la selección de las diferentes modalidades.



Ordenador	COMMODORE 64
Soporte	Disco
Tipo	Educativo
Origen	Hes Ware
Distribuidor	Micr. y Control
Precio	8.000 ptas.

Gráficos/estética	**
Sonido	**
Facilidad de control	***
Originalidad	**
Adicción	*
Resumen	**

el correo del lector

La revista **TU MICRO** también tiene un espacio reservado para reflejar las sugerencias de los lectores y responder a sus consultas.

Las cartas para «El correo del lector» debe dirigirlas a:

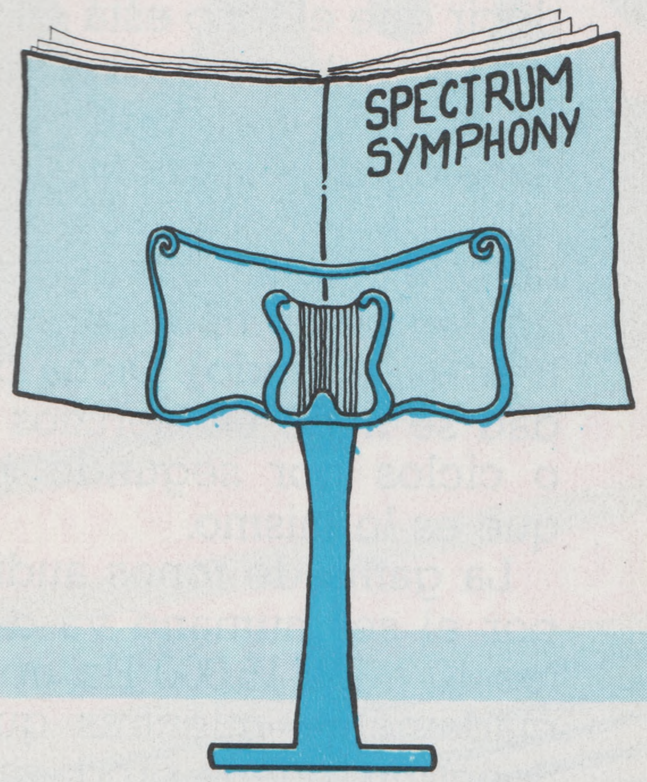
TU MICRO
Apdo. de Correos 61.294
28080MADRID.

ANUNCIOS BREVES

Un servicio gratuito para los lectores de **TU MICRO**

- Compra/venta e intercambio de equipos, accesorios y programas.
- Avisos y comunicaciones de clubs de usuarios
- ...

Vea tarjeta ANUNCIOS BREVES



El Spectrum en concierto

Uno de los aspectos que ha adquirido mayor relevancia en los últimos tiempos, dentro del ámbito de los ordenadores personales, es la capacidad que poseen algunos de ellos para generar cierta clase de sonidos. Esta aptitud es más que evidente en la pléyade de programas de juego que inundan el mercado. Programas que suelen incorporar hábiles subrutinas virtuosas en la generación de sonidos, combinadas, a la sazón, con otras subrutinas no menos hábiles que crean en la pantalla representaciones gráficas a todo color.

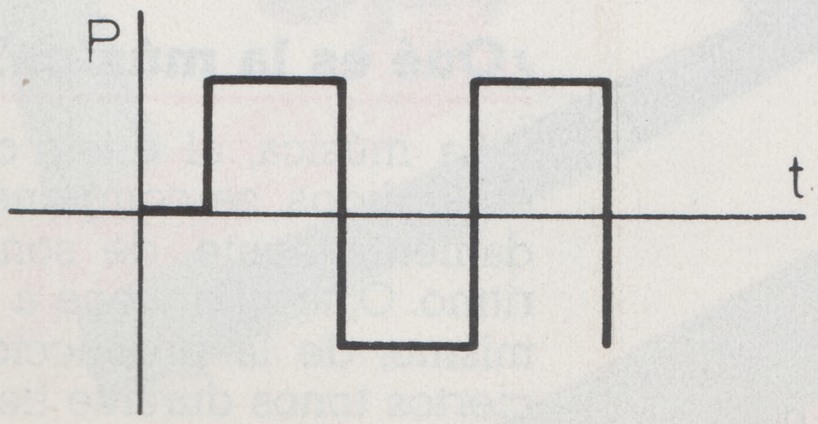
En esta exposición intentaremos enseñarle a «tocar» el ordenador, concretamente el Spectrum, proporcionándole en primer lugar algunas nociones musicales que resultan indispensables para tal cometido. Una vez que el lector conozca estas técnicas, podrá crear y componer sus propias tonadas y melodías (¡y quién sabe si hasta sinfonías!) y, por supuesto, podrá convertirse en un maestro del «Sensuround» y los efectos especiales, sin más que investigar un poco por su cuenta y riesgo.

¿Qué sabe usted del sonido?

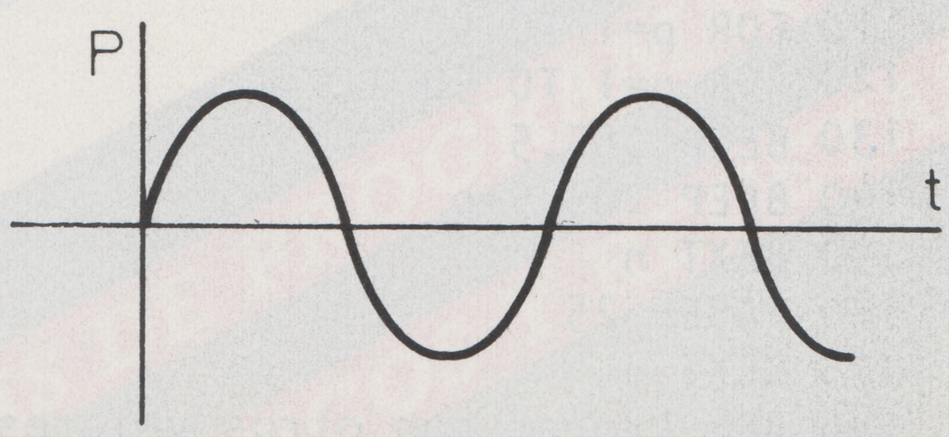
El sonido se produce cuando un elemento físico perturba la atmósfera, creando variaciones en la presión del aire que le rodea. Dichas variaciones de presión se propagan en forma de onda esférica, y, si son interceptadas por la membrana del tímpano, se transforman en impulsos nerviosos que van al cerebro... ¡Oimos!

Si la presión pasa instantá-

1



onda «cuadrada»



onda «sinusoidal»

LISTADO

PROGRAMA 1

```
100 REM Programa para comprobar
110 REM las posibles duraciones
120 LET d=0.01
130 FOR n=1 TO 10
140 PRINT AT 1,1; "Duracion =  ";d;"segundos";
150 BEEP d,0
160 PAUSE 25
170 LET d=d*2
180 NEXT n
```

PROGRAMA 2

```
100 REM Gama completa de tonos
110 FOR p=-60 TO 69
120 PRINT AT 1,1; "Tono =  ";p;
130 BEEP 0.5,p
140 PAUSE 12
150 NEXT p
```

END

neamente a su valor máximo y permanece en él durante un breve intervalo de tiempo, volviendo bruscamente al valor mínimo, entonces se califica a la onda como «cuadrada», a diferencia de la ya conocida y más suave onda «sinusoidal» (fig. 1).

LISTADO

PROGRAMA 3

```
100 REM Efecto de sirena
110 FOR p=5 TO 15
120 BEEP .05,p: NEXT p
130 FOR p=15 TO 5 STEP -1
140 BEEP .05,p: NEXT p
150 GOTO 110
```

PROGRAMA 4

```
100 REM Efecto de trémolo
110 FOR p=1 TO 4
120 FOR n=1 TO 50
130 BEEP .05,5
140 BEEP .05,5+p
150 NEXT n
160 PAUSE 25
170 NEXT p
180 REM Probar con otros valores de p
```

END

El tono de un sonido puede ser, hablando sumarialmente, más grave o más agudo. Con mayor propiedad, podemos decir que el tono está en relación directa con la frecuencia del sonido producido, pudiéndose definir ésta como la cantidad de variaciones completas que tienen lugar durante la unidad de tiempo (para nosotros, el segundo). Dicha cantidad se mide en Hertzios (Hz) o ciclos por segundo (c/s), que es lo mismo.

La gama de tonos audibles por el ser humano va desde los 40 a los 15.000 Hz, aproximadamente, mientras que el Spectrum puede producir una gama de tonos, de onda cuadrada, situados de 10 a 15.000 Hz, aproximadamente. La forma de generar dichos sonidos consiste en conmutar y desconmutar electrónicamente, a intervalos regulares, una señal tensión constante, y llevar las variaciones, amplificadas, a un pequeño altavoz alojado en su interior. El control del sonido, tanto en la duración como en la frecuencia, lo efectúa la CPU del Spectrum por medio de la instrucción BEEP d, p, en donde el parámetro d representa la duración, en segundos, y el parámetro p representa el código del tono o frecuencia; este último puede tomar cualquier valor entero comprendido ente –59 (≅ 10 Hz) y + 69 (≅ 15.000 Hz). Los dos ejemplos siguientes, programas 1 y 2, aclararán lo anteriormente expuesto:

Concretamente, cuando p es igual a cero, la frecuencia generada corresponde al tono del Do intermedio de la escala del piano, y vale, aproximadamente, 261 Hz.

Veamos dos ejemplos ilustrativos más en los programas 3 y 4.

¿Qué es la música?

La música, al decir de los entendidos, se compone, fundamentalmente, de sonido y ritmo. O, lo que viene a ser lo mismo, de la producción de ciertos tonos durante tiempos determinados. Es decir, que volvemos a encontrarnos con

VENTAMATIC

presenta

EL CARNET DE SOCIO DEL CLUB NACIONAL DE USUARIOS DE LOS ZX (ZX81, ZX-SPECTRUM Y QL)

Y SUS EXTRAORDINARIAS VENTAJAS

* Boletín informativo exclusivo para los socios.
* Servicio telefónico de consultas.
* Fabulosas ofertas especiales.
* Descuento del 10% y plaza preferente en los cursos de informática de VENTAMATIC en Barcelona.
* Grandes descuentos para los socios de hasta el 35% en sus compras de ordenadores, periféricos, accesorios y programas (SINCLAIR, COMMODORE, ORIC, etc.) a VENTAMATIC.
* Sorteos de programas en cada boletín.
* Posibilidad de comercializar los programas realizados por los socios.
* Regalo a elegir al formalizar la inscripción.



club NACIONAL DE USUARIOS DE LOS ZX

NOMBRE:

SOCIO N.º: CADUCA:

FIRMA AUTORIZADA

1. ESTA TARJETA ES PERSONAL E INTRANSFERIBLE.
2. CADUCA A LA PUBLICACION DEL BOLETIN CUYO NUMERO SE ESPECIFICA EN EL ANVERSO.
3. ES IMPRESCINDIBLE LA PRESENTACION DE ESTA TARJETA PARA BENEFICIARSE DE LOS DESCUENTOS EN COMPRAS A VENTAMATIC Y DEMAS VENTAJAS OFRECIDAS POR EL CLUB.

Club Nacional de Usuarios de los ZX c/ Córcega, 89, entlo. BARCELONA-29

SINCLAIR GRAPH, S.A.

INDICES ULTIMOS NUMEROS BOLETINES

N.º 10
EDITORIAL, APUNTES DE PROGRAMACION (I), REALIZACIONES DE LOS SOCIOS, CONCURSO DE PROGRAMAS, SOFT-STARS, NOVEDADES, SOFT-BANK, SPECTRUCOS, JOYSTICKS (I), EL QL, CORRESPONDENCIA DEL CLUB, BOLSA DEL CLUB, PROGRAMAS (Privatización de programas - Laberinto - Conservación de gráficos en memoria - 4 en raya - Quiniela ahorro - Reloj - Cassette - Seguridad - Ordenación).

N.º 11
EDITORIAL, APUNTES DE PROGRAMACION (II), REALIZACIONES DE LOS SOCIOS, CONCURSO DE PROGRAMAS, SOFT-STARS, NOVEDADES,SOFT-BANK, SPECTRUCOS, JOYSTICKS (II), LOS MICRODRIVES, CORRESPONDENCIA DEL CLUB, BOLSA DEL CLUB, INFORMATICA Y ENSEÑANZA, GRAFPAD (Tablero gráfico), EL RINCON DEL CODIGO MAQUINA, RUTINAS DE LA ROM, ENTREVISTA CON MATTHEW SMITH, PROGRAMAS (Zylones - Tron - Simulación de reconocimiento de Voz - Número e - caracteres - gráfica 3D - Slalom espacial - Batalla Naval).

N.º 12
EDITORIAL, APUNTES DE PROGRAMACION (III), REALIZACIONES DE LOS SOCIOS, CONCURSO DE PROGRAMAS, SOFT-STARS, NOVEDADES, SOFT-BANK, EL RINCON DEL CODIGO MAQUINA, SPECTRUCOS, CORRESPONDENCIA, BOLSA DEL CLUB, BIBLIOGRAFIA DEL SPECTRUM, LAS NUEVAS REVISTAS INFORMATICAS, COMENTARIO A LA ENCUESTA DEL NUM. 10,TECLAS DE FUNCION EN EL SPECTRUM,ORGANIZACION DE LA MEMORIA EN EL ZX-81, CONSULTAS, TRUCOS CON EL MICRODRIVE, TRUCOS CON EL ZX-81, PROGRAMAS (Tragaperras - Reloj en tiempo casi real - Barcos - Ruleta - Revuelto - Renumerador - Laberinto 3D - Super Save - Zoom de Funciones - El Canal del Llobregat - Crancs).

BOLETIN DE INSCRIPCION

Enviar a: **VENTAMATIC** - Dpto. Venta por Correo - Avda. de Rhode, n.º 253 ROSAS (Gerona), o personalmente en VENTAMATIC - Exposición, Venta y Cursos - c/ Córcega, n.º 89, entlo. - 08029 BARCELONA.

☐ Deseo ser inscrito como socio del **Club Nacional de Usuarios de los Usuarios de los ZX** y recibir **6 boletines** a partir del N.º _____ inclusive (último N.º publicado: 11). Por formalizar mi inscripción antes del **15 de Octubre de 1984** tengo derecho a recibir el regalo que señalo con una cruz, **totalmente gratis.**

REGALO (escoger uno solo por inscripción)	*P.V.P.*
☐ Libro «Guía práctica del BASIC del ZX81 y del ZX-SPECTRUM»	1.200 ptas.
☐ Funda de plástico para ZX-SPECTRUM	400 ptas.
☐ Paquete de cinco cintas vírgenes C-15	500 ptas.
☐ Revista en cassette SPECTRUMANIA N.º 1	750 ptas.
☐ Juego de plantillas para teclado de ZX-SPECTRUM	100 ptas.

NOMBRE: _____

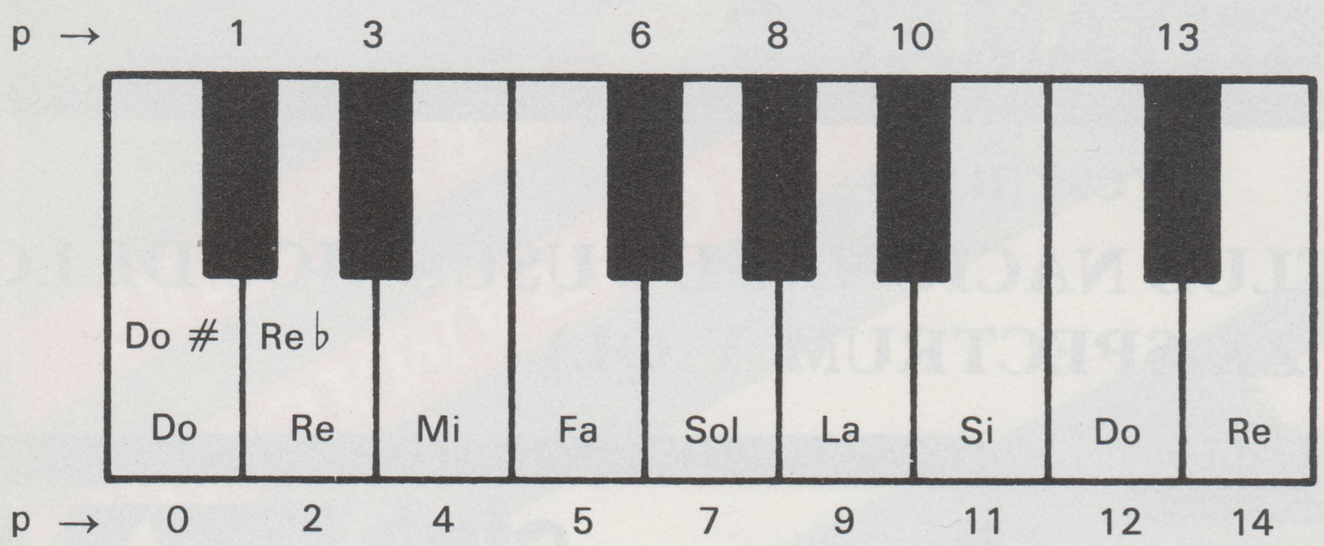
APELLIDOS: _____

CALLE: _____ N.º _____

CODIGO POSTAL: _____ POBLACION: _____

PROVINCIA _____

Envío 2.500 ptas. mediante:
☐ Talón adjunto.
☐ Giro postal N.º: _____ de fecha: _____



Distribución de las teclas en un piano convencional.

los dos parámetros d y p de la instrucción BEEP: duración y tono. La diferencia entre el lenguaje musical y el del Spectrum es únicamente cuestión de formas, como demostraremos más adelante. Pero, antes que nada, conozcamos los componentes elementales de la música, las notas, cuya representación escrita nos indica la altura de tono en que nos encontramos y la duración del mismo.

Para una más fácil comprensión por parte de los no iniciados, consideremos el teclado de un piano. En él existen un conjunto de teclas blancas centrales, correspondientes a las notas llamadas Do, Re, Mi, Fa, Sol, La, Si (C, D, E, F, G, A, B, en nomenclatura internacional), que constituyen una «octava» y a las que corresponden ciertos tonos o frecuencias y unos códigos de tono «p» determinados, como ya hemos visto anteriormente. La figura 2 ilustra la distribución del teclado, en el cual existen otras teclas, negras, que corresponden a los tonos intermedios que quedaban por definir.

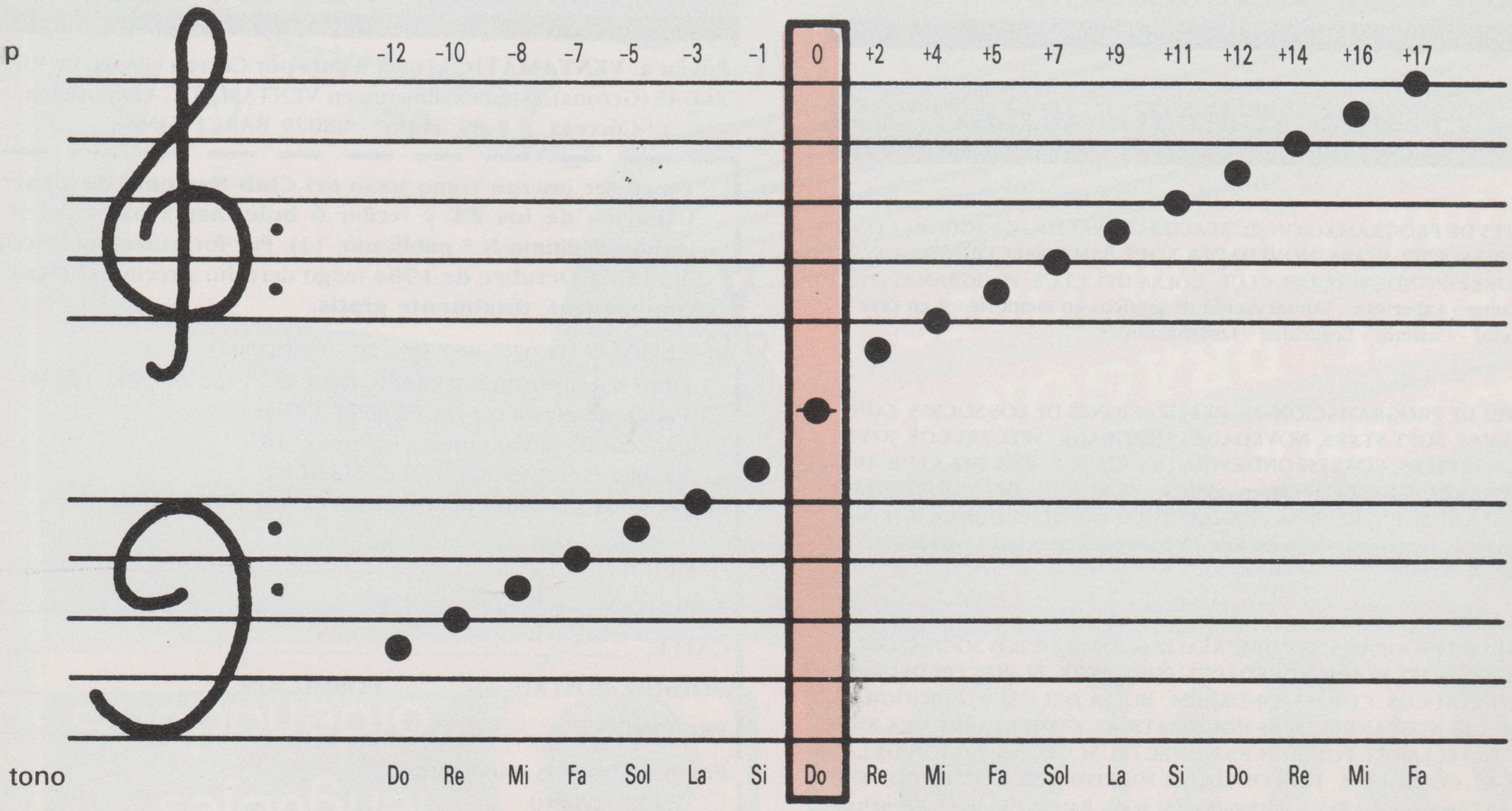
2

Estos tonos intermedios se pueden apellidar «sostenido» (#) y adoptar el nombre de la nota anterior, o bien, «bemol» (♭), si el nombre adoptado es el de la nota posterior. Por ejemplo, el tono del Do # coincide con el del Re ♭ . Y no nos pregunten por la razón de toda esta aparentemente caprichosa distribución, porque verdaderamente se sitúa fuera del contexto elemental en el que nos encontramos. Cabe añadir que la distancia o diferencia entre dos tonos consecutivos (dos teclas contiguas, blancas, o blanca y negra), se denomina semitono.

La escritura de las notas se realiza sobre un conjunto de cinco líneas, llamado pentagrama. En nuestro caso utilizaremos dos de sus muchas versiones: el de clave de Sol, en

Claves de Sol (arriba) y de Fa en 4.ª (abajo). Los valores de «p» dan los códigos de tono.

3



segunda línea, y el de clave de Fa en cuarta línea. La figura 3 y el programa 5 contribuirán a clarificar el tema con la generación de la escala natural.

¿Y las duraciones?

No vayan a creer que se nos olvidaba. Si no existe un control de la duración de cada nota, no puede haber ritmo, ni música... ¡ni madre del cordero!

Pues bien. La duración de cada nota se indica junto a la propia nota, mediante un dibujo que se añade a la misma. La figura generalmente aceptada como referencia es la «negra», y su duración es, aproximadamente, de 1/4 de segundo. Las restantes figuras de duración se denominan: redonda, blanca, corchea, semicorchea, fusa... y ya vale, por el momento. De esta manera, cada sonido queda completamente determinado por su duración, mediante un dibujo adicional, y por su altura de tono, mediante su posición en el pentagrama. Por lo tanto, su transcripción a los parámetros d y p, resulta trivial y puramente mecánica. No obstante, para facilitar más aún la labor, podemos tomar como referencia el valor de la duración de la fusa, que es de 1/32 segundos (figura 4). De esta forma, los valores de las duraciones del resto de las notas serán las sucesivas potencias de 2, lo cual facilitará en gran medida la labor de programación, como se puede apreciar en el programa 6. Este contiene un sencillo programa generador de notas completas; es decir, con duraciones y tonos, que van siendo introducidos en el programa por pares de datos.

Sin embargo, sería poco menos que ofensivo para los grandes compositores decir que la teoría musical termina aquí, cuando todavía ni siquiera hemos comenzado. Pero para lo que nosotros necesitamos, nos conformaremos con definir brevemente dos cosas más: el puntillo y los silencios.

El puntillo, colocado a la derecha de cualquier nota, aumenta su duración en la mitad.

LISTADO

PROGRAMA 5

```
100 REM Generación de la escala musical
110 FOR k=1 TO 20
120 RESTORE
130 FOR n=1 TO 8
140 READ p
150 BEEP .25,p
160 NEXT n
170 PAUSE 25
180 NEXT k
190 DATA 0,2,4,5,7,9,11,12
```

END

Los silencios son los tiempos muertos en los que el compositor ha decidido que no debe haber sonido alguno. Todo ello queda claramente reflejado en las figuras 5 y 6.

Existen dos alternativas para «interpretar» los silencios y, como es obvio, ambas utilizan la instrucción PAUSE. La primera consiste en poner un signo menos en la duración, que será detectado por un comparador. La segunda podría consistir en añadir un tercer conjunto de datos de entrada, de manera que si el dato leído es cero, el comparador lo ignora y se genera el tono; por el contrario, si no es cero, entonces el ordenador estaría en presencia de un silencio y

LISTADO

PROGRAMA 6

```
100 REM Programa generador de una melodía
110 REM con duraciones relativas "l" y tonos
120 LET c=1/32
130 FOR n=1 TO 26
140 READ l,p
150 BEEP c*l,p
160 NEXT n
170 DATA 8,11,4,14,8,11,8,11
180 DATA 4,14,8,11,4,12,4,14
190 DATA 4,12,8,9,4,11,4,12
200 DATA 4,11,8,7,8,11,4,14
210 DATA 8,11,8,11,4,14,8,11
220 DATA 4,12,4,14,4,12,8,9
230 DATA 4,11,8,7
```

END

se desviaría el control hacia la instrucción PAUSE correspondiente. El único y mínimo inconveniente es que PAUSE funciona en unidades de 1/50 de segundo y la coincidencia de duraciones con los tonos puede no ser absoluta.

¡Música, maestro!

Y ya estamos preparados para rozar con nuestra varita el Spectrum y convertirlo en un instrumento musical. A partir de ahora, podremos «tocar» el Spectrum de igual forma que si tocásemos el piano; el programa 7, «El concertista», se ocupará de instruir oportunamente a la máquina.

Nuestro piano, pues, consta de 9 teclas blancas (Do, Re, Mi, Fa, Sol, La, Si, Do, Re), localizadas en el teclado del SPECTRUM sobre la línea media de letras (teclas: a, s, d, f, g, h, j, k, l,), y de 6 teclas negras (Do #, Re #, Fa #, Sol #, La #, Do #), situadas en las teclas, w, e, t, y, u, o, de la fila superior de letras. De cualquier forma, nuestro piano es generosamente didáctico y nos va a mostrar en pantalla la distribución correcta (ver figura 7).

Y ya no vamos a entretenerle con mayores precisiones. Introduzca, sin demora, nuestro programa musical en el SPECTRUM y ...¡a jugar! o a tocar el piano, como Vd. prefiera.

tuMicro

4

Nota	Duración relativa R	Duración absoluta d=c*l
Fusa	1	1/32 seg.
Semicorchea	2	1/16 seg.
Corchea	4	1/8 seg.
Negra	8	1/4 seg.
Blanca	16	1/2 seg.
Redonda	32	1 seg.

Figuras de duración y correspondencia de duraciones relativas y absolutas.

5

Nota	l	d=c*l
.	1,5	≅ 0,05 seg.
.	3	≅ 0,09 seg.
.	6	≅ 0,19 seg.
.	12	≅ 0,38 seg.
.	24	≅ 0,75 seg.
.	48	≅ 1,50 seg.

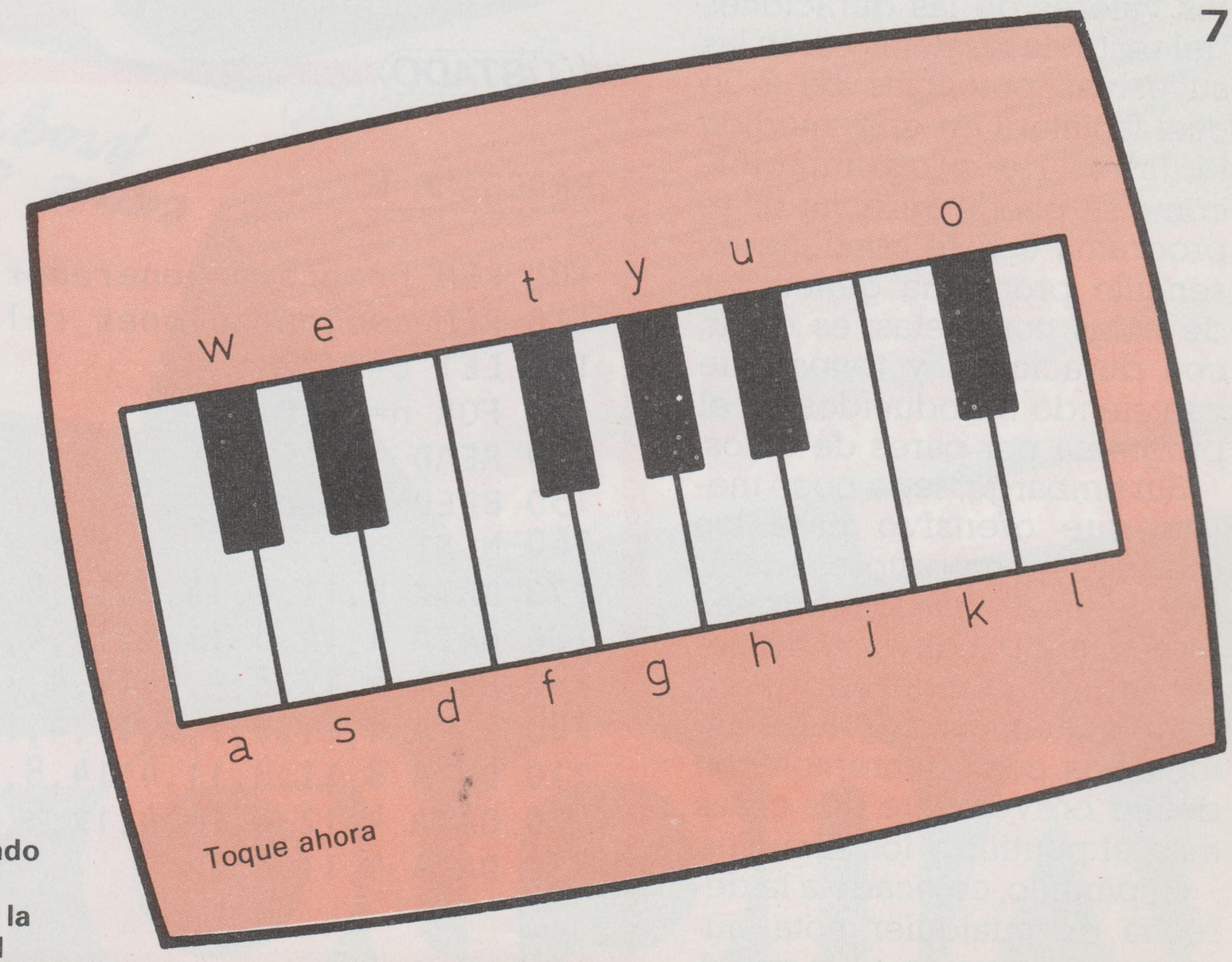
El puntillo, colocado a la derecha de cualquier nota, aumenta su duración en la mitad.

6

	Fusa
	Semicorchea
	Corchea
	Negra
	Blanca
	Redonda

Los silencios o tiempos muertos se representan por medio de las figuras que aparecen en la tabla.

7



Representación gráfica del teclado a utilizar en el programa 7: «El compositor». Las letras denotan la equivalencia del teclado musical con las teclas del SPECTRUM.

LISTADO

PROGRAMA 7

```
100 REM El concertista
110 DIM p(25)
115 REM Formación de la tabla de tonos
120 FOR n=1 TO 25
130 READ p(n)
140 NEXT n
150 DATA 0,64,64,4,3,5,7,9,64,11,12,14,64
160 DATA 64,13,64,64,64,2,6,10,64,1,64,8
170 GOSUB 500
175 REM Bucle para interpretar notas
180 LET a$=INKEY$ : IF a$="" " THEN GOTO 180
190 BEEP .25,p(CODE a$-96) : GOTO 180
200 STOP
490 REM Subrutina de representación
500 LET b$="" ""+CHR$ 133+CHR$ 138
510 FOR r=1 TO 4
520 PRINT AT 3+r,3;b$;
530 FOR k=1 TO 7
540 IF k=2 OR k=6 THEN PRINT "  "; :NEXT k
550 PRINT b$; :NEXT k
560 NEXT r
570 PRINT AT 2,4; "w e t y u o";
580 PRINT AT 13,3; "a s d f g h j k l"
590 FOR k=0 TO 8
600 PLOT 16+k*24,80
610 DRAW 24,0
620 DRAW 0,64
630 DRAW -24,0
640 DRAW 0,-64
650 NEXT k
660 PRINT AT 17,1; "Toque ahora";
670 RETURN
```

Descripción del programa 7: «El concertista»

1.º Se construye un conjunto p de 25 elementos (ya que no se utiliza la z). Cada elemento del mismo contiene un valor de tono, que será de 64 (inaudible) para las letras no utilizadas.

2.º La detección de la pulsación tiene lugar gracias a la instrucción INKEY$ y a la variable a$ asociada a ella.

3.º Se utiliza el teclado de minúsculas; por tanto, al ser 97 el código mínimo (el correspondiente a la «a»), se resta 96 del código de cada letra pulsada para obtener los valores 1 a 25 de los índices del conjunto p.

4.º La subrutina 500 representa en pantalla el dibujo del teclado.

END

REM

Fichero biblioteca

TU MICRO n.º 3/4. Pág. 91.

Algunos lectores nos han consultado acerca de la posibilidad de incrementar la longitud de los bloques de información procesables por este programa.

Los poseedores de un SPECTRAVIDEO, saben que una de las características de este equipo es la de admitir un máximo de 256 caracteres dentro de cada variable de cadena. Con esta limitación, no cabe duda que el fichero aceptará un reducido volumen de datos.

Tal inconveniente puede solucionarse. En esencia, la solución consiste en reservar un mayor espacio de memoria para las variables. Para ello, hay que echar mano de una sentencia "CLEAR N", en donde N coincidirá con el número de bytes a reservar. Sólo va a ser preciso modificar dos líneas del programa para acondicionarlo a la nueva situación. Veamos cual es el método a seguir:

a) Modificar la línea 111, en la que aparecía la instrucción RESTORE, de tal forma que su nueva expresión sea:
   111 CLEAR 7000 : RESTORE
b) Modificar la línea 700 para que, a partir de ahora, aparezca de la siguiente forma:
   700 CLEAR 7000 : PRINT "PONGA EN MARCHA EL CASETE PARA CARGAR EL FICHERO" : OPEN "cas:t" FOR INPUT AS 1.

Con estos ligeros retoques, el SPECTRAVIDEO reservará un espacio de 7 Kbytes para variables. En la versión SV-328, de mayor memoria, puede incrementarse la zona reservada para variables sin más que sustituir el parámetro 7000 por un número superior.



Ruleta rusa

La rancia aristocracia europea, gran amante de la ruleta rusa, mataba (¡y nunca mejor dicho) sus ratos de ocio, encomendando su sien al capricho de un revólver. Con el tiempo, el arriesgado juego fue extendiéndose hacia otras regiones del mundo en las que imperaba el «sarao». Fruto de esta diáspora es, por ejemplo, la obstinada afición de algunos «buenos» por dilucidar sus altercados con los más perversos pistoleros del lejano oeste, a base de sesiones de ruleta rusa.

Lo cierto es que el pasatiempo —eterno para los más desafortunados— tiene su gran dosis de emoción. A nuestro parecer, su encanto se vería aún realzado si el resultado del juego fuera menos «drástico». Y ésta ha sido, precisamente, la idea que nos ha movido a crear una versión programada de la ruleta rusa.

El propio SPECTRAVIDEO va a ser quien se encargue de colocar la bala «sorpresa» en una de las seis recámaras del revolver, y también será quien se ocupe de echar la suerte girando el tambor.

Las reglas son muy sencillas. Pueden participar hasta un total de seis temerarios. Por riguroso turno, previamente establecido, cada uno de los jugadores puede optar por dispararse, respondiendo afirmativamente a la pregunta al respecto que le formulará la máquina. Acto seguido, el interfecto puede ya accionar el fatídico gatillo sin más que pulsar cualquiera de las teclas del SPECTRAVIDEO. Si desea no arriesgar su sien en esta jugada, puede rechazar el ofrecimiento del revólver y pasar el turno al siguiente jugador.

Hay que tener en cuenta que, a medida que se hayan realizado más disparos fallidos, mayor será la probabilidad de que el próximo sea el definitivo.

Esto significa que habrá que andarse con tiento a la hora de pasar el turno al vecino, puesto que una de las reglas que nos queda por mencionar es que, a partir del cuarto disparo realizado, ningún jugador puede rechazar el dispararse. Así, pues, cualquier jugador está



expuesto a que le llegue el turno con disparo no rehusable, debido a que rechazó el disparo en una de las primeras jugadas de menor riesgo.

A medida que van saliendo del revólver disparos con «bala sorpresa», el corro de participantes se va reduciendo al ser eliminados los suicidas. Y, desde luego, ni que decir tiene que el vencedor será el que sobreviva a los restantes participantes.

Observe que el listado no incluye ningún caracter especial (caracteres gráficos o de control), con lo que la introducción del programa se limitará a una reproducción literal del listado adjunto. Antes de que la mano se arroje sobre el teclado para introducir la mágica palabra RUN, es conveniente que obtenga una copia en casete del programa. De no tomar esta precaución, puede darse el caso de que pierda el programa, pacientemente introducido caracter a caracter, por el mero hecho de haber cometido un error en la introducción. En esta tesitura, suelen darle a uno ganas de probar con la versión real del juego...

tu Micro

ESTRUCTURA DEL PROGRAMA

20- 160	Portada de presentación del juego, confeccionada en la pantalla de gráficos.
180- 200	Consulta para mostrar las instrucciones del juego.
210- 320	Introducción del número y nombre de los jugadores.
330- 360	Inicialización de los datos fijos del juego.
370- 970	Bucle principal del programa.
980-1020	Subrutina auxiliar para dar tiempo a la lectura de los mensajes en pantalla.
1030-1170	Subrutina que proclama al vencedor del juego y ofrece la posibilidad de realizar una nueva partida.
1180-1230	Subrutina para la creación de los nombres de los participantes.
1240-1750	Definición de las instrucciones.
1760-1820	Subrutina para el salto de página.
1830-1890	Subrutina para definir la situación de la bala en el cargador.

CUADRO DE VARIABLES

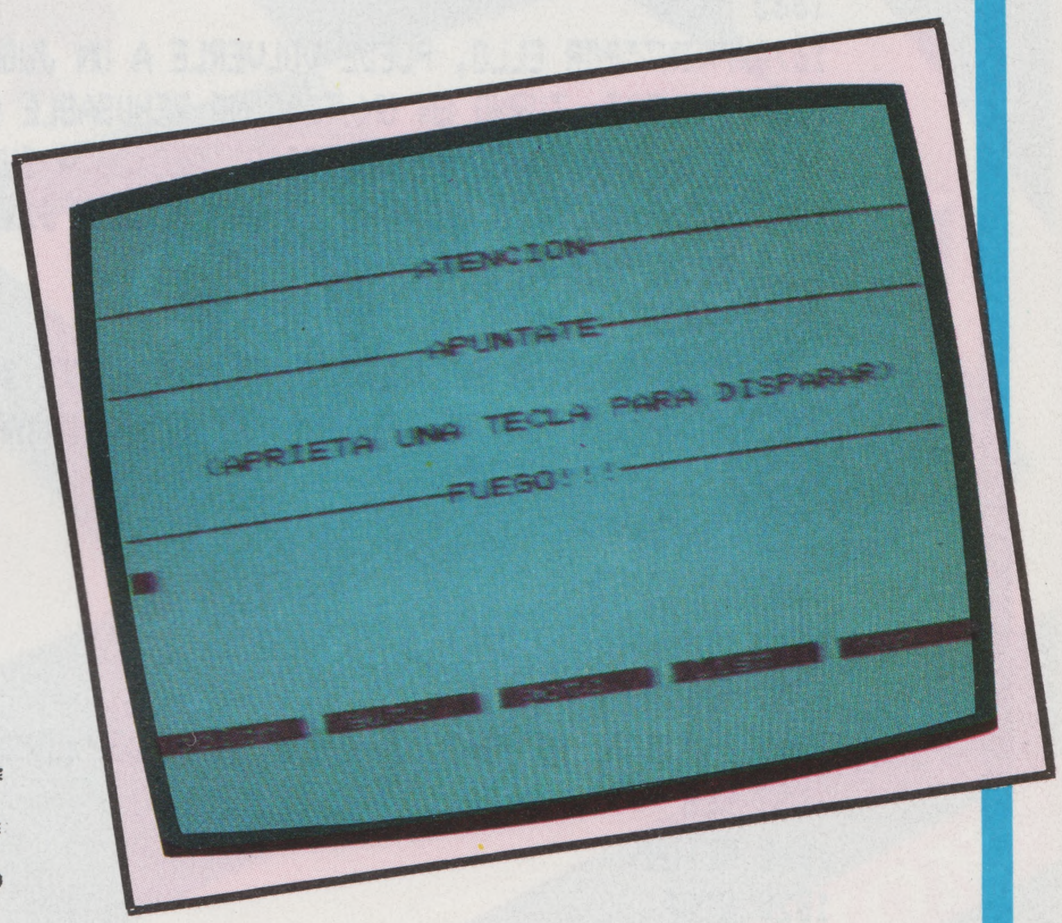
ARRI$	Variable para aceptar o rechazar el disparo.
BALA	Posición de la bala en el cargador.
CARGAD()	Matriz que define la situación del cargador
DISPAR	Número de disparos realizados.
EL	Número de jugadores eliminados.
ELIM(I)	Jugadores ya eliminados.
I, J, K	Indices para bucles FOR-NEXT.
INS$	Variable para la aceptación de la lectura de instrucciones.
JUG	Número de participantes en el juego.
JUG$()	Matriz para almacenar los nombres de los jugadores.
PIP	Variable para bucle de tiempo.
SIGU$	Variable para seguir o no al ser eliminado un jugador.
TIRO$	Instante del disparo.

LISTADO

```
10 REM CARLOS RUBIO
20 SCREEN 1
30 FOR I = 1 TO 3
40 IF (I MOD 3)<>1 THEN  GOTO 60
50 PLAY "L3T3203ABCo4L3ADFE"
60 FOR J = 1 TO 14
70  COLOR 1,I+1,J
80 LOCATE 90,70
90 PRINT"LA RULETA RUSA"
100 LOCATE 20,160
110 PRINT"CARLOS RUBIO, 1984"
120 FOR K = 1 TO 10
130 PIP = 1/K
140 NEXT :NEXT
150 CLS :NEXT
160 CLS : COLOR 15,5,9
170 LOCATE 40,70
180 INPUT "QUIERES LEER LAS INSTRUCCIONES,(S/N)";INS$
190 IF ((LEFT$(INS$,1)="S") OR (LEFT$(INS$,1)="s")) THEN  GOSUB 1240
```

```
200 IF ((LEFT$(INS$,1)="N") OR (LEFT$(INS$,1)="n")) THEN GOTO 210 ELSE GOTO 180
210 CLS: COLOR 15,4,5
220 INPUT"CUANTOS JUGADORES VAN A PARTICIPAR (DE 2 A 6)";JUG
230 IF (JUG)6 OR JUG(2) THEN  GOTO 220
240 DIM JUG$(JUG,39):PRINT:PRINT
250 FOR I=1 TO JUG
260 PRINT"JUGADOR NUMERO ";I;"DAME TU NOMBRE"
270 INPUT JUG$(I,1):PRINT
280 NEXT
290 CLS
300 PRINT"LUEGO VUESTROS DATOS SON"
310 PRINT
320 GOSUB 1190
330 FOR I=1 TO JUG
340 ELIM(I)=0
350 NEXT
360 DIM CARGAD(6)
370 GOSUB 1830
380 FOR I = 1 TO JUG
390 IF ELIM(I)=1 THEN GOTO 960
400 DISPAR = DISPAR+1
410 PRINT"JUGADOR";I;"QUIERES ARRIESGARTE A DISPA- RAR (S/N) ?"
420 INPUT ARRI$
430 IF (ARRI$="S" OR ARRI$="s") THEN  GOTO  480
440 IF (ARRI$="N" OR ARRI$="n") THEN GOTO 450 ELSE  GOTO 410
450 IF DISPA(4 THEN  GOTO  820
460 PRINT:PRINT"LO SIENTO, PERO TE HAS DE DISPARAR"
470 GOSUB 980
480 COLOR 1,7 :CLS
490 PRINT:PRINT:PRINT"---------------ATENCION---------------"
500 PRINT:PRINT:PRINT"---------------APUNTATE---------------"
510 PRINT:PRINT:PRINT"    (APRIETA UNA TECLA PARA DISPARAR)"
520 PRINT:PRINT:PRINT"---------------FUEGO!!!---------------"
530 TIRO$=INKEY$
540 IF TIRO$ ="" THEN  GOTO  530
550  COLOR 1,6
560 CLS
570 PRINT"               T";
580 FOR J=1 TO 1000
590 J=J+1
600 IF J MOD 100=0 THEN PRINT"R";
610 NEXT
620 IF CARGAD(DISPAR)=0 THEN GOTO 870
630 CLS: COLOR 15,1:
640 LOCATE 12,10
650 PRINT"PUM!!!!!!!!!!!!!"
660 PLAY"O4T250L64M60000V15D"
670 GOSUB 980
680 PRINT"LO SIENTO JUGADOR ";I;" HAS MUERTO Y POR TANTO QUEDAS ELIMINADO DEL JUEGO"
690 ELIMI(I)=1
700 FOR K=1 TO JUG
710 EL=ELIMI(K)+EL
720 NEXT
730 IF EL=JUG-1 THEN GOSUB 1030
740 PRINT"QUEREIS SEGUIR ARRIESGANDOOS LOS QUE"
750 PRINT"QUEDAIS"
760 INPUT SIGU$
770 IF (SIGU$="N" OR SIGU$="n")THEN STOP
780 CLS
```

```
790 I=0:J=0:K=0: COLOR 15,4
800  GOSUB 1830
810 GOTO 380
820 DISPAR=DISPAR-1
830 PRINT"DE ACUERDO,PASA EL REVOLVER A JUGADOR"
840 PRINT"SIGUIENTE":PRINT
850 GOSUB 980
860  GOTO 960
870  COLOR 1,7
880 LOCATE 12,10
890 PRINT"CLICK!!!!!!!!!!!!"
900 PLAY"S206A"
910 LOCATE 10,20
920 PRINT"UFFFFFFF!!!!!!!!!!"
930 GOSUB 980
940 CLS
950  GOTO 960
960 IF I=JUG THEN I = 0
970 NEXT
980 FOR K=1 TO 300
990 PIP=1/K
1000 NEXT
1010 CLS
1020 RETURN
1030 GOSUB 980
1040 LOCATE 6,10
1050 PLAY"V703ABDEF"
1060 PRINT"ENHORABUENA,CAMPEON!!!!!!!!!!"
1070  COLOR 1,5:GOSUB 980:CLS
1080 FOR R=1 TO JUG
1090 IF ELIMI(R)=1 THEN 1130
1100 LOCATE 10,10
1110 PLAY"04BGADE"
1120 PRINT JUG$(R,1)",ERES EL MEJOR."
1130 NEXT
1140 LOCATE 15,12
1150 INPUT"OTRA PARTIDA ";JUE$
1160 IF (JUE$="S" OR JUE$="s")THEN  GOTO 1180
1170 STOP
1180 ERASE JUG$:ERASE CARGAD: GOTO 20
1190 FOR I = 1 TO JUG
1200 PRINT"JUGADOR ";I;" NOMBRE: " JUG$(I,1):PRINT
1210 NEXT
1220 GOSUB 980
1230 RETURN
1240 CLS
1250 COLOR 4,10,9
1260 PRINT"                                        "
1270 PRINT"              LA RULETA RUSA            "
1280 PRINT"                                        "
1290 PRINT:PRINT:PRINT
1300 PRINT"LA RULETA RUSA ES UN JUEGO DE 2 A 6"
1310 PRINT"JUGADORES, (NO PUEDE SER 1 PUES SERIA"
1320 PRINT"COMO JUGAR A UN SUICIDIO)."
1330 PRINT
1340 PRINT"EL JUEGO PARTE DE LA DISPOSICION DE"
1350 PRINT"UN HIPOTETICO REVOLVER CON CAPACIDAD"
1360 PRINT"PARA 6 BALAS EN EL CUAL SOLO SE"
1370 PRINT"INTRODUCE UNA. A CONTINUACION SE HACE"
```



```
1375 PRINT"GIRAR EL TAMBOR."
1380 PRINT:PRINT:PRINT
1390 GOSUB 1790
1400 PRINT"COMIENZA LA PARTIDA. CADA JUGADOR"
1410 PRINT"PUEDE OPTAR POR DISPARARSE O NO."
1420 PRINT
1430 PRINT"SI RECHAZA LA OPCION PASA EL TURNO AL"
1440 PRINT"SIGUIENTE. SI DECIDE JUGAR PODRAN"
1450 PRINT"PASAR 2 COSAS:"
1460 PRINT
1470 PRINT"PRIMERA, QUE ESA POSICION FUERA LA DE"
1480 PRINT"LA BALA,CON LO QUE EL JUGADOR MORIRIA"
1490 PRINT"Y CONCLUIRA SU JUEGO."
1500 PRINT
1510 PRINT"SEGUNDA, QUE NO ESTUVIERA AHI, CON LO"
1520 PRINT"QUE NO MORIRIA Y PASARIA EL TURNO AL"
1530 PRINT"SIGUIENTE JUGADOR."
1540 PRINT:
1550 PRINT"UNA VEZ PASADO ESTE TURNO SE REPITE"
1560 PRINT"DE NUEVO EL PROCESO."
1570 PRINT:PRINT:PRINT:PRINT
1580 GOSUB 1790
1590 PRINT"HAY QUE CONSIDERAR LAS POSIBILIDADES"
1600 PRINT"DE MORIR AL RECHAZAR LOS DISPAROS,"
1610 PRINT"PUES, A MAS DISPAROS EFECTUADOS,MAYOR"
1620 PRINT"ES LA PROBABILADA DE QUE SE DISPARE"
1625 PRINT"EL ARMA."
1627 PRINT
1630 PRINT"PARA EVITAR PROBLEMAS CON LOS LISTILLOS"
1640 PRINT"TAMBIEN ES REGLA EL QUE NO SE PUEDE"
1650 PRINT"RECHAZAR EL DISPARARSE A PARTIR DEL"
1660 PRINT"CUARTO DISPARO."
1665 PRINT
1670 PRINT"POR ELLO, PUEDE VOLVERLE A UN JUGADOR"
1680 PRINT"EL TURNO EN UN TIRO NO-REHUSABLE O EN"
1690 PRINT"UN MOMENTO PELIGROSO, DEBIDO A QUE"
1700 PRINT"RECHAZO UN TIRO EN UNA JUGADA SIN"
1710 PRINT"RIESGO."
1711 PRINT
1720 PRINT"GANARA EL JUEGO EL PARTICIPANTE QUE"
1730 PRINT"QUEDE VIVO, CUANDO LOS DEMAS HAYAN"
1740 PRINT"MUERTO."
1750 PRINT
1760 GOSUB 1790
1770 INS$="N"
1780 RETURN
1790 INPUT"PULSE [ENTER] PARA CONTINUAR";INS$
1800 CLS
1810 RETURN
1820 STOP
1830 BALA=1+(INT(RND(-TIME)*6))
1840 FOR I =1 TO 6
1850 CARGAD(I)=0
1860 NEXT
1870 CARGAD(BALA)=1
1880 DISPAR=0:EL=0
1890 RETURN
```

END

tu Micro

Y
PEEK & POKE

convocan la:

Segunda edición del concurso

EL PROGRAMA DEL LECTOR

Primer premio: un viaje a Londres, con tres días de estancia y visita a una Feria de Informática.
(El premio cubre el viaje, la estancia y la visita a la Feria de Informática a celebrar en el primer trimestre del próximo año).

FECHA TOPE PARA LA RECEPCION DE PROGRAMAS:
1 DE DICIEMBRE DE 1984

¡Vaya preparando sus programas!...

BASES DEL CONCURSO

1. EL PROGRAMA DEL LECTOR es un concurso de programas de convocatoria periódica, abierto a la participación de los lectores de la revista TU MICRO. Las presentes bases corresponden a la segunda convocatoria del concurso.

2. Los programas presentados a concurso por los lectores deben ser inéditos.

3. Cada concursante puede remitir tantos programas como desee.

4. Los programas deben enviarse listados en papel y grabados en un soporte magnético (casete, disco flexible...); una vez fallado el concurso, el soporte será devuelto al concursante. El envío se hará a TU MICRO (El programa del lector). Avda. de Alfonso XIII, 141. 28016 MADRID.

5. Los programas deben estar destinados a algún modelo de microordenador presente en el mercado nacional.

6. Los programas pueden ser de cualquier tipo (juegos, utilidades, gestión, educativos) y pueden estar escritos en cualquier lenguaje habitual en el campo de los microordenadores: código máquina, ensamblador, BASIC, FORTH, PASCAL, LOGO...

7. Cada programa presentado a concurso debe acompañarse de una ficha de presentación en la que deben figurar los siguientes datos:
   - Datos personales del concursante.
   - Nombre del programa.
   - Ordenador para el que está destinado.
   - Lenguaje en el que está escrito el programa.
   - Breve descripción del programa detallando las indicaciones necesarias para su ejecución.
   - Constancia por escrito de que el programa enviado a concurso es inédito.

8. La fecha tope para la recepción de los programas presentados a esta segunda convocatoria del concurso «EL PROGRAMA DEL LECTOR» será el próximo día 1 de diciembre.
   El fallo del jurado se publicará en la revista TU MICRO correspondiente al mes de enero de 1985.

9. Relación de premios:
   - 1.er PREMIO: un viaje a Londres, con tres días de estancia y visista a una Feria de Informática.
     (El premio cubre el viaje, la estancia y la visita a la Feria de Informática a celebrar en los primeros meses del próximo año).
   - 2.º PREMIO: 25.000 pesetas en material informático.
   - 1.er ACCESIT, PARA LOS 5 FINALISTAS: Colección ENCICLOPEDIA PRACTICA DE LA INFORMATICA (4 tomos).
   - 2.º ACCESIT, PARA LOS 10 SEMIFINALISTAS: Lote de 5 cintas TU MICRO/especial ordenador.

10. Los programas que se hagan acreedores al primer y segundo premios, se publicarán en las páginas de TU MICRO y quedarán como propiedad de la revista, renunciando sus autores a cualquier otra compensación distinta al premio.

11. Los programas seleccionados como finalistas y semifinalistas del concurso, serán publicados en TU MICRO, figurando el nombre del autor. Estos programas serán adquiridos por la editorial, aplicando la tarifa vigente.

12. El jurado decidirá sobre todos los aspectos no contemplados en estas bases y su decisión será inapelable.

DRAGON 64

Ordenador:
DRAGON 64

Fabricante:
Dragon Data Ltd.

Distribuidor:
IDS
Españoleto, 25
28004-Madrid



La compañía británica Dragon Data Ltd. ha puesto en el mercado al sucesor del popular DRAGON 32, el nuevo DRAGON 64. Un equipo que mantiene una íntima relación con su predecesor, relación que llega hasta el punto de que uno de los dos modos de funcionamiento que admite, el modo 32, conduce al DRAGON 64 a un comportamiento idéntico al de su digno precedente.

Una primera ojeada a la máquina puede conducir incluso a la confusión, ya que tanto el D32 como el neonato D64 se presentan en la misma caja. Una observación algo más atenta diluye el posible entuerto, al revelar una ligera diferencia en el color de ambas cajas. Desde luego, la diferencia resulta ya apreciable si conectamos el equipo y seleccionamos el modo 64.

Las analogías de los DRAGONES

Dado su íntimo parentesco, el análisis del DRAGON 64 pasa por relacionar, en primer término, las características comunes a ambas máquinas.

Ocupando el espacio superior de la caja, nos encontramos con un clásico teclado en disposición QWERTY. De las 53 teclas que lo integran, 26 son alfabéticas y 10 numéricas (todas ellas, excepto la tecla del cero, soportan además un signo); el repertorio alfanumérico se completa con 6 teclas de puntuación (con 2 signos por tecla), 4 «flechas», una tecla @, dos teclas SHIFT, la barra espaciadora y tres teclas de control: BREAK, CLEAR y ENTER.

En uno de los laterales del mueble, protegido frente a posibles activaciones accidentales, se encuentra el botón de

Del DRAGON 32 al 64: ¡nunca un parentesco fue tan acusado!

Pág. 46, Tu Micro N.º 6

RESET, mientras que el interruptor de puesta en marcha reside en el panel posterior.

Puesta en marcha

Una vez conectada la máquina, a través del alimentador que se entrega con el equipo, puede ya lanzar su dedo índice en busca del interruptor de encendido y colocarlo en la posición «ON». Acto seguido, hay que ocuparse de precisar la sintonía del receptor de TV, hasta conseguir una buena imagen en la pantalla. Como es habitual en esta categoría de equipos, el diálogo se establece bajo la batuta de un intérprete de lenguaje BASIC; en esta ocasión se trata de una versión del conocido BASIC Microsoft.

El primer mensaje que ocupa la pantalla pone al usuario al corriente de esta situación:

```
(C) 1982 DRAGON DATA LTD.
16K BASIC INTERPRETER 1.0
(C) 1982 BY MICROSOFT
```

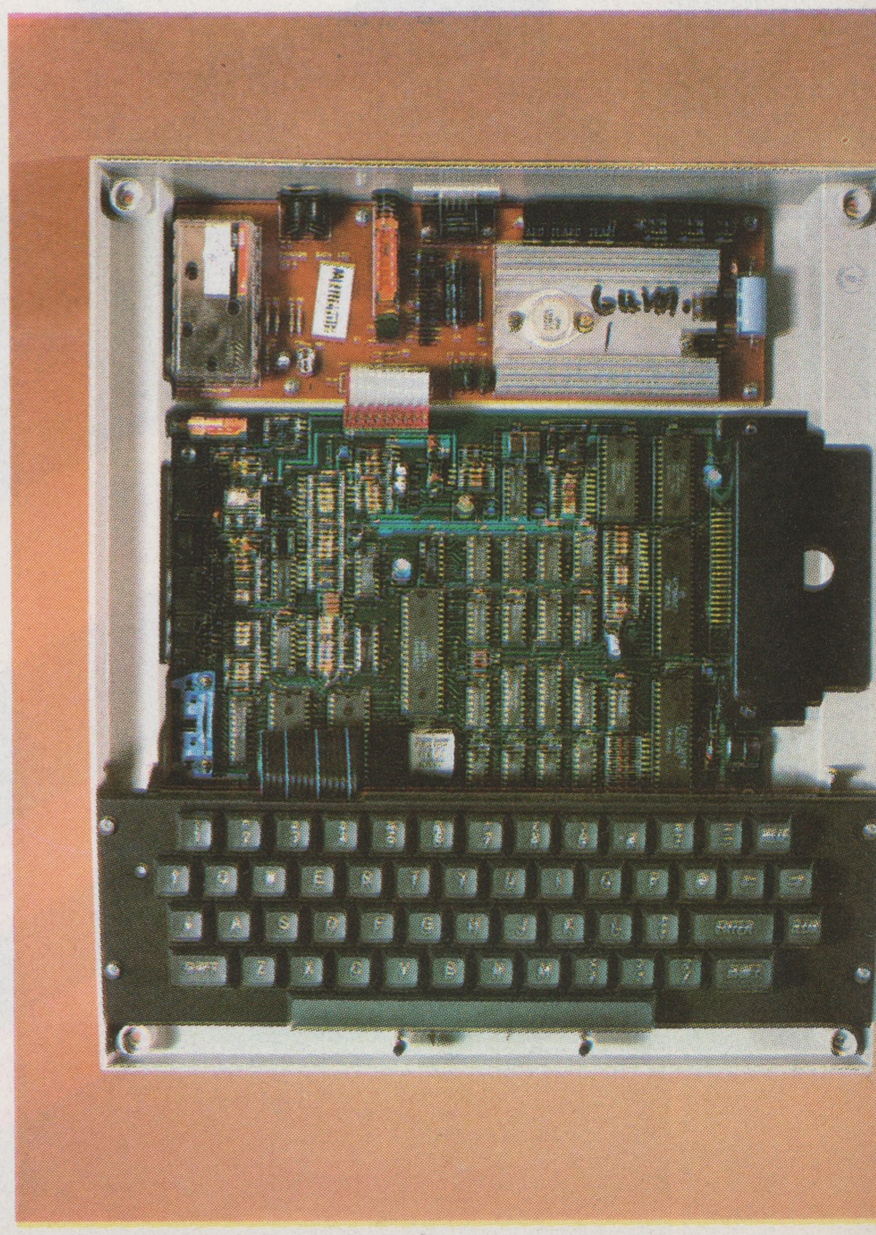
El contenido del mensaje de entrada revela una primera característica del equipo: el ordenador «formatea» la pantalla y escribe, automáticamente, en mayúsculas. De hecho, las minúsculas pueden aparecer en la impresora, e incluso pueden generarse en pantalla como caracteres, pero no desde el teclado. Cabe añadir que las teclas no están dotadas de «autorrepetición» EN EL MODO 32 y que —¡otra de arena!— el desplazamiento del cursor resulta, cuando menos, incómodo. En este punto, es significativo mencionar que la única flecha que realmente actúa sobre el cursor es la de retroceso («←») y, por mor de males, en su desplazamiento borra los caracteres sobre los que pasa. Si a los datos señalados le añadimos que, para el trabajo en modo texto, la pantalla del D32 o del D64 en modo 32 sólo cuenta con 16 líneas de 32 caracteres, no es gratuito pensar que el tratamiento de textos no es uno de los puntos fuertes del DRAGON.

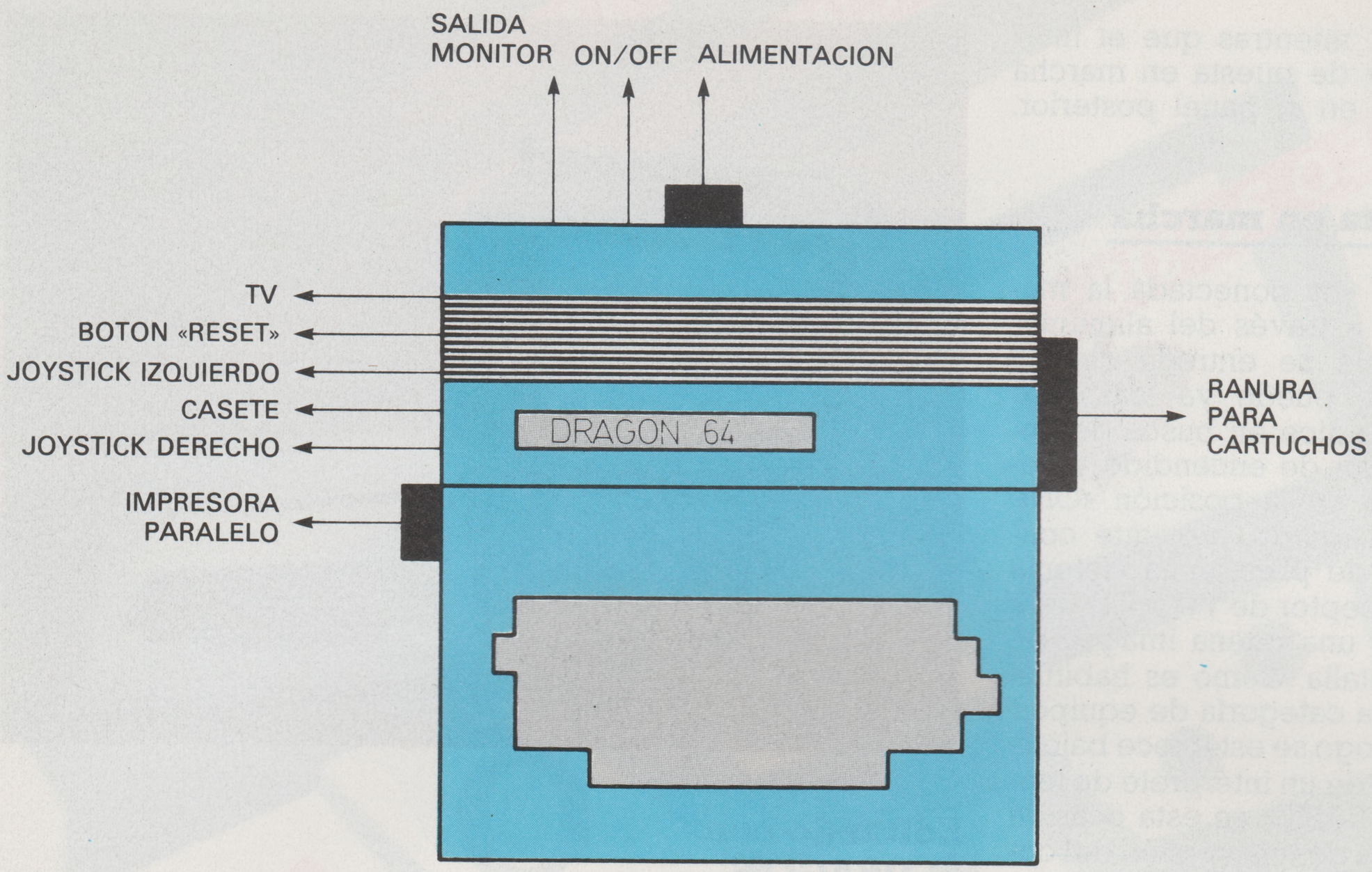


Editando con el DRAGON

Pero sigamos enunciando características comunes a ambos modelos. La edición en pantalla se organiza en torno al grupo de comandos al efecto que suelen compartir la mayor parte de los microordenadores de esta categoría. Así, por ejemplo, el comando LIST abre la posibilidad de listar total o parcialmente los programas residentes en RAM, a la vez que reordena el emplazamiento de las líneas de instrucción. El borrado de la pantalla se consigue con una acción directa sobre la tecla CLEAR, mientras que la detención de un programa en curso de ejecución se logra pulsando la tecla BREAK. Las dos teclas SHIFT mantienen el funcionamiento habitual y el clásico RETURN se ve sustituido por la tecla de control ENTER.

La corrección de los programas se convierte en una tarea un tanto engorrosa, al menos para el usuario no familiarizado con el equipo. En principio, no es posible ordenar al cursor que pase a la línea superior y tampoco puede llevarse el cursor a la zona de listado, de ahí que la corrección de líneas de programa haya que realizarla sobre una edición parcial de la línea a modificar. El comando EDIT es el encargado de facilitar esta tarea. No cabe duda que siem-





Distribución de conectores para comunicación externa del Dragón-64.

pre es más sencillo y rápido corregir un programa desplazando el cursor a través del listado que editando líneas parciales; no obstante, hay que señalar en este caso la potencia del comando EDIT. Este contempla todas las posibilidades de corrección, de tal forma que, aunque su empleo parezca arduo y complejo en un principio, llega a constituirse, con la práctica, en una potente herramienta de edición.

No podemos dejar en el olvido al comando PRINT, reemplazable en el DRAGON por el signo de interrogación "?". El empleo asociado de PRINT con el símbolo @ facilita el posicionamiento del cursor en cualquier punto de la pantalla.

RENUM es uno de los comandos predilectos de los aficionados a la programación que podemos encontrar en el repertorio del DRAGON. Además de su función habitual en la renumerción de las líneas de un programa, su empleo asociado a «RUN nº de línea» resulta de gran utilidad, por ejemplo, para «chequear subrutinas».

Cabe añadir que para la puesta a punto de programas se dispone de los comandos tradicionales para este menester: TRON y TROFF. Por lo demás, el editor incluye un total de 24 mensajes de error; mensajes que reflejan el número de línea en el que se detecta la situación anómala, pero que resultan incómodos para el operador al aparecer en pantalla codificados. El repertorio de mensajes es insuficiente, hasta el punto de enmudecer ante errores tan frecuentes como la presencia de un GOSUB sin el correspondiente RETURN o de un FOR sin NEXT. En función de este último comentario, no debe sorprender el hecho de que el equipo no posea ningún comando específico para el tratamiento de errores.

Pantalla y gráficos

Este parece ser uno de los aspectos más destacables de la saga DRAGON. Si el usuario es un entusiasta de los programas de juego y de la animación de gráficos, podrá disfrutar poniendo en práctica las posibilidades que brindan ambos modelos.

Son varias las modalidades de presentación; una de ellas, ya comentada, es el modo texto, con una distribución en

pantalla de 16 líneas de 32 caracteres. El juego de caracteres directamente utilizable puede examinarse, en su totalidad, sin más que ejecutar las siguientes líneas de programa:

```
10 FOR I=1 TO 255
20 PRINT CHR$(I);" ";
30 NEXT I
```

Los caracteres que van del 1 al 32 corresponden a comandos de control y no generan visualización alguna en pantalla. Los 96 restantes (hasta completar un total de 127) son ya, en efecto, símbolos gráficos; estos coinciden con los introducibles a través del teclado, además de las minúsculas y algunos caracteres adicionales. A continuación, se encuentran 128 caracteres gráficos de tipo «bloque»; aunque, en honor a la verdad, este bloque está ocupado por un grupo de únicamente 16 caracteres distintos, repetido ocho veces, una en cada color generable por el DRAGON.

Completando al modo texto existen varios tipos de pantallas gráficas. La primera, de baja resolución, distribuye el espacio utilizable en 32×64 células y admite la programación de efectos de desplazamiento basados en el uso adecuado de los comandos SET y RESET.

La espectacularidad llega con la pantalla de alta resolución, en la que el usuario puede operar con una distribución mínima 128×96 y máxima de 256×192 puntos. La tabla adjunta resume las cinco variantes que admite el modo de alta resolución y revela las alternativas de color propias de cada modalidad.

El trabajo en la pantalla de alta resolución se realiza esgrimiendo los comandos PSET y PRESET, además del repertorio de comandos típicos para el trazado de presentaciones gráficas: LINE (dibujo de líneas), CIRCLE (confección de círculos y elipses) y un potente DRAW, con el que es posible eslabonar una cadena de puntos a través de la pantalla y «sin levantar el lápiz». El toque colorista lo pone el comando PAINT, adecuado para «rellenar» figuras con el color que elija el usuario.

Hay que mencionar aquí varios aspectos positivos del DRAGON. Uno de ellos es la posibilidad de cambiar páginas e incluso mezclarlas con el comando PCOPY. También hay que reseñar la facilidad con la que puede gestionarse el desplazamiento de figuras, por complejas que éstas sean, recurriendo a los comandos GET y PUT.

La mayor parte de los ordenadores encuadrados en el mismo segmento que el DRAGON, poseen una RAM de visualización fija; esto no sucede con el equipo que nos ocupa, puesto que la zona de RAM-vídeo es dimensionable y, por ende, adaptable a cada necesidad específica. La zona que puede utilizarse al efecto se distribuye en 8 páginas de 1.536 bytes cada una. Tal posibilidad permite optimizar el aprovechamiento del volu-



TABLA DE COMANDOS BASIC

GRAFICOS	SONIDO	I/O	EDICION Y CONTROL
POINT	MOTOR ON	PRINT USING	PRINT
CLEAR	MOTOR OFF	OPEN	PPRINT
SET	AUDIO ON	PRINT	BREAK
RESET	AUDIO OFF	CLOSE	CLS
PCLEAR	SOUND	CSAVE	CONT
PSET	PLAY	CLOAD	DEL
PRESET		SKIPF	EDIT
PMODE	**FUNCIONES**	INPUT	EXEC
SCREEN	ABS	LINE INPUT	LIST
PPOINT	ATN	INKEY$	NEW
LINE	COS	POS	REM
PAINT	EXP	JOYSTK	RENUM
CIRCLE	FIX		RUN
DRAW	INT	**CADENAS**	STOP
PCOPY	LOG	HEX$	TRON
GET	RND	STR$	TROFF
PUT	SGN	LEFT$	END
COLOR	SIN	MID$	CHR$
	SQR	RIGHT$	
MAQUINA	TAN	STRING$	**PROGRAMA**
DEF USR	DEF FN	ASC$	GOTO
PEEK		INSTR	ON...GOTO
POKE		LEN	GOSUB
VARPTR		VAL	ON...GOSUB
CSAVEM			RETURN
CLOADM			READ
EXEC			DATA
MEM			RESTORE
TIMER			FOR...TO...STEP...NEXT
			IF...THEN...ELSE

CASIO

FP-200

COMPUTADORA PERSONAL PORTATIL

LANZA AL MERCADO SU PRIMER ORDENADOR PERSONAL PORTATIL



...y a un precio increíble: 79.500 pts.

—Juegos II
- Slalom
- Batalla naval
- Muertos y heridos

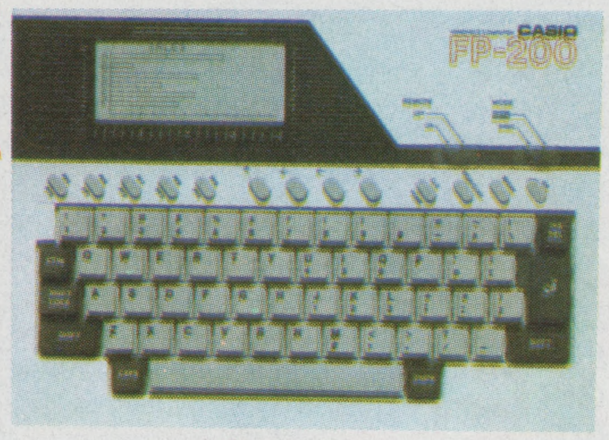
—Disassembler y Volcado Hexadecimal

—Juegos I
- La gallina saltarina
- Batalla de tanques
- El gusano loco

—Matemáticas II
- Derivadas 1.ª y 2.ª
- Integrales
- Máximos y mínimos
- Resolución de f(x)=0

—Tratamiento de textos I
—Tratamiento de textos II
Con gestión del archivo de direcciones.

CETL	Módulo ROM de 8k bytes para ampliar el lenguaje CETL del FP-200
Base de datos	Disquette para la creación de archivos
Gráficos	Cassette para potenciar las posibilidades gráficas del FP-200
Estadística	Cassette para el campo estadístico.



—Dietario y Agenda
—Agenda y Contabilidad
—Dietario, Agenda y Contabilidad

—Matemáticas I
- Interpolación lineal
- Interpolación parabólica
- Resolución de ecuaciones
- Ajustes por mínimos cuadrados
- Polinomios de colocación
- Cambios de base

—Contabilidad de clientes
Con explotación, volúmenes 300 clientes y 1.000 asientos, aprox.

—Vídeo clubs (en preparación)
—Bingos (en preparación)

ARQUITECTURA
—Pilares metálicos
—Viga biapoyada
—Cerchas
—Calc. estructuras método Matricial (en preparación).

—Archivo de clientes y Base da Datos.
—Facturación I

—Facturación II
Con archivo de clientes 400 aprox.

IMPORTADOR EXCLUSIVO

OTESA

VENTA A PROFESIONALES

Miguel Yuste, 16 - MADRID-17
Tels. 745 34 66/33 00 - 204 59 76
Telex 22686 OTESA E

OTESA
10 AÑOS
1974-1984

ELCO-I

Electrónica de Consumo - I, S. A.

Venta a revendedores y tercer canal

VIRGEN DE LOURDES, 40 posterior - Nave 4
Tels. 405 02 00 - 405 02 61
M A D R I D -27

men total de memoria RAM incorporado al equipo.

Un último dato a señalar, aunque esta vez no muy loable, es que al operar en modo alta resolución no se reserva ni tan siquiera una línea para la presentación de texto.

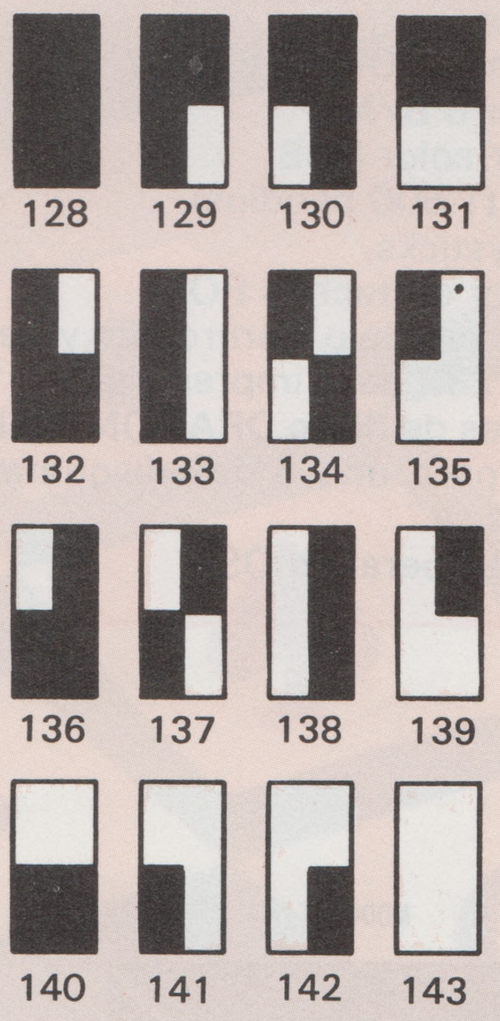
El rugir del DRAGON

Tampoco son desdeñables las posibilidades acústicas del DRAGON. Con comandos como "MOTOR ON (OFF)" y "AUDIO ON (OFF)", es posible controlar el motor y la salida audio de un casete para dotar a los programas de efectos musicales de fondo, etc. Para la generación de sonidos dentro del propio microordenador, están disponibles dos comandos: un sencillo «SOUND» que genera un tono puro (controla frecuencia del tono y duración, pero no volumen), y un completísimo «PLAY» con el que realmente se puede hacer música, ya que está capacitado para el control de nota (en nomenclatura sajona o cifra, admite ambas posibilidades) octava, duración, tempo, volumen e incluso de los silencios; hay que tener en cuenta que el mencionado comando no se utiliza nota a nota, sino con largas cadenas musicales. La escala cromática disponible, que se eleva a 5 octavas, puede recorrerse por medio de las siguientes instrucciones:

```
1Ø FOR I=1 TO 5:A$="O"+STR$(I)+";"
2Ø FOR J=1 TO 12:PLAY A$+STR$(I)
3Ø NEXT J,I
```

Quienes sepan algo de música no tardarán en apreciar las posibilidades que brinda el DRAGON; con un solo comando es capaz de interpretar una melodía completa, por compleja que ésta sea. Lamentablemente, el sistema no incorpora altavoz, por lo que es necesario conectarlo al receptor de TV o a un equipo exterior para utilizarlo como «organillo» e impresionar a los amigos con nuestra capacidad musical.

CARACTERES GRAFICOS DE TIPO «BLOQUE»



Los 16 caracteres gráficos de tipo «bloque» representados en la figura están disponibles en 8 colores distintos. Para generarlos debe utilizarse la función CHR$ seguida del código correspondiente. En representación normal los caracteres aparecen de color verde; el cambio de color se obtiene sumando los siguientes valores de C al código de caracter.
CHR$ (código de caracter + C).

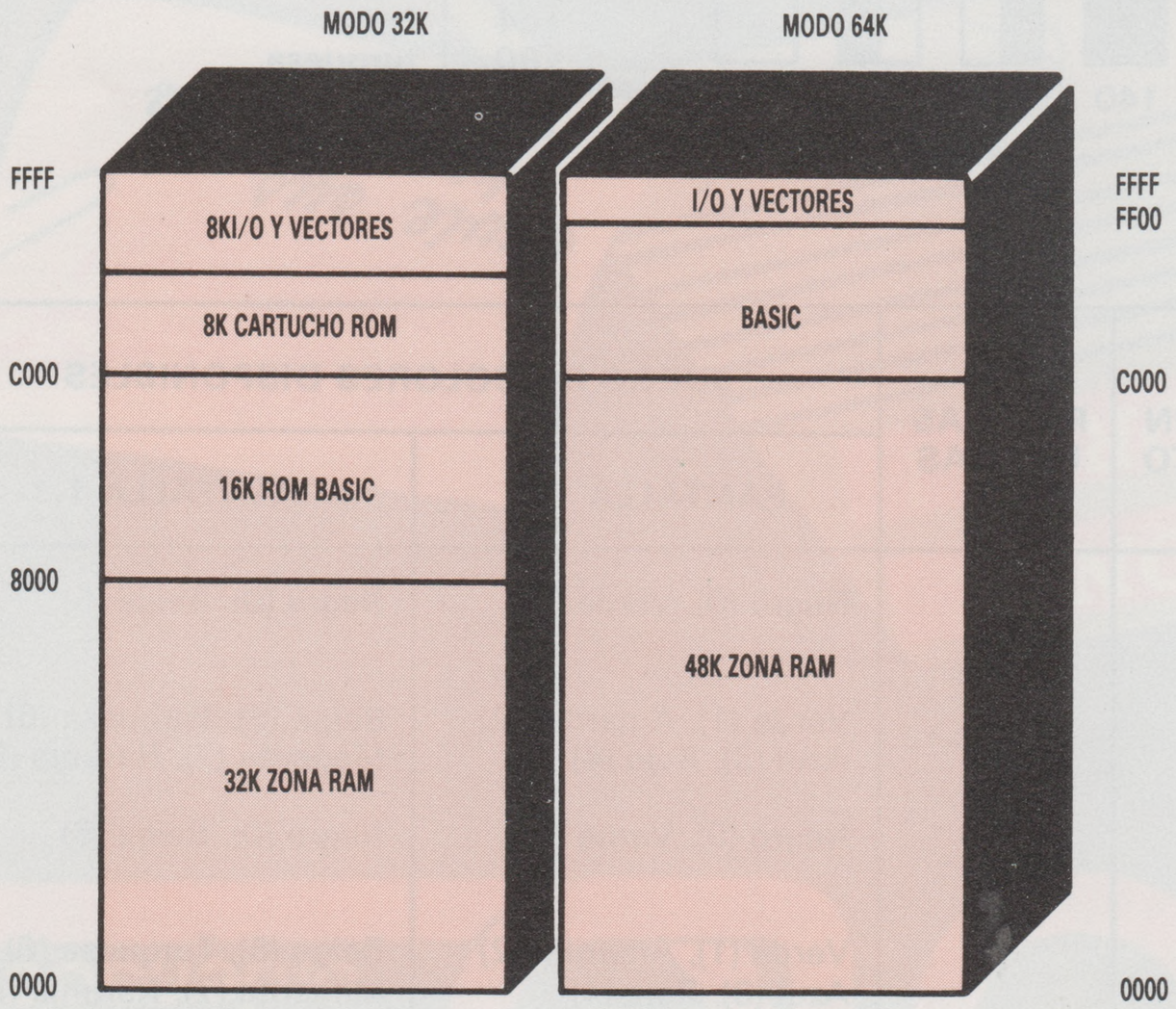
C	color
+16	amarillo
+32	azul
+48	rojo
+64	beige
+80	turquesa
+96	magenta
+112	naranja

NUMERO DE MODO	DIMENSION DEL PANEL	DIMENSION DEL PUNTO	PAGINAS USADAS	JUEGO DE COLORES DISPONIBLES	
				PANTALLA 1.0	PANTALLA 1.1
0	128 × 96	■ ■ ■ ■	1	Negro (0), Verde (1)	Negro (0), Beige (5)
1	128 × 96	■ ■ ■ ■	2	Verde (1), Amarillo (2) Azul (3), Rojo (4)	Beige (5), Turquesa (6) Magenta (7), Naranja (8)
2	192 × 128	■ ■	2	Negro (0), Verde (1)	Negro (0), Beige (5)
3	192 × 128	■ ■	4	Verde (1), Amarillo (2) Azul (3), Rojo (4)	Beige (5), Turquesa (6) Magenta (7), Naranja (8)
4	256 × 192	■	4	Negro (0), Verde (1)	Negro (0), Beige (5)

Modalidades de alta resolución seleccionables en ambos equipos: DRAGON 32 y DRAGON 64.

Características más significativas

Microprocesador	6809 (Motorola)
Lenguaje residente	BASIC Microsoft versionado
Memoria RAM	64 Kbytes.
Memoria ROM	16 Kbytes.
Teclado	53 teclas en disposición QWERTY: 26 teclas alfabéticas, 10 numéricas y 17 teclas adicionales para símbolos y control. 59 caracteres gráficos. Teclas BREAK, CLEAR y ENTER.
Editor	24 mensajes de error codificados. Funciones RENUM, RUN x, DEL y EDIT.
Resolución	Modalidad texto: 16×32 en modo 32 y 24×51 caracteres en modo 64. Alta resolución máxima de 192×256 puntos. Gráficos en color.
Caracteres y color	96 caracteres gráficos. 16 caracteres tipo «bloque». 8 colores.
Sonido	5 octavas. Funciones SOUND y PLAY.
Salidas y expansiones	TV color o B/N. Monitor color RGB. Casete (1.500 baudios). Dos Joysticks. Conector cartuchos ROM. Salidas paralelo Centronics y serie RS/232 para impresora. Unidades de disco DRAGON DISK DRIVE para uno o dos disquetes de 5 y 1/4". Sistema operativo OS9.



Mapa de memoria del Dragón-64 en ambos modos de funcionamiento modo 32 y modo 64.

Periféricos y expansiones

El DRAGON puede convertirse en un equipo muy completo. Admite la conexión de cualquier tipo de TV color o de monitor de alta resolución, así como de cualquier casete de tipo común. También incorpora una interface que permite la conexión directa de los joysticks, e incluso está preparado para recibir la conexión directa de cartuchos ROM con programas. Destinado a la impresora, posee una interface de tipo paralelo Centronics (el modelo dispone también de una interface serie RS/232). En todo caso, tal vez la posibilidad más interesante sea la de conectarle hasta cuatro unidades de disco flexible de 5 y 1/4". Esta conexión se establece a través del controlador de discos que en la propia placa de circuitería, aloja una memoria ROM con el sistema operativo. Aunque no llega a ser un verdadero DOS, sí aporta algunas ventajas importantes como, por ejemplo, comandos para tratamiento de errores. El modelo 64 ofrece otra importante posibilidad que comentamos más adelante.

Particularidades del DRAGON 64

Ha llegado el momento de ver las ventajas del hermano mayor. Como ya sabemos, el comportamiento del D64 en modo 32 coincide con el propio del D32; de ahí que todo lo comentado es de aplicación directa a ambos modelos. De hecho, cuando se conecta, el D64 empieza a funcionar en modo 32, si bien, para pasar al modo 64, basta con teclear el comando "EXEC".

El modo 64 evidencia las novedades que caracterizan al nuevo modelo: teclas autorrepetitivas e incremento de la RAM disponible, además de la presencia de una salida serie RS 232, como ya hemos visto. La figura adjunta ilustra los mapas de memoria asociados a ambos modos de funcionamiento; se observa que el

modo 64 «roba» los 8K para cartuchos y reubica en RAM los 16K del intérprete BASIC. Esta particularidad, difícil de conseguir con otros micros como el Z80 o 6502, ofrece otras dos ventajas aparte del incremento de memoria: al estar ahora el BASIC en zona RAM, podemos redefinir nuestros propios comandos, e incluso podemos borrar una parte o todo el intérprete para aprovechar así este espacio de memoria en otros menesteres. Y frente a estas importantísimas ventajas, está la más esencial: el modo 64 es capaz de soportar un auténtico sistema operativo en disco (DOS), el OS9, que aun sin gozar de popularidad que otros SO, brinda excelentes posibilidades al programador; por ejemplo, la pantalla de textos pasa a ser de 24×51 caracteres, con mayúsculas y minúsculas, lo que facilita su aplicación al tratamiento de textos. La única «pega» del modo 64 es que no admite el trabajo con cartuchos al «absorber» la memoria necesaria; pero esto es un mal menor, puesto que, en caso necesario, se puede volver tranquilamente al modo 32.

Conclusiones

Ambos modelos DRAGON son equipos con notables posibilidades gráficas y sonoras que pueden hacer las delicias de los amantes de los juegos. En su versión base (sin expandir) presentan un editor con ciertas limitaciones, que dificulta la confección de programas complejos; no obstante, estas limitaciones pueden superarse con creces si el usuario está dispuesto a aprovechar la máquina utilizando los sistemas operativos de discos disponibles, sobre todo el OS9 para el D64, que convierte al DRAGON 64 en un potente equipo.

Por lo demás, cabe añadir que todo el software existente para el 32 es aplicable al 64, sin problemas; incluso el manual es compartido por ambos modelos, completándose en el caso del DRAGON 64 con un

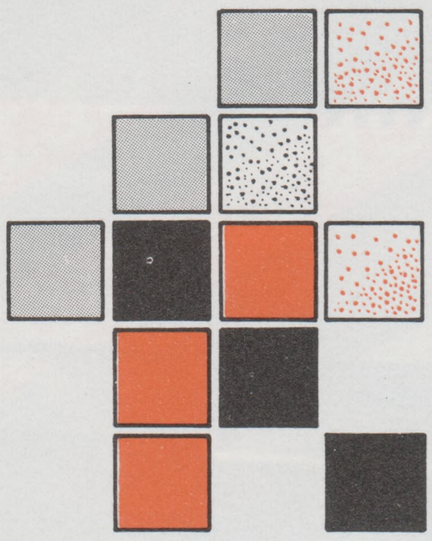


cuadernillo a modo de apéndice. El manual está redactado en castellano y constituye una buena guía de introducción a la programación BASIC de la mano del DRAGON, con un buen número de ejemplos demostrativos. Sin entrar en excesivas profundidades, contiene información suficiente para aprender y «soltarse» en los diálogos en BASIC, aunque, bien es cierto que puede parecer un poco sumarial para los más avezados en el tema.

tu Micro

El precio del sistema

El modelo DRAGON 32 es ya un viejo ocupante de los escaparates de las tiendas de ordenadores; su precio actual oscila en torno a las 46.900 ptas., mientras que el del nuevo DRAGON 64 asciende a 67.900 ptas. En ambos casos, con la unidad central se entrega el alimentador, los cables de TV y casete y el manual. A partir de aquí, puede ya entrar en funcionamiento al conectarle cualquier receptor doméstico de TV, en color o B/N, y un simple magnetófono a casetes convencional... ¡por el momento, no necesita soltar ni un duro más! Sin embargo, si le interesan los joysticks deberá pagar unas 7.300 ptas. por la pareja, aunque sólo en el caso del DRAGON 32, puesto que éstos se entregan acompañando al DRAGON 64. Puede optar por cualquier impresora de tipo paralelo, con interface Centronics (o RS 232 si se trata del D64), por lo que en este capítulo el desembolso dependerá de sus propias exigencias (un precio orientativo para una impresora media puede ser de 60.000 ptas.). Asimismo, si quiere obtener una imagen óptima y decide adquirir un monitor RGB, le valdrá cualquiera de los que hay en el mercado (precio indicativo de 60.000 ptas.). Si por fin decide expandir el equipo para completar un sistema microordenador, debe saber que la unidad DRAGON DISK DRIVE, con uno o dos lectores de disco (controlador incluido), se pone en las 72.500 ó 120.000 ptas. respectivamente, mientras que el precio del sistema operativo en disco para el DRAGON 64, el OS9, asciende a las 9.000 ptas.



DISPLAY

Una utilidad en código máquina para solucionar las «desavenencias» entre la pantalla del SPECTRUM y el programador.

Una curiosa organización

La pantalla del ZX-SPECTRUM tiene la peculiaridad de estar dividida en tres tercios (sin duda debido a razones «duras»... razones *Hard* queremos decir); cada uno de estos sectores se encuentra, a su vez, organizado de una manera un tanto caótica, lo que nos permite asegurar, sin temor a exagerar, que semejante pantalla es casi el sueño de un programador loco.

La escritura en esta pantalla en permanente alta resolución, se convierte en algo complejo si no se desea recurrir al PRINT del sistema, y la verdad es que resignarse a no poder mover un carácter pixel a pixel por la pantalla y tener que hacerlo a saltos de 8 en 8, se hace bastante cuesta arriba; máxime cuando la alta resolución nos quita, nada más y nada menos, que 6 K... ¡Demasiada memoria para no aprovechar sus posibilidades!

Pues bien, la subrutina que hoy presentamos permitirá escribir cualquier caracter en la pantalla (incluso en las dos líneas reservadas al sistema), con la particularidad de indicar el lugar de impresión en pixels (alta resolución), dando la posición del punto superior izquierdo del caracter a escribir.

Características de la subrutina

En una subrutina como la nuestra, de una relativa complejidad, se pueden seguir diferentes caminos, implementando unas u otras características. Por ello, es posible que encuentren otras aplicaciones destinadas a este fin, de mayor o menor longitud, más rápidas o más lentas... distintas, en definitiva.

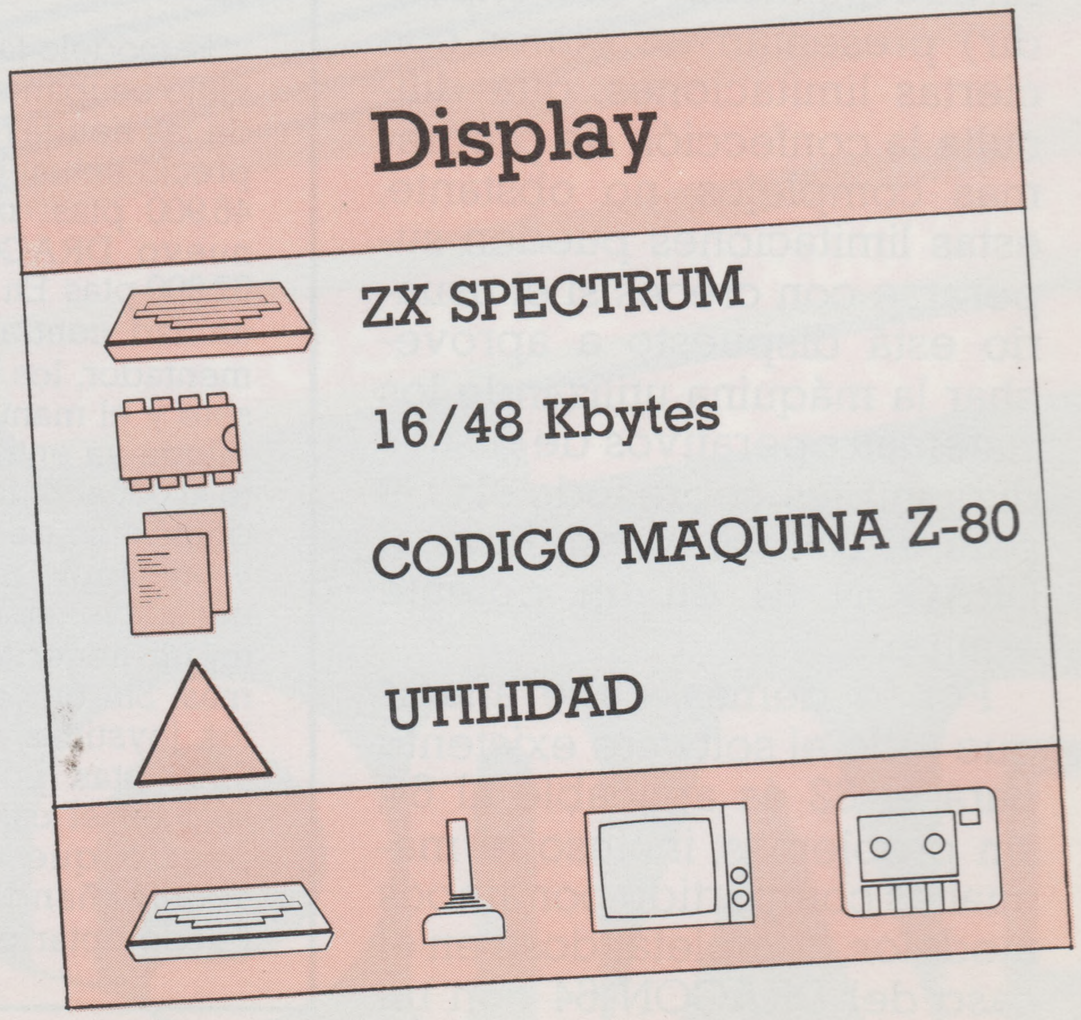
En nuestro caso, el programa se caracteriza por las siguientes particularidades:

- *Un sencillo sistema de inicialización de datos.*

Con sólo tres POKEs se indican el código ASCII del caracter a representar, la fila de representación y la columna de representación.

- *Flexibilidad absoluta en el juego de caracteres.*

Algunas subrutinas con semejante finalidad, no pueden representar los caracteres gráficos o los gráficos definidos del usuario. Nuestro programa no sólo contempla y resuelve estas dos posibilidades, sino que también tiene en cuenta el hecho de que el generador principal de



LISTADO

(LISTADO-1)

```
32000   254   32    48    4     62    32
32006   24    6     254   165   56    2
32012   62    32    254   144   245   48
32018   25    254   128   48    11    58
32024   54    92    111   58    55    92
32030   103   241   24    23    71    205
32036   56    11    33    146   92    241
32042   24    32    58    123   92    111
32048   58    124   92    103   37    241
32054   6     112   144   213   22    0
32060   95    203   35    203   18    203
32066   35    203   18    203   35    203
32072   18    237   90    209   62    8
32078   213   213   245   62    127   186
32084   48    10    1     0     80    122
32090   22    128   146   87    24    18
32096   62    63    186   48    10    1
32102   0     72    122   22    64    146
32108   87    24    3     1     0     64
32114   229   213   96    105   122   14
32120   8     6     0     145   56    3
32126   4     24    250   129   4     5
32132   40    8     17    32    0     167
32138   237   90    16    252   71    4
32144   5     40    3     36    16    253
32150   209   123   14    0     6     8
32156   144   56    5     12    24    250
32162   24    171   128   6     0     9
32168   22    8     95    68    77    225
32174   197   175   187   40    8     67
32180   126   203   63    16    252   24
32186   1     126   193   2     175   187
32192   40    14    197   62    8     147
32198   71    126   203   39    16    252
32204   193   3     2     11    241   209
32210   61    40    4     20    35    24
32216   201   209   122   6     0     14
32222   8     145   56    3     4     24
32228   250   129   79    123   22    0
32234   30    8     147   56    3     20
32240   24    250   131   95    33    0
32246   88    175   58    141   92    4
32252   5     40    9     213   17    32
32258   0     237   90    16    252   209
32264   20    21    40    4     66    35
32270   16    253   119   28    29    40
32276   3     35    119   43    12    13
32282   40    12    1     32    0     237
32288   74    119   28    29    40    2
32294   35    119   201   0     0     0
```

END

caracteres puede no encontrarse en su posición habitual en ROM (3D00-3FFF, hexadecimal), porque haya sido «desviado» a una zona de RAM por el usuario.

- *Total reubicabilidad.*

Este punto es muy importante, tanto para el usuario BASIC, cuanto más para el de código máquina. Gracias a ello, la subrutina puede ser situada en cualquier punto de la memoria, sin que se vea alterado su funcionamiento.

- Los caracteres de control o no representables *(TOKENs* de más de un caracter), cuyos códigos van del 0 al 31 y del 165 al 255, son sustituidos por un espacio.

LISTADO

(LISTADO-2)

```
 10 REM FERNANDO LOPEZ MARTINEZ # CARGADOR DE CODIGO MAQUINA # ZX-SPECTRUM
 20 LET B$=CHR$ 8+CHR$ 8+CHR$ 8+CHR$ 8+"   "+CHR$ 8+CHR$ 8+CHR$ 8: DEF FN P$()=
("   "+A$)(LEN A$ TO )
 30 POKE 23658,8: BORDER 1: PAPER 1: INK 7: CLS
 40 INPUT "DIRECCION DE INICIO ?";A: LET P=A
 50 PRINT A;TAB 7;
 60 FOR I=1 TO 6
 70 INPUT "BYTE ?"; LINE A$
 80 IF A$="F" THEN  GO TO 160
 90 IF A$="E" THEN  GO TO 230
100 IF VAL A$<0 OR VAL A$>255 THEN  GO TO 40
110 POKE A,VAL A$
120 LET A=A+1
130 PRINT FN P$();" ";
140 NEXT I: PRINT
150 GO TO 50
160 INPUT "NOMBRE DEL FICHERO ?";A$
170 IF A$="" THEN  GO TO 30
180 IF LEN A$>10 THEN  GO TO 160
190 SAVE A$CODE P,A-P+1
200 PRINT #0;"     PREPARE LA VERIFICACION"
210 VERIFY A$CODE P,A-P+1
220 RUN
230 IF I=1 THEN  PRINT AT 23-PEEK 23689,31;: LET I=7
240 PRINT B$;
250 LET A=A-1: LET I=I-1
260 GO TO 70
```

END

Manejo de «display»

La subrutina puede ser utilizada tanto desde el BASIC como desde el código máquina.

En el caso de que la subrutina sea controlada desde el código máquina, esta podrá situarse en cualquier punto de la memoria, accediendo a ella mediante un CALL a la primera dirección. Es necesario inicializar previamente el acumulador con el código ASCII del caracter a representar, el registro D con la fila (en pixels) a partir de la cual se construirá el caracter y el registro E con la columna. Tenga en cuenta que se considera como punto 0,0 la esquina superior izquierda de la pantalla.

Si la subrutina se controla desde el BASIC, habrá que realizar las mismas inicializaciones, para lo cual tendremos que añadir seis bytes al comienzo de la subrutina: un 62, el código del carácter a imprimir, un 22, la fila de escritura, un 30 y la columna de escritura; realizando el USR a la primera dirección.

Así, por ejemplo, si cargáramos la subrutina base a partir de 32000 habría que realizar los siguientes POKEs:

```
POKE 31994,62
POKE 31995, (código del caracter a imprimir)
POKE 31996,22
POKE 31997, (fila de impresión)
POKE 31998,30
POKE 31999, (columna de impresión)
```

y, a continuación, USR 31994. Siempre es conveniente proteger el código máquina de la sobreescritura por el BASIC, mediante un CLEAR a la dirección inmediatamente anterior al primer byte a proteger; en el ejemplo anterior la protección se garantizaría con 31993.

En otro orden, hay que destacar que la subrutina no comprueba si excede los límites de la pantalla, de ahí que para evitar desagradables sorpresas, sea necesario asegurar que el valor de fila esté comprendido entre 0 y 184, y el de columna entre 0 y 248.

Adopción del código máquina

El código máquina base está constituido por 297 bytes que aparecen en el listado 1, localizados a partir de la dirección 32000 (esta dirección de inicio es meramente orientativa, ya que al ser reubicable puede cargarse a partir de cualquier posición). Como quiera que introducir los 297 bytes a fuerza de POKEs puede resultar bastante «duro», en el listado 2 aparece el cargador de código máquina, ya publicado en el primer número de nuestra revista, que facilitará la tarea. Veamos cuál es el camino por andar:

—Introduzca el programa BASIC «cargador de código máquina» (listado 2).

—Ejecútelo con la orden RUN.

—A la pregunta "DIRECCION DE INICIO 7", responda 32000.

—Entre los 297 bytes de código máquina.

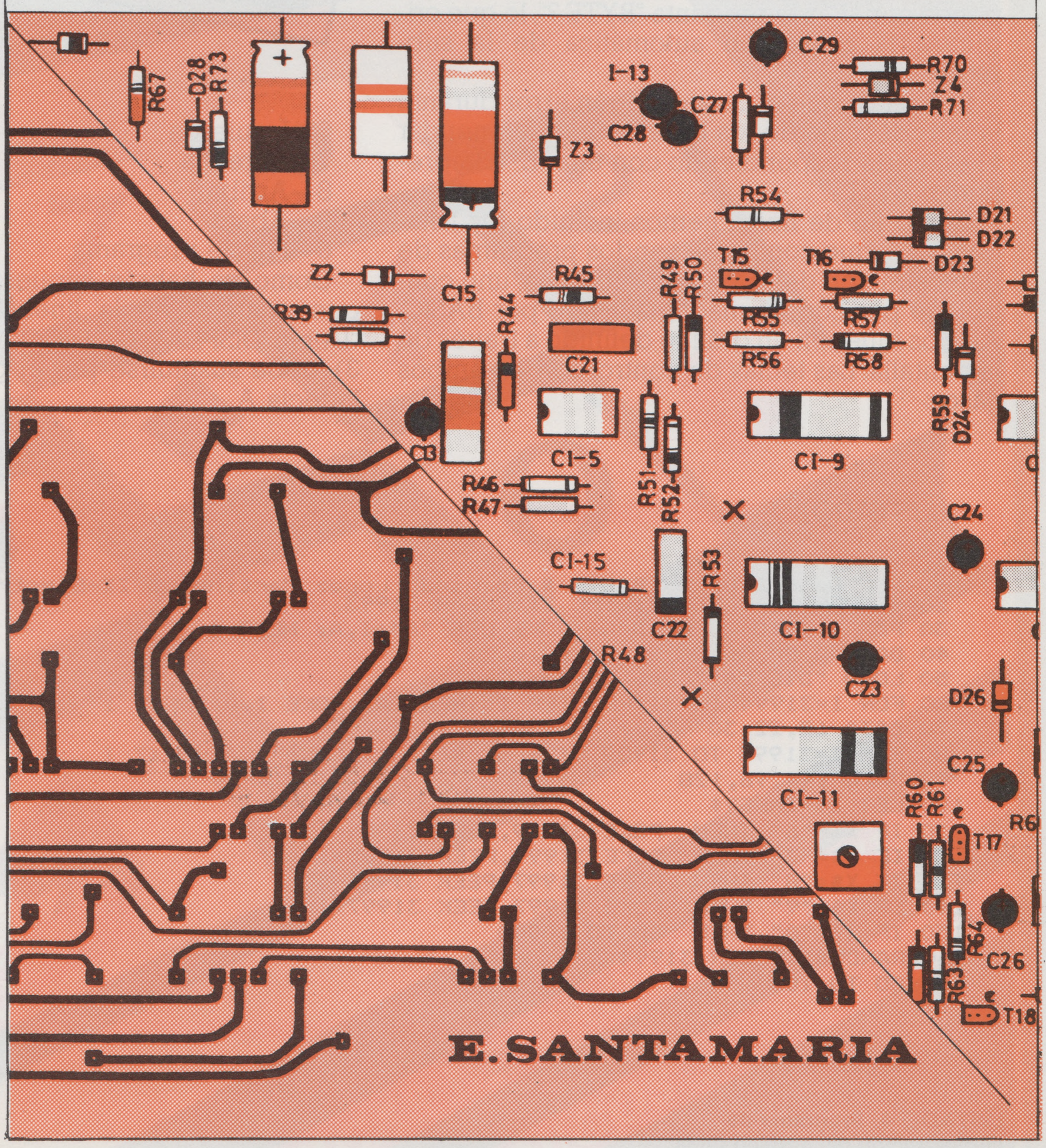
—Una vez introducido el último número, res-

¡A LA VENTA!

Manuales Técnicos Universitarios

Toda la electrónica digital:
desde el transistor hasta los
circuitos de muy alta escala de integración.

ELECTRONICA LOGICA Y MICROPROCESADORES



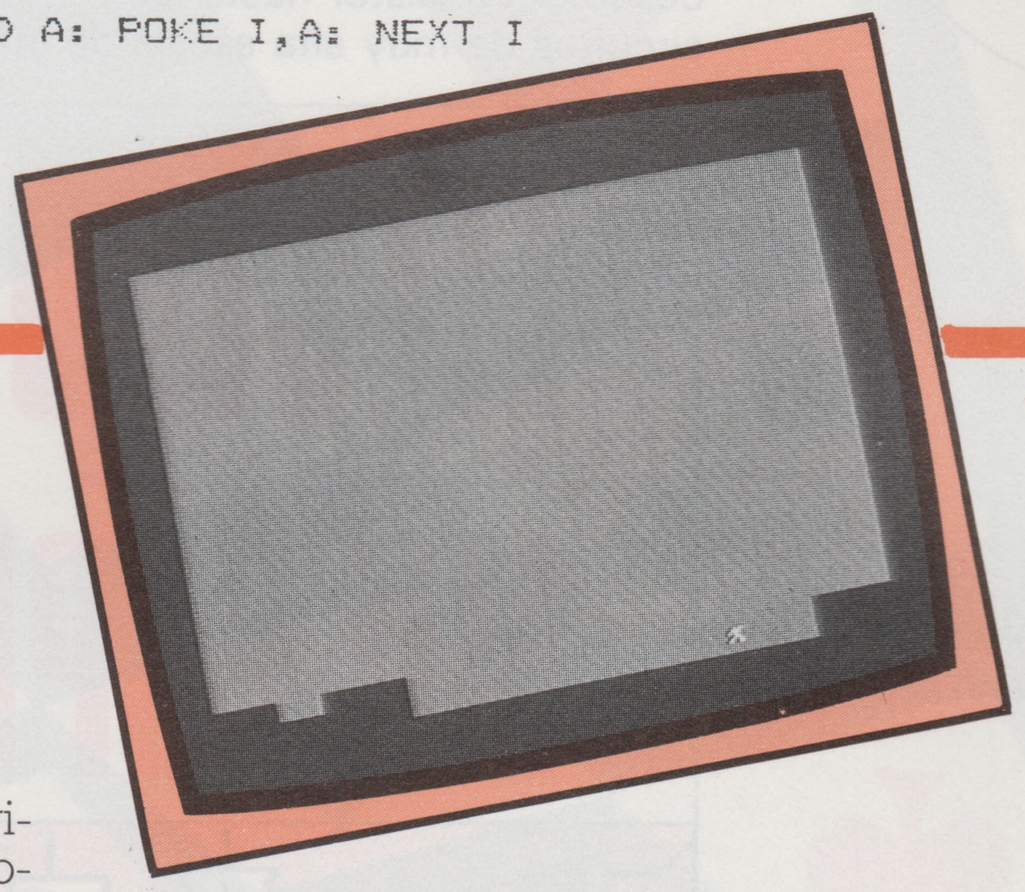
P.V.P.: 2.300 ptas.
Suscriptores: 2.000 ptas.

1.ª Parte
ELECTRONICA DIGITAL
Algebra de Boole
Puertas-Flip/flops
Contadores-Registros de desplazamiento-
Memorias-Conversión AD/ y D/A
2.ª Parte
MICROPROCESADORES
Teoría de los microprocesadores y estudio de los más importantes elementos de entrada/salida.

272 páginas

LISTADO

```
 10 REM Demo DISPLAY CODIGO MAQUINA # Fernando Lopez Martinez # ZX-Spectrum
 20 BORDER 1: PAPER 2: INK 7: CLEAR 31951
 30 FOR I=0 TO 7: READ A: POKE USR "A"+I,A: NEXT I
 40 DATA 0,20,56,80,24,36,68,0
 50 LOAD "DISPLAY"CODE 32000: CLS
 60 FOR I=31952 TO 31999: READ A: POKE I,A: NEXT I
 70 DATA 62,0,22,0,30,0,245,1,254,239,237,120,203,71,32,2,241,201,203,79,32,1,2
0,203,87,32,1,21,203,95,32,1,28,203,103,32,1,29,241,213,245,205,0,125,241,209,24
,214
 80 FOR I=31953 TO 31957 STEP 2: READ A: POKE I,A: NEXT I
 90 DATA 144,96,128
100 PRINT AT 12,16;"A"
110 RANDOMIZE USR 31952
```



LISTADO 3. Programa de demostración del empleo de la subrutina DISPLAY, controlado por código máquina.

ponda con "F" a la pregunta "BYTE ?", lo que originará la grabación automática de la subrutina.

Recuerde que si comete algún error durante la introducción de los bytes, puede enmendar el entuerto respondiendo con una "E" a la pregunta "BYTE ?".

Como fin de fiesta hemos incluido los listados 3 y 4, que contienen sendos programas de demostración del uso de la subrutina, uno controlado por código máquina y otro por BASIC. Ambos mueven un gráfico definido a través de la pantalla. Las teclas de control son, en ambos casos:

6—Izquierda
7—Derecha
8—Arriba
9—Abajo
0—Fin de programa.

Como ya habrán supuesto, la A subrayada que aparece en la línea 100 del programa "Demo DISPLAY CODIGO MAQUINA" (listado 3), debe sustituirse por el gráfico definido de la tecla A.

tuMicro

LISTADO

```
 10 REM Demo DISPLAY BASIC # Fernando Lopez Martinez # ZX-Spectrum
 20 BORDER 1: PAPER 2: INK 7: CLEAR 31993
 30 FOR I=0 TO 7: READ A: POKE USR "A"+I,A: NEXT I
 40 DATA 0,20,56,80,24,36,68,0
 50 LOAD "DISPLAY"CODE 32000: CLS
 60 FOR I=31994 TO 31998 STEP 2: READ A: POKE I,A: NEXT I
 70 DATA 62,22,30
 80 FOR I=31995 TO 31999 STEP 2: READ A: POKE I,A: NEXT I
 90 DATA 144,96,128
100 LET X$=INKEY$
110 IF X$="6" THEN  POKE 31999,PEEK 31999-1
120 IF X$="7" THEN  POKE 31999,PEEK 31999+1
130 IF X$="8" THEN  POKE 31997,PEEK 31997-1
140 IF X$="9" THEN  POKE 31997,PEEK 31997+1
150 IF X$="0" THEN  GO TO 10000
160 RANDOMIZE USR 31994
170 GO TO 100
```

LISTADO 4. Programa de demostración del empleo de la subrutina DISPLAY, controlado por BASIC.

tuMicro anuncios breves

TEXTO DEL ANUNCIO

Escriba de forma clara y en mayúsculas una sola letra por casilla. No olvide indicar su nombre y dirección o teléfono en la zona de datos personales. Evite abreviaturas. No olvide poner en el sobre las siglas ABS.

DATOS PERSONALES

Nombre: .....

Dirección: .....

.....

El servicio de ANUNCIOS BREVES se rige por las normas que se detallan en la página de «Servicios TU MICRO».

tuMicro suscripciones

Apellidos:
Nombre:
Domicilio:
Localidad: D.P.:
Provincia: Teléfono:
País: Profesión:

Deseo suscribirme a TU MICRO por el período de un año a partir del número ....., al precio de ☐ 2.500 ptas. (correo ordinario) o ☐ 2.800 ptas. (correo certificado).

El importe lo haré efectivo mediante:

Firma

☐ Talón bancario a nombre de INGELEK.
☐ Reembolso (por el importe de la suscripción más gastos de envío) al recibir el primer ejemplar.
☐ Giro postal n.º ..... Fecha .....
Localidad .....

(No olvide poner en el sobre las siglas STM.)

Bucle: Secuencia de instrucciones que se ejecutan de forma cíclica e iterativa hasta que un indicador evidencie que se satisface la condición especificada para «salir» del bucle y bifurcar a otro punto del programa.

Buffer: Almacenamiento intermedio concebido para compensar los desequilibrios en la transmisión de datos entre el ordenador y los dispositivos periféricos. Puede utilizarse para el almacenamiento temporal de información. También se conoce como «memoria intermedia» o «memoria tampón».

Bug: Palabra inglesa cuyo significado literal es «gazapo» y que se aplica a los errores que se producen en un programa. Se trata, pues, de un error de programación.

Bus: Mazo de conductores que enlaza entre sí a circuitos electrónicos. Dentro de un ordenador, los buses canalizan las señales eléctricas portadoras de información. El funcionamiento de un microordenador está regido por el microprocesador; éste dirige la actividad del sistema a través de tres buses:

«C»: Lenguaje de programación de alto nivel. Dentro del marco de esta categoría de lenguajes, el «C» es uno de los más próximos a la estructura de diálogo de la máquina. Diseñado en su origen para trabajar con el sistema operativo UNIX, el «C» es un lenguaje de programación de aplicación general que sintetiza una notable economía en las expresiones, con un moderno control de las estructuras y del flujo de los datos y un rico repertorio de operadores.

Cadena: En informática, la cadena de caracteres *(String)* es una palabra o conjunto de caracteres alfanuméricos que recibirán por parte del ordenador un tratamiento distinto al que se aplica a los datos numéricos.

Canal: Dispositivo encargado de realizar las transacciones de información entre la unidad central y los periféricos.

Caracter: Elemento de información —letra (A-Z), número (0-9) o símbolo— que, normalmente, se codifica por medio de un byte u octeto. Mientras el bit es el elemento básico de información para el trabajo del ordenador, el caracter lo es a la hora de representar la información de forma utilizable por el usuario.

—**Bus de datos** *(Data bus).*
Transmite los datos e instrucciones en tratamiento.
—**Bus de direcciones** *(Address bus).*
Conjunto de líneas que transportan las direcciones de memoria en las que el microprocesador va a realizar operaciones de lectura o escritura.
—**Bus de control** *(Control bus).*
Agrupa al conjunto de líneas eléctricas que canalizan las órdenes y señales de gobierno que acceden al microprocesador, o que parten de éste con destino a las restantes unidades del sistema.

Byte: Palabra binaria de 8 bits (un octavo) que es tratada como una sola unidad. Suele ser la unidad básica para la codificación binaria de cada carácter o símbolo de un alfabeto.

(Las referencias cruzadas se indican con un asterisco.)

anuncios breves

tuMicro

Recorte esta ficha y envíela dentro de un sobre a:
Ediciones INGELEK, S. A.
Apto. de Correos 61.294
28080 MADRID

suscripciones

tuMicro

Recorte esta ficha y envíela dentro de un sobre a:
Ediciones INGELEK, S. A.
Apto. de Correos 61.294
28080 MADRID

Carga: Este término tiene varias acepciones en el terreno informático. Puede aplicarse a la lectura de datos externos para su introducción en el ordenador o a la preparación de un periférico para que, a continuación pueda ser objeto de acceso por parte del ordenador o del usuario.

Cargador: Los cargadores o *loaders* en terminología inglesa, son programas auxiliares cuyo cometido es la introducción en la memoria del sistema de otros programas que deben ser ejecutados por el equipo.

Cartucho: La acepción más común dentro del mundo del microordenador es la de módulo que aloja un programa, residente en memoria ROM, y que suele ser directamente enchufable al equipo para su ejecución.
También se denomina cartucho al medio de almacenamiento cuyo soporte es una cinta magnética de tamaño superior a la de tipo casete.

(Las referencias cruzadas se indican con un asterisco.)

tu MicroDIVERSION

el editor

Cualquier aficionado sabe que una de las características más apreciadas del editor de un intérprete BASIC es su capacidad para detectar errores en los programas y, por supuesto, ponerlos en conocimiento del programador a través del correspondiente mensaje.
Suponga, por un instante, que el editor de su equipo se ha ido de vacaciones y que ud. debe realizar su cometido... ¿Cómo anda su capacidad de detección? ¡Póngala a prueba con el siguiente programa e intente localizar los 5 errores que contiene!

```
10 REM El editor
20 FOR I=1 TO 20 STEP -1
30 FOR K=10 TO 30
40 INPUT A$
50 IF A$=100 THEN GOTO 250
60 PRINT "Este es el "resultado" de la operacion:"
70 PRINT A$*I+K
80 NEXT I
90 NEXT K
100 END
```

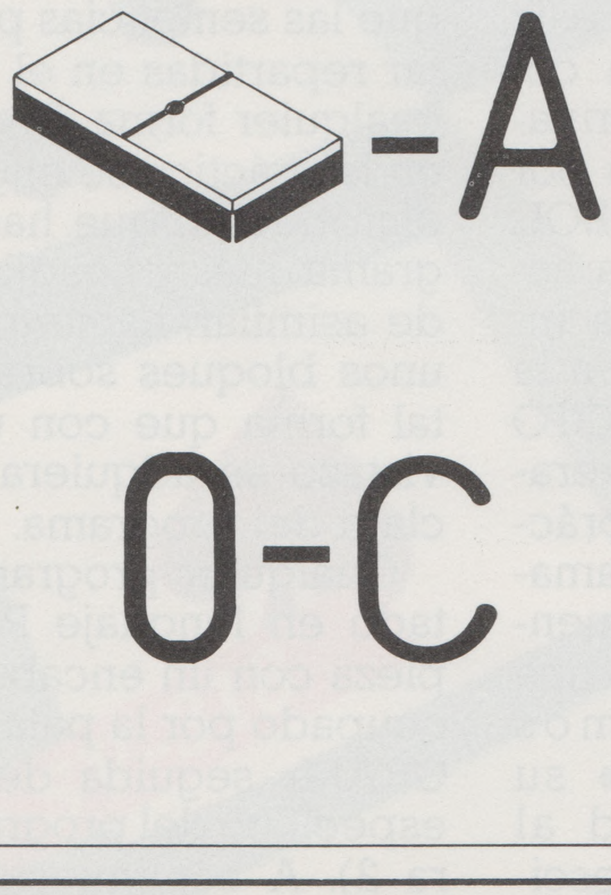
revoltijo de comandos

Ordenando las letras de las palabras obtendrá los nombres de 8 comandos BASIC. Una vez colocados los nombres de los comandos, podrá construir la frase secreta combinando las letras encuadradas.
La frase secreta expresa una moraleja que nos complacería popularizar.

Moraleja: __ _____ __, _______

PITUN □____□
SILT __□□__
NUR __□__
BOGUS □__□___
NIRTP __□□___
LARCE □___□□
MER □__□
DOCAL □__□□__

jeroglífico



Soluciones a los pasatiempos de TU MICRO número 5

¡abreviando, que es gerundio!
1-d, 2-e, 3-b, 4-h, 5-j, 6-a, 7-g, 8-c, 9-i, 10-f.

jeroglífico
Pascal (P/AS/CA/L)

sopa informática
Y aquí están los diez términos microinformáticos caidos en la sopa.

M	M	U	S	U	B	P
O	O	P	U	A	L	I
R	R	M	P	M	R	A
P	A	I	C	A	R	U
M	E	M	O	R	I	A

La arquitectura de un lenguaje estructurado

PASCAL (1)

Iniciamos aquí una serie de artículos en los que trataremos de sacudirnos los «vicios» que, según los entendidos, tienen los lenguajes no estructurados tales como el FORTRAN o el BASIC. Para ello, utilizaremos como guía el lenguaje PASCAL: actualmente, el miembro más asequible de la familia de los «lenguajes estructurados».

Los lenguajes estructurados suponen la materialización de las ideas sobre una nueva forma de programar que empezaron a fluir de la mente de los expertos hace algo más de diez años.

El primer lenguaje de alto nivel que se popularizó en el mundo de la informática fue el FORTRAN. Muy pronto se comprobó que la propia sintaxis de este lenguaje inducía a poner en práctica técnicas de programación propensa a que se cometieran errores; esta forma de programar era común a otros lenguajes derivados del FORTRAN, como por ejemplo, el BASIC o el COBOL.

Según Dijstra (1965), «la calidad de un programa es inversamente proporcional a la cantidad de sentencias GOTO que contenga». Esta aseveración, tan contundente, es práctica habitual en los programadores actuales. Probablemente, más de una vez hemos «palpado» su veracidad al intentar descifrar qué hacía un programa escrito en BASIC y encontrarnos saltando de un extremo a otro del programa, de la mano de inflexibles GOTOS. El resultado no era otro que un progresivo aburrimiento de quien pretendía analizar el programa y, lo que es peor, una acusada sensación de impotencia al ver la imposibilidad de culminar tan arduo análisis.

El PASCAL, al igual que cualquier otro lenguaje estructurado, ayuda en gran media a librarse de la tiranía de los GOTOs, dando lugar a programas claros, elegantes y fáciles de depurar.



Estructura de un programa Pascal

El lenguaje Pascal tiene un formato libre, lo cual significa que las sentencias pueden estar repartidas en el listado de cualquier forma. Desde luego, en la práctica se elige aquella distribución que haga al programa más inteligible y fácil de asimilar, haciendo resaltar unos bloques sobre otros de tal forma que con un simple vistazo se adquiera una idea clara del programa.

Cualquier programa redactado en lenguaje Pascal empieza con un encabezamiento ocupado por la palabra «PROGRAM» seguida del nombre específico del programa (figura 2). A continuación, aparecen las palabras «input» y «output»; su presencia indica que el programa necesita unos datos de entrada que,

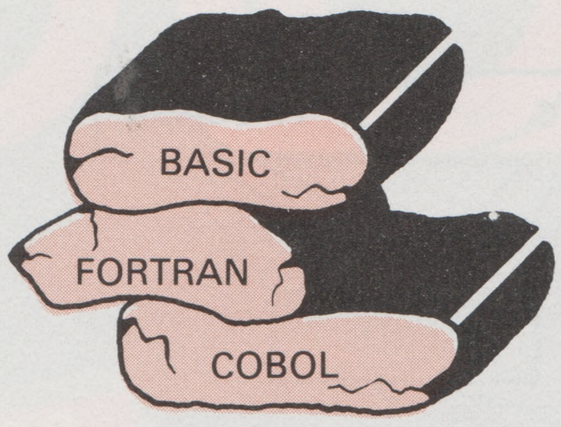
tras ser procesados, darán lugar a unos datos de salida. Si la acción de nuestro programa no exigiera datos de entrada, la cabecera se modificaría de la siguiente forma:

PROGRAM nombre (output);

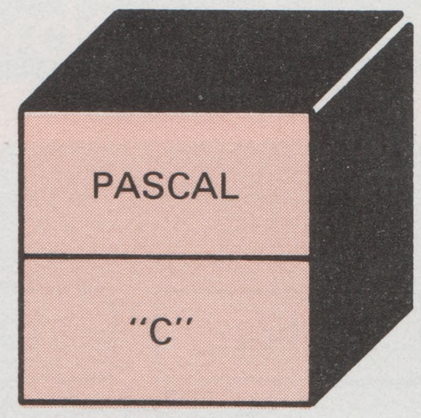
si observamos de nuevo la estructura típica de un programa en Pascal (figura 2), nos encontamos con una zona de «declaraciones»: en ella se indican las variables del programa, también se especifican las constantes, los tipos, los procedimientos y las funciones, términos de los que hablaremos más adelante. El cuadro de la figura 3 muestra cómo especificar las variables; por el momento consideraremos que las hay de cuatro tipos: INTEGER (entera), REAL (real), BOOLEAN (variable lógica que sólo puede tomar los valores TRUE —verdadera— o FALSE —falsa—) y CHAR (variable de tipo carácter). El por qué del signo ";" se verá más adelante.

Salida

La salida de datos se gestiona por medio de dos sentencias: WRITE y WRITELN. Así, por ejemplo, dentro de un programa en Pascal podemos escribir:
WRITE ("Pepe");
(Cuidado con el signo de puntuación ;)



El ordenador responderá escribiendo la palabra Pepe, ajustada al principio de la línea; si ahora se le ordena escribir cualquier otra palabra, número o frase, lo hará exactamente a continuación de la última «e» de Pepe. Sin embargo, si hubiéramos sustituido la anterior orden de escritura por:
WRITELN ("Pepe");
el texto que deba escribir a continuación aparecerá ajustado al principio de la línea siguiente.



1

Una primera clasificación de los lenguajes de programación de alto nivel, distingue entre lenguajes no estructurados y estructurados. Hoy en día, el Pascal es el más popular de los lenguajes catalogados en este segundo grupo.

ESTRUCTURA DE UN PROGRAMA EN LENGUAJE PASCAL

```
PROGRAM nombre (input,output);
    declaraciones
BEGIN
    sentencias
END.
```

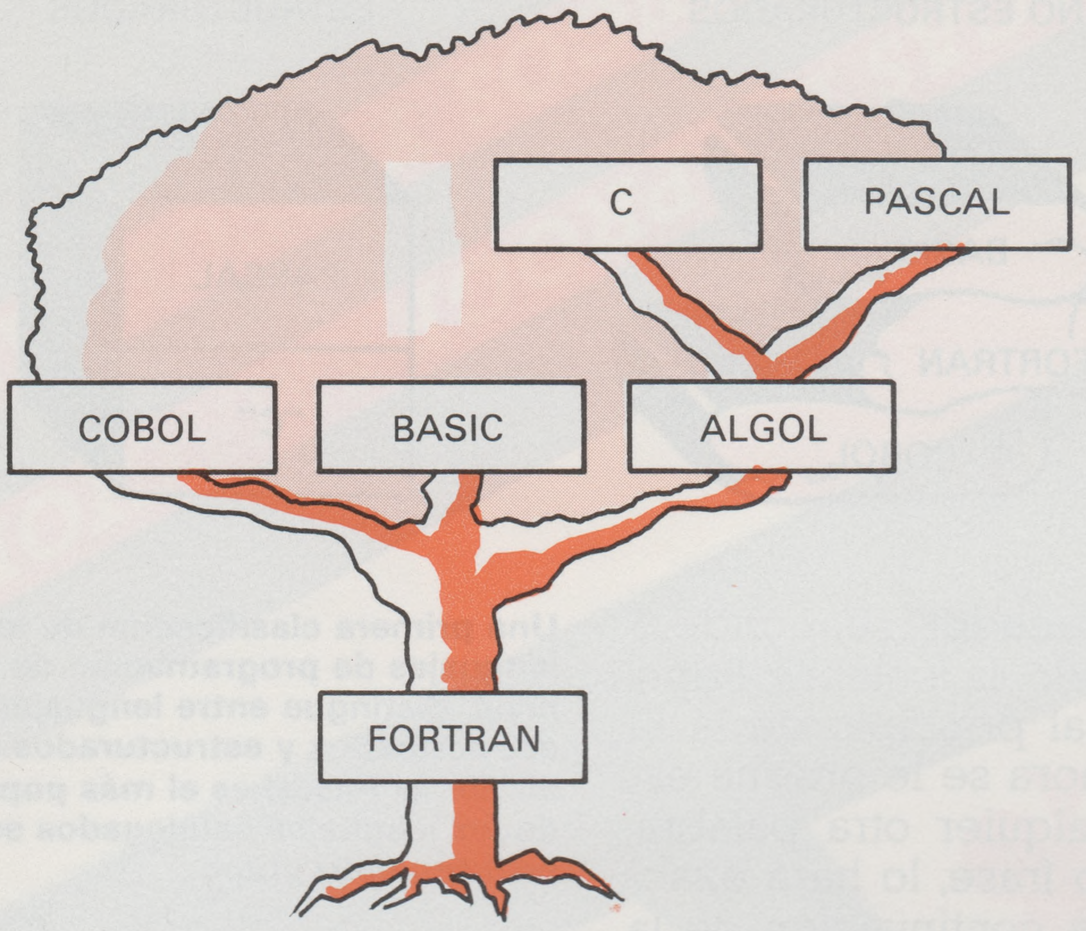
2

ESPECIFICACION DE VARIABLES

```
PROGRAM nombre (input,output);
  VAR
    contador:integer;
    radicando:real;
    ciertofalso:boolean;
    primeraletra,segundaletra:char;
BEGIN
    .....
END.
```

3

Como se observa en el ejemplo, es posible especificar más de una variable dentro de una misma línea; la última línea de «declaración» define dos variables de tipo caracter.



LA GENESIS DEL PASCAL

Al ejecutar las dos sentencias del ejemplo 1, el ordenador escribirá TUMICRO. Cualquier otra sentencia WRITE o WRITELN que se ejecute a continuación se traducirá en una escritura en la línea inmediatamente inferior.
La ejecución del ejemplo 2 da lugar a la siguiente presentación:
Pascal es
un lenguaje estructurado
La próxima operación de escritura se realizará justo a continuación de la última letra de la palabra «estructurado»; así lo determina la sentencia WRITE ejecutada en última instancia.

En resumen, la acción de WRITELN puede resumirse en: «escribe lo que tengas entre paréntesis y sitúate al principio de la siguiente línea», mientras que en el caso de utilizar la sentencia WRITE, sería: «escribe lo que tengas entre paréntesis y quédate quieto en ese punto de la línea». Los dos ejemplos de la figura 5 ayudarán a clarificar el comportamiento del ordenador en ambos casos.

WRITE y WRITELN también permiten escribir el valor asignado a las variables. Supongamos que existe una variable llamada «contador», variable que hemos declarado en la sección VAR como entera por medio de la expresión siguiente:

VAR contador: integer;

Para extraer su valor podemos recurrir a:

WRITE (contador);

o, como alternativa, a:

WRITELN (contador);

Otra forma de utilizar estas sentencias puede ser la siguiente.

Supongamos que tenemos tres variables v1, v2 y v3; podemos agruparlas dentro de una misma sentencia:

WRITE (v1, v2, v3);

o WRITELN (v1, v2, v3);

El resultado de la ejecución de éstas sentencias será la escritura o salida de los valores que tengan asignados las tres variables, uno detrás de otro en la misma línea. Las dos sentencias propuestas se diferencian —tal como ya sabemos— en que la próxima acción de escritura empezará en la misma línea o en la siguiente.

Los comentarios

Los comentarios son una ayuda decisiva a la hora de hacer legible y facilitar la comprensión de un programa. Pueden ir colocados en cualquier zona del mismo, pero siempre deben ir encerrados entre corchetes [{ }]. La figura 6 incluye un ejemplo, simple aunque claramente ilustrativo, de la utilidad de los comentarios; la declaración de variables se completa con dos frases o comentarios reveladores de la aplicación de las variables.

5

```
(Ejemplo 1)

WRITE ('TU');
WRITELN ('MICRO');

(Ejemplo 2)

WRITELN ('Pascal es');
WRITE ('un lenguaje');
WRITE ('estructurado');
```

Procedimientos del Pascal

El papel de los procedimientos del Pascal es análogo al de las subrutinas del lenguaje BASIC; no obstante, su filosofía es completamente distinta, ya que con los procedimientos se busca «modularizar» al máximo un programa, intentando, como siempre, que sea lo más claro y fácil de entender posible.

Un buen programa en lenguaje Pascal cuenta con una serie de procedimientos definidos en el bloque de «declaraciones», mientras que el cuerpo del programa (la zona comprendida entre el BEGIN y el END de la figura 2) suele incluir básicamente, llamadas a los procedimientos.

Un procedimiento permite agrupar a un conjunto de sentencias bajo un nombre común, las cuales serán ejecutadas cada vez que se haga una llamada al mismo. La estructura de un procedimiento es, en esencia, la misma que la del propio programa Pascal, con su zona de declaraciones y con el cuerpo del mismo encerrado en un bloque BEGIN-END.

Veamos un sencillo ejemplo. El objetivo es programar la escritura de los diez primeros números naturales en dos líneas, cada una de ellas ocupada por cinco números, y repetir tres veces la operación de escritura. Cada una de las tres acciones de salida se realiza al ejecutar el par de sentencias:

WRITELN ("01234");
WRITELN ("56789");

En definitiva, se trata de una operación que podemos agrupar en un procedimiento que vamos a denominar «DibujarDosLíneas»; su expresión en Pascal será:

```
PROGRAM DibujarBloques (output);
    PROCEDURE DibujarDosLineas;
      BEGIN
         WRITELN ('01234');
         WRITELN ('56789');
      END;
```

Un procedimiento que no utiliza variable alguna, de ahí que no exista ninguna sentencia de declaración al efecto. Una vez definido el procedimiento, habrá que encuadrarlo dentro de un programa cuya ejecución realice la tarea solicitada (figura 7).

6

```
PROGRAM nombre (input,output);
    VAR
       cont1, cont2:integer;  {éstos serán contadores}
       ndear:integer;         {variable con el número de}
                              {ecuaciones a resolver}
       .....
```

Los comentarios al programa pueden colocarse en cualquier zona del mismo, si bien, siempre deben estar encerrados entre llaves (...).

Al ejecutar el programa en cuestión, el ordenador buscará el BEGIN que corresponde al cuerpo del programa; en éste encontrará la sentencia DibujarDosLíneas, la cual provocará una llamada al procedimiento del mismo nombre que será ejecutado, traduciéndose en la ejecución de las sentencias WRITELN.

7

```
PROGRAM DibujarBloques (output);
    PROCEDURE DibujarDosLineas;
      BEGIN              {Aquí empieza el procedimiento}
         WRITELN ('01234');
         WRITELN ('56789');
      END;               {Aquí termina el procedimiento}
  BEGIN                  {Principio del programa principal}
    DibujarDosLineas;
    WRITELN;
    DibujarDosLineas;
  END.                   {Fin del programa principal}
```

(Una vez ejecutado el programa anterior,
la salida será la que se indica a continuación)

```
01234
56789

01234
56789

01234
56789
```

Los «procedimientos» constituyen una potente herramienta de programación del lenguaje Pascal, su papel es análogo al de las subrutinas del BASIC. El programa adjunto utiliza un «procedimiento» como base para ejecutar una escritura repetitiva.

Tu Micro N.º 6, Pág. 65

dose en la escritura de las dos líneas de números. Acto seguido, se devuelve el control al cuerpo del programa. Esta vez, será la sentencia "WRITELN;" la localizada; al ejecutarla, el ordenador no escribirá nada y se limitará a saltar a la línea siguiente (su efecto es dejar una línea en blanco). El programa seguirá ejecutándose de forma análoga hasta llegar al último END. En el listado del programa ejemplo (figura 6), hemos utilizado las características del «formato libre» para resaltar el bloque correspondiente al procedimiento, escribiéndolo algo más a la derecha que el resto del programa; concretamente, hemos recurrido a la posibilidad de «indentación». De igual manera se «indentan» las sentencias que están dentro de un mismo bloque BEGIN-END, tal es el caso de las sentencias del programa principal.

¿Qué es la programación estructurada?

El PASCAL es un lenguaje de programación desarrollado fundamentalmente por Niklaus Wirth en la década de los setenta, como fruto de las ideas sobre programación estructurada que empezaron a tomar cuerpo a principios de esa misma década.

En la figura 8 aparece lo que se denomina «ciclo de vida» de un programa; un gráfico representativo de la «vida» de un programa; desde que «nace» (fase de diseño) hasta que, por diversas razones, deja de ser útil.

La fase de diseño corresponde a la etapa en la que se define el problema y se estudian los posibles resultados del programa y su forma de presentarlos al exterior. En la fase de codificación, el problema planteado se «traduce» a un lenguaje de ordenador (en nuestro caso el Pascal). La tercera fase, de depuración consiste en eliminar los pequeños (o grandes) errores que hayan podido surgir en la fase de codificación. A partir de entonces, el programa puede entrar en funcionamiento. De cualquier forma, durante la utilización del mismo pueden aparecer situaciones que no fueron previstas en ninguna de las fases anteriores y que obliguen a retocar el programa: ésta sería la fase de mantenimiento. El mantenimiento es, pues, una especie de depuración permanente del programa a lo largo de su existencia.

La programación estructurada aporta una ayuda decisiva en las tres últimas fases del ciclo de vida de un programa (de hecho, la fase de diseño es la menos conectada con el resultado final: el programa).

Los componentes de la programación estructurada son:

— *Filosofía de diseño de arriba a bajo (top-down).* En la propia fase de diseño hay que partir de unas ideas generales del resultado final. Hay que fijar objetivos generales para luego descender a lo particular y en este nivel es donde debe empezar la codificación de pequeñas tareas.

— *Módulos independientes.* Cada una de las pequeñas tareas mencionadas se denomina «módulo»; éstas permiten construir un programa complejo a partir de pequeños fragmentos fácilmente codificables y que, por lo tanto, estarán en gran medida libres de errores. Además, de esta forma el programador estará más concentrado en una sola tarea, sin la necesidad de mantener en la cabeza la estructura global del programa. Si durante la depuración o en el uso posterior del programa se produce un fallo, éste derivará, muy probablemente, de un solo módulo y, en consecuencia, bastará con sustituir este módulo sin cambiar el resto del programa.

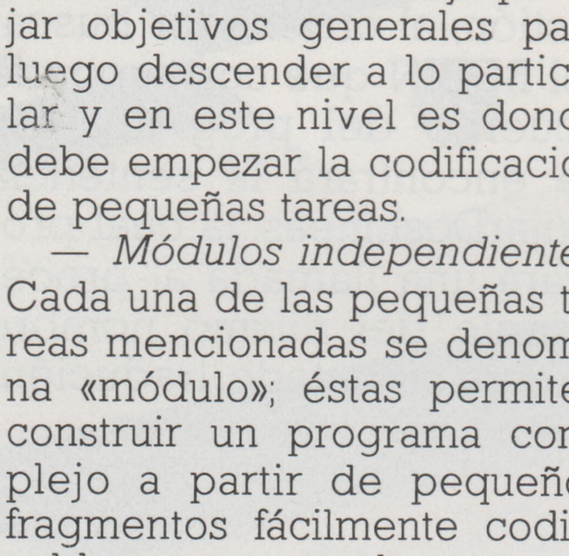
—*Codificación estructurada.* Cuando se utiliza un lenguaje estructurado, la fase de codificación se basa en los principios de

Secuencia: las sentencias se ejecutan una tras otra. Bien es cierto que ésta no es una característica esencial de un lenguaje estructurado, sino que es la esencia misma de un programa.

Iteración: repetición de un cierto fragmento de un programa. El Pascal permite diversos tipos de iteraciones a través de las sentencias FOR DO, WHILE DO y REPEAT UNTIL.

Selección: posibilidad de que al llegar a un cierto punto de un programa se puedan realizar distintas acciones. En el lenguaje Pascal esto se lleva a cabo por medio de las sentencias IF THEN ELSE y CASE OF.

Resulta obvio que bajo éstos principios se elimina la necesidad de una sentencia GOTO que imponga una ruptura de secuencia. Los programas codificados de esta forma son más claros y fáciles de entender y, en definitiva, las dos últimas fases del ciclo de vida de un programa son más sencillas de realizar.



CICLO DE VIDA DE UN PROGRAMA

Enciclopedia Práctica de la

INFORMATICA

***52** fascículos coleccionables y encuadernables en **4** tomos*

VIVIMOS EN EL MUNDO DE LA INFORMATICA



¡DOMINALO!

En cada fascículo 6 secciones

- INFORMATICA BASICA: Los conocimientos fundamentales.
- HARDWARE: Los 52 ordenadores más vendidos.
- SOFTWARE: Los programas: como son y como se hacen.
- PERIFERICOS: Como comunicarse con el ordenador.
- APLICACIONES: Presentación y comentario de programas.
- EL MUNDO DE LA INFORMATICA: Lo insólito y lo práctico.

MAS DE **1.048** PAGINAS A TODO COLOR MAS DE **3.000** ILUSTRACIONES

UNA OBRA EXCEPCIONAL REALIZADA POR LOS MEJORES ESPECIALISTAS

Deseo recibir la obra ENCICLOPEDIA PRACTICA DE LA INFORMATICA en la forma que indico a continuación:

☐ *4 TOMOS ENCUADERNADOS (7.900 ptas.)**
☐ *52 fascículos mas 4 tapas para la encuadernación (7.400 ptas.)**
*** oferta sólo válida hasta el 30 de octubre de 1984.**

APELLIDOS _____
NOMBRE _____
DOMICILIO _____
LOCALIDAD _____ CODIGO POSTAL _____
PROVINCIA _____ TELEFONO _____

El importe de mi pedido lo haré efectivo de la manera siguiente:
☐ Talón a nombre de INGELEK, S. A., que adjunto en el sobre. n.º ..... Banco .....
☐ Giro postal n.º ..... Fecha .....
☐ Reembolso del importe mas gastos de envío.

Recorte o copie este cupón y envíelo a EDICIONES INGELEK, S. A. Avda. Alfonso XIII, 141. 28016 MADRID.

definidor de SPRITES

Una de las características comunes a la mayoría de los microordenadores llegados al mercado últimamente, es el manejo de sprites. Podríamos considerar un sprite, como una unidad gráfica definible por el usuario, tanto en su forma (perfil) y color, como en su posición en la pantalla (posición absoluta, plano de representación).

Cada microordenador posee un método, particular y más o menos cómodo, para el manejo de los sprites y, por supuesto, presenta unas características de colores utilizables, máxima dimensión del gráfico, número máximo de sprites, etc... totalmente arbitrarias.

El equipo que hoy nos ocupa, el SPECTRAVIDEO SV-318, tiene la virtud de la sencillez en el tratamiento de los sprites; al menos en comparación con otras máquinas de su categoría como, por ejemplo, el COMMODORE 64. Y,



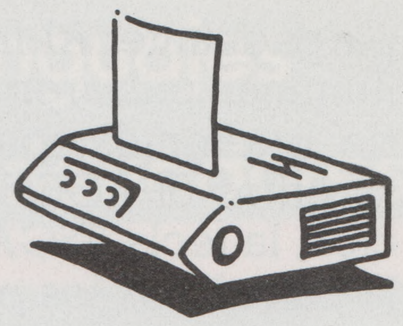
como no, también tiene contrapartidas poco loables: las dimensiones mínimas del gráfico, sin ir más lejos, que se elevan a 8×8 puntos.

En todo caso, una afirmación meridiana y aplicable a los procesos de definición de sprites en cualquier equipo, es que la tarea de dar forma al nuevo caracter gráfico puede llegar a convertirse en un suplicio.

La idea de almacenar en una sentencia DATA la configuración del sprite (pattern), es bastante buena; sin embargo, tal como se propone en el manual, el almacenamiento de cadenas en binario ocupa excesivo espacio y, sobre todo, tiende a producir equivocaciones en la maraña de unos y ceros.

El programa que presentamos en esta ocasión, permitirá definir cómodamente los sprites, entregando, como fruto de su ejecución, la serie de ocho números (en base decimal) que configuran el gráfico y que deben incluirse en la instrucción DATA correspondiente. El programa puede memorizar hasta 32 sprites (número máximo disponible en el aparato), lo que permite remodificar las veces que sea necesario cualquiera de los gráficos, hasta que se dé por terminado el programa.

Durante el proceso de definición, aparece en la pantalla un tablero (grid) de 64 casillas (8×8), cada una de las cuales corresponde a un punto del sprite (pixel). Para deslizarnos por el grid utilizaremos el joystick; en todo momento, nuestra posición está señalada por un cursor



CUADRO DE VARIABLES

C	Columna del grid sobre la que nos encontramos.
F	Fila del grid sobre la que nos encontramos.
I, J	FOR de diversa utilidad.
P	SWICTH para generación de cursor parpadeante.
S	Sprite en tratamiento.
A$	Caracter sobre el cual se encuentra el cursor.
B$	Cadena para construcción del valor binario del pattern.
C$	Cadena para construcción del valor decimal del pattern.
D$	Cadena para reconstrucción de A() a partir de B().
X$	Carácter de almacenamiento de INKEY$.
A(7,7)	Pattern del grid del sprite actual en binario.
B(31,7)	Pattern de los sprites en decimal.

LISTADO



```
10 REM Lopez Martinez # Definidor de SPRITES # Spectravideo SV-3
18
20 DIMA(7,7):DIMB(31,7):S=0
30 CLS
40 PRINTTAB(7)"- DEFINIDOR DE SPRITES -"
50 LOCATE0,3
60 FORI=0TO7
70 PRINTTAB(15)"PPPPPPPP"
80 NEXTI
90 LOCATE0,15:PRINT"Deslicese por el tablero con el stick."
100 LOCATE0,17:PRINT"ENTER para marcar la celdilla."
110 LOCATE0,19:PRINT"SPACE para liberar la celdilla."
120 LOCATE0,21:PRINT"ESCAPE para terminar la definicion."
130 LOCATE0,7:PRINT"SPRITE:"S
140 FORI=0TO7
150 FORJ=0TO7
160 LOCATE15+J,3+I
170 IFA(I,J)THENPRINT"F":GOTO190
180 PRINT"F"
190 NEXTJ
200 NEXTI
210 F=0:C=0:P=0
220 IFA(F,C)THENA$="F":GOTO240
230 A$="P"
240 LOCATE15+C,3+F
250 IFPTHENPRINT"Y":P=0:GOTO270
260 PRINTA$:P=1
270 X$=INKEY$
280 FORI=0TO49:NEXTI
290 IFX$=CHR$(27)THEN370
```

parpadeante. Al llegar al pixel que deseamos iluminar pulsaremos ENTER; si, por el contrario, queremos apagar un punto iluminado pulsaremos SPACE. Una vez terminada la definición, la tecla ESCAPE nos permitirá abandonar la pantalla. A los pocos segundos, aparecerá la configuración decimal que debe figurar en la DATA correspondiente y acto seguido, el programa nos preguntará cuál es el número del sprite que se quiere definir a continuación; por supuesto, su número debe estar comprendido entre 0 y 31.

Si lo que deseamos es modificar un gráfico ya definido anteriormente, bastará con dar el número de sprite que se le otorgó y, antes de pasar a la definición, el gráfico será representado en el grid. Si, por equivocación, se accede a un sprite que no debe ser modificado, puede abandonarse la fase de modificación sin que el sprite se vea alterado, sin más que accionar la tecla ESCAPE.

Para dar por terminada la definición, hay que responder introduciendo el número 32 ante la pregunta «¿cuál es el nuevo sprite a definir?» Hay que señalar que el paso con el cursor por encima de las celdillas no produce ninguna modificación.

Poco hay que añadir acerca de la introducción del programa; simplemente, recordar que los caracteres que aparecen con subrayado simple corresponden al gráfico izquierdo de la tecla, y los de subrayado doble al gráfico derecho de la tecla cuyo símbolo principal figura en el listado.

tu Micro

ESTRUCTURA DEL PROGRAMA

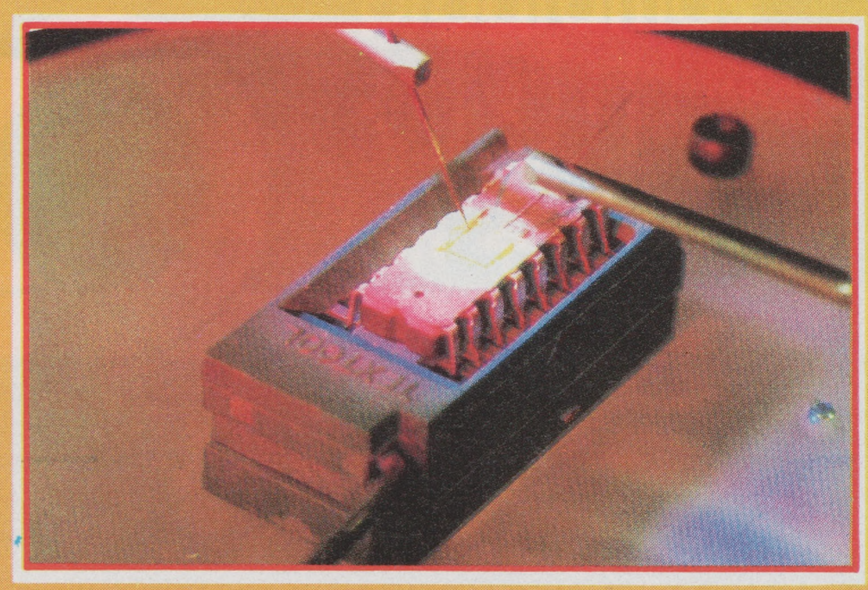
10-20	Inicializaciones.
30-130	Construcción de la pantalla.
140-200	Generación del nuevo pattern en el grid.
210-360	Definición del sprite.
370-440	Localización del valor numérico del pattern.
450-540	Representación del pattern en decimal y selección de nuevo sprite.
550-630	Construcción de A() a partir de B().

```
300 IFX$=CHR$(13)THENA(F,C)=1:LOCATE15+C,3+F:PRINT"■":GOTO220
310 IFX$=CHR$(32)THENA(F,C)=0:LOCATE15+C,3+F:PRINT"■":GOTO220
320 IFX$=CHR$(28)ANDC<7THENLOCATE15+C,3+F:PRINTA$:C=C+1:GOTO220
330 IFX$=CHR$(29)ANDCTHENLOCATE15+C,3+F:PRINTA$:C=C-1:GOTO220
340 IFX$=CHR$(30)ANDFTHENLOCATE15+C,3+F:PRINTA$:F=F-1:GOTO220
350 IFX$=CHR$(31)ANDF<7THENLOCATE15+C,3+F:PRINTA$:F=F+1:GOTO220
360 GOTO240
370 FORI=0TO7
380 B$=""
390 FORJ=0TO7
400 B$=B$+RIGHT$(STR$(A(I,J)),1)
410 NEXTJ
420 B$="&b"+B$
430 B(S,I)=VAL(B$)
440 NEXTI
450 CLS
460 LOCATE4,6:PRINT"EL SPRITE"S"TIENE LA SIGUIENTE"
470 LOCATE 14,8:PRINT"DEFINICION"
480 C$=""
490 FORI=0TO7
500 C$=C$+RIGHT$(STR$(B(S,I)),LEN(STR$(B(S,I)))-1)+","
510 NEXTI
520 C$=LEFT$(C$,LEN(C$)-1)
530 LOCATE19-LEN(C$)/2,10:PRINTC$
540 LOCATE1,12:INPUT"PROXIMO SPRITE A DEFINIR (0-31)";S:IFS=32TH
ENCLS:END
550 S=INT(S):IFS<0ORS>31THEN540
560 FORI=0TO7
570 D$="0000000"+BIN$(B(S,I))
580 D$=RIGHT$(D$,8)
590 FORJ=0TO7
600 A(I,J)=VAL(MID$(D$,J+1,1))
610 NEXTJ
620 NEXTI
630 GOTO30
```

END

Electro OCIO

electrónica, micro-informática
y tecnología para todos



N.º 12 - octubre 1984 200 Ptas.

Satélites de comunicaciones

Inteligencia del microprocesador
¿Un cerebro de cables y circuitos?

LA SUSCRIPCION ES LA FORMA MAS BARATA COMODA Y SENCILLA DE ADQUIRIR SU REVISTA FAVORITA

POR ELLO NOS PERMITIMOS SUGERIRLE QUE SE SUSCRIBA AHORA; POR UN AÑO Y OBTENGA LAS SIGUIENTES VENTAJAS:

* Ahorro de 400 ptas.
* Importantes descuentos en el momento de adquirir otros productos de nuestra editorial.
* Recepción de la revista comodamente en su domicilio.

Recorte o copie este cupón y envíelo a EDICIONES INGELEK, S. A. Avda. Alfonso XIII, 141. 28016 MADRID.

APELLIDOS
NOMBRE
DOMICILIO
LOCALIDAD CODIGO POSTAL
PROVINCIA TELEFONO

Deseo suscribirme a la revista Electro Ocio por el periodo de un año a partir del mes de inclusive, al precio de ☐ 1.800 ptas. (correo ordinario), ☐ 2.100 ptas. (correo certificado). El importe lo haré efectivo mediante:

☐ Talón a nombre de INGELEK, S. A., que adjunto en el sobre. n.º ..... Banco .....
☐ Giro postal n.º ..... Fecha .....
☐ Reembolso del importe mas gastos de envío.

Firma,

ANUNCIOS BREVES

VENTAS

Amplificadores de sonido para SPECTRUM, varios modelos: 1.600 ptas. Reguladores de luz al tacto: 3.000 ptas.
Ernesto López. Rebeco, 27; 2.º-1.º. Teléf.: 93/3546221. Barcelona.

ZX-81 ampliado a 16 K, con sus accesorios y fotocopia de «34 amazing games» y 30 programas. Todo por 15.000 ptas.
Santiago A. Cárdenas. San Millán, 7; 8.º-27. 29013 Málaga.

Programas para COMMODORE 64, vendo o cambio. Precio a convenir.
Jesús Godas Limia. Blanco Rajoy, 20. Teléf.: 988/410348. Verín (Orense).

Manual de reparaciones del ZX-81 con esquemas y averías típicas (3.500 ptas.). También programa para diseño de circuitos electrónicos (3.000 ptas.).
Nelson. Apartado 2109. Teléf.: 965/858553. Benidorm (Málaga).

Impresora ORIC MCP-40, con accesorios, caja, etc.; sólo por 34.000 ptas.
José Luis García Ordóñez. Mosén Manyá, 5; 4.º-3.º. Teléf.: 977/441799. Tortosa (Tarragona).

100 programas para SPECTRUM por 3.000 ptas.
Antonio Rodríguez. Queipo de Llano, 2; 6.º D. Orense.

Ampliación de memoria 16 K (13.000 ptas.). Cartucho SUPEREXPANDER (5.800 ptas.).
Manuel Fuentes Sorrivas. Apartado 1473. Teléf.: 986/471768. Vigo (Pontevedra).

SPECTRUM 48 K y periféricos. También vendo programas. Precio a convenir.
José M. Vicente Esteban. Fernando Rojas, 6; 2.º C. Teléf.: 923/228353. 37005 Salamanca.

Programas de juego y otras aplicaciones en casete para el ORIC. Precios razonables.
Ramón Rubies Alvarez. Teléf.: 93/3407581. Barcelona.

Programador de memorias EPROM para VIC-20. Hago cualquier copia de memorias.
Félix Portabella Padró. Forn de Sta. Lucía, 1; 2.º. Teléf.: 93/8722297. Manresa (Barcelona).

NEW BRAIN AD, con manual, guía del principiante, casete y programas; como nuevo. Todo por 50.000 ptas.
Angel Osinaga Rubio. José Jimeno, 2; 2.º. Teléf.: 948/113507. 31015 Pamplona.

SPECTRUM 16 K, nuevo, con garantía hasta enero del 85, adaptador y cables (30.000 ptas.). Regalo 5 programas.
Antonio Crespo Ramos. María Moliner, 67. Teléf.: 976/272423. 50007 Zaragoza.

SPECTRUM 48 K por 35.000 ptas. con manual, alimentador, etc.
Joaquín Bayón López. Capitán Almeida, 28; 1.º Izq. 33009 Oviedo.

Programa educativo COMMODORE 64: «Palabras y deletrear», para niños de 2 a 7 años. En casete (1.500 ptas.), disco (2.000). Enviar cheque.
MS. V. Innes (Toddlersoft). 1 CATLANE, EWELME, OXON OX9 6HX, Gran Bretaña.

COMPRAS

Cinta original o copia de demostración de SPECTRUM-48 K que Investrónica da con la ampliación RAM.
Fernando García Hidalgo. Fuentecillas, 10, 2.º. Teléf.: 917/209437. 09001 Burgos.

BBC micro, modelo B. De segunda mano.
Fernando Ramos Ortega. Sol Oriente, 11-15, 4.º-G. 37002 Salamanca.

COMUNICACIONES E INTERCAMBIOS

Deseo intercambiar programas para SPECTRUM. Interesados enviar lista.
Juan Castilla Barragán. Sta. Eulalia, 30; 2.º-1. Berga (Barcelona).

Cambio y vendo programas en cinta para SPECTRUM.
Alejo Alcover Mascaró. Mitja de Mar, 10, n.º 24. Porto Cristo (Baleares).

Cambio programas de todo tipo para SPECTRUM 16 ó 48 K. Llamad al teléfono de Zaragoza (976) 317156 de noche.
Pedro Antonio Galindo. Peña Oroel, 13. Zaragoza.

Deseo intercambiar información y programas con usuarios del ORIC-Atmos. Vendo programas para los equipos ZX.
V. Sastre. Génova, 9. Muro (Baleares).

Cambio o vendo programas para SPECTRUM 16-48 K. Copias perfectas. Vendo listados para todo tipo de ordenadores.
Salvador Piñol Jané. Cataluña, 76-78; 3.º-1. Teléf.: 93/3797497. Prat de Llobregat (Barcelona).

Intercambio todo tipo de programas para los usuarios del NEW BRAIN.
José M. Pedrals Serra. Montserrat, 16. Ametlla de Merola (Barcelona).

Doy programas para SPECTRUM a cambio de programas para ORIC-1/ORIC-Atmos.
Vicenc Sastre Calvó. Génova, 9. Muro (Baleares).

Cambio programas del SPECTRUM; gran variedad. También estaría interesado en esquemas de montajes.
Alexis Herrero Segura. Jacinto Benavente, 23; 7.º. 12006 Castellón.

Desearía contactar con usuarios de SPECTRAVIDEO SV-318 ó 328 para intercambio de ideas, programas, etc.
Manuel Angel Rodríguez. D. Urraca, 17. 24009 León.

Cambio programas para SPECTRUM 16/48 K. En la misma cinta que me envíes tu programa te enviaré otro distinto.
Mauricio Heredero. Apartado de Correos 133. Zamora.

Interesados en la creación de un club de informática en Granada capital. Dirigirse por carta a la dirección siguiente:
Miguel Puerta Carrasco. San Jerónimo, 23; 2.º. 18001 Granada.

SPECTRAVIDEO, desearía contactar con usuarios para intercambio de programas.
Iñaki Castillo. Cataluña, 27; 3.º B. Teléf.: 94/4402999. Basauri (Vizcaya).

Cambio o vendo a 200 ptas. programas para el SPECTRUM. Interesado en los de gestión y contabilidad.
Jaime Serrano Rodríguez. Libertad, 15, 5.º B. Teléf. 91/6141664. Móstoles (Madrid).

Cambiaría plato con cápsula Shure y osciloscopio por ZX-81 o SPECTRUM con accesorios. De 3 a 5 h. Teléf.: 263505.
Miguel Rodríguez. Las Moradas, 17, 1.º. 47010 Valladolid.

Club de microordenadores. Equipos propios y más de 300 programas para intercambio de todo tipo.
Círculo Ourensan-Vigues. Venezuela, 48; Entlo. Vigo (Pontevedra).

MICROGESA

Silva, 5, 4.º - Teléfono 2422471 - Madrid 28013

TODO LO QUE NECESITE EN MICROINFORMATICA

HARDWARE

Spectrum 48 K	41.900 pts.	1.239 al mes
Micro. + Inter.....	35.000 pts.	1.084 al mes
Commodore 64.....	69.000 pts.	2.168 al mes
Spectravideo.....	76.000 pts.	2.322 al mes
MPF-II.....	62.500 pts.	1.858 al mes
Katson.....	92.500 pts.	2.787 al mes

ACCESORIOS: Moden, magnetófono especial, joystik, lápiz óptico, impresoras y monitores.
Programas standard y a medida.

SOFTWARE

Programas adaptados a Microdrive ZX.
—Contabilidad (Pl. Nacional C.).
—Administración de fincas.
—Programa agente de bolsa.
—Proceso de textos.
—Calc (hoja electrónica).
—Gestión archivos.
—Control de stocks.
—Superquinielas (imprime boletos)
Programas también disponibles en cassette.

DIRECTORIO

DIOTRONIC S.A.

LOS MAS AVANZADOS

Componentes Electrónicos, Microprocesadores, Librería Técnica.

Muntaner, 49
Pasaje Valeri Serra 24 y 26
Teléf.: 3232266-3232208
08011 Barcelona

C. Borrell, 108
Teléf.: 2544530
08015 Barcelona

MICRO WORLD

ORDENADORES PERSONALES Y MICROORDENADORES DE GESTION

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

SOFTWARE STANDARD Y A MEDIDA

CURSILLOS Y FORMACION

En Madrid:
MODESTO LAFUENTE, 63. Teléf. 253 94 54

LUGO

COMPONENTES
AUTOSERVICIO
BARQUILLO, 40
4198742-4198751

ELECTRONICA SANDOVAL S.A.

DISTRIBUIDORES DE:
ORIC-1
CASIO FP200
ROCKWELL-AIM-65
VIDEO GENIE-EG-2000
CASIO FX-9000P
SINCLAIR ZX81
OSBORNE 1
DRAGON-32
NEW BRAIN
EPSON HX-20

ELECTRONICA SANDOVAL, S. A.
C/ SANDOVAL, 3, 4, 6 - MADRID-10
TELEX 47784 SAVL-E

digital s.a.

— Los ordenadores personales son demasiado importantes como para permitirse el lujo de ignorarlos.

— Ordenadores personales software, hardware, libros periféricos.

LASER 200 COLOR COMPUTER
COMMODORE 64
ZX SPECTRUM
ORIC
VIC 20
NewBrain

Distribuimos el software publicado por tu MICRO

C/ PILAR DE ZARAGOZA, 45 (semiesquina a Cartagena). 28028 MADRID
TELEFS.: 2464920-2465663

MICROGESA
ESPECIALISTAS MICROINFORMATICA

- Sinclair
- Commodore
- Oric
- Spectravideo
- Katson
- New Brain
- MPF-II

Periféricos y accesorios
Programas
Cursos de Basic

Silva, 5-4.º Teléf.: 2422471 - Madrid 28013

ALICANTE
A. W.

Oric-1
New Brain
M.P.F. 1, 2, 3
XEROX
Vic
Spectrum
Spectravideo
Sharp
T.R.S.-80
Impresoras

Cursos de programación todos los meses
C/ Calderón de la Barca, 2
Teléfono 21 91 28

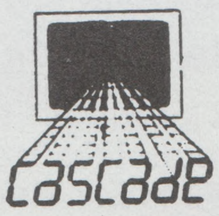
CONTERSA



·Xerox ·Oric Atmos
·Alphatronic ·Toshiba
·New Brain ·Atari

Impresoras Microdrives
Programas

General Ricardos, 117-3°D
☎ 471 22 14/23 62



LA CINTA DEL AÑO

VIC-20 / ORIC
SPECTRUM / BBC / APPLE
DRAGON
ATARI / COMMODORE-64

Cristóbal Bordiú, 35 - 5º Dpcho. 522
Telf.: 254 24 30 - MADRID-3

RELACION DE ANUNCIANTES

APD	75
A. W. Alicante	73
Cascade	73, 74
Contersa	73
Digital	73
Diotronic	73
Dynadata	02
Investrónica	76
Lugo (Electrónica)	73
Micro-World	73
Microgesa	72, 73
Otesa	50
Sandoval (Electrónica)	73
Ventamatic	35

CASCADE CASCADE CASCADE CASCADE

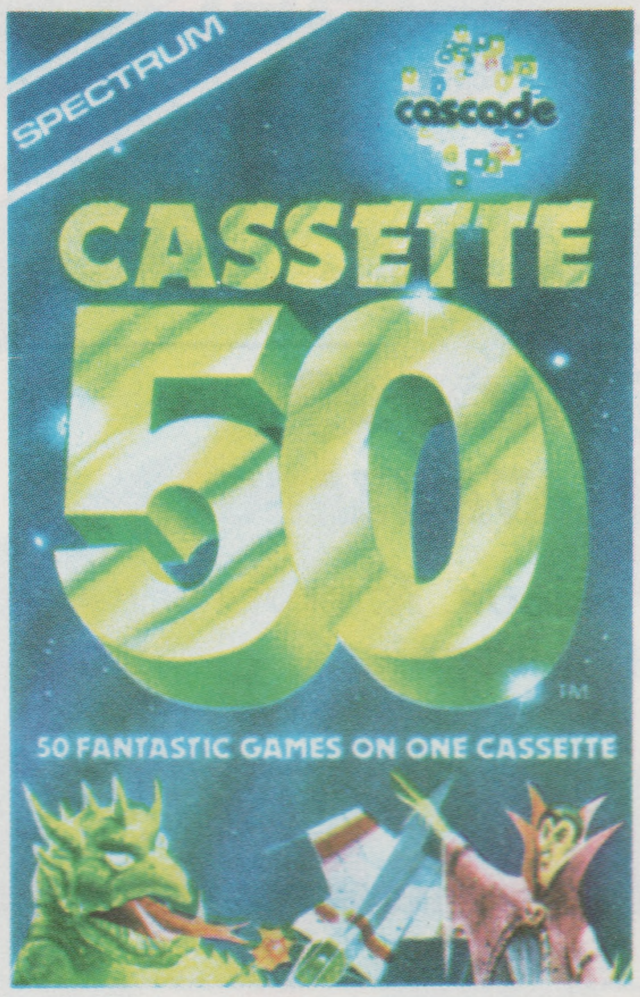
CASCADE CASCADE

LA CINTA DEL AÑO
por sólo 3.900 Pts.

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

N.º Y NOMBRE DEL JUEGO

1. MUNCHER (MASTICADOR)
2. SKI JUMP (SALTO DE SKI)
3. BASKETBALL (BALONCESTO)
4. FROGGER (RANAS)
5. BREAKOUT (ESCAPE)
6. CRUSHER (APLASTADOR)
7. STARTREK (VIAJE A LAS ESTRELLAS)
8. MARTIAN KNOCK OUT (KAO MARCIANO)
9. BOGGLES (DESORBITADO)
10. ALIEN ATTACK (ATAQUE EXTRATERRESTRE)
11. LUNAR LANDER (ATERRIZAJE LUNAR)
12. MAZE EATER (COMEDOR DE LABERINTO)
13. MICROTRAP (PEQUEÑA TRAMPA)
14. MOTORWAY . (AUTOPISTA)
15. LABYRINTH (LABERINTO)
16. SKITTLES (JUEGO DE BOLOS)
17. RACE TRACK (PISTA DE CARRERAS)
18. SKI RUN (SLALON)
19. TANKS (TANQUES)
20. SOLAR SHIP (NAVE SOLAR)
21. TEN PINS (10 ALFILERES)
22. CARS (COCHES)
23. STOMPER (BEISBOL)
24. PINBALL (FLIPPER)
25. CAVERN (CUEVA)
26. LASER (LASER)
27. ALIEN (EXTRATERRESTRE)
28. CARGO (CARGA)
29. THE RACE (CARRERA)
30. THE SKULL (EL CRANEO)
31. ORBIT (ORBITA)
32. MUNCH (MUNCH)
33. BOWLS (BOLOS)
34. RAIDERS (SAQUEADORES)
35. FIELD (EL CAMPO)
36. DRAGGOLD (DRAGON DE ORO)
37. SPACE SEARCH (BUSQUEDA EN EL ESPACIO)
38. INFERNO (INFIERNO)
39. NIM (NIM)
40. VOYAGER (VIAJANTE)
41. SKETCH PAD (CUADERNO DE ESBOZOS)
42. BLITZ (ALARMA ANTIAEREA)
43. FISHING MISSION (MISION DE PESCA)
44. MYSTICAL DIAMONDS (DIAMANTES MISTICOS)
45. GALAXY DEFENSE (DEFENSA DE LA GALAXIA)
46. CYPHER (CIFRAR)
47. JETMOBILE (COCHE JET)
48. BARREL JUMP (SALTAR EL BARRIL)
49. ATTACKER (ATRACADOR)
50. SPACE MISSION (MISION ESPACIAL



Felicidades. Ya existe una sensacional cinta con 50 juegos que se llaman CASSETTE-50, que CASCADE ha traido para tí.

CASSETTE-50, contiene 50 excitantes juegos para cargar y disfrutar utilizando completamente las características de tu computadora.

CASSETTE-50, atraerá a gente de todas las edades, y los juegos proporcionarán muchas horas de entretenimiento para toda la familia, a la fracción del coste de otros juegos de computadora.

La mayoría de los juegos que contiene CASSETTE-50, pueden jugarse con la computadora de tamaño standard. Solamente algunos de los juegos más grandes requieren una memoria más larga (MORE RAM).

VIC-20 / ORIC
SPECTRUM / BBC / APPLE
DRAGON
ATARI / COMMODORE-64
en Julio 84

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

NOMBRE .....
APELLIDOS .....
DIRECCION .....
POBLACION ..... D.P. .....
PROVINCIA ..... TELEFONO .....
FORMA DE PAGO: ☐ TALON CONFORMADO ☐ GIRO POSTAL ☐ CONTRA REEMBOLSO

CASCADE CASCADE CASCADE CASCADE

tu Micro

DEPARTAMENTO DE PUBLICIDAD

Teléfs. 457 69 23 - 250 55 79

Correspondencia con la revista

Para lograr una mayor rapidez en la gestión de la correspondencia dirigida a la revista, debe mencionar en la esquina superior izquierda del sobre la sigla que corresponda.

DR	Director.
RD	Redacción.
PB	Publicidad.
STM	Suscripciones.
PTM	Pedidos.
CDD	Cambio de dirección.
SDI	Solicitud de información.
ABS	Anuncios breves.
CDL	Correo del lector.
SC	Servicio comercial.

Dirección para correspondencia: Apdo. de Correos 61.294. Madrid.

Precios

Precio del ejemplar
Número sencillo: 250 pesetas.
Extra julio/agosto: 450 pesetas.

Suscripciones

España
Correo ordinario: 2.500 pesetas.
Correo certificado: 2.800 pesetas.

Europa
Superficie: 3.200 pesetas.
Avión: 3.400 pesetas.

América
Superficie: 4.100 ptas. o 27 $.
Avión: 6.000 ptas. o 40 $.
Derechos de envío certificado para el extranjero: 800 pesetas.

NUEVA INFORMATICA

Lo curioso es que todos los puntos que hacen a APD altamente rentable son debidos a su nueva tecnología. Lógico.

El costo de los equipos APD es sensiblemente más bajo. Compárelo.
Sus posibilidades de crecimiento son superiores. Descúbralo.
Sin contactos, sin cables, sin problemas de refrigeración...
Un diseño MONOPLACA, compacto, exclusivo y fiable.
APD: Máxima rentabilidad.

El costo, el crecimiento y un sistema monoplaca.



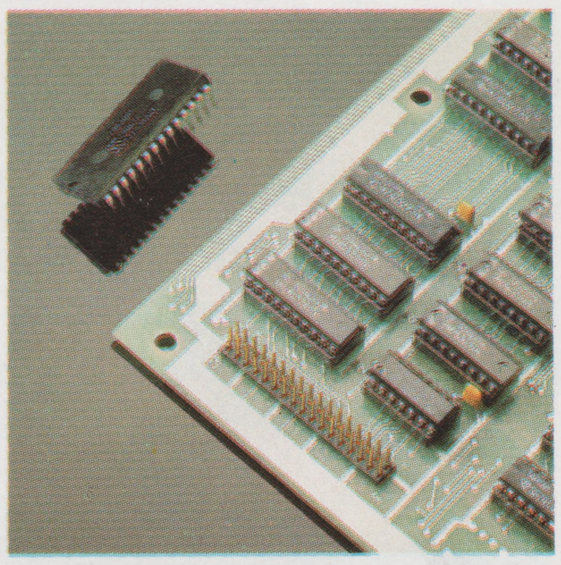
Comparta la opinión de todos los expertos, ponga a su alcance el Sistema Operativo más evolucionado y poderoso (OASIS®).
Tan sencillo que en pocas horas podrá dominarlo.
Tan poderoso que nunca se le quedará "corto".
Un sistema operativo fuerte para un equipo fiable y potente es la fórmula APD.

La superioridad del sistema operativo OASIS®



Un año es un período de garantía muy largo. Sólo equipos muy probados, muy duros y muy fiables pueden mantenerlo.
APD se lo ofrece de entrada para todos nuestros productos. Y para Vd., un año de confianza y seguridad.

¿Por qué 1 AÑO de garantía?



Un equipo APD puede controlar muchos puestos de trabajo por el mismo precio que otros sólo controlan uno. Puede realizar varios trabajos a la vez, por muy diferentes que éstos sean: CIENTIFICOS, TECNICOS, DE GESTION... Y SIMULTANEAMENTE.
APD: Tecnología, versatilidad y potencia.

Unidades inteligentes y muy versátiles.



APD©

APD es más que un ordenador, es un sistema basado en la experiencia y en las necesidades reales. Pida información.

TECNOLOGIA RENTABLE

Si desea recibir mayor información, no dude en llamarnos.

SU CONTACTO CON APD©

MADRID 435 22 65
435 23 76
435 26 77
Castelló, 63. MADRID-1

BARCELONA 209 70 00
Aribau, 267 / Párroco Ubach
BARCELONA-21

VALLADOLID 22 53 51
Acera de Recoletos, 11-4.º D
VALLADOLID

SEGUIMOS BUSCANDO DISTRIBUIDORES



16 K



GARANTIZAMOS EL CRECIMIENTO DE TU SPECTRUM

Si tienes un SINCLAIR ZX SPECTRUM de 16 K y deseas ampliarlo a 48 K ahora puedes hacerlo con toda **garantía.**

Acude a tu Concesionario Autorizado INVESTRONICA y en breve espacio de tiempo dispondrás de tu Spectrum con MAS POTENCIA.

Además INVESTRONICA ampliará por tres meses la garantía de tu equipo, independientemente de la fecha de adquisición y te obsequiará con una Cinta de Demostración de 48 K.

IMPORTANTE:

Al adquirir los productos **SINCLAIR** exija la TARJETA DE GARANTIA INVESTRONICA, única válida en todo el territorio nacional y llave para cualquier resolución de duda o reparación. INVESTRONICA no prestará ningún servicio técnico a todos aquellos aparatos que carezcan de la correspondiente garantía.

DE VENTA EN CONCESIONARIOS AUTORIZADOS.



48 K

Amplía,
ahora, tu SPECTRUM
de 16 K a 48 K.

Con garantía...
Con más garantía.

DISTRIBUIDOR EXCLUSIVO:
INVESTRONICA
CENTRAL COMERCIAL: Tomás Bretón, 60.
Tel. 468 03 00 Telex: 23399 IYCO E Madrid.
DELEGACION CATALUÑA: Camp, 80 - Barcelona - 22

J. M. PUBLICIDAD