

AMSTRAD

Semanal

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

160 Ptas.

Canarias 165 pts.

LOS SECRETOS DEL CASSETTE, UNO POR UNO

¡NOVEDAD EN EXCLUSIVA! MAPA Y POKES DEL CAMELOT WARRIORS

SONIDO EN LENGUAJE MAQUINA (Y II)

Salva a la galaxia de un peligro mortal. Flash Fire es la clave.

EL TIEMPO NO PASA PARA TU AMSTRAD. ¿O, SI?



ORDEMANIA SOFT

por fin..

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 INFORMÁTICA
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
Tele. 455544 - 455533
20011 SAN SEBASTIAN

AMSTRAD

Sumario

Año II • Número 54 • 23 al 29 de Septiembre de 1986
 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

José A. Esteban

Cristina Gómez

Francisco Martín

Jesús Alonso

Pedro S. Pérez

Amalio Gómez

Alberto Suñer

Secretaría Redacción

Carmen Santamaría

Fotografía

Carlos Candel

Chema Sacristán

Portada

J. Igual

Ilustradores

J. Igual, J. Pons, F. L. Frontán,

J. Septien, Pejo, J. J. Mora

Edita

HOBBY PRESS, S.A.

Presidente

María Andriño

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

ROTEDEC, S. A. Ctra. de Irún.

Km. 12,450 (MADRID)

Fotocomposición

Novocomp, S.A.

Nicolás Morales, 38-40

Fotomecánica

GROF

Ezequiel Solana, 16

Depósito Legal:

M-28468-1985

Derechos exclusivos

de la revista

COMPUTING with

the AMSTRAD

Representante para Argentina, Chile,

Uruguay y Paraguay, Cia.

Americana de Ediciones, S.R.L. Sud

América 1.532. Tel.: 21 24 64. 1209

BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace

necesariamente solidaria de las

opiniones vertidas por sus

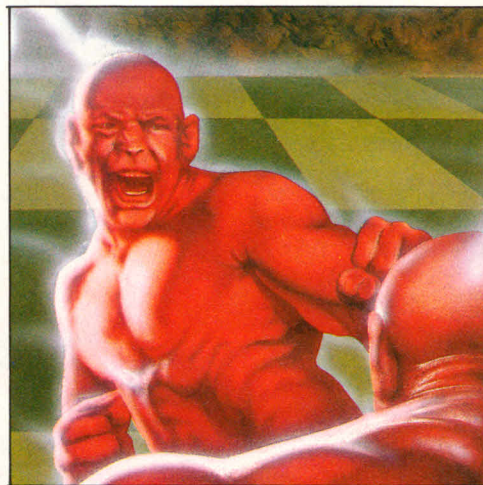
colaboradores en los artículos

firmados. Reservados todos los

derechos.

Primera plana 5

Tenemos noticias de nuevos Joysticks, y de dos casas de software muy importantes y conocidas: Compulogical y Micro World.



Código Máquina 18

Últimos toques a las rutinas de sonido para hacer música en lenguaje máquina.



Primeros pasos 6

Seguimos analizando la «trastienda» de los Amstrad, con la misma profundidad y desenfado que caracterizan a Primeros Pasos. Este estudio se revelará de gran utilidad en capítulos posteriores, Ya veréis.



Utilidades del programador 22

El cassette, y cómo usarlo, es algo que todo programador, sobre todo si tiene un 464, debe conocer. Por tanto, merece la pena leer con atención el presente artículo.

Mr. Joystick 12

Esta vez tenemos algo muy especial. Un completo análisis, con historia y mapa incluido, de uno de los mejores juegos para Amstrad: «Camelot Warriors», creado por Dinamic.

Serie Oro 26

Flash Fire es un juego de «Arcade» en la más pura tradición galáctica, siempre controvertida, ya conocida pero siempre, también, tremendamente viva. Os gustará.

micro BYTE

P.º de la Castellana, 179-1.º
28046 MADRID
Teléf.: 442 54 33/44



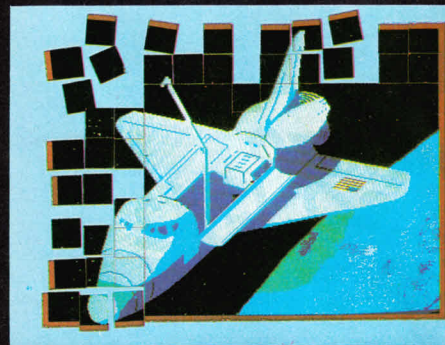
MERCENARIO

Cada pantalla es un nuevo desafío, una misión suicida, una dura batalla. Te encontrarás solo frente a compañías de asalto, aviones, morteros, bombas de mano, etc. Hace falta mucha sangre fría y mucha agilidad mental para salir con vida.
P.V.P. CASSETTE 2.000 Ptas.+I.V.A.
DISCO 2.900 Ptas.+I.V.A.



OBSIDIAN

La estación espacial Obsidian, construida en el interior de un asteroide se precipita hacia un agujero negro. Tú eres la única persona a bordo con el suficiente valor y experiencia para reactivar los sistemas de energía, arrancar los motores y salvar a tu tripulación.
P.V.P. CASSETTE 1.900 Ptas.+I.V.A.



PUZZLES

Por fin unos puzzles donde no se pierden piezas. Un reto a tu capacidad y a tu imaginación espacial. Un auténtico rompecabezas.
P.V.P. DISCO 2.900 Ptas.+I.V.A.

MIRRORSOFT



SPITFIRE 40

Trasládate a los años 40, Segunda Guerra Mundial, ponte a los mandos de un Spitfire. Siente la sensación de volar y combatir en aquellos aparatos donde la pericia del piloto era el 90% del éxito.
P.V.P. CASSETTE 2.200 Ptas.+I.V.A.
DISCO 2.900 Ptas.+I.V.A.

JUMP JET



JUMP JET

Estás en la cubierta de un portaaviones, debes despegar verticalmente en tu Harrier y, de repente, te encontrarás sólo en el aire, sin ver otra cosa que agua hasta que aparezcan los aviones enemigos dispuestos a destruirte.
P.V.P. CASSETTE 2.200 Ptas.+I.V.A.
DISCO 2.900 Ptas.+I.V.A.



VIAJE FIN DE CURSO

Tomy ha acabado el curso y se quiere ir de vacaciones, pero antes debe pasar por todas las aulas y laboratorios para recoger sus calificaciones. El colegio es muy grande y, si no le ayudas, Tomy se quedará sin viaje fin de curso.
P.V.P. DISCO 2.900 Ptas.+I.V.A.

COMPULOGICAL ABRE CAMINOS

Con una clara intención de no dormirse en los laureles, Compulogical ha firmado nuevos acuerdos con varias casas de software como Softek/Edge, Aackosoft, Leisure Promotions, Accolade, Epix.

Gracias a estos compromisos podremos contar en breve con juegos como Fairlight de Softex.

Este juego está diseñado con técnicas de construcción en tres dimensiones, y los gráficos, así como la acción, se desenvuelven en un ambiente de gran realismo. Todos los objetos que aparecen en pantalla tienen las características de un mundo real; nosotros tendremos cinco bolsillos donde llevarlos, y como es natural, cogiendo los más ligeros iremos más deprisa, y podremos llegar a un Castillo lleno de pistas que nos conducirá a... «El Resplandor».

Esta casa tiene además juegos para Spectrum, Msx y Commodore.

De Epix, no podemos dejar sin nombrar un entretenido y super bien realizado juego como el Winter Games.

En esta época estival es refrescante ver, y jugar, a patinar sobre hielo, esquiar, slalom, etc.

Todo esto con unos gráficos realmente impresionantes y un buen movimiento.

¡Lucha por la medalla de oro!

Con Leisure Promotions disfrutaremos de la posibilidad de hacernos nuestro propio equipo de super estrellas, para el cual podrá diseñar su propio uniforme, y sin más proclamarnos campeones.

Para este programa existen versiones en disco y cassette.

De momento Accolade o Aackosoft no tienen nuevos títulos para **Amstrad**, tal vez esto sea, en el caso de la segunda, por que se dedica principalmente a los MSX, pero no dudamos que teniendo en cuenta la importancia que tiene el mercado **Amstrad**, esta firma de software se plantee en un futuro próximo la posibilidad de hacer juegos para este ordenador.

MICRO WORLD PANASONIC Y MICROPRO

Micro World, compañía dedicada a la distribución de productos de microinformática, ha firmado sendos acuerdos con la firma japonesa Panasonic, y la empresa francesa Micropro.

De esta manera, Micro World, dispondrá de los diskettes de 3", 3 1/2", y 5 1/4", de Panasonic para su comercialización, así como del Paquete de Tratamientos de Textos Word Star de Micropro.

Micro World, que viene distribuyendo marcas tan importantes como **Amstrad**, Spectravideo o Sperry, cuenta ahora con otros productos de marcas reconocidas a nivel internacional como las ya citadas.

Primera PLANA

MANTENGA LIMPIO SU CPC 6128

First, S.A., distribuidor para España de la firma IAN, ha lanzado al mercado la lan Key-Cover, que pretende ser una útil funda de metacrilato que salvaguarde a nuestro 128 del nocivo polvo cuando no lo estemos utilizando, así como del temible derrame de café que se puede originar en un descuido cuando estemos descansando de un fatigoso «picado».

La lan Key-Cover es transparente y se ajusta a las formas del ordenador. Su precio es de 2.408 ptas., IVA incluido.



NUEVO JOYSTICK DE LA FIRMA ERGOJOYSTICK

Este nuevo joystick consta de cuatro botones de fuego para control de disparo, como de una desarrollada forma anatómica para el total ajuste de la mano a la empuñadura.

Tiene también un control fino de

definición para los juegos más famosos que poseamos, además de las utilísimas ventosas que confieren un excelente agarre.

El Ergojoystick VG-125 es compatible para **Amstrad**, Spectrum, Commodore 64 y 128, Spectravideo, etc.

La distribución de este periférico la realiza First, S.A., y su precio es de 1.900 ptas.

U.C.P., TAN EXACTA COMO UN RELOJ

En nuestra última reunión hicimos una pequeña excursión por el interior de nuestro ordenador. Empezamos a conocer, aunque de un modo bastante superficial, qué son y para qué sirven todas y cada una de esa especie de «pastillas» negras que lo componen. Vimos que siempre tendrá que existir un equipo físico que ejecute correctamente las instrucciones que le vamos dando por medio de las líneas de un programa. Hardware y Software, ¿recuerda?

Una aclaración antes de continuar. Al hablar de instrucciones no nos estamos refiriendo a las Basic, que tan bien conocemos, o a las de cualquier otro lenguaje «simbólico» de alto o bajo nivel, sino a las escritas en código, o lenguaje máquina, que son las únicas que un ordenador entiende sin problemas, por muy sofisticado y complejo que sea.

Decodificación de órdenes en secuencias

Ya saben el aspecto que tiene: un verdadero lío. Son una sucesión de ceros y unos que el Z80 del Amstrad maneja con suma facilidad, pero que para nosotros representa una verdadera tarea de chinos (y de muchos chinos) el poder interpretarla correctamente.

Bueno, pues el micro toma estas instrucciones y todavía las descompone en otras más elementales: leer una instrucción, acceder a la memoria, enviar una señal que ponga en marcha la unidad de disco, etc... A estas órdenes, que son las más sencillas que podemos encontrarnos, desde ahora en adelante las vamos a llamar «secuencias».

Podemos deducir, con bastante tino y acierto, que cada una de las instrucciones en lenguaje máquina con la que se encuentre estará formada por varias «secuencias». Ejecutándolas de forma adecuada y procurando no saltarse ninguna, nos proporcionarán el resultado que nosotros

esperábamos si hemos tenido la paciencia de analizar lo que representan, o lo que quieren decir, para nosotros los humanos las cifras «binarias» que el ordenador maneja con tanta soltura. (Anote el adjetivo «binario»).

Si hemos hablado también, en algún escondido rincón de esta serie de artículos, de que la Unidad de Control determinará «cuándo» tiene que ejecutarse una cierta secuencia o instrucción. No es muy descabellado pensar que llevará en su interior algún mecanismo o reloj, electrónico por supuesto, que genere unos «impulsos» o señales eléctricas con unos intervalos de tiempo entre ellas que sean siempre iguales, y que nos permitan medir de alguna forma el tiempo que va transcurriendo desde que hemos conectado el ordenador.

¡Ah!, su Amstrad también lo tiene. ¿No se lo cree? Bien, póngalo en marcha. ¿Qué observa? No observamos nada anormal. En la pantalla sale la cabecera que ya conocemos y punto.

Aunque no parece que existan signos externos de que el Z80 está trabajando, lo que ocurre en realidad es muy diferente. De momento hay un mensaje escrito en la pantalla que no desaparece, un teclado que está dispuesto para que a través suyo demos al ordenador el comando que estemos pensando, y así todo cuanto quiera.

El reloj interno

Pero además se ha puesto en marcha el reloj interno de nuestro ordenador. ¿Sigue sin creérselo? Para sacarle de su incredulidad, si es que existe, le proponemos un experimento. Teclee:

PRINT TIME

y a continuación pulse RETURN. Ahora sí que observamos algún cambio, aunque nada más sea porque se va a ejecutar una instrucción PRINT.

En efecto, se ha visualizado un número en la pantalla. Pero así, de momento, ese número no nos dice mucho, ¿verdad? Inténtelo de nuevo:

PRINT TIME

Ahora la cantidad obtenida es diferente, ya que aparece un número distinto del anterior. Y así ocurrirá cuantas veces repitamos la operación.



Quizá el principal componente de un Amstrad, y de muchos de los ordenadores caseros que circulan por el mercado, es su «microprocesador» Z80: es el corazón de todo el sistema. Su función está muy clara ya que «sólo» se encarga de «controlar» y «procesar».

Y para llevar a cabo todo esto con éxito necesitará estar unido físicamente (bueno, mejor eléctricamente) a todas las partes que componen la máquina, ya que será quien les diga cómo y cuándo han de ponerse en funcionamiento. Además ha de estar al tanto de todo cuanto ocurra en ellas, por muy alejadas que se encuentren. Por eso le llamamos CPU, que son las siglas de «Unidad Central de Proceso», pero en inglés, claro.

Para realizar las labores de coordinación y control, está, cómo no, la Unidad de Control. Su misión es interpretar las instrucciones que hay almacenadas en la memoria y, partiendo de ellas dar las órdenes oportunas para que el resto de los componentes las realicen.



Primeros PASOS

Podríamos pensar que la cifra que nos sale es simplemente aleatoria y que no tiene más sentido su variación. Vuelva a escribir el comando y analice el resultado comparándolo con los anteriores. Se encontrará con que no se trata de unos números elegidos al azar por el ordenador, sino que parecen seguir una cierta lógica: de momento van en orden creciente.

O sea, que su valor aumenta conforme lo haga el tiempo transcurrido desde que encendimos el ordenador. Así que podemos asegurar sin miedo a equivocarnos demasiado, que el número que aparece en la pantalla cada vez que el ordenador ejecuta el comando

PRINT TIME

está íntimamente ligado a su reloj interno. Y así es, TIME nos está dando, cada vez que lo invocamos, una medida del tiempo transcurrido. Diremos que TIME nos devuelve el contenido de un contador que se va incrementando cada vez que el reloj de su ordenador le envía una determinada señal. Teniéndolo en cuenta, imaginamos que no le resultará muy complicado realizar una especie de «marcador» real de tiempo. Si no es así, eche un vistazo al Programa 1 y lo podrá ver de un modo práctico.

Programa uno

Su funcionamiento es bastante sencillo. Sacamos primero el valor del contador y se lo asignamos a una variable a la que llamaremos «inicial» (línea 30). Ya hemos determinado con ello el origen de «tiempo».

A continuación hacemos otra medida, utilizando de nuevo la función TIME, y en esta ocasión asignamos el número obtenido a la variable «actual» (línea 50). Si se observa la diferencia entre el valor inicial y éste que acabamos de calcular, y lo introducimos en «tiempo», lo que contendrá esta última variable será lo que acabamos de decir: una diferencia de tiempo. ¿De acuerdo?



Y si luego imprimimos el resultado obtenido, y que está almacenado en la variable «**tiempo**», recuerde, en un lugar fijo de la pantalla (*líneas 70 y 80*) habremos logrado alcanzar el objetivo deseado: un «**algo**» que nos vaya indicando el tiempo transcurrido. Ejecute el programa y lo verá.

Pero, al menos a nosotros, se nos está ocurriendo una pregunta: ¿en qué unidades nos lo está reflejando?

Es evidente que no son horas, ya que la velocidad con la que cambia de valor es bastante rápida, pero decir que distinguimos perfectamente si son décimas o centésimas de segundo nos parece ya demasiado atrevido.

Paciencia, hombre, que ya se lo vamos a «**contar**». Hemos puntualizado anteriormente que TIME es una función que nos devuelve el contenido de un contador. Añadamos algo más. Dicho contador se va incrementado en 1 cada vez que el reloj interno del ordenador ha pasado 1/300 segundos, o la tercera parte de una centésima de segundo. ¡Qué rápido!

Sí, mucho. Tenga en cuenta que en la «**CPU**» (*le suena, ¿verdad?*) se realizan unas operaciones muy elementales, tanto como podamos imaginar, pero que se ejecutan a una gran velocidad permitiendo realizar instrucciones muy complejas en un tiempo relativamente corto. No podemos comparar la escala de rapidez de ejecución de una «**secuencia**» por parte del ordenador con los resultados que pueden apreciar cualquiera de nuestros sentidos; nosotros somos muchos más lentos.

Bueno, pues corregir el Programa anterior para que nos marque exactamente, y en segundos, el tiempo real que va transcurriendo está ya «**chupado**». Cambie la línea 60 por la siguiente:

```
60 tiempo = INT ((actual-inicial)/300)
```

y ahora tendremos ante nosotros un verdadero reloj, o mejor un verdadero segundero, ya que el tiempo que señale estará medido en dichas unidades. Utilizamos la función INT en la nueva línea 60 para evitar que aparezcan largos números decimales que nos desfiguren la medida realizada. De esta manera es mucho más concreto el resultado, aunque menos exacto, pero contribuye a añadir un poco de claridad y hasta elegancia a la respuesta.

¿Se imagina lo bueno que podría resultar aplicar este contador al juego de arcade que siempre hemos soñado realizar? Tendríamos que afinar nuestros reflejos al límite para conseguir salir victoriosos en la lucha contra el cronómetro. O también seríamos capaces de controlar el tiempo transcurrido en cualquier programa-juego deportivo y ver de esta manera si logramos batir algún récord, simplemente colocarnos entre las mejores marcas del día, o bien hacer sonar el pitido final en un par-

tido de fútbol disputado con nuestro ordenador. En fin, tenemos un montón de posibilidades.

Es necesario también que comentemos sus aspectos negativos. Aunque el contador en teoría se incrementa cada 1/300 segundos, esto no es del todo exacto. Hay que tener en cuenta que el micro gasta un tiempo en realizar todo el proceso con los datos que nosotros le damos. Bueno, pues durante esta operación no cuenta el tiempo.

Pero ya hemos dicho que el **Amstrad** posee un microprocesador que es infinitamente más rápido que nosotros. Ejecuta instrucciones individuales a tal velocidad que nos da la impresión que se trata de grandes y complejas tareas de alto nivel en lugar de tratarse de centenares de pequeñas operaciones o «**secuencias**» elementales que resultan de la composición de las anteriores.

```
10 REM PROGRAMA I
20 CLS
30 inicial=TIME
40 WHILE -1
50 actual=TIME
60 tiempo=actual-inicial
70 LOCATE 20,1
80 PRINT tiempo
90 WEND
```

Así que podemos dar como válida la precisión con la que funciona el reloj que acabamos de fabricar, ya que, para nosotros, los pobres y limitados humanos, una milésima de segundo no va a ningún sitio.

Si quiere, y le apetece, puede hacer unos cuantos cambios en el Programa anterior y conseguir que no sea solamente un exacto segundero, sino que se comporte como un verdadero reloj, marcándonos también las horas y los minutos. De esta manera el «**invento**» resultará sin duda un poquito más completo y la medida del tiempo se ajustará a lo que tradicionalmente estamos acostumbrados.

Para hacerlo sin ningún problema bastaría con crearnos tres contadores auxiliares, o variables, donde iríamos almacenando cada uno de los tipos de unidades que vamos a manejar —horas, minutos y segundos— y de momento los igualaríamos a cero.

¿Qué hacer después? Si ha seguido detenidamente todo lo anterior no tendrá ninguna pega en contestar usted mismo esta pregunta. Bastaría con hacer sucesivas medidas

del tiempo transcurrido en 1/300 de segundo, mediante la función TIME, y transformarlas posteriormente a las unidades elegidas. ¿Verdad que resulta sencillo? Pues manos a la obra.

¿Ya lo tiene? Nosotros también lo hemos estado intentando, no crea, y el resultado de nuestro trabajo queda reflejado en el Programa 2.

Programa dos

Su modo de funcionamiento es muy simple. Crea e inicializa a cero los tres contadores de los que hablamos mediante las líneas 30 a 50. Hasta aquí sin ningún problema.

Tomamos, como antes, la primera medida de tiempo en la 60, y lo almacenamos en la variable «inicial» y entramos en un bucle sin fin cuya misión es ir haciendo sucesivas medidas de tiempo y actualizar los datos del reloj que aparece en la pantalla.

¿Cómo obtenemos los segundos transcurridos? Pues muy sencillo, calculamos la diferencia entre los contenidos del contador de tiempo en el momento inicial, que está almacenado en la variable «inicial», y lo que vale en el instante que estamos haciendo la medida y que cuidadosamente lo tenemos guardado en la variable «actual» (original, ¿no?), y lo dividimos por 300. Bueno, pues ya tenemos los segundos.

En el momento que la variable «segundos» llegue a valer 60, nuestro programa tendrá que incrementar en una unidad el valor que esté almacenado en «minutos» ya que justamente habrá transcurrido un minuto más. Y así lo hace en la línea 160.

Y no sólo eso. También necesitaremos «poner a cero» el segundero y comenzar otra vez el proceso tomando un nuevo origen de tiempo (no olvide la variable «inicial», es preciso que su valor se actualice en este mismo instante).

Algo semejante podríamos decir para cuando se cumpla la condición de que hayan pasado 60 minutos: no es muy complicado deducir que tenemos que aumentar en uno el contenido de la variable «horas» además de todo lo anterior (actualizaciones, puestas a cero, etc., etc., etc.).

Pues de todo esto se encarga la rutina comprendida entre las líneas 140 a 190. La dejamos en sus manos para que sea analizada exhaustiva-

mente, si es posible. Pero le aconsejamos que no se quede ahí. Vaya un poquito más lejos e intente mejorar este pequeño programa añadiéndole más posibilidades como, por ejemplo, que marque además los días o las décimas de segundo, o que en lugar de aparecer unas cifras que nos indiquen la hora, tengamos sobre la pantalla un reloj de los clásicos con esfera y manecillas incluidas. O también cualquier otra cosa que se le esté ocurriendo. No ponga límites a su imaginación y pase sus ideas a la memoria del **Amstrad**.

```

10 REM PROGRAMA II
20 CLS
30 segundos=0
40 minutos=0
50 horas=0
60 inicial=TIME
70 WHILE -1
80 actual=TIME
90 segundos=INT((actual-inicial)/300)
100 IF segundos=60 THEN GOSUB 140
110 LOCATE 15,1
120 PRINT USING "##.";horas;:PRINT USING "##.";minutos;:PRINT USING "#";segundos
130 WEND
140 REM Rutina de ajuste y actualización
150 segundos=0
160 minutos=minutos+1
170 IF minutos=60 THEN minutos=0:horas=horas+1
180 inicial=TIME
190 RETURN

```

¿Ha quedado demostrado suficientemente que en el interior del ordenador hay «algo» que marca y regula el tiempo? Sinceramente, creemos que sí.

Además de saber cuándo ha de ejecutarse una instrucción, la Unidad de Control tendrá que estar informada de cuál es la orden que en determinado instante tiene que atender.

Para ello tiene otro «contador» pero ahora, en lugar de recoger el tiempo transcurrido desde que el ordenador se puso en funcionamiento, lo que contiene es la dirección donde se encuentra la siguiente sentencia a ejecutar.

De ahí la extrae y después de analizarla, ya sabe qué circuitos deben intervenir en esta ocasión y qué conexiones debe realizar para dar las órdenes oportunas a los demás elementos del ordenador y que éstos se pongan en marcha para cumplir correctamente sus encargos. En pocas palabras, acaba de repartir las tareas.

Una vez que la instrucción ya se ha ejecutado, lo que llamaremos «contador de instrucciones» tendrá que

Primeros PASOS

actualizarse convenientemente y «apuntar» a la siguiente operación a realizar para comenzar este ciclo nuevamente.

Bueno, por el momento terminamos pero no sin antes dejarle algo de entretenimiento. Se trata del Programa 3, que está basado en lo que hemos visto a lo largo de este artículo. Pero además servirá también para entretenerle, relajarle, o quizá ponerle más nervioso todavía, y probar si sus reflejos están a punto y en forma después de las vacaciones (quien las haya tenido).

Programa tres

Se trata simplemente de pulsar la tecla que contenga la letra que aparece en la pantalla lo más rápidamente posible. Se mantendrá visible durante un tiempo cuya duración de-

```

10 REM PROGRAMA III
20 MODE 1
30 LOCATE 10,25
40 INPUT "DIFICULTAD 1/5 ",dificultad
50 IF dificultad<1 OR dificultad>10 THEN GOTO 20
60 FOR contador=1 TO 10
70 MODE 0
80 codigo=97+INT(RND*26)
90 LOCATE 10,12
100 PRINT CHR$(codigo)
110 GOSUB 210
120 NEXT contador
130 CLS
140 LOCATE 8,5
150 PRINT "PUNTOS"
160 LOCATE 10,12
170 PRINT puntos
180 LOCATE 6,21
190 PRINT "DE NUEVO ?"
200 IF INKEY$="s" THEN RUN ELSE IF INKEY$(">") THEN GOTO 200 ELSE END
210 REM Rutina de retardo y análisis
220 inicial=TIME
230 tiempo=0
240 WHILE tiempo<=5/dificultad
250 IF INKEY$=CHR$(codigo) THEN puntos=puntos+1:RETURN
260 tiempo=(TIME-inicial)/300
270 WEND
280 RETURN

```

pendrá del grado de dificultad que haya elegido. El estudio y análisis del Programa es otra cosa diferente y eso ya corre de su cuenta, ¿comprendido?

Así que, sin más, nos despedimos hasta la próxima.

José Manuel Rodríguez González

TUBO EN 3D

Eduardo Cabrera Choclán.

Este programa no tiene otra finalidad que la de mostrar, una vez más, los increíbles gráficos que se pueden conseguir en el MODE 2 (alta resolución) con un Amstrad.

Modificando algunos valores de las líneas 40-80-120 y 160 se pueden conseguir otros gráficos diferentes.

Estructura del programa:

- 10-40 Dibuja un tubo en tres dimensiones.
- 10-90 Dibuja una especie de agujero en 3 dimensiones.
- 100-130 Dibuja un amplio cráter.
- 140-180 Dibuja un cono cortado a la diagonal
- 190 Mensaje final del programa.

```

IDEA541 CA
10 MODE 2:CLS:PRINT "TUBO EN 3 DIMENSIONES":DEG
20 FOR n=1 TO 360
30 ORIGIN 320,200
40 PLOT 100*COS(n),100*SIN(n):DRAW
100+100*COS(n),100+100*SIN(n)
50 NEXT
60 CALL &BB06:CLS:PRINT "AGUJERO 3 DIMENSIONES":DEG
70 FOR n=1 TO 360
80 PLOT 100*COS(n),100*SIN(n):DRAW
200*COS(n),100+80*SIN(n)
90 NEXT
100 CALL &BB06:CLS:PRINT "EL MAS PROFUNDO CRATER":DEG
110 FOR n=1 TO 360
120 PLOT 200*COS(n),100*SIN(n):DRAW
300*COS(n),100+80*SIN(n)
130 NEXT
140 CALL &BB06:CLS:PRINT "HAS VISTO ALGUNA VEZ ESTO?":DEG
150 FOR n=1 TO 360
160 PLOT 100*COS(n),100*SIN(n):DRAW
300*COS(n),100+20*SIN(n)
170 NEXT
180 CALL &BB06
190 CLS:PRINT " ESTO ES SOLO UN EJEMPLO MAS DE LOS SENSACIONALES GRAFICOS QUE SE PUEDEN CONSEGUIR CON UN AMSTRAD CPC
-464--472--664--6128.":CALL &BB06
    
```

Un programa de entretenimiento no ha de ser necesariamente largo y complicado. Este, diseñado a medir la rapidez de reflejos, puede ser una muestra.

Después de pulsar una tecla, a petición del ordenador, y de una demora variable, se imprime en cualquier lugar de la pantalla un asterisco. Se debe entonces (pero no antes), pulsar lo más rápidamente posible una tecla, y nos dirá la velocidad de respuesta y su calificación. Si se pulsa antes de salir el asterisco resultará nulo el intento.

La rutina base utiliza la función TIME y es extremadamente corta (líneas 170-180), lo que permite incluir, además, formato de la presentación (líneas 90-100 y 190-200), y perforaciones adicionales, tales como posicionamiento al azar del asterisco (líneas 110-130), tiempo variable de demora (línea 130), control de tecleo prematuro (línea 150), tintas (línea 60), calificación (líneas 70-80) etc., en un número de líneas realmente reducido.

```

10 /
20 / ** Reflejos **
30 /
40 / ** J. M. Rodriguez **
50 /
60 MODE 1:INK 1,26:INK 2,15:INK 3,10,20:BOARDER 5
70 FOR i=1 TO 9:READ a$(i):NEXT
80 DATA excelente,notable,bueno,aceptable,regular,lento,lentisimo,destroso,fatal
90 PEN 3:LOCATE 13,1:PRINT "Pulsa una tecla"
100 WHILE INKEY$="" :WEND:CLS
110 PEN 1:a=RND*3000+2000
120 x=RND*40+1:y=RND*25+1
130 FOR i=1 TO a:NEXT
140 LOCATE x,y:PRINT "*" :b=TIME
150 IF INKEY$<>"" THEN LOCATE 14,25:PRINT "Intento nulo":WHILE INKEY$<>"" :WEND:GOTO 90
160 WHILE INKEY$="" :WEND
170 b=ROUND((TIME-b)/300,3)
180 c=INT(b*10):IF c>9 THEN c=9 ELSE IF c<1 THEN c=1
190 PEN 2:LOCATE 1,25:PRINT "Tiempo=";TAB(15);"seg.":TAB(23);"Calif.:"
200 PEN 1:LOCATE 8,25:PRINT b:LOCATE 31,25:PRINT a$(c);
210 GOTO 90
    
```

DEFINICION DE VENTANAS

Luis Peña Díaz

Propósito del programa: Creación de caracteres por el usuario.

Explicación del funcionamiento:

- Líneas 20-60 Definición de ventanas
- 70-190 Bucle WHILE encargado de recibir del teclado y por la ventana 1 los 0 y 1 de los bytes correspondientes. Dibuja por la ventana 2 el carácter gráfico. Por la ventana 3 aparecerá la matriz binaria. Para los bytes a hexadecimal y los introduce en un vector llamado igual.
- 200-250 Presenta por la ventana 4 el comando Basic "SYMBOL", listo para ser usado, quedando indeterminado el número de carácter.

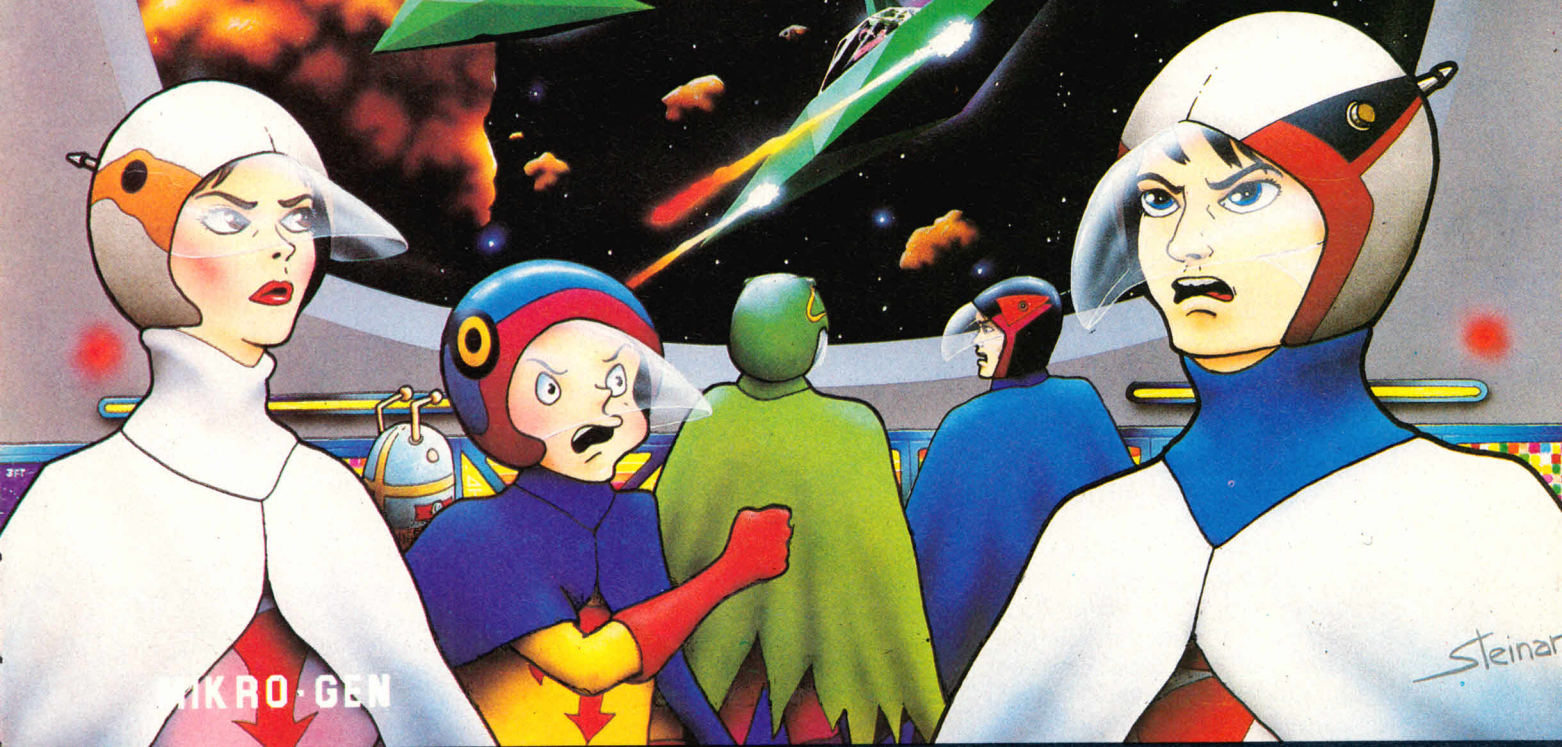
```

10 MODE 2
20 REM DEFINICION DE VENTANAS
30 WINDOW#1, 1,80,1,9:PAPER #1,0:PEN #2,13:REM PETICION DE DATOS
40 WINDOW#2, 6,21,12,20:PAPER #2,13:PEN #2,0:REM CARACTER GRAFICO
50 WINDOW#3, 45,69,12,20:PAPER #3,13:PEN #3,0:REM MATRIZ BINARIA DEL SIGNO
60 WINDOW#4, 15,75,22,22:PAPER #4,13:PEN #4,0:REM PRESENTACION COMANDO SYMBOL
70 REM CREACION DEL SIMBOLO
80 WHILE n<8
90 n=n+1:CLS #1:decimal=0
100 PRINT#1, "FILA NUMERO";n;" :CON UN 1, PIXEL ENCENDIDO, CON 0, PIXEL APAGADO"
110 FOR i=8 TO 1 STEP -1
120 INPUT#1, byte(i)
130 IF byte(i)<>0 AND byte(i)<>1 THEN EN 120
140 decimal=2^(i-1)*byte(i)+decimal
150 IF byte(i)=0 THEN PRINT #2, CHR$(143)+CHR$(143); ELSE PRINT #2, SP C(2);
160 PRINT#3, byte(i);
170 NEXT i
180 hexadecimal$(n)="%"+HEX$(decimal)
190 WEND
200 REM PRESENTACION DEL COMANDO SYMBOL
210 PRINT#4, "SYMBOL X";
220 FOR m=1 TO 8
230 PRINT #4, ",",hexadecimal$(m);
240 NEXT m
250 CLS#1:END
    
```

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

BATTLE OF THE PLANETS

UNETE AL "COMANDO G" EN SU ULTIMA AVENTURA CONTRA ZOLTAR EN UN MARAVILLOSO JUEGO REALIZADO CON UNOS GRAFICOS TRIDIMENSIONALES QUE HAN DE SERVISTOS PARA SER CREIDOS.



BATTLE OF PLANETS ES EL PROGRAMA SELECCIONADO PARA EL CAMPEONATO INTERNACIONAL DE JUEGOS DE ORDENADOR. CONSIGUE LA MAXIMA PUNTUACION ESPAÑOLA Y PARTICIPA EN LA FINAL DE LONDRES. (LAS BASES DEL CONCURSO ESTAN EXPLICADAS EN EL PROGRAMA)

DISTRIBUIDOR EXCLUSIVO PARA ESPAÑA ERBE SOFTWARE C/. STA. ENGRACIA, 17. 28010 MADRID, TFNO.: (91) 447 34 10
DELEGACION BARCELONA, AVDA. MISTRAL, N.º 10 - TFNO.: (93) 432 07 31

CAMELOT WARRIORS

En el reino de Camelot, han aparecido cuatro extraños objetos que vienen del futuro. Sólo el más valiente de los caballeros de la tabla redonda, osa intentar averiguar el secreto al que conducen.

Ha pero existe el software español? Hasta ahora bien pudiéramos decir que no. Entre la dilatada gama de juegos que hemos tenido oportunidad de ver, pocas obras hemos visto con el marchamo «**Made in Spain**».

Haciendo memoria podemos hablar de tres títulos: **ROCKY, WEST BANK y SIR FRED**.

Tres juegos bien distintos, en los que la calenturienta mente de los creadores hispanos nos ha llevado por los más insospechados decorados.

En Rocky, nos encontramos en un dramático combate de boxeo, en el cual encajamos directos con toda la cara, y nuestros puños demoledores castigan la mandíbula de nuestro adversario.

Sir Fred, es una aventura de búsqueda por los recónditos parajes de un castillo, en el que nuestro intrépido espadachín se las ve y se las desea para llegar a la terraza que ocupa Julieta.

West Bank, está ambientado en el oeste americano. En esta aventura nos encontramos tras la ventanilla de un banco y con la inestimable ayuda de nuestro colt 42, debemos hacer una cuidadosa selección de los clientes que visitan nuestro local.

Dos son las casas responsables de esta «**invasión de software ibérico**»; Dinamic y Made in Spain software, de los segundos no hemos tenido más noticias desde el lanzamiento de sir Fred, su primera y única obra por ahora (*esperemos que no sea su obra postuma*).

De Dinamic, nos llega su última producción para **Amstrad**: Camelot Warriors. Los guerreros de Camelot son primos lejanos de la mítica saga de los guerreros de la tabla redonda, valerosos caballeros cuya más alta meta era recuperar el Santo Grial.

Pero el hecho de ser primos, que no hermanos, no les quita nada de mérito a nuestros entrañables guerreros; el Santo Grial ha sido sustituido por cuatro objetos muchísimo más valiosos; el fuego que no quema, el espejo de la sabiduría, el elixir de la vida y la voz de otro mundo.

Gracias a un hechizo mágico, nos encontramos en el interior de una pesada coraza y comenzamos nuestro interminable viaje por las tierras de **Camelot**.

Estamos a los pies del árbol de Medinazaratz y sólo sabemos que la tierra se acaba ante nuestros pies. Antes de que nos dé tiempo a pensarlos dos veces, un ave de la noche vuela amenazadoramente hacia nosotros.

Empuñando a dos manos la pesada espada que completamente nuestro atuendo de guerrero, damos un certero golpe en la cabeza del monstruo volador, y respiramos de nuevo.

Antes de ser atacados por más criaturas hostiles, nos lanzamos hacia lo desconocido. Tras una corta caída tocamos de nuevo suelo firme y emprendemos el camino siempre avanzando hacia delante.

Saltamos sobre las mortales cabezas andantes que salen a nuestro paso, demasiado bajas para ser alcanzadas por nuestra espada, y un nuevo cambio de nivel interrumpe nuestra marcha.



Ahora tenemos dos elecciones: hacia arriba, o dejarnos caer de nuevo hasta tocar el suelo; elegimos la solución buena y saltamos hacia el próximo risco, desde él, otro saltito y llegamos a una pequeña plataforma llena de peligros.

Saltando subimos por una empinada cuesta, donde de un certero sablazo eliminamos a una avispa gigante, y volvemos a saltar hacia delante alcanzando una nueva meseta. Avanzamos sin descanso evitando cabezas andarinas y abatiendo las avispas gigantes que nos acosan.

Unos metros nos separan del final de la plataforma, y de repente nuestros ojos contemplan estupefactos el primero de los objetos mágicos que



Nuestro anfitrión, esta vez, es el todopoderoso Neptuno, que empuñando su tridente, espera cómodamente sentado a que le traigamos su presente.

Nos ponemos a la búsqueda, y tras largos paseos por las isletas y plataformas, cuando ya nuestros brazos están cansados por el continuado ejercicio de destripar pirañas, llegamos ante el segundo objeto mágico.

Ante nuestros ojos, dentro de una misteriosa caja vemos algo que se mueve, misteriosas imágenes salen de ella y una cara horripilante nos habla.

Nuestro hombre retrocede aterrado ante lo que ven sus ojos, pero en un acto de arrebató (*digno de verdaderos valientes*), cierra los ojos y se dirige hacia la caja mágica. Las voces arrecian e intentan alejarnos, pero el poder de nuestro traje y nuestra determinación hace que logremos apoderarnos de ella.

En nuestras manos, el efecto mágico desaparece y podemos transportarla sin peligro: hemos llegado hasta **«el espejo de la sabiduría»**.

Rápidamente lo llevamos hasta los pies de Neptuno, el cual nos transporta hasta la tercera dimensión de las simas del Reino de **Camelot**.

El aire se hace irrespirable; sin lugar a dudas cualquier mortal sucumbiría en cinco minutos. Nosotros, gracias a los poderes de nuestro traje podemos seguir avanzando hacia nuestra próxima meta.

En el Reino de las profundidades, hemos de conseguir el elixir de la vida con el cual el guardián de este nivel nos dejará pasar a la cuarta dimensión.

Caminamos con rumbo desconocido, hasta encontrar alguna pared que nos sirva de referencia. Criaturas de la noche, arañas del tamaño de un elefante nos acompañan con no muy buenas intenciones.

Cuando nuestra espada ha dado buena cuenta de ellas, escalamos hasta un soporte iluminado y desde éste saltamos hacia próxima plataforma, en la que nuestros atónitos ojos encuentran la tercera forma del futuro.

Una estructura metálica de superficie cilíndrica, envuelta en vistosas formas en rojo y blanco, atrapa nuestros sentidos; no hay ningún signo de movimiento, cautelosamente nos acercamos hacia ella y llegamos a tocarla.

En su parte superior hay una extraña anilla; con decisión la empuña-

Mister JOYSTICK

mos energicamente y tiramos hacia nosotros, la fuerza de nuestro traje nos permite arrancarla. Con ella en la mano, presenciamos el prodigio: un extraño ruido y una lluvia de burbujas son despedidas por el hueco dejado por la anilla.

Desde el suelo nos lanzamos hacia el cilindro metálico, y bebemos ávidamente de su contenido. Sus efectos son instantáneos, una rara perturbación recorre nuestro cuerpo y nuestras agotadas fuerzas se reestablecen al momento.

Nos apoderamos del extraño cilindro y buscamos nuestro camino hacia el guardián de las profundidades. En el otro extremo de la caverna nos aguarda el dragón que vigila la entrada a la cuarta dimensión.

Cuando le entregamos el elixir, las puertas de la cuarta dimensión son abiertas y penetramos en el Castillo de Camelot.

Espesos muros de piedra franquean nuestro camino, pesadas columnas nos conducen hasta el salón donde se encuentra la famosa tabla redonda. Proseguimos nuestro camino rodeados de armaduras y algún que otro fantasma.

Llegamos al final de la sala y saltamos sobre una mesa, desde ella divisamos los dos altos muros que nos llevarán hasta el último de los objetos mágicos.

Escalamos los muros saltando de uno a otro, aprovechando los pequeños salientes que cubren las paredes, y de nuevo los poderes de nuestro traje nos permiten saltar con una coraza de 200 kg de peso.

Cuando llegamos al final, nuestros ojos son sorprendidos con una extraña visión: sobre una pequeña mesa se encuentra un objeto de color rojo cuya forma no puedo describir, estoy ante **«la voz de otro mundo»**.

Paralizado espero a que me hable o a oír algo, pero mis esperanzas caen en saco roto, todo sigue en calma. Con paso decidido me aproximo ante la última de las visiones y alargo la mano hacia ella.

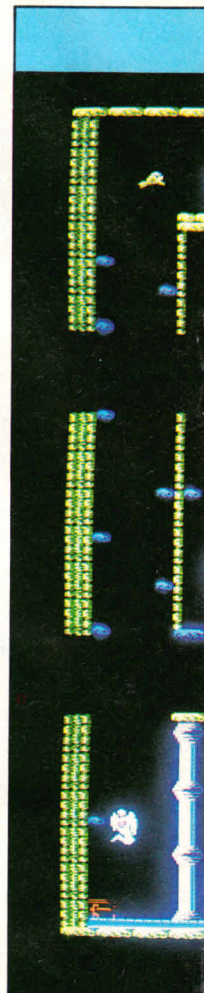
Con la voz en mi poder, me dirijo a presentarme al rey Arturo, descendiendo por las paredes, cruzo el hall del

nos llevarán a nuestro destino. Estamos ante **«el fuego que no quema»**, inseguros avanzamos hacia él y nos aventuramos a tocarlo, luego nos apoderamos de él y nos marchamos hacia el emplazamiento del Druida.

Cuando le entregamos el objeto que portamos, somos transportados a las profundidades del lago. La magia de nuestra coraza nos permite caminar por las profundidades y respirar sin necesidad de oxígeno: **«el fuego que no quema»** nos ha hecho anfibios.

Abriéndonos pasos entre pirañas de dos metros y medusas eléctricas, caminamos por las plataformas inmersas en las simas del lago de Camelot.

MAPA DEL BOSQUE Y DEL LAGO



castillo y por las relucientes escaleras de mármol, ascendiendo a los pisos superiores, varias escaleras más y llego a la cámara del rey.

Ante su presencia extraigo «**la voz de otro mundo**» y se la entrego. Cuando va a cogerla, un ruido ensordecedor se apodera de la sala, como el relincho de 100 caballos, y

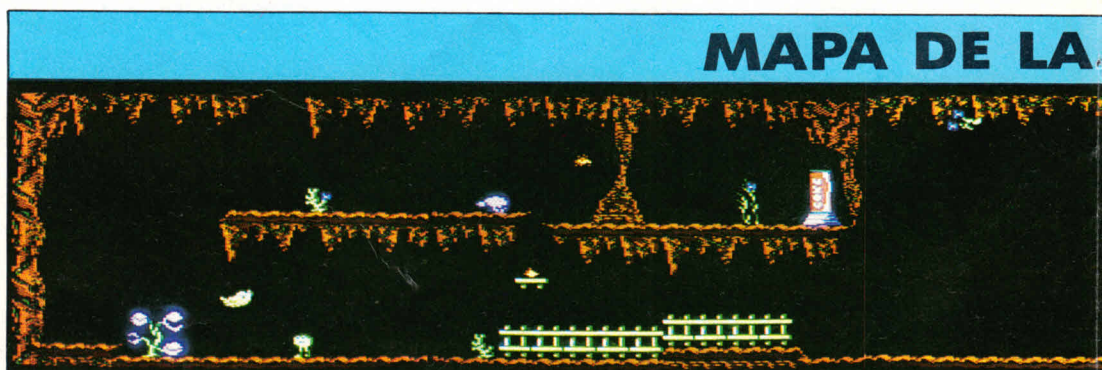
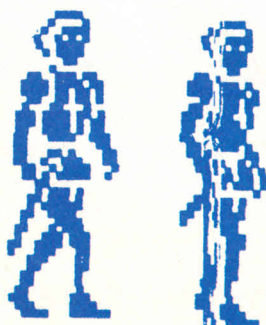
el extraño artefacto emite sus cánticos rítmicamente, pareciendo que quiere hacer estallar mis tímpanos, desenfundando la espada y me dispongo a descargar su peso sobre el misterioso artefacto.

RING.....RING.....RING.....RING

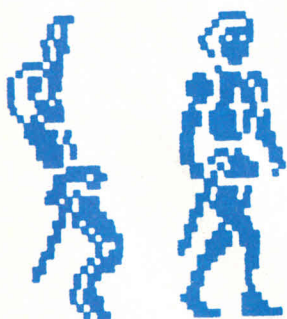
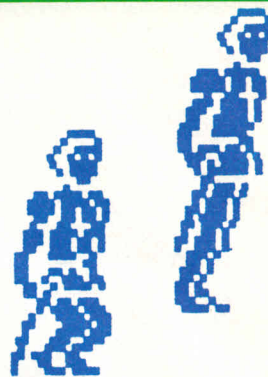
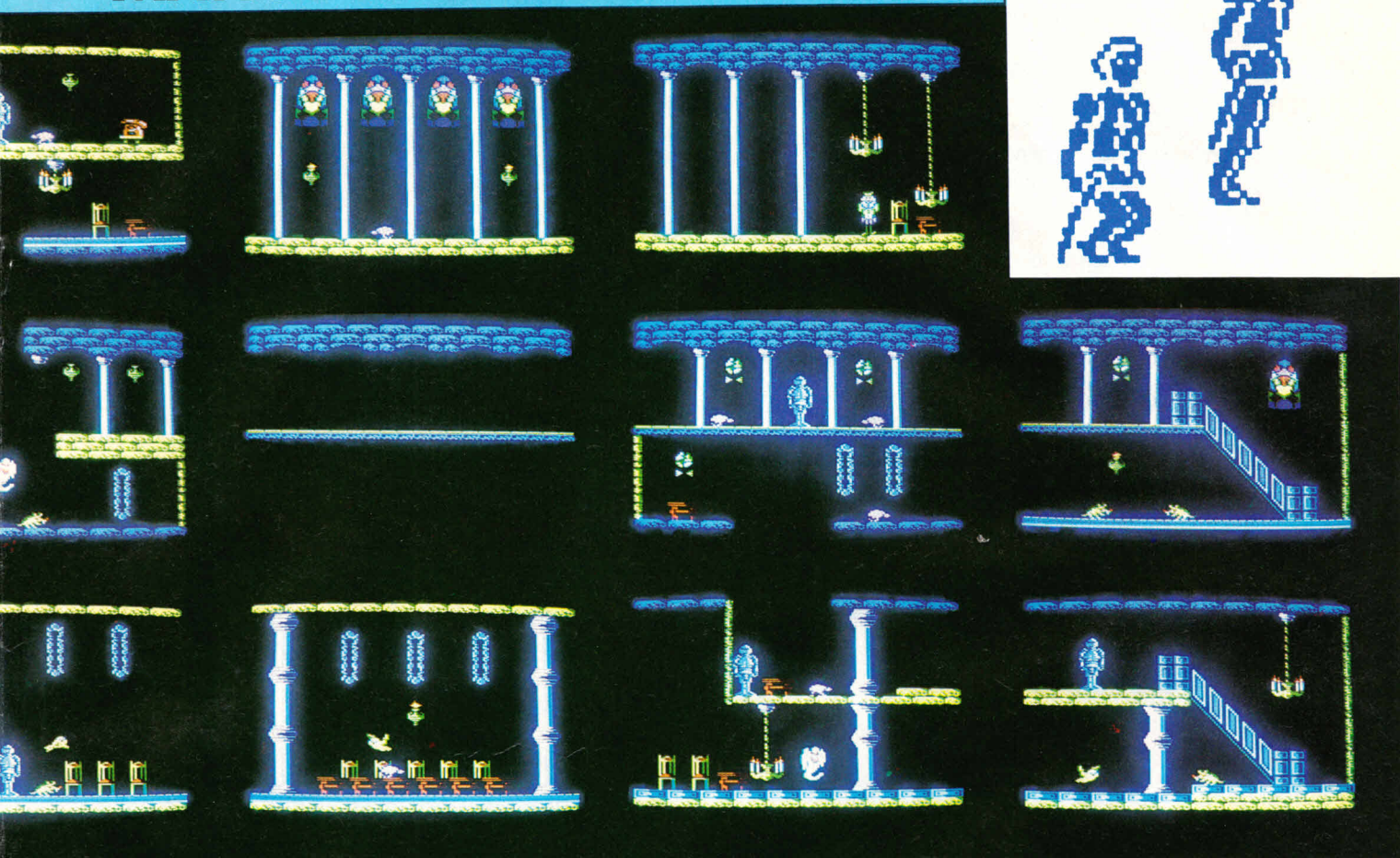
Me despierto en mi habitación empapado en sudor, el teléfono no de-

ja de sonar, con una mano lo des-cuelgo y «**la voz del otro mundo**» me advierte que son las 10, 25 minutos, 30 segundos.

Tumbado en la cama miro hacia el techo y me encuentro con una forma espectral que se encuentra en su mismo centro; «**el fuego que no quema**», es en mi habitación de 100 vatios.



MAPA DEL CASTILLO



Aterrorizado por mis descubrimientos miro al frente, como los buenos guerreros, y acciono el mando a distancia: en **«el espejo de la sabiduría»** están poniendo Dinastía.

Despavorido me doy la vuelta y me dispongo a esconderme debajo de las sábanas, antes de sumergir la cabeza, un bote cilíndrico que descansa en mi mesilla capta mi atención, **«el elixir de la vida»** me sirvió anoche para hacerme un cubata. **«Me he traído todos los objetos de la pesadilla.»** Salto de la cama y hecho a correr hasta el salón, donde a salvo de tanta visión perturbadora, **«el humo de la felicidad»**, que en el si-

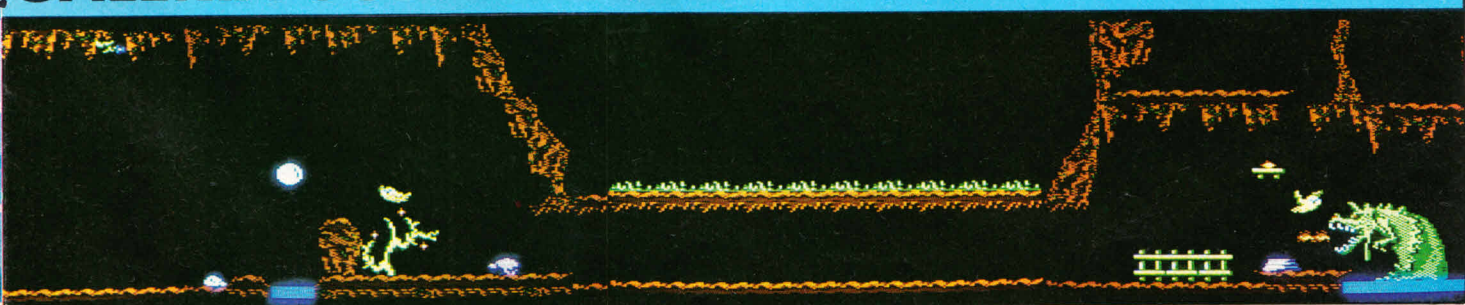
glo XX viene en paquetes de 20, me hará recuperar la calma.

Dinamic hasta ahora nos había deleitado con buenas creaciones de software. Su principio en el mundo comercial, como el de las más prestigiosas casas de software, fue haciendo programas para el Spectrum.

Entre ellos tenemos la famosa trilogía de las aventuras de Mindiana Jhon's: Babalisa, Saimazoon y Profanation.

Posteriormente otros títulos como Rocky, West Bankm, Ole toro, Sgrizan, etc. fueron lanzados al mercado, algunos de los cuales han sido adaptados para el **Amstrad**.

GALERIA SUBTERRANEA



Estamos hablando de programas de calidad, hechos por un verdadero equipo de profesionales, pero hasta ahora, los resultados nunca habían alcanzado la brillantez del Camelot Warriors.

Sin duda, nos encontramos ante la mejor aventura de la firma, desde luego en esta creación Dinamic ha dado el DO de pecho.

Nos sumergimos en un mundo en el que debemos pasar por los cuatro niveles del Reino de Camelot, en cada uno de ellos debemos recoger un objeto del futuro situado en alguna parte, y entregárselo al guardián de dicho estrato para poder pasar al siguiente.

Para ello disponemos de un número de vidas limitadas, y de nuestra espada que nos sacará de las más comprometidas de las situaciones.

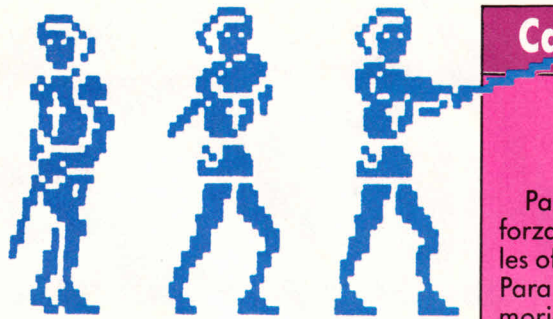


Nuestro guerrero acorazado dispone de un reducido repertorio de movimientos, lo cual facilita enormemente la tarea de llevarle por los recónditos parajes que visitamos.

Fácilmente controlable con el joystick, podemos caminar hacia la derecha o la izquierda, saltar en cualquiera de esos dos sentidos y descargar el golpe de nuestra espada.

La dificultad del juego reside en la confección de los parajes que constituyen cada uno de los mundos que visitamos, cada uno integrado por un considerable número de pantallas; las cuales debemos conocer a la perfección para descubrir el recorrido que nos llevará a las puertas del siguiente mundo subterráneo...

La ambientación de los distintos mundos y pantallas, es de una gran vistosidad y colorido. En esta ocasión Dinamic, casa caracterizada por la



calidad de sus gráficos, se ha esmerado al máximo.

Cada reino que visitamos está caracterizado por su distinta constitución; en el bosque todo es vegetación y animales silvestres, el paso a las profundidades del lago nos sumerge en una oscuridad azulada donde pirañas y algas amenazan nuestra marcha. El paso a las profundidades de la tierra nos lleva a un paraje desolado, con escasos habitantes todos ellos deseosos de comer carne fresca.

Por fin la llegada al último estadio, el castillo de Camelot, donde muros robustos, pesadas techumbres, columnas de desmesurado espesor y magníficas balaustradas, nos llevan a los aposentos del rey Arturo.

Nos encontramos ante un conjunto de 134 pantallas, en cada una de las cuales se ha cuidado todo al máximo, tratando cada zona con un colorido acertado y que dota al conjunto de auténtica variedad y belleza.

A parte de la construcción del juego en sí y su dinámica interna, debemos elogiar la idea de colocar en un mundo medieval los cuatro objetos del futuro que presiden el desarrollo de la aventura. Sin duda una idea calenturienta de esas que surgen en las largas horas nocturnas de programación.

La voz de otro mundo, El espejo de la sabiduría, El fuego que no quema y El elixir de la vida, tienen su miga. *10 puntos para Dinamic.*

Viendo Camelot Warriors, tenemos un destacado ejemplo de lo que debía de ser el software español, no por desgracia lo que es actualmente, esperemos ver más programas de la misma calidad y belleza plástica.

Una aventura larga, en la que descubrir nuevas zonas inexploradas, y encontrar la forma de superar esos obstáculos que nos impiden continuar, nos dará horas y horas de diversión, llevando a nuestro guerrero acorazado a través de los desconocidos mundos, que la mente de sus creadores ha tenido el privilegio de dejarnos compartir.

Cargador del Camelot

Para aquellos que no quieran esforzarse excesivamente en la victoria, les ofrecemos un cargador del juego. Para utilizarlo basta cargarlo en memoria, teclear TAPE si se es usuario del 664 ó 6128, y RUN, poner la cinta del juego en el cassette, y responder con «s» a las opciones que queramos habilitar durante el juego.

Opciones:

VIDAS INFINITAS: respondiendo con «s» tu caballero tendrá 255 vidas, es decir, casi casi invencible.

NO DANGER: respondiendo «s» a esta opción ninguno de los peligros hara mella en tu caballero, ha alcanzado la inmortalidad.

Tras haber elegido las opciones, la carga de la cinta se efectuará normalmente. Suerte y a jugar ¡La victoria está próxima!

Al cierre de este artículo, recibimos la noticia de que se está preparando la versión inglesa del juego, próximamente en el mercado Británico. Según la noticia esta nueva versión va a salir al mercado considerablemente dulcificada, con un grado de dificultad notablemente inferior a la versión española. Que le vamos a hacer, la conquistadora sangre española no corre en todas las venas.

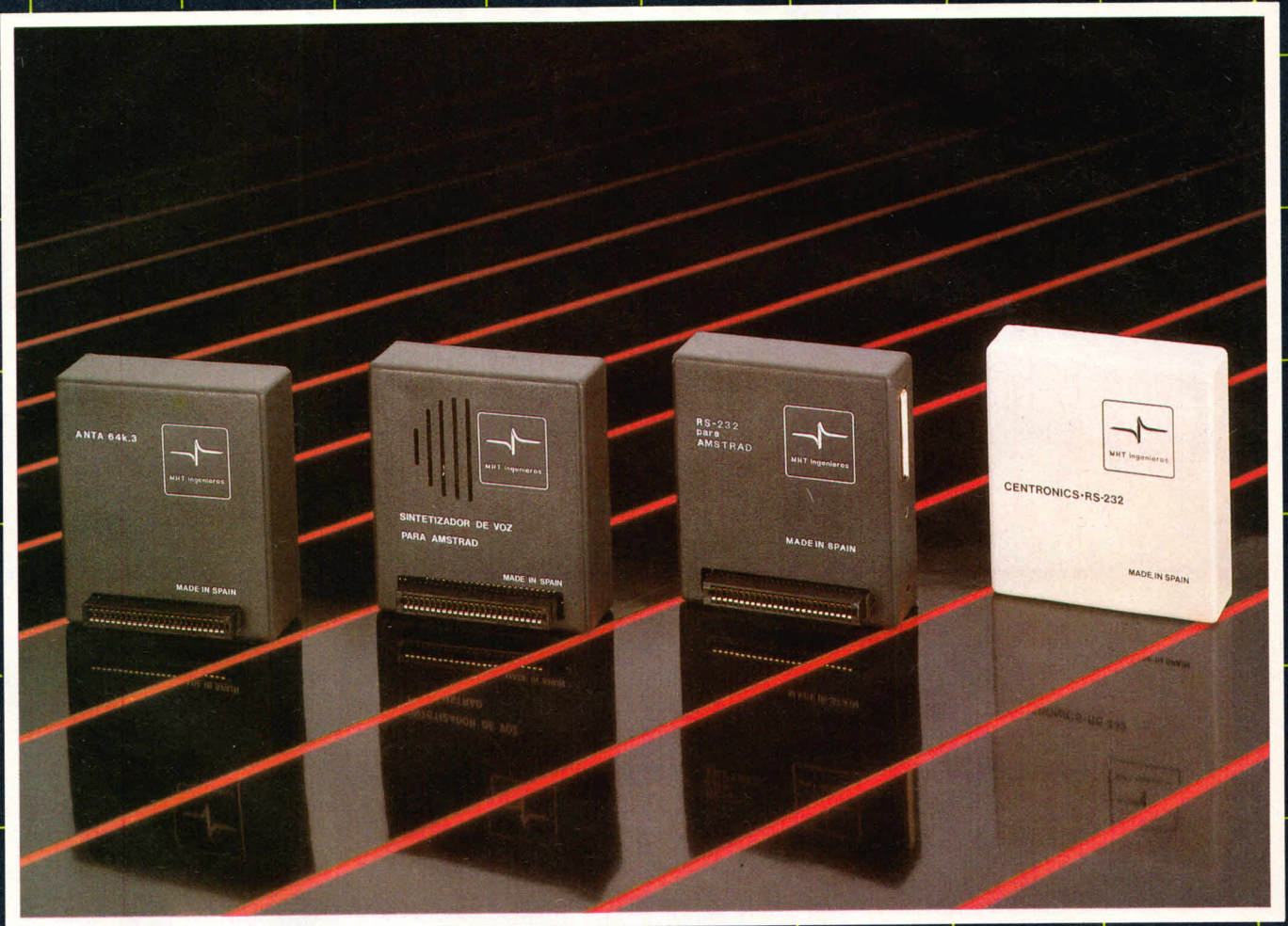
```

10 RESTORE
20 MEMORY 9999:MODE 2
30 FOR n=36330 TO 36342:READ a:POKE
n,a:NEXT
40 LOCATE 1,10:PRINT"Infinitas vidas
";:INPUT a$:io=53:IF a$="s" THEN iv
=255
50 LOCATE 1,10:PRINT"No danger
";:INPUT a$:nd=1:IF a$="s" THEN nd
=0
60 MODE 0:INK 0,0:BORDER 0:LOCATE 2,
10:PRINT"Espera un momento"
70 LOAD":warriors
80 POKE 36098,234:POKE 36099,141:POK
E 36331,iv:POKE 36336,nd
90 CALL 36000
100 DATA 62,53,50,90,1,62,1,50,175,1
7,195,74,1

```

Periféricos
para AMSTRAD

CONECTAMOS CON TUS IDEAS.



PERIFERICOS PARA LOS MODELOS CPC 464, CPC 664 Y CPC 6128

- ANTA 64 K.3 Ampliación de memoria, buffer de impresora y Ram Disk.
- SINTETIZADOR DE VOZ

El programa que controla este sintetizador, contiene las reglas básicas de pronunciación en castellano y permite su funcionamiento, tanto en modo directo, como bajo el control de un programa.

- RS-232-C
Permite comunicar el ordenador con impresoras y plotters con entrada serie, modems, y otros ordenadores.

PERIFERICOS PARA LOS MODELOS PCW 8256 Y PCW 8512

CENTRONICS-RS 232

Proporciona al ordenador dos canales de comunicación:

- Canal paralelo (centronics) para el manejo de impresoras.
- Canal serie (RS-232) para comunicar con otros ordenadores, modems, plotters, etc.).

MHT ingenieros



Distribución: L.S.B., S.A. Sánchez Pacheco, 78 - 28002 Madrid.

Para mayor información, escribir a:
L.S.B., S.A. Sánchez Pacheco, 78 - 28002 Madrid.

Nombre/Apellido

Domicilio

Localidad

Provincia

C.P.

Profesión

Edad

Información sobre: Periféricos Amstrad otros

MUSICA EN LENGUAJE MAQUINA

En el capítulo anterior iniciábamos el estudio del sonido desde código máquina. Hoy dejaremos por finalizado el tema de la música, repasando lo que no se ha visto todavía, como son las envolventes de tono y de volúmen.



Para definir una envolvente de volumen desde Basic, nos servimos del siguiente comando:

ENV N,S,S,S,S

donde N deberá ser el número de la envolvente que se desea definir, y los parámetros S indican cada una de las secciones que puede contener dicha envolvente.

Cada una de estas secciones de envolvente consta de 2 ó 3 parámetros, según el tipo de envolventes que se deseen utilizar.

Si se quiere definir una envolvente de software, entonces el número de parámetros que debe contener cada una de las secciones, será de tres. Si por el contrario, deseamos definir una envolvente de hardware, en este caso serán dos el número de parámetros de que debe constar cada una de las secciones de envolvente.

Veamos cuáles son los parámetros a indicar en cada uno de los casos:

Envolventes de software:

- 1.—Número de escalones
- 2.—Altura de cada escalón
- 3.—Tiempo de pausa

Envolventes de hardware

- 1.—Envolvente de hardware
- 2.—Periodo de envolvente

La diferencia fundamental que existe entre cada uno de los dos tipos de definición, es que en el primer caso, nosotros somos lo que definimos la envolvente según nuestra conveniencia, y en el segundo caso, tomamos una envolvente definida previamente en nuestro ordenador.

Todo lo dicho hasta el momento corresponde a la definición de envolventes de volúmen desde Basic, veamos ahora qué pasos deberemos dar

para efectuar dicha definición, pero desde código máquina.

Para ello, nos servimos de la siguiente rutina del firmware:

DEFINE UNA ENVOLVENTE DE VOLUMEN. #BCBC

Se usa para definir cada una de las 15 amplitudes de volumen posibles.

Condiciones de entrada.

En el acumulador se debe indicar el número de la amplitud de volumen deseado, y en el registro doble HL, se deben indicar la dirección de los datos para dicha envolvente.

Condiciones de salida.

En el caso de que la envolvente se haya definido correctamente:

- Carry flag verdadero
- HL contiene la dirección de los datos + 16
- A y BC se pierden

Si el número de envolvente ha sido erróneo:

- Carry flag falso
- A, B y HL se preservan

En cualquier caso:

— Se corrompen DE y los demás flags, y los demás registros se preservan.

Código MAQUINA

Vamos a ver a continuación cómo deben ser colocados los datos de cada envoltente que deseemos definir, en la memoria, así como el significado de cada uno de ellos.

Byte 0: Número de secciones de la envoltente.

Bytes 1-3: Primera sección

Bytes 4-6: Segunda sección

Bytes 7-9: Tercera sección

Bytes 10-12: Cuarta sección

Bytes 13-15: Quinta sección

En el caso en que se desee definir una envoltente de software, el contenido de cada sección, deberá ser el que se indica a continuación:

Byte 0: Número de escalones

Byte 1: Altura de cada escalón

Byte 2: Periodo de pausa

Si lo que se desea es definir una envoltente de hardware, los datos que deben contener dichos bytes, serán los siguientes:

Byte 0: Envoltente de hardware

Byte 1: Periodo de envoltente

Sólo nos queda ya por ver, la definición de las envoltentes de tono.

Desde Basic, podemos definir dichas envoltentes a través del siguiente comando:

ENT N,S,S,S,S

El significado de dichos parámetros, es semejante al visto anteriormente para las envoltentes de volumen.

Cada una de las secciones 'S' de envoltente, puede constar de dos o tres parámetros, en el caso de que sean tres, éstos serán:

- 1.— Número de escalones
- 2.— Altura de cada escalón
- 3.— Tiempo de pausa

si por el contrario, se utilizan únicamente dos de dichos parámetros, su significado será:

- 1.— Periodo de tono
- 2.— Tiempo de pausa

Para producir este mismo efecto, pero desde código máquina, disponemos de la siguiente rutina del firmware:

DEFINE UNA ENVOLVENTE DE TONO. #BCBF

Se utiliza para definir una de las 15 amplitudes posibles.

LISTADO DESENSAMBLADO

Hisoft GENA3.1 Assembler. Page 1.

Pass 1 errors: 00

```

A000          10          ORG  #A000
              20 ;
              30 ;DEFINICION DE ENVOLVENTE
              40 ;DE VOLUMEN
              50 ;
^A000 3E01     60          LD  A,1
A002 211DA0   70          LD  HL,ENV
A005 CDBCB0   80          CALL #BCBC
A008 213AA0   90          LD  HL,DATOS1
A00B CD2BA0  100         CALL MUSIC
              110 ;
              120 ;DEFINICION DE ENVOLVENTE
              130 ;DE TONO
              140 ;
A00E 3E01     150         LD  A,1
A010 2124A0   160         LD  HL,ENT
A013 CDBFBC   170         CALL #BCBF
A016 2143A0   180         LD  HL,DATOS2
A019 CD2BA0   190         CALL MUSIC
A01C C9       200         RET
              210 ;
              220 ;DATOS DE LAS ENVOLVENTES
              230 ;
A01D 020FFF0A 240 ENV:   DEFB 2,15,-1,10,15,1,10
A024 020ACE0A 250 ENT:   DEFB 2,10,-50,10,10,50,10
              260 ;
              270 ;RUTINA DE MUSICA
              280 ;
A02B 3E01     290 MUSIC: LD  A,1
A02D E5       300         PUSH HL
A02E CDADBC   310         CALL #BCAD
A031 E1       320         POP  HL
A032 E607     330         AND  7
A034 28F5     340         JR   Z,MUSIC
A036 CDAABC   350         CALL #BCAA
A039 C9       360         RET
              370 ;
A03A 010100   380 DATOS1: DEFB 1,1,0
A03D C800     390         DEFW 200
A03F 000F     400         DEFB 0,15
A041 2C01     410         DEFW 300
A043 010001   420 DATOS2: DEFB 1,0,1
A046 F401     430         DEFW 500
A048 000A     440         DEFB 0,10
A04A C800     450         DEFW 200

```

Pass 2 errors: 00

```

DATOS1 A03A  DATOS2 A043  ENT   A024
ENV     A01D  MUSIC   A02B

```

Table used: 71 from 170



PROGRAMA CARGADOR

```
DATA54 CA
10 REM ***PROGRAMA CARGADOR***
20 FOR N=&A000 TO &A04C
30 READ A:SUMA=SUMA+A
40 POKE N,A
50 NEXT
60 IF SUMA<>&1B34 THEN PRINT "ERROR
  EN DATAS"
70 DATA 62,1,33,29,160,205,188
80 DATA 188,33,58,160,205,43,160
90 DATA 62,1,33,36,160,205,191
100 DATA 188,33,67,160,205,43,160
110 DATA 201,2,15,255,10,15,1
120 DATA 10,2,10,206,10,10,50
130 DATA 10,62,1,229,205,173,188
140 DATA 225,230,7,40,245,205,170
150 DATA 188,201,1,1,0,200,0
160 DATA 0,15,44,1,1,0,1
170 DATA 244,1,0,10,200,0,0
```



PROGRAMA BASIC

```
MAQ54 CA
10 ENV 1,15,-1,10,15,1,10
20 SOUND 1,200,300,15,1
30 ENT 1,10,-50,10,10,50,10
40 SOUND 1,500,200,10,,1
```



Código MAQUINA

Condiciones de entrada.

El acumulador debe contener el número de envoltente de tono, y el registro HL debe indicar la dirección de los datos para dicha envoltente.

Condiciones de salida.

Son las siguientes para cada uno de los casos que se indican:

Si la envoltente se ha definido correctamente:

- Carry verdadero
- HL contiene la dirección de los datos+16

Si el número de envoltente ha sido erróneo:

- Carry falso
 - A, BC y HL se preservan
- En cualquier caso:
- DE y los demás flags se corrompen, y los demás registros se preservan.

El contenido y la posición de cada uno de los datos que definen la envoltente, deberá ser la siguiente:

Byte 0: Número de secciones de la envoltente

Bytes 1-3: Primera sección

Bytes 4-6: Segunda sección

Bytes 7-9: Tercera sección

Bytes 10-12: Cuarta sección

Bytes 13-15: Quinta sección

Dentro de cada una de las secciones, se deben indicar los siguientes parámetros:

Byte 0: Número de escalones

Byte 1: Altura de cada escalón

Byte 2: Tiempo de pausa

Puede parecer que a la vista de tal cantidad de parámetros, resulte difícil la generación de sonidos, y la definición de las envoltentes, pero si nos fijamos, el formato para su creación, es muy similar a como se hace desde Basic.

Para que os resulte más fácil la comprensión, hemos preparado un programa en Basic, que se encarga de crear una envoltente de tono y otra de volumen con las que se producen diversos sonidos.

Si comparáis el programa Basic con el programa realizado en código máquina, podréis comprobar que los datos son los mismos, cambiando únicamente la forma de introducirlos.

Con lo visto hasta el momento, queda zanjado el capítulo de la música desde código máquina.



EL CASSETTE Y EL AMSTRAD

Todo ordenador deberá ser capaz de almacenar un programa o fichero para su posterior utilización. El Amstrad CPC no va a ser menos, y permite salvar o almacenar un programa tanto en cinta de cassette como en disco en sus modelos 464-472-664 y 6128.

A

unque cada modelo de Amstrad incorpora de serie un medio distinto para almacenar datos, es posible dotarlo opcionalmente y por cuenta del usuario con el medio complementario. El sistema de almacenamiento en cassette se conoce como **fichero secuencial** mientras que el de disco es del tipo **fichero de acceso aleatorio**. En el fichero secuencial los programas o datos se graban en serie, esto es, uno detrás de otro, siendo necesario para buscar un dato o programa concreto, recorrer la cinta desde el principio hasta alcanzar el deseado. Ocurre lo contrario con los ficheros de acceso aleatorio, en los que llegaremos al programa buscado casi inmediatamente sin tener que recorrer el fichero.

Está claro que todas las ventajas están a favor del fichero de acceso aleatorio (o disco), pero existe una buena razón que otorga a la cinta un encanto muy especial: la **economía**. Mientras que una cinta C-60 no llega a sobrepasar las 200 ptas., un diskette de 3 pulgadas con poco menos de la misma capacidad ronda las 1.100 ptas. Además se puede adquirir el software comercial en soporte cassette haciendo con ello menos elevado el desembolso que debe sufrir el pobre usuario.

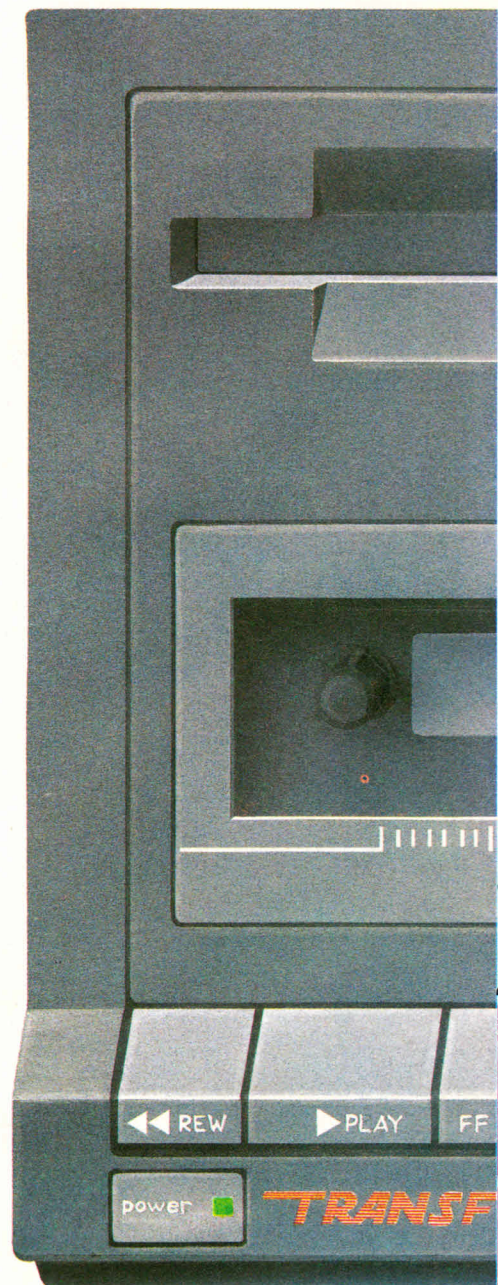
Por otra parte, la unidad de almacenamiento o aparato de cassette es lógicamente más barato que la unidad de disco y, en la mayoría de los casos, los poseedores de los modelos CPC 664 y 6128 (que son los que originalmente no disponen del cassette) cuentan con uno de audio doméstico en su residencia. Es por ello, por lo que nos vemos obligados a comentar con algo más de detenimiento este medio de almacenamiento.

Todo fichero viene precedido de una cabecera

Como todos estaréis hartos de oír, vuestro ordenador funciona con 1 ó 0 (bit). La forma de distinguir el estado de un bit en un medio más dirigido a la función de audio, es precisamente por sonidos, y como el ordenador sólo funciona con dos estados, a cada uno se le hace corresponder una frecuencia determinada de manera que sean claramente distinguibles. Así en el Amstrad, el bit 1 tiene la **mitad** de frecuencia que el bit 0, o lo que es lo mismo, el estado uno tardará el **doble** en cargarse que el cero. Debido a que el Amstrad puede grabar los programas en cinta con distinta velocidad el valor de las frecuencias no se encuentra estandarizado.

Esta es la forma de grabar el **registro de datos**. Sin embargo, el programa no puede entrar por las buenas en la memoria del Amstrad y deberá ir presentado con un «carnet de identidad» que es lo que conocemos por **registro de cabecera** en el que se incluyen un total de 64 bytes (1 byte=8 bits) que dan información al ordenador sobre el nombre del programa, si es Basic o binario, su longitud, dirección en la que se va a ubicar dentro de la memoria, etc.

No hay forma de leer desde Basic todos los datos que nos entrega el registro de cabecera y para ello debemos recurrir al «firmware» del Amstrad con la ayuda del código máquina. La rutina del firmware a la que nos referimos es la &BCA1 denominada «CAS READ» cuya misión es leer un registro de la cinta. Las condiciones de entrada de esta rutina vienen dadas por los registros HL (dirección en la que se van a almacenar los datos), DE (longitud de los datos) y A que es el **código de sincronismo** del registro; en el caso que nos ocupa —el registro de cabecera— A vale &2C, pero si lo que queremos leer es el registro de datos de un fichero previamente grabado en bloques de 2K, A debe valer &16. Existen otros valores de A aunque estos se suelen utilizar



únicamente en los llamados «programas sin cabecera» para las protecciones del software comercial.

Una vez que hallamos depositado en una determinada dirección HL tendremos los siguientes valores de los 64 bytes (= DE) del registro de cabecera:

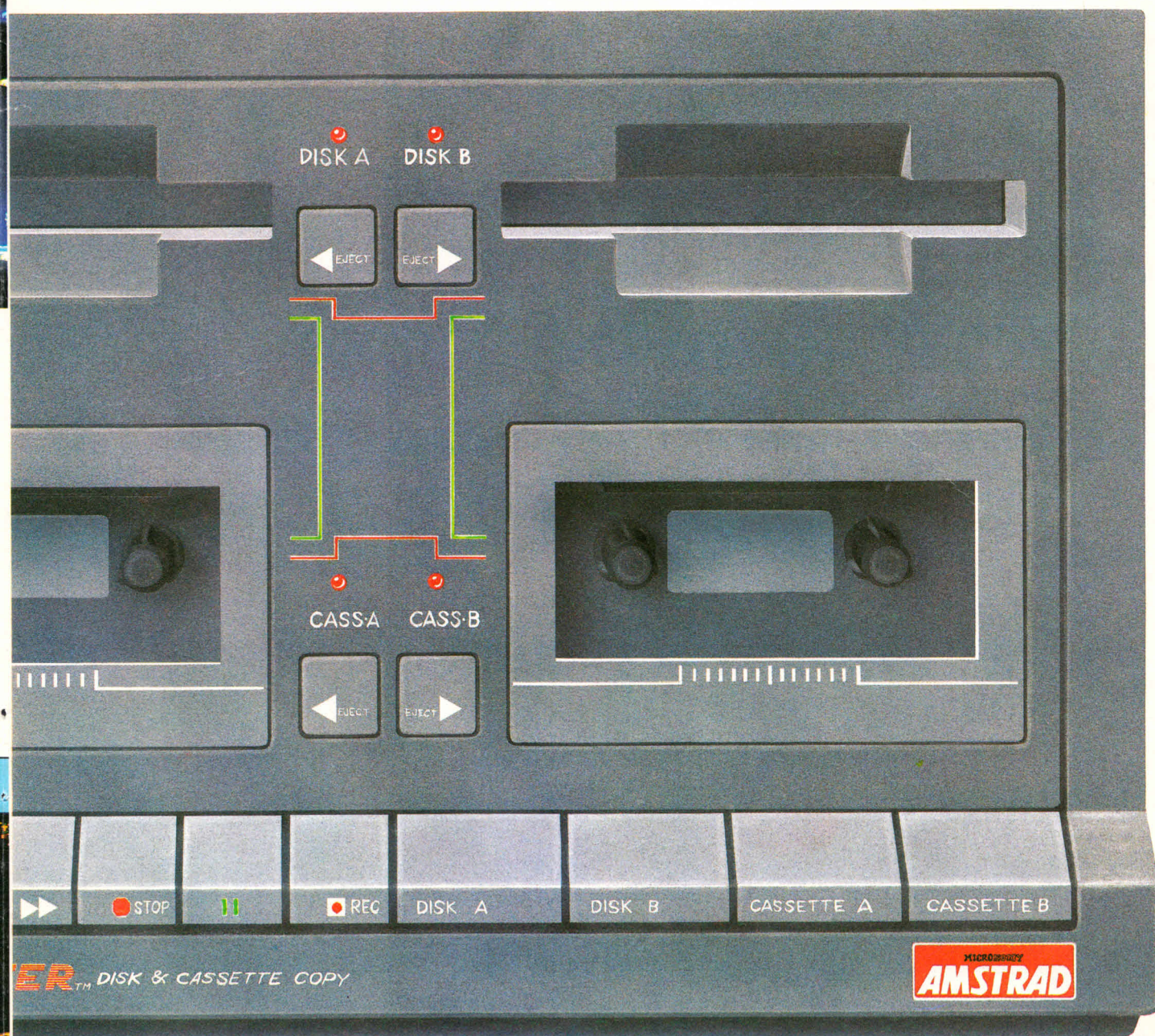
Utilidades del PROGRAMADOR

Bytes 0-15 .. Nombre del fichero (cada byte es el código ASCII de cada carácter alfanumérico).
 Byte 16 Número de bloque.
 Byte 17 Distinto de cero (generalmente 255=&FF) si se trata del último bloque.
 Byte 18 Tipo de fichero:
 Bit 0: 0 si está desprotegido;
 1 si está protegido.

Bits 1-3: &X000 BASIC
 &X001 Binario
 &X010 Copia de pantalla
 &X011 ASCII
 &X100 No especificado
 &X101 No especificado

Como aplicación al contenido del registro de cabecera os proponemos como aplicación el programa «Listador» que os podrá ser muy útil para conocer el catálogo de una cinta de cassette conociendo además del nombre de cada programa grabado, los datos más importantes que contiene su cabecera. Os aconsejamos que lo tecleéis y lo salvéis posteriormente en cassette o disco (en este caso añadir la línea 5 ITAPE).

El programa utiliza una pequeña subrutina en código máquina basada en la instrucción



```

10 MEMORY 7999:FOR n=&1F40 TO &1F4B
:READ a:POKE n,a:NEXT
20 MODE 1:LINE INPUT "Nombre de la
cinta: ";n#:n#=n#+SPACE$(63):n#=LEF
T$(n#,63)
30 LINE INPUT "Cara: ";c#:c#=c#+ " "
:c#=LEFT$(c#,1)
40 PRINT "Tienes impresora? (s/n)"
50 i#=UPPER$(INKEY$):IF i#="N" OR i
#="S" THEN 60 ELSE 50
60 MODE 2:GOSUB 80:IF i#="S" THEN W
IDTH 80:GOSUB 130
70 GOTO 180
80 PRINT CHR$(150);STRING$(78,CHR$(
154));CHR$(156);
90 PRINT CHR$(149);" CINTA: ";n#;TA
B(73);"CARA: ";c#;TAB(80);CHR$(149)
;
100 PRINT CHR$(151);STRING$(78,CHR$(
154));CHR$(157);
110 PRINT CHR$(149);" N. TIPO PROT
. NOMBRE          BLQ. LONG.BLOQ. L
ONG.TOT. DIR.INIC. DIR.EJEC. ";CHR$(
149);
120 PRINT CHR$(147);STRING$(78,CHR$(
154));CHR$(153);:RETURN
130 PRINT#8,STRING$(80,"*")
140 PRINT#8,"* CINTA: ";n#;TAB(73);
"CARA: ";c#;TAB(80);"*"
150 PRINT#8,STRING$(80,"*")
160 PRINT#8;"* N. TIPO PROT. NOMBR
E          BLQ. LONG.BLOQ. LONG.TOT
. DIR.INIC. DIR.EJEC. *"
170 PRINT#8,STRING$(80,"*")
180 WINDOW#1,1,80,6,25:c=1
190 CALL 8000:t=PEEK(9018)
200 f#=BIN$(t,8):p=VAL(RIGHT$(f#,1)
):IF p=0 THEN p#="NO":GOTO 220
210 p#="SI"
220 tipo=VAL("&X"+MID$(f#,5,3))
230 IF tipo=0 THEN t#="BAS."
240 IF tipo=1 THEN t#="BIN."
250 IF tipo=2 THEN t#="PANT."
260 IF tipo=3 THEN t#="ASCII"
270 IF tipo>3 THEN t#=" -"
280 n=PEEK(9016)
290 lb=PEEK(9019)+256*PEEK(9020)
300 lt=PEEK(9024)+256*PEEK(9025)
310 dir=PEEK(9021)+256*PEEK(9022)
320 de=PEEK(9026)+256*PEEK(9027)
330 IF de=0 THEN d#=" -":GOTO 350
340 d#=STR$(de)
350 IF PEEK(9000)=0 THEN n#="SIN NO
MBRE":GOTO 410
360 n#=""
370 FOR k=9000 TO 9015
380 IF PEEK(k)=0 THEN 400
390 n#=n#+CHR$(PEEK(k))
400 NEXT
410 canal=1:GOSUB 430:IF i#="S" THE
N canal=8:GOSUB 430
420 c=c+1:GOTO 190
430 PRINT#canal,TAB(2);c;TAB(6);t#;
TAB(13);p#;TAB(18);n#;TAB(34);n;TAB
(39);lb;TAB(50);lt;TAB(60);dir;TAB(
70);d#;
440 RETURN
1000 DATA 062,044,017,064,000,033
1010 DATA 040,035,205,161,188,201
2000 SAVE "LISTADOR":SAVE "!LISTADO
R"

```

&BCA1 del firmware que ya hemos mencio-
nado. En este caso depositamos los 64 bytes
de la cabecera a partir de la dirección 9000
(=&2328).

El firmware del cassette

En ocasiones, el Basic no puede hacer na-
da ante situaciones como cargar o salvar un
programa en código máquina a partir de una
dirección más baja de la 370 por ejemplo. El
firmware del **Amstrad** puede resultarnos en
estos casos de gran ayuda, y por ello nos ve-
mos obligados a exponer las instrucciones que
se refieren a nuestro tema.

Algunos de los comandos del firmware que
vamos a exponer cumplen el mismo cometido
para ficheros aleatorios en disco, por lo que,
en el caso de disponer de un CPC con dicha
unidad debemos preceder nuestro programa
con ITAPE, ITAPE.IN, etc.

&BC65 CAS INITIALISE

Inicializa el uso del cassette fijando una ve-
locidad de escritura lenta (SPEED WRITE 0) y

activando los mensajes de ayuda. Siempre que
se utilice esta orden desactivamos el disco
(equivalente a ITAPE).

&BC68 CAS SET SPEED

Define una velocidad de grabación en el
cassette. Los registros que deben preceder esta
orden son A (precompensación) y HL (perío-
do de escritura). La precompensación debe va-
ler entre 0 y 225 (microsegundos). HL define
la duración de medio periodo del bit 0 en mi-
crossegundos. Para conocer la velocidad de
escritura en baudios se puede utilizar la si-
guiente fórmula: $1016 (3*HL)$ con HL entre
130 y 480. Así para 2.000 baudios (SPEED
WRITE 1) tenemos HL=167 y A=50; para
1.000 baudios (SPEED WRITE 0) HL=333 y
A=50.

&BC6B CAS NOISY

Controla los mensajes de ayuda del casset-
te permitiendo que aparezca en pantalla el
clásico mensaje de «Press PLAY then any key»
colocando en el registro A el valor 0 y otro
valor distinto en caso contrario. El uso de es-
te comando inhabilita el disco.

&BC6E CAS START MOTOR

Espera a que se ponga en marcha el cas-
sette y se establezca su velocidad. Sin condicio-
nes de entrada. Inhabilita el disco.

&BC71 CAS STOP MOTOR

Detiene el motor del cassette. Sin condicio-
nes de entrada.

&BC71 CAS RESTORE MOTOR

Restaura el estado previo del motor. El re-
gistro A contiene el estado previo del motor.

&BC77 CAS IN OPEN

Es el equivalente a la orden OPENIN del
Basic del **Amstrad** siendo su función la de
abrir un fichero de entrada del cassette (o dis-
co). Dicho comando debe ser precedido por
los registros: B que contiene la longitud del
nombre del fichero (máximo 16 caracteres),
HL la dirección en la que se encuentran los ca-
racteres del nombre, DE la dirección del área
de 2k que se va a utilizar como **buffer** (pue-
de valer incluso la zona de memoria de pan-
talla).

&BC7A CAS IN CLOSE

El equivalente del CLOSEIN en Basic. Se li-
mita a cerrar el fichero de lectura o entrada

de cassette (o disco). No va precedido de ningún registro.

&BC7D CAS IN ABANDON

Abandona la lectura de un fichero de cassette (o disco). No tiene condiciones de entrada.

&BC80 CAS IN CHAR

Lee un carácter de un fichero de cassette (o disco) y lo devuelve en el registro A. Sin condiciones de entrada.

&BC83 CAS IN DIRECT

Lee el registro de datos de un fichero de cassette (o disco) a partir de una dirección inicial que se introduce previamente en el registro HL. Esta orden debe ir intercalada entre «CAS IN OPEN» y «CAS IN CLOSE» descritas anteriormente.

&BC86 CAS RETURN

Devuelve el último carácter leído al fichero de entrada de cassette (o disco).

&BC89 CAS TEST EOF

Como su denominación inglesa aclara es equivalente a la orden EOF (End Of File) del Basic, comprobando si se ha alcanzado el final del fichero. En caso afirmativo el acarreo queda en alto mientras que en caso contrario (errores de lectura o si se ha pulsado la tecla de Escape) quedará en bajo.

&BC8C CAS OUT OPEN

El equivalente a OPENOUT del Basic. Abre un fichero de salida (se dispone a grabar) de cassette (o disco). Debe ir precedido por los registros: B con la longitud del nombre del fichero (máximo 16 caracteres en cassette), HL con la dirección en la que está localizado dicho nombre, y DE la dirección de la zona de memoria de 2k que se va a utilizar como «buffer».

&BC8F CAS OUT CLOSE

Equivalente a la orden CLOSEOUT del Basic. Da por concluida la grabación del fichero en cassette (o disco). Sin condiciones de entrada.

&BC92 CAS OUT ABANDON

Abandona la grabación de un fichero en cassette (o disco). Sin condiciones de entrada.

&BC95 CAS OUT CHAR

Escribe un carácter en el fichero que se va a grabar. El código ASCII de dicho carácter debe ir en el registro A.

&BC98 CAS OUT DIRECT

Graba el registro de datos de un fichero en cassette o disco a partir de la dirección HL, con una longitud DE, una dirección de ejecución BC (en caso de tenerla).

El tipo de fichero viene dado por A: 0 programa en Basic, 1 programa en Basic protegido y 2 programa en código máquina.



Esta orden debe ir intercalada entre «CAS OUT OPEN» y «CAS OUT CLOSE» explicadas anteriormente.

&BC9B CAS CATALOG

Realiza la misma función que la orden CAT del Basic, esto es, lista los programas de una cinta. Obliga a introducir la dirección de una zona de memoria de 2k en el registro DE que actuará como buffer.

&BC9E CAS WRITE

Graba un registro en cinta. Esta orden **no vale** para disco. El registro HL contiene la dirección de los datos que se van a grabar, DE la longitud de los datos, y A el tipo de registro.

Si lo que se pretende grabar es el registro de cabecera A debe valer &2C, si por el contrario es el registro de datos A será &16. Otros valores de A se utilizan para grabar registros de datos que no están en formato de bloques de 2k y es muy utilizado para protección de programas comerciales.

&BCA1 CAS READ

Equivalente al comando anteriormente descrito, pero utilizado para lectura de registros. **No vale** para ficheros en disco. Los registros HL, DE y A se deben asignar antes del comando con el mismo significado citado en «CAS WRITE».

&BCA4 CAS CHECK

Muy útil para verificar programas comparando el fichero que se han de leer con el contenido en memoria. HL debe contener la dirección inicial de los datos que se van a comparar, DE, la longitud de los datos, y A, el tipo de registro especificado en «CAS WRITE». Esta orden **no vale** para ficheros en disco.

En el caso que la verificación del programa haya sido correcta coloca el acarreo en alto. Si por el contrario la verificación ha sido incorrecta el acarreo queda en bajo y A contiene

un código de error: 0 si se ha pulsado la tecla Escape, 1 error de lectura, 2 comprobación CRC incorrecta y 3 verificación fallida.

Ajuste del cassette del Amstrad CPC 464/472

El modelo CPC 464 equipa de origen un cassette integrado en el ordenador. Como el resto de los cassettes que existen en el mercado debe revisarse periódicamente, y en el caso de arrastrar mal las cintas, sustituir inmediatamente la correa de arrastre por otra del mismo diámetro y sección que podemos adquirir a bajo precio en tiendas de electrónica especializada o solicitándolo a Indescomp.

Para el ajuste del azimut de la cabeza lectora debemos disponer de un voltímetro digital que mida en valores medios y un destornillador plano y estrecho para introducir por la pequeña abertura que hay junto a la tapa del cassette y regular el alineamiento de la cabeza.

Con la ayuda de un generador de frecuencias efectuaremos una grabación en otro aparato de cassette de un tono constante de 6 KHz. Insertaremos esa cinta de prueba en el cassette del **Amstrad** ajustando su volumen al máximo. Conectaremos el voltímetro entre masa y el terminal 7 del integrado IC 302 que se encuentra sobre la placa del cassette. Ajustando el azimut debemos obtener una lectura óptima entre 330 mV y 520 mV.

Para comprobar la velocidad del cassette debemos necesitar un osciloscopio y comprobar en el rango de 100 microsegundos que el cassette lee correctamente los 6 KHz. Una variación de velocidad del orden del 4% no afectará la carga del software.

Excepto la sustitución de la correa de tracción las operaciones de ajuste deben llevarse a cabo por verdaderos entendidos. Así que, chapuzas abstenerse.

FLASH FIRE

Como consecuencia de la terrible explosión ocurrida en las minas de Uranio del tercer planeta del quinto sistema, cientos de trabajadores huyeron al espacio.

TABLA DE RUTINAS

- Portada
- Pantalla 2
- Instrucciones
- Gráficos en tres colores
- Jugador
- Pantallas del juego (1, 2 y 3)
- Datos
- Comienzo
- Fin
- Puntos y récord
- Disparo
- Anula vidas
- Subpuertas

RSX

- Imput,3=Coordenada RSX pinta androides.
- Imput,4=Pinta paredes
- Imput,2=Pinta obstáculos
- Imput,0=Pinta nave.
- Imput,1=Pinta nave.

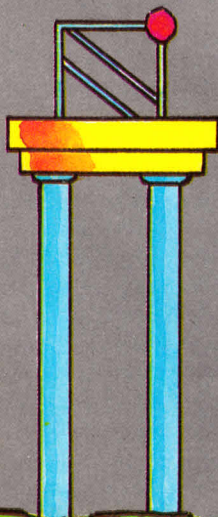
NOTA: Para poder usar joystick tienes que seguir las siguientes indicaciones

- Capslock+Control+K=↑
- Capslock+Control+J=↓
- Capslock+Control+H=←
- Capslock+Control+I=→

Cuando el combustible se les agotó hubieron de saltar de la nave. Ahora flotan dispersos en el espacio sideral y tu misión consiste en recogerlos con tu nave GALAX-25, pero ¡cuidado!, el cinturón de asteroides en el que se encuentran los mineros no te lo va a poner fácil.

El programa tiene algunas llamadas de memoria que son las siguientes: a) CALL & BC02, que restablece los colores del juego; b) CALL & BB16, que hace la función de un INKEY.

El movimiento es un modo XOR, por lo que la nave siempre permanecerá en movimiento.



VARI

la=parte correspondiente al primer input.
 Da opción a las instrucciones o al juego.
 cde\$=texto del scroll de instrucciones.
 lo=longitud de línea
 k\$=para el scroll
 dire=dirección de memoria RSX
 soi=lector de datos
 h=himen
 li=vidas
 el\$=input el nombre del jugador
 t=parte correspondiente al PLOT
 pe=parte correspondiente del PLOT encargado del eje X.
 i=parte correspondiente del PLOT encargado del eje Y.
 na=dirección de memoria principal correspondiente al eje X.
 nb=dirección de memoria principal correspondiente al eje Y.
 poi=coordenada de X de la puerta 1.
 iop=coordenada de X de la puerta 1.
 lkj=coordenada Y de la puerta 1.

ESTRU

Núm. de línea	Característica
20	Portada
110	Menú de opciones
150	Instrucciones
460	Gráficos en 3 colores
1020	Jugador
1050	Pantallas (1, 2 y 3)
1460	Datos
1480	Comienzo del juego
1490	Teclas

ABLES

qww=coordenada X de la puerta 2.
 asd=coordenada X de la puerta 2.
 zxc=coordenada Y de la puerta 2.
 poi=coordenada X de la puerta 3.
 iop=coordenada X de la puerta 3.
 lkj=coordenada Y de la puerta 3.
 sc=puntos.
 lk=coordenada X de la puerta 4.
 uj=coordenada X de la puerta 4.
 yh=coordenada Y de la puerta 4.
 pl=coordenada X de la puerta 5.
 qaz=coordenada X de la puerta 5.
 wsx=coordenada Y de la puerta 5.
 o\$=inkey para teclas del movimiento.
 a=coordenada del eje X de la nave.
 b=coordenada del eje Y de la nave.
 po=dirección del movimiento de la nave.
 re=color para utilizar el test.
 lg=número multiplicando de la puntuación.
 hi=récord.
 j=longitud del disparo.

CTURA

1570	Test
1580 a	
1620	Mueve nave
1630 a	
1660	Límites de pantalla
1670 a	
1740	Límites de puertas
1770	Pinta nave
1840	Fin del juego
1910	Puntos y récord
2030	Disparo
2150	Anula vidas
2190	Subpuertas

CA
 1 *****
 2 *** POR ***
 3 *** OSCAR ***
 4 *** VALLADARES ***
 5 *** MARTINEZ ***
 6 *** 1986 ***
 7 *****
 10 CALL &BC02
 20 *****PORTADA*****

```

30 PEN 1:PAPER 1:INK 0,0:BORDER 0:MO
DE 0
40 FOR f=1 TO 50
50 WINDOW RND(1)*19+1,RND(1)*19+1,RN
D(1)*24+1,RND(1)*24+1
60 PAPER INT(15*RND(1)+1):SOUND 1,50
0,3,10,,1:CLS:NEXT
70 CLG 6:WINDOW #7,17,4,7,19:PAPER #
7,0:CLS #7:LOCATE #7, 6,4:PEN #7,3:P
RINT #7," FLASH FIRE":PEN #7,2:LOCA
TE #7,3,7:PRINT#7,"Por O.V.M."
80 PLOT 1,1:DRAW 640,1,3:PLOT 1,1:DR
AW 1,398:PLOT 1,398:DRAW 640,398:PLO
T 639,1:DRAW 639,398
90 FOR f=1 TO 3000:NEXT:CLG 1:FOR f
=1 TO 200:NEXT
100 GOTO 1470
110 '*****PANTALLA 2*****
120 CLS:MODE 0:PAPER 0:PEN 2:INK 0,0
:BORDER 0:PLOT 1,1:DRAW 640,1,3:PLOT
1,1:DRAW 1,398:PLOT 1,398:DRAW 640,
398:PLOT 639,1:DRAW 639,398
130 LOCATE 5,6:PRINT"1=INSTRUCCIONES
":LOCATE 5,8:PEN 1:PRINT"2=JUEGO":LO
CATE 2,15:PEN 3:PRINT"Pulsar 1 o 2 p
ara":LOCATE 6,17:PEN 2:PRINT"SEGUIR"
:LOCATE 9,19:PEN 6:INPUT;la
140 ON la GOTO 150,470
150 '*****INSTRUCCIONES*****
*****
160 MODE 1:INK 0,0:PEN 2
170 PRINT"
"
180 PRINT" MMM M   MMM MMM M M   MMM
MMM MMM MMM"
190 PRINT" M   M   M M M   M.M   M
M   M M M "
200 PRINT" MM  M   MMM MMM MMM  MM
M   MM MM "
210 PRINT" M   M   M M   M M'M   M
M   M M M "
220 PRINT" M   MMM M M MMM M M   M
MMM M M MMM"
230 PLOT 1,1:DRAW 640,1:PLOT 1,1:DR
AW 1,398:PLOT 1,398:DRAW 640,398:PLOT
639,1:DRAW 639,398:PLOT 1,300:DRAW
640,300:PLOT 1,260:DRAW 640,260:PLOT
1,200:DRAW 640,200:PLOT 1,170:DRAW
640,170:PLOT 1,40:DRAW 640,40
240 LOCATE 15,8:PEN 1:PRINT"INSTRUCC

```

```

IONES":LOCATE 17,14:PRINT"TECLAS":LO
CATE 4,24:PEN 2:PRINT"(SPACE) para S
EGUIR"
250 WINDOW #7,35,27,17,22:PEN #7,3
260 PRINT #7,"F= ABORT"
270 PRINT #7,"A= UP  "
280 PRINT #7,"Z= DOWN "
290 PRINT #7,"N= LEFT "
300 PRINT #7,"M= RIGHT"
310 PRINT #7,"X= FIRE "
320 LOCATE 3,20:PRINT"PUEDES USAR JO
YSTICK"
330 lo=20*(1+1)-20*(1=2)
340 cde$="          TU MISION CONSISTE
EN RECOGER TODOS LOS ANDROIDES QUE H
AN SIDO DISPERSADOS POR LOS TRES SIS
TEMAS ESTELARES.          TEN CUIDADO P
OR QUE POR CADA ANDROIDE QUE RECOJAS
SE ELEVA LA DIFICULTAD DEL JUEGO.
          *SUERTE*          "
350 IF LEN(cde$)<lo THEN cde$=cde$+S
PACE*(lo-LEN(cde$))
360 loi=LEN(cde$)
370 y=11
380 LOCATE 1,y:PRINT LEFT$(ty$,lo);
390 WHILE INKEY(47)<0
400 cde$=MID$(cde$,2)+LEFT$(cde$,1)
410 LOCATE 1,y
420 PRINT LEFT$(cde$,lo);
430 WEND
440 WHILE INKEY$("<") " ";k$=INKEY$:WEND
450 CALL &BB16:GOTO 120
460 '*****GRAFICOS EN 3 COLORES*
*****
470 MODE 0:CLS:LOCATE 5,12:PRINT"PLE
ASE WAIT": PLOT 1,1:DRAW 640,1,14:PL
OT 1,1:DRAW 1,398,15:PLOT 1,398:DRAW
640,398,14:PLOT 639,1:DRAW 639,398,
15
480 h=HIMEM
490 RESTORE 550
500 dire=h:FOR soi=1 TO 82
510 READ b
520 POKE dire,b
530 dire=dire+1
540 NEXT
550 DATA 0,0,136,152,152,158,158,156
,136,136,17,145,145,151,151,147,17,1

```

```

7,255,3,0,112,112,0,3,255,255,8,8,19
2,192,8,8,255,176,113,243,231,231,24
3,113,176,208,232,252,126,126,252,23
2,208
560 DATA 7,51,87,187,171,48,34,2,14,
204,174,221,93,192,68,4,240,247,199,
215,215,199,247,240,240,254,62,190,1
90,62,254,240
570 RESTORE 780
580 h=HINEM
590 POKE 0,0:h=h-192:dirac=h:MEMORY
(h-1)
600 FOR t=1 TO 192:READ a$
610 p=VAL("&"a$):POKE dirac,p
620 dirac=dirac+1
630 NEXT
640 r=h+9:a=INT(r/256):b=r-(a*256)
650 POKE h+1,b:POKE h+2,a
660 r=h+17:a=INT(r/256):b=r-(a*256)
670 POKE h+4,b:POKE h+5,a
680 r=h+21:a=INT(r/256):b=r-(a*256)
690 POKE h+9,b:POKE h+10,a
700 r=h+30:a=INT(r/256):b=r-(a*256)
710 POKE h+41,b:POKE h+42,a
720 POKE h+63,b:POKE h+64,a
730 r=h+31:a=INT(r/256):b=r-(a*256)
740 POKE h+122,b:POKE h+123,a
750 r=h+194:a=INT(r/256):b=r-(a*256)
760 POKE h+31,b:POKE h+32,a
770 CALL h
780 DATA 01,29,4e,21,31,4e,c3,d1
790 DATA bc,35,4e,10,18,00,18,11
800 DATA 00,00,00,00,00,4d,50,55
810 DATA d4,47,50,55,d4,00,00,30
820 DATA 75,f5,af,18,03,f5,3e,01
830 DATA 32,3e,4e,f1,a7,c8,47,f5

```

```

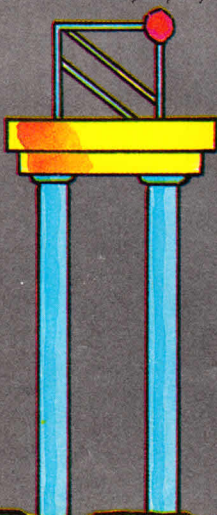
840 DATA dd,23,dd,23,10,fa,cd,11
850 DATA bc,47,3e,02,90,f5,3a,3e
860 DATA 4e,a7,20,0a,cd,78,bb,2d
870 DATA 25,cd,1a,bc,18,1b,cd,c6
880 DATA bb,b7,cb,1c,cb,1d,cd,11
890 DATA bc,f1,f5,a7,20,08,47,b7
900 DATA cb,1a,cb,1b,10,f9,cd,1d
910 DATA bc,f1,c1,c5,f5,dd,2b,dd
920 DATA 2b,4f,06,03,80,47,79,e5
930 DATA ed,5b,3f,4e,dd,6e,00,26
940 DATA 00,29,10,fd,19,eb,e1,87
950 DATA 20,01,3c,47,c5,e5,06,00
960 DATA 3a,05,00,a7,20,09,fe,01
970 DATA 20,0b,fe,02,1a,18,00,1a
980 DATA a7,20,05,10,02,1a,ae,77
990 DATA cd,26,bc,13,10,e2,e1,cd
1000 DATA 20,bc,c1,10,d7,f1,c1,10
1010 DATA b2,c9,00,00,00,00,00,00
1020 '*****JUGADOR*****
1030 CLS:MODE 0:PLOT 1,1:DRAW 640,1,
9:PLOT 1,1:DRAW 1,398:PLOT 1,398:DR
AW 640,398:PLOT 639,1:DRAW 639,398:LO
CATE 5,3:PEN 3:PRINT**JUGADOR**:LO
CATE 2,12:PEN 14:PRINT"OMBRE=.....
":LOCATE 3,17:PEN 2:PRINT" (SPACE)
para SEGUIR"
1040 PLOT 1,1:DRAW 640,1,3:PLOT 1,1:
DRAW 1,398:PLOT 1,398:DRAW 640,398:P
LOT 639,1:DRAW 639,398:LOCATE 2,10:P
EN 8:PRINT"( SOLO 5 LETRAS )":LOCATE
8,12:PEN 6:INPUT;e1$:CALL &BB16:GOT
O 1000
1050 '*****PANTALLAS*****
1060 '*****PANTALLA 1*****
1070 GOTO 1090
1080 MODE 1:i=5
1090 CLS:MODE 1:t=2:FOR f=1 TO 60:pe
=INT(RND*640+1):h=INT(RND*26+1):INK
f,11:INK 2,24:INK 3,6:i=400-INT(RND*
352+1):PLOT pe,i,t:NEXT
1100 na=15007:nb=15008:qwe=0:asd=0:z
xc=0:ik=0:uj=0:yh=0
1110 POKE na,37:POKE nb,18
1120 FOR f=1 TO 8:ce=INT(RND*37)+3:d
e=INT(RND*19)+4:LOCATE ce,de:PEN 1:i
MPUT,2,2,2:PRINT:NEXT:FOR f=i TO 16:

```

```

ce=INT(RND*37)+3:de=INT(RND*19)+4:LO
CATE ce,de:PEN 3:IMPUT,3:PRINT:NEXT
1130 PEN 2:LOCATE 7,15:IMPUT,3:PRINT
:PEN 9:FOR d=10 TO 30:LOCATE d,4:IMP
UT,4:PRINT:NEXT d:PEN 9:FOR d=10 TO
30:LOCATE d,8:IMPUT,4:PRINT:NEXT d:F
OR d=4 TO 8:LOCATE 10,d:IMPUT,4:PRIN
T:NEXT d:PEN 2:FOR d=12 TO 28:LOCATE
d,6:IMPUT,3:PRINT:NEXT d
1140 PEN 9:FOR d=4 TO 8:LOCATE 30,d:
IMPUT,4:PRINT:NEXT d
1150 LOCATE 3,20:IMPUT,4:PRINT:LOCAT
E 4,20:PEN 2:PRINT CHR$(143)+CHR$(14
3):LOCATE 6,20:IMPUT,4:PRINT:poi=4:i
op=5:1kj=20
1160 LOCATE 38,21:PEN 2:PRINT CHR$(1
43):LOCATE 37,21:PRINT CHR$(143):LOC
ATE 39,21:IMPUT,4:PRINT:LOCATE 36,21
:IMPUT,4:PRINT
1170 WINDOW #1, 40,1,1,22:WINDOW #3,
40,1,23,25:PLOT 0,46:DRAW 640,46:PEN
#3,3:PAPER#3,0
1180 BORDER 10: LOCATE #3,3,3:PRINT
#3,"SCORE: ";(sc):LOCATE #3,3,2:PRIN
T #3,"HI SCORE: ";(hi):PEN 3:LOCATE
#3,27,2:PRINT #3,"PLAYER:":LOCATE #3
,27,3:PRINT#3,"LIVES:":LOCATE #3,35,
3:PEN.#3,3:PRINT #3,(li)
1190 LOCATE #3,35,2:PEN 3:PRINT #3,
e1$:GOTO 1400
1200 '*****PANTALLA 2*****
*****
1210 MODE 1:CLS:GOTO 1230
1220 MODE 1:CLS:POKE na,4:POKE nb,19
:GOTO 1240
1230 POKE na,33:POKE nb,3
1240 FOR f=1 TO 8:ce=INT(RND*37)+3:d
e=INT(RND*19)+4:LOCATE ce,de:PEN 1:i
MPUT,2,2,2:PRINT:NEXT:FOR f=1 TO 16:
ce=INT(RND*37)+3:de=INT(RND*19)+4:LO
CATE ce,de:PEN 3:IMPUT,3:PRINT:NEXT
1250 PEN 9:FOR d=3 TO 30:LOCATE d,8:
IMPUT,4:PRINT:NEXT d:PEN 9:FOR d=3 T
O 30:LOCATE d,12:IMPUT,4:PRINT:NEXT:
FOR d=8 TO 12:LOCATE 6,d:IMPUT,4:PRI
NT:NEXT:FOR d=8 TO 12:LOCATE 12,d:IM

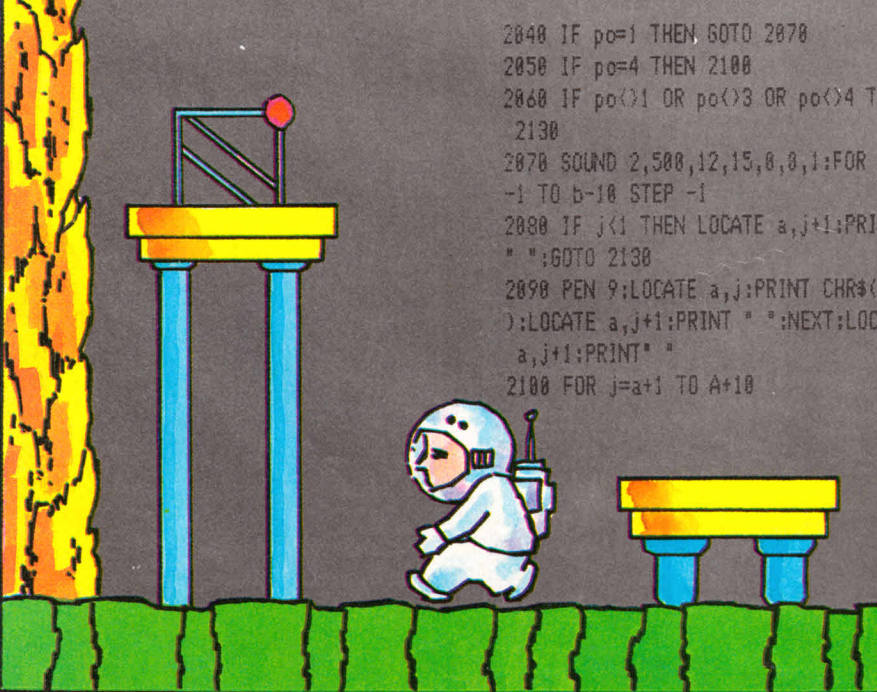
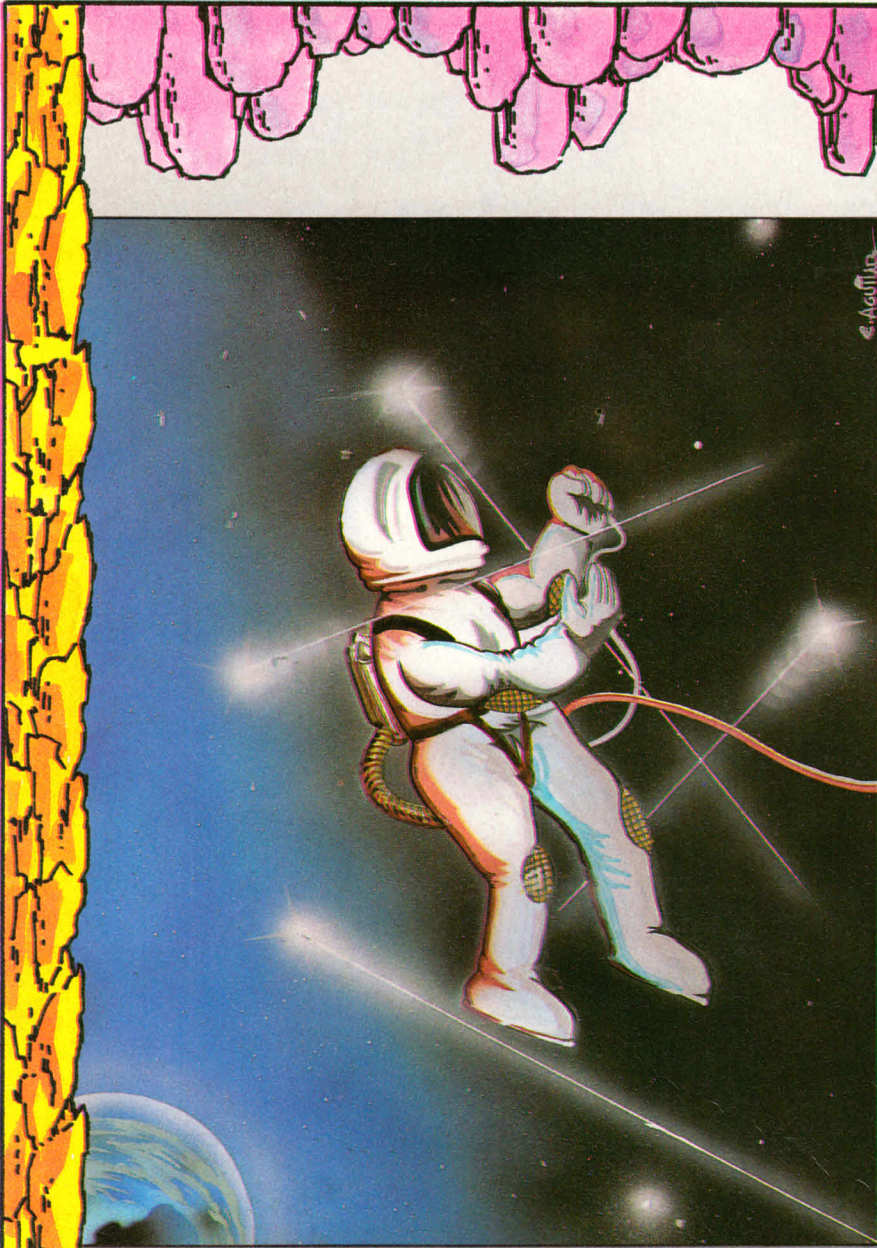
```



```

PUT,4:PRINT:NEXT:FOR d=8 TO 10:LOCATE
E d,10:INPUT,3:PRINT:NEXT
1260 PEN 3:FOR d=31 TO 40:LOCATE d,1
8:INPUT,3:PRINT:NEXT:PEN 9:FOR d=31
TO 40:LOCATE d,16:INPUT,4:PRINT:NEXT
:FOR d=31 TO 40:LOCATE d,20:INPUT,4:
PRINT:NEXT:FOR d=16 TO 20:LOCATE 31,
d:INPUT,4:PRINT:NEXT
1270 qwe=32:asd=33:zxc=2:LOCATE 31,2
:INPUT,4:PRINT:LOCATE 32,2:PEN 2:PRI
NT CHR$(143)+CHR$(143):PEN 9:LOCATE
34,2:INPUT,4:PRINT:FOR d=8 TO 11:LOC
ATE 30,d:INPUT,4:PRINT:NEXT:FOR f=13
TO 29:LOCATE f,10:INPUT,3:PRINT:NEX
T
1280 FOR n=1 TO 80
1290 pe=INT(RND*640+1)
1300 h=INT(RND*26+1)
1310 INK 1,11:INK 2,24:INK 3,6
1320 i=400-INT(RND*352+1)
1330 PLOT pe,i,t
1340 NEXT
1350 LOCATE 3,20:INPUT,4:PRINT:LOCAT
E 4,20:PEN 2:PRINT CHR$(143)+CHR$(14
3):LOCATE 6,20:INPUT,4:PRINT:poi=4:i
op=5:ikj=20
1360 sc=sc:GOTO 1170
1370 '*****PANTALLA 3*****
*****
1380 CLS:MODE 1:POKE na,4:POKE nb,3:
poi=0:iop=0:lkj=0:qwe=0:asd=0:zxc=0:
ik=4:uj=5:yh=2
1390 FOR n=1 TO 80:pe=INT(RND*640+1)
:h=INT(RND*26+1):INK 1,11:INK 2,24:I
NK 3,6:i=400-INT(RND*352+1):PLOT pe,
i,t:NEXT
1400 FOR f=1 TO 8:ce=INT(RND*37)+3:d
e=INT(RND*19)+4:LOCATE ce,de:PEN 1:
INPUT,2,2,2:PRINT:NEXT:FOR f=1 TO 16:
ce=INT(RND*37)+3:de=INT(RND*19)+4:LO
CATE ce,de:PEN 3:INPUT,3:PRINT:NEXT
1410 FOR f=20 TO 40:LOCATE f,20:IMPU
T,4:PRINT:NEXT:FOR f=20 TO 40:LOCATE
f,16:INPUT,4:PRINT:NEXT:FOR f=20 TO
40:LOCATE f,18:INPUT,3:PRINT:NEXT:F
OR f=16 TO 21:LOCATE 19,f:INPUT,4:PR
INT:NEXT:FOR f=1 TO 4:LOCATE f,16:IM
PUT,4:PRINT:NEXT
1420 FOR f=20 TO 40:LOCATE f,21:IMPU
T,4:PRINT:NEXT:FOR f=10 TO 39:LOCATE
f,6:PEN 9:INPUT,4:PRINT:NEXT:FOR f=
10 TO 39:LOCATE f,10:INPUT,4:PRINT:N
EXT:FOR f=6 TO 10:LOCATE 10,f:INPUT,
4:PRINT:LOCATE 39,f:INPUT,4:PRINT:NE
XT
1430 FOR f=12 TO 37:LOCATE f,8:PEN 3
:INPUT,3:NEXT
1440 LOCATE 3,2:INPUT,4:PRINT:LOCATE
4,2:PEN 2:PRINT CHR$(143)+CHR$(143)
:LOCATE 6,2:INPUT,4:PRINT
1450 sc=sc:GOTO 1170
1460 '*****DATOS*****
***
1470 li=5:lg=10:pl=37:qaz=38:wsx=21:
sc=0:na=15007:nb=15008:GOTO 120
1480 '*****EMPIECE DEL JUEGO
*****
1490 o$=INKEY$
1500 IF o$="f" OR o$="F" THEN GOTO 1
10
1510 IF o$="a" OR o$=CHR$(11) OR o$=
"A" THEN po=1
1520 IF o$="z" OR o$=CHR$(10) OR o$=
"Z" THEN po=2
1530 IF o$="n" OR o$=CHR$(8) OR o$="
N" THEN po=3
1540 IF o$="m" OR o$=CHR$(9) OR o$="
M" THEN po=4
1550 IF o$="x" OR o$="X" OR o$="X" T
HEN GOTO 2040
1560 a=PEEK(na):b=PEEK(nb)
1570 DEF FN point(a,b)=TEST((a*16)-2
,(406-b*16))
1580 IF po=0 THEN po=4
1590 IF po=1 THEN b=b-1
1600 IF po=2 THEN b=b+1
1610 IF po=3 THEN a=a-1
1620 IF po=4 THEN a=a+1
1630 IF b=22 THEN b=2
1640 IF b=1 THEN b=21
1650 IF a=40 THEN a=1
1660 IF a=0 THEN a=40
1670 IF a=qaz AND b=wsx THEN GOTO 2
200
1680 IF a=qwe AND b=zxc THEN GOTO 22
20
1690 IF a=asd AND b=zxc THEN GOTO 22
20
1700 IF a=pl AND b=wsx THEN GOTO 22
00
1710 IF a=poi AND b=lkj THEN GOTO 2
230
1720 IF a=iop AND b=lkj THEN GOTO 2
230
1730 IF a=ik AND b=yh THEN GOTO 224
0
1740 IF a=uj AND b=yh THEN GOTO 224
0
1750 IF li=0 THEN GOTO 1850
1760 re=FN point(a,b)
1770 LOCATE a,b:IF po=1 OR po=2 THE
N INPUT,0:PRINT:ELSE INPUT,1:PRINT
1780 LOCATE PEEK(na),PEEK(nb):PRINT
" "
1790 POKE na,a:POKE nb,b
1800 IF re=3 THEN sc=sc+lg*10:GOTO
1930
1810 IF re=1 THEN GOTO 2160
1820 GOTO 1490
1830 RETURN
1840 '*****FIN DEL JUEGO*****
*****
1850 INK 0,23: BORDER 23
1860 po=0:sc=0:lg=10
1870 FOR s=1 TO 20
1880 INK 0,0:PEN 6:LOCATE 16,10:PRIN
T"GAME OVER"
1890 NEXT s
1900 GOTO 1030
1910 '*****PUNTOS*****
1920 '*****RECORD*****
1930 IF sc>hi THEN hi=sc
1940 IF sc=10000 THEN SOUND 1,60,0,3
,2,1:li=li+1:LOCATE #3,35,3:PEN#3,3:
PRINT #3,(li):GOTO 1490
1950 IF sc=25000 THEN SOUND 1,60,0,3
,2,1:li=li+1:LOCATE #3,35,3:PEN#3,3:
PRINT #3,(li):GOTO 1490
1960 IF sc=50000 THEN SOUND 1,60,0,3
,2,1:li=li+1:LOCATE #3,35,3:PEN#3,3:
PRINT #3,(li):GOTO 1490
1970 IF sc=75000 THEN SOUND 1,60,0,3
,2,1:li=li+1:LOCATE #3,35,3:PEN#3,3:
PRINT #3,(li):GOTO 1490
1980 IF sc=100000 THEN SOUND 1,60,0,
3,2,1:li=li+1:LOCATE #3,35,3:PEN#3,3
:PRINT #3,(li):GOTO 1490
1990 LOCATE #3,11,3:PEN 2:PRINT #3,(
sc)
2000 LOCATE #3,14,2:PRINT #3,(hi)
2010 ENV 1,1,14,1,7,-2,4
2020 PEN 1:SOUND 1,0,-1,0,1,0,6
2030 '*****DISPARO*****

```



```

2110 IF j>40 THEN LOCATE j-1,b:PRINT
" ":GOTO 2130
2120 NEXT j
2130 GOTO 1490
2140 RETURN
2150 '*****ANULA VIDAS*****
2160 ENV 2,1,14,1,14,-1,20:SOUND 4,0
,0,0,2,0,15:OUT &BC00,8:OUT &BD00,1:
FOR Z=0 TO 26:INK 0,Z:BORDER Z:FOR Y
=1 TO 50:NEXT Y,2:OUT &BC00,8:OUT &B
D00,0:1i=1i-1:LOCATE #3,35,3:PRINT #
3,(1i):INK 0,0:BORDER 10
2170 LOCATE a,b-1:PRINT " ":LOCATE
a,b+1:PRINT " ":LOCATE a-2,b:PRINT "
":LOCATE a+1,b:PRINT " "
2180 GOTO 1490
2190 '*****SUBPUERTAS*****
2200 CLS:INK 0,0:MODE 0:LOCATE 9,4:P
EN 2:PRINT"O.K":LOCATE 5,9:PEN 15:PR
INT"LEVEL":LOCATE 13,9:PEN 3:PRINT"2
":LOCATE 6,15:PEN 1:PRINT"(SPACE) fo
r cont":LOCATE 9,12:PEN 4:PRINT;e1$:
CALL &BB16:GOTO 1210
2220 CLS:INK 0,0:MODE 0:LOCATE 9,4:P
EN 2:PRINT"O.K":LOCATE 5,9:PEN 15:PR
INT"NIVEL":LOCATE 13,9:PEN 3:PRINT"1
":LOCATE 6,15:PEN 1:PRINT"(SPACE) pa
ra SEGUIR":LOCATE 9,12:PEN 4:PRINT;e
1$:CALL &BB16:GOTO 1050
2230 CLS:INK 0,0:MODE 0:LOCATE 9,4:P
EN 2:PRINT"O.K":LOCATE 5,9:PEN 15:PR
INT"NIVEL":LOCATE 13,9:PEN 3:PRINT"3
":LOCATE 6,15:PEN 1:PRINT"(SPACE) pa
ra SEGUIR":LOCATE 9,12:PEN 4:PRINT;e
1$:CALL &BB16:GOTO 1380
2240 CLS:INK 0,0:MODE 0:LOCATE 9,4:P
EN 2:PRINT"O.K":LOCATE 5,9:PEN 15:PR
INT"NIVEL":LOCATE 13,9:PEN 3:PRINT"2
":LOCATE 6,15:PEN 1:PRINT"(SPACE) pa
ra SEGUIR":LOCATE 9,12:PEN 4:PRINT;e
1$:CALL &BB16:GOTO 1220

```

```

2840 IF po=1 THEN GOTO 2870
2850 IF po=4 THEN 2100
2860 IF po<>1 OR po<>3 OR po<>4 THEN
2130
2870 SOUND 2,500,12,15,0,0,1:FOR j=b
-1 TO b-10 STEP -1
2880 IF j<1 THEN LOCATE a,j+1:PRINT
" ":GOTO 2130
2890 PEN 9:LOCATE a,j:PRINT CHR$(124
):LOCATE a,j+1:PRINT " ":NEXT:LOCATE
a,j+1:PRINT " "
2100 FOR j=a+1 TO A+10

```



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

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.

LISTADOS BORROSOS

En el número 1 de la revista anteriormente citada correspondiente al mes de marzo, hay un fallo de impresión en el programa sobre el IVA que ustedes editaron. Les agradecería me contestaran diciéndome cuál es

el contenido de la línea 480 del programa, pues es casi imposible leerlo en el listado.

José A. Noguera Delgado

Don José, tiene usted toda la razón; aprovechamos para presentarle a usted y al resto de nuestros lectores, nuestras disculpas por los «despistes» de imprenta que causan que ciertas líneas del listado de un programa sean apenas legibles, con las innumerables molestias que ello conlleva. La primera y principal, que el programa no funcione.

*En fin, la línea 480 del programa IVA del **Amstrad ESPECIAL** número 1 es:*

$c\$ = \text{left} \$ (a\$, 1)$

Esperamos que disfrute con nuestro programa.

PEDIRLE PERAS AL OLMO

Trabajando en CP/M se pueden grabar programas en cinta mediante el utensilio CSAVE, pero se graban como ficheros ASCII, independientemente del tipo de fichero original en disco. ¿Cómo se puede evitar esto?, es decir, si el programa en disco está en BASIC, quiero que se grabe en cinta en BASIC, no es ASCII.

José A. de la Fuente

Hemos querido expresar con el titular de su consulta, de forma apertamente no muy ortodoxa, la sensación de impotencia que nos produce tener que responderle que el

utensilio CSAVE de CP/M está pensado ESPECIFICAMENTE para grabar cualquier cosa en formato ASCII.

Para poder realizar lo que usted, con toda lógica, desea, en mi opinión sería necesario desensamblar CSAVE, ver cómo trabaja y cambiar el programa para que mandara a la cinta de una cabecera de fichero además de los datos propiamente dichos.

Es decir, necesitará usted DDT (me refiero a la herramienta de CP/M, por supuesto) y unas grandes dosis de tiempo y paciencia.

¿Mucha publicidad?

En los últimos días he descubierto, no sin cierto asombro que las páginas de publicidad han aumentado considerablemente. Llegando a alcanzar algunas semanas la cifra de 17 páginas. No sé a qué puede deberse este hecho, pero me parece que no es lógica esta subida vertiginosa del espacio dedicado a publicidad.

Joaquín Romero Ruiz

Evidentemente no te falta razón, es cierto que el número de páginas de publicidad ha aumentado, pero es igualmente cierto que el número de páginas total de la revista, cuando esto ocurre, también aumenta en cuatro u ocho páginas más, sin que aumente su precio. Al final el número de páginas no publicitarias, de la revista es superior o a lo sumo igual que en números anteriores. Tú mismo puedes comprarlo.

COMERCIAL BAZAR LEVANTE TETUAN

Distribuidor oficial de Amstrad, Spectravideo, Dynadata, Toshiba, etc.

— AMSTRAD 6128 VERDE-DISCO	84.500 PTAS.
— AMSTRAD 6128 COLOR-DISCO	119.500 PTAS.
— AMSTRAD PCW 8256	129.000 PTAS.
— AMSTRAD 472 VERDE-CASSET	59.000 PTAS.
— AMSTRAD 472 COLOR-CASSET	90.000 PTAS.
— ORDENADOR DYNADATA	
— DPC 200-64 K	34.000 PTAS.
— IMPRESORA BROTHER M-1009	36.000 PTAS.
— IMPRESORA BROTHER HR-S	21.000 PTAS.
— JOYSTICK INVESTIK	1.200 PTAS.
— JOYSTICK QUICK SHOT II	1.600 PTAS.
— IVA INCLUIDO	

SERVICIO TECNICO

Arenal, 9, 28013 MADRID Tel.: 265 68 55

MASTER HARD

Servicio Técnico
Para **AMSTRAD**

en Galicia,
León
y Asturias.

C/ Magdalena, 213

El Ferrol

Tel.: (981) 35 84 32

MASTER COMPUTER

Centro Comercial Guadalupe
Ctra. Canillas, 136-1.ª planta
Tel. 200 80 65 MADRID

Centro Comercial El Bulevar
La Moraleja
Tel. 654 16 12 MADRID

También abierto domingos de 10 a 2
Centro Comercial Ciudad Sto. Domingo
Ctra. de Burgos, km 28.
Tel.: (91) 622 12 89
Algete (Madrid) (Central)

HA LLEGADO LA TRIGONOMETRIA



Comenzamos con uno de los temas clásicos en las Matemáticas tradicionales: los ángulos y su medida. ¿Hace falta recordar qué es un radián?, pues por si acaso ahí va. Radián es: el ángulo cuyo arco de circunferencia tiene la misma longitud que el radio de ésta.

Y como una circunferencia tiene una longitud igual a:

$$2 * \pi * \text{radio}$$

es fácil intuir que dividiendo por el «radio» obtendríamos el número de radianes que completan una circunferencia: $2 * \pi$.

Podemos obtener el valor exacto en grados de uno tecleando:

$$\text{radián} = 360 / (2 * \pi)$$

y con:

PRINT radián

nos aparecerá la solución en la pantalla.

Todo esto no es más que un intento de refrescar nuestra memoria trigonométrica, puesto que el **Amstrad**, para hacer cálculos, precisará conocer las unidades que va a manejar.

De forma que si utilizamos la función DEG entenderá que las medidas vienen dadas en grados, mientras que RAD le indica que trabajamos con radianes. Y tenga en cuenta algo importante: si no hacemos ningún tipo de aclaración el ordenador entenderá que debe manejar radianes.

Pero entremos de lleno en la Trigonometría. La primera función con la que nos encontramos es SIN. Su misión es calcular el valor del seno del número o expresión numérica que va entre paréntesis. La manera de hacerlo es la siguiente:

```
DEG  
PRINT SIN (45)
```

o

```
RAD  
PRINT SIN (PI/2)
```

nos visualizará en pantalla el valor que toma esta función para ángulos cuyos valores son 45 grados y $\pi/2$ radianes. Pruebe las distintas variaciones que se le ocurran cambiando la expresión del paréntesis.





El siguiente paso va a ser COS, que nos calcula el valor del coseno de un ángulo. Su funcionamiento y utilización es semejante a los del caso anterior, así es que le vamos a proponer un experimento. Teclee el Programa 1 que calcula los cosenos de los ángulos entre 0 y 360 grados tomándolos en intervalos de 30.

Programa uno

Una vez los haya ejecutado y visto sus resultados, cambie la palabra «coseno» por «seno» y la función COS por SIN cuantas veces aparezca e inténtelo de nuevo. Ahora tendrá los valores de los senos, en vez de los cosenos. Los valores de los ángulos también estarán expresados en grados y comprobará que su funcionamiento es idéntico.

Tradicionalmente nos han dicho que la razón (o división) entre el seno y el coseno de un ángulo es la «tangente». Bueno, pues nuestra maravillosa máquina también viene dotada con una función que nos ahorra operaciones: TAN.

Además, su sistema de trabajo es semejante a las anteriores. Con:

```
DEG
PRINT TAN (60)
```

obtendremos el valor de la «tangente» de un ángulo de 60 grados. Pero veámoslo gráficamente con el Programa 2.

```
10 REM PROGRAMA I
20 CLS
30 DEG
40 PRINT "ANGULO", "COSENO"
50 PRINT "-----", "-----"
60 FOR angulo=0 TO 360 STEP 30
70 coseno=COS(angulo)
80 PRINT angulo, coseno
90 NEXT angulo
```

```
10 REM PROGRAMA II
20 ON ERROR GOTO 120
30 CLS
40 DEG
50 PRINT "ANGULO", "TANGENTE"
60 PRINT "-----", "-----"
70 FOR angulo=0 TO 360 STEP 30
80 tangente=TAN(angulo)
90 PRINT angulo, tangente
100 NEXT angulo
110 END
120 PRINT angulo, " INFINITO"
130 RESUME 100
```

Primeros repasos

Programa dos

Como puede ver es muy semejante a los anteriores pero hemos colocado dos líneas nuevas con las que detectamos los errores que se producen cuando las tangentes son demasiado grandes o demasiado pequeñas.

Hemos agrupado ambos casos y generalizado el tratamiento dando el valor «infinito» a las tangentes de los ángulos con los que se produzca el error. Estudie los distintos valores que nos devuelve esta función.

La última función es ATN, y su trabajo consiste en calcular el arco cuya tangente es la expresión numérica que colocamos entre paréntesis. Su forma general es:

$$\text{arctg} = \text{ATN} (\text{expresión})$$

También puede devolvernos el valor del arco en grados o radianes:

```
DEG
PRINT ANT (1)
```

nos dará el valor en grados y:

```
RAD
PRINT ANT(1)
```

lo hará en radianes. ¿Entiende por qué hay diferencia entre los dos valores obtenidos? Pruebe encontrar los valores del arco para tangentes de muy distinto rango, verá que los problemas que encontraremos se presentarán con los casos límites, por eso es muy importante que hagamos bien los tratamientos de error.

Y éstas son todas las funciones trigonométricas de nuestro ordenador, con ellas y las fórmulas que nos legaron los antiguos no habrá ningún problema que se nos resista y recuerde: lo verdaderamente interesante es su aplicación práctica, de modo que, ¿por qué no intenta escribir un programa que, con alguna de estas funciones, le resuelva cualquier problema cotidiano?

Mercado común

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

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

Envíanos tu anuncio mecanografiado a: **HOBBY PRESS, S.A.**

AMSTRAD SEMANAL.

Apartado de correos 54.062
28080 MADRID

¡ABSTENERSE PIRATAS!

Deseo ponerme en contacto con algún usuario de **Amstrad** que tenga el manual de COBOL en español. Mi dirección es: B. Seguí Apto, 227 07500 Manacor (Mallorca).

Desearía contactar con usuarios de **Amstrad** para intercambiar juegos, trucos, utilidades e impresiones. Dispongo de más de 130 juegos y utilidades. Interesados escribir a: Jesús Salvador Alvarez Casado. C/ Ruiseñor, 5 3.º A Izq. 18014 (Granada).

Vendo Amstrad CPC-664 Monitor fósforo verde; 11 meses, con manuales y fundas más cassette grabador especial para ordenador, más 1 joystick; más 3 discos llenos de programas y 4 vírgenes; más 10 cintas de juegos (Ajedrez 3D, Tennis, Dragontork, etc.). Todo por sólo 75.000 ptas. Además regalo colección de revistas de **Amstrad USER**. Fernando Alvarez Alonso. Tel. 460 33 94. Getxo (Vizcaya).

Intercambio programas de **Amstrad**. Contestaré a todos. Dirigirse a Vicente Bresó Flores. C/ Julio Colomer 5-1. Alfafar (Valencia).

Vendo Amstrad CPC-464 con monitor de color, en perfecto estado. Con él regalo más de 80 programas comerciales (juegos, compiladores, ensambladores, copiadores...) así como dos libros sobre el **Amstrad y el curso autodidacta de BASIC II. Llamad o escribir a: Carlos Francisco Cano Lanzadera. C/ Sáinz de Baranda, 99. Tel. (91) 273 00 18. Apto. 28007 (Madrid).**

Vendo Spectrum 48 K. con cassette, numeración, libros y juegos. Poquísimo uso. Tel. (91) 747 22 60.

Vendo Amstrad CPC-464 completo con monitor fósforo verde, acompañado de 12 juegos y 2 libros. Como nuevo. 50.000 pts. Tel. (91) 747 22 60.

Se intercambian juegos y copiones para equipo CPC-464, interesados dirigirse a. Antonio Martínez López. C/ Ramón Gallud 2-2.º. 30000 (Murcia).

Vendo ZX Spectrum con interface KEMPSTON, joystick y varios programas, todo por 15.000 pts. O cambio por productos para el CPC-464 como ratones, sintetizadores de voz, etc. Interesados a: José Ignacio Vila Alonso. Plaza de España, 27, 2.º B. 27001 (Lugo) Tel. (982) 22 07 04.

Vendo ordenador Amstrad CPC-464 con monitor en color, manual, guía de referencia BASIC para el programador; más de 200 programas (juegos, utilidades, gestión, etc.), todo en perfecto estado, por 85.000 pts. Interesados llamar al (93) 345 57 33. (Barcelona).

Vendo Amstrad CPC-464, con monitor en color, en buen estado, con 16 juegos comerciales, copiones, programas de gestión, revistas, libros de programación, etc. Todo por 70.000 pts. Tel. 31 76 03. Reus, preguntar por José M^a.

Quisiera establecer contacto con usuarios de **Amstrad** en Granada, para cambiar ideas, listados y programas, tanto de juegos como de utilidades, lenguajes, profesionales, etc. Estoy interesado en conseguir un ensamblador/desensamblador. Preguntar por Salvador Polo Navarro. C/ Sos del Rey Católico 7-6.º C. 18006 (Granada).

Intercambio toda clase de programas para **Amstrad**. Dirigirse a: Pedro Suero. C/ Mudela, 29.º 4-C. 28018 (Madrid).

Intercambio juegos para el **Amstrad CPC-464** con usuarios de Sevilla y provincia. Interesador escribir a: Rafael Joaquín Bernal. C/ Poniente, 42 o bien llamen al Tel. 78 04 89.

Vendo Amstrad CPC-464 con monitor fósforo verde en perfecto estado más de 17 juegos y guía del usuario y un manual de referencia BASIC. Todo por 50.000 pts. Interesados escribir a: Rafael Joaquín Bernal. C/ Poniente, 42. Alcalá del Río (Sevilla). Abstenerse personas que no sean de Sevilla o provincia.

Vendo videojuego **TRENSI** para TV blanco y negro o color. Dispone de tenis, hockey, frontón, squash, tiro al blanco y tiro al plato. Se entrega con dos mandos para los jugadores, rifle desmontable, alimentador e instrucciones. Está en perfecto estado y lo vendo por tan sólo 5.000 pts. Interesados llamar al Tel. (988) 74 41 38 y preguntar por Dani.

UNA JUGADA MAESTRA



Todo sobre el baloncesto americano en fascículos. BASKET USA.

Los gigantes de la cancha y sus técnicas.

Cómo se hace un campeón.

Los grandes equipos de la NBA y la liga Amateur...

BASKET USA.

Haz la mejor jugada, vé a tu quiosco... ¡Mételo en casa!

52 fascículos semanales: **195** ptas. c/u.

Oferta de lanzamiento,
los números 1 y 2 por sólo **195** ptas.

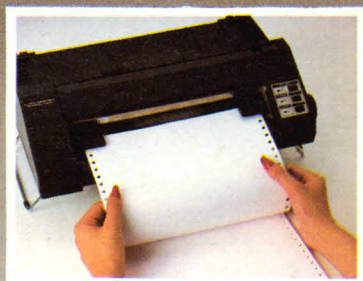
HOBBY PRESS. Para gente inquieta.

AMSTRAD DMP 2000

NO ENCONTRARA UNA IMPRESORA QUE LE HAGA TAN BUEN PAPEL.



Soportes abatibles que permiten colocar el papel bajo la impresora.

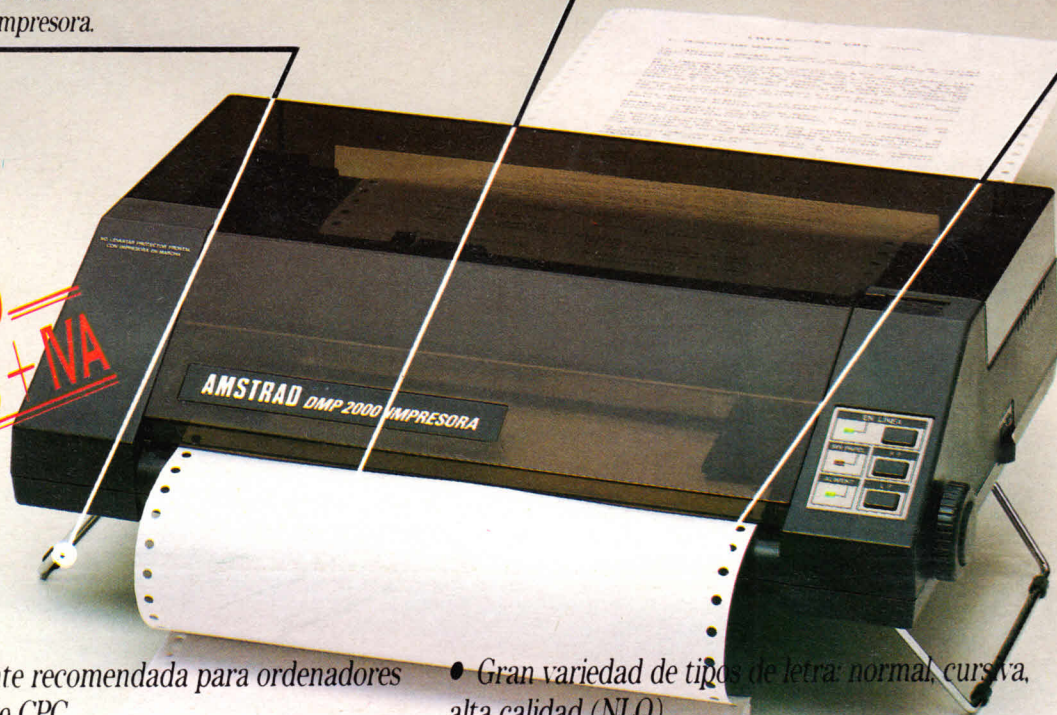


Cómodo sistema de carga frontal del papel.



Admite diferentes anchos de papel, tanto continuo (de 114 a 254 mm.) como hojas sueltas (102 a 241 mm.)

**POR SOLO
39.500 PTAS + IVA**



• Especialmente recomendada para ordenadores AMSTRAD serie CPC.

• Conectable a cualquier ordenador con interface centronics.

• Velocidad de impresión de 105 caracteres por segundo.

• Gran variedad de tipos de letra: normal, cursiva, alta calidad (NLQ)

• 40, 66, 80 y 132 caracteres por columna.

• Impresión de gráficos punto a punto en diferentes densidades.

• 96 caracteres ASCII y 8 subjuegos internacionales.

¡¡ Increíble !!

GRUPO INDESCOMP

AMSTRAD ESPAÑA