

AÑO II N.º 50

**MICROHOBBY**

# AMSTRAD

*Semanal*

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

AÑO II N.º 50

**160 Ptas.**

Canarias 165 pts.

“PASA” DE  
INTERRUPCIONES  
COMO PROTEGER  
PROGRAMAS  
EN BASIC.

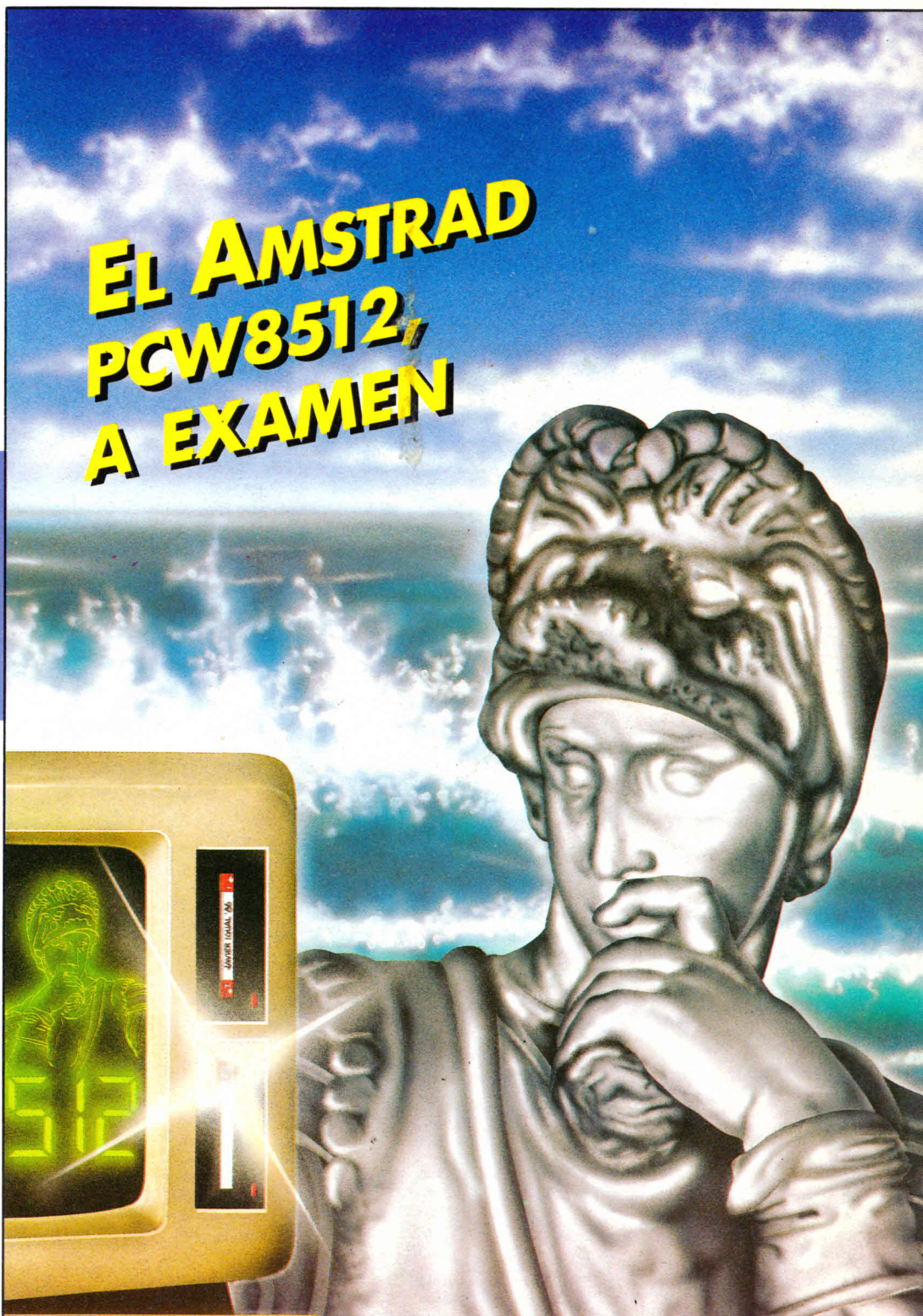
**ESPECIAL JUEGOS:**  
TRES PROGRAMAS  
PARA DIVERTIRSE  
A TOPE

**Música**  
profesional  
para todos  
con «The music  
system»

**CODIGO MAQUINA**

**Y SIN EMBARGO,  
SE MUEVE**

**EL AMSTRAD  
PCW8512,  
A EXAMEN**



HOBBY PRESS

# DOS OFERTAS EXCEPCIONALES PARA COMPLETAR TU COLECCION DE MICROHOBBY AMSTRAD

(Válidas hasta el 31 de octubre)

## ahorra más de 2.000 ptas.

- Veinte revistas (del número 21 a 40)
  - Cintas 6 a 10 con todos los programas publicados en los números 21 a 40 de Microhobby Amstrad
  - Estuche para contener las cintas y las revistas (incluye índice)
- Precio real 7.650 ptas.  
Precio oferta 5.500 ptas.**



## ahorra 1.000 ptas.

- Cintas números 6 a 10, con todos los programas publicados en los números 21 a 40 de Microhobby Amstrad.
  - Estuche para contener las cintas y las revistas (incluye índice)
- Precio real 4.450 ptas.  
Precio oferta 3.350 ptas.**



Recorta o copia este cupón y envíalo a Hobby Press, S. A. Apartado de Correos 232. Alcobendas (Madrid)

Apellidos \_\_\_\_\_ Nombre \_\_\_\_\_ de nacimiento \_\_\_\_\_

Domicilio \_\_\_\_\_ Localidad \_\_\_\_\_ Provincia \_\_\_\_\_

C. Postal \_\_\_\_\_ Teléfono \_\_\_\_\_ Profesión \_\_\_\_\_ ¿Eres suscriptor de Microhobby Amstrad? \_\_\_\_\_

Marco con una cruz la opción que más me interesa

- Deseo recibir en mi domicilio las revistas 21 a 40 de **Microhobby Amstrad**, las cintas 6 a 10 correspondientes a estos números y el estuche para mantener la colección ordenada (contiene índice) **al precio especial de 5.500 ptas.**, lo que me supone un ahorro de 2.150 ptas.
- Deseo recibir en mi domicilio las cintas 6 a 10 que incluyen todos los programas publicados por **Microhobby Amstrad** en los números 21 a 40, y el estuche para mantener la colección ordenada (contiene índice) **al precio especial de 3.350 ptas.**, lo que me supone un ahorro de 1.000 ptas.

Forma de pago:

- Mediante talón bancario adjunto a nombre de Hobby Press, S. A.  Mediante giro postal n.º \_\_\_\_\_
- Mediante tarjeta de crédito  Visa  Master  Charge  American Express

Número de la tarjeta \_\_\_\_\_  
Fecha de caducidad de la tarjeta \_\_\_\_\_

Fecha y firma \_\_\_\_\_

# AMSTRAD

## sumario

Año II • Número 50 • 19 al 25 de Agosto  
 160 ptas. (incluido I.V.A.)  
 Canarias, 155 ptas. + 10 ptas. sobretasa aérea  
 Ceuta y Melilla, 155 ptas.

**Director Editorial**  
 José I. Gómez-Centurión

**Director Ejecutivo**  
 José M.ª Díaz

**Redactor Jefe**  
 Juan José Martínez

**Diseño gráfico**  
 Fernando Chaumel

**Colaboradores**  
 Eduardo Ruiz  
 Javier Barceló  
 David Sopena  
 Robert Chatwin  
 Francisco Portalo  
 Pedro Sudón  
 Miguel Sepúlveda  
 Francisco Martín  
 Jesús Alonso  
 Pedro S. Pérez  
 Amalio Gómez  
 Alberto Suñer

**Secretaría Redacción**  
 Carmen Santamaría

**Fotografía**  
 Carlos Candel  
 Chema Sacristán

**Portada**  
 Javier Igual

**Ilustradores**  
 J. Igual, J. Pons, F. L. Frontán,  
 J. Septien, Pejo, J. J. Mora

**Edita**  
 HOBBY PRESS, S.A.

**Presidente**  
 María Andrión

**Consejero Delegado**  
 José I. Gómez-Centurión

**Jefe de Producción**  
 Carlos Peropadre

**Marketing**  
 Marta García

**Jefe de Publicidad**  
 Concha Gutiérrez

**Publicidad Barcelona**  
 José Galán Cortés  
 Tel: (93) 303 10 22/313 71 62

**Secretaría de Dirección**  
 Marisa Cogorro

**Suscripciones**  
 M.ª Rosa González  
 M.ª del Mar Calzada

**Redacción, Administración y Publicidad**  
 Ctra. de Irún km 12,400  
 (Fuencarral) 28049 Madrid  
 Teléfonos: Suscrip.: 734 65 00  
 Redacción: 734 70 12

**Dto. Circulación**  
 Paulino Blanco

**Distribución**  
 Coedis, S. A. Valencia, 245  
 Barcelona

**Imprime**  
 ROTEDIC, S. A. Ctra. de Irún.  
 Km. 12,450 (MADRID)

**Fotocomposición**  
 Novocomp, S.A.  
 Nicolás Morales, 38-40

**Fotomecánica**  
 GROF  
 Ezequiel Solana, 16

**Déposito Legal:**  
 M-28468-1985

Derechos exclusivos  
 de la revista

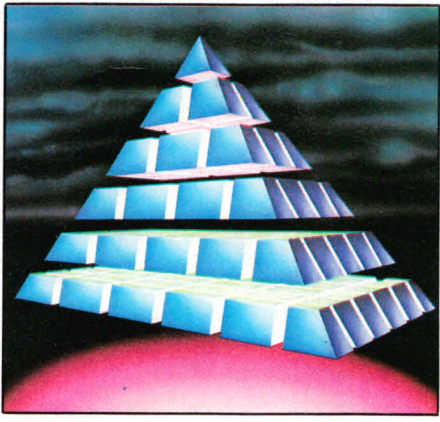
**COMPUTING with  
 the AMSTRAD**

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.

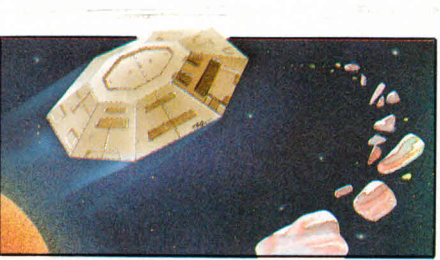
## Primeros 6 pasos

Tal vez todos, en algún momento, hayamos sucumbido a la obsesión de proteger nuestros programas contra curiosos y piratas, a imagen y semejanza de las más renombradas casas de software comercial. Aunque no existe la protección perfecta, el Amstrad Basic es uno de los pocos que cuenta con comandos para permitirnoslo hacer, en mayor o menor medida, como demuestra este artículo.



## Código Máquina 12

Estudiamos cómo podemos desplazar un objeto desde código máquina, respondiendo a las pulsaciones en el teclado.



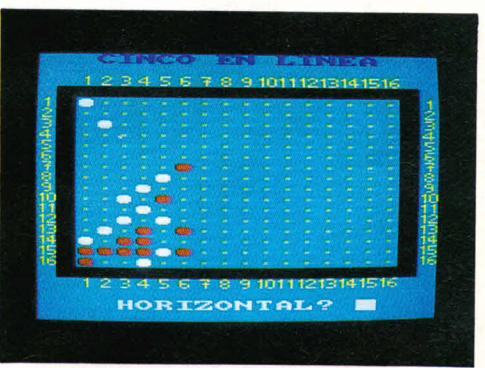
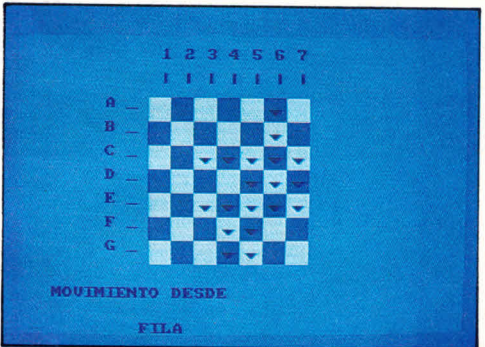
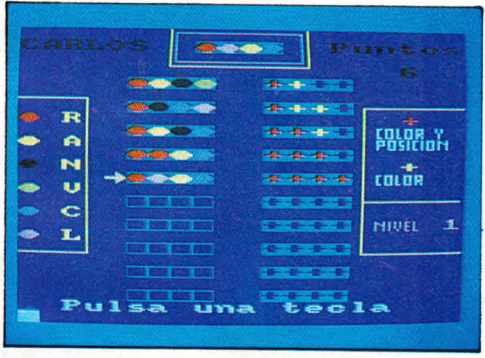
## 16

### Mr. Joystick

Esta semana Mr. Joystick se presenta con una auténtica estrella del software: «The music system», uno de los programas más potentes y sencillos de manejar que hemos visto nunca. Te permitirá realizar lo que quieras con el sonido del Amstrad.

## 24 Serie Oro

Serie oro os ofrece, para las vacaciones, algo muy especial: tres programas cortos, fáciles de teclear y que os harán pasar horas muy entretenidas, para el «relajo» de las tardes de agosto.



## 28

### PCW

Estudiamos a fondo el nuevo ordenador de Amstrad, el PCW8512, intentando responder a la pregunta de qué plantean de nuevo e interesante las dos unidades de disco y la memoria extra.

# Anote este nombre y recuerde esta imagen

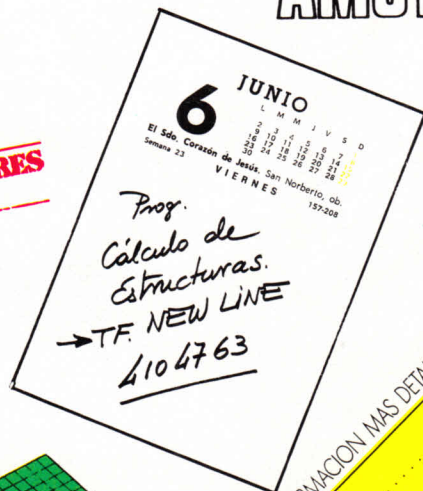


- **Si necesita uno de los mejores programas originales de gestión.**
  - Gestión integrada, facturación, stock, clientes.
- **Si desea mecanizar su negocio.**
  - Clínicas veterinarias.
  - Vídeo clubs.
  - Administración de fincas.
  - Distribuidoras de cine.
- **Si precisa un buen programa técnico.**
  - Cálculo de estructuras.
  - Mediciones y presupuestos.
  - Cálculo de vigas.
  - Estructuras espaciales.
  - Andamios.
  - Cálculos en hormigón.
- **Si lo que quiere es una aplicación puntual.**
  - Declaración de la Renta.
  - lotería primitiva.
  - Agenda multiuso.
- **Si busca que le hagan un programa a la medida.**

**BUSCAMOS DISTRIBUIDORES  
EN TODA ESPAÑA**

**SOFTWARE para**

**AMSTRAD**



**SOFTWARE NEW LINE, S. A.  
Gabinete de Informática**

Zurbano, 4 - 28010 Madrid - Telfs. 410 40 98 - 410 47 63

SOLICITE INFORMACION MAS DETALLADA

Nombre y apellidos  
Empresa  
Dirección  
Interesado en programas:

## CONTROLADORES DE DISCO

**E**nfa Ibérica, una joven empresa informática dedicada a los ordenadores **Amstrad**, ha lanzado al mercado español un controlador de disco para unidades de 5 pulgadas y 1/4, es decir, el estándar

del mercado, que respondan a la mecánica SHUGART o TEACH (más estándares, y más «gordos» aún).

Estos en el caso del 664 y, suponemos, del 6128.

Para el sufrido 464 hay que conectar el floppy de 5 1/4 a través de la unidad de 3 pulgadas y el controlador.

**Enfa Ibérica vende el aparato a 5.500 pesetas, y los puedes encontrar en:**

**Balandro 39-bis, segundo b  
28042 Madrid**

**Tel.: (91) 742 18 92 y 742 91 51**



## MAS OPCIONES IVA PARA EL PROGRAMA PLACON

**E**l reciente anuncio de actualización gratuita del manual a los usuarios registrados del programa **PLACON** sobre microordenadores **AMSTRAD**, hay que añadir otra buena noticia para los usuarios del programa de contabilidad editado por **MICROMOUSE**: la opción LIVA. Este programa opcional permite obtener el libro registro de facturas emitidas y recibidas, exigido tras la implantación del Impuesto sobre el Valor Añadido, y tras pasa los datos contables al programa **PLACON**.

Sin embargo, y a pesar del gran número creciente de usuarios del programa **PLACON**, **MICROMOUSE** no se ha dormido en los laureles obtenidos y continúa trabajando sobre su versión **AMSTRAD**. Los poseedores de un **AMSTRAD 6128** con una sola unidad de disco podrán utilizar el programa **PLACON** para llevar su contabilidad y en breve se encontrará disponible la versión del programa **PLACON** para gestorías.

**DIRECCION**  
Pza. Ciudad de Viena, 7  
Tel. 253 22 31/253 22 22  
Telex. 45316 LXTE  
2804 MADRID

## Primera PLANA

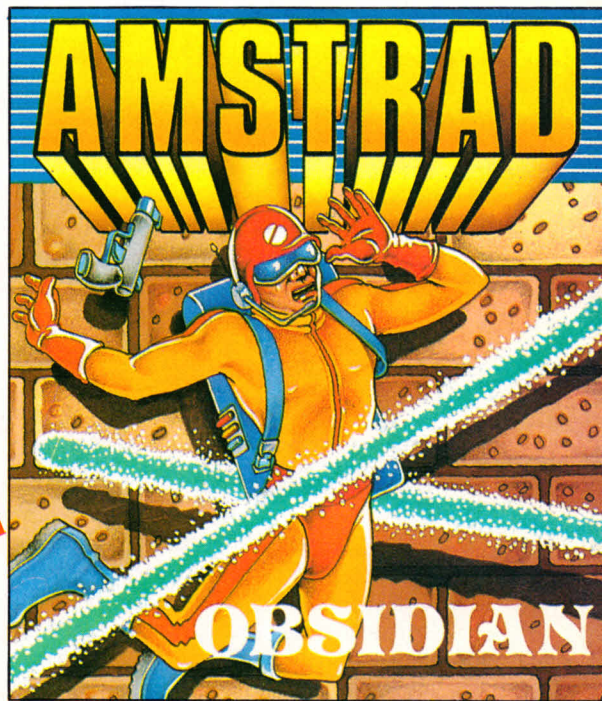
## OBSIDIAN DE ACE SOFTWARE

**Y**a está disponible en el mercado el nuevo juego de **ACE Software**, **OBSIDIAN**.

Con él nos adentraremos en una estación espacial, **OBSIDIAN**, que se haya enclavada en un asteroide y en la cual corremos el peligro de perecer debido a que se dirige irremisiblemente a un agujero negro, y aunque la nave tiene un campo de gravedad de protección, este no nos protegerá a nosotros de la destrucción. Para poder salvar esta situación debemos entrar en otro universo y buscar naves del mismo tipo que **OBSIDIAN**.

Pero después de haber logrado esto, descubrimos que una extraña radiación está destruyendo los dos sistemas de energía de la nave. Como sólo tú en la nave eres lo suficiente valiente y hábil para enfrentarte a este problema, se te ha designado para introducirte en la estación, reactivar los sistemas de energía, inicializar los motores y volver a fijar el plan de vuelo.

Juego entretenido y de habilidad distribuido por **MICROBYTE**.



# «PASE» DE INTERRUPCIONES

*Continuando por el camino iniciado en nuestro anterior artículo que nos situaba en el mundo de las instrucciones ON..., llegamos a un grupo de sentencias que va a inhibir o eliminar la posibilidad de cortar la ejecución de un programa pulsando dos veces la tecla ESC y después, en el momento que nosotros queramos, poder volver a disponer de esta facilidad.*



Le recomendamos que antes de ejecutar cualquier programa de los que vamos a analizar, tenga la precaución de grabarlos en disco o cinta, ya que pudiera ocurrir que, al no poder detenerla, sea imposible después hacerlo mediante la orden SAVE.

La ejecución de un programa escrito en **BASIC**, sobre el que no se hayan tomado medidas especiales de protección, puede detenerse pulsando una vez la tecla **ESC** e interrumpirla definitivamente si presionamos de nuevo dicha tecla una vez esté parada.

Escribamos un programa que no termine nunca de ejecutarse. Como muestra puede servirle el I que, aunque es bastante concreto, nos sirve perfectamente para ver sobre la práctica todo lo que le vamos a contar.

## Programa uno

El Programa 1 no finaliza nunca. Siguiendo su ejecución linealmente nos encontramos que en la línea 40 hay un salto incondicional (o sea, que se produce siempre, independientemente de que se cumpla o no una cierta condición) de vuelta a la línea 20.

De esta manera se repite de nuevo el proceso imprimiéndose el texto especificado en la línea 30 para después hacer un salto atrás y pasar a la línea 20.

Si se pulsa una vez la tecla **ESC**, la ejecución del programa se detiene y observará que en la pantalla aparece de nuevo el cursor de textos en una determinada posición.

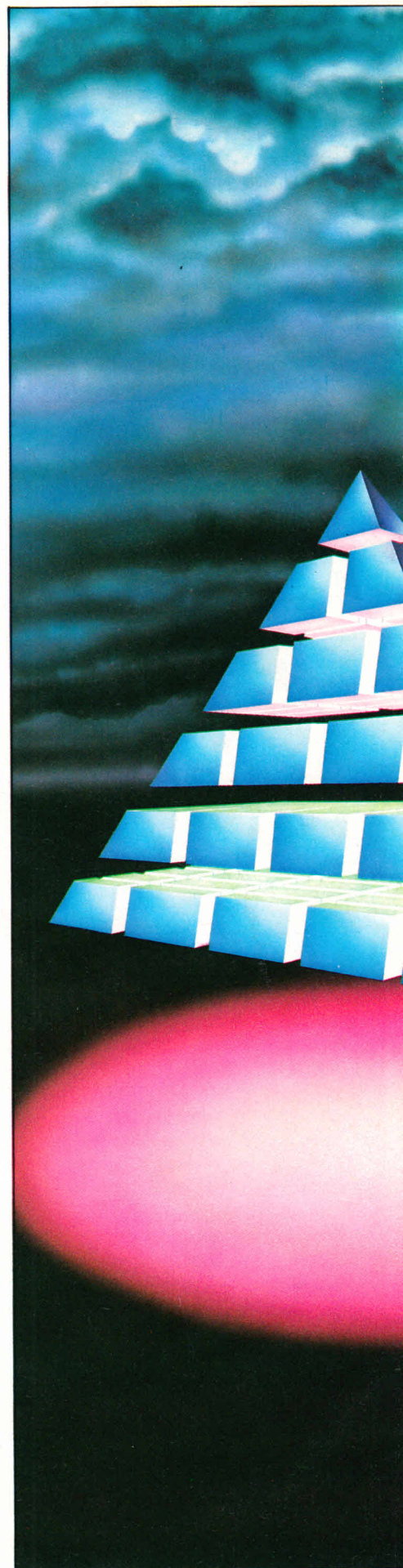
```
10 REM PROGRAMA I
20 CLS
30 PRINT"NO TERMINO NUNCA"
40 GOTO 20
```

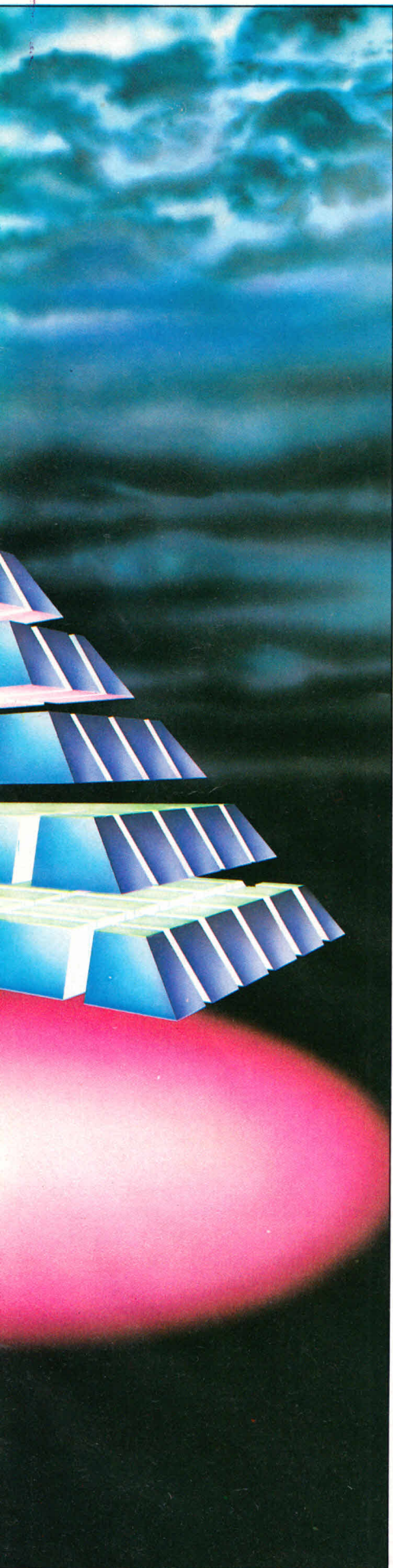
¿En qué momento aparece? Está claro que será cuando el programa se detenga. Como el bucle que estamos repitiendo continuamente es el comprendido entre las líneas 20 a 40, será después de ejecutar cualquiera de éstas, dependiendo del momento en el que pulsemos ESC.

Así por ejemplo, si estaba en la línea 20, la pantalla aparecerá totalmente limpia, ya que se ejecuta por completo la orden CLS, y el cursor se situará en la primera posición de la línea superior.

Pero si presionamos la tecla ESC cuando se está visualizando el mensaje (línea 30), en la pantalla se escribirá:

**NO TERMINO NUNCA**





completamente y el cursor quedará colocado en la primera columna de la segunda línea de texto, ya que después de imprimir el mensaje salta de línea y se coloca en el borde izquierdo. Y lo mismo ocurrirá si detenemos el programa durante la ejecución de la línea 40. ¿Alguna dificultad?

En cualquier caso podremos volver a poner en marcha el programa simplemente pulsando una línea distinta de ESC. Compruébelo con el programa del ejemplo y verá que siempre sucede así.

Cuando deseemos interrumpir definitivamente la ejecución en lugar de presionar una tecla cualquiera, tendríamos que hacerlo de nuevo con **ESC**. Ahora se abandona definitivamente el programa y en la pantalla nos aparece el mensaje:

### BREAK IN x

donde x es el número de línea donde lo hemos detenido, y a continuación, en la siguiente fila de texto, la palabra:

«Ready»

con la que el ordenador nos está indicando que está de nuevo listo p preparado para recibir cualquier nuevo comando.

En este caso si desea reanudar la ejecución del programa, no es suficiente con pulsar cualquier tecla diferente de ESC, sino que tendremos que teclear **RUN** y comenzar de nuevo. Recuerde que estaba parado totalmente.

Ahora bien, puede darse el caso de que, en vez de empezar de nuevo, queramos continuar con la ejecución del programa en el mismo punto donde lo habíamos abandonado. ¿Cómo hacerlo?

El **Amstrad**, o mejor sus creadores, son muy previsores. Pensaron en una instrucción Basic que nos lo resolviera, y de hecho la encontraron: **CONT**.

Ejecuta de nuevo el Programa 1. Deténgalo totalmente presionando dos veces la tecla ESC. Le aparecerá en la pantalla la palabra «Ready» indicándole que puede escribir un nuevo comando.

Teclee ahora la orden **CONT**. ¿Qué ocurre? Sencillamente que el micro se coloca en el punto donde había abandonado la ejecución y desde allí continúa la misma sin ningún problema.

Visto y comprendido todo esto, ¿no?, comience a escribir el Programa 2. ¿Qué ocurrirá?

## Programa dos

En esencia es muy semejante al anterior. Se trata también de un bucle sin fin en el que se imprime un texto indefinidamente.

Pero ahora hemos utilizado la sentencia **WHILE... WEND** con la única salvedad que la condición se va a cumplir indefinidamente (líneas 40 a 60).

# Primeros PASOS

```
10 REM PROGRAMA 11
20 ON BREAK CONT
30 CLS
40 WHILE -1
50 PRINT"NO PUEDES DETENERME CON ES
C"
60 WEND
```

Aparte del tipo de bucle empleado, ¿qué es lo que los diferencia? Echele una ojeada detenidamente. Le seguimos recordando la sugerencia de salvarle antes de su ejecución, no sea que luego haya problemas.

Observe que en la línea 20 hemos introducido un nuevo comando. Concretamente ha sido la instrucción:

### ON BREAK CONT

¿Qué sucede cuando se ejecuta esta línea? Teclee **RUN** para poner en marcha el programa. Aparentemente no ocurre nada nuevo o anormal. Un mensaje aparece continuamente inundándonos la pantalla debido a la repetición del bucle que anteriormente le comentábamos.

Trate ahora de interrumpir la ejecución del mismo pulsando **ESC** cuantas veces desee. Pero, ¿qué ocurre? Pues está muy claro:

### ¡NO PUEDE DETENERLO!

Estos son los efectos producidos por el nuevo comando. **ON BREAK CONT** ha imposibilitado que podamos detenerle metiéndonos, esta vez sí, en un verdadero «bucle» que no tiene fin.

La única posibilidad de hacerlo es reiniciando el ordenador o apagándolo definitivamente, por eso le hemos repetido varias veces que antes de ejecutar los programas, y debido a este tipo de **protección**, era muy conveniente asegurarse que ya estaba guardado correctamente en cinta o disco. De lo contrario, la única forma de poder conseguir un listado del mismo, sería teclearlo de nuevo y cuando sea muy largo, no nos resultará muy rentable, ¿verdad?

Entonces quedamos que **ON BREAK** le indica al ordenador que cuando detecte que queremos detener o romper (**BREAK**) la ejecución del programa, no haga caso de esta interrupción y continúe a su aire. ¡No habrá olvidado ya lo que hacía la orden **CONT**!

Pero con el **Amstrad** podemos hacer que estos efectos, que nos proporcionan una primitiva pero efectiva manera de proteger un programa, puedan no ser permanentes a lo largo de la ejecución del mismo.

Queremos decir con esto que estamos, o vamos a estar, en disposición de inhibir o habilitar estas interrupciones en las partes del programa que deseemos. Teclee el Programa 3 y sobre él lo veremos más claramente.

## Programa tres

En la línea 20 estamos activando esta protección que **prohíbe** detenciones. A continuación, después de borrar la pantalla, se visualiza un texto informativo de lo que va a pasar (línea 40) y debajo de él aparece el valor de una variable que llamaremos «**bucle**» y que varía desde 1 a 50.

Le hemos dado ese nombre porque se trata de la variable de control de un bucle FOR...NEXT que comienza en la línea 50 y termina en la 90. Su única misión es servirnos como de una especie de contador visual que nos determinará la parte de la ejecución del programa en la que no será posible interrumpirlo.

En la línea 80 encontramos otro bucle del mismo tipo, pero en este caso sólo nos sirve para retardar el cambio de valores producidos en la variable «**bucle**».

Así pues, durante todo este tiempo el programa está «protegido» contra posibles detenciones debidas a la tecla ESC. Intente hacerlo mientras el **Amstrad** está escribiendo los valores de esta especie de contador comprendidos entre 1 y 50. No se puede.

La línea 120 del programa imprime otro mensaje en la pantalla. Ahora nos informa que ya sí permite interrupciones, ¿será cierto?, y el bucle de la línea 130, también sin fin, tiene como única finalidad que el programa no termine nunca, ya que se trata de un WHILE...WEND en el que la evaluación de la condición es siempre cierta y por tanto el bucle gira sobre sí mismo sin hacer nada. ¿Cuál es el cuerpo del mismo?

Probemos pulsar de nuevo la tecla ESC. Si lo hacemos una vez en esta ocasión, la ejecución del programa sí se detiene, y si la presionamos otra vez, se interrumpe definitivamente y se abandona por completo.

En la pantalla nos aparece de nuevo el mensaje:

BREAK IN 130

```
10 REM PROGRAMA III
20 ON BREAK CONT
30 CLS
40 PRINT " NO ME PUEDES INTERRUPTIR
  HASTA QUE      LLEGUE
  A 50"
50 FOR bucle=1 TO 50
60 LOCATE 19,4
70 PRINT bucle
80 FOR retardo=1 TO 200:NEXT retardo
90 NEXT bucle
100 ON BREAK STOP
110 LOCATE 8,7
120 PRINT"AHORA YA SI PUEDES HACERLO"
130 WHILE -1:WEND
```



indicándonos el lugar donde se ha producido la parada (línea 130) y a continuación:

Ready

¿Por qué ahora sí podemos interrumpir la ejecución? Observe la línea 100 del Programa. En ella descubrimos una nueva instrucción Basic:

ON BREAK STOP

Con esta orden conseguimos anular los efectos producidos por el ON BREAK CONT de la 20. Por tanto no podremos detener la marcha del mismo mientras se estén ejecutando las líneas comprendidas entre la 20 y la 90, pero cuando pase por la 100 con:

ON BREAK STOP

permite que el programa pueda ser de nuevo interrumpido con la tecla ESC.

Puede convenirnos, en algún momento, que el ordenador avise al «atrevido» que haya intentado parar un programa de lo que está queriendo hacer y además impedirselo. Escriba ahora el Programa 4 y sávelo antes de ejecutarlo.

## Programa cuatro

Hemos introducido en el nuevo comando en la línea 20:

ON BREAK GOSUB

Hasta el momento hemos visto que podíamos hacer que el ordenador permitiera o no que el usuario del mismo (o sea, nosotros) pudiera detener la ejecución de un programa o le estuviera totalmente prohibido hacerlo. Empleando convenientemente las instrucciones ON BREAK CONT y ON BREAK STOP lo conseguimos con toda facilidad.

El Programa 4 es, de nuevo, un bucle sin fin que nos escribirá continuamente un mismo

mensaje en la pantalla. Este es el efecto que producen las líneas 40 y 50.

Ante este **desafío** intentemos interrumpirlo pulsando dos veces ESC, a ver qué pasa.

Ahora ocurre algo totalmente diferente a lo que sucedía en los programas anteriores. Por un lado vemos que el hecho de presionar ESC una vez sí produce efecto sobre la ejecución: «la detiene». Esto es lo que estábamos acostumbrados a ver antes de comenzar a leer este artículo.

```
10 REM PROGRAMA IV
20 ON BREAK GOSUB 100
30 CLS
40 PRINT"TRATA DE DETENERME"
50 GOTO 40
100 CLS
110 PRINT" AL TRATAR DE DETENERME
  EJECUTO LA      RUTINA DE LAS LI
  NEAS 100 A 130"
120 FOR RETARDO=1 TO 1000:NEXT RETARDO
130 RETURN
```

Pero «dele» de nuevo a ESC. ¿Qué nueva sorpresa nos espera?

No ocurre lo mismo que con el Programa 1, donde se abandonaba definitivamente la ejecución, ni como en el Programa 2, en el que se hacía caso omiso a la tecla.

En esta ocasión, cuando presionamos dos veces ESC, salta a ejecutar la subrutina comprendida que comienza en la línea 100. Este es el efecto que produce la orden:

ON BREAK GOSUB

Cuando se trate de «**romper**» —BREAK— la marcha de un programa se va a recorrer una subrutina (GOSUB) que comienza en el número de línea que hay al final de la 20 — justo detrás de GOSUB.

Naturalmente, tras ejecutar la rutina y una vez hayamos llegado a la línea 130, el RETURN que allí se encuentra «retorna» o devuelve el control al punto en el que fue interrumpido el programa, o sea, entre las líneas 40 y 50.

Tantas veces como intentemos detenerla veremos en la pantalla el **mensaje** que nos indica que se ha producido el salto a la 100 para luego continuar donde lo habíamos dejado.

¿Cómo salir de esta encerrona?

La instrucción ON BREAK GOSUB, al igual que ON BREAK CONT, se anulan mediante la orden ON BREAK STOP. ¿Recuerdas?

Con el Programa 5 vamos a ver la manera de hacerlo. Intente interrumpirle en los diferentes momentos que aparezcan en la pantalla.

## Programa cinco

Es evidente que tras pasar por la línea 20, si pulsamos dos veces la tecla **ESC** mientras nos aparece en la pantalla «MOMENTO 1», estamos obligando al ordenador a que ejecute la subrutina que comienza en la línea 100.

Allí, lo primero que hacemos es desactivar la prohibición de interrupción mediante el:

ON BREAK STOP

de la propia línea 100.

A continuación se visualiza el mensaje indicativo del lugar donde estamos —120— y cambiamos en la 130 el valor de la variable «momento».

```
10 REM PROGRAMA V
20 ON BREAK GOSUB 100
30 CLS
40 momento=1
50 PRINT "MOMENTO"; momento
60 GOTO 50
100 ON BREAK STOP
110 CLS
120 PRINT "ROUTINA PERMITIR RUPTURA"
130 momento=2
140 FOR retardo=1 TO 1000:NEXT retardo
150 RETURN
```

Tras el bucle de retardo de la línea 140, volvemos al punto donde la ejecución fue interrumpida. Pero ahora el «momento» es otro y así nos lo dice el texto que aparece en la pantalla.

Si en este momento, o sea en el 2, pulsamos dos veces ESC, habremos «roto» la ejecución del programa ya que desactivamos la prohibición de hacerlo, y de nuevo veremos el correspondiente mensaje:

BREAK IN...

seguido de:

«Ready»

Como habrá podido comprender, todo esto es debido a que al ejecutarse la rutina que comienza en la línea 100, la instrucción ON BREAK GOSUB acaba con los efectos que produce el ON BREAK STOP de la 20. Por tanto, al pulsar ahora ESC ya no saltamos de nuevo a dicha subrutina, sino que abandonamos definitivamente la ejecución.

# Primeros PASOS

¿Qué pasa si intentamos parar entre las líneas 100 a 150? Es el momento en el que programa está sobre la propia subrutina. Pues, ¡inténtelo, hombre!

¡Vaya!, parece que no ocurre nada. ¿Es que no funcionará con ella el efecto de ON GOSUB STOP?

Pues parece que así es. Solamente lo notaremos, o será apreciable, cuando estemos de vuelta de esta rutina al lugar donde se intentó detener el programa. Así que, ¡cuidado!

¿Qué le parece todo este nuevo conjunto de instrucciones? Le dotan al **Amstrad** de toda una potencia que ya quisieran para sí alguno que otro de los grandes ordenadores que circulan por el mundo.

Y seguro que usted, con su imaginación y ganas de trabajar, será capaz de producir con ellas (y otras más, claro) pequeñas obras de arte que harán palidecer de envidia a más de un programa avezado. Sólo nos queda repetirle el consejo de salvar los programas antes de ejecutarlos. Nos lo agradecerá.

Y en la próxima ocasión más cosas. ¡Hasta ella!

## I A N KEY - COVER



- Ian Key Cover, es el cubreteclados ideal para el teclado de su Amstrad 6128.
- Protección firme del teclado (realizado con resistente metacrilato transparente de una sola pieza).
- Ajuste perfecto en su ordenador con suave precisión y cobertura total.



- Protección continua contra elementos externos (derrames de líquidos, polvos, golpes).
- Resguardo permanente contra el polvo y la polución (el 50% de las averías de teclado son por este motivo).

**Ian Key Cover para amstrad 6128... 2.480 pts. (Iva incluido).**

## COMPUPRO FLOPPY DISK DRIVE



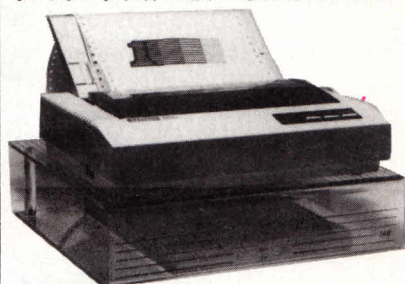
- Altísima calidad.
- Silenciosos, ágiles y manejables.
- Conectable a su Amstrad 6128 directamente a través del cable y conector (incluido en el precio).

— Completamente compatible con todos sus programas y diskettes de 3" para su Amstrad 6128.

— Compupro, líder en la fabricación de floppys para Apple, Metamorphic, PC, Dunross PC-XT y ahora Amstrad.

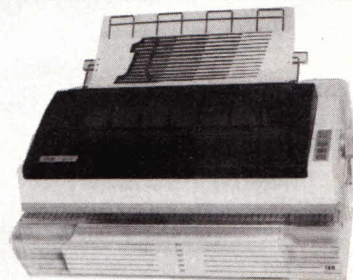
**Compupro 3" disk drive + conector... 24.000 pts.**  
**First S.A. Importador de Compupro.**

## I A N PRINTER STAND 80



- Ian Printer Stand, es el soporte ideal para toda impresora.
- Soporte del peso de su impresora (realizado con resistente metacrilato transparente de una sola pieza).
- Indicador óptico del número de unidades existentes de su formulario específico (con la numeración oportuna en la parte frontal de Ian Printer Stand).

## I A N PRINTER STAND 132



- Ahorro de espacio que Ud. precisa para su mesa o escritorio.

Ian Printer Stand 80 Col... 4.424 pts. (Iva incluido).  
Ian Printer Stand 132 Col... 5.880 pts. (Iva incluido).  
First S.A. Distribuidor exclusivo para España de Ian.

Catálogo Amstrad: mande 500 pts. en sellos de correo y especifique el tipo de catálogo.  
Venta por Correo: Precio total = precio artículo + gastos envío. Gastos envío: Ian Key Cover 500 pts., Ian Printer Stand, 800 pts., Compupro no tiene. Mande giro postal o telegráfico, o cheque confirmado.  
Pedidos por importe superior a 25.000 pts. no tienen gastos de envío. Contrareembolso: precio total = precio artículo + gastos de contrareembolso. Gastos de contrareembolso: son el 4 por 100 del precio del producto. Instrucciones: A) mande el 30 por 100 del importe del precio total, en sellos de correo con su pedido. B) la diferencia se la cobrarán con la entrega de la mercancía en su domicilio. Todos los envíos se realizan por correo certificado al domicilio del comprador.

**FIRST S.A.**



T.M. Valencia, 195, TDA.  
08011-Barcelona  
☎ 323 03 90  
Télex: 53947 FIRS E



# ORDENADOR

## SERIE CPC

### UNIDAD CENTRAL. MEMORIAS

- Microprocesador Z80A - 64K RAM ampliables - 32K ROM ampliables
- **TECLADO.** • Teclado profesional con 74 teclas en 3 bloques - Hasta 32 teclas programables - Teclado redefinible
- **PANTALLA** • Monitor RGB verde (12") o color (14")

	Normal	Alta Res.	Multicolor
Col x líneas	40 x 25	80 x 25	20 x 25
Colores	4 de 27	2 de 27	16 de 27
Puntos	320 x 200	640 x 200	160 x 2

- Se pueden definir hasta 8 ventanas de texto y 1 de gráficos • **SONIDO**
- 3 canales de 8 octavas moduladas independientemente - Altavoz interno regulable - Salida estéreo • **BASIC**
- Locomotive BASIC ampliado en ROM - Incluye los comandos AFTER y EVERY para control de interrupciones

## AMSTRAD CPC 464

- **CASSETTE** • Cassette incorporada con velocidad de grabación (1 ó 2 Kbaudios) controlada desde Basic • **CONECTORES**
- Bus PCB multiuso, Unidad de Disco exterior, paralelo Centronics, salida estéreo, joystick, lápiz óptico, etc.
- **SUMINISTRO** • Ordenador con monitor verde o color - 8 cassettes con programas - Libro "Guía de Referencia BASIC para el programador" - Manual en castellano - Garantía Oficial AMSTRAD ESPAÑA. •

**TODO POR** 59.900 Pts. (monitor verde)  
90.900 Pts. (monitor color)

## AMSTRAD CPC 6128

- **UNIDAD DE DISCO** • Unidad incorporada para disco de 3" con 180K por cara • **SISTEMAS OPERATIVOS**
- AMSDOS, CP M 2.2, CP M Plus (3.0)
- **CONECTORES** • Bus PCB multiuso, paralelo Centronics, cassette exterior, 2.ª Unidad de Disco, salida estéreo, joysticks, lápiz óptico, etc.
- **SUMINISTRO** • Ordenador con monitor verde o color - Disco con CP M 2.2 y lenguaje DR. LOGO - Disco con CP M Plus y utilidades - Disco con 6 programas de obsequio - Manual en castellano - Garantía Oficial AMSTRAD ESPAÑA.

**TODO POR** 84.900 Pts. (monitor verde)  
119.900 Pts. (monitor color)

# PCW - 8256

# AMSTRAD CPC - 6128



# ES AMSTRAD

# Increíble!!

## AMSTRAD PCW 8256

### UNIDAD CENTRAL. MEMORIAS

- Microprocesador Z80A - 256K RAM de las que 112K se utilizan como disco RAM
- **TECLADO** • Teclado profesional en castellano (ñ, acento...) de 82 teclas
- **PANTALLA** • Monitor verde de alta resolución - 90 columnas x 32 líneas de texto • **UNIDAD DE DISCO** • Disco de 3" y 173K por cara - Opcionalmente, 2.ª Unidad de Disco de 1 Mbyte integrable
- **SISTEMA OPERATIVO** • CP / M Plus de Digital Research • **IMPRESORA** • Alta calidad (NLQ) a 20 c.p.s. - Calidad estándar a 90 c.p.s. - Papel continuo u hojas sueltas - Alineación automática del papel - Caracteres normales, comprimidos, expandidos, control del paso de letra (normal, cursiva, negrita, subíndices, superíndices, subrayado, etc).
- **OPCIONES** • Kit de Ampliación a 512K RAM y 2.ª Unidad de Disco - Interface Serie RS 232C y paralelo

Centronics • **SUMINISTRO** • Ordenador completo con teclado, pantalla, Unidad de Disco e Impresora - Discos con el procesador de Texto LocoScript, CP / M Plus, Mallard, BASIC, DR. LOGO y diversas utilidades - Manuales en castellano - Garantía Oficial AMSTRAD ESPAÑA.

**TODO POR 129.900 Pts.**



Los más prestigiosos paquetes de **Software Profesional**, en formato AMSTRAD... a "precios AMSTRAD"

Existe también la versión **PCW 8512** con **512K RAM** y la 2.ª Unidad de Disco de 1 Mbyte incorporada. **PVP. 169.900 Pts.**  
\* El **PCW 8256** puede utilizarse como terminal y en comunicaciones.

El I.V.A. no está incluido en los precios.

**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 unidades de discos).

# AMSTRAD

ESPAÑA

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

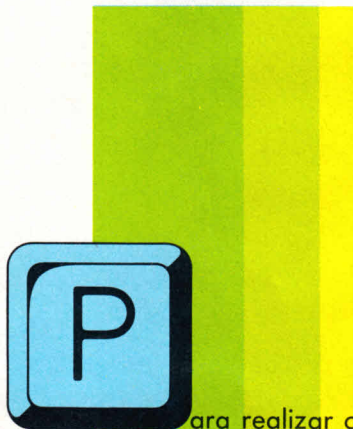
Delegación Cataluña: Tarragona, 110 - Tel. 325 10 58.  
08015 BARCELONA

MOVIMIENTO

# ¡SE MUEVE!

**En anteriores capítulos hemos visto cómo realizar la impresión en cualquier parte de la pantalla, así como la forma de detectar la pulsación de cualquier tecla.**

**Hoy hemos realizado un programa que combina las dos cosas; se trata del movimiento en pantalla de un carácter gráfico. Este se moverá a un lado u otro según la tecla del cursor que se haya pulsado en ese momento.**



Para realizar cualquier programa de este tipo, en primer lugar se deben efectuar las inicializaciones correspondientes. Dado que únicamente se va a producir el movimiento de un carácter, deberemos definir las coordenadas iniciales del gráfico en pantalla.

Para poder almacenar esas variables en código máquina, se debe reservar un espacio de memoria que sea capaz de contenerlas sin que se puedan perder en cualquier momento.

Dado que el único registro simple que puede trabajar con espacios de memoria es el acumulador, utilizaremos éste para colocar las variables anteriormente mencionadas en dichos espacios de memoria.

Para ello efectuamos las siguientes operaciones:

```
LD A,12  
LD (POSX),A
```

en este momento hemos almacenado la coordenada horizontal del gráfico a imprimir en la posición de memoria indicada por la variable POSX.

Seguidamente deberemos almacenar la posición vertical, para ello hacemos:

```
LD A,20  
LD (POSY),A
```

Para que nuestro programa tenga conocimiento de cuáles son las posiciones de memoria utilizadas para dicho fin, deberemos indicárselo de la forma siguiente:

```
POSX:DEFB 12  
POSY:DEFB 20
```

De esta forma dichas direcciones de memoria nos indicarán en cada momento cuál es la posición del carácter que se está moviendo en pantalla.

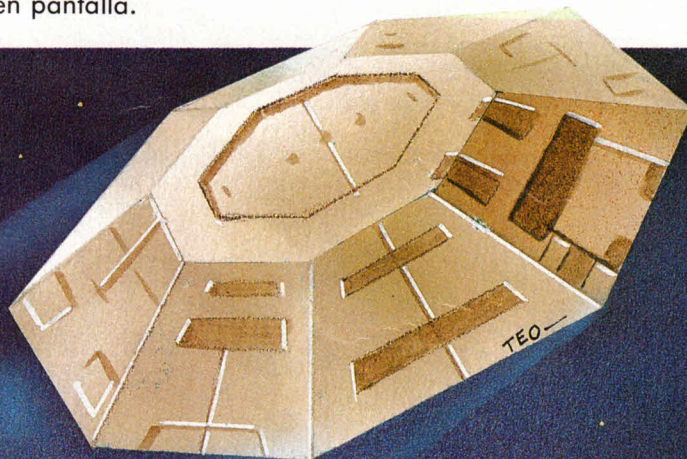
Seguidamente nos encontramos con el bucle principal, que será el encargado del control del programa, llamando a las diferentes rutinas que se utilicen.

En este caso el bucle únicamente contiene la llamada a la rutina de teclas que es la que realiza la función de desplazamiento de pantalla de nuestro gráfico.

De esta forma, dicho bucle hace una llamada a la rutina TECLAS y, una vez que retorne, se hace un salto relativo para que se vuelva a efectuar la llamada:

```
BUCLE:CALL TECLAS  
JR BUCLE
```

A continuación viene la lectura de teclado, que es la encargada de leer las teclas del cursor, y enviar el pro-



grama a diferentes rutinas según la tecla que se haya pulsado en cada momento.

De esta forma cuando se pulse «cursor derecha» se envía el programa a la rutina DERECH. De la misma forma cuando se pulse cualquier otra tecla del cursor, se enviará el programa a la rutina correspondiente.

Vamos a ver ahora cada una de las rutinas que se encargan del movimiento propiamente dicho. Con la primera que nos encontramos es con la rutina DERECH.

Dicha rutina se encarga de incrementar la coordenada horizontal para que nuestro carácter se desplace horizontalmente hacia la derecha.

Al entrar en dicha rutina lo primero que hacemos es la comprobación de la anterior coordenada horizontal, para ello cargamos en el acumulador su valor y lo comparamos con 39. Dicha comprobación se efectúa para que el gráfico no se salga por la parte derecha de la pantalla.

```
LD A,(POSY)
CP 39
RET Z
```

Si su valor es 39, retorna al programa principal sin incrementar las coordenadas. Si por el contrario dicha coordenada es menor, entonces se incrementará.

Antes de actualizar dicha variable, deberemos borrar de pantalla el carácter, para lo cual se realizará una llamada a la rutina BORRA.

```
CALL BORRA
```

A continuación tomamos de memoria el valor de la variable para incrementarlo, seguidamente se vuel-

ve a guardar en dicha posición en memoria:

```
LD A,(POSY)
INC A
LD (POSY),A
```

Sólo queda ya imprimir el carácter en las coordenadas actuales de pantallas, para lo cual realizamos una llamada a la rutina PINTAR y retorna al programa principal.

```
CALL PINTAR
```

```
RET
```

Seguidamente se encuentra la rutina IZQUI que es la encargada de decrementar la posición horizontal del carácter a imprimir. Antes de entrar en esta rutina se compara la coordenada con el valor 2, ya que ésta es la última posición de pantalla en la cual se desea imprimir.

```
LD A,(POSY)
CP 2
RET Z
```

Si el valor de dicha coordenada es 2, se retorna al programa principal, de lo contrario se decrementa la coordenada y se imprime el carácter en la nueva posición.

Las rutinas encargadas de hacer subir o bajar a nuestro gráfico son ABAJO y ARRIB, que se encargan de actualizar los valores de la coordenada vertical.

Al entrar en ellas se compara si la posición vertical del gráfico está en el límite de la pantalla en el que deseamos imprimir; si es así, la rutina retorna sin actualizar el valor de dicha coordenada, de lo contrario, ésta es actualizada y se imprime el carácter gráfico en la nueva posición de pantalla.

## Código MAQUINA

Por último nos encontramos con las rutinas encargadas de borrar y pintar el gráfico de la pantalla.

La primera de ellas se encarga de borrar el gráfico de la anterior posición de la pantalla, para ello se llama en primer lugar a una rutina que coloca en el registro HL las coordenadas de nuestro carácter, para luego llamar a la rutina del firmware que coloca el cursor en dicha posición:

```
CALL PONCOR
CALL #BB75
```

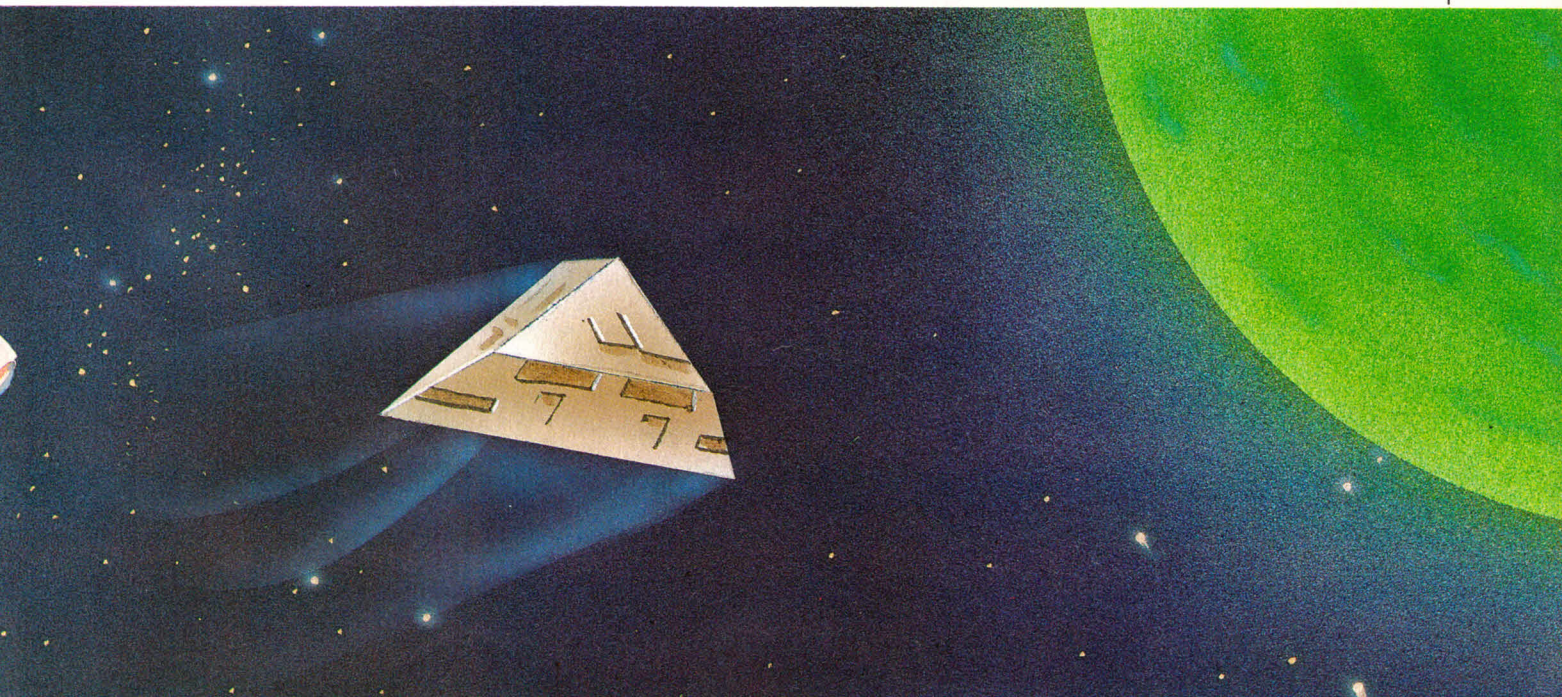
a continuación se debe imprimir en dicha posición de pantalla el carácter «espacio» cuyo código ASCII es 32,

```
LD A,32
CALL #BB5A
RET
```

una vez hecho esto se retorna al bucle principal.

La rutina de impresión en pantalla funciona de forma similar a la anterior, pero esta vez se toman las coordenadas actuales y se imprime en pantalla el carácter 248, que es el que se mueve por la pantalla.

Para ello se llama a la rutina PONCOR que coloca las coordenadas actuales en el registro HL, para luego



# PROGRAMA BASIC

```

10 REM ***INICIALIZACION***
20 REM '
30 MODE 1
40 x=20:y=12:LOCATE x,y:PRINT CHR$(
248)
50 REM '
60 REM ***BUCLE PRINCIPAL***
70 REM '
80 GOSUB 130
90 GOTO 80
100 REM '
110 REM ***LECTURA DE TECLADO***
120 REM '
130 IF INKEY(1)=0 THEN GOSUB 210:RE
TURN
140 IF INKEY(8)=0 THEN GOSUB 280:RE
TURN
150 IF INKEY(0)=0 THEN GOSUB 350:RE
TURN
160 IF INKEY(2)=0 THEN GOSUB 420:RE
TURN
170 RETURN
180 REM '
190 REM ***INCREMENTO DERECHA***
200 REM '
210 IF x=39 THEN RETURN
220 LOCATE x,y:PRINT CHR$(32)
230 x=x+1:LOCATE x,y:PRINT CHR$(248
)
240 RETURN
250 REM '
260 REM ***INCREMENTO IZQUIERDA***
270 REM '
280 IF x=2 THEN RETURN
290 LOCATE x,y:PRINT CHR$(32)
300 x=x-1:LOCATE x,y:PRINT CHR$(248
)
310 RETURN
320 REM '
330 REM ***INCREMENTO ARRIBA***
340 REM '
350 IF y=2 THEN RETURN
360 LOCATE x,y:PRINT CHR$(32)
370 y=y-1:LOCATE x,y:PRINT CHR$(248
)
380 RETURN
390 REM '
400 REM ***INCREMENTO ABAJO***
410 REM '
420 IF y=24 THEN RETURN
430 LOCATE x,y:PRINT CHR$(32)
440 y=y+1:LOCATE x,y:PRINT CHR$(248
)
450 RETURN
    
```

llamar a la rutina del firmware que pone el cursor en dicha posición.

Una vez hecho esto se carga en el acumulador el valor del carácter a imprimir y se llama a la rutina encargada de la impresión (#BB5A).

La rutina que coloca las coordenadas en el registro doble HL, lo que hace es tomar la posición de pantalla de las variables POSX y POSY, y las asigna a los registros H y L para luego retornar:

```

LD A, (POSX)
LD L,A
LD A,(POSY)
LD H,A
RET
    
```

Una vez hayamos ejecutados los programas Basic y máquina, podremos comprobar la rapidez de uno y otro, aunque la velocidad del código máquina se aprecia mucho mejor cuando no se utilizan rutinas del sistema (es decir del firmware), aunque para realizar un programa sin la utilización de dichas rutinas se debe ser un experto en la programación.

## LISTADO DESENSAMBLADO

10	ORG #A000	580	CP 2
20	;	590	RET Z
30	; INICIALIZACION	600	CALL BORRA
40	;	610	LD A, (POSY)
50	LD A,1	620	DEC A
60	CALL #BC0E	630	LD (POSY),A
70	LD A,12	640	CALL PINTA
80	LD (POSX),A	650	RET
90	LD A,20	660	;
100	LD (POSY),A	670	; INCREMENTO ARRIBA
110	CALL PINTA	680	;
120	POSX: DEFB 12	690	ARRIB: LD A, (POSX)
130	POSY: DEFB 20	700	CP 2
140	;	710	RET Z
150	; BUCLE PRINCIPAL	720	CALL BORRA
160	;	730	LD A, (POSX)
170	BUCLE: CALL TECLAS	740	DEC A
180	JR BUCLE	750	LD (POSX),A
190	;	760	CALL PINTA
200	; LECTURA DE TECLADO	770	RET
210	;	780	;
220	TECLAS: LD A,1	790	; INCREMENTO ABAJO
230	CALL #BB1E	800	;
240	JR Z,PAST1	810	ABAJO: LD A, (POSX)
250	CALL DERECH	820	CP 24
260	RET	830	RET Z
270	PAST1: LD A,B	840	CALL BORRA
280	CALL #BB1E	850	LD A, (POSX)
290	JR Z,PAST2	860	INC A
300	CALL IZQUI	870	LD (POSX),A
310	RET	880	CALL PINTA
320	PAST2: XOR A	890	RET
330	CALL #BB1E	900	;
340	JR Z,PAST3	910	; Rutina de impresion
350	CALL ARRIB	920	;
360	RET	930	PINTA: CALL PONCOR
370	PAST3: LD A,2	940	CALL #BB75
380	CALL #BB1E	950	LD A,248
390	RET Z	960	CALL #BB5A
400	CALL ABAJO	970	RET
410	RET	980	;
420	;	990	; Rutina de borrado
430	; INCREMENTO DERECHA	1000	;
440	;	1010	BORRA: CALL PONCOR
450	DERECH: LD A, (POSY)	1020	CALL #BB75
460	CP 39	1030	LD A,32
470	RET Z	1040	CALL #BB5A
480	CALL BORRA	1050	RET
490	LD A, (POSY)	1060	;
500	INC A	1070	; COLOCA COORDENADAS EN EL
510	LD (POSY),A	1080	; REGISTRO DOBLE HL
520	CALL PINTA	1090	;
530	RET	1100	PONCOR: LD A, (POSX)
540	;	1110	LD L,A
550	; INCREMENTO IZQUIERDA	1120	LD A, (POSY)
560	;	1130	LD H,A
570	IZQUI: LD A, (POSY)	1140	RET

## LISTA DE VARIABLES

Pass 2 errors: 00

ABAJO	A07F	ARRIB	A06B	BORRA	A09F
BUCLE	A014	DERECH	A043	IZQUI	A057
PAST1	A024	PAST2	A02F	PAST3	A039
PINTA	A093	PONCOR	A0AB	POSX	A012
POSY	A013	TECLAS	A019		

Table used: 182 from 264

## PROGRAMA CARGADOR

```

10 FOR N=&A000 TO &A0B4
20 READ A:SUMA=SUMA+A
30 POKE N,A
40 NEXT
50 IF SUMA<>&55CB THEN PRINT "ERROR
EN DATAS"
60 DATA 62,1,205,14,188,62,12
70 DATA 50,18,160,62,20,50,19
80 DATA 160,205,147,160,12,20,205
90 DATA 25,160,24,251,62,1,205
100 DATA 30,187,40,4,205,67,160
110 DATA 201,62,8,205,30,187,40
120 DATA 4,205,87,160,201,175,205
130 DATA 30,187,40,4,205,107,160
140 DATA 201,62,2,205,30,187,200
150 DATA 205,127,160,201,58,19,160
160 DATA 254,39,200,205,159,160,58
170 DATA 19,160,60,50,19,160,205
180 DATA 147,160,201,58,19,160,254
190 DATA 2,200,205,159,160,58,19
200 DATA 160,61,50,19,160,205,147
210 DATA 160,201,58,18,160,254,2
220 DATA 200,205,159,160,58,18,160
230 DATA 61,50,18,160,205,147,160
240 DATA 201,58,18,160,254,24,200
250 DATA 205,159,160,58,18,160,60
260 DATA 50,18,160,205,147,160,201
270 DATA 205,171,160,205,117,187,62
280 DATA 248,205,90,187,201,205,171
290 DATA 160,205,117,187,62,32,205
300 DATA 90,187,201,58,18,160,111
310 DATA 58,19,160,103,201,0,0
    
```

# ORDEMANIA SOFT

*por fin..*

CONTABILIDAD

No. ASIENTO.: 1      FECHA.: 09/06/86      MANUAL

# GESPACK

Paquete integrado de gestión que le permite a Vd., de forma fácil y optimizando el tiempo, la gestión global de su empresa.

Este paquete incorpora los programas de CONTABILIDAD, FACTURACION y CONTROL DE STOCKS además de un programa de CONTROL DE PEDIDOS y todo de forma interactiva. Controle su empresa por sólo **29.900 pts.**

## Contabilidad

Contabilidad de fácil manejo y de gran potencia que permite trabajar con cuentas de hasta cuatro niveles, con capacidad según diskette de 500/1.000 cuentas y de 2.000/10.000 asientos.

Permite modificar o dar de baja apuntes ya integrados en el Mayor, programaciones de cierres, ficheros de Contabilidad y Cuenta de Explotación, ejecución de balances comparativos, reenlazándolos por meses, clave acceso restringido, etc...

**19.900 ptas.**

## Facturación

Programa de gran agilidad y rapidez que incorpora el Control de Clientes, con gran capacidad de datos, Artículos, Albaranes, Facturas y recibos.

Generación automática y manual de documentos, valoración, a voluntad de los albaranes, todo tipo de listado, incluido el del IVA de las facturas emitidas para la declaración de Hacienda, etc...

**15.500 ptas.**

## Control de stocks

Gran capacidad de datos, le permitirán a Vd. llevar con claridad y sencillez el control de su stock.

El programa le permite llevar un libro de entradas/salidas, reorganizarlo, hacer listados de stocks..., le avisará de los límites de stocks, mínimo y máximo por artículo, etc...

Todo para la llevanza de su almacén.

**14.900 ptas.**

disponibles para: PCW 8256  
PCW 8512

Asimismo, Contabilidad disponible para CPC 664/6128 **9.900 ptas.**

**OFITES INFORMATICA**  
**DISTRIBUIDOR EXCLUSIVO DE ORDEMANIA SOFT**

● IVA NO INCLUIDO

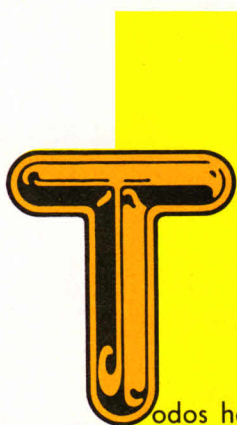
Condiciones especiales para distribuidores

Si tiene alguna dificultad en obtenerlos dirijase a

**Ofites**  
Informática  
Avda. Isabel II, 16 - 8º  
Tels. 455544 - 455533  
Télex 36698  
20011 SAN SEBASTIAN

# THE MUSIC SYSTEM CINTA/DISCO

*¿Es posible imaginar que nuestro ordenador pueda producir desde moderna música tecno, hasta la más selecta música clásica? The Music System no sólo la interpreta, sino que además, nos hace la partitura.*

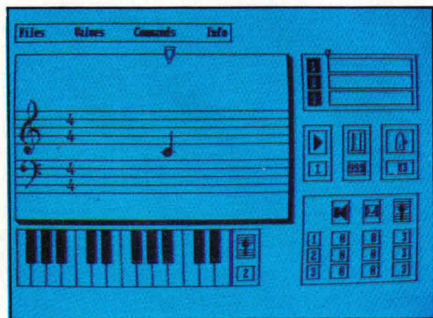


Todos hemos visto esos vídeos musicales en los que en la sección de teclados aparece un ordenador (generalmente un Macintosh), y en los que la música tiene un inconfundible carácter futurista.

Podríamos pensar que la presencia del ordenador es simplemente una anécdota o un mero adorno, pero la verdad es que es el principal responsable de que los teclados suenen así.

Gracias a las posibilidades de **síntesis** del sonido, los teclistas pueden generar cualquier tipo de sonido imaginable, e introducirlo en los teclados para su ejecución.

Con ello podemos emular los sonidos de los teclados de **Vangelis**, de **Jean Michael Jarre**, **Maniobras Orquestales** y demás grupos en la cresta de la música de actualidad.



Sólo con tener el software adecuado esto es posible. Pero hasta la fecha los poseedores de un **Amstrad** no teníamos la posibilidad de introducirnos en la música electrónica.

## Software para hacer música

Gracias a la aparición de **The Music System** ahora disponemos de un potente programa de desarrollo musical.

Con este programa tenemos a nuestra disposición un amplio número de posibilidades musicales, que no solamente se limitan a hacer que suene el teclado del ordenador como si de un órgano se tratara, sino que añade interesantes funciones que le hacen francamente interesante para el conocimiento de la música.

The Music System abarca simultáneamente los campos de interpretación, síntesis de sonido y composición musical, constituyendo la obra más completa de software dedicada a este arte.

A la hora de interpretar melodías, las dos líneas superiores del teclado constituyen un órgano monofónico de dos octavas, con las cuales podemos tocar a nuestras anchas.

Pero no sólo hemos de limitarnos a la música monofónica (una melodía solamente), también podemos hacer música polifónica (tres melodías sonando simultáneamente).

Para poder crear ésta, nuestro órgano es capaz de grabar melodías



distintas en cada una de las tres pistas de la memoria de sonido, las cuales se pueden escuchar por separado, o mezclar a nuestra conveniencia.

Podemos realizar un **play back**, en el cual una de las melodías es el contenido de la pista elegida y la melodía solista la producimos nosotros desde el teclado.

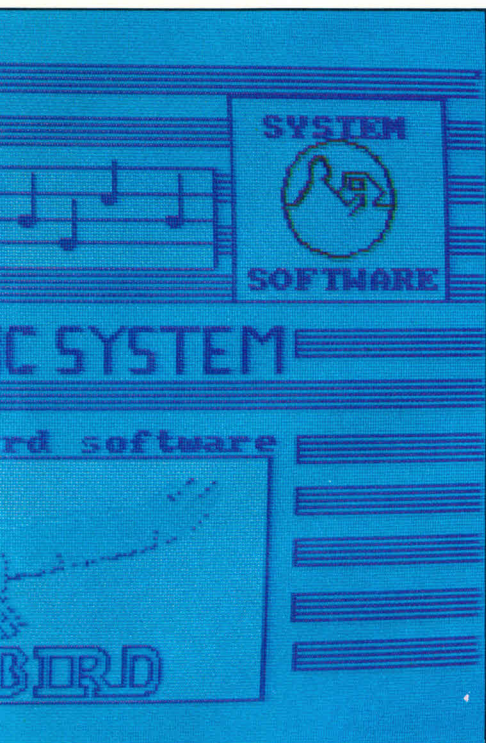
Otra ventaja que apreciarán sobre todo los poseedores de disco, es la de poder grabar nuestras composiciones o la de cargar el repertorio de melodías de demostración que contiene el programa, la audición de las mismas nos da una idea de las maravillas que se pueden realizar con esta pieza de software.

## Más que interpretación

Si las posibilidades de interpretación son importantes, la parte más espectacular del programa es la dedicada a la transcripción de partituras.

Al mismo tiempo que interpretamos nuestra obra maestra, podemos ver las notas aparecer en el **pentagrama**, obteniendo de forma automática la partitura de lo que estamos tocando.

Para los amantes del pentagrama y la composición, podemos recorrer el camino a la inversa, escribiendo la música directamente en pantalla, como si de una partitura se tratase, pu-



la comunicación entre él y el usuario se realiza de la forma más fácil posible.

The Music System utiliza la forma más evolucionada de comunicación existente hasta nuestros días, la basada en la utilización de iconos.

Por medio de éstos, la utilización de diversas opciones resulta inmediata, sin tener que recurrir a un interminable repertorio de teclas de control que no harían más que compliarnos la vida.

El manual de instrucciones constituye una pieza clave para aprovechar al máximo las posibilidades del programa, con una extensión de 47 páginas se nos explican detalladamente todas las opciones del mismo y cómo explotarlas a conciencia.

Pero no todo iba a ser un campo de rosas: el único fallo que podemos echarle en cara a este programa es, que las 47 páginas de instrucciones están escritas en inglés, por lo cual muchos de los españoles que aún no están integrados plenamente en la Comunidad Económica Europea no se enterarán de nada.

### La evolución del programa

The Music System es el resultado del trabajo realizado a lo largo de más de dos años por un equipo experto en el campo de la música.

El primer antecedente fue el Music Editor, programa creado por Philip Black para el BBC micro que apareció en abril de 1983.

# Mister JOYSTICK

En agosto del mismo año, System comercializa un programa realizado por Geoffry Ellis llamado The Envelope Generator.

En octubre de ese año, Adrian Boot de Island Logic se pone en contacto con System para el desarrollo conjunto de un paquete integrado de música escrito íntegramente con código máquina.

En el mes de febrero, el equipo empieza a trabajar en el proyecto decidiéndose realizar el programa a base de iconos para facilitar el uso del mismo.

En octubre de 1984 se obtiene la primera versión comercial de The Music System realizada para el BBC micro.

En diciembre, el equipo del Music System inició el proyecto de una versión mejorada para el Commodore 64, incluyendo un nuevo módulo MIDI para la conexión del programa con teclados y sintetizadores.

La versión definitiva queda terminada en agosto de 1985, desarrollando System simultáneamente una versión simplificada del programa, especialmente concebida para ser comercializada en cassette.

Posteriormente llega la versión de **Amstrad**, basada íntegramente en la de Commodore.

diendo escuchar el resultado en cualquier momento.

Esta función nos permite copiar cualquier partitura de música y luego escuchar cómo suena, todo ello sin tener ni los más mínimos conocimientos de solfeo.

La parte del programa dedicada al **sintetizador** nos permite crear sonidos a nuestra medida, añadir efectos electrónicos a instrumentos musicales y luego utilizarlos en nuestro órgano como un sonido más.

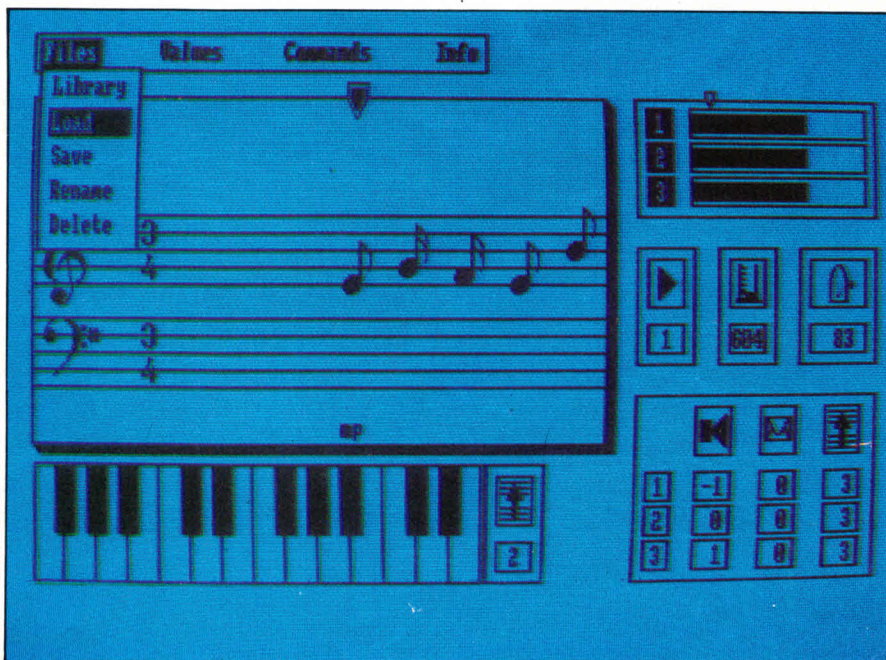
### No sólo para músicos

La cantidad de aspectos musicales que toca este programa, hace pensar que su manejo puede resultar extremadamente complicado y que su utilización requiere un gran número de horas de práctica, además de unos profundos conocimientos musicales.

Nada más lejos de la realidad, si este programa tiene algo que le hace destacar sobre las montañas de software existente, es que acerca la música hasta la gente que no la haya oído ni de lejos.

Introduciéndonos más en el tema, «The Music System» es ideal para aquellas personas que quieran estudiar música, constituyendo un inmejorable tutor musical tanto para niños como para adultos.

Por si todo esto fuera poco, el programa está realizado de forma que



## The Music System en acción

Cargado el programa en la memoria del ordenador tenemos en el monitor la pantalla básica, en la cual están contenidos todos los elementos fundamentales del programa y de la cual surgen los submenús y demás módulos de programa.

Está constituida por cinco bloques independientes: Teclado, Pentagrama, Secuenciador, Pistas de sonido y Menú principal.

## El menú principal

Está constituido por una línea situada en la parte superior de la pantalla conteniendo cuatro palabras; FILES, VALUES, COMMANDS, INFO.

Cada opción entra en funcionamiento al pulsar la correspondiente tecla de función, surgiendo de la nada, con la más pura técnica de la icografía informática, el submenú de opciones.

FILES, es el menú querige los archivos en cinta/disco.

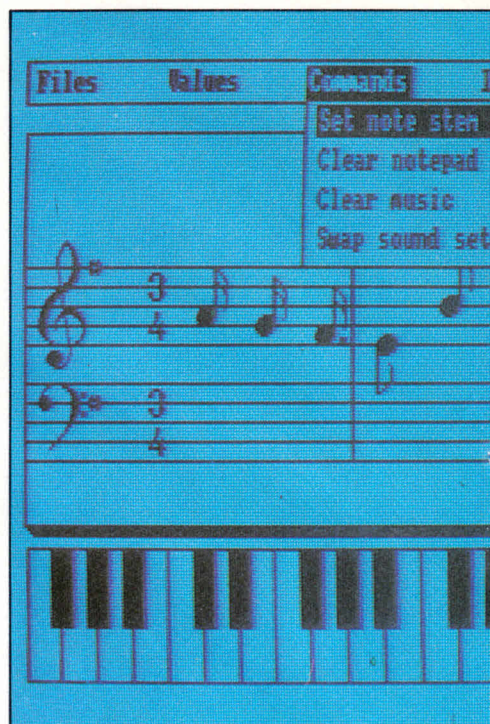
Hay dos tipos de archivos: MUSIC FILES para guardar melodías y canciones, y SOUND FILES, que nos permiten guardar sonidos sintetizados y sonidos de instrumentos.

El programa contiene una serie de canciones de demostración e instrumentos grabados.

VALUES nos da toda la información referente a la melodía que tenemos en memoria, detallándonos el compás, tempo, pista de sonido utilizada, clave y más información de valor inapreciable para el conocimiento de la obra que estamos ejecutando.

COMMANDS es utilizado para manejar el pentagrama electrónico, pudiendo de esta manera borrar la melodía de memoria, borrar determinada pista, limpiar la pantalla del pentagrama e intercambiar el contenido de dos voces.

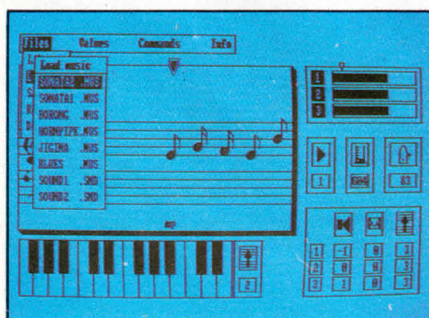
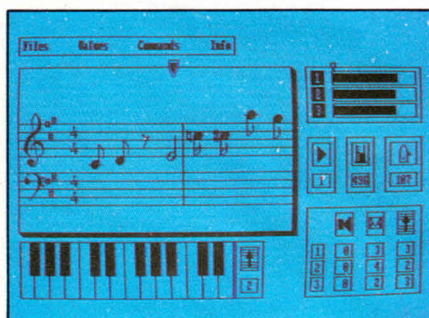
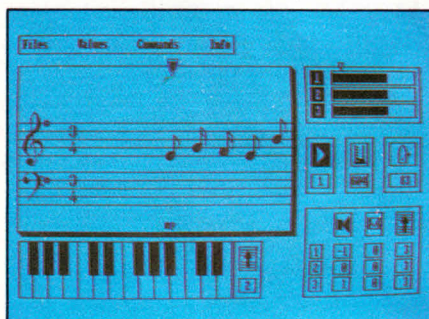
INFO nos facilita toda la información relativa a las notas contenidas en las tres pistas de sonido; de esta forma, conocemos el número de notas existentes en la pista número 1, en la 2 y en la 3; de igual modo, sabemos el sonido que existe en cada pista y el nombre con que tenemos archivada dicha composición.



La utilización del play back nos permite ejecutar una melodía con el fondo contenido en cualquiera de las pistas, de esta forma podemos componer canciones con melodía solista y dos acompañamientos.

Con el sistema de pistas podemos tener hasta tres instrumentos musicales sonando simultáneamente con las grandes posibilidades musicales que ello representa.

Cuando estemos satisfechos de nuestra composición, ésta se graba en cinta o disco para la posteridad.



## Tocando música

La interpretación de música con el teclado del ordenador reproduce exactamente las funciones de un órgano electrónico.

Disponemos de un instrumento de 22 teclas, semejante al teclado que aparece en pantalla, formando las teclas numéricas 1, 2, 3, las notas negras y las Q, W, E, etc., las notas blancas.

El cambio de octava se realiza a toque de tecla, disponiendo de tres posibilidades; grave, media, aguda.

## La música escrita

Una de las más impresionantes funciones de este programa, es la posibilidad de obtener partituras de forma automática.

El pentagrama electrónico es el responsable de esta impresionante aplicación, con él podemos obtener la partitura de una melodía a la vez que la estamos interpretando en el teclado.

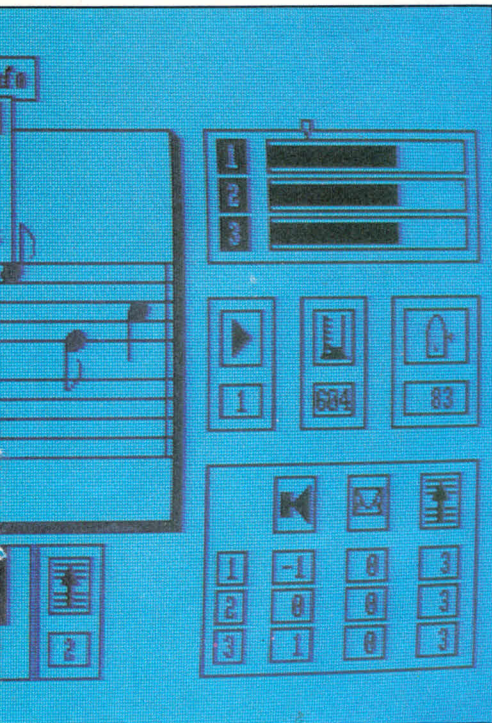
A cada toque de tecla, aparecerá en el pentagrama la nota con su duración correspondiente. Cuando hemos acabado de interpretar podemos reproducir la partitura completa en el pentagrama.

Otra posibilidad del pentagrama electrónico es la de escribir música directamente, como si lo realizásemos con lápiz y sobre papel pautado.

Para ello seleccionaremos en primer lugar la duración de la nota; re-

## Grabando pistas

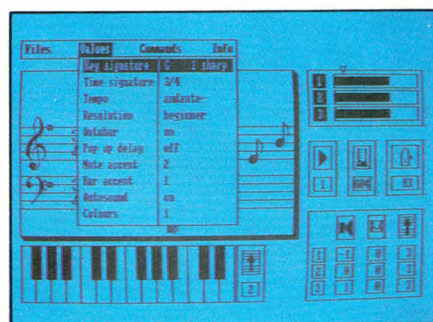
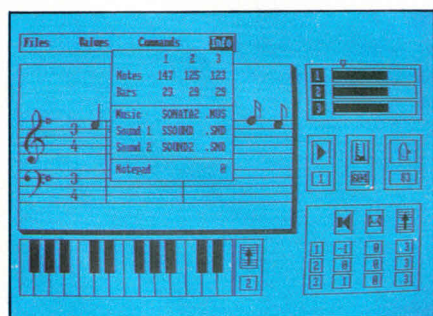
Nuestro órgano dispone de tres pistas distintas de grabación, en cada una de las cuales podemos almacenar una melodía distinta. Esta queda registrada según tocamos en el teclado, y puede ser corregida desde el punto deseado.



donde, corchea, etc., son representadas en un icono por el que nos movemos a toque de tecla hasta situarnos en la elegida.

Luego, la situaremos a la altura del pentagrama correspondiente al tono de la misma. Las notas pueden ser borradas e introducidas entre otras existentes.

Componer de esta forma sólo está al alcance de muy pocos, pero la escritura en pantalla se puede aprovechar para reproducir partituras y comprobar cómo suena lo impreso en partituras.



Las posibilidades que pone a disposición del usuario el pentagrama electrónico hacen que este programa sea de valor incalculable para los estudiantes de música, resolviendo de un plumazo todos los problemas que acarrea el solfeo.

### Síntesis de sonido

La música electrónica también está representada en este programa gracias al sintetizador.

Con un sintetizador, podemos construir los sonidos a nuestro gusto, pudiendo de esta manera reproducir el sonido de instrumentos de música; piano, guitarra, órgano, flauta, etc. o dotar a nuestro teclado de los sonidos representativos de la más avanzada música tecno.

La síntesis de sonido consiste en el diseño geométrico de la forma que va a tener nuestra onda de sonido en función del tiempo de emisión (envolvente sonora).

Después de lo cual podemos añadir efectos de eco, reverberación, filtros sonoros, resonancia, ruido y demás factores que integran el sonido electrónico.

Una vez obtenido nuestro sonido, nos serviremos del secuenciador para utilizarlo en nuestro teclado.

Mediante éste podemos controlar el volumen de esa voz, la altura tonal de la misma, y seleccionar el tipo de onda al que queremos que se ajuste nuestro sonido.

### Conclusiones

Tenemos ante nuestros ojos una auténtica obra maestra del software que permite acercar tanto la música clásica como la más avanzada música electrónica, al usuario de un ordenador.

Las superpotentes cualidades del pentagrama electrónico hacen de este programa un inmejorable tutor musical, idóneo tanto para los que no saben lo que es una corchea, como para los estudiantes de solfeo.

A los adictos a la interpretación, les impresionará la gama de sonidos que puede conseguir el sintetizador, y gracias a la utilización del play back, conseguirán sonar como una auténtica orquesta.

Todo lo cual se realiza de una forma fácil y espontánea gracias al in-

# Mister JOYSTICK

mejorable diseño del programa mediante el uso de iconos facilitando la comunicación con el usuario de forma extraordinaria.

### The Advanced Music System

Hasta ahora hemos hablado de la versión simplificada del programa, pero no hemos de olvidar que The Music System fue concebido como un completo programa de desarrollo musical.

The Advanced Music System, es el hermano mayor de nuestro programa, y con él obtendremos una máquina de auténtico tratamiento del sonido.

Esta versión está especialmente orientada a la utilización de un **MIDI** con el cual poder comunicar el ordenador con instrumentos musicales.

Gracias a él, se puede procesar el sonido de un órgano alterándolo a nuestra conveniencia, consiguiendo un potentísimo sintetizador, ideal para introducirnos en la experimentación sonora.

La versión avanzada es idónea para profesionales de la música o verdaderos entusiastas de la interpretación musical, constituyendo una herramienta de primera mano para cualquier grupo musical que quiera darle a su sonido una nueva dimensión.

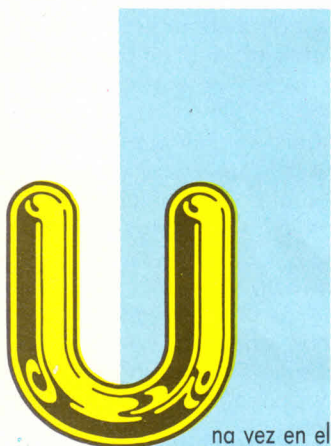
Este programa es distribuido por: **SERMA.**



# AMSTRAD PCW 8512 PEGANDO FUERTE

**Alan Sugar potencia el PCW 8256 y da un paso más dentro del mundo del pequeño ordenador de gestión.**

Alberto García S.



Una vez en el mercado el modelo compañero del gran CPC 6128, esperado PCW 8256, y en vista del éxito obtenido por este último, la firma británica ha decidido «mejorar lo presente» y ha sacado al mercado un modelo más avanzado: el PCW 8512. Con él, se ha eliminado la principal limitación del «256»: su capacidad de almacenamiento de programas y datos. Con una segunda unidad de disco y una —refrescante— ampliación de 256 K para el disco virtual. M (RAM interna), se ha logrado que, en comparación, sea el 512 un aparato con el que ya se puede trabajar con más soltura. Las posibilidades que ofrece son, naturalmente, más que las del pequeño de la familia, el 256. Y es que ya existe memoria para poder trabajar. Intentamos explicar todo ello.

## Generalidades

Externamente, unidad apagada, el 512 apenas si se diferencia del «pequeñín» 256. El color es un tanto más oscuro, las etiquetas nos dicen que nos encontramos ante un 512 y, lo más importante, la segunda unidad de disco,

que se encuentra debajo de la primera, justo donde el 256 tiene la placa que indica que el monitor es de fósforo verde. Del resto, todo es exactamente igual: impresora, monitor y teclado son los ya probados en el 256, que ya tienen más que confirmada su fiabilidad. Hasta aquí, poca cosa... Pero entremos en el aparato.

Al encender la unidad y cargar el disco de sistema, nos encontramos con más diferencias: el mensaje inicial es apreciablemente distinto. En vez de 112 K de memoria interna, nos encontramos con 368 K, o sea, 256 K adicionales. Grata sorpresa; por fin podremos tener programas completos en el disco virtual, entre otras cosas, y no es poco. Además de esto, el monitor nos comunica que tenemos dos unidades instaladas. Cabe decir aquí que es posible tener un 256 con segunda unidad de disco, pero sin ampliación de memoria interna, lo que nos situaría a medio camino entre los dos modelos. La diferencia es notable, y el aparato resulta un híbrido que no me atrevo a calificar. El tema de la ampliación de M lo tocaremos más adelante.

## 2.ª Unidad: diferencias con la 1.ª; ventajas e inconvenientes

En el PCW 512, la primera unidad de disco es exactamente igual a la del 256. La segunda es considerablemente más potente. Varía, en este caso, el formateado, que es más «den-





so», con lo que se consigue el poder almacenar mayor cantidad de información. La longitud del disco se aprovecha mejor. Y ojo, que la unidad B formatea las dos caras AL MISMO TIEMPO. No se asuste, entonces el usuario cuando, por error, formatea (inicializa) un disco en B de la misma forma que lo hace en A, es decir, primero una cara y luego la otra. Cuando se hace así, ocurre que la unidad borra el formato de la cara opuesta y, claro, al verificar nos dice que falta la primera marca de direcciones en la dicha cara. Para el usuario «orgullosote» (entre los que, lo reconozco, me incluyo) lo primero que viene a la cabeza es un posible defecto de la unidad. Y lo es, pero de la unidad que se sienta enfrente del aparato.

Y vamos a las capacidades: al aplicar la utilidad SHOW del sistema operativo a un disco recién formateado en la segunda unidad de disco tenemos como resultado que hay 706 K libres en éste (y no 720, que es lo que nos dice el fabricante). Aun así, la diferencia no es tanta, y esto ya permite un uso con unas posibilidades bastante más interesantes que las del pequeño de la familia PCW. Pero todo tiene sus peros: hay que decir que, en lo tocante a la segunda unidad, los discos pensados para la primera, los denominados CF-2, son válidos, pero no los propios. Para esta segunda unidad, el tipo de disco ha de ser el llamado CF2DD, o sea, de doble cara, pero de **doble densidad**. Esto debería de ser así porque, tal y como hemos dicho, la información, dado el formateado, va más condensada, y por ello por lo que es más conveniente utilizar un disco con una emulsión más sensible. Aun así, que quede claro que los discos CF-2 son válidos, aunque exista el peligro potencial de que causen problemas.

En lo que respecta a la compatibilidad entre unidades, también hay tela que cortar: lógicamente, un disco con formato para «720» K NO puede ser leído en la primera unidad, concebida para un formato de 172 K. Cuando se comete esta torpeza, la unidad «llora», emitiendo una especie de gruñidos, en señal de protesta por la incorrecta utilización. Tengan cuidado y no se equivoquen (sugiero que los discos formateados en B se marquen) porque parece ser poco difícil el que una unidad se quede bloqueada. Y a lo mejor éste es motivo para ello. En el otro sentido, es decir, de B para A, nos encontramos con todo un detalle por parte del fabricante: un disco de la unidad A SI puede ser leído por la segunda unidad de discos. Y digo todo un detalle porque esto no lo hacen todos los que se dedican a producir Hard. Esta cualidad resulta muy útil, sobre todo porque evita el engorro de tener que estar cambiando de disco constantemente.

La copia de discos, por ejemplo, es considerablemente más cómoda leyendo en B y escribiendo en A. Y también resulta práctico cuando necesitamos alguna utilidad del CP/M y tenemos M y A (sobre todo A) ocupados. En

## Para... PCW

fin, que las combinaciones son múltiples y todas son mejoras sobre el 256. Y ya que el tema son las unidades de disco, quisiera traer a colación otro tema: cuando se lleva mucho tiempo trabajando con el ordenador, se aprecia un calentamiento de los discos que no parece normal. La pregunta que surge es si este calentamiento no perjudicará al disco, y si no sería conveniente sugerir la implantación de un pequeño ventilador, al igual que las grandes unidades lo tienen. Y es que creo que se debería cuidar el aspecto de la seguridad de los datos almacenados en los Floppys. ¿No les parece?



### El virtuosismo de la memoria virtual

Es con la ampliación de memoria, de 256 K, con lo que se llega a redondear el aparato. Las 368 K libres nos permiten utilizar la memoria interna igual que si fuese otro disco, con la enorme ventaja que ello conlleva, porque, al suprimirse el accionamiento mecánico que guía el cabezal en una disquetera y el proceso de lectura, la ejecución de programas por M y el acceso a los datos es considerablemente más rápido. Vendría entonces esta memoria interna a reemplazar, por su capacidad, a la unidad A, pudiendo ésta quedar disponible para otros menesteres, como el de almacena-

miento de datos. Y dense cuenta de si esta ampliación representa avance en el momento en que, contando con la 2.ª unidad, disponemos de algo más de 1 Megabyte para almacenar datos. Me parece que sobran comentarios, ¿no?

### Acerca del soft...

No existen problemas. Todos los programas ejecutables por el 256 lo son igualmente por el 512. Lo que es más, hay muchos programas pensados inicialmente para el 256 que, en previsión, tienen rutinas para aprovechar la 2.ª unidad de disco del 512. Como ejemplo, el tener en A el disco de programa y en la unidad B, los datos que surgen del proceso de éste. En lo concerniente a los programas

las que nos dan las opciones de utilizar la unidad B con el programa mencionado. En el resto del soft no se han encontrado variaciones.

### En definitiva

Toda resignación que pudiese tener el usuario de un 256 al ver que su aparato se le quedaba «chiquito» queda aliviada, sin duda, con la comercialización del 512 —por una parte— y de la ampliación, sumada a la segunda unidad, por otra.

Salvando el temor que puede inspirar el uso de discos CF-2 en la 2.ª unidad (y con la esperanza de que los discos de doble densidad salgan YA, que todavía no han salido). (La verdad es que me pregunto para qué se ponen a comercializar la 2.ª unidad sin tener en

rato pero sí la voluntad de tenerlo, mi consejo es que se vayan al 512, que merece la pena. Esto siempre y cuando se le piense expresar, que, si bien se deja, hay mucho que aclarar. Para los que ya tienen un 256, quisiera aportar el dato de que el montaje de la segunda unidad de disco, más la ampliación, les viene a salir aproximadamente por la diferencia entre ambos equipos. Y a propósito de montaje, decir que éste parece ser que resulta bastante difícil, repitiendo palabras de los técnicos. Como último detalle curioso, para aquel que tenga acceso a las denominaciones de los «escarabajos» que componen la ampliación de 256 K, que vayan y pregunten, por piezas sueltas, el precio. A lo mejor hay sorpresas.

El camino de **Amstrad** parece claro. NO a la expansión a un mismo nivel, sino la ascensión de categorías paulatinamente. En esta ascensión resulta un peldaño sólido el de la —corta— familia PCW. Se ve que lo meditan.

Aparte de una tercera unidad de 10 Megs en disco duro para esta serie (lo que ya sería rizar el rizo) parece que se confirma la salida, allá para septiembre, de un compatible, demolador en lo tocante a precios. Aquí, a la espera de verlo. Entretanto... A sacarle partido a su «512».

**NOTA:** los discos de doble densidad se encontrarán ya disponibles cuando usted lea este artículo, cosa que no ocurría en el momento de su confección.



que vienen con el equipo, no hay diferencias, exceptuando el hecho de que, con el 512 se suministra ya una versión más avanzada del Loco Script, la 1.21 (esta versión ya venía con algunas unidades del 256 de última hornada). La diferencia primordial, aparte de algunas simplificaciones con vistas a una mayor rapidez de uso (la 1.21 tiene una ventana que agrupa casi todas las opciones de edición, de manera que no haya que ir ventana por ventana) es que la última versión ofrece la posibilidad de pasar cualquier fichero creado a ASCII, con lo cual éste puede ser interpretado por otros aparatos que trabajen bajo CP/M.

Por otra parte, la utilidad DISCKIT, al tener las dos unidades instaladas, ejecuta otras rutinas que, si utilizamos el mismo disco en un 8256, no aparecen. Estas rutinas son, claro,

el mercado los discos apropiados para ella, y lo que es más, nos dicen, claro, que la seguridad de los datos grabados en un CF-2 peligra si éstos se utilizan en formato «720». Más bien indignante, francamente) es el AMSTRAD PCW 8512 un aparato FIABLE (no, no resulta una contradicción, pues la culpa de lo anteriormente dicho no es del Hardware), además de ser un equipo con mucha madera de profesional, perfectamente encuadrable en el terreno de la gestión a medio nivel (conozco aparatos que están llevando satisfactoriamente contabilidades completas de empresas).

Con un incremento de la memoria que resulta ser cuatro veces mayor que la del 256 (706 K de disco B, más unas 320 de A, más 368 de M suman 1.384 K), ya no tiene por qué haber temor a quedarse cortos. En lo tocante al precio, para aquellos que no tienen apa-

## FICHA TECNICA

- Modelo:** Amstrad PCW 8512.
- Microprocesador:** Z-80 de Zilog. Frecuencia de trabajo: 4 Mhz.
- Monitor:** Fósforo verde, 12". Control de brillo.
- Teclado:** Profesional, tipo QWERTY. ñ y acentos. Teclado numérico y teclas redefinibles por el usuario.
- Impresora:** 80 columnas. Cabezal matricial, 20 caracteres por segundo (N.L.Q.), 90 caracteres por segundo en calidad estándar. Papel continuo u hojas sueltas. Bajo Loco-Script, hasta 108 tipos diferentes de letra.
- Floppys:** Dos, uno para formatear a 172 K por cara, y otro para 720 K (formatea el disco de una vez).
- Memoria interna:** 368 K disco virtual. 61 K memoria de usuario.

# NewPrint

## La personal más profesional

Amplia gama para que usted seleccione la impresora ideal para su micro.  
Desde 80 col. y 80 cps. hasta 132 col. y 130 cps.



**DSE** S.A.

DISTRIBUIDORA DE SISTEMAS ELECTRONICOS, S.A.

AV. ANT. CARRETERA DEL PRAT/PJE. DOLORES  
TEL. (93) 336 33 62 TLX 97760 DSIE-E  
L'HOSPITALET DE LLOBREGAT (BARCELONA)

● INFANTA MERCEDES, 83  
TELS. (91) 279 11 23 / 279 36 38  
28020 MADRID

# MASTERMIND

Javier Acero Villán

**El Mastermind no necesita presentación. Ya sabes, es lo que siempre se dice cuando no se tiene ni..., lo que yo te diga, idea de qué escribir acto seguido, y el papel, como la más hambrienta e implacable de las Parcas, o como un Ford Escort, sin ir más lejos, reclama más y más combustible: carnada, carnada. Pero en este caso, es cierto. Todos conocéis el Mastermind, todos habéis jugado, ganado o perdido a él, pero, no a éste. Creedme: no es igual a los demás; por presentación y por programación.**



El programa que les presento, es el clásico juego de salón del mismo título, aprovechando las posibilidades multicolores del Amstrad.

Como en el juego de mesa, se trata de adivinar un código de cuatro colores de entre seis posibles, en el Nivel 1, o entre siete en el Nivel 2. Teniendo en cuenta que los colores pueden repetirse, obtenemos 1.296 ó 2.401 posibles combinaciones respectivamente.

A cada código propuesto por nosotros nos irá dando información mediante cruces rojas, si hemos acertado la posición y el color, o amarillas, si adivinamos el color pero no la posición correcta.

El programa lo he realizado para que vaya controlando el juego de hasta 5 jugadores, apareciendo la clasificación, ordenada de mayor a menor puntuación, cuando todos los jugadores hayan cubierto su turno, e indicado los puntos obtenidos en la partida y el total acumulado.

Las instrucciones que figuran en el programa, creo que son lo suficientemente explicativas para no necesitar más aclaraciones.

En cuanto al programa, en el listado se puede comprobar que está muy estructurado, diferenciando claramente cada rutina, con lo que seguirlo y entenderlo no ofrece la menor dificultad.

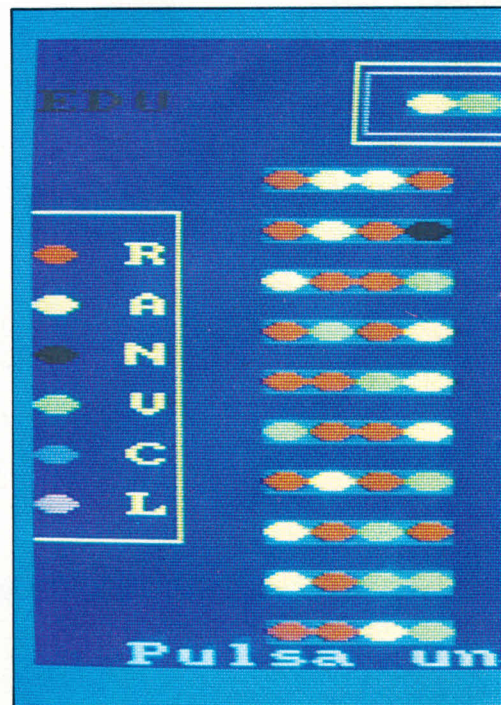
Las variables y subrutinas más importantes utilizadas son las siguientes:

## VARIABLES

color	Lista de colores elegidos por el ordenador
coladi	Colores a adivinar por el jugador
jugador\$	Lista de jugadores
jug\$	Lista de jugadores a clasificar
puntos	Puntuación de cada jugador
pun	Puntuación a clasificar
totales	Puntuaciones acumuladas
tot	Totales a clasificar
J	Número de jugadores
nivel	Niveles de juego
tirada	Controla el ciclo que nos da 10 oportunidades de acierto
cruzar	Aciertos en color y posición
cruza	Aciertos sólo en color
swr	Switch que controla el número de cruces rojas
swa	Idem amarillas
pista	Núm. de pistas por tirada
m, t, d	Número de notas, tono y duración

## SUBROUTINAS

80	Dimensiona las listas
100-240	Crea caracteres
250-350	Pregunta si desea instrucciones
380-630	Dibuja la pantalla de juego
640-700	Carga la lista de 4 colores al azar, asignando el color 12 si sale un 2, o el color 7 si sale un 4.
710-890	Selecciona y dibuja los colores elegidos por el jugador así como la puntuación
900-1030	Compara los colores del ordenador con los del jugador preparando las cruces rojas o amarillas que luego tendrá que escribir
1040-1140	Va dando las pistas y restando puntos si no se acierta
1150-1330	Música de fallo o acierto
1340-1430	Descubre el código oculto
1440-1590	Acumula totales y ordena la tabla de clasificación
1600-1770	Escribe la tabla de clasificación y pregunta por otra partida
1780-1920	Carga el número de jugadores
1930-2000	Elige nivel
2010-2040	Rótulo
2050-2380	Instrucciones

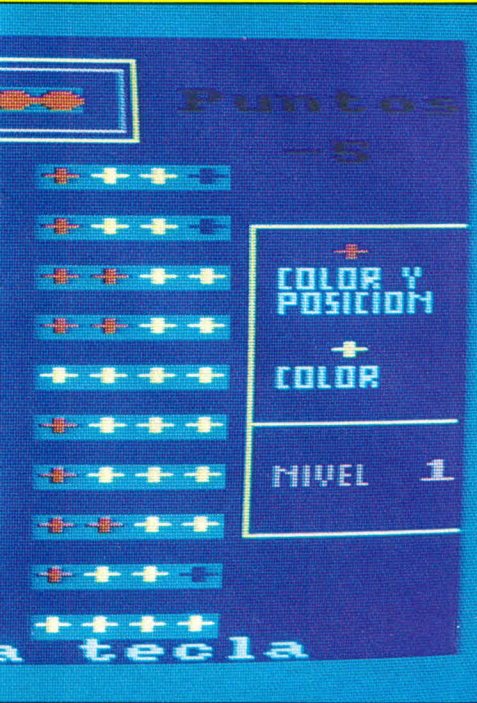


CA

```

10 *****
20 *** MASTER-MIND **
30 *** por **
40 *** Javier Acero **
50 *** 1.986 **
60 *****
70 RANDOMIZE TIME
80 REM ----- Dimensiones de listas -----
90 DIM color(4),coladi(4),jugador$(5),jug$(5),puntos(5),pun(5),totales(5),tot(5)
100 REM ----- Creacion de caracteres -----
110 SYMBOL AFTER 242
120 SYMBOL 242,0,128,128,128,128,0,128,128
130 SYMBOL 244,0,109,73,73,109,69,69,109
140 SYMBOL 245,0,221,85,85,221,21,21,1,21
150 SYMBOL 246,0,174,42,42,42,42,42,174
160 SYMBOL 247,0,149,213,181,149,149,9,148,148
170 SYMBOL 248,0,90,82,82,90,82,146,155
180 SYMBOL 249,0,221,149,149,149,149,9,149,221
190 SYMBOL 250,0,59,42,42,43,43,42,186
200 SYMBOL 251,0,133,133,133,130,2,130,130
210 SYMBOL 252,0,238,170,170,234,138,138,142
220 SYMBOL 253,0,214,148,148,212,84,84,214
230 SYMBOL 254,0,186,171,170,170,170,64,170,186
240 SYMBOL 255,0,64,64,192,64,64,64,64
250 REM ----- Instrucciones -----
260 MODE 1:BORDER 15:LOCATE 7,12:PRINT"QUIERES INSTRUCCIONES? (S/N)"
270 a$=INKEY$:IF a$="" THEN 270
280 a$=UPPER$(a$):IF a$="S" THEN 310
290 IF a$="N" THEN 360
300 GOTO 270
310 GOSUB 2020
320 LOCATE 14,24:PRINT"PULSA ESPACIO"
330 WINDOW#1,1,40,22,4
340 PAPER#1,2:PEN#1,0:CLS#1
350 GOSUB 2060

```



```

360 REM ----- Numero de jugadas -----
370 GOSUB 1780
380 FOR I=1 TO J
390 REM ----- PANTALLA -----
400 MODE 0:BORDER 10:puntos(I)=10
410 PLOT 221,335,1:DRAW 418,335:DRAW 418,385:DRAW 221,385:DRAW 221,335
420 WINDOW#1,8,13,2,4
430 PAPER#1,5:CLS#1
440 LOCATE 8,3:PAPER 8:PEN 3:PRINT"CODIGO"
450 PEN 0
460 FOR n=6 TO 24 STEP 2
470 LOCATE 6,n:PRINT STRING$(4,CHR$(233))
480 LOCATE 12,n:PRINT STRING$(4,CHR$(43))
490 NEXT
500 LOCATE 1,9:PAPER 0:PEN 3:PRINT CHR$(231);:PEN 1:PRINT" R"
510 LOCATE 1,11:PRINT CHR$(231);:PRINT" A"
520 LOCATE 1,13:PEN 5:PRINT CHR$(231);:PEN 1:PRINT" N"
530 LOCATE 1,15:PEN 12:PRINT CHR$(231);:PEN 1:PRINT" V"
540 LOCATE 1,17:PEN 6:PRINT CHR$(231);:PEN 1:PRINT" C"
550 LOCATE 1,19:PEN 7:PRINT CHR$(231);:PEN 1:PRINT" L"
560 IF nivel=2 THEN LOCATE 1,21:PAPER 8:PEN 0:PRINT CHR$(231);:PAPER 0:PEN 1:PRINT CHR$(244)CHR$(245)CHR$(246)
570 IF nivel=1 THEN PLOT 0,80:DRAW 100,80:DRAW 100,290:DRAW 0,290:GOTO 590
580 PLOT 0,50:DRAW 125,50:DRAW 125,290:DRAW 0,290
590 LOCATE 18,9:PEN 3:PRINT CHR$(43);:LOCATE 17,10:PEN 2:PRINT CHR$(249)CHR$(250)CHR$(251):LOCATE 17,11:PRINT CHR$(252)CHR$(253)CHR$(254)CHR$(255)
600 LOCATE 18,13:PEN 1:PRINT CHR$(43);:LOCATE 17,14:PEN 2:PRINT CHR$(249)CHR$(250)CHR$(251)CHR$(242)
610 PLOT 639,80:DRAW 495,80:DRAW 495,280:DRAW 639,280
620 PLOT 495,153:DRAW 635,153
630 LOCATE 17,18:PEN 7:PRINT CHR$(247)CHR$(248);STR$(nivel)
640 REM ----- Codigo de colores para el ordenador -----
650 FOR c=1 TO 4
660 IF nivel=1 THEN color(c)=INT(6*RND+1)
670 IF nivel=2 THEN color(c)=INT(7*RND)
680 IF color(c)=2 THEN color(c)=12
690 IF color(c)=4 THEN color(c)=7
700 NEXT
710 REM ----- Colores elegidos por el jugador -----
720 LOCATE 1,3:PEN 5:PRINT jugador$(I)
730 FOR tirada=6 TO 24 STEP 2
740 LOCATE 15,3:PAPER 0:PEN 5:PRINT"Puntos":LOCATE 17,5:PRINT puntos(I)
750 LOCATE 5,tirada:PAPER 0:PEN 14:PRINT CHR$(243):PEN 1
760 PAPER 8
770 FOR c=1 TO 4
780 a$=INKEY$:IF a$="" THEN 780
790 a$=UPPER$(a$)
800 IF a$="R" THEN coladi(c)=3:GOTO 880
810 IF a$="A" THEN coladi(c)=1:GOTO 880
820 IF a$="N" THEN coladi(c)=5:GOTO 880
830 IF a$="V" THEN coladi(c)=12:GOTO 880
840 IF a$="C" THEN coladi(c)=6:GOTO 880
850 IF a$="L" THEN coladi(c)=7:GOTO 880
860 IF nivel=2 THEN IF a$=" " THEN coladi(c)=0:GOTO 880
870 GOTO 780

```

# Serie ORO

```

880 LOCATE 5+c,tirada:PEN coladi(c):PRINT CHR$(231):PRINT CHR$(7)
890 NEXT c
900 REM ----- Compara colores -----
910 cruzr=0:cruza=0:DIM swr(4),swa(4)
920 FOR c=1 TO 4
930 IF color(c)=coladi(c) THEN cruzr=cruzr+1:swr(c)=1:swa(c)=1
940 IF cruzr=4 GOTO 1030
950 NEXT c
960 FOR z=1 TO 4
970 IF swr(z)=1 THEN 1020
980 FOR c=1 TO 4
990 IF swa(c)=1 THEN 1010
1000 IF coladi(c)=color(z) THEN cruzr=cruza+1:swa(c)=1:GOTO 1020
1010 NEXT
1020 NEXT
1030 ERASE swr:ERASE swa
1040 REM ----- Da las pistas -----
1050 FOR pista=1 TO cruzr
1060 LOCATE 11+pista,tirada:PEN 3:PRINT CHR$(43):SOUND 1,250,30,5:SOUND 1,300,70,3:FOR r=1 TO 1000:NEXT
1070 NEXT pista
1080 IF cruzr=4 GOTO 1250
1090 FOR pista=1 TO cruza
1100 LOCATE 11+pista+cruzr,tirada:PEN 1:PRINT CHR$(43):SOUND 1,150,30,5:SOUND 1,100,70,3:FOR r=1 TO 1000:NEXT
1110 NEXT pista
1120 LOCATE 5,tirada:PAPER 0:PRINT CHR$(32):PAPER 8
1130 puntos(I)=puntos(I)-1
1140 NEXT tirada
1150 REM ----- Fallo -----
1160 puntos(I)=-5:LOCATE 17,5:PAPER 0:PEN 5:PRINT puntos(I)
1170 RESTORE 1230
1180 FOR m=1 TO 16
1190 READ t,d
1200 SOUND 3,t,d,6
1210 SOUND 3,0,5
1220 NEXT
1230 DATA 179,50,179,50,179,50,190,25,213,25,190,50,190,50,213,25,239,25,213,50,213,50,239,25,213,25,239,50
1240 GOTO 1340
1250 REM ----- Acierto -----
1260 RESTORE 1330
1270 FOR m=1 TO 20
1280 READ t,d
1290 SOUND 1,t/3,d,6
1300 SOUND 2,t,d,6
1310 SOUND 3,0,5
1320 NEXT
1330 DATA 319,100,358,100,358,20,379,100,379,20,358,20,319,20,319,50,319,20,319,20,319,50,358,20,379,20,319,20,319,50,358,20,379,20,319,20,358,100
1340 REM ----- Descubre codigo -----
1350 PAPER#1,0:CLS#1:LOCATE 9,3
1360 PLOT 228,340,15:DRAW 408,340:DRAW 408,379:DRAW 228,379:DRAW 228,340
1370 FOR c=1 TO 4
1380 PAPER 8:PEN color(c):PRINT CHR$(231);
1390 NEXT c
1400 LOCATE 3,25:PAPER 0:PEN 2:PRINT"Pulsa una tecla"
1410 a$=INKEY$:IF a$="" THEN 1410
1420 NEXT I

```

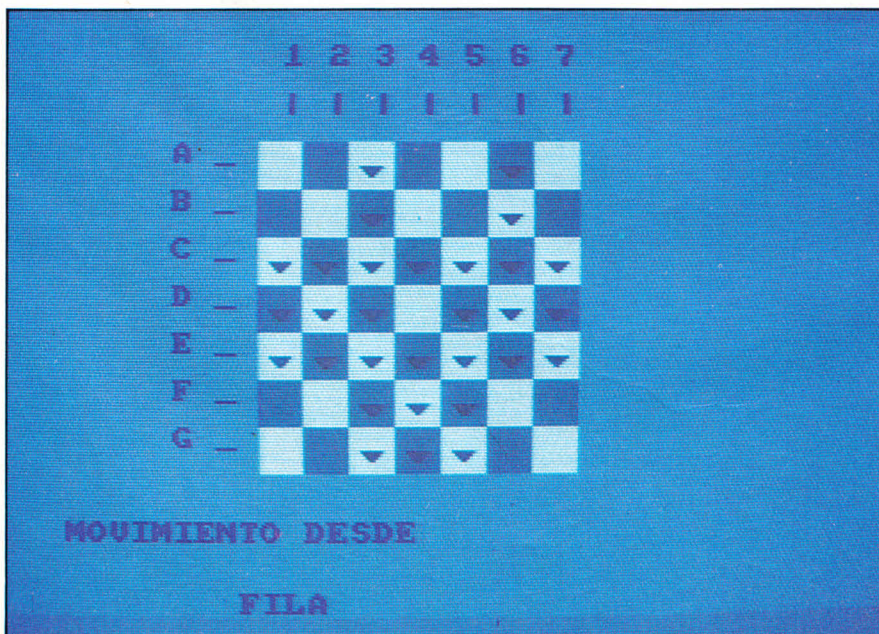


## ESTRUCTURA

10-60 70-230	Título Inicialización Limpia el buffer del teclado (y desactiva la tecla ESC), selecciona el modo, ventanas y colores, define los símbolos que formarán la ficha y las diversas variables	830	Comprueba que las 4 coordenadas no son distintas
240-610 250-310	Dibujo del tablero	840-980	Halla la situación de la ficha comida y comprueba que la casilla está ocupada. Si no lo está, salta a la línea 1110
320-380	Dibujo del tablero (parecido al del ajedrez)	990-1090	Realización del movimiento Dibuja la ficha situada en la casilla de origen y la ficha comida en el color de sus respectivas casillas (equivalente a borrarlas) y coloca la ficha en la casilla de destino. Disminuye el n.º de fichas y comprueba cuántas quedan
390-550	Dibuja las rayas situadas encima y a la izquierda del tablero Dibuja las fichas sobre el tablero en modo transparente (CHR\$(22)—CHR\$(1)). La línea 550 borra la casilla central	1110-1150 1160-1320	Movimiento inválido Resultado Muestra el n.º de jugadas y de fichas que han quedado en el tablero, el tiempo durante el que se ha jugado, la lista de las jugadas y pregunta si se juega de nuevo
560-590	Escribe las letras y los números de las casillas		
600-610	Limpia el buffer del teclado y desactiva la tecla ESC. Halla el tiempo del reloj al empezar		
620-980	Toma de datos y comprobación del movimiento		
630-730	Asigna fila y columna de origen y comprueba que la casilla está llena (tinta 3). Si no lo está salta a la línea 1110	1390-1600	Subrutina de toma de datos del movimiento (fila y columna). Pide fila y columna. Almacena la fila y la columna elegidas como elementos de las matrices fff\$(,) y ccc\$(,), respectivamente
740-820	Asigna fila y columna de destino y comprueba que la casilla no está llena (tinta 3). Si lo está, va a la línea 1110		

El programa se basa en la conversión de coordenadas PRINT en coordenadas PLOT, comprobando posteriormente el color de toda la casilla

ocupada por la ficha mediante la función TEST, para detectar si hay alguna ficha en la casilla o si la casilla está vacía.



## Serie ORO

### VARIABLES

ccc\$(,)	Columnas ocupadas en cada jugada ccc\$(,0): columna de origen ccc\$(,1): columna de destino
fff\$(,)	Filas ocupadas en cada jugada fff\$(,0): fila de origen fff\$(,1): fila de destino
p	N.º de fichas que quedan
m	N.º de movimientos
ta1\$ y ta2\$	Cuadrados para crear el tablero
x, y	Variables usadas en el dibujo del tablero
a\$	Cadena que contiene el dibujo de la ficha
flag	Bandera de movimiento (O---movimiento desde...; 1---movimiento hasta...)
f\$, c\$	Fila y columna elegidas
fi, ci	Fila y columna inicial
ff, cf	Fila y columna final (de destino)
fm, cm	Fila y columna de la ficha comida
er\$	Cadena con el mensaje de movimiento inválido
yn\$	Respuesta a la pregunta de ver lista de jugadas
an\$	Respuesta a la pregunta de intentarlo de nuevo
ret	Bucle de retardo
ccm	Color de la casilla de la ficha comida
cci	Color de la casilla de la ficha que salta
ti	Hora del reloj interno al comenzar
tf	Hora del reloj interno al terminar
tiempo	Tiempo transcurrido (en segundos)
minutos	Minutos durante los que se ha jugado
segundos	Segundos transcurridos
tiempo\$	Cadena con información del tiempo jugado

NOTA: Para dejar de jugar, pulsar "?" (el signo de interrogación), con lo que se accede a la pantalla de información, desde la que se puede abandonar el programa con el fin de acceder al listado.

```

10 REM *****
20 REM * SALTO DE DAMA *
30 REM *****
40 REM * Antonio Fernando *
50 REM * Aguilera Romera *
60 REM *****
70 REM * inicializacion *
80 RANDOMIZE TIME
90 CALL &BBO3:REM **** En el CPC 64
4 y CPC 612B sustituir por: ON BREA
K CONT: CLEAR INPUT ****
100 INK 0,11:INK 1,26:INK 2,13:INK
3,0:BORDER 2
110 PEN 1:PAPER 0:CLS
120 MODE 1
130 WINDOW #0,1,40,1,24:WINDOW #1,1
,40,25,25:PEN #1,3:PAPER #1,0:CLS #
1
140 DIM ccc$(32,1),fff$(32,1)
150 p=32:m=0
160 SYMBOL 255,15,7,3,1,0,0,0,0
170 SYMBOL 254,240,224,192,128,0,0,
0,0
180 a$=CHR$(255)+CHR$(254)
190 er$="MOVIMIENTO INVALIDO"
200 PRINT CHR$(22)+CHR$(0)
210 CLS
220 ta1$=CHR$(14)+CHR$(1)+" "+CHR$(
14)+CHR$(2)+" "
230 ta2$=CHR$(14)+CHR$(2)+" "+CHR$(
14)+CHR$(1)+" "
240 REM * dibujo del tablero *
250 FOR y=6 TO 20 STEP 4
260 FOR x=13 TO 27 STEP 4:LOCATE x,
y:PRINT ta1$:LOCATE x,y+1:PRINT ta1
$
270 LOCATE x,y+2:PRINT ta2$:LOCATE
x,y+3:PRINT ta2$
280 NEXT x
290 NEXT y
300 FOR y=20 TO 21:PAPER 0:LOCATE 1
3,y:PRINT " ":NEXT
310 FOR y=6 TO 20:PAPER 0:LOCATE 27
,y:PRINT " ":NEXT
320 PEN 3
330 FOR y=6 TO 19 STEP 2
340 LOCATE 11,y:PRINT "_"
350 NEXT y
360 FOR x=14 TO 26 STEP 2
370 LOCATE x,4:PRINT "!"
380 NEXT
390 PRINT CHR$(22)+CHR$(1)
400 FOR x=13 TO 25 STEP 2
410 FOR y=11 TO 15 STEP 2
420 LOCATE x,y:PRINT a$
430 NEXT y
440 NEXT x
450 FOR x=17 TO 21 STEP 2
460 FOR y=7 TO 13 STEP 2
470 LOCATE x,y:PRINT a$
480 NEXT y
490 NEXT x
500 FOR x=17 TO 21 STEP 2
510 FOR y=17 TO 19 STEP 2
520 LOCATE x,y:PRINT a$
530 NEXT y
540 NEXT x
550 LOCATE 19,13:PEN 1:PRINT a$
560 PEN 3:FOR i=1 TO 7
570 LOCATE 9,4+2*i:PRINT CHR$(64+i)
580 LOCATE 11+i*2,2:PRINT i
590 NEXT i
600 CALL &BBO3:REM **** En el CPC 6
4 y CPC 612B sustituir por: CLEAR
INPUT ****
610 ti=TIME
620 REM * toma de datos y comprobac
ion del movimiento *

```

```

630 PEN 3
640 flag=0:LOCATE 4,22:PRINT "MOVIM
IENTO DESDE"
650 GOSUB 1400:IF F$="" THEN 640
660 fi=(ASC(f$)-64)*2+4
670 ci=c*2+11
680 FOR i=ci*16 TO (ci+1)*16
690 FOR j=399-(fi*16) TO 399-((fi+1
)*16) STEP -1
700 IF TEST(i,j)=3 THEN 740
710 NEXT j
720 NEXT i
730 GOTO 1110
740 LOCATE 25,22:PRINT "HASTA"
750 flag=1:GOSUB 1400:IF f$="" THEN
GOTO 750
760 ff=(ASC(f$)-64)*2+4
770 cf=c*2+11
780 FOR i=cf*16 TO (cf+1)*16
790 FOR j=399-(ff*16) TO 399-((ff+1
)*16) STEP -1
800 IF TEST(i,j)=3 THEN 1110
810 NEXT j
820 NEXT i
830 IF fi<>ff AND ci<>cf THEN 1110
840 IF ci<>cf THEN 890
850 IF ABS(fi-ff)<>4 THEN GOTO 1110

860 fm=fi+2*SGN(ff-fi)
870 cm=ci
880 GOTO 920
890 IF ABS(ci-cf)<>4 THEN 1110
900 cm=ci+2*SGN(cf-ci)
910 fm=fi
920 FOR i=cm*16 TO (cm+1)*16
930 FOR j=399-(fm*16) TO 399-((fm+1
)*16) STEP -1
940 IF TEST(i,j)=3 THEN 980
950 NEXT j
960 NEXT i
970 GOTO 1110
980 m=m+1
990 REM * realizacion del movimient
o *
1000 PRINT CHR$(22)+CHR$(1);
1010 cci=TEST((ci-1)*16,399-(fi*16)
)
1020 PEN cci:LOCATE ci,fi+1:PRINT a
$
1030 PEN 3:LOCATE cf,ff+1:PRINT a$
1040 ccm=TEST((cm-1)*16,399-(fm*16)
)
1050 PEN ccm:LOCATE cm,fm+1:PRINT a
$
1060 FOR i=0 TO 2:SOUND 1,INT(RND*1
00+50),3:NEXT
1070 p=p-1
1080 IF p>1 THEN PEN 3:GOTO 640
1090 GOTO 1170
1100 REM * movimiento erroneo *
1110 FOR i=3 TO 21:LOCATE 37,i:PRIN
T MID$(er$,i-2,1):NEXT
1120 FOR ret=0 TO 1999:NEXT ret
1130 FOR i=3 TO 21:LOCATE 37,i:PRIN
T " ":NEXT
1140 CLS #1
1150 LOCATE 1,22:PRINT STRING$(40,3
2):f$="":GOTO 640
1160 REM * resultado de la partida
*
1170 PEN 3:tf=TIME:INK 0,10:MODE 1:
FOR I=1 TO 100 STEP 3:SOUND 1,1,10:
NEXT:LOCATE 5,11:PRINT CHR$(175);"F
ELICIDADES! LO HAS CONSEGUIDO":GOTO
1200
1180 INK 0,10:MODE 1:LOCATE 11,10:P
RINT "EMPLEASTE"m"JUGADAS"
1190 LOCATE 12,12:PRINT "QUEDARON"p
"PIEZAS"
1200 tiempo=(tf-ti)/300: minutos=INT
(tiempo/60): segundos=tiempo MOD 60:
tiempo$="TIEMPO==>" +STR$(minutos)+
" MINUTOS"+STR$(segundos)+" SEGUNDO
S"
1210 LOCATE INT(21-(LEN(tiempo$)/2)
),14:PRINT tiempo$:IF segundos=1 AN
D minutos=1 THEN LOCATE 24,14:PRINT

```

```

" ":LOCATE 35,14:PRINT " ":GOTO 12
50
1220 IF segundos=1 AND minutos<10 T
HEN LOCATE INT(21-(LEN(tiempo$)/2))
+30,14:PRINT " "
1230 IF segundos=1 AND minutos>9 TH
EN LOCATE INT(21-(LEN(tiempo$)/2))+
31+ABS(minutos>99),14:PRINT " "
1240 IF minutos=1 THEN LOCATE INT(2
1-(LEN(tiempo$)/2))+19,14:PRINT " "
1250 LOCATE 2,16:PRINT CHR$(174);"Q
UIERES VER LA LISTA DE MOVIMIENTOS?
"
1260 yn$=UPPER$(INKEY$):IF yn$="" T
HEN GOTO 1260
1270 IF yn$="N" THEN GOTO 1330
1280 IF yn$<>"S" THEN GOTO 1260
1290 MODE 1:FOR xx=0 TO m-1:IF xx<1
6 THEN LOCATE 8,xx+4:PRINT fff$(xx,
0);ccc$(xx,0);"-";fff$(xx,1);ccc$(x
x,1):ELSE IF xx=16 THEN PRINT CHR$(
30);:LOCATE 27,xx-12:PRINT fff$(xx,
0);ccc$(xx,0);"-";fff$(xx,1);ccc$(x
x,1)
1300 IF xx>16 THEN LOCATE 27,xx-12:
PRINT fff$(xx,0);ccc$(xx,0);"-";fff
$(xx,1);ccc$(xx,1)
1310 NEXT
1320 LOCATE 12,24:PRINT "PULSA UNA
TECLA":WHILE INKEY$="" :WEND
1330 CLS:LOCATE 6,14:PRINT CHR$(174
);"QUIERES INTENTARLO DE NUEVO?"
1340 an$=INKEY$
1350 an$=UPPER$(an$)
1360 IF an$="S" THEN RUN
1370 IF an$<>"N" THEN SOUND 1,INT(R
ND*100+150),5:GOTO 1340
1380 MODE 1:CLS:END
1390 REM * subrutina de toma de dat
os del movimiento (fila y columna)
*
1400 PRINT CHR$(22)+CHR$(0)
1410 LOCATE #1,1,1:PRINT#1,"
FILA ";:
1420 f$=INKEY$:IF f$="" THEN 1420
1430 IF f$<>CHR$(32) THEN PRINT #1,f
$:ELSE GOTO 1410
1440 FOR ret=1 TO 300:NEXT ret
1450 IF f$="?" THEN tf=TIME:GOTO 11
80
1460 f$=UPPER$(f$)
1470 IF ASC(f$)<65 OR ASC(f$)>71 TH
EN CLS #1:SOUND 1,53,20:GOTO 1410
1480 LOCATE 21+10*(ABS(flag=1)),22:
PRINT f$
1490 LOCATE #1,1,1:PRINT#1,"
COLUMNA ";:
1500 c$=INKEY$:IF c$="" THEN 1500
1510 IF c$<>CHR$(32) THEN PRINT #1,c
$:ELSE GOTO 1490
1520 FOR ret=1 TO 300:NEXT ret
1530 IF c$="?" THEN tf=TIME:GOTO 11
80
1535 IF c$="." OR c$="+" OR c$="&"
THEN CLS #1:SOUND 1,40,20:GOTO 1490
1540 c=VAL(c$)
1550 IF c<1 OR c>7 THEN CLS #1:SOUN
D 1,40,20:GOTO 1490
1560 LOCATE 23+10*ABS(flag),22:PRIN
T c$
1570 ccc$(m,flag)=c$:fff$(m,flag)=f
$
1580 CLS #1
1590 IF flag=1 THEN LOCATE 1,22:PRI
NT SPACE$(39)
1600 RETURN

```



**P**ara que tus dedos no realicen el trabajo duro, M.H. AMS-TRAD lo hace por ti. Todos los listados que incluyen este logotipo se encuentran a tu disposición en un cassette mensual, solicítanoslo.

# CINCO EN LINEA

## (Juego para dos)

Saturnino Cachón Rodríguez

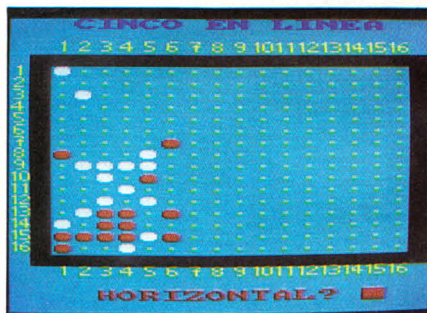
**Este último juego, del tipo de aquéllos que los ingleses, con su amor a las palabras sencillas, llaman «braintwist games», es el postre de AMSTRAD SEMANAL ha preparado para el especial Serie Oro de verano. Es como las cuatro en raya, sólo que más difícil, y está pensado, en un alarde de gremialismo, para que dos jugadores se rompan la cabeza uno contra otro sin la menor intervención del Amstrad, que actúa como árbitro impassible del fratricida encuentro.**



El juego consiste en colocar cinco fichas del mismo color en línea recta, bien de forma horizontal, vertical o diagonal.

Se juega con dos colores: blanco y rojo, de forma alternativa por cada jugador.

El color blanco siempre juega primero.



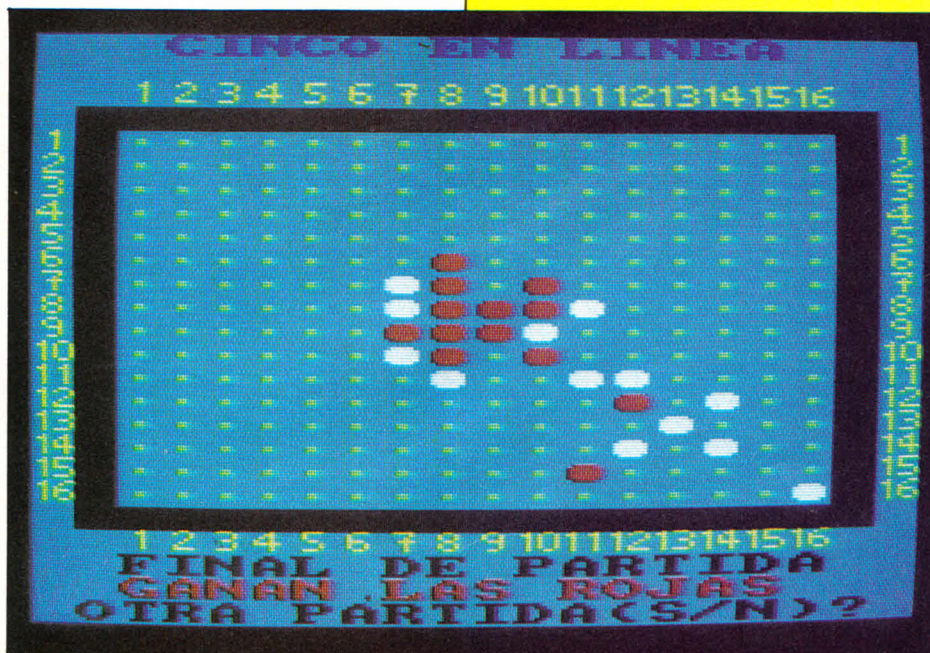
Cada jugador puede elegir cualquier punto del tablero, siempre y cuando no haya sido elegido con antelación.

### LISTA DE SUBROUTINAS

30-390	Dibujo pantalla
400-480	Pregunta por punto elegido
490-540	Localiza, dibuja y memoriza punto elegido
550-1390	Comprobaciones de juego
1400-1410	Cambia de jugador
1420-1510	Final de partida
1520-1610	Instrucciones iniciales

### LISTA DE VARIABLES

jug	Jugador (1 ó 2)
loca ()	Localización de ficha en tablero
a, b	Puntos de localización de tirada
x, y	Puntos de localización en pantalla



# Serie ORO

```

10 REM <<< CINCO EN LINEA >>>
20 * 01986: SATURNINO CACHON RODRIGU
EZ
30 GOTO 1530
40 REM <<< PANTALLA >>>
50 MODE 0: BORDER 0: INK 0, 13: INK 1, 2
6: INK 2, 6: INK 3, 0: INK 4, 24: INK 5, 18
: INK 6, 1
60 PAPER #1, 0: PAPER #2, 0
70 CLEAR
80 jug=1
90 DIM loca(24, 24)
100 WINDOW #1, 1, 20, 1, 22
110 WINDOW #2, 1, 20, 23, 25
120 SYMBOL AFTER 139
130 FOR n=1 TO 17
140 READ c, d, e, f, g, h, i, j, k
150 SYMBOL c, d, e, f, g, h, i, j, k
160 NEXT n
170 DATA 239, 0, 60, 126, 126, 126, 126, 6
0, 0
180 DATA 240, 8, 24, 8, 8, 8, 8, 0, 241, 2
4, 36, 4, 8, 16, 32, 60, 0, 242, 56, 4, 4, 24, 4
, 4, 56, 0, 243, 8, 24, 40, 72, 124, 8, 8, 0
190 DATA 244, 60, 32, 32, 24, 4, 4, 56, 0, 2
45, 28, 32, 32, 56, 36, 36, 24, 0, 246, 60, 4,
8, 28, 8, 8, 0, 247, 24, 36, 36, 24, 36, 36,
24, 0
200 DATA 248, 24, 36, 36, 28, 4, 4, 24, 0, 2
49, 38, 105, 41, 41, 41, 41, 38, 0, 250, 34, 1
02, 34, 34, 34, 34, 34, 0, 251, 38, 105, 33, 3
4, 36, 40, 47, 0
210 DATA 252, 46, 97, 33, 38, 33, 33, 46, 0
, 253, 34, 102, 42, 42, 47, 34, 34, 0, 254, 47
, 104, 40, 46, 33, 33, 46, 0, 255, 35, 100, 40
, 46, 41, 41, 38, 0
220 LOCATE #1, 4, 1: PEN #1, 6: PRINT #1
, "CINCO EN LINEA"
230 FOR n=1 TO 16
240 LOCATE #1, 2+n, 3: PEN #1, 4: PRINT
#1, CHR$(239+n);
250 NEXT n
260 LOCATE #1, 2, 4: PEN #1, 3: PRINT #1
, STRING$(18, 143)
270 FOR n=1 TO 16
280 LOCATE #1, 1, 4+n: PEN #1, 4: PRINT
#1, CHR$(239+n)
290 LOCATE #1, 2, 4+n: PEN #1, 3: PRINT

```

```

#1,CHR$(143)
300 LOCATE #1,19,4+n:PEN #1,3:PRINT
#1, CHR$(143)
310 LOCATE #1,20,4+n:PEN #1,4:PRINT
#1, CHR$(239+n)
320 NEXT n
330 LOCATE #1,2,21:PEN #1,3:PRINT #
1,STRING$(18,143)
340 FOR n=1 TO 16
350 LOCATE #1,2+n,22:PEN #1,4:PRINT
#1,CHR$(239+n);
360 NEXT n
370 FOR n=1 TO 16
380 LOCATE #1,3,4+n:PEN #1,5:PRINT
#1,STRING$(16,144)
390 NEXT n
400 REM <<< PREGUNTA >>>
410 LOCATE #2,5,2:PEN #2,jug:INPUT
#2,"HORIZONTAL";a
420 IF a<1 OR a>16 THEN CLS #2:GOTO
410
430 a=a+4:CLS #2
440 SOUND 1,200,10,6
450 LOCATE #2,5,2:PEN #2,jug:INPUT
#2,"VERTICAL";b
460 IF b<1 OR b>16 THEN CLS #2:GOTO
450
470 b=b+4:CLS #2
480 IF loca(a,b)<>0 THEN LOCATE #2,
4,2:PEN #2,3:PRINT #2,"PUNTO OCUPAD
0":SOUND 1,500,80,4,0,0,3:FOR n=1 T
O 500:NEXT:CLS #2:GOTO 410
490 REM <<< LOCALIZACION >>>
500 x=a-2
510 y=b
520 LOCATE #1,x,y:PEN #1,jug:PRINT
#1,CHR$(239)
530 SOUND 1,100,20,6
540 loca(a,b)=jug
550 REM <<< COMPROBACIONES >>>
560 ' horizontales
570 FOR n=0 TO 4
580 IF loca(a+n,b)<>jug THEN 610
590 NEXT n
600 GOTO 1420
610 FOR n=0 TO 4
620 IF loca(a-1+n,b)<>jug THEN 650
630 NEXT
640 GOTO 1420
650 FOR n=0 TO 4
660 IF loca(a-2+n,b)<>jug THEN 690
670 NEXT
680 GOTO 1420
690 FOR n=0 TO 4
700 IF loca(a-3+n,b)<>jug THEN 730
710 NEXT
720 GOTO 1420
730 FOR n=0 TO 4
740 IF loca(a-4+n,b)<>jug THEN 780

750 NEXT
760 GOTO 1420
770 ' verticales
780 FOR n=0 TO 4
790 IF loca(a,b+n)<>jug THEN 820
800 NEXT
810 GOTO 1420
820 FOR n=0 TO 4
830 IF loca(a,b-1+n)<>jug THEN 860

840 NEXT
850 GOTO 1420
860 FOR n=0 TO 4
870 IF loca(a,b-2+n)<>jug THEN 900
880 NEXT
890 GOTO 1420
900 FOR n=0 TO 4
910 IF loca(a,b-3+n)<>jug THEN 940
920 NEXT
930 GOTO 1420
940 FOR n=0 TO 4
950 IF loca(a,b-4+n)<>jug THEN 990
960 NEXT
970 GOTO 1420
980 ' diagonales
990 FOR n=0 TO 4
1000 IF loca(a+n,b-n)<>jug THEN 103
0
1010 NEXT
1020 GOTO 1420
1030 FOR n=0 TO 4
1040 IF loca(a-1+n,b+1-n)<>jug THEN
1070

```

```

1050 NEXT
1060 GOTO 1420
1070 FOR n=0 TO 4
1080 IF loca(a-2+n,b+2-n)<>jug THEN
1110
1090 NEXT
1100 GOTO 1420
1110 FOR n=0 TO 4
1120 IF loca(a-3+n,b+3-n)<>jug THEN
1150
1130 NEXT
1140 GOTO 1420
1150 FOR n=0 TO 4
1160 IF loca(a-4+n,b+4-n)<>jug THEN
1200
1170 NEXT
1180 GOTO 1420
1190 ' diagonales
1200 FOR n=0 TO 4
1210 IF loca(a-n,b-n)<>jug THEN 124
0
1220 NEXT
1230 GOTO 1420
1240 FOR n=0 TO 4
1250 IF loca(a+1-n,b+1-n)<>jug THEN
1280
1260 NEXT
1270 GOTO 1420
1280 FOR n=0 TO 4
1290 IF loca(a+2-n,b+2-n)<>jug THEN
1320
1300 NEXT
1310 GOTO 1420
1320 FOR n=0 TO 4
1330 IF loca(a+3-n,b+3-n)<>jug THEN
1360
1340 NEXT
1350 GOTO 1420
1360 FOR n=0 TO 4
1370 IF loca(a+4-n,b+4-n)<>jug THEN
1400
1380 NEXT
1390 GOTO 1420
1400 IF jug=1 THEN jug=2:GOTO 410
1410 IF jug=2 THEN jug=1:GOTO 410
1420 REM <<< FINAL DE PARTIDA >>>
1430 CLS #2
1440 LOCATE #2,3,1:PEN #2,3:PRINT #
2,"FINAL DE PARTIDA"
1450 IF jug=1 THEN LOCATE #2,3,2:PE
N #2,jug:PRINT #2,"GANAN LAS BLANCA
S": GOTO 1470
1460 IF jug=2 THEN LOCATE #2,3,2:PE
N #2,jug:PRINT #2,"GANAN LAS ROJAS"
1470 LOCATE #2,2,3:PEN #2,3:PRINT #
2,"OTRA PARTIDA(S/N)?"
1480 a$=INKEY$
1490 IF a$="S" OR a$="s" THEN 50
1500 IF a$="N" OR a$="n" THEN MODE
1:END
1510 IF a$<>"S" OR a$<>"s" OR a$<>"
N" OR a$<>"n" THEN 1480
1520 REM <<<<< INSTRUCCIONES >>>>>
1530 MODE 1:INK 0,13:INK 1,0
1540 LOCATE 13,3:PRINT "CINCO EN LI
NEA"
1550 LOCATE 12,5:PRINT "(Juego para
dos)"
1560 LOCATE 1,9:PRINT " 1 El juego
consiste en colocar cinco fichas
has en linea, bien en forma
horizontal, vertical o diagonal."
1570 LOCATE 1,14:PRINT " 2 Se jueg
a con dos colores: blanco y
rojo, de forma alternativa."
1580 LOCATE 1,18:PRINT " 3 El colo
r blanco siempre juega primero."
1590 LOCATE 5,23:PRINT "PRESIONA <E
SPACIO> PARA JUGAR"
1600 IF INKEY(47)<>0 THEN 1600
1610 GOTO 50

```



**P** ara 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, solicítanoslo.

# **GANA** **100.000 PESETAS** **CON MICROHOBBY** **AMSTRAD** **SEMANAL**

**P** orque 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.

- No se admitirán programas que contengan caracteres de control, debido a que no son correctamente interpretados por las impresoras.

- 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.

## **¡ENVIANOS** **TU PROGRAMA!**

Adjuntando los siguientes datos:

**Nombre y apellidos,**  
**dirección y teléfono.**

Indicando claramente en el sobre:

**AMSTRAD SEMANAL**  
 a HOBBY PRESS, S. A. La Granja, 39  
 Pol. Ind. Alcobendas (Madrid)

# MUSICA DE FONDO



Utiliza la capacidad de interrupciones del Amstrad para acceder cada cierto tiempo a una subrutina que interpreta la melodía. El tiempo entre interrupciones se puede variar de acuerdo con la melodía elegida de forma que no se produzcan lagunas en la cola de sonidos, procurando que sea el mayor posible.

Aunque se podrían haber leído las datas directamente desde la subrutina de sonido, se ha preferido soportar las notas en una matriz para evitar problemas si se usan más datos en el programa.

La descripción de rutina es la siguiente:

- 30 Iniciación.
- 40 Carga de la matriz de notas.

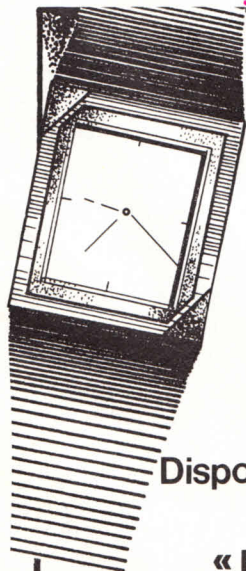
- 60 Instrucciones en la pantalla.
- 90 Comienzo de la melodía e inicio de las interrupciones.
- 100 Programa principal; en este caso es un contador.

```

10 / *** Musica de fondo ***
20 / ** por J.M.Rodriguez **
30 DIM notas(25,2):n=1:m=1:l=1:REST
ORE 50
40 FOR i=1 TO 25:READ notas(i,1),no
tas(i,2):NEXT
50 DATA 119,30,106,10,119,20,134,20
,142,20,134,20,119,40,159,20,142,20
,134,40,142,20,134,20,119,40,119,30
,106,10,119,20,134,20,142,20,134,20
,119,40,159,40,119,40,142,20,179,60
,0,80
60 MODE 1:LOCATE 2,2:PRINT "Mientra
s esta contando toca la melodía"
70 LOCATE 2,23:PRINT "Pulsa una tec
la para parar la musica"
80 PRINT " Pulsa otra vez para volv
er a la musica"
90 GOSUB 120:EVERY 40,1 GOSUB 120
100 WHILE INKEY$="":LOCATE 17,12:PR
INT USING "#####":m=m+1:WEND
110 a=REMAIN(1):l=-1:IF l=1 THEN 90
ELSE 100
120 SOUND 1,notas(n,1),notas(n,2),7
130 n=n+1:IF n=26 THEN n=1
140 IF SQ(1)>128 THEN 120
150 RETURN
    
```



- 110 Fin de la música y opciones de operador.
- 120 Rutina de sonido (subprograma).
- 140 Cuando está llena la cola de sonidos regresa al programa.



## ii RECUPERA LAS ASIGNATURAS PENDIENTES !

■ SOCIALES ■ NATURALES ■ MATEMATICAS ■ LENGUAJE ■

DE : 5' 6' 7'y 8' EGB

Programa para ejercicios de ortografía EDORTO

Programa para ejercicios de atención y comprensión EDACLE

Programa para ejercitar el instrumento lector EJEILE

Disponible en MSX y AMSTRAD.. 1800 ptas Cassette — 2750 Disco

«PRECIO ESPECIAL» CURSO COMPLETO EGB..... 3500 ptas.

PEDIDOS : **GOSLINE** CUARTELES, 43-1°-29002-MALAGA-TEL. 311877

... ..regalo de un reloj por cada cassette o disco

# 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.

## LA INSTRUCCION SUB EN CODIGO MAQUINA

*Mi problema es el siguiente: me gustaría saber si existe la instrucción de código máquina SUB rr,rr' (por ejemplo SUB HL,BC), y en caso afirmativo que me dijeran sus OPCODS.*

P.D.: Debido a que me urge conocer la respuesta por necesitarla para un programa sobre el que estoy trabajando, les adjunto un sobre franqueado para que me la envíen, confiando en que esto no les supondrá mucha molestia.

**Fernando Alvarez Serrano**

La instrucción que nos comenta en su carta no es posible ejecutarla en el microrprocesador Z80. Aunque existe una similar que es la siguiente:

**SBC HL,DE**

pero debemos decir, que su funcionamiento no es igual al de la instrucción SUB que Vd. nos propone, ya que en este caso se efectúa una resta con Carry, por lo que antes de

efectuar dicha operación, debemos asegurarnos de que el flag Carry esté a cero. Una forma de conseguirlo puede ser la siguiente:

SCF  
CCF

## MAPA DE MEMORIA DE PANTALLA DEL 6128

*Me dirijo a Vds., con el fin de solicitarles quieran informarme sobre lo siguiente:*

1.º *Mapa de memoria de pantalla en modo 2 del CPC 6128, como así coordinadas de destino en código máquina, en el mismo modo (2) del 6128.*

2.º *Si es posible conseguir y donde las publicaciones "A GUIDE TO CP/M PLUS", SOFT971 y "A GUIDE TO CP/M", SOFT159, como así sobre el FIRMWARE del 6128 y SISTEMA OPERATIVO, si Vds. me dicen que me informe en AMSTRAD ESPAÑA, no vale, ya que omiten dar esa información pese a mi consulta.*

3.º *Otorgar un DIEZ DE CLASIFICACION por vuestra publicación, está muy completa e interesante, sin Vds., nosotros los usuarios estaríamos perdidos, vuestra labor es inmejorable.*

4.º *Sólo falta, a mi criterio, un poco más extensión en el tema del código máquina, como así las modificaciones que sean necesarias para cada modelo 464, 664 y 6128.*

**Hugo R. Cruz Videla**

Al referirse al mapa de memoria de pantalla, suponemos que querrá decir las direcciones de inicio de fin de pantalla y cuáles son los saltos de línea a línea y de carácter a carácter. Así pues, a continuación le indicamos dichos datos:

INICIO DE PANTALLA: #C000

FIN DE PANTALLA: #FFFF

SALTO DE LINEA: 2048 BYTES

SALTO DE CARACTER: 80 BYTES

teniendo en cuenta que la dirección inicial de pantalla es alterable debido a los scrolls que se producen. Debemos decirle también que dichas direcciones son las mismas para cualquier modo de pantalla.

En cuanto a la forma de conseguir las publicaciones, le aconsejamos se dirija a una tienda de informática o una librería especializada en dichos temas.

## EL COMANDO "CALL"

*Después de felicitaros por vuestra completa revista (y sobre todo al apartado de Lenguaje Máquina-Ensamblador), os desearía realizar una pregunta que nunca nadie me ha sabido responder.*

*El problema está en el comando "CALL". La función de este comando la tengo bastante clara (semejante a la de GOSUB en Basic), pero me asaltó la duda al observar en un programa que el comando "CALL" tenía más de un parámetro utilizable (además de la dirección de memoria).*

*Les agradecería que me aclarasen mi terrible duda.*

**Ramón Reigada Sanz**

Los comandos Basic "CALL" aceptan más de un parámetro como Vd. habrá podido observar. Cuando se ejecuta un comando de este tipo, al entrar en el programa en máquina, el registro IX apunta a la dirección del último parámetro del comando, y el registro A contiene el número de parámetros introducidos, teniendo en cuenta que cada uno de los parámetros tiene una longitud de dos bytes. Así pues, si por ejemplo ejecutamos lo siguiente:

CALL, 10000,A,B

el registro IX apuntará a la dirección del parámetro B, y el acumulador contendrá el valor de 2.

## QUE ES UN MAPA DE MEMORIA

*Me dirijo a Vds. para que resuelvan dos dudas.*

*La primera: ¿qué se especifica en un mapa de memoria?*

*La segunda: quiero saber una o varias direcciones en donde puedo comprar un libro del Firmware del Amstrad CPC 464.*

**J. M. González**

En un mapa de memoria se especifican las siguientes cuestiones:

— Zona de RAM disponible para el programador.

— Memoria de pantalla.

— Zona de ROM del sistema.

— Zona de variables del sistema.

Respecto a tu segunda pregunta, te recomendamos dirigirte a cualquier tienda especializada en informática.

SOMOS MAYORISTAS

# MICRO-1

PRECIOS INCLUIDO IVA

C/ Duque de Sesto, 50. 28009 Madrid  
Tel.: (91) 274 53 80

(Metro O'Donnell o Goya)  
Aparcamiento gratuito en Felipe II

Ofertas en software: 2 programas al precio de 1 y además regalo fin de curso una calculadora completamente gratis. ¡¡Asombroso!!  
¿Verdad?

BAT MAN _____	<b>2.300</b> ptas.	KUNG-FU MASTER _____	<b>2.300</b> ptas.
ROCK'N LUCHA _____	<b>2.300</b> ptas.	SABOTEUR _____	<b>2.300</b> ptas.
YIER AR KUNG FU _____	<b>2.300</b> ptas.	PING PONG _____	<b>2.300</b> ptas.
THE WAY OF THE TIGER _____	<b>2.300</b> ptas.	MILLION II _____	<b>0000</b> ptas.
WEST BANK _____	<b>2.100</b> ptas.	OLE TORO _____	<b>2.300</b> ptas.
CAMELOT WARRIORS _____	<b>2.300</b> ptas.	TURBO ESPRIT _____	<b>2.100</b> ptas.
RAMBO _____	<b>2.300</b> ptas.	SABRE WULF _____	<b>1.650</b> ptas.
WORLD CUP (DISCO) _____	<b>3.300</b> ptas.	BATALLA DE LOS PLANETAS _____	<b>2.100</b> ptas.
MILLION II (DISCO) _____	<b>3.300</b> ptas.	SABOTEUR-COMBAT LINX DISCO _____	<b>3.300</b> ptas.
RAMBO-MATCH DAY (DISCO) <b>3.300</b> ptas.			

SOFTWARE DE REGALO (OFERTA 2x1)  
DECATHLON BEACH HEAD SOUTHERN BELLE  
DRAGONTORC

**LAPIZ OPTICO**  
**3.295 PTAS.**

**CASSETTE ESPECIAL ORDENADOR**  
**5.295 PTAS.**

**SINTETIZADOR DE VOZ EN**  
**CASTELLANO**  
**7.650 PTAS.**

**AMPLIACION DE MEMORIA ANTA 64 K.3**  
**12.500 PTAS.**

## IMPRESORAS 20% DE DESCUENTO SOBRE P.V.P.

TAPA DE METACRILATO 464	895
CABLE CENTRONICS	3.175
CABLE SEGUNDA UNIDAD D.	1.790
CABLE SEPARADORES 6128	1.975
INTERFACE RS232	9.265
CINTA VIRGEN C15	69

CABLE AUDIO	795
CABLE ADAPTADOR 2 JOYSTICK	2.390
CABLE SEPARADORES 464	1.390
CABLE SEPARADOR 8256	2.900
CABLE RS232	2.500
DISKETTES 3"	990

## PRECIOS EXCEPCIONALES PARA TU AMSTRAD CPC-464, CPC-6128, PCW-8256

¡¡LLEGARON LAS REBAJAS DE VERANO A MICRO 1!!

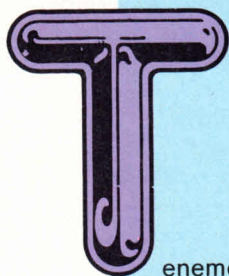
OFERTAS EN JOYSTICKS	
QUICK SHOT I	1.395
QUICK SHOT II	1.695
QUICK SHOT V	1.695

**PEDIDOS CONTRA REEMBOLSO SIN NINGUN GASTO DE ENVIO TEL. (91) 274 53 80**  
**O ESCRIBIENDO A: MICRO-1. C/ DUQUE DE SESTO, 50. 28009 MADRID.**

Tiendas y distribuidores grandes descuentos.  
Dirigirse a Dipromsa. C/ Galatea, 25. Tel. (91) 742 20 19 ó 742 79 68

# CASUALIDADES Y DEFINICIONES

**Parece que en el universo de los números todo sucede dentro de un orden lógico, pero en la vida hay multitud de hechos que ocurren por azar. Vamos a intentar introducir un poco de «casualidad» en nuestros programas mediante la función RND que nos va a generar una serie de números al azar. A esta «casualidad» la llamaremos «números aleatorios».**



tenemos que hacerles una confesión: el ordenador nunca producirá números al azar, sino una secuencia cíclica que hace que parezcan aleatorios. Es casi imposible conseguir una función que «cree» números totalmente casuales, pero con RND tenemos una gama de posibilidades lo suficientemente amplias como para dar esa impresión.

Si escribimos:

```
PRINT RND
```

el ordenador visualizará en la pantalla cualquier número aleatorio comprendido entre 0 y 1. Si repetimos esta instrucción varias veces los resultados obtenidos son diferentes pero nunca nos dará el valor máximo: el 1. Parece que RND va a tener entonces muchas limitaciones porque, ¿qué habrá que hacer para simular las tiradas de un dado, por ejemplo, que siempre serán números enteros y comprendidos entre 1 y 6?

La solución es sencilla: combinar el trabajo de RND con el de la función INT que recortará todos los valores obtenidos para dejarles en su parte entera. Vamos a hacerlo efectivo con el programa 1.

## Programa uno

Si multiplicamos el valor generado con RND por 6 obtendremos con la función INT valores enteros comprendidos entre 0 y 5. Para alcanzar

```
10 REM PROGRAMA I
20 CLS
30 INPUT "NUMERO DE TIRADAS ",TIRAD
AS
40 PRINT
50 FOR CONTADOR=1 TO TIRADAS
60 ALEATORIO=INT(RND*6)
70 DADO=ALEATORIO+1
80 PRINT DADO
90 NEXT CONTADOR
```

los valores del dado, ¿qué hacemos? Sencillamente sumarle 1 en la línea 70.

Sin embargo, el programa no es perfecto, tiene un error «garrafal» que descubrirá si lo ejecuta repetidamente después de resetear el ordenador (no olvide grabar el programa primero). En efecto, siempre aparece la misma serie, pero el **Amstrad** lo tiene previsto y nos sacará del apuro con la función RANDOMIZE, mediante la que informamos al micro de la posición de la secuencia desde la que queremos empezar.

Pruébelo con las instrucciones:

```
RANDOMIZE (1)
PRINT RND
```

y variando la expresión entre paréntesis cambiará también el número aleatorio obtenido. Intente no poner ningún número, es decir no darle la posición inicial. El ordenador le dará un mensaje:

Random number seed?

algo así como: ¿número aleatorio preseleccionado?, al que tendremos que contestar con el índice elegido.

La función RND también admite tres tipos de parámetros:

— Números positivos. Devuelve el valor del siguiente número aleatorio en la secuencia cíclica.

— 0. Nos devuelve el número anterior.

— Números negativos. Admite de -1 a -16. Nos entrega el valor correspondiente y hace que la secuencia empiece de nuevo.

# Primeros repastos

Le proponemos que insista en probar al máximo estas dos palabras claves y sus diferentes parámetros para conseguir en cualquier momento la secuencia de números aleatorios deseada.

Quizá se haya encontrado alguna vez con la necesidad de utilizar una expresión numérica complicada para obtener un determinado valor que asignaremos a una variable. Si realizamos los cálculos sólo una vez, no hay ningún problema: los hacemos y se acabó.

Pero a lo mejor en un programa le ha tocado repetir el trabajo muchas veces con distintos parámetros. El **Amstrad** ha previsto ahorrarnos tales repeticiones con la función DEF FN.

Todo lo que debemos hacer es crear una función con dicha instrucción seguida de un nombre y su definición real y después llamarla desde cualquier parte del programa, cada vez que la necesitemos, con FN y el nombre definido.

Por ejemplo, en la línea 30 del programa 2 hemos creado la función con el nombre «Pitágoras» y en las variables entre paréntesis le pasaremos los valores de los catetos, que obtendremos en las líneas 40 a 70, para que dicha función los utilice como parámetros.

## Programa dos

Calculamos el valor de la hipotenusa en la línea 80 (nos lo devuelve la función «Pitágoras») y lo almacenamos en la variable «hipotenusa».

Con este método de trabajo, aparte de ahorrarnos una serie de operaciones, conseguimos claridad.

```
10 REM PROGRAMA II
20 CLS
30 DEF FNPITAGORAS(X,Y)=SQR(X^2+Y^2)
)
40 INPUT "CATETO MAYOR: ",CATETO1
50 IF CATETO1<=0 THEN GOTO 40
60 INPUT "CATETO MENOR: ",CATETO2
70 IF CATETO2<=0 THEN GOTO 60
80 HIPOTENUSA=FNPITAGORAS(CATETO1,CATETO2)
90 PRINT
100 PRINT "HIPOTENUSA =";HIPOTENUSA
```

# Un número de película A partir del día 1 estreno en toda España

Batman, Bomb Jack, Commando...  
todos tus héroes preferidos juntos en este número,  
para divertirse a tope.

# MICRO

## Manía

Año 11 · N° 15      Sólo para adictos      300 Ptas.

Canarias, Ceuta y Melilla 256 ptas.



SPECTRUM	<b>ALIEN HIGHWAY</b>	MSX	<b>NIGHT SHADE</b>
	<b>SPINDIZZY</b>		<b>y además</b>
AMSTRAD	<b>FRANKIE GOES TO HOLLYWOOD</b>	SPECTRUM MSX	<b>Gunflight y Starquake</b>
	<b>BOMB JACK</b>		AMSTRAD COMMODORE
	<b>COMMANDO</b>		

### Utensilios y Cachivaches

UNA AUTENTICA ORQUESTA SINFONICA

**HISTORIA DE SOFT** *Aventura en el castillo del Conde Drácula*

*Pídelo en tu kiosco*

# 4X1

**paga uno y llevate cuatro**

## **AMSTRAD**

COMBAT LINX  
GREMLINS  
DUMMY RUM  
DRAGONTORC  
MATCH DAY  
BASEBALL  
FIGHTING WARRIOR  
MAP GAME  
YIE AR KUNG-FU  
HYPERSPORTS  
ZORRO  
SUPERTEST  
PING-PONG

## **COMMODORE**

BASEBALL  
DROPZONE  
BEACH HEAD  
HYPERSPORTS  
SUPER ZAXXON  
FIGHTING WARRIOR  
SPY HUNTER  
TAPPER  
BC-II  
BOUNTY BOB  
POLE POSITION

## **MSX**

DISC WARRIOR  
JET SET WILLY II  
SHOWJUMPER

## **SPECTRUM**

ZAXXON  
FRANKIE  
BLUE MAX

BRUCE LEE  
RAID OVER MOSCOW  
BASEBALL  
DRAGONTORC  
ASTROCLONE  
GYROSCOPE  
MAP GAME  
ZORRO

COSMIC WARTOAD  
N.O.M.A.D.  
BATTLE OF PLANETS  
DYNAMITE DAN  
LEYENDA AMAZONAS  
BRIAN BLOODAXE  
PSYTRAXX

.... **y mil títulos más**

**!!! absolutamente  
originales!!!**

# 500

**ptas.**

***sinclair store***

**SOMOS PROFESIONALES**

**BRAVO MURILLO, 2**  
(Glorieta de Quevedo)  
Tel. 446 62 31 - 28015 MADRID  
Aparcamiento GRATUITO Magallanes, 1

**DIEGO DE LEON, 25**  
(Esq. Núñez de Balboa)  
Tel. 261 88 01 - 28006 MADRID  
Aparcamiento GRATUITO Núñez de Balboa, 114

**AV. FELIPE II, 12**  
(Metro Goya)  
Tel. 431 32 33 - 28009 MADRID  
Aparcamiento GRATUITO Av. Felipe II