

MICROHOBBY

AÑO I N.º 1

AMSTRAD

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

Semanal

AÑO I N.º 1

150 Ptas.

Canarias 160 ptas.

**CONVIERTE
TU AMSTRAD
EN UN
MAGNIFICO
PINTOR**

**EL
CODIGO
MAQUINA
BIT a BIT**

**664:
¿Qué aporta
el nuevo
Amstrad?**

SOFTWARE

**KNIGHT LORE
Analizamos el
programa n.º 1
en el mundo**



SI BUSCAS LO MEJOR **ERBE** Software LO TIENE

NOVEDADES PARA AMSTRAD



Pole Position

Siente el vértigo y desafío de la velocidad participando en las carreras de Fórmula 1. Tu habilidad y resistencia van a ser puestas a prueba en cualquiera de los más importantes circuitos mundiales. Consigue ser el N.º 1.



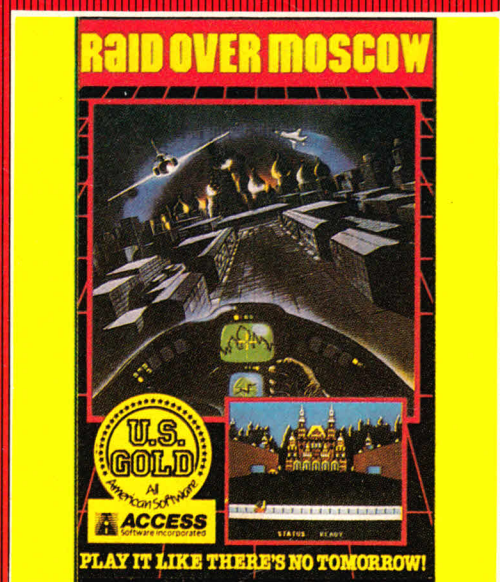
Beach Head

Por fin para tu Amstrad el juego que mayor popularidad alcanzara en Commodore y Spectrum. Desembarca a tus tropas, lucha contra los barcos y aviones enemigos, destruye sus tanques y enfrentate a la fortaleza enemiga.



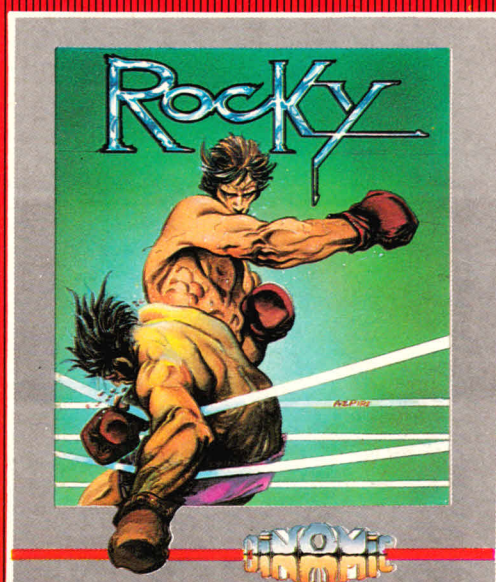
Gremlins

Vive paso a paso la película y siéntete protagonista de esta magnífica aventura íntegramente en castellano. 100 pantallas diferentes en las que tendrás que eliminar la ciudad de los temibles Gremlins. ¡Apasionante!



Raid Over Moscow

Defiende a U.S.A. y Canadá del ataque nuclear que ha lanzado Rusia contra ellos. Con tu escuadrilla habrás de hacer un viaje lleno de peligros hasta llegar al mismísimo Kremlin y destruir las bases de lanzamiento soviéticas. Gráficos y acción sensacionales.



R o c k y

Situación perfecta de un combate de boxeo en el que tendrás que luchar por el título mundial contra los más duros enemigos.

Año I • Número 1 • 3 al 10 de Septiembre de 1985
150 ptas. (sobretasa Canarias, 10 ptas.)

Director Editorial
José I. Gómez-Centurión

Director Ejecutivo
Víctor Prieto

Subdirector
José María Díaz

Redactora Jefe
Marta García

Diseño
José Flores

Colaboradores
Francisco Portalo
Pedro Sudón
Miguel Sepúlveda
Francisco Martín
Jesús Alonso

Secretaría Redacción
Carmen Santamaría

Fotografía
Carlos Candel
Javier Martínez

Portada
González Teja

Ilustradores
J. Igual, J. Pons, F. L. Frontán,
J. Sempitien, Pejo, J. J. Mora

Edita

HOBBY PRESS S.A.

Presidente

María Andrino

Consejero Delegado

José I. Gómez-Centurión

Jefe de Publicidad

Concha Gutiérrez

Secretaría de Dirección

Marisa Cogorro

Suscripciones

M.º Rosa González
M.º del Mar Calzada

**Redacción,
Administración
y Publicidad**

La Granja, s/n
Polígono Industrial de
Alcobendas
Tel.: 654 32 11

Dto. Circulación

Carlos Peropadre

Distribución

Coedis, S. A. Valencia, 245
Barcelona

Imprime

ROTEDEC, S. A. Crta. de Irún.
Km. 12,450 (MADRID)

Fotocomposición

Novocomp, S.A.
Nicolás Morales, 38-40

Fotomecánica

GROF
Ezequiel Solana, 16

Depósito Legal:

M-28468-1985

Derechos exclusivos
de la revista

**COMPUTING with
the AMSTRAD**

Representante para Argentina, Chile,
Uruguay y Paraguay, Cia.
Americana de Ediciones, S.R.L. Sud
América 1.532. Tel.: 21 24 64. 1209
BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace
necesariamente solidaria de las
opiniones vertidas por sus
colaboradores en los artículos
firmados. Reservados todos los
derechos.

Se solicitará control OJD

4 Primera plana

Las últimas novedades y acontecimientos acaecidos en el mundo del **AMSTRAD**, semana a semana.

Banco de pruebas 6

464 Versus 664.

Dos concepciones diferentes de la misma idea, compatibles entre sí. ¿Puede ocupar el 664 en el hogar el sitio de un ordenador profesional?

10 Mr. Joystick

**Knight Lore
Combat Lynx.**

Cada día más y más casas de software se pasan a **AMSTRAD**.

Para que conozcas de cerca los mejores programas de juegos para tu ordenador, **AMSTRAD** semanal te trae las últimas novedades del mercado.

14 ProgramAcción

**Ventanas 1.
El AMSTRAD**

es una máquina con sorprendentes posibilidades, para sacar el máximo partido de ella y dar un aspecto más profesional a tus programas, nada más práctico y sencillo de manejar que las ventanas de texto.

Análisis 17

Para comprender la estructura y funcionamiento de un programa, es imprescindible saber qué misión tiene cada línea en el mismo y como afecta ésta al resultado final. **AMSTRAD ANALISIS** estudia cada semana un programa que será de gran utilidad en tus propias creaciones.

18 Serie Oro

Easy Draw.

Los grandes programas no sólo se encuentran en cintas comerciales, también programadores aficionados pueden realizar grandes creaciones. **Serie oro** es un claro exponente de que la programación de alto nivel no es un coto privado de las casas comerciales.



28 Código Máquina

Del bit al byte.

Cualquiera que al examinar un listado encuentre una sucesión interminable de números sin aparente sentido, pensará que el lenguaje máquina es un mundo impenetrable; para demostrarte lo contrario, desvelamos paso a paso todos sus secretos.

Sin duda alguna 33

El largo camino existente entre la compra del ordenador y el completo dominio del mismo, está repleto de fatigas y problemas sin aparente solución. **Sin duda alguna** recorre el camino por ti.

34 Mercado Común

Para facilitar el intercambio de ideas, software y hardware entre nuestros lectores, **Mercado Común** abre sus fronteras a todo aquel que quiera comunicarnos sus inquietudes.

Primera plana

TRES EN UNO

La casa de software Picturesque, conocida, entre otras cosas, por su excelente ensamblador para el ZX Spectrum, comercializa ahora un paquete de utilidades para los usuarios del CPC464, ávidos de adentrarse en el mundo de la programación en lenguaje de máquina.

El «pack» consta de tres programas: un editor, un ensamblador y un monitor; están diseñados para trabajar juntos o por separado y, aunque, según Picturesque, el monitor es menos complejo que otros sistemas, posee todos los comandos necesarios para ejecutar y depurar programas en lenguaje máquina.

Los programas vienen en cassette, con la opción para realizar copias de trabajo en cinta o disco; esperemos que esta iniciativa de facilitar al usuario copias de seguridad, sea prontamente imitada por el resto de los fabricantes de software para el Amstrad.

El precio de estos tres programas, en Inglaterra, rondará las 4.600 pesetas.

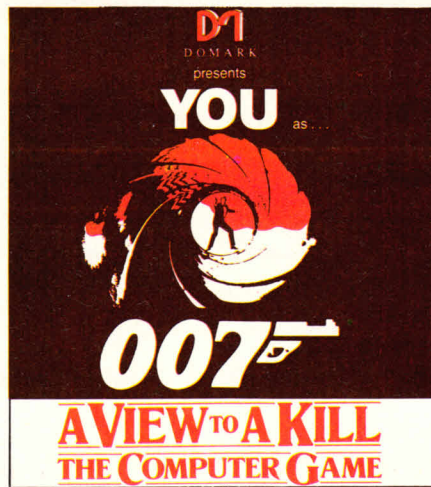
SOFTWARE DE IBM PARA AMSTRAD

En la ciudad alemana de Bonn, un científico y un economista han desarrollado una segunda unidad de disco para los ordenadores **AMSTRAD**, en formato estándar de 5 ¼ pulgadas, capaz, según ellos, de ejecutar una significativa porción del software profesional escrito para IBM que funciona bajo el sistema operativo CP/M (más de 300 programas).

Esta unidad también debería capacitar a los **AMSTRAD** para ejecutar la inmensa cantidad de software CP/M que se encuentra en formato 5 ¼, en lugar del formato de tres pulgadas escogido por la compañía inglesa.

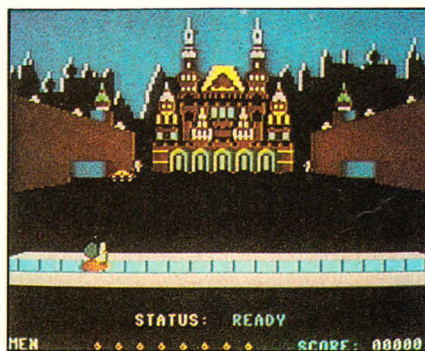
El drive soportará la transferencia de ficheros de texto desde y hasta el disco propio de **AMSTRAD**. Sin embargo, los programas CP/M no podrán transferirse.

El precio del nuevo periférico rondará las 70.000 ptas. (en Inglaterra).



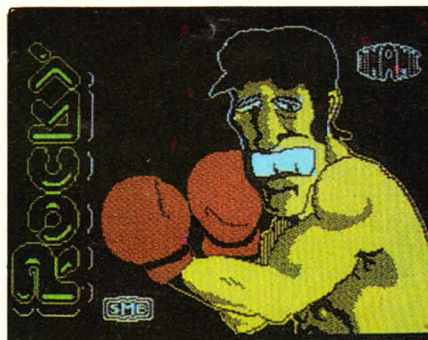
JAMES BOND HA LLEGADO A AMSTRAD

Los felices poseedores de un **AMSTRAD** no vamos a perdernos de ninguna manera las emocionantes aventuras de James Bond, el inefable y seductor 007. En esta línea, **DOMARK** presenta la versión informática de la famosa película «A VIEW TO A KILL». Consigue tu licencia para matar.



7, 8, 9, 10... KNOCK OUT

Vive la emoción de un encarnizado combate de boxeo, con Rocky. La última creación de la conocida firma española de software Dinamic. Tendrás que pelear duro para conseguir el campeonato del mundo; sólo los puños de tu oponente se interponen en tu camino hacia el triunfo final.



ESTACION LUNAR ALFA

Una excitante aventura en las más profundas cavernas lunares protagonizada por un astronauta del año 1995.

NODES OF YESOD es la última creación del sello ODIN de Computer Graphics LTD. Buena suerte en tu arriesgada misión.



OBJETIVO KREMLIN

U. S. Gold lanza para **AMSTRAD** una increíble aventura en tres dimensiones basada en un ataque nuclear soviético sobre los Estados Unidos y Canadá. Somos el jefe de un escuadrón de reactores con los cuales debemos localizar y destruir las bases soviéticas de lanzamiento de misiles.

La seguridad de Occidente está en tus manos.

A LO MACINTOSH

A la vista del éxito obtenido por su sintetizador de voz, DK'Tronics lanza ahora su nuevo producto, esta vez relacionado con el mundo gráfico.

Se trata de un lápiz óptico arropado por un sofisticado paquete de software que posibilita la elección de una paleta de color, del grosor del pincel con el que dibujamos y hasta de la posibilidad de conseguir un efecto de aerógrafo, permitiendo el diseño de espectaculares pantallas.

El usuario puede definir sus propios gráficos, moverlos por toda la pantalla y aumentarlos y disminuirlos a voluntad. Por supuesto, los dibujos finalizados pueden ser conservados en cinta o disco y editados para corregirlos cuando se desee.

El programa se gestiona mediante menús que emplean el concepto de iconos, en el más puro estilo del Apple Macintosh.

Buenas noticias para aquellos decididos a emplear su ordenador en faenas artísticas.

EDITORIAL

e

uando, hace casi un año, apareció en el mercado español la Revista Microhobby Semanal, resultó una sorpresa para muchos, tanto por su periodicidad, nunca vista antes en una publicación técnica, como por su estilo diferente, ágil y desenfadado de abordar el mundo de la informática.

Muchas voces se levantaron para augurarnos que Microhobby no cumpliría ni tres meses de vida. A la vista está que se equivocaron quienes así pensaban, puesto que ha quedado demostrado que al parecer cada semana se puede ofrecer a los lectores una información tan completa y elaborada como sea preciso, pero mucho más puntual y atractiva.

Por ello, hoy, al poner en circulación este «**Microhobby Amstrad**», también semanal, sabemos que la fórmula es válida y podemos conseguir una revista en la que la actualidad y la técnica se aúnen para mejorar el resultado final.

Aquellos lectores que conozcan ya Microhobby, verán varias coincidencias entre ambas publicaciones. El cuidado en la edición, la calidad de impresión, la abundancia de color e ilustraciones y textos de calidad, creemos que nos asemejan. Sin embargo, nos hemos esforzado en obtener una personalidad propia, y creemos haberla conseguido. El lector será quien juzgue si el resultado es o no de su agrado.

NACE UNA «AMSTRELLA»

Este ha sido nuestro *slogan* publicitario y creemos que tiene algún sentido, aparte de echarnos flores a nosotros mismos. Sin duda el Amstrad ha conquistado un puesto estelar en el mundo de la micro informática, hasta el punto de que puede decirse que ha sido el *boom* de 1985.

Aún no ha conseguido Amstrad, por número de unidades vendidas, el puesto de cabeza que ocupan otros ordenadores, pero su crecimiento ha sido espectacular y justificado. Sin duda, la relación calidad-precio que ofrece ha sido y es apabullante, y la causa de su éxito, aunque recientes disposiciones de aumento de aranceles aduaneros pue-

da enturbiar en alguna medida su futuro inmediato.

En números próximos hablaremos más a fondo del alcance de estas medidas, que afectan a todos los ordenadores del mercado, a excepción, según parece, de algunos fabricados en España y cuya dificultad para vender unidades parece que es semejante a su influencia para promover proteccionismos carentes de sentido.

Pese a todo ello, apostamos por el Amstrad, a quien auguramos que mantendrá su fuerte crecimiento en el mercado español. Al mismo tiempo queremos dejar bien sentado que somos totalmente independientes, sin vinculación ninguna al fabricante y al importador y que haremos valer esta independencia para señalar cuantos errores o defectos estimemos, o estimen nuestros lectores.

En relación a esto, no tenemos más remedio que acudir al tópico de pedir, desde aquí, la colaboración de todos para que nos escribáis con sugerencias, críticas, artículos, programas, etcétera, puesto que nuestra intención de mantener a **Microhobby Amstrad** en punta de actualidad y de interés es imposible sin vuestra colaboración.

Por último, señalar que contamos en nuestras páginas con magníficos artículos y programas procedentes de la revista británica «Computing with the Amstrad», quizás la más prestigiosa de las revistas sobre Amstrad. En este número, sirva de ejemplo el «Easy Draw», un programa para dibujar con el ordenador, que, a pesar de estar escrito íntegramente en Basic es de una potencia increíble.

En próximos números ofreceremos nuevos programas que tenemos preparados, utilidades preferentemente, y magníficos juegos. Para estar seguros de vuestras preferencias, es importante que nos escribáis en respuesta a nuestra encuesta, que encontraréis en la página 33. Además, pueden corresponderos estupendos premios.

¡Hasta la próxima semana!

LA REDACCION.

CPC664 VERSUS CPC464

Hace poco tiempo AMSTRAD volvió a sorprender a todo el mundo que pensaba, con razón, que el CPC464 (Colour Personal Computer) era todo un hito en el campo de la informática casera, de los «home computers». La compañía inglesa, en su incesante afán de superación, presentó su nueva máquina, el CPC664, revolucionando una vez más el tándem prestaciones/precio.

El equipo está formado por monitor, unidad central con unidad de disco integrada formato Hi-tachi (3 pulgadas), y sistema operativo CP/M 2.2.

Estas son sus características a grandes rasgos, y en estas líneas vamos a tratar de responder a unas preguntas que mucha gente, sobre todo usuarios potenciales, sin duda se hacen: **¿qué aporta de nuevo el CPC664 respecto a su antecesor el CPC464?** ¿Es mejor? ¿Es peor? ¿O solamente es distinto, con un campo de aplicación diferente?

Lo elegante al servicio de lo útil

Lo primero que salta a la vista del CPC664, es su nueva línea, un diseño totalmente reformado. Mientras que el CPC464 incorpora teclas multicolores, que le dotan de un aspecto tal vez excesivamente «alegre», el CPC664 muestra una apariencia mucho más profesional; un teclado bicolor, de tonos suaves, agradables a la vista y dividido en dos bloques:

— Alfanumérico, tipo QWERTY, con las letras y símbolos de costumbre más las teclas especiales que cumplen funciones de control, tales como CTRL, ESC, TAB, SHIFT, etc.

— Numérico, donde pueden verse las teclas de control del cursor, de aspecto y tacto mucho más agradable que las del CPC464, y las numéricas propiamente dichas, que además son programables, actuando como teclas de función.

Pensando sin duda en su uso por profesionales, para tratamiento de textos o tareas que requieren un uso intenso del teclado, se le ha dotado de unas

teclas robustas y fiables, tipo máquina de escribir, que responden a la primera pulsación y poseen, tanto aisladamente como en su conjunto, la curvatura adecuada para facilitar al máximo su uso; la ergonomía no ha sido olvidada a la hora de diseñar el equipo.

Unidad de disco integrada. ¡Por fin!

Un poco a la derecha del bloque numérico del teclado, se encuentra la unidad de disco integrada del CPC664, pequeña, compacta y funcional. En su parte superior, AMSTRAD ha colocado un cuadro que refleja los códigos de los diferentes colores del 664 y los de cada tecla, trantando de combinar estética con utilidad. La unidad es capaz de alojar en su interior «minidiscos» de 3 pulgadas, con una capacidad de 169 kbytes por cara una vez formateados, muy cómodos de manejar y transportar debido a su exiguo tamaño. Poco puede reprochársele a este lector de discos, salvo quizás dos cosas:

1. La capacidad de almacenamiento del disco puede llegar a ser un poco escasa para aquellos que deseen manejar grandes cantidades de datos. El empleo de una segunda unidad de discos puede llegar a ser muy necesaria. Sin embargo, los que requieran un almacenamiento de datos de mediano tamaño, o planeen emplear el 664 en labores de gestión «caseras», tendrán más que suficiente con una sola unidad. Recordemos que la capacidad TOTAL del disco, sumando ambas caras, ronda o incluso supera los 340 kbytes (según el tipo de formateo que hayamos escogido).



J. IGUAL

2. La unidad es relativamente ruidosa. Aunque el sonido que produce no es desagradable en absoluto, cuando está en activo se le oye con claridad y una especie de suave temblor atraviesa al 664. Esta peculiaridad, al principio nos sorprendió mucho, pero al poco nos acostumbramos a ella hasta dejar de advertirla. Además, que nosotros sepamos, la calificación más deferente que ha recibido alguna unidad de disco en el caso de otras máquinas ha sido «notablemente silenciosa». Dicho con otras palabras, también las demás hacen ruido, ¡y a un precio mucho más alto!

La velocidad de acceso al disco es bastante aceptable; se encuentra posicionada en esa amplia tierra de nadie entre las unidades famosas por su lentitud, y aquéllas de las que sólo hemos oído hablar por su exorbitante precio.

La biblioteca CP/M al alcance de la mano

No hay que olvidar en ningún caso que la unidad de disco dota al 664 de una enorme potencia; no sólo en rapidez para almacenar y recuperar programas, sobre todo lo faculta para acceder a la gran cantidad de software profesional que funciona bajo el sistema operativo CP/M. Por «software profesional» se entiende programas de todo tipo hechos por expertos, programas de gestión empresarial y comercial, lenguajes de programación, etc. Tengamos la seguridad de que casi cualquier programa que se nos ocurra ya ha sido hecho y funciona bajo CP/M.

Para los que aducen que la mayoría de los programas CP/M están en disco de formato 5 1/4 y no en el formato AMSTRAD, el número de aplicaciones CP/M adaptadas al AMSTRAD crece día a día; sin embargo, en el momento actual, la balanza se inclina por el lado de los discos 5 1/4 en cuanto al número de programas disponibles.



Banco de pruebas



Así pues, una de cal y otra de arena, porque la implementación del lenguaje LOGO que viene en la otra cara del disco es francamente buena; posee los comandos tradicionales de gráficos tortuga y, además, casi todos los necesarios para el tratamiento de listas, funciones matemáticas y trigonométricas, etc.

Los programas del segundo disco están muy bien; son potentes y sencillos de manejar. Por ejemplo, el programa «RANDOM FILES», nos permite usar ficheros aleatorios desde nuestros propios programas, añadiendo nuevos comandos al Basic AMSTRAD. Más de un propietario del 664 con inclinaciones artísticas disfrutará enormemente con el programa de diseño de gráficos, y los que deseen emplear la máquina en labores de gestión, encontrarán una buena ayuda en los programas de proceso de texto y base de datos. Debe quedar claro, no obstante, que el uso a plena potencia de estos programas probablemente requiera una impresora y una segunda unidad de disco.

Los manuales mejoran

En cuanto a los manuales, los que conozcan la documentación del 464 probablemente se temerán lo peor, es decir, un diccionario de comandos traducido con bastante «alegría». Bien, afortunadamente, la documentación del 664 está bastante más elaborada, y la traducción del original inglés se ha cuidado bastante; su lectura resulta agradable y amena.

En cuanto al sistema operativo y al Basic del 664, Locomotive Software ha hecho cambios importantes. Las modificaciones del firmware han sido profundas, aunque la mayoría de ellas permanecen transparentes al usuario. Respecto al Basic, se han añadido nuevos comandos que afectan a muchas facetas de la programación: gráficos, tratamiento de errores y de interrupciones, etc.

También se ha mejorado el manejo de ficheros de disco desde Basic respecto al 464. Por ejemplo, para borrar un programa del disco en éste, habría que teclear:

```
a$ = «programa. bas»
| era ,@a$
```

mientras que en el 664 basta con decir:

```
| era ,«programa.bas»
```

y así sucesivamente con el resto de los comandos del AMSDOS.

Buen paquete de programas de regalo

Al comprar el 664, AMSTRAD entrega una serie de programas con idea de que el usuario pueda comenzar a utilizar su ordenador inmediatamente. Están contenidos en dos discos:

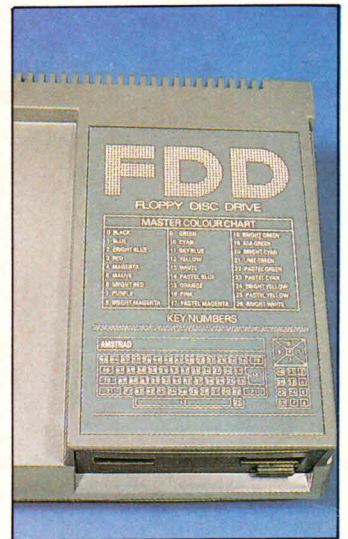
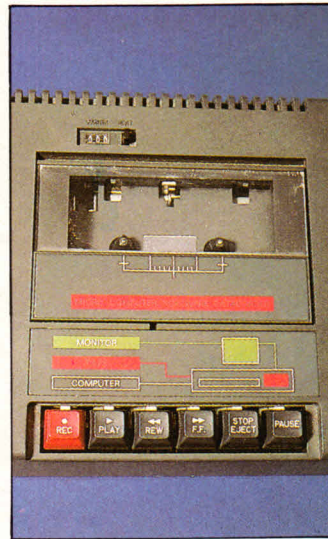
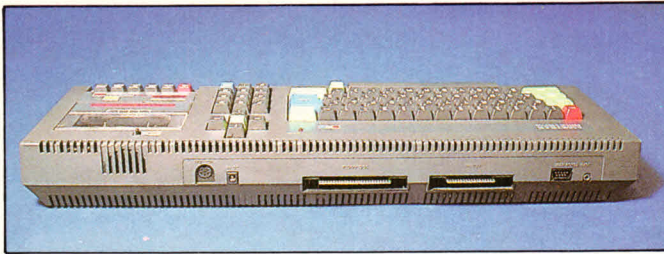
- sistema operativo CP/M y una versión de LOGO de Digital Research.
- Un segundo disco que contiene los siguientes programas:
 - BASE DE DATOS
 - PROCESO DE TEXTOS
 - RANDOM FILES (ficheros aleatorios de disco)
 - DISEÑADOR DE GRAFICOS
 - PUZZLE
 - ANIMAL, VEGETAL Y MINERAL (el clásico juego en que el ordenador adivina en qué animal, vegetal o mineral estamos pensando mediante las res-

puestas que le damos a una serie de preguntas).

El CP/M es un conjunto de programas de utilidades para el manejo de ficheros de disco; incluye dos programas que podríamos calificar de sorprendentes: un ensamblador y un «debugger» para depurar programas escritos en lenguaje máquina. La sorpresa proviene de que el ensamblador sólo acepta código fuente del procesador 8080 (!) y no del Z80, o sea, que aprendemos los mnemónicos del 8080 o no podemos utilizarlo. De este desastre no tiene la culpa AMSTRAD, sino Digital Research, creadora del CP/M, y la inclusión por AMSTRAD de un sistema operativo que incorpora en el disco programas inútiles se justifica por la gran cantidad de software a la que tenemos acceso.

NUEVOS COMANDOS DEL CPC664

	COMANDO	TAREA QUE REALIZA
TEXTO	CLEAR INPUT	BORRA EL BUFFER DE TECLADO
	COPYCHR\$	LEE UN CARACTER DE PANTALLA EN LA POSICION MARCADA POR EL CURSOR
	CURSOR	MUESTRA O NO EL CURSOR EN PANTALLA
GRAFICOS	FILL	LLENA UNA SUPERFICIE CON COLOR
	FRAME	SINCRONIZA EL MOVIMIENTO DE FIGURAS EN PANTALLA PARA EVITAR PARPADEOS
	GRAPHICS PAPER GRAPHICS PEN MASK	COLOR DE FONDO SOBRE EL QUE SE DIBUJA TINTA PARA EL DIBUJO DE RECTAS Y PUNTOS
INTERRUPCIONES	ON BREAK CONT	CONFIGURACION DE PIXELS ON/OFF EN UN BYTE QUE CORRESPONDE A UNA DIRECCION DE PANTALLA
ERRORES	DERR	SE IGNORA LA PULSACION DE LA TECLA ESCAPE
CALCULO	DEC\$	CONTIENE EL NUMERO DE ERROR PRODUCIDO EN UN PROCESO DE LECTURA/ESCRITURA EN EL DISCO
		ESCRITURA FORMATEADA DE NUMEROS, USANDO UNA CONVENCION PARECIDA A «PRINT USING»



No se ha olvidado comentar con la suficiente profundidad los dos sistemas operativos que posee el AMSTRAD, el AMSDOS y el CP/M. La descripción del programa de LOGO está tan bien conseguida como en el caso del manual para el disco del 464, siempre que quede claro que su objetivo no es enseñar a programar en este lenguaje.

Compatibles, sí pero...

Ya que hablamos de software, es conveniente afirmar que la compatibilidad entre programas escritos para el 464 y el 664 es casi absoluta; «casi» quiere decir que los programas escritos en el Basic del 464 correrán todos en el 664, pero la recíproca no es cierta si se usan los nuevos comandos de este. Respecto a los programas comerciales, juegos sobre todo, si los programadores han respetado las especificaciones de AMSTRAD respecto al uso del firmware, no debe haber problema: funcionarán en el 664; de lo contrario...

Tanto el 464 como el 664 se encuentran bien comunicados con el mundo exterior; el 464 posee las siguientes conexiones:

- 1— Conexión al monitor.
- 2— Conexión a la red.
- 3— Interface para la unidad de disco.
- 4— Salida Centronics (paralela) para impresora.
- 5— Port de usuario (conexión para joystick, entre otras cosas).
- 6— Salida estéreo de sonido.

Mientras que su hermano mayor tiene:

- 1— Conexión para la segunda unidad de disco.
 - 2— Salida al monitor.
 - 3— Salida a red.
 - 4— Conexión a red del lector de discos a través del monitor.
 - 5— Bus de expansión.
 - 6— Salida Centronics.
 - 7— Conexión para cassette.
 - 8— Conexión para joystick.
 - 9— Salida estéreo de sonido.
- Aunque son bastantes parecidos, el 664 tiene algunas posibilidades de comunicación más.

Es el momento de responder a

QUE APORTA DE NUEVO...

A) EL TECLADO:
— compacidad — facilidad de manejo — belleza — ERGONOMIA

B) LA UNIDAD DE DISCO:
— integración
— rapidez en la transferencia de datos.
— promocionar el software en disco.
— ACCESO A SOFTWARE PROFESIONAL HASTA AHORA RESTRINGIDO A EQUIPOS DE MAYOR CATEGORIA Y PRECIO.

C) EL BASIC:
— grandes facilidades para el programador aficionado en el manejo de gráficos/color.
— comandos más potentes para el tratamiento de errores (incluso del disco) e interrupciones.
— comandos más simples y lógicos para el manejo de ficheros en disco, en la línea del CP/M.
— MAYOR SOFISTICACION UNIDA A MAYOR SIMPLICIDAD.

D) LOS PROGRAMAS DE REGALO:
— sencillez de manejo y potencia.
— variedad.
— QUE SON DE REGALO.

E) LAS COMUNICACIONES:
— mayores posibilidades y flexibilidad.

F) LOS MANUALES:
— bien estructurados.
— información suficiente pero no exhaustiva de cara a los usuarios muy experimentados.
— lenguaje sencillo.
— TRADUCCION MAS ELABORADA.

Llegados ya a la recta final de este artículo, podemos intentar responder a las preguntas que nos planteábamos al principio del mismo. Aunque se ha dado primacía al AMSTRAD CPC664 por tratarse, a nuestro parecer, de una auténtica revolución en el campo de los ordenadores caseros, si no por tecnología puntera, si por prestaciones/precio, el otro retoño de AMSTRAD, el CPC464, también posee un lugar bajo el sol muy definido. No cabe duda que éste ha sido pensado para pisar fuerte en el terreno de la diversión, de los juegos por ordenador y de las pequeñas utilidades que pueden funcionar satisfactoriamente con cinta de cassette, mientras que el

664 mantiene un enfoque profesional a cara descubierta. Es, o pretende ser, el ordenador del pequeño negocio, de la gestión del hogar y del programador que aspira a manejar un equipo serio y eficiente, abierto al mundo standard del CP/M y de los 8 bits. Sin embargo, estos dos campos de aplicación, lejos de ser antagónicos, en el caso de AMSTRAD pueden convivir amigablemente. Con un gran alarde de previsión, el 464 puede acceder al mundo del disco y el 664 al del cassette. Dos ideas diferentes pero compatibles, uniendo estrechamente dos mundos separados hasta ahora por el precio. ¿Quién da más?

Suscríbete... y uno de estos tres sensacionales juegos será tuyo... ¡GRATIS!

M.H. AMSTRAD te da a elegir entre tres de los mejores juegos existentes en el mercado para AMSTRAD; **POLE POSITION**, **DALEY THOMPSON'S DECATHLON** y **BEACH HEAD**, cualquiera de los cuales puede ser tuyo solamente con suscribirte a nuestra revista. **Aprovecha esta ocasión excepcional** y ahorra 2.100 pesetas (precio de venta del programa) más el importante descuento que se produce en el precio de cada número, por el hecho de ser suscriptor. Disfruta de las ventajas que supone recibir cómodamente tu revista a domicilio y de la seguridad de tener tu ejemplar aunque se haya agotado en los quioscos.

Envíanos tu boletín de suscripción y no le des más vueltas, el número de juegos para regalos de suscripción aunque grande, es limitado y éstos se podrían agotar mientras lo estás pensando.

BEACH HEAD producido por U.S. GOLD es una misión de desembarco en una costa fuertemente defendida por las fuerzas aeronavales enemigas. Debes conducir tu flota hacia la bahía y repeler el ataque aéreo, si lo consigues tu siguiente obstáculo será una flotilla de destructores y acorazados, superada la cual desembarcarás tus anfibios en las arenas de la bahía, éstos deben superar las defensas costeras y llegar a la fortaleza que es el objetivo final.

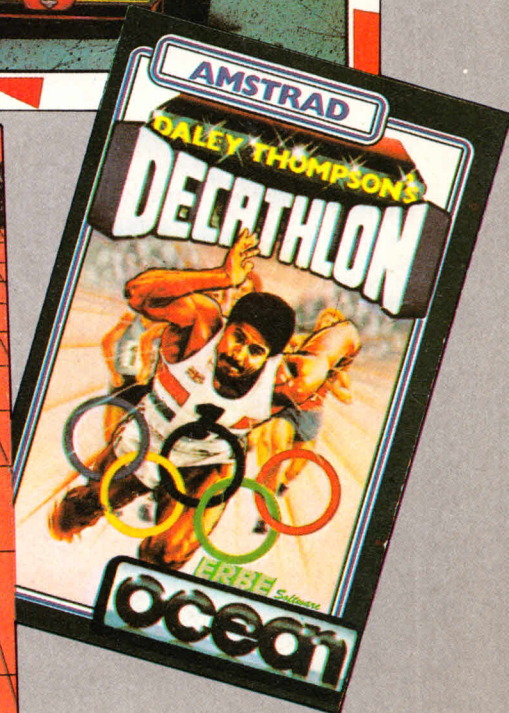
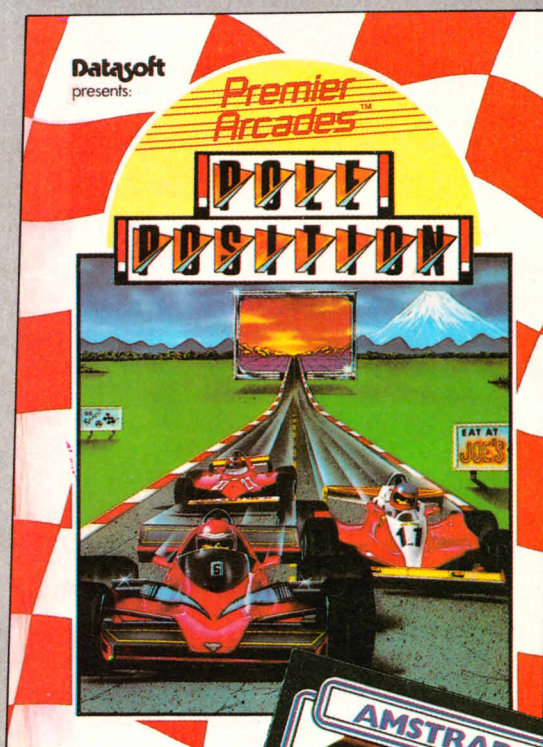
POLE POSITION es la última creación de DATSOFT en la que podrás experimentar toda la emoción de un gran premio automovilístico, vuelta de clasificación y carrera cronometrada contra tus adversarios.

DALEY THOMPSON'S DECATHLON con este juego OCEAN enciende la llama olímpica y te reta a superar los récords de los campeones de todos los tiempos, el decathlon se desarrolla en dos días de competición y se compone de las siguientes pruebas:

PRIMER DIA: 100 m lisos, salto de longitud, lanzamiento de peso, salto de altura y 400 m lisos.

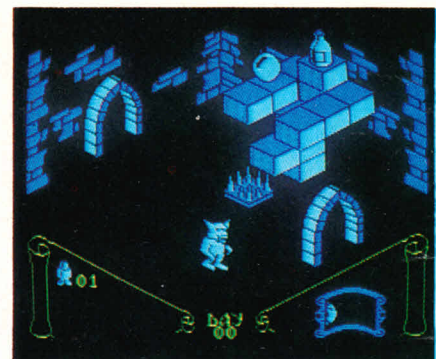
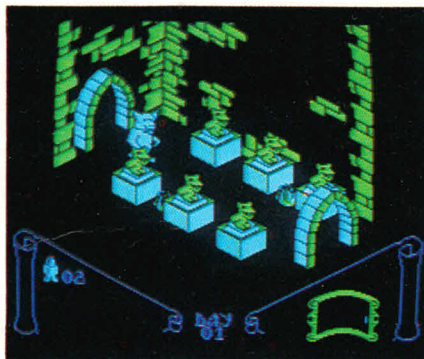
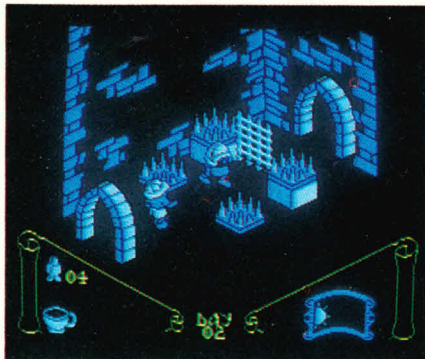
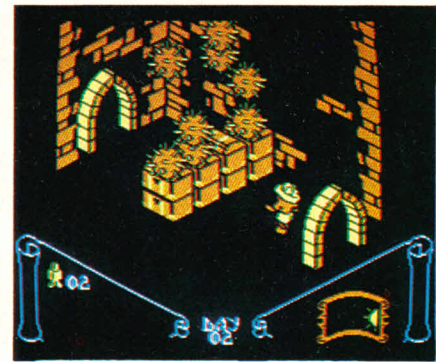
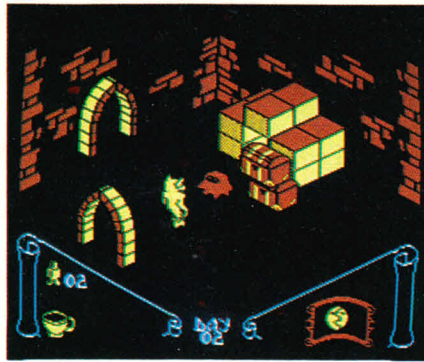
SEGUNDO DIA: 110 m vallas, lanzamiento de disco, salto con pértiga, lanzamiento de jabalina y los 1.500 m.

Utiliza el cupón
adjunto a la revista
o suscríbete por
teléfono
(91) 733 50 12
(91) 733 50 16

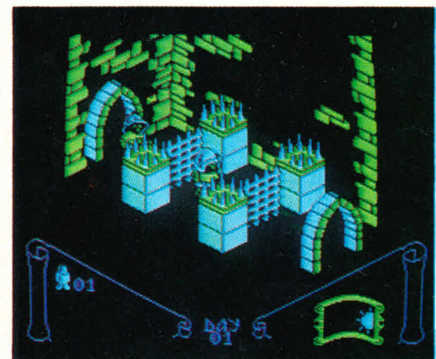
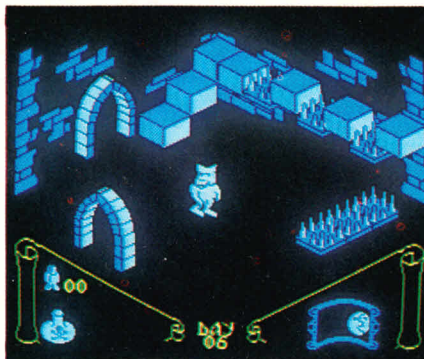


KNIGHT LORE

ULTIMATE



KNIGHT LORE, sin duda, es el juego que ha marcado la frontera entre el viejo software, y el software de la segunda generación. Su difusión y popularidad han llegado a tales cotas, que puede considerarse como el número uno mundial entre los programas para ordenador. Con su lanzamiento, **ULTIMATE** asombró a los consumidores de software recreativo, presentando un juego con una estructura completamente revolucionaria. Sorprendiendo también a los programadores más expertos, por los sensacionales gráficos conseguidos y el movimiento logrado, difícilmente imaginables hasta su aparición.

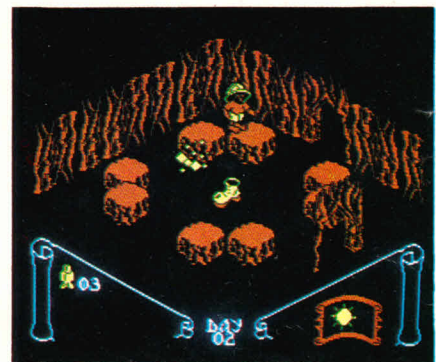


El **KNIGHT LORE** vio la luz como juego en el **ESPECTRUM**, ordenador doméstico para el cual han sido creados muchos de los juegos que posteriormente se han adaptado al **AMSTRAD**.

Con su aparición, **ULTIMATE** marcó un hito en el mundo de los juegos de ordenador, utilizando las técnicas de programación más sofisticadas, que dotan a los personajes de un movimiento prácticamente perfecto.

A lo que añadió una concepción espacial totalmente revolucionaria, que nos permite tener una visión tridimensional de todas las pantallas en las que nos encontramos.

La única limitación fue el propio **SPECTRUM**, por lo cual el juego hubo de realizarse solamente en dos colores. La versión para **AMSTRAD** utiliza el modo 1 de pantalla, con lo cual obtenemos una definición superior a la del **SPECTRUM**, y dispone-



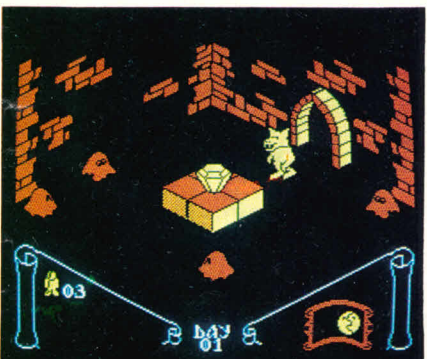
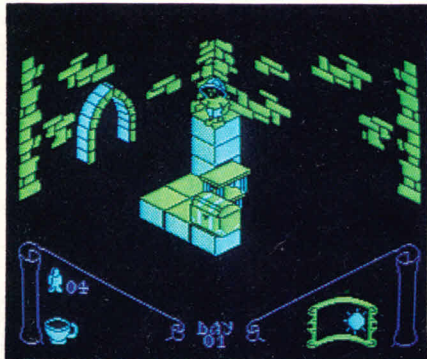
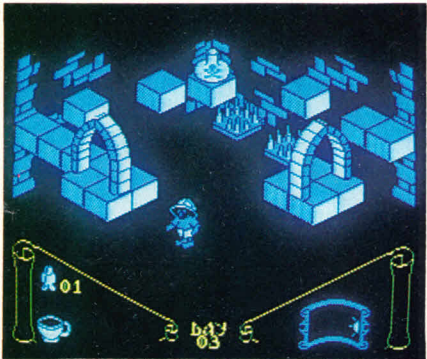
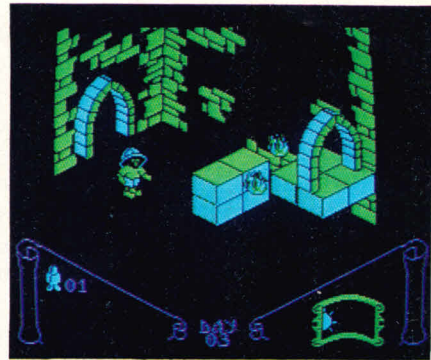
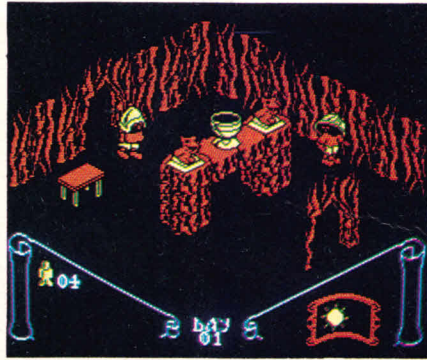
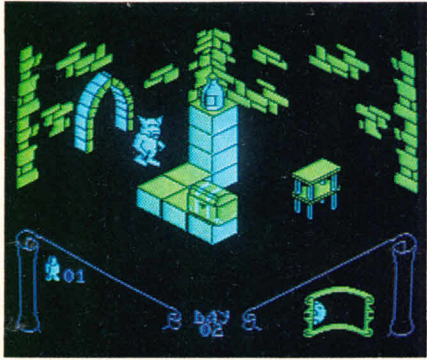
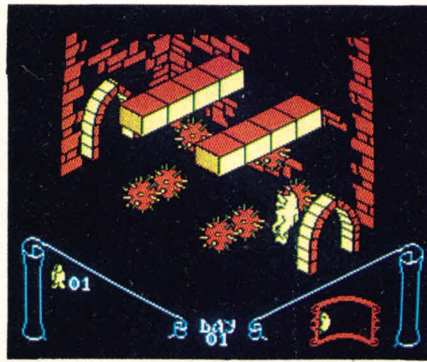
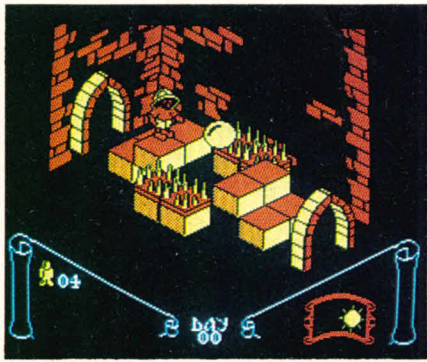
mos de cuatro colores por pantalla, lo que mejora considerablemente la versión original.

KNIGHT LORE es la última aventura de una trilogía de juegos creados para **SPECTRUM**, su protagonista es **SABREMAN** un intrépido explorador en busca de las aventuras más audaces, que no teme ningún peligro.

La primera aventura es **SABRE WULF** y se desarrolla entre la exuberante vegetación de la jungla



Mr. Joystick



amazónica, nuestra meta es encontrar el amuleto mágico que nos abre la cueva del hechicero.

La segunda UNDERWULDE, tiene lugar en un fastuoso castillo del siglo XVII, del cual debemos escapar por una de las salidas custodiadas por el diablo alado.

La culminación de la trilogía es **KNIGHT LORE**, en esta nueva aventura SABREMAN está poseído por un hechizo que le transforma en hom-

bre lobo, con la salida de la luna.

Nos encontramos en el castillo del mago Melkior, donde vagando por sus intrincadas salas y estancias, debemos ir recogiendo los ingredientes necesarios para componer la pócima que acabará con el hechizo maldito.

Estos deben ser vertidos en la gigantesca caldera de Melkior en un orden determinado, de forma que si nos acercamos a la olla mágica con el ingrediente adecuado, la nube de estrellas que emana de la caldera se convertirá en centelleante. En caso de no llevar el ingrediente preciso, seremos atacados por la masa de estrellas obligándonos a abandonar la estancia a toda prisa.

Además de los ingredientes que componen la pócima mágica, existen diversos objetos que son imprescindibles para superar los obstáculos que surgen a nuestro paso.







Aprender a utilizarlos en nuestro

favor es vital para el desarrollo del juego. Colocándolos estratégicamente y subiéndonos en ellos podremos escalar muros demasiado altos para nuestra estatura.

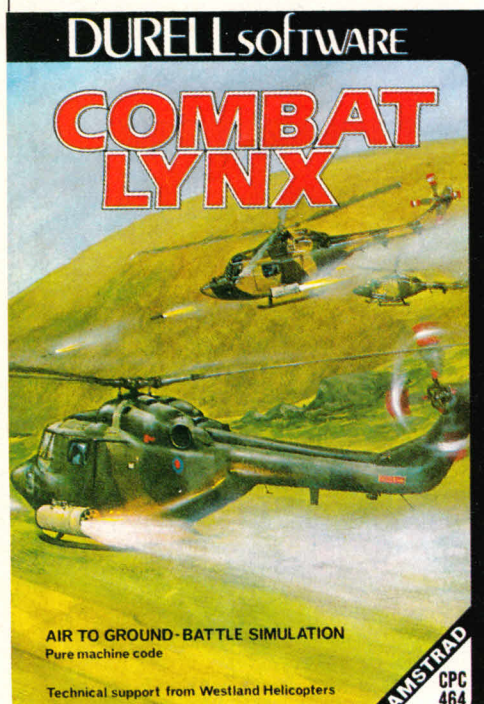
El castillo está formado por un intrincado laberinto de 128 salas, plagadas de obstáculos y los más inesperados peligros, en algunas de las cuales se encuentran repartidos los objetos que necesitamos para lograr el éxito.

Sacar a SABREMAN del hechizo maléfico que le posee es una tarea que sólo tú puedes lograr, suerte.



-  HORRIBLE
-  UN ROLLO
-  PASABLE
-  BUENO
-  MUY BUENO
-  OBRA MAESTRA

COMBAT LYNX



dos los aspectos que constituyen la realización de una misión de este tipo. Por tanto debemos tener en cuenta el combustible que cargamos, el peso del personal que transportamos de una base a otra, el peso de la munición de distintos calibres y misiles con que armamos el helicóptero, y las minas que transportamos para lanzarlas en las inmediaciones de nuestras bases.

También debemos manejar los controles de vuelo. Altimetro, indicador de velocidad, indicador de rumbo, temperatura de motores, nivel del fuel, etc., son datos a tener en cuenta en el desplazamiento entre base y base.

La comunicación con nuestro cuartel general, es de vital importancia para recibir información sobre los movimientos de tropas enemigas y pedir informes sobre la situación de nuestras bases, tanto de su posición en el mapa, como de las existencias actualizadas de combustible y armas.

Por su parte, el ordenador central de la división de inteligencia de nuestro estado mayor, nos transmitirá a la pantalla de mapas un plano completo de la porción de terreno que queramos observar, con la posición de las tropas enemigas y la de nuestras bases, siempre que solicitemos información.

Las misiones de caza que debemos efectuar, estarán condicionadas por la clase, cantidad de armamento y munición que llevemos en el helicóptero. Este puede ser armado en cada base en

la que tomemos tierra, con la única limitación del peso máximo de carga que admite el mismo. Al armar nuestro helicóptero repetidas veces en la misma base, la reserva de combustible, munición, minas y misiles contenida en ella se agota, siendo únicamente la base correspondiente al cuartel general la que tiene una capacidad ilimitada para surtirnos de cualquier tipo de material.

Nuestros objetivos son los blindados enemigos, transportes de tropas, nidos de ametralladoras, helicópteros y reactores.

Las misiones de transporte, tienen por objeto suministrar soldados de refresco a las distintas guarniciones de las bases, y transportar a los soldados heridos al cuartel general para su rehabilitación y posterior utilización en otras zonas de combate.

Nuestro enemigo intentará derribarnos lanzando sobre nosotros misiles aire-aire, para los cuales somos un blanco fácil en vuelo a gran altura, una buena medida para evitar ser alcanzado por éstos, es mantenernos en vuelo rasante, ya que a baja altura no podemos ser detectados por el radar enemigo.

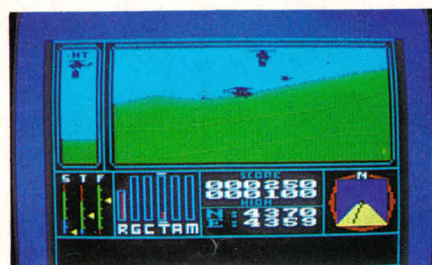
Todos estos factores hacen que nos encontremos ante un juego donde la estrategia es una pieza clave, a lo que hay que unir la conducción de un helicóptero, combinada con el fuego de nuestras armas contra objetivos móviles, algunos de los cuales son inmunes a los

En la guerra moderna el helicóptero es un arma vital tanto en misiones de ataque como en misiones de apoyo, transporte de tropas y abastecimiento de unidades en primera línea. Debido a su maniobrabilidad y capacidad de volar a baja altura, y a la gran variedad de armas que puede incorporar, resulta un enemigo fatal para blindados, transportes de tropas, y todo tipo de unidades armadas con medios convencionales que poseen una menor movilidad.

COMBAT LYNX es una simulación de una misión de defensa de un grupo de bases aliadas del ataque de una división acorazada enemiga. El objetivo principal de las tropas enemigas es tomar el cuartel general aliado, emplazado en la BASE 1.

Tu misión, con ayuda de tu **COMBAT LYNX**, un helicóptero ligero de caza y transporte, es dar cobertura aérea a todas las bases que constituyen la fuerza aliada. Para lo cual debes aprovechar al máximo todas las posibilidades que te ofrece la máquina que pilotas.

Para darle un aspecto real al juego, se han estudiado minuciosamente to-



efectos de armas de bajo calibre y requieren la utilización de armamento altamente sofisticado, del cual debemos proveernos en los puntos adecuados.





PRESENTA

UN ORDENADOR **MUY PERSONAL**



el nuevo

AMSTRAD CPC-664

(CON UNIDAD DE DISCO INCORPORADA)

¡A UN PRECIO INCREIBLE!

AMSTRAD CPC-664 (f.verde) 109.500 Ptas

AMSTRAD CPC-664 (color) 134.500 Ptas

VEN A VERLO A NUESTROS CENTROS

Modesto Lafuente, 63 ■ José Ortega y Gasset, 21 ■ Colombia, 39 ■ Padre Damián, 18 ■ Fuencarral, 100
Tel. 254 88 36 ■ Tel. 411 28 50 ■ Tel. 458 61 71 ■ Tel. 259 86 13 ■ Tel. 221 23 62

SOLICITA GRATUITAMENTE NUESTRO CATALOGO DE PRODUCTOS

LAS VENTANAS I

VENTANAS GRÁFICAS

El Amstrad posee una característica revolucionaria en equipos de su categoría: la posibilidad de disponer de hasta ocho áreas distintas en pantalla, cada una de las cuales se comporta de la misma forma que una pantalla completa, con sus propios colores, scroll, etc. El uso de esta técnica de programación añade una dimensión completamente nueva a los programas escritos para esta máquina.

Antes de estudiar en profundidad su manejo, vamos a pasar revista a algunas peculiaridades del Amstrad Basic que nos permitirán entenderlas mucho mejor.

Cuando encendemos el Amstrad, o al realizar un RESET (pulsando **SHIFT+CTRL+ESCAPE**), hay dos «pantallas» presentes en el monitor, las llamadas «pantallas de texto» y «pantalla de gráficos». Una se emplea para escribir caracteres, palabras, en ella, y en la otra puede hacerse todo tipo de dibujos.

La razón de que no se aprecie a simple vista la existencia de ambas, es que solapan una sobre otra y, además, ocupan toda la pantalla.

Vamos a comprobarlo mediante un pequeño experimento: realiza un RESET del ordenador y teclea CLS (de las palabras inglesas *Clear Screen*, borrar pantalla); el mensaje READY se coloca en la parte superior de la pantalla.

Haciendo RESET de nuevo y al teclear CLG (*Clear Graphics*, borrar gráficos), el mensaje READY aparece en otro lugar.

Así que realmente hay dos pantallas superpuestas diferentes, ocupando la totalidad del espacio visual del ordenador. De momento vamos a dejarlas así, aunque luego estudiemos las distintas formas de manejarlas, normalmente fraccionándolas en áreas diferentes llamadas «ventanas».

Los tres modos de pantalla

Como ya es sabido, el Amstrad puede trabajar en tres modos distintos (comando **MODE**); la pantalla de texto se ve afectada por ellos en el número de caracteres que pueden escribirse por línea, y en el tamaño de los mismos, como se puede apreciar en el siguiente esquema:

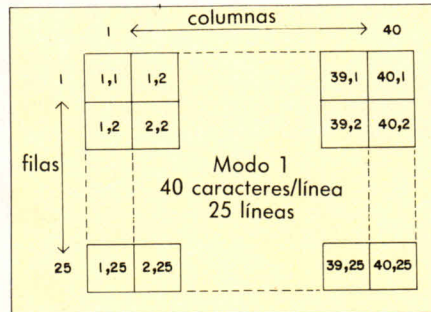


Figura 1: Coordenadas de la pantalla de texto, modo 1.

	CARACTERES		
	FILAS	COLUMNAS	POR FILA
MODE 0	25	20	20
MODE 1	25	40	40
MODE 2	25	80	80

Cada modo convierte la pantalla en una matriz de dimensión filas-columnas, ajustando el tamaño del carácter según convenga.

Existe un sistema de coordenadas que utilizaremos para saber con exactitud en que parte de la pantalla se situará nuestro texto (figuras 1, 2 y 3), y dos comandos básicos que lo manejan, **TAB** y **LOCATE**.

El primero sólo nos permite controlar en qué columna escribimos (ver programa número 1), mientras que mediante el segundo, podemos especificar la columna y la fila, por este orden, donde irá a parar nuestro texto (ver figura 1 y programa número 2).

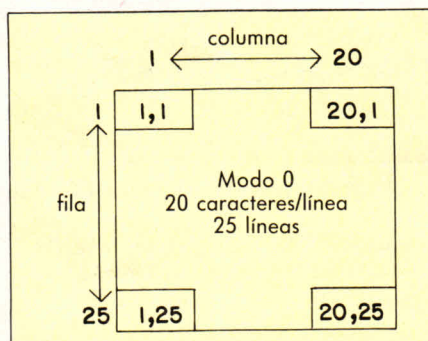
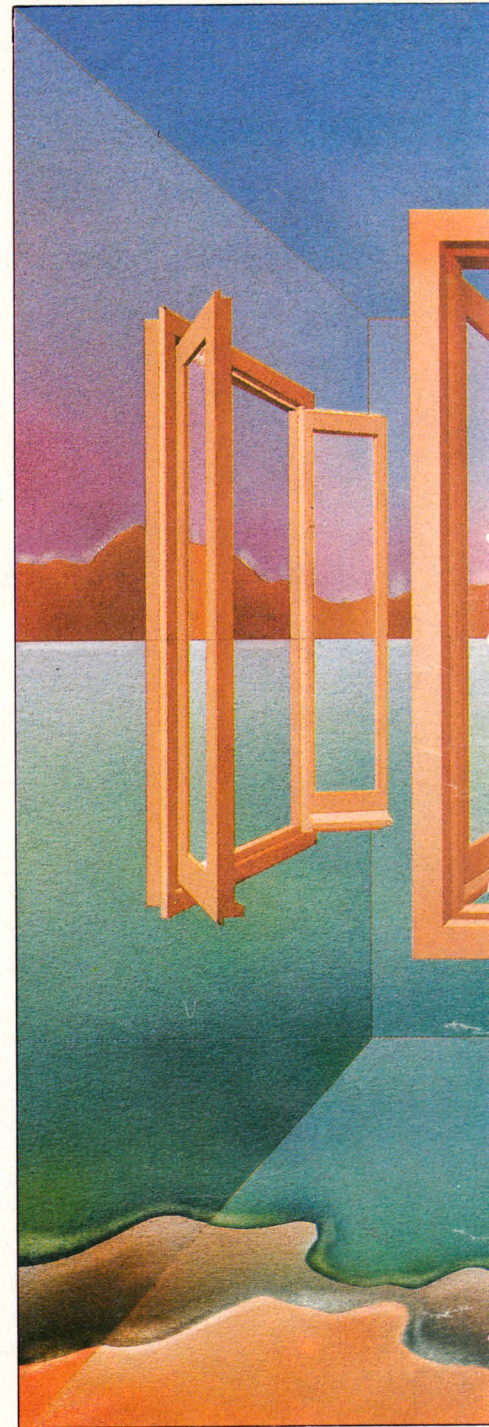


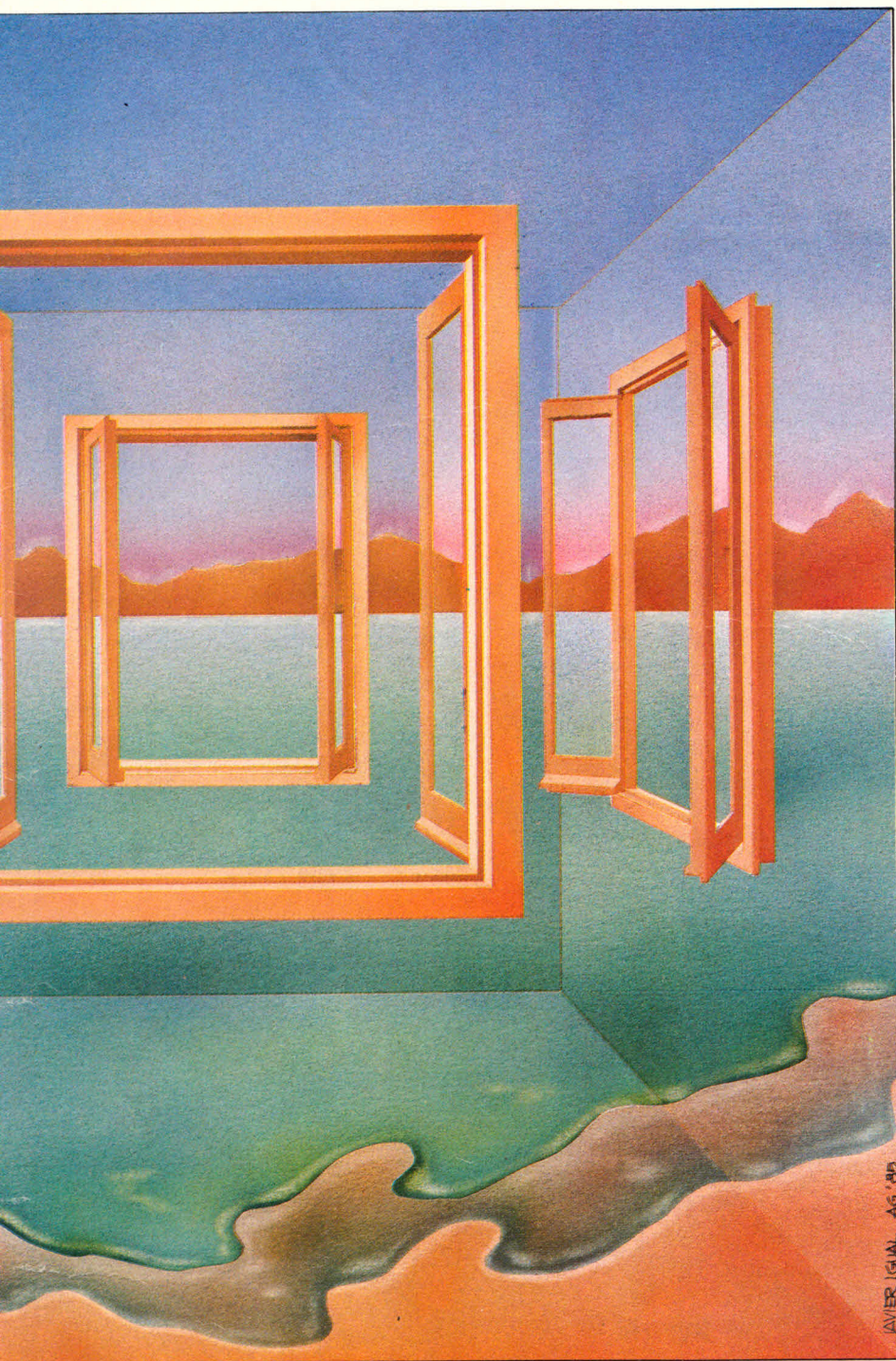
Figura 2: Pantalla de texto, Modo 0.



Llegados a este punto, si has ejecutado los dos primeros programas, tal vez te preguntes con algo de sorpresa para qué sirve el comando **LOCATE** si produce el mismo resultado que **TAB**.

Esperamos que el programa número 3 responda a esa pregunta. La versatilidad de **LOCATE** permite realizar con sencillez y lógica «recorridos» de texto a lo largo de la pantalla que, mediante **TAB**, necesitarían una programación bastante más forzada.

Comparando una vez más las figuras 1, 2 y 3, se puede ver que exis-



ten notables diferencias entre los tres modos, lo cual puede producir enojosos problemas con el comando LOCATE si, en nuestros programas, cambiamos inadvertidamente de modo manteniendo fijos los parámetros del anterior.

El comando Window

Una vez vistos estos comandos que nos serán de gran utilidad muy pronto, podemos entrar verdaderamente en materia; coloca al ordenador en MODE 1 (o bien haz un RESET) y tecllea:

WINDOW 10,20,12,16

Lo primero que llama la atención es que el mensaje READY aparece en un lugar un tanto extraño; eso quiere decir que todo va bien. Lo que hemos hecho es restringir la pantalla de texto a una ventana (¡por fin!), cuyas dimensiones y posición se muestran en la figura 4.

Para poder verla con mayor claridad, tecllea:

PAPER 3:CLS

y, por supuesto, ENTER. Suceden dos cosas interesantes:

— primero, estos comandos no aparecen donde deben

Program Acción

— segundo, el comando CLS ha hecho patente la superficie de la ventana, y, como puedes ver, no afecta más que a dicha área; el texto del resto de la pantalla aparece inalterado.

Como no hemos hecho nada que afecte a la pantalla de gráficos, el comando CLG conseguirá que TODA la imagen desaparezca.

El comando WINDOW trabaja de la siguiente manera: los dos primeros números definen, respectivamente, los lados izquierdo y derecho de la nueva ventana, en términos del sistema de coordenadas de la pantalla de texto. Análogamente, los dos parámetros restantes configuran las partes superior e inferior (figura 4).

Los comandos TAB, LOCATE, PRINT, PAPER y PEN hacen relación ahora a la nueva ventana; trabajan en la forma habitual, excepto que están sujetos a las restricciones de tamaño que la dimensión de la ventana marque.

Tecllea, por favor, LOCATE 1,1 : PRINT « * ». El asterisco aparecerá en la esquina superior izquierda; una vez que «abrimos» una nueva ventana, mientras no se indique lo contrario, las coordenadas para LOCATE y otros comandos de este tipo, comienzan siempre en la esquina superior izquierda (figura 5). Observa, de paso, que nuestra ventana tiene 5 líneas de 11 caracteres cada una, en lugar de 4 por 10 como cabría esperar.

La mejor forma de apreciar las posibilidades de esta opción del Ams-

trad es practicar; una buena idea puede ser experimentar con ventanas de diferentes tamaños, empleando los comandos de escritura y color mencionados anteriormente.

Para volver a la pantalla de texto normal, se puede teclear
WINDOW 1,40,1,25
o, más sencillamente **MODE 1**.

Utilidad de las ventanas

Tal vez convenga aclarar en este momento para qué puede servir emplear ventanas en nuestros propios programas. Al fin y al cabo, la única función que parecen cumplir es restringir el tamaño de la pantalla, y complicarnos la vida con comandos que requieren nada menos que cuatro parámetros (*por el momento*).

Parte de la respuesta es que, en un momento dado, se puede necesitar más de una sección de la pantalla para efectuar algunos efectos «**especiales**», o dedicar cada una de esas partes a cumplir una función específica. Por ejemplo, en un programa que requiera mantener una conversación con el usuario, una ventana puede ser la de respuestas y otra la de las preguntas; esto añade claridad y belleza a cualquier aplicación, dotándola de una estructura simple y lógica.

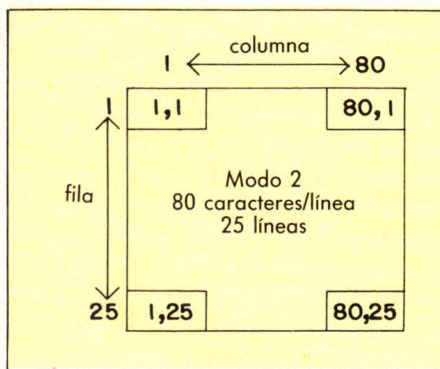


Figura 3: Pantalla de texto, Modo 2.

Otro ejemplo extremadamente claro, se encuentra en el *programa número 4*; muestra una manera sencilla de dibujar rectángulos coloreados, que pueden utilizarse, entre otras cosas, para implementar diagramas de barras en una aplicación (*sencilla*) de gráficos empresariales. Vamos a echarle una mirada: Cada vez que pulsamos una tecla, aparece en la pantalla un rectángulo de color (*línea 60*) y tamaño (*línea 50*) aleatorios. Esto es suficiente para mostrar la técnica; no sería nada complejo hacer que el color y la

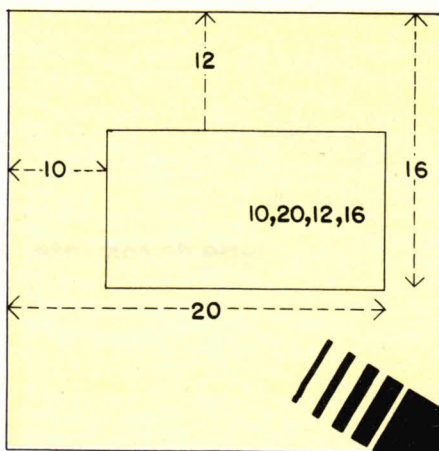


Figura 4: Definiendo una ventana.

magnitud de los bloques estuvieran predefinidos, para obtener así nuestro diagrama de barras.

La línea 20 del programa es una útil precaución que merece la pena destacar; asigna a la tecla ENTER del bloque numérico una cadena que restaura las condiciones de color y tamaño de pantalla originales. Así, cualquier «**desbarajuste**» que organicemos en la pantalla experimentando con colores y/o ventanas, puede solucionarse pulsando una sola tecla.

```

PROGRAMAS
DISCO 1A-VBNT12
10 REM PROGRAMA I
20 MODE 1
30 FOR bucle% = 1 TO 20
40 PRINT TAB(bucle%)*"*
50 NEXT bucle%

DISCO 1A-VBNT13
10 REM PROGRAMA II
20 MODE 1
30 FOR bucle% = 1 TO 20
40 LOCATE bucle%,bucle%
50 PRINT "*"
60 NEXT bucle%

DISCO 1A-VBNT14
10 REM PROGRAMA III
20 MODE 1
30 desplazamiento% = 21
40 FOR bucle% = 1 TO 20
50 LOCATE bucle%, desplazamiento% - bucl
e%
60 PRINT "*"
70 NEXT bucle%

DISCO 1A-VBNT15
10 REM PROGRAMA IV
20 KEY 139,"MODE 1: CALL &BC02: PEN 1: P
APER 0"+CHR$(13)
30 MODE 0
40 WHILE -1
50 WINDOW RND(1)*19+1,RND(1)*19+1,RND(1)
*24+1,RND(1)*24+1
60 PAPER INT(15*RND(1)+1) : CLS
70 WHILE INKEY$="": WEND
80 WEND

```

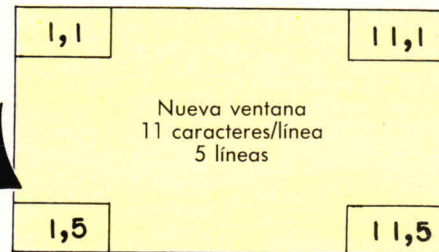


Figura 5: Coordenadas en la nueva ventana.

EN RESUMEN

— El Amstrad admite tres modos de trabajo en pantalla; cada uno de ellos la dota de diferente resolución en cuanto a texto y gráficos.

— Existen dos áreas «**superpuestas**» en la pantalla independientes una de otra: área de texto y de gráficos.

— Podemos tener simultáneamente en el monitor hasta ocho áreas, denominadas ventanas, que se comportan como micropantallas de texto, a las cuales podemos dirigir, y recibir a través de ellas de forma independiente, mensajes.

— Para activar la ventana, se emplea el comando **WINDOW** seguido de cuatro parámetros, que especifican las posiciones relativas de sus cuatro esquinas, respecto al sistema de coordenadas de la pantalla original de texto.

— Cualquier comando de escritura y/o color, si no se especifica lo contrario, queda referido a la ventana en curso, y si no se ha definido ninguna, a la que el sistema establece por defecto.

— El uso de las ventanas como herramienta de programación, puede llegar a ser imprescindible para clarificar programas con salida confusa y, para simular convincentemente **MULTITAREA**.

Esperamos que esto haya sido suficiente para abrirnos el apetito acerca de las ventanas en el Amstrad. La próxima semana examinaremos en profundidad como tratar simultáneamente las 8 ventanas, incluyendo trucos y técnicas especiales de manejo de las mismas, tales como el canje entre ellas (**WINDOW SWAP**).

MOVIMIENTO GRÁFICO

Este es el primero de una serie de comentarios en los que analizamos detalladamente pequeños programas, explicando el funcionamiento y misión de cada línea dentro del mismo. El elegido en esta ocasión utiliza el juego de caracteres del Amstrad, para colocar en pantalla un muñeco bailarín.

10-20. Dos frases REM en las que se citan el nombre del programa y se indica el principio del mismo.

30. Selecciona el MODO 1 de pantalla.

40. Limpia la pantalla.

50-70-90-110. Llama a la subrutina que comienza en la línea 150 y termina en la 190.

60-80-100-120. Estas líneas tienen la misión de visualizar el bailarín en pantalla. Los distintos números contenidos entre paréntesis, representan un tipo diferente de figura.

130. El **GOTO** devuelve la ejecu-

ción del programa a la línea ⁵⁰ 150, con lo que se produce la repetición indefinida del mismo.

140-200. Los **REM** y los asteriscos utilizados, tienen la misión de mostrar con claridad donde se encuentra la subrutina.

150-190. Constituyen la subrutina de movimiento.

150. Expresa brevemente lo que hace dicha subrutina.

160-170. EL bucle **FOR NEXT** produce un retardo en el movimiento del bailarín, de manera que seamos ca-

paces de apreciar visualmente los cambios de imagen.

180. El **LOCATE** hace que la silueta del bailarín se pinte siempre en la misma posición cada vez que llamamos a la subrutina. El efecto de movimiento se consigue debido a que las distintas imágenes del bailarín se superponen unas a otras haciendo que la figura parezca que baila.

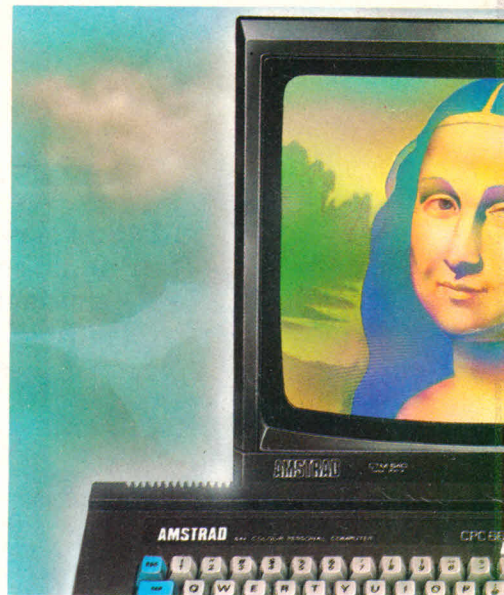
190. El **RETURN** constituye el final de la subrutina, provocando que el ordenador regrese a la línea siguiente a la que hace la llamada a dicha subrutina.



EASY DRAW

CINTA MICRO HOBBY 000-168 LA "EASYDRAW"
 DISCO LA "EASYDRAW" 20K
 FICHA 1 DISEÑADOR DE PANTALLA

Todos los usuarios de Amstrad conocen las magníficas posibilidades gráficas de esta máquina, su alta resolución y su gran número de colores. Para que podáis usar al máximo todas estas cualidades con comodidad, sin necesidad de recurrir a programaros «a mano» la rutina que os dibuje un gráfico en particular, os presentamos a Easydraw.



G. TEJA

Easydraw es un completísimo programa de utilidad para la gestión de gráficos, escrito parte en Basic y parte en lenguaje máquina, que podrás emplear para crear espectaculares dibujos, títulos o pantallas de fondo para tus programas de juegos o utilidades.

Naturalmente, podrás grabar en cinta o disco el dibujo que estés realizando en cualquier momento que desees, bien para almacenarlo con vistas a un uso futuro, bien para posteriormente seguir trabajando en él hasta que el resultado obtenido te deje satisfecho.

El dibujo así creado puedes usarlo en tus propios programas cargándolo directamente desde cinta o disco a la pantalla del ordenador, o en lugar de a la pantalla, a un lugar reservado al efecto en la memoria, donde permanecerá almacenado hasta que tu programa lo necesite.

Las pantallas que podéis ver en estas páginas, se confeccionaron en muy poco tiempo de trabajo con el programa; lógicamente vosotros tardaréis un poco más hasta familiarizaros por completo con todas las posibilidades que Easydraw ofrece. Podemos contemplar un breve re-

sumen de éstas en la tabla 1 (*Sumario de opciones*).

Las instrucciones para el manejo del programa deberían leerse «en pequeñas dosis», ya que hay 19 opciones y cada una de ellas necesita algo de práctica para comprenderla plenamente y utilizarla con eficiencia. Así que, por favor, lee la Guía de usuario para el Easydraw comenzando por la opción «DIBUJAR» e intenta dibujar líneas coloreadas verticales, horizontales y diagonales haciendo uso de las cuatro funciones que la preceden. Una vez que consideres haber alcanzado un dominio suficiente de estas cinco funciones básicas, practica con las demás hasta que comprendas su uso perfectamente y las manejes con soltura.

Para los que usen el programa por primera vez, creemos que los siguientes comentarios pueden servir de gran ayuda.

PRIMERO

Los cálculos de las posiciones de comienzo para el dibujo de figuras complejas compuestas de círculos y polígonos, al principio pueden parecer bastante complicados, por lo que una ayuda visual facilitará bastante la tarea. Sugerimos dibujar en plástico transparente un círculo de unos 100 mm de diámetro como el de la figura 1. Cuando queráis calcular la posición de comienzo de un círculo, colocarlo sobre la pantalla y podréis haceros una idea del lugar idóneo de comienzo de vuestra figura, por compleja que sea.

Como podéis observar en las figuras 2 y 3, se pueden obtener figuras muy interesantes de forma muy simple mediante este método (para una mayor claridad,

Posición de partida: Cualquier número decimal menor que 1.

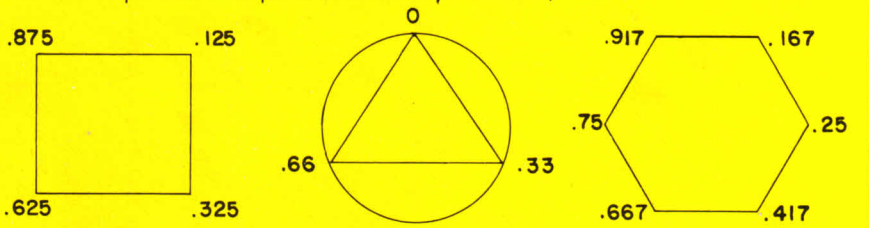


Figura 2



Figura 1

Puede usarse cualquiera de las posiciones de comienzo.

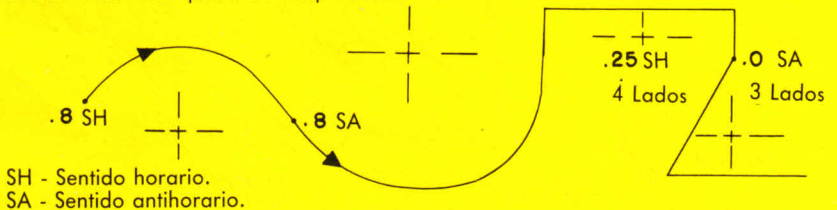


Figura 3



Serie Oro

QUINTO

Si quieres salvar en la memoria de tu Amstrad una figura, puedes hacerlo mientras la dibujas. Una manera sería preparar una parrilla de líneas en la pantalla para que te sirva de referencia (usa «DIBUJAR CON EL TABULADOR» + «MOVER TABULADOR»). Cambia el color de dibujo, «FIJA TABULADOR», mueve hasta la siguiente posición de tabula-

ción previamente fijada y dibuja una línea hasta la anterior; acto seguido, fija la siguiente posición de tabulación y repite el ciclo. De esta forma tan sencilla, la figura se «memoriza» al mismo tiempo que se dibuja. Una buena idea es asignar a la figura un tamaño aproximadamente igual a la mitad de la pantalla; recuerda que es más fácil reducir una figura grande que aumentar una pequeña.

leer la Guía de usuario, en la opción «CIRCULOS, ELIPSES Y POLIGONOS»).

SEGUNDO

Recuerda que puedes cambiar el color mientras el programa dibuja, para conseguir figuras multicolores; cuando el programa te pregunte el número de lados, elige un valor alto (*no hay límite*). Un polígono con 1.000 lados se dibujará lentamente, dándote el tiempo suficiente para cambiar colores o detener el dibujo sin problemas.

TERCERO

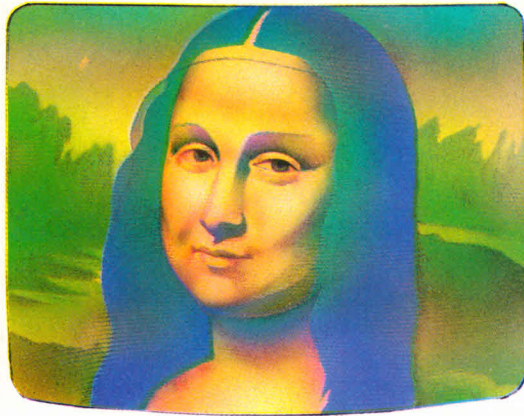
La opción «DIBUJAR CON EL TABULADOR» puede ponerse en activo cuando dibujemos círculos, con lo que obtendremos un área cónica, mientras que activar la opción «MOVIMIENTO DEL TABULADOR» antes de dibujar el círculo, y entonces «DIBUJAR CON EL TABULADOR» producirá áreas cilíndricas.

CUARTO

Si cometes un error al dibujar una figura, puedes corregirlo inmediatamente seleccionando el color de fondo, y repitiendo exactamente el mismo proceso que seguiste para crearla. Como puede ser difícil recordar los tamaños y posiciones que elegiste, un buen truco es emplear la opción «FIJAR TABULADOR», y hacerlo en uno de los lados de la figura; usa luego «DIBUJAR CON EL TABULADOR» y, a medida que muevas el cursor, la figura se irá borrando.

VARIABLES PRINCIPALES

NOMBRE	SIGNIFICADO
picture\$	Nombre del dibujo en pantalla.
movx	Desplazamiento del cursor según el modo de pantalla elegido.
beam	Dibujar o mover.
beamtrip	Pasos entre B y Copy.
x e y	Posiciones horizontal y vertical del cursor.
tes	Resultado del comando TEST.
jump, jumpy	Tamaño del desplazamiento horizontal y vertical del cursor.
xp, yp	Posiciones de tabulación.
fil	Dibujo hasta el tabulador (si/no).
tb	Definición de tabuladores.
infil	Rellenar un área con color (si/no).
movetab	Movimiento de tabuladores (si/no).
extrax, extray	Tamaño del desplazamiento del tabulador.
mem	Memoria para figuras.
memtrip	Almacenamiento en memoria (si/no).
radx, rady	Radios horizontal y vertical.
stpos	Posición de comienzo del círculo.
side	Número de lados.
delay	Retardo en el dibujo de un polígono entre punto y punto.
cw	Sentido de las agujas del reloj, o contrario a las mismas.
startat	Posición de comienzo, sobre la circunferencia o en su interior.
tba	Dibujar hasta el tabulador.
poly	Circunferencia dividida por el número de lados (polígono).
rstep	Movimiento del cursor en pixels (<i>mediante las teclas del cursor</i>).
rtes	Opción TEST.
sh, shape	Número de figuras.
mag	Reducción o ampliación de dibujos.
space, minus	Posiciones para blank\$(n).
character\$(n)	Cadena con datos necesarios para el manejo de las figuras.
blank\$(n)	Posición de «espacio» o «signo menos» en caracte.\$.



SEXTO

Para abandonar el Easydraw, pulsa «CTRL + ?», como si fueras a salvar una pantalla en la memoria, pero respondiendo «N» a la pregunta del programa. La tecla ESCAPE sólo es atendida cuando se nos solicita un INPUT, por ejemplo la opción «CIRCULO RAPIDO»; este es el momento ideal de acceder al listado para investigar posibles errores de tecleo en el mismo.

Si uno de estos errores causa la detención del programa, ANTES DE HACER OTRA COSA, tecllea GOTO 345 y salva tu dibujo en cinta o disco de la forma habitual, es decir, verás una pantalla en blanco, tecllea «CTRL + G» y volverás al principio del programa, con tu dibujo todavía en memoria. Ve contestando a las preguntas hasta que tu dibujo retorne a la pantalla; en ese momento, emplea la opción salvar pantalla (CTRL + ?), y ya tranquilo respecto a la seguridad de tu obra, averigua qué error causó la detención del programa. Esto funcionará con casi todos los errores, excepto aquéllos que afecten al propio dibujo o las rutinas en lenguaje máquina que lo manipulan, lógicamente.

SEPTIMO

En la opción «FIGURAS A MEMORIA», ten en cuenta que si pulsas las teclas necesarias para que trabaje muchas veces seguidas, esto causará que el ordenador reserve áreas en su memoria para figuras vacías, que no existen, con los consiguientes problemas de pérdida de espacio y confusión.

OCTAVO

Algunas opciones usan la misma línea para mensajes, por lo cual, si usas varias a un tiempo, recuerda desactivar una antes de acceder a las otras; de lo contra-

rio, puedes encontrarte un auténtico caos en la ventana de mensajes.

NOVENO

Antes de salvar una pantalla, elimina el cursor de la misma, ya que Easydraw la conserva íntegra en memoria o cinta/disco.

Una pantalla conservada en cinta/disco puede volcarse directamente en la pantalla, donde será visible inmediatamente.

CONCLUSIONES

Este programa, sin duda, puede calificarse de excepcional, pues no sólo posee una gran versatilidad y potencia, sino que se ha tenido un exquisito cuidado para que sea agradable y fácil de utilizar, lo cual justifica la aparentemente excesiva longitud del programa.

Tabla 1

SUBROUTINAS

NUMERO DE LINEA	FUNCION
60	Comienzo del programa, ajuste del Ramtop a un valor más bajo del normal y carga del código máquina.
320	Inicialización y definición de las ventanas de pantalla junto con los colores de las mismas.
480	Programa principal.
930	Círculo rápido.
1020	Cambio del color de la pluma.
1210	Círculos.
2040	Radios.
2140	Coloca texto en la posición del cursor.
2330	Relleno de áreas con color.
2740	Trasladar el dibujo de la pantalla a memoria.
2870	Salvar memoria en cinta o disco.
2920	Desplazamiento del cursor en pixels.
3060	Dibujo desde memoria.
3380	Salvar figuras.
3510	¿Cargar figuras?
3660	Cargar dibujo a memoria.
3780	Menú principal.
3840	Código máquina. Permite salvar y recuperar dibujos de un área reservada de memoria.

SUMARIO DE OPCIONES

COMANDO	TECLADO
Activa/desactiva modo dibujo	CTRL + B
Movimiento del cursor	TECLAS DEL CURSOR
Línea hasta el tabulador	SHIFT + TAB
Borrar pantalla	SHIFT + CLR
Círculo/Polígono	CTRL + CLR
Dibujar con/sin tabulador	CTRL + D
Rellenar superficies	CTRL + F
Menú principal	CTRL + G
Cargar figuras en memoria	CTRL + I
Desplazamiento del cursor	CTRL + J
Cargar pantalla a memoria	CTRL + L
Guardar figuras en memoria	CTRL + M
Cambio de color	CTRL + P
Círculo rápido	CTRL + Q
Recuperar figuras	CTRL + R
Salvar figuras (cinta/disco)	CTRL + S
Salvar pantalla (cinta/disco)	CTRL + ?
Fijar tabulador	TAB
Tabulador activado/desactivado	CTRL + TAB
Texto/Gráficos «CTRL»	CTRL + T

EASY DRAW.

GUIA DEL USUARIO

Elegir colores

Para cambiar el color con el que estamos dibujando, pulsar simultáneamente SHIFT más la tecla correspondiente a la letra del color que desees. Dicho color aparece al lado de la palabra SHIFT, en pantalla.

Cambio de colores

Pulsar CTRL + P seguido por la apropiada letra correspondiente al color elegido. Este se volverá negro, y en este momento pulsando SHIFT podrás moverte a través de la paleta de colores; cuando pulses ENTER el color quedará fijo en el dibujo de pantalla.

Control del cursor

Se realiza mediante las cuatro teclas que rodean a la tecla COPY; cada una corresponde a la dirección que indica la flecha. Junto con SHIFT, harán que el cursor se desplace más rápido, de 20 en 20 pixels.

Desplazamiento del cursor

CTRL + J modificará el desplazamiento del cursor, indicándole al programa la magnitud horizontal y vertical del mismo.

Dibujar

CTRL + B activa/desactiva el dibujo, es decir, si nos encontramos en modo dibujo, CTRL + B nos permitirá movernos por la pantalla mediante las teclas del cursor, pero sin dibujar; obviamente, si nos encontramos en modo «mover», CTRL + B pasará a modo «dibujo». La opción CTRL + J también puede ser utilizada junto con ésta. Alternativamente, CTRL + COPY habilita a esta tecla para dibujar, pero sólo cuando está oprimida. Con este último uso, la opción «DESPLAZAMIENTO DEL CURSOR» no sirve para dibujar; sólo para movernos por la pantalla.

Fijar tabulador

La tecla TAB «fija» la posición del cursor, la cual se memoriza y muestra en la parte inferior de la pantalla, en amarillo y encima de las coordenadas del cursor. Pueden trazarse líneas hasta este punto desde cualquier parte de la pantalla.

Dibujar línea

SHIFT + TAB trazará una línea desde la posición actual del cursor hasta la que fue

memorizada mediante la opción anterior.

Dibujar con el tabulador

CTRL + D activa/desactiva esta opción, que permite dibujar líneas hasta la posición memorizada de tabulación, según el cursor se va moviendo por la pantalla; es útil para no tener que andar pulsando SHIFT + TAB como en la opción anterior, si queremos trazar gran cantidad de líneas.

Rellenar una superficie

Coloca el cursor dentro del área que quieras rellenar, pulsa CTRL + F y se coloreará una línea de pixels con la tinta en uso. Moviendo el cursor arriba y abajo con las teclas correspondientes, llenarás el área poco a poco. Al pasar a otra superficie se debe repetir la operación CTRL + F.

Borrado

Seleccionando como color de dibujo el color de fondo, con las opciones «DIBUJAR», «DIBUJAR HASTA EL TABULADOR» o «RELLENAR», se borrarán los colores deseados. Para un completo borrado de pantalla, usar SHIFT + CLR.

Círculo rápido

Mediante las teclas de control del cursor, desplázate por la pantalla hasta el lugar de la misma donde se desee colocar el centro del círculo.

Pulsa CTRL + Q y, o bien dale al programa un número (el radio del círculo), o bien pulsa Ø. Esto último te permitirá usar las teclas del cursor derecha e izquierda hasta alcanzar el radio que desees; luego pulsa ENTER y el círculo aparecerá en pantalla.

5. Círculos, elipses y polígonos

Usa las teclas del cursor para definir el centro de la figura y pulsa CTRL + CLR; el programa te pedirá las dimensiones horizontal y vertical de la figura; puedes suministrarlas bien directamente mediante un número, bien empleando las teclas del cursor para marcarlas.

A continuación, el programa te pedirá la posición de comienzo, un número entre 0 y 1.

Esto se calcula usando fracciones decimales en un círculo, en el cual se cuen-

Serie Oro

ta en el sentido de las agujas del reloj (figura 1).

También se debe aclarar si se desea que el dibujo de nuestra figura comience en la posición del cursor elegida al principio, o bien queremos que dicha posición se considere el centro del dibujo.

Responde 0 ó 1 a la pregunta de si quieres que el dibujo se haga en el sentido de las agujas del reloj, o en el contrario.

Por último, el número de lados. Si es menor que 20, se nos pedirá la cuantía del retardo en el dibujo entre punto y punto, el cual sólo es necesario si por alguna razón queremos detenerlo pulsando la barra espaciadora.

Movimiento del tabulador

La posición del cursor memorizada mediante «FIJAR TABULADOR» puede moverse con el cursor pulsando CTRL + TAB. Con esto y la opción SHIFT + TAB pueden dibujarse formas paralelas. También funcionará en la opción anterior, pero debe usarse antes CTRL + CLR.

Figuras a memoria

CTRL + M almacena 40 figuras en memoria de cualquier tamaño, que pueden ser «llamadas» en cualquier momento. Pulsa TAB, y mientras dibujas una línea continua, pulsa TAB de nuevo cada vez que cambies de dirección.

CTRL + S salva una figura en cinta o disco.

CTRL + I cargará una nueva figura desde cinta o disco.

Volver a dibujar figuras

CTRL + R y el número y tamaño de figura para aumentar o reducir el original.

Números entre 0 y 1 reducirán la figura de tamaño. Valores mayores que 1 la aumentarán.

Texto

Pulsa CTRL + T para colocar texto, números o gráficos.

«CTRL» en la posición del cursor.

Menú principal

Pulsa CTRL + G.

Salvar pantalla

CTRL + ? salva la pantalla en curso a cinta o disco.

Cargar pantalla

Usa la opción al efecto al principio del programa, o CTRL + L en cualquier momento ya una vez dentro del mismo.

```

10 REM *****
20 REM * EASYDRAW *
30 REM *
40 REM * By Glynne Davies *
50 REM *****
60 MEMORY &645A : REM * reserva memoria
para el volcado de pantalla y disco *
70 GOSUB 3850: REM * carga el c/m para
el volcado de pantalla *
80 REM ** Presentacion **
90 MODE 0:FOR n=PI TO 3 STEP PI/16:pt=
(PI+1) MOD 7:pc=PI+1
100 TAG: PLOT 00+(20*SIN(n)),200+(120*CO
S(n)),pc
110 PRINT "--EASYDRAW--":NEXT:TAGOFF
120 FOR n=1 TO 600:NEXT
130 CALL &BC02
140 MODE 2:INK 15,1:INK 0,13:INK 1,0:x=2
00:y=150:beam=0:itestab=3
150 jump=20:jumy=20:DIM blank$(100):DIM
characters$(40)
160 beam=0:LOCATE 17,12:INPUT "Deseas ca
rgar un dibujo en Memoria S/N?;pic$:pic$
=UPPER$(pic$): IF pic$(0)="" AND pic$(0)=""
THEN GOTO 160
170 LOCATE 17,14:INPUT "Disco o Cassette
D/C?;sw$:sw$=UPPER$(sw$):IF sw$="D" THE
N w=:DISC:ELSE IF sw$="C" THEN w=:TA
PE
180 IF sw$("<D") AND sw$("<C") THEN GOTO 1
70
190 IF pic$="" THEN GOTO 220
200 LOCATE 17,16:INPUT "Con que nombre?;
picture$:picture$=UPPER$(picture$):IF pi
cture$="" AND w=: THEN GOTO 200
210 LOAD picture$,&6475
220 CLS:LOCATE 20,12:INPUT "En que modo
deseas trabajar (ie. 0 or 1)?;modescree
n:IF modescree n > 1 OR modescree n < 0 TH
EN GOTO 220
230 IF modescree n = 1 THEN movx= 2:ELSE m
ovx=4
240 IF modescree n > 1 THEN GOTO 220
250 MODE modescree n:INK 1,24
260 LOCATE(2*(modescree n)+1),12: INPUT
"Conservar dibujo en pantalla S/N?;Y$
270 CLS
280 IF Y$="S" OR Y$="s" THEN CALL 26215
290 GOSUB 330 : REM * inicializacion *
300 CLS
310 END
320 REM ** inicializa la pantalla **
330 WINDOW #2,1,2,1,22:WINDOW #3,1,00/mo
vx,23,25:WINDOW #0,1,00/movx,23,25:WINDO
W #4,3,00/movx,1,22
340 PAPER #2,0:PAPER #3,15:CLS #2
345 CLS #3
350 ORIGIN 64,48,64,64,48,400
360 DRAW 574,1,3:DRAW 574,350,3:DRAW 1,3
50,3:DRAW 1,1,3
370 PEN #2,0:LOCATE #2,2,20:PRINT #2,CHR
$(143):
380 REM ** Dibuja colores **
390 FOR count=0 TO 15
400 PEN #2,count
410 LOCATE #2,1,count+1:PRINT #2,CHR$(65
+count):CHR$(233)
420 NEXT count
430 PEN #2,4:LOCATE #2,1,1:PRINT #2,"A"
440 PEN #2,5:LOCATE #2,1,17:PRINT #2,"*"
450 IF movx=2 THEN LOCATE #2,1,1:PRINT #
2,"A":LOCATE #2,1,5:PRINT #2,"E":LOCATE
#2,1,9:PRINT #2,"I":LOCATE #2,1,13:PRINT
#2,"M"
460 shift$="SHIFT":FOR count=1 TO 5:LOCA
TE #2,1,17+count:PRINT #2,MID$(shift$,c
ount,1):NEXT
470 PEN #3,1:LOCATE #3,13,3:PRINT #3,"Mo
ver ":REM marca pixel sin pintar
480 REM ** programa principal **
490 ON ERROR GOTO 490
500 WHILE exit < 1
510 IF INKEY (1)=0 THEN PLOT x,y,tes:x=x
+movx:IF movtab=1 THEN extrax=extrax+mov
x
520 IF INKEY (0)=0 THEN PLOT x,y,tes:x=x
-movx:IF movtab=1 THEN extrax=extrax-mov
x
530 IF INKEY (0)=0 THEN PLOT x,y,tes:y=y
+2:GOSUB 2340:IF movtab=1 THEN extray=ex
tray+2:REM rutina de fill
540 IF INKEY (2)=0 THEN PLOT x,y,tes:y=y
-2:GOSUB 2340:IF movtab=1 THEN extray=ex
tray-2
550 IF INKEY (1)=32 THEN PLOT x,y,tes:x=x
+jump:IF movtab=1 THEN extrax=extrax+ju
mp
560 IF INKEY (0)=32 THEN PLOT x,y,tes:x=x
-jump:IF movtab=1 THEN extrax=extrax-ju
mp
570 IF INKEY (0)=32 THEN PLOT x,y,tes:y=y
+jumy:IF movtab=1 THEN extray=extray+j
umy
580 IF INKEY (2)=32 THEN PLOT x,y,tes:y=y
-jumy:IF movtab=1 THEN extray=extray-j
umy
590 IF mem=1 AND xp=XPOS AND yp=YPOS AND
memtrip=1 THEN character$(sh)=character
$(sh)+STR$(xrel)+STR$(yrel):memtrip=0:IF
LEN(character$(sh)) > 240 THEN SOUND 1,3
00,25: PEN #3,1:LOCATE #3,1,1:PRINT #3,"
Es el limite "mem=0
600 IF INKEY(50)=128 THEN SOUND 1,100,20
:GOSUB 3660:REM Dibuja desde memoria las
"figuras
610 IF INKEY(36)=128 THEN SOUND 1,100,15
:GOSUB 3660: REM carga el dibujo en la m
emoria
620 IF INKEY(16)=32 THEN CLS #4:MOVE 0,0
:DRAW 574,1,3:DRAW 574,350,3:DRAW 1,350,
3:DRAW 1,1,3:MOVE x,y:REM limpia la pant
alla
630 IF INKEY (9)=128 THEN SOUND 1,120,10
:beamtrip=(beamtrip+1) MOD 2:IF beamtrip
=0 THEN PEN #3,1:LOCATE #3,13,3:PRINT #3
,"Mover ":REM enciende/apaga
640 IF beamtrip=0 THEN GOTO 660
650 IF INKEY (9)=0 THEN beam=1:PEN #3,10
:LOCATE #3,13,3:PRINT #3,"Dibujar ":ELS
E beam=0:PEN #3,10:LOCATE #3,13,3:PRINT
#3,"COPY Mov":REM enciende /apaga
660 IF beamtrip=1 THEN GOTO 680
670 IF INKEY (54)=128 THEN SOUND 1,100,1
0:beam=(beam+1) MOD 2:IF beam=1 THEN PEN
#3,1:LOCATE #3,13,3:PRINT #3,"Dibujar "
:ELSE beam=0:PEN #3,1:LOCATE #3,13,3:PR
INT #3,"Mover ":REM enciende / apaga
680 IF INKEY(60)=128 THEN SOUND 1,100,20
:movtab=(movtab+1) MOD 2:IF movtab=1 THE
N PEN #3,1:LOCATE #3,1,2:PRINT #3,"Tabul
ador activado ":ELSE PEN #3,1: LOCATE #
3,1,2:PRINT #3,"Tabulador desactiv. ":xp
=x+extrax:yp=y+extray:extrax=0:extray=
0:REM tabular
690 IF INKEY(35)=128 THEN SOUND 1,100,10
:GOSUB 3510 : REM cargar figuras desde c
inta/disco
700 IF INKEY (21)=32 THEN GOSUB 1030: RE
M color de pluma
710 IF INKEY (45)=128 THEN SOUND 1,100,2
5:GOSUB 2920: REM salto de cursor
720 IF INKEY (68)=0 THEN SOUND 1,100,5:x
rel=x-xp:yrel=y-yp:xo=x:yo=y:PEN #3,1:LO
CATE #3,1,2:PRINT #3,"X";XPOS;"Y";YPOS;"
Posn TAB":tb=1:memtrip=1:REM definiendo
tabulador
730 IF sh) 39 THEN GOTO 750
740 IF INKEY (38)=128 THEN SOUND 1,100,5
:mem=(mem+1) MOD 2:IF mem=1 THEN PEN #3,
1:LOCATE #3,1,1:PRINT #3,"Figuras Archiv
adas " :sh:SPACE$(15):ELSE PEN #3,1:LO
CATE #3,1,1:PRINT #3,"Figura archivada
":SPACE$(20):sh=sh+1
750 IF INKEY (61)=128 THEN SOUND 1,100,5
:fil=(fil+1) MOD 2:IF fil=1 THEN LOCATE
#3,1,1:PRINT #3,"Dibuja con tabulador":E
LSE LOCATE #3,1,1:PRINT #3,"Dibuja sin t
abulador": REM dibujo segun tabulador on
/off
760 IF INKEY(52)=128 THEN SOUND 1,200,25
:GOSUB 3700:REM regresa al menu de opcio
nes
770 IF INKEY (60)=128 THEN SOUND 1,100,5
: GOSUB 3300 : REM salvar en cinta / dis
co
780 IF INKEY (67)=128 THEN SOUND 1,100,2
5:PLOT x,y,14:GOSUB 940 :REM dibuja circ
ulo rapidamente
790 IF INKEY (60)=32 THEN DRAW xp+extra
x-XPOS,y+extray-YPOS,p :PLOT x,y,14
800 IF INKEY (16)=128 THEN SOUND 1,100,1
0:PLOT x,y,14: GOSUB 1220 : REM Circulos
, poligonos
810 IF INKEY (53)=128 THEN SOUND 1,100,5
:infil=(infil+1) MOD 2:IF infil=1 THEN f
il=TEST(x-2,y-2):LOCATE #3,1,1:PRINT #3
,"Rellenar supericies";SPACE$(20):ELSE
LOCATE #3,1,1:PRINT #3,"Modo normal
":SPACE$(20):
820 IF INKEY(27)=128 THEN SOUND 1,100,5:
GOSUB 2540 :REM * cambio color tinta *
830 IF INKEY (38)=128 THEN SOUND 1,100,5
:MOVE 0,0:DRAW 574,1,3:DRAW 574,350,3:DR
AW 1,350,3:DRAW 1,1,3:GOTO 2750 : REM *
saiva dibujo *
840 IF INKEY(51)=128 THEN SOUND 1,100,5:
GOSUB 2160 : REM * posicionamiento del c
ursor *
850 IF x=xx AND y=yy THEN GOTO 920
860 IF beam=1 THEN DRAW x,y,p:tes=p
870 IF beam=0 THEN tes=TEST(x,y):xx=x:yy
=y:PLOT x,y,14
880 IF tb=1 AND fil = 1 THEN DRAW xp+ex
trax-XPOS,y+extray-YPOS,p
890 PEN #3,4:LOCATE #3,1,3:PRINT #3,"X";
x:"Y";y
900 IF xp=xpt AND yp=ypt AND extrax=xpt
na AND extray=yextra THEN GOTO 920
910 PEN #3,1:LOCATE #3,1,2:PRINT #3,"X";
xp+extrax;"Y";yp+extray:" " :xpt=x
p:yp=y:yp+extrax=extrax:yextra=extray
920 WEND
930 REM ** circulo rapido **
940 CALL &BB18:IF INKEY(67)=128 THEN GOT
O 940:CALL &BB18
950 oc=1:PEN #3,1:LOCATE #3,1,1:INPUT #3
,"DEF RADIO cursor (0):DEF RADIO valor";r
adx$:IF radx$="" OR radx$="" THEN GOTO
950
960 IF ASC(radx$) < 48 THEN GOTO 950
970 rady=VAL(radx$):LOCATE #3,1,1:PRINT
#3,SPACE$(40):IF radx=0 THEN GOSUB 194
0
980 radx=radx+(radx MOD movx) :MOVE x+ra
dx,y
990 FOR count=0 TO 360 STEP 2:DEG:DRAW x
+(radx*COS(count)),y+(radx*SIN(count)),p
:NEXT
1000 LOCATE #3,1,1:PRINT #3,SPACE$(40)
1010 RAD:oc=0:RETURN
1020 REM ** cambio color pluma **
1030 IF INKEY (69)=32 THEN SOUND 1,150,5
:0=0 :REM * A *
1040 IF INKEY (54)=32 THEN SOUND 1,150,5
:0=1 :REM * B *
1050 IF INKEY (62)=32 THEN SOUND 1,150,5
:0=2 :REM * C *
1060 IF INKEY (61)=32 THEN SOUND 1,150,5
:0=3 :REM * D *
1070 IF INKEY (58)=32 THEN SOUND 1,150,5
:0=4 :REM * E *
1080 IF INKEY (53)=32 THEN SOUND 1,150,5
:0=5 :REM * F *
1090 IF INKEY (52)=32 THEN SOUND 1,150,5
:0=6 :REM * G *
1100 IF INKEY (44)=32 THEN SOUND 1,150,5
:0=7 :REM * H *
1110 IF INKEY (35)=32 THEN SOUND 1,150,5
:0=8 :REM * I *
1120 IF INKEY (45)=32 THEN SOUND 1,150,5
:0=9 :REM * J *
1130 IF INKEY (37)=32 THEN SOUND 1,150,5
:0=10:REM * K *
1140 IF INKEY (36)=32 THEN SOUND 1,150,5
:0=11:REM * L *
1150 IF INKEY (38)=32 THEN SOUND 1,150,5
:0=12:REM * M *
1160 IF INKEY (46)=32 THEN SOUND 1,150,5
:0=13:REM * N *
1170 IF INKEY (34)=32 THEN SOUND 1,150,5
:0=14:REM * O *
1180 IF INKEY (27)=32 THEN SOUND 1,150,5
:0=15:REM * P *
1190 PEN #2,p:LOCATE #2,2,20:PRINT #2,CH
R$(233):
1200 RETURN
1210 REM ** circulos **
1220 CALL &BB18:IF INKEY(16)=128 THEN GO
TO 1220:CALL &BB18
1230 CLS #3:LOCATE #3,1,1:INPUT #3,"Def.
Eje cursor (0) Def. Eje num";radx$:IF
radx$="" OR radx$="" THEN GOTO 1230
1240 IF ASC(radx$) < 48 THEN GOTO 1230
1250 rady=VAL(radx$)
1260 LOCATE #3,1,1:PRINT #3,SPACE$(40):
IF VAL(radx$) < 4 THEN GOSUB 1940:GOTO 1
310
1270 CLS #3
1280 LOCATE #3,1,1:INPUT #3,"EJE Vertica
l";2:rady$:IF radx$="" OR radx$="" THE
N GOTO 1280
1290 IF ASC(radx$) < 48 THEN GOTO 1290
1300 rady=VAL(radx$):IF rady < 2 THEN GOS
UB 1030
1310 CLS #3
1320 CLS #3:LOCATE #3,1,1:INPUT #3,"posi

```

Estas pantallas han sido realizadas con el Easy Draw.



```

cion de partida 0-1 (pe. 0.65)*stpos:1
F stpos=" " OR stpos="*" THEN GOTO 1320
1330 IF ASC(stpos) < 46 THEN GOTO 1320
1340 stpos=VAL(stpos):IF VAL(stpos) > 1
THEN CLS #3:LOCATE #3,1,1:PRINT #3,"Ent
ne 0 y 1 pe .25":FOR n=1 TO 3000:NEXT:GO
TO 1310

```

```

1350 CLS #3:LOCATE #3,1,1:INPUT #3,"Nume
ro de lados (pe. 70 circulos)":side$:IF
side$=" " OR side$="*" THEN GOTO 1350

```

```

1360 IF ASC(side) < 48 THEN GOTO 1350
1370 side=VAL(side):IF VAL(side) < 3 T
HEN GOTO 1350

```

```

1380 pol=(2*PI)/side
1390 LOCATE #3,1,1:PRINT #3,SPACE$(40)
1400 IF side > 19 THEN GOTO 1430 :REM re
tardo

```

```

1410 IF side < 20 THEN CLS #3:LOCATE #3,
1,1:INPUT #3,"Retardo de dibujo
0-200 ";delay$:IF delay$=" " OR delay$
="*" THEN GOTO 1410

```

```

1420 IF side < 20 AND ASC(delay) < 48 T
HEN GOTO 1410 : ELSE delay=VAL(delay)
1430 CLS #3:LOCATE #3,1,1:INPUT #3,"sent
ido horario : se. antihorario 0";cw$:I
F cw$=" " OR cw$="*" THEN GOTO 1430

```

```

1440 IF ASC(cw) < 48 THEN 1430
1450 cw=VAL(cw):IF VAL(cw) > 1 THEN GOTO
1430

```

```

1460 CLS #3
1470 LOCATE #3,1,1:INPUT #3,"Trazar des
de cursor/centro 1/0":startat$:IF startat
$=" " OR startat$="*" THEN GOTO 1460

```

```

1480 IF ASC(startat) < 48 THEN GOTO 1470
1490 startat=VAL(startat):IF VAL(startat)
> 1 THEN GOTO 1470

```

```

1500 IF tb=0 THEN GOTO 1540
1510 LOCATE #3,1,1:INPUT #3,"Con tabulad
or activado 1 o 0 apagado (30)":tba$:IF
tba$=" " OR tba$="*" THEN GOTO 1510

```

```

1520 IF ASC(tba) < 48 THEN GOTO 1510
1530 tba=VAL(tba):IF VAL(tba) > 1 THEN
GOTO 1510

```

```

1540 LOCATE #3,1,1:PRINT #3,SPACE$(40)
1550 LOCATE #3,1,1:PRINT #3,"ESPACIA para
parar"

```

```

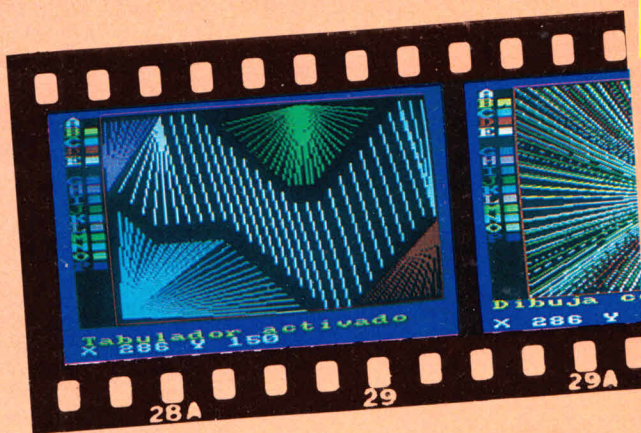
1560 radx=radx+(radx MOD movx)
1570 IF cw=1 THEN GOTO 1770: REM sentido
contrario a las agujas del reloj

```

```

1580 IF startat=0 THEN PLOT x,y,tes

```



```

1590 IF startat=1 THEN x=(radx*SIN(st
pos*2*PI)):y=(radx*COS(stpos*2*PI)):MO
VE x,y
1600 MOVE x+(radx*SIN(stpos*2*PI)),y+(ra
dx*COS(stpos*2*PI))
1610 REM ** circulo en el sentido de las
agujas del reloj **
1620 FOR count=stpos*2*PI TO (4*PI)+pol
Y STEP pol
1630 DRAW x+(radx*SIN(count)),y+(radx*CO
S(count)),p:IF movtab=1 THEN extrax=(XP
S-x)-radx*SIN(stpos*2*PI):extray=(YPOS-y
)-radx*COS(stpos*2*PI)
1640 IF INKEY (60)=32 THEN DRAW xp+extr
ax-XPOS,yp+extray-YPOS,p:MOVE x+(radx*SI
N(count)),y+(radx*COS(count))
1650 IF INKEY (21)=32 THEN GOSUB 1030: R
EM color de la pluma
1660 IF tba=1 THEN DRAW xp+extrax-XPOS,
yp+extray-YPOS,p:MOVE x+(radx*SIN(count)
),y+(radx*COS(count))
1670 IF INKEY (47)=0 THEN count=(4*PI)+p
oly
1680 IF xyoff=1 THEN GOTO 1700
1690 LOCATE #3,1,3:PRINT #3,"X";XPOS;"Y"
;YPOS
1700 FOR rest=0 TO delay:IF INKEY (47)=0
THEN count=(4*PI)+poly:ELSE NEXT
1710 NEXT
1720 LOCATE #3,1,1:PRINT #3,SPACE$(60):x
=XPOS;y=YPOS
1730 LOCATE #3,1,3:PEN #3,4:PRINT #3,"X"

```

```

;XPOS;"Y";YPOS:IF beam=1 THEN PEN #3,1:L
OCATE #3,13,3:PRINT #3,"Dibujar";ELSE b
eam=0:PEN #3,1:LOCATE #3,13,3:PRINT #3,"
Mover ";REM encendido / apagado
1740 extrax=CINT(extrax):extray=CINT(extr
ay)
1750 RETURN
1760 REM ** circulo en sentido contrario
a las agujas del reloj **
1770 stpos=stpos+0.5:IF startat=0 THEN P
LOT x,y,tes
1780 IF startat=1 THEN x=(radx*SIN(st
pos*2*PI)):y=(radx*COS(stpos*2*PI))
1790 MOVE x-(radx*SIN(stpos*2*PI)),y-(ra
dx*COS(stpos*2*PI))
1800 FOR count=2*PI*stpos TO -(2*PI)+po
ly STEP -poly
1810 DRAW x-(radx*SIN(count)),y-(radx*CO
S(count)),p :IF movtab=1 THEN extrax=(XP
OS-x)+radx*SIN(2*PI*stpos):extray=(YPOS-
y)+radx*COS(2*PI*stpos)
1820 IF INKEY (60)=32 THEN DRAW xp+extr
ax-XPOS,yp+extray-YPOS,p:MOVE x-(radx*SI
N(count)),y-(radx*COS(count))
1830 IF tba=1 THEN DRAW xp+extrax-XPOS,
yp+extray-YPOS,p:MOVE x-(radx*SIN(count)
),y-(radx*COS(count))
1840 IF INKEY (47)=0 THEN count=-(2*PI
)+poly
1850 LOCATE #3,1,3:PRINT #3,"X";XPOS;"Y"
;YPOS
1860 IF INKEY (21)=32 THEN GOSUB 1030: R

```

```

EM color de la pluma
1870 FOR rest=0 TO delay:IF INKEY (47)=0
THEN count=-(2*PI)+poly:ELSE NEXT
1880 NEXT
1890 LOCATE #3,1,1:PRINT #3,SPACE$(60):x
=XPOS;y=YPOS:count=0
1900 LOCATE #3,1,3:PEN #3,4:PRINT #3,"X"
;XPOS;"Y";YPOS:IF beam=1 THEN PEN #3,1:L
OCATE #3,13,3:PRINT #3,"Dibujar ";ELSE
beam=0:PEN #3,1:LOCATE #3,13,3:PRINT #3,
"Mover ";REM encendido / apagado
1910 extrax=CINT(extrax):extray=CINT(extr
ay)
1920 RETURN
1930 REM ** definir radio horizontal med
iante el cursor **
1940 WHILE radx =0
1950 LOCATE #3,1,1:PRINT #3,"Mover curso
r (-) Enter fijar posicion";
1960 IF INKEY(1)=0 THEN PLOT x+rstep,y,r
tes:rtes=TEST(x+rstep-movx,y):rstep=rste
p+movx:PLOT x+rstep,y,14
1970 IF INKEY (8)=0 THEN PLOT x+rstep,y,
rtes:rtes=TEST(x+rstep-movx,y):rstep=rst
ep-movx:PLOT x+rstep,y,14
1980 IF INKEY (18)=0 THEN radx=ABS(rstep
):PLOT x+rstep,y,rtes
1990 PLOT x,y,14
2000 WEND:rtes=14:rstep=0:CALL &BB18
2010 IF oc=1 THEN RETURN
2020 LOCATE #3,1,3:PRINT #3,SPACE$(20):L
OCATE #3,1,3:PRINT #3,"Eje horz.":radx
2030 rady=0
2040 REM ** radio inferior/superior defi
nido por el cursor **
2050 WHILE rady=0
2060 LOCATE #3,1,1:PRINT #3,"Mover curso
r Eje verEnter fija posicion"
2070 IF INKEY(0)=0 THEN PLOT x,y+rstep,r
tes:rtes=TEST(x,y+rstep+2):rstep=rstep+2
:PLOT x,y+rstep,14
2080 IF INKEY(2)=0 THEN PLOT x,y+rstep,r
tes:rtes=TEST(x,y+rstep-2):rstep=rstep-2
:PLOT x,y+rstep,14
2090 IF rstep=0 AND INKEY(63)=0 THEN rad
y=radx
2100 IF INKEY(10)=0 THEN rady=ABS(rstep
):PLOT x,y+rstep,rtes
2110 PLOT x,y,14
2120 WEND:rstep=0:rtes=0:CALL &BB00
2130 CALL &BB18:RETURN

```

MICROHOBBY AMSTRAD SEMANAL

LE OFRECE AHORA SUS PROGRAMAS YA GRABADOS, PARA QUE VD. NO TENGA QUE TECLEARLOS

Todos los programadores y aficionados a la microinformática sabemos lo tedioso y penoso a errores que resulta el teclear un listado de un programa. Para facilitar tu labor al máximo y que no tengas que estar horas sobre el teclado de tu ordenador tratando de descifrar incomprensibles mensajes de error, **AMSTRAD SEMANAL** te ofrece cada mes los programas publicados de los cuatro números correspondientes en una cinta de cassette, sólo por 675 ptas. (sin más gastos por envío).

Envíanos con la menor demora posible, el cupón correspondiente.



Para que tus dedos no realicen el trabajo duro, M.H. AMSTRAD lo hace por ti. Todos los listados que incluyen este logotipo se encuentran a tu disposición en un cassette mensual, solicitanoslo.

```

2140 REM ** posiciona cursor **
2150 CALL &BB1B
2160 LOCATE #3,1,1:PRINT #3," Modo
Texto "
2180 LOCATE #3,1,2:PRINT #3," (Enter) p
ara fin "
2190 FOR n=1 TO 500:NEXT:CALL &BB00
2200 WHILE t<1
2210 type%=INKEY%
2220 IF INKEY(18)=0 THEN PLOT x,y,0: t=1
:padr=0:GOTO 2290
2230 TAG:PLOT x,y,p
2240 IF type%="" GOTO 2280
2250 IF ASC(type%)=127 THEN PLOT x,y,0:
x=(movx*4):MOVE x,y:PRINT " ":GOTO 22
80
2260 PRINT type%;
2270 x=(movx*8):MOVE x,y
2280 type%=""
2290 WEND
2300 t=0:LOCATE #3,1,1:PRINT #3,SPACE$(4
0)
2310 TAGOFF
2320 RETURN
2330 REM ** rutina de relleno **
2340 IF infil = 0 THEN RETURN
2350 WHILE lin < 1
2360 IF TEST (x,y) (<) filc THEN lin =1:G
OTO 2420
2370 incl=incl+movx:telf=TEST (x-incl,y)
2380 IF telf (<) filc THEN MOVE x,y:DRAW
x-(incl-movx),y,p:lin =1
2390 IF x<2 THEN lin=1
2400 IF y<2 THEN lin=1
2410 IF y>340 THEN lin=1
2420 WEND
2430 lin=0
2440 WHILE lin < 1
2450 incr=incr+movx:terf=TEST (x+incr,y)
2460 IF x>575 THEN lin=1
2470 IF y<2 THEN lin=1
2480 IF y>340 THEN lin=1
2490 IF terf (<) filc THEN MOVE x,y:DRAW
x+(incr-movx),y,p:lin =1
2500 WEND
2510 lin=0:incl=0:incr=0:telf=17:terf=17
2520 RETURN
2530 REM ** cambio de tinta **
2540 CALL &BB1B:IF INKEY(27)=128 THEN 25
40:CALL &BB1B
2550 CLS #3:LOCATE #3,1,1:INPUT #3,"Camb
io de color S/N"; y$:IF y$="S" OR y$="s"
THEN GOTO 2560:ELSE LOCATE #3,1,1:PRINT
#3,SPACE$(40):RETURN
2560 LOCATE #3,1,1:INPUT #3,"Color a sus
tituir ";in$:IF in$="" THEN GOTO 2560
2570 IF ASC(in$) (< 65 OR ASC(in$) >112
THEN LOCATE #3,1,1:PRINT #3,SPACE$(40):G
OTO 2540

```

```

2570 IF ASC(in$) (< 65 OR ASC(in$) >112 T
HEN LOCATE #3,1,1:PRINT #3,SPACE$(40):G
OTO 2540
2580 in$=UPPER$(in$):IF ASC(in$) > 80 TH
EN GOTO 2540
2590 p=ASC (in$)-65
2600 IF p>15 THEN GOTO 2540
2610 CLS #3
2620 WHILE inchange < 1
2630 LOCATE #3,1,1: PRINT #3,"Cambio col
or (Shift)"
2640 PEN #3,p
2650 LOCATE #3,1,2: PRINT #3,"Fijar col
or (Enter)"
2660 LOCATE #3,1,3: PRINT #3," TINTA";
p1,";";i1
2670 IF INKEY(21)=32 THEN i=(i+1) MOD 27
2680 IF INKEY(18)=0 OR INKEY (6)=0 THEN
inchange=1
2690 INK p,i
2700 WEND
2710 i=0:inchange=0
2720 CLS #3:PEN #3,4:LOCATE #3,1,3:PRINT
#3,"X":XPOS;"Y":YPOS
2730 RETURN
2740 REM ** salva la pantalla en memoria
**
2750 FOR n=1 TO 100:NEXT:CALL &BB00
2760 PAPER #2,15:CLS #2:CLS #3
2770 CALL 26203
2780 CLS #4
2790 FOR n=1 TO 2000:NEXT n
2800 CALL 26215
2810 FOR n=1 TO 2000:NEXT
2820 CALL &BB00
2830 LOCATE 2,23:INPUT"Salvar pantalla S
/N":ipt$
2840 IF ipt$="s" OR ipt$="S" THEN GOSUB
2800
2850 MODE 1:y=y+2:LOCATE 12,12:INPUT "FI
N DEL PROGRAMA S/N";y$:IF y$="s" OR y$="
S" THEN CALL &BC02:MODE 1:LOCATE 6,2:PRI
NT "Damos por finalizado el programa":EN
D
2860 MODE 2:GOTO 160
2870 REM ** Salva en cinta **
2880 SPEED WRITE 1:LOCATE 1,23:INPUT "Di
me el nombre ";name$
2890 IF w=1 AND name$="" THEN GOTO 2880
2900 SAVE name$.B.&6675.&4000
2910 RETURN
2920 REM ** Desplazamiento del cursor en
pixeis **
2930 CALL &BB1B:IF INKEY(45)=128 THEN GO
TO 2930:CALL &BB1B
2940 CALL &BB00
2950 CLS #3:PEN #3,1:LOCATE #3,1,1:INPUT
#3," Desp. Horizontal. ";jump$:IF jump$="
" OR jump$="" THEN GOTO 2950

```

```

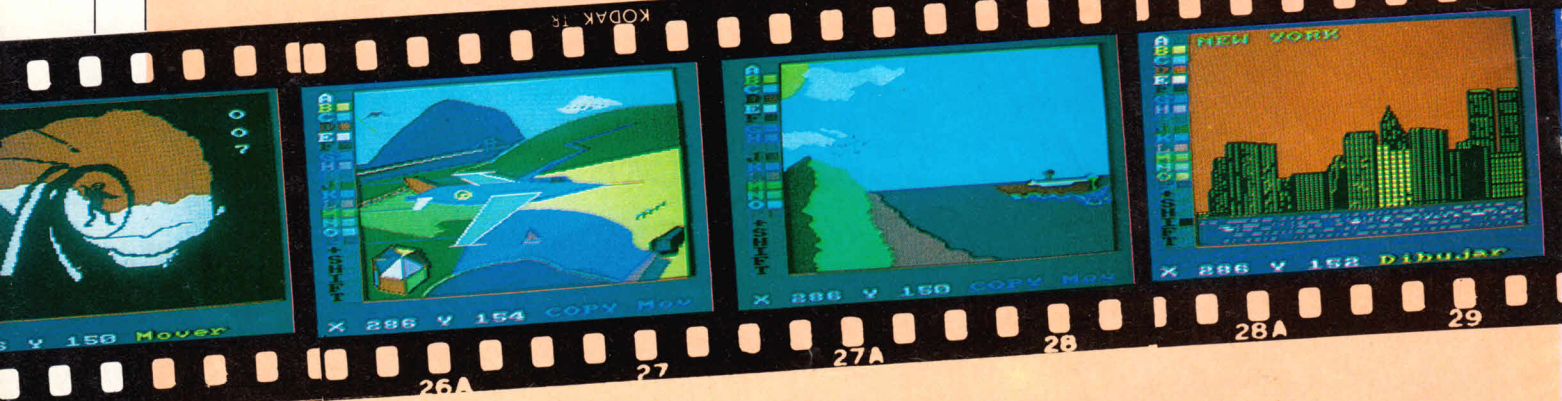
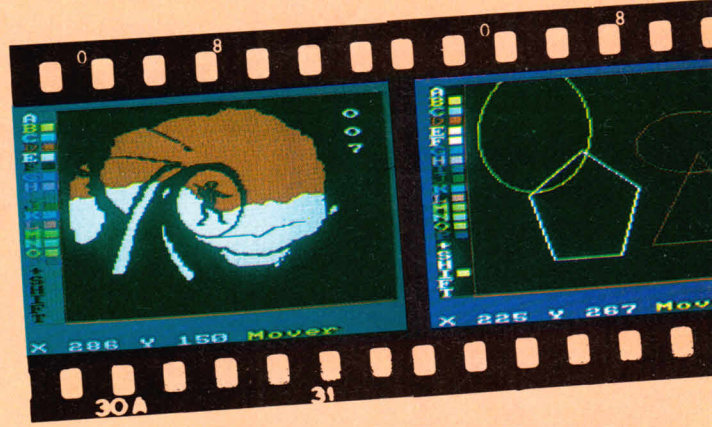
2960 IF ASC(jump$) (< 48 THEN GOTO 2950
2970 IF VAL(jump$)<4 THEN LOCATE #3,1,1:
PRINT #3,SPACE$(40):GOTO 2950
2980 jump=VAL(jump$)
2990 PRINT #3, SPACE$(40):LOCATE #3,1,1:
INPUT #3," Desp. Vertical ";jump$:IF jump
y$=" " OR jumpy$="" THEN 2990
3000 IF ASC(jumpy$) (< 48 THEN GOTO 2990
3010 IF VAL(jumpy$)<2 THEN LOCATE #3,1,1
:PRINT #3,SPACE$(40):GOTO 2990
3020 jumpy= VAL(jumpy$)
3030 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3040 LOCATE #3,1,3:PRINT #3,SPACE$(20):L
OCATE #3,1,3:PEN #3,4:PRINT #3,"X":XPOS;
"Y":YPOS:IF beam=1 THEN PEN #3,1:LOCATE
#3,1,3:PRINT #3,"Dibujar ";ELSE beam=0
:PEN #3,1:LOCATE #3,1,3:PRINT #3,"Mover
";REM Encendido / Apagado
3050 RETURN
3060 REM ** dibuja desde la memoria **
3070 CALL &BB1B:IF INKEY(50)=128 THEN 30
70:CALL &BB1B
3080 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3090 IF sh=0 THEN LOCATE #3,1,1:PRINT #3
,"No existen figuras archivadas":FOR n=
1 TO 3000:NEXT:LOCATE #3,1,1:PRINT #3, S
PACE$(40):RETURN
3100 LOCATE #3,1,1:INPUT #3,"Numero de f
igura";shape$:IF shape$=" " OR shape$=""
THEN GOTO 3100
3110 IF ASC(shape$) (< 48 THEN GOTO 3100
3120 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3130 IF VAL(shape$) >= sh THEN LOCATE #3
,1,1:PRINT #3,"Solo existen figuras archi
vs del 0 al";sh-1:FOR n=1 TO 3000:NEXT:G
OTO 3080

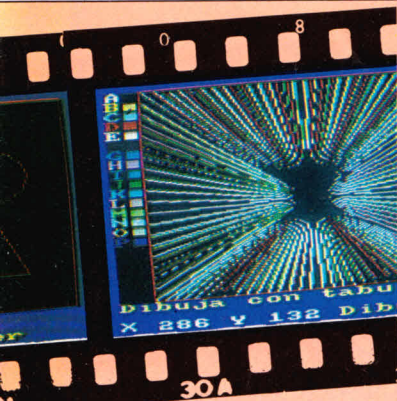
```

```

3140 shape= VAL(shape$)
3150 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3160 LOCATE #3,1,1:INPUT #3,"Ampliacion"
:mag$;mag= VAL(mag$)
3170 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3180 IF mag < 0.1 THEN LOCATE #3,1,1:PRI
NT #3,"Demasiado pequena ";FOR n= 1 TO
1000:NEXT:GOTO 3150
3190 c=0
3200 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3210 IF INSTR(character$(shape)," - ")
1 THEN GOTO 3230
3220 character$(shape)=character$(shape)
+" - "
3230 count=0
3240 WHILE count< (LEN(character$(shape)
)-3)
3250 count=count+1
3260 space=INSTR(count,character$(shape)
." ") :minus=INSTR(count,character$(shape)
."-")
3270 IF space < minus THEN count=space
3280 IF space > minus THEN count=minus
3290 blank$(c)=STR$(count):c=c+1
3300 WEND
3310 REM ** Dibuja desde memoria **
3320 FOR count=2 TO c-3 STEP 2
3330 drax= VAL(MID$(character$(shape),VA
L(blank$(count)),VAL(blank$(count+1))-VA
L(blank$(count))))
3340 dray= VAL(MID$(character$(shape),VA
L(blank$(count+1)),VAL(blank$(count+2))-
VAL(blank$(count+1))))
3350 DRAWR mag*drax,mag*dray,p

```





```

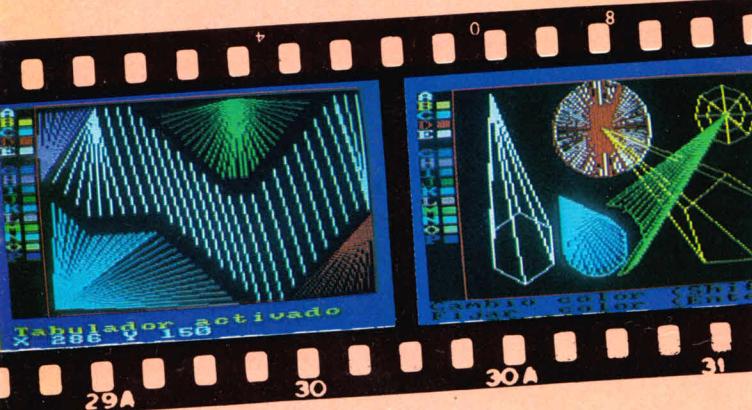
3360 NEXT count
3370 count=0:RETURN
3380 REM ** Salva figuras a cinta o disc
o **
3390 CALL &BB1B: IF INKEY(60)=128 THEN G
OTO 3390:CALL &BB1B
3400 CLS #3:LOCATE #3,1,1:INPUT #3, "Sal
var figuras S/N":satap#:LOCATE #3,1,1:PR
INT #3,SPACE$(40):IF satap#="S" OR satap
#="s" THEN GOTO 3410:ELSE GOTO 3500
3410 LOCATE #3,1,1:INPUT #3, "Nombre de
figura":file#:IF w=1 AND file#="" THEN G
OTO 3410
3420 IF LEN(file#) > 8 THEN LOCATE #3,1,
1:PRINT #3,"Max. 8 letras":FOR n=1 TO 30
0:LOCATE #3,1,1:PRINT #3,SPACE$(40):GOTO
3410
3430 OPENOUT file#
3440 PRINT #9,sh
3450 FOR count=0 TO sh
3460 PRINT #9,character$(count)
3470 NEXT count
3480 CLOSEOUT
3490 CLS #3:PEN #3,4:LOCATE #3,1,3:PRINT
#3,"X":XPOS:"Y":YPOS;
3500 RETURN
3510 REM ** Cargar figuras **
3520 CALL &BB1B:IF INKEY(35)=128 THEN GO
TO 3530:CALL &BB1B
3530 LOCATE #3,1,1:INPUT #3, "Cargar fig
uras S/N":lotap#:LOCATE #3,1,1:PRINT #3,
SPACE$(40):IF lotap#="S" OR lotap#="s" T
HEN GOTO 3540:ELSE GOTO 3650
3540 LOCATE #3,1,1:INPUT #3, "Nombre de
figura":file#:IF w=1 AND file#="" THEN G
OTO 3540
3550 IF LEN(file#) > 8 THEN LOCATE #3,1,

```

```

1:PRINT #3,"Max. 8 letras":FOR n=1 TO 30
0:LOCATE #3,1,1:PRINT #3,SPACE$(40):GOTO
3540
3560 OPENIN file#
3570 INPUT #9,sh
3580 FOR count=0 TO sh
3590 INPUT #9,character$(count)
3600 NEXT count
3610 FOR count=0 TO sh
3620 IF LEFT$(character$(count),1)="-" T
HEN GOTO 3630:ELSE character$(count)="
"+character$(count)
3630 NEXT count:count=0
3640 CLS #3:PEN #3,4:LOCATE #3,1,3:PRINT
#3,"X":XPOS:"Y":YPOS;
3650 RETURN
3660 REM ** Carga dibujo en memoria **
3670 CALL &BB1B:IF INKEY(36)=128 THEN GO
TO 3670:CALL &BB1B
3680 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3690 LOCATE #3,1,1:INPUT #3,"Cargar un d
ibujo S/N":pict#:
3700 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3710 IF pict#="s" OR pict#="S" THEN GOTO
3720:ELSE RETURN
3720 LOCATE #3,1,1:INPUT #3,"Nombre del
dibujo":picture#
3730 IF w=1 AND picture#="" THEN GOTO 37
20
3740 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3750 LOAD picture#,&6675
3760 CLS #3
3770 RETURN
3780 REM ** Regresa al menu de opciones
**
3790 CALL &BB1B:IF INKEY(52)=128 THEN GO
TO 3790:CALL &BB1B
3800 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3810 LOCATE #3,1,1:INPUT #3,"Volver al m
enu S/N":yes#:IF yes#="S" OR yes#="s" TH
EN MODE 2:;y+2: GOTO 160: REM retorna a
l menu de opciones
3820 LOCATE #3,1,1:PRINT #3,SPACE$(40)
3830 RETURN
3840 REM ** salva el dibujo en memoria *
*
3850 FOR n=26203 TO 26226
3860 READ x
3870 POKE n,x
3880 NEXT n
3890 RETURN
3900 DATA 1,0,64,33,0,192,17,117,102,237
,176,201
3910 DATA 1,0,64,33,117,102,17,0,192,237
,176,201

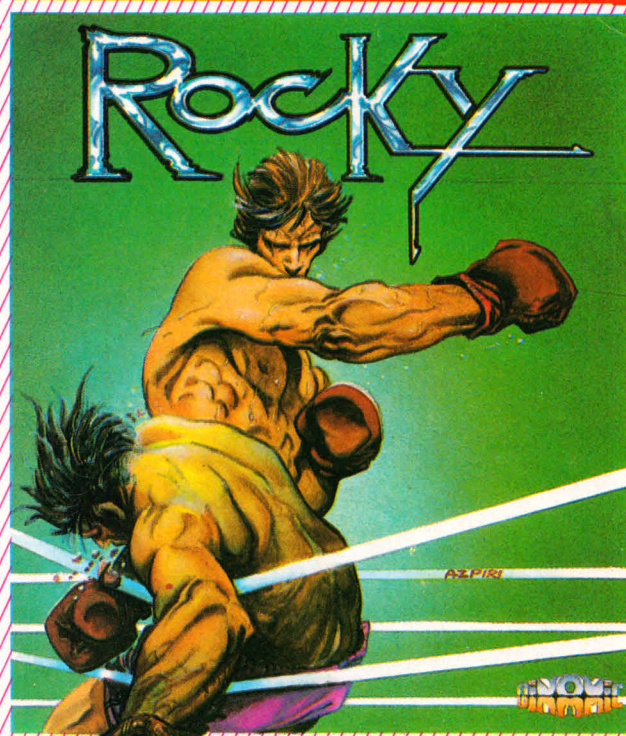
```





Para que tus dedos no realicen el trabajo duro, M.H. AMSTRAD lo hace por ti. Todas las listas que incluyen este logotipo se encuentran a tu disposición en un cassette mensual, solicítaslas.

AMSTRAD



2.100 Pts.

DINAMIC SOFTWARE EN AMSTRAD

Por fin nos hemos decidido a dar el salto, y lo hemos hecho a lo grande, con la versión AMSTRAD de «ROCKY», el programa que ha pulverizado las listas de éxitos de España e Inglaterra y ha recibido las mejores críticas de las revistas especializadas de toda Europa. MICROHOBBY (****), COMPUTER AND VIDEO-GAMES (10, 9, 9, 9). Y solamente piensa, si era excitante en el Spectrum, ¿cómo será en tu AMSTRAD?

¡¡¡NO TE LO PIERDAS!!!

Tiendas y distribuidores	Pedidos contra reembolso
 (91) 447 34 10	 (91) 715 00 67



«MANSION» DINAMIC
 C./ Tilos 2, n.º 21. Montepíncipe. (Boadilla del Monte). Madrid

AMSTRAD CPC-464 CON MONITOR Y MAGNETOFONO INCORPORADO



AMSTRAD: EL *increíble* ORDENADOR

Estamos viviendo la era del ordenador personal. Más de un millón de personas comprarán equipos informáticos en los próximos años: estudiantes, empresarios, educadores, profesionales, comerciantes, los utilizará como herramienta imprescindible en sus actividades. Usuarios cada vez mejor informados, más selectivos y exigentes para los que AMSTRAD, gigante británico de la industria electrónica ha fabricado el ordenador idóneo.

Si en la primavera de 1984 AMSTRAD conmocionó al mundo informático con un modelo CPC 464, la aparición ahora de CPC 664 -en el que el magnetófono ha sido sustituido por una unidad de disco de 3" (180 K) incorporada- vuelve a despertar el

entusiasmo de especialistas y público. El éxito arrollador de ambos modelos encuentra su explicación en la filosofía de diseño de AMSTRAD. Una filosofía que ofrece:

Un sistema completo que incluye la unidad central, el monitor y el magnetófono o la unidad de disco. Un equipo compacto, listo para funcionar sin cableados engorrosos ni necesidad de adquirir más periféricos. Sólo requiere desembalarlo y enchufar un cable -un sólo cable- a la red. Con un paquete de **programas de obsequio** y, además, el Sistema Operativo CP/M y el lenguaje LOGO incluidos en el suministro del CPC 664.

Unas prestaciones del más alto nivel, con 64 K. de memoria RAM, 32 K de memoria ROM, con resolución de 640 x 200 puntos, 27 colores,

20, 40, u 80 columnas de texto en pantalla, 8 "ventanas" de trabajo, teclado profesional con 32 teclas programables, sonido estéreo con 3 canales y 8 octavas por canal. Y un BASIC super-ampliado y dotado incluso de comando de control del microprocesador (Every, After...).

Una tecnología contrastada y fiable basada en el popular microprocesador Z80A y en una electrónica depurada y con un riguroso control de calidad.

Una extensa biblioteca de programas que se incrementa literalmente día a día y que ya dispone de centenares de títulos para todos los gustos y necesidades: gestión profesional (Contabilidad, Control de Stocks, Bases de Datos, Hojas de Cálculo, Procesadores de Texto,...), educación, lenguajes, y ayuda a la programación

NUEVO AMSTRAD CPC-664 CON MONITOR Y UNIDAD DE DISCOS INCORPORADA



ORDENADOR PERSONAL

(Ensamblador, Desensamblador, Pascal, Forth, Logo, Diseñador de Gráficos, Diseñador de Sprites...), de toma de decisiones (Proyect Planner, Decisión Maker,...), juegos de habilidad (La Pulga, Manic Miner, Decathlon, Android,...), juegos de inteligencia (Ajedrez, Backgamon,...), juegos de estrategia (Batalla de Midway, Il Guerra Mundial, ...), juegos de aventuras (Hobbit, Sherlock Homes,...), juegos de simulación (simulador de Vuelo, Tenis, Billar, Mundial de Fútbol,...).

Una asistencia técnica rápida y eficaz que **AMSTRAD ESPAÑA** garantiza **exclusivamente** a los equipos adquiridos a través de su Red Oficial de Distribuidores y acompañados de la **Tarjeta de Garantía de AMSTRAD ESPAÑA**.

Unos precios increíbles que no admiten comparación con los de cualquier otro ordenador personal de sus características.

- * Ordenador CPC 464, con magnetófono incorporado. Manual del Usuario y obsequio del Libro "Guía de Referencia del programador" y de 8 Programas:
 - Con Monitor de fósforo verde (12")... **64.900 pts.**
 - Con Monitor color (14")..... **93.900 pts.**
- * Ordenador CPC 664, con Unidad de Disco incorporada, Manual del Usuario, incluyendo Sistema Operativo CP/M, Lenguaje Logo y obsequio de cinco programas (**Base de Datos, Proceso de Textos, Diseñador de Gráficos, Random Files, Puzzle y Animal, Vegetal, Mineral.**

- Con Monitor de fósforo verde (12").. **109.500 pts.**
- Con Monitor color (14")..... **134.500 pts.**

AMSTRAD [®]
ESPAÑA

Avda. del Mediterráneo, 9
Tels. 433 45 48 - 433 48 76
28007 MADRID

Delegación Cataluña:
Tarragona, 100 - Tel. 325 10 58
08015 BARCELONA

NOTA: Es muy importante verificar la garantía del aparato ya que sólo **AMSTRAD ESPAÑA** puede garantizarle la ordenada reparación y sobre todo materiales de repuesto oficiales (Monitor, ordenador, cassette o unidad de discos).

Es una marca registrada del Grupo Indescomp

Código máquina

DEL BIT AL BYTE

En el mundo de los ordenadores, existen dos palabras mágicas que despiertan o el terror o el apetito de todos los entusiastas de la microinformática: **CODIGO MAQUINA**. Todos sabemos que los lenguajes de alto nivel como el BASIC, fueron introducidos en los computadores para facilitar la comunicación hombre-máquina; también sabemos que adolecen de considerables limitaciones frente al lenguaje máquina, fundamentalmente velocidad e imposibilidad de aprovechar al máximo las posibilidades del ordenador.

El pleno dominio del AMSTRAD sólo puede lograrse trabajando en código máquina. Teniendo este hecho en la mente, a lo largo de esta serie de artículos se tratará de responder a preguntas del tipo «¿qué es exactamente el código máquina?», «¿qué son esos símbolos tan raros llamados mnemónicos?» y un largo etcétera que, si bien no nos convertirá en el máximo exponente de la programación en lenguaje máquina, sí nos faculta para entender como funciona este asunto y escribir nuestros propios programas rápidamente, con sencillez y efectividad.

¿Qué es el código máquina?

Es el momento de preguntarse «¿qué es EXACTAMENTE un programa en lenguaje máquina?»

Para empezar, TODOS los programas, BASIC incluido, son lenguaje máquina; el significado de esto se hará claro dentro de un minuto.

Siguiendo escrupulosamente las más puras tradiciones, éste sería el momento de abrumar al lector haciéndole saber que el AMSTRAD posee en su interior, como «corazón del sistema», un microprocesador Z80, el cual tiene un juego de registros que pueden tratarse como de 16 bits y un bus de direcciones también de 16 bits; para completar el cuadro, se recomendaría al lector que estudiara cuidadosamente un ominoso diagrama de bloques en donde se pretende hacer ver con toda claridad la «arquitectura del sistema».

No va a haber que soportar nada de eso aquí; sólo nos interesa saber que un programa en código máquina, es exactamente un montón de números. Más concretamente, un montón de números comprendidos entre 0 y 255.

La memoria de un microprocesador está compuesta de gran cantidad de «células de memoria» interconectadas; algo bastante parecido a nuestro propio cerebro, salvando las distancias. Y, como ocurre con nosotros, una célula puede recordar una cantidad limitada de cosas. En el caso del Z80, sólo puede contener un byte a un tiempo, y un byte resulta ser un número que varía cíclicamente de 0 a 255.

Aunque estos límites pueden parecer algo arbitrarios, hay una buena razón para ello y, para poderla entender, no va a quedar más remedio que echar una mirada (rápida) a la trastienda del AMSTRAD.

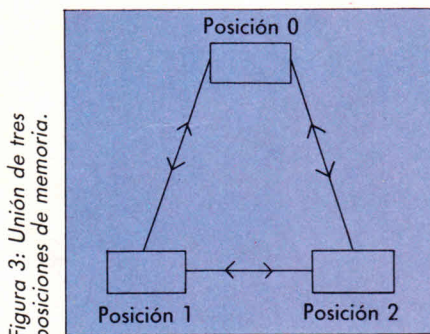


Figura 3: Unión de tres posiciones de memoria.

La memoria del AMSTRAD

Cada célula de memoria, o posición de memoria como a partir de ahora la llamaremos, consiste en una serie de ocho «interruptores» cada uno de los cuales puede estar encendido o apagado; casualmente, existen 255 formas distintas de colocar 8 interruptores unos encendidos y otros apagados, por lo que somos capaces de codificar en cada posición de memoria 255 «cosas», que, casualmente una vez más, son NUMEROS.

Si observamos la Tabla 1, puede apreciarse que lo único que refleja es la asociación de un número con cada uno de los 8 interruptores de una hipotética posición de memoria, numerados del 0 al 7 (los ordenadores comienzan a contar por el 0).

Debajo de cada interruptor («switch» en jerga informática), está el número relacionado con él, cuyo valor se va doblando a medida que avanzamos hacia el séptimo.

Resulta muy sencillo codificar guarismos empleando este sistema. Por ejemplo, si el «switch» 4 está encendido (ON) y todos los demás apagados (OFF), hemos almacenado en nuestra posición de memoria el número 16. Análogamente, si sólo el 7 estuviera encendido, tendríamos el 128.

De aquí se sigue que, al mantener en estado

Interruptor	7	6	5	4	3	2	1	0
Valor	128	64	32	16	8	4	2	1

Tabla 1: Valores asociados a cada interruptor.

Interruptor	7	6	5	4	3	2	1	0
Valor	128	64	32	16	8	4	2	1
Estado	off	off	on	off	off	off	on	off

Suma - - - - > 32 + 2 = 34

Figura 1: Formato del n.º 34 con interruptores.

«ON» más de un interruptor a la vez, podremos codificar bastantes más cifras que los que aparecen en la Tabla 1. La Figura 1 muestra como. Ahora tienen pleno sentido los límites superior e inferior de 255 y 0 que exigíamos a los números capaces de almacenarse en un byte; en efecto, todos los interruptores apagados suman 0, y, todos encendidos, 255.

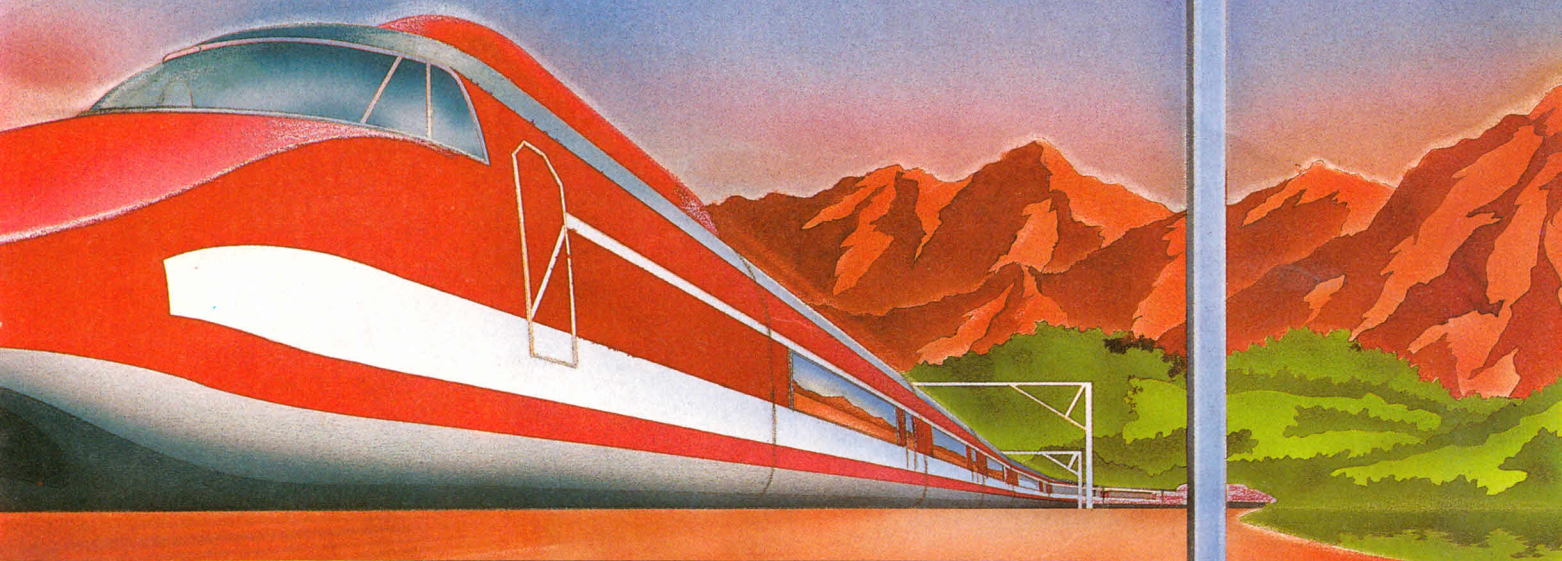
Lo verdaderamente interesante de este método, es que cada número posee una única configuración de switches encendidos y apagados, por lo cual pueden identificarse unívocamente, excluyendo toda posibilidad de confusión.

Almacenar números

Volviendo a la figura 1, resulta obvio que escribir el valor 34 como «off off on off off on off» no tiene sentido; por ello los matemáticos decidieron en su día que, al existir sólo dos estados distintos para cada switch, el número 1 representaría «ON» y el 0 «OFF», lo que llamamos NUMEROS BINARIOS. Por ejemplo, 106 en binario se escribe %01101010 (Figura 2). La razón de anteponer el signo «%» es para evitar confusiones con sus homólogos decimales. Incidentalmente, a cada switch (o interruptor) se le conoce como un bit, y, puesto que un byte consta de 8 de tales switches, podemos decir que un byte



AMSTRAD



«tiene» 8 bits. Con ellos podemos almacenar en la memoria cualquier número entre 0 y 255, en total 256 números, pero si queremos que un ordenador sirva para algo, necesitamos una cantidad mucho mayor de bytes.

El AMSTRAD se las arregla para manipular 65536 posiciones de memoria, numeradas de 0 a 65535. ¿Por qué precisamente este número? bien, la explicación se encuentra en la tabla 2, la cual es muy parecida a la Tabla 1 pero con 8 switches más, esto es, otro byte.

Al conjunto de estos dos bytes se le denomina una DIRECCION DE MEMORIA, tradicionalmente dividida en dos partes: byte alto (*bits 8 a 15*) y byte bajo (*bits 0 a 7*). Si todos los bits están colocados a 1, el número resultante sería:

%1111111111111111	
= 32768 +	
16384	} valores del byte alto
8912	
4096	
2048	
1024	
512	
256	
128	} valores del byte bajo
64	
32	
16	
8	
4	
2	
1	
TOTAL 65535	

Así decimos que el microprocesador Z80 es de 8 bits, y que puede direccionar 65536 bytes.

Todo son números

La razón de estudiar con cierto detalle unas cuestiones que pueden parecer elementales, estriba en el hecho de que todo lo referente al código máquina gira en torno a números (bytes) almacenados en la memoria del ordenador y a su movimiento a través de la misma, de una posición de memoria a otra.

Esta «danza» conlleva sustanciales cambios de SIGNIFICADO en la información codificada en cada byte, esto es, el mismo número representa cosas muy diferentes en CONTEXTOS distintos.

Por ejemplo, al encender el AMSTRAD, comienza a ejecutarse un programa en código máquina conocido como sistema operativo o, más técnicamente, como el «firmware».

Una de sus misiones es escribir en la pantalla el mensaje de bienvenida que el ordenador muestra al encenderlo.

Este texto se encuentra almacenado en algún lugar de la memoria, y, por el simple hecho de aparecer en pantalla, es decir, de CAMBIAR DE POSICION DE MEMORIA, pasa a ser una letra que nosotros podemos entender, en lugar de %01100111, pongamos por caso.

El mismo tipo de TRANSFERENCIA DE DATOS ocurre cuando pulsamos una tecla; el sistema operativo envía el valor numérico (*el número codificado en binario*) de la tecla pulsada desde la posición de memoria que «recuerda» la tecla de que se trata, hasta otra posición correspondiente a la zona de memoria de pantalla, con lo cual se hace visible y cobra significado para nosotros.

También, cuando salvamos un programa BASIC

en cinta/disco, una parte del programa en lenguaje máquina del firmware, lo mueve desde su lugar habitual en la memoria del ordenador hasta el «port» de salida de datos.

Todo se realiza moviendo bytes de datos de «acá para allá». Vamos a emplear una analogía para tratar de entenderlo un poco mejor. La figura 3 muestra un esquema muy simple de un «mimi-amstrad» con sólo 3 posiciones de memoria; la intercomunicación y el camino que seguirían los datos resulta obvio. Sin embargo, observando la figura 4, donde sólo hay 6 posiciones de memoria, no resulta tan sencillo seguirle la pista a los bytes que se transfieren. Si extrapolamos estos diagramas al caso real, donde hay decenas de miles de posiciones de memoria, es fácil concluir que se necesita una estructura del tipo de la figura 5, donde a cada posición se accede a través de una única vía, atravesando necesariamente un «controlador de tráfico», que también es una posición de memoria y que se conoce en la jerga de computadores como REGISTRO. En este caso, el registro se denomina A (*de ACUMULADOR*), y sólo es capaz de almacenar un byte a la vez. Ahora supongamos que se desea mover un byte de información desde la posición de memoria 0, hasta la 5, vía el registro A. El ordenador necesita recibir 2 instrucciones:

1. Cargar el registro A con el número CONTENIDO en la posición de memoria 0.
2. Cargar la posición 5 con el número que ésta en el registro A.

Hablando muy en general, éste es el método ideal para transferir datos de un lugar a otro de la memoria: siempre a través de un estadio intermedio, el registro. Un esquema más real de este movimiento puede verse en la figura 6.

Es fácil imaginar que, para que el ordenador pue-

LIBROS EN CASTELLANO PARA TU ORDENADOR

AMSTRAD SPECTRAVIDEO **sinclair ZX Spectrum y QL** **MSX**



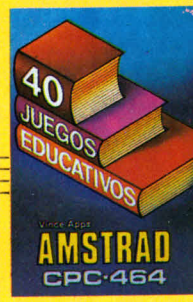
*Manual de Referencia Basic del Program. AMSTRAD. La más autorizada y completa guía para programar en Locomotive Basic. 3.400.— Pts.



*Sensacionales Juegos AMSTRAD. Listados completos de 27 estupendos juegos de muy diversos estilos. 1.950.— Pts.



*Programando con AMSTRAD. Fundamental para el usuario principiante. Ameno y repleto de ejemplos. 2.400.— Pts.



*40 Juegos Educativos AMSTRAD. Listados completos (matemáticas, geografía, música, etcétera) para aprender divirtiéndose. 1.950.— Pts.



*Lenguaje Máquina... AMSTRAD. Ideal para iniciarse en el código máquina del Z80 Y EN EL SISTEMA OPERATIVO DEL AMSTRAD. 2.100.— Pts.



*Inteligencia Artificial AMSTRAD. Convierta su AMSTRAD en un compañero inteligente. 1.500.— Pts.



*Sonidos y Música AMSTRAD. Programe música y efectos sonoros y convierta su AMSTRAD en un sintetizador. 1.200.— Pts.



*Programación Básica SPECTRAVIDEO. Imprescindible para iniciarse en el dominio de las estructuras fundamentales del Basic. 1.800.— Pts.



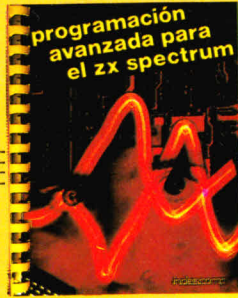
*Programación Avanzada SPECTRAVIDEO. Para "saber más": ficheros, subrutinas, gestión de errores, funciones definibles, etcétera. 2.400.— Pts.



*Codigo Máquina SPECTRUM. Las instrucciones fundamentales del Z80 para iniciarse en el código máquina. 2.100.— Pts.



*Los 20 mejores programas. Selección de excelentes programas en Basic. 1.800.— Pts.



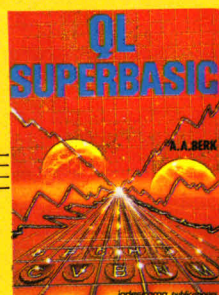
*Programación avanzada. Subrutinas, trucos y análisis para mejorar tus programas. 2.200.— Pts.



*Las 40 mejores SUBROUTINAS. Las más útiles rutinas en código máquina reunidas en un solo volumen. 1.950.— Pts.



*Programando con QL. Texto introductorio, claro, útil y ameno. 1.950.— Pts.



*QL Superbasic. Un curso avanzado para dominar el excelente Basic de tu QL. 1.950.— Pts.



*Programando con MSX Basic. Curso completo y detallado, con numerosos ejemplos prácticos. 2.200 Pts.



*El libro de Juegos MSX. Listados completos y comprobados de 21 excelentes juegos. 1.900.— Pts.

indescorp[®]
PUBLICACIONES

Avda. del Mediterráneo, 9
Tels. 433 45 48 - 433 48 76
28007 MADRID
Delegación en Cataluña:
Tarragona, 110 - Tel. 325 1058
08015 BARCELONA

DE VENTA EN *El Corte Inglés*
Y TIENDAS ESPECIALIZADAS

(tm) Marca registrada por el Grupo Indescorp

Byte alto	Bit	15	14	13	12	11	10	9	8
	valor	32768	16384	8192	4096	2048	1024	512	256
Byte bajo	Bit	7	6	5	4	3	2	1	0
	valor	128	64	32	16	8	4	2	1

Tabla II: Formato de un n.º de 16 bits - comparar con Tabla I.

Interruptor	7	6	5	4	3	2	1	0
Valor	128	64	32	16	8	4	2	1
Estado	off	on	on	off	on	off	on	off
Binario	0	1	1	0	1	0	1	0

Suma ---- > 64 + 32 + 8 + 2 = 106 decimal
 ---- > %01101010 binario

Figura 2: Representación binaria de i06.

da funcionar a pleno rendimiento, debe existir más de un «controlador de tráfico», fundamentalmente por dos razones:

A. Evitar «ataascos» en la transferencia de datos.

B. La necesidad de trabajar con números grandes, de más de un byte.

Efectivamente, el Z80 tiene varios registros que resuelven estas dificultades. Posteriormente serán analizados en otros artículos.

Una vez convencidos que en código máquina todo se realiza moviendo números a través de la memoria, surge la pregunta de que **¿cómo le decimos a la máquina lo que tiene que hacer?**

La respuesta es muy simple: creamos una lista de números binarios y los almacenamos en la memoria (!).

No es broma, de veras; esos bytes tienen significado para el Z80 por sí mismos, y todo lo que hay que hacer es decirle que salte a la posición de memoria donde está el primer byte. En ese contexto, el microprocesador lo interpretará como un programa en código máquina a ejecutar, no como bytes de datos.

Ejecutar un programa

Para cumplir la tarea discutida más arriba de trasladar un byte desde la posición 0 a la 5, necesitaríamos colocar en la memoria la siguiente serie de números:

58 0 0 50 5 0 201

aunque estén escritos en decimal, el micro los leerá en binario, por supuesto. La figura 7 explica el significado, para el Z80, de lo que le estamos diciendo que haga.

El 58 es un «opcode» (byte de programa que implica realizar una acción, no de datos) que le indica al Z80 que cargue el registro A con el CONTENIDO de una particular posición de memoria.

Además, sabe que la dirección donde está el número que busca se encuentra en los 2 bytes inmediatamente siguientes al opcode (recordar la necesidad de 2 bytes para especificar una DIRECCION de memoria). Una vez comprendido esto, el Z80 traslada su atención a la dirección especificada, y coloca en el registro A el número que encuentra allí.

Contenido de la posición de memoria	58	00	00	50	05	00	201
Significado del valor de cada posición	Cargar A con el contenido de la siguiente dirección	Los dos bytes especifican el valor requerido por el código de operación anterior		Cargar la siguiente dirección con el contenido del registro A	Los dos bytes especifican el valor que necesita el código de operación anterior		Regreso a la rutina principal

Figura 7: Explicación de un sencillo programa de código máquina.

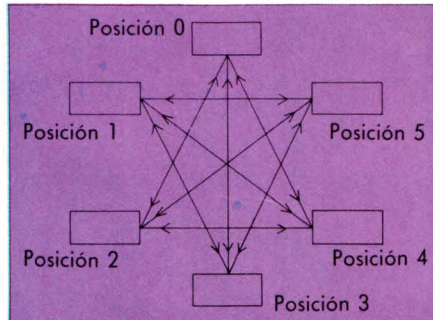


Figura 4: La unión de seis registros resulta más complicada.

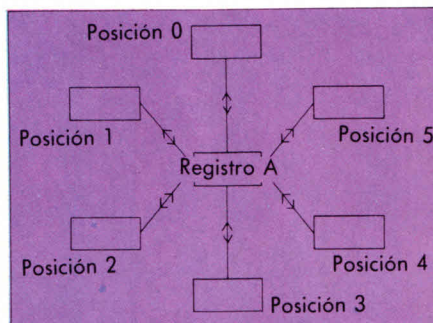


Figura 5: Posiciones de memoria unidas por el registro A.

Los primeros tres bytes de nuestro programa han sido ejecutados.

Ahora le llega el turno al 50, el cual le dice al micro que cargue la posición de memoria especificada por los 2 bytes siguientes con el número contenido en el registro A (operación inversa de la anterior).

Una vez hecho esto, el séptimo y último byte, 201, le dice al microprocesador que se detenga y RETORNE a la tarea que estuviera realizando anteriormente, sea cual sea (en BASIC, el equivalente es RETURN).

No es necesario decirle al ordenador DONDE tiene que retornar al finalizar la ejecución de un programa en código máquina. Cuando esta rutina fue llamada, el Z80 precavidamente almacenó el lugar de registro. De paso, también conviene observar que

los dos números que especifican la posición de memoria se colocan en el orden inverso al esperado, esto es, 5 0 en lugar de 0 5.

Si se coloca el programa de esta manera:

58 0 0

50 5 0

201

no puede decirse que hayamos adelantado mucho en cuanto a su comprensión a golpe de vista, ya que nuestra mente está más habituada a las palabras que a los números. Sin embargo, si escribimos:

LD A, (0) 58 0 0

LD (5), A 50 5 0

RET 201

los símbolos del lado izquierdo son conocidos como mnemónicos (recordatorios), aunque por desgracia, en inglés. Así, LD recuerda a Load (cargar) y RET a RETURN.

La versión definitiva de nuestro programita quedaría así:

LD A, (0) Carga el registro A con el contenido de la posición de memoria 0.

LD (5), A Carga el contenido de la posición de memoria 5 con el contenido del registro A.

RET Regresa al programa que llamó a esta rutina y continúa ejecutándolo donde lo dejaste.

Existen programas especiales llamados ENSAMBLADORES que permiten escribir programas en lenguaje máquina en este formato, o sea, en mnemónicos que luego el ensamblador se encarga de colocar en la memoria traducidos a números binarios.

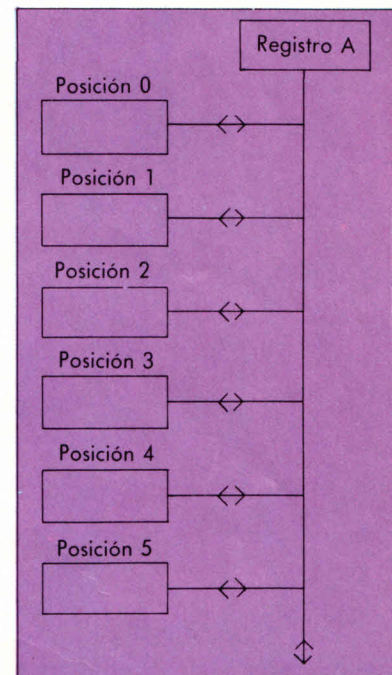
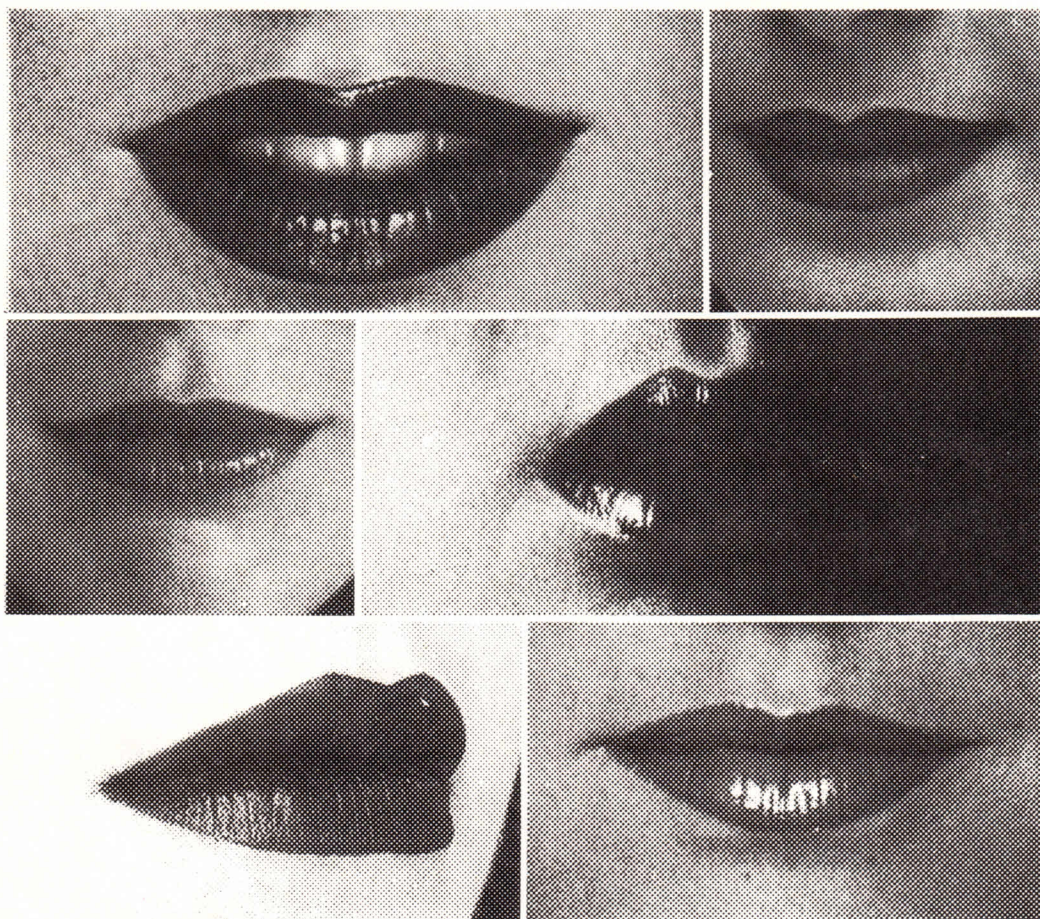


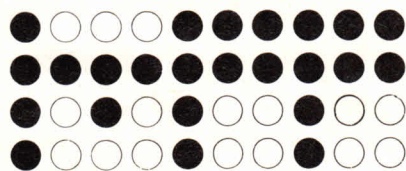
Figura 6: Una representación más ajustada a la realidad de la unión de registros.

Por ahora, esto es todo. Esperamos que os hayáis podido hacer una clara idea de qué es el código máquina.

La Microinformática más profesional,
a su servicio.



“No hablamos por
hablar”



Microtodo. Todo en Microinformática.

C/ Orense, 3. Tfno.: 253 21 19. 28020 - MADRID.

GANAR UN AMSTRAD CPC664 PARTICIPANDO EN NUESTRA ENCUESTA

M. H. AMSTRAD, para acercarnos más y más a la revista a los gustos y preferencias de nuestros lectores, plantea la siguiente encuesta que estamos seguros ayudará a hacer una revista abierta a todo tipo de tendencias dentro del mundo de la informática.

Entre todas las cartas recibidas, sorteamos un **AMSTRAD CPC664** y 4 unidades de disco.

Rellenad la encuesta que a continuación os adjuntamos colocando una X en la casilla apropiada y enviadla a:

HOBBY PRESS, S.A.
AMSTRAD SEMANAL

Apartado de Correos 54.062
28080 Madrid

Nombre y apellidos Edad

Domicilio

Localidad C. Postal

Provincia Teléfono

Modelo de AMSTRAD	CPC464	<input type="checkbox"/>	CPC664	<input type="checkbox"/>				
¿Para qué lo usas?	Juegos	<input type="checkbox"/>	Gestión	<input type="checkbox"/>	Otros	<input type="checkbox"/>		
¿Te interesa la programación?	Sí	<input type="checkbox"/>	No	<input type="checkbox"/>				
Lenguajes que utilizas	Basic	<input type="checkbox"/>	Cód. M.	<input type="checkbox"/>	Pascal	<input type="checkbox"/>	Logo	<input type="checkbox"/>
¿Te gustaría aprender nuevos lenguajes?	Sí	<input type="checkbox"/>	No	<input type="checkbox"/>				
¿Cuáles?	Basic	<input type="checkbox"/>	Cód. M.	<input type="checkbox"/>	Pascal	<input type="checkbox"/>	Logo	<input type="checkbox"/>
	Forth	<input type="checkbox"/>	C	<input type="checkbox"/>				

Programas

¿Sueles teclear los programas de las revistas? Casi todos 1 de 2 1 de 4 1 de 8 Ninguno

¿Qué tipo de programas te interesan? Juegos Utilidades

Juegos Marcianos Deportivos Aventuras animadas

Estrategia Inteligencia Aventuras de texto

Utilidades Procesar textos Bases de datos Hojal calc.

Lenguajes Gestión comercial Gráficos

¿Te gusta que se comenten juegos en tu revista?

¿Cuántas páginas/semana? Ninguna 1 3 5 Más

¿Qué opinas de los artículos de Cód. M.? ¿Te interesan?

¿Cuántas páginas/semana? Ninguna 1 3 5 Más

¿Te gustaría que se hablase del ordenador y sus periféricos? ¿Cuántas páginas/semana?

Ninguna 1 3 5 Más

¿Te interesaría una sección dedicada a principiantes en Basic Amstrad? ¿Cuántas páginas/semana?

Ninguna 1 3 5 Más

¿Te interesaría una sección dedicada a gráficos y sonido en el Amstrad? ¿Cuántas páginas/semana?

Ninguna 1 3 5 Más

¿Te gustaría artículos acerca de programas de aplicación comerciales? ¿Cuántas páginas/semana? ...

Ninguna 1 3 5 Más

¿Qué secciones añadirías a la revista?

¿Qué secciones quitarías?



Sin duda alguna

A través de esta sección se pretende resolver, en la medida de lo posible, todas las posibles dudas que «atormenten» a todas las personas interesadas en el mundo del AMSTRAD, sean o no poseedores de uno y, si lo son, se encuentren en cualquier nivel de destreza en su manejo.

Semanalmente, aparecen en estas páginas las consultas de la mayor cantidad de usuarios posible; ello redundará en un mejor servicio y en un contacto más estrecho entre todos nosotros a través de la revista.

SIN DUDA ALGUNA está abierta a todos.

Seréis, semana a semana, los encargados de construir esta página con vuestras consultas. En más de una ocasión, aquello que os preocupa ya ha sido contestado antes a otro lector o, por el contrario, puede suceder que determinada consulta aclare muchos quebraderos de cabeza de otros aficionados.

Las cartas «sin duda alguna», nos servirán de gran ayuda. Gracias a ellas podremos ir evaluando vuestras necesidades y, de este modo, modificando el contenido de MICROHOBBY AMSTRAD acorde con ello.

¡Os esperamos!

GANAR 100.000 PESETAS CON MICROHOBBY

AMSTRAD SEMANAL

Porque pretendemos que **AMSTRAD SEMANAL** sea también vuestra revista, hemos abierto una sección en la que se publicarán los mejores programas originales recibidos en nuestra redacción. Vosotros seréis los encargados de realizar estas páginas, en las que podréis aportar ideas y programas interesantes para otros lectores.

Las condiciones son sencillas:

- Los programas se enviarán a **AMSTRAD SEMANAL** en una cinta de cassette, sin protección en el software, de forma que sea posible obtener un listado de los mismos.
- Cada programa debe ir acompañado de un texto explicativo en el cual se incluyan:
 - Descripción general del programa.
 - Tabla de subrutinas y variables utilizadas, explicando claramente la función de cada una de ellas.
 - Instrucciones de manejo.

— Todos estos datos deberán ir escritos a máquina o con letra clara para mayor comprensión del programa.

— En una sola cinta puede introducirse más de un programa.

— Una vez publicado, **AMSTRAD SEMANAL** abonará al autor del programa de **15.000 a 100.000** pesetas, en concepto de derechos de autor.

— Los autores de los programas seleccionados para su publicación, recibirán una comunicación escrita de ello en un plazo no superior a dos meses a partir de la fecha en que su programa llegue a nuestra redacción.

— **AMSTRAD SEMANAL** se reserva el derecho de publicación o no del programa.

— Todos los programas recibidos quedarán en poder de **AMSTRAD SEMANAL**.

— Los programas sospechosos de plagio serán eliminados inmediatamente.

¡ENVÍANOS TU PROGRAMA!
a **HOBBY PRESS, S. A. La Granja, n.º 8. Pol. Ind. Alcobendas (Madrid)**

Mercado común

Con el objeto de fomentar las relaciones entre los usuarios de **AMSTRAD**, **MERCADO COMUN** te ofrece sus páginas para publicar los pequeños anuncios que relacionados con el ordenador y su mundo se ajusten al formato indicado a continuación.

En **MERCADO COMUN** tienen cabida, anuncios de ventas, compras, clubs de usuarios de **AMSTRAD**, programadores, y en general cualquier clase de anuncio que pueda servir de utilidad a nuestros lectores.

Envíanos tu anuncio mecanografiado a: **HOBBY PRESS, S.A. AMSTRAD SEMANAL.**
Apartado de correos 54.062
28080 MADRID
¡ABSTENERSE PIRATAS!

COMPUTER CENTER

Comandante Zorita, 13
28003 MADRID

MICRO-1. Jorge Juan, 116. Tel.: (91) 274 53 80.
MADRID.

MICROLID. Gregorio Fdez., 6. Tel.: (983) 35 26 27.
VALLADOLID.

BYTE. Plaza Padre Damián, 2. Tel. (967) 23 78 55.
ALBACETE.

AMSTRAD 464 Verde	57.900	AMSTRAD 464 Color	84.900
AMSTRAD 664 Verde	93.900	AMSTRAD 664 Color	111.500
IMPRES. STAR Géminis 10X	54.900	INTERFACE DISCO 5.25"	5.900
DISKETTE 3"	895	CINTA C-15 ESPECIAL ORDEN	85

SOFTWARE ENTRETENIMIENTO (CASSETTE)

COMBAT LYNX	1.925	ALIEN-8	1.875	JUMP JET	2.095
DEAT PIT	1.925	KNIGHT LORE	1.875	SIMULAD. VUELO 737	1.795
BEACH HEAD	2.395	HARRIER ATTACK	995	FRUIT MACHINE	995

SOFTWARE GESTION (DISCO)

CONTAB. GRAL.	11.995	CONTROL STOCK	6.995	BASE DE DATOS	6.995
---------------	--------	---------------	-------	---------------	-------

LIBROS

CURSO AUTODIDACTICO BASIC AMSTRAD (Contiene manual y dos cassettes)	2.695	40 JUEGOS EDUCATIVOS	800
HACIA LA INTELIGENCIA ARTIF.	1.300	MUSICA Y SONIDO PARA AMSTRAD	1.300
		PROGRAMANDO CON AMSTRAD	1.900

Tu pedido lo puedes recibir contra-reembolso (LIBRE de GASTOS), llamando a los teléfonos (91) 233 07 35 y (91) 233 07 81.

¡EL NO VA MAS... DEL VERANO!



En Sinclair Store, este verano, te ponemos la informática aún más fácil. Hemos revisado todos los precios con objeto de rentabilizar, aún más, tu inversión.

Hemos traído todos los periféricos para tu ordenador favorito... Y todos los programas están rebajados hasta con el 40% de dto.

Te destacamos, entre otras, las siguientes ofertas:

LO MAS SERIO Y PROFESIONAL

Con el **CPC 664** te regalamos un disco con el CP/M y lenguaje LOGO y otro disco con la base de datos, proceso de texto, diseñador de gráficos, juegos, etc.



Con el **CPC 464** te regalamos 8 magníficos programas y el libro «Manual de referencia Basic del programador».

Adquiere el increíble AMSTRAD desde 58.400 ptas.

Ordenador
SONY
62.910 pts.

OFERTA
IMPRESORA
GP-50-S
19.990 ptas.

OFERTA MOCHILA COMMODORE

Contiene: 1 Commodore 64,
1 cassette, 4 libros,
2 programas, 1 cartucho
1 joystick y la MOCHILA
COMMODORE



AMSTRAD-COMMODORE-SONY-SPECTRAVIDEO
SPECTRUM-QL

sinclair store
SOMOS PROFESIONALES

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

Presentando este anuncio se descontaran 10.000 pts. al comprar el increíble AMSTRAD (con monitor y unidad de disco incorporada)