

# Mundo Spectrum

LOS MEJORES LISTADOS PARA TU MICRO

EDITA:  SGRANSA

• año - I • número - I •

**mundo  
precio**  
PVP  
IVA INC 200 pesetas

**SUPER  
MUELLE**  
**MISTER  
HUEVO**

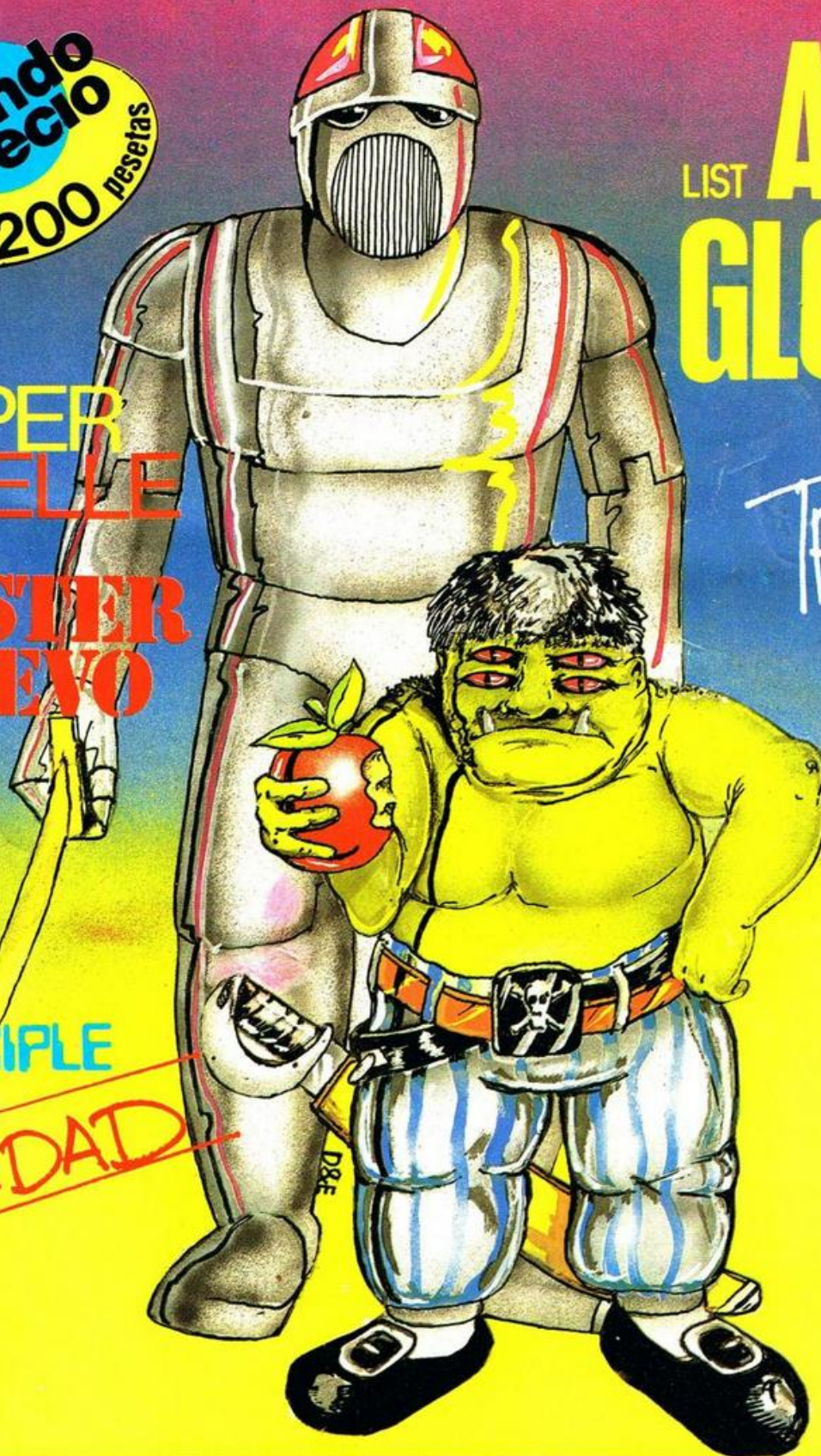
**DISCIPLE**

**NOVEDAD**

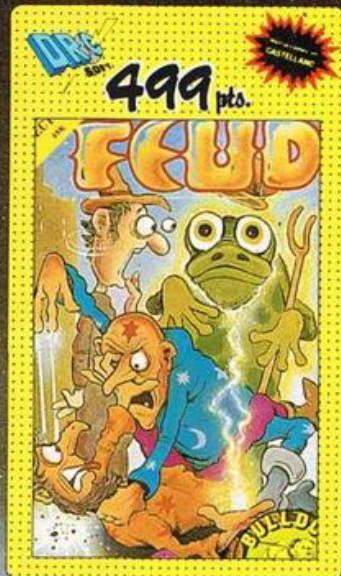
LIST **APPLE  
GLOTON**

lunar car

TRUCOS



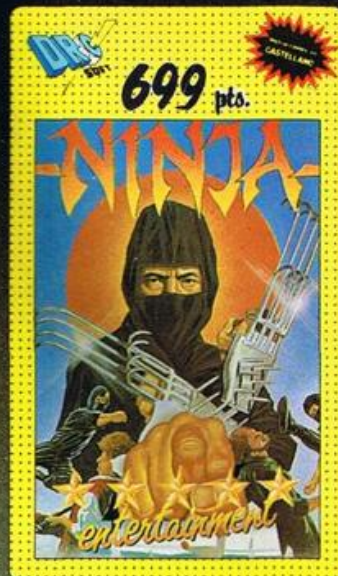




**FEUD**  
Con tu sabiduría puedes encontrar los ingredientes de las POCIONES MAGICAS, mientras recitas los conjuros de tu maligno enemigo LEANORIC.



COMMODORE  
SPECTRUM  
AMSTRAD  
MSX



**NINJA**  
Eres un NINJA, demuestra tu destreza luchando contra los mejores Budokas en un combate a MUERTE.

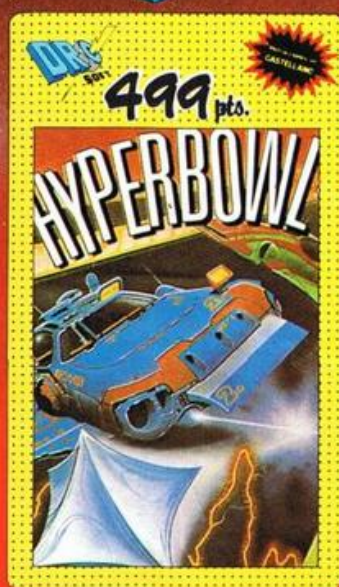
COMMODORE  
SPECTRUM  
AMSTRAD

# ¡NUNCA PUEDE SER!!

499 pts.

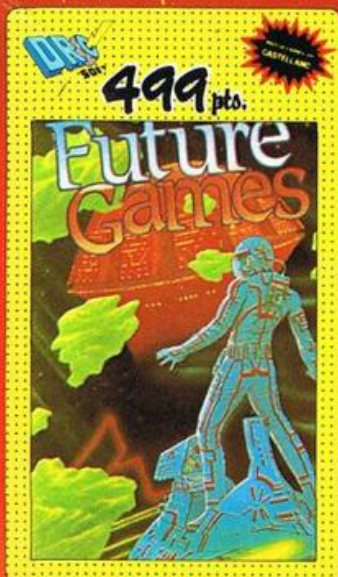
SERIE M.A.D. 699 pts.

COMMODORE  
SPECTRUM  
AMSTRAD  
MSX



**HYPERBOWL**  
En el año 3.600 el hockey sobre hielo ha alcanzado el nivel de máxima tecnología. Diez clases diferentes de naves, se enfrentan en un campo metálico...

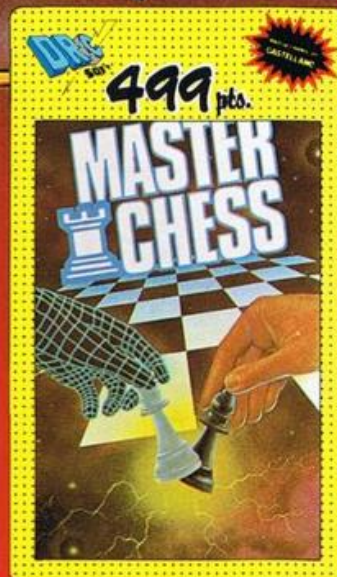
SPECTRUM



**FUTURE GAMES**  
Juega tu libertad contra la muerte. Si ganas, la consigues, si pierdes, desde el lugar donde estés, ya no te importará.



**TERMINUS**  
Un grupo de gamberros, que siembra el pánico por donde pasa, ha decidido rescatar a su líder del planeta prisión TERMINUS.



**MASTER CHESS**  
Tu ordenador es tu mejor contrincante. Piensa más rápido que tu, pero tu puedes jugar mejor y ganarle. ¡INTENTALO!

SPECTRUM  
AMSTRAD  
MSX



Y otros 120 juegos mas.

MASTERTRONIC



# EDITORIAL

Bienvenidos a este primer número del Mundo Spectrum. Como podréis observar, el grueso de nuestra revista consta de programas listados, y esto es así porque pensamos que la mayoría de los usuarios del Spectrum debemos en buena parte nuestros conocimientos sobre informática al "muy noble" arte de teclear programas. De forma que nuestro empeño estará centrado en ofrecer un amplio repertorio de programas de calidad que nos permitan seguir aprendiendo de la manera más sencilla, útil y divertida posible.

En este número hemos seleccionado cuatro divertidos juegos así como una utilidad en Código Máquina y los hemos acompañado de unos amplios comentarios, que os permitirán saber línea a línea, qué es lo que estamos haciendo y porqué.

Además de un par de interesantes trucos, hemos incluido un artículo sobre la memoria del Spectrum, como elemento de repaso y estudio y otro sobre el novedoso Disciple, un producto que supone un notable avance en el tema de la ampliación de prestaciones del Spectrum.

No queremos terminar sin deciros que estamos abiertos a todo tipo de consultas y sugerencias que deseéis hacernos. Pretendemos además que esta relación sea fluida y constante. Ahora bien, no podemos comprometernos a responder por correo, por lo que en el próximo número abriremos una sección para responder o aclarar cualquier tema solicitado.

Por otro lado, también podréis contar con una sección de intercambio directo entre lectores, por lo que desde ya, esperamos vuestros anuncios.

# SUMARIO

<b>4 TRUCOS</b>	Movimiento continuo Lectura del teclado con IN
<b>5 SOFTWARE</b>	La organización de la memoria en el Spectrum
<b>8 HARDWARE</b>	El Disciple
<b>12 MUNDO LIST</b>	Apple Glotón
<b>16</b>	Luna Car
<b>23</b>	Super Muelle
<b>27</b>	Mister Huevo
<b>31</b>	UTILIDADES EN CODIGO MAQUINA: Blanco y Negro
<b>18 JUEGOS</b>	Spirits. Don Quijote. Dustin.
<b>34 LIBRERIA</b>	Ficheros en Basic El ordenador en la Educación Básica

## RECOMENDACIONES PARA TECLEAR UN PROGRAMA

Para aquellos que no estéis muy acostumbrados a teclear programas, os daremos a continuación una serie de instrucciones o pasos a seguir para que os resulte más sencillo y las probabilidades de que se produzcan errores en la ejecución de los programas sean menores:

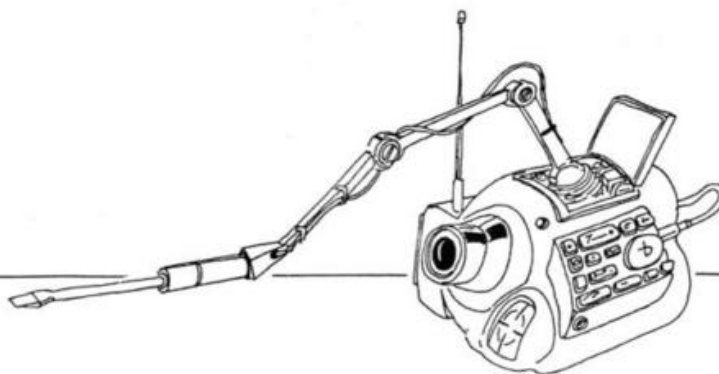
1. Como muchos ya sabréis, las líneas que después de su número correspondiente tienen una instrucción REM o un apóstrofe, no son consideradas por el programa sino que únicamente sirven como comentario aclaratorio para el programador y por lo tanto no es necesario que las copieis completas cuando aparezcan en nuestros listados, en cambio si es conveniente que pongáis el número de línea y la instrucción REM o ' (ejemplo 200 REM o 200') para evitar que puedan producirse errores al ejecutarse en el programa alguna instrucción de salto (GOSUB, GOTO, etc.)

2. Es fundamental que los números de línea los copiéis exactamente como figuran en el programa original.

3. Es conveniente, sobre todo en programas largos, salvar periódicamente a cinta o disco la parte que llevemos tecleada del listado antes de probar su funcionamiento para evitar la pérdida completa del programa en caso de que exista un error en el código máquina o bien se produzca un corte accidental de corriente, etc.

Dichas grabaciones periódicas es aconsejable no efectuarlas sobre la copia anterior o con el nombre en caso de hacerlas en disco.

Mundo Spectrum es una publicación del Grupo Editorial SYGRAN S.A. Polig. Ind. Valdonaire. C/ Apolonia Hernández. HUMANES (Madrid). **Director:** Angel Herrero Fernández. **Director Adjunto:** Andrés Franco. **Director Técnico:** Luis Sanguino. **Coordinador Software:** Angel García. **Secretaría Redacción:** Mercedes Matons. **Colaboradores:** Antonio García, Angeles Guindal, Carmen Torres, J. Vicente Pons. **Diseño e Ilustraciones:** Javier Caballero. **Publicidad:** GENESIS. Tomás López, 3-6º, 28009 Madrid, Tel. 401 77 54 **Fotocomposición:** Fermar C/. Sillfide 10. **Imprime:** Gráficas Osiris S.A. Brañuelas, 29. Distribuye: G.M.E., plaza Castilla, 3. Madrid. Depósito Legal: M-1 31875 1987. Reservados todos los derechos.



## MOVIMIENTO CONTINUO

El valor que contiene, es el código ASCII de la tecla pulsada. ¡¡¡Prueba el ejemplo y verás!!!

```

10  REM *****
20  REM * MOVIMIENTO CONTINUO *
30  REM *****
40  REM
50  LET X=10
60  LET Y=10
70  LET T=PEEK 23560
80  IF T=81 THEN LET X=X-1
90  IF T=85 THEN LET X=X+1
100 IF T=79 THEN LET Y=Y-1
110 IF T=80 THEN LET Y=Y+1
120 IF X<0 THEN LET X=21
130 IF X>21 THEN LET X=0
140 IF Y<0 THEN LET Y=31
150 IF Y>31 THEN LET Y=0
160 PRINT AT X,Y; "*"
170 BEEP .05,10
180 PRINT AT X,Y; " "
190 GO TO 70

```

## LECTURA DEL TECLADO CON IN.

número de orden de la tecla, de 0 a 5 contadas de izquierda a derecha en la tabla.

En el pequeño ejemplo que se acompaña aparecen las 8 direcciones de las semifilas con los valores de las teclas que están pulsadas. Observa como cambian al pulsar las teclas de tu Spectrum.

[illegible]

```
0000      *****  
0000      * LBB TEOLDOO CON IN *  
0000      *****  
0000  
0000      FOR N=0 TO 7  
0000      LET B=B84+B86*(B85-B+N)  
0000      PRINT AT N,0;D;" -> ";IN D  
0000      NEXT N  
0000      GO TO 30
```

SOFTWARE

# LA ORGANIZACION DE LA MEMORIA



## LA ORGANIZACION DE LA MEMORIA EN TU SPECTRUM

De todos es sabido, que el Spectrum incorpora como microprocesador, el Z80A, una variación más rápida del Z80 construido en California por Zilog.

El Z80A es un microprocesador muy rápido,

es capaz de procesar 875.000 instrucciones simples por segundo. Además, es capaz de direccionar 65536 direcciones de memoria (64 Kbytes). El Spectrum 48k dispone de las 65536 direcciones, mientras que el 16K solo dispone de 32.768.

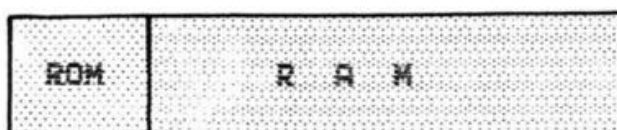


Figura 1



# SOFTWARE

Estas direcciones de memoria se reparten como sigue:

**La ROM** (memoria solo de lectura). Ocupa los 16K primeros, y contiene el sistema operativo (7K), el traductor BASIC (8K) y el juego de caracteres (1K). Los integrados que albergan la ROM, están sellados de forma que no se puede alterar su contenido.

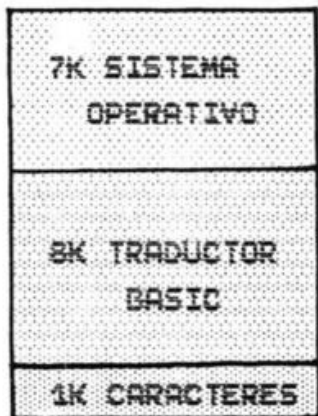


Figura 2

**La RAM** (Memoria de lectura y escritura). Ocupa en el Spectrum 16K, 16K y en el de 48K, 48K (de ahí su denominación). La información que contienen los chips de la RAM puede modificarse, puesto que estos no están sellados. Con la pérdida de la tensión, los datos de la RAM se borran, por ello es necesario grabarlos en un soporte magnético (disco, cinta ó microdrive).

La memoria RAM se divide en muchas partes, unas de ellas ocupan direcciones fijas, otras varían de tamaño y localización, de forma que para saber donde comienzan, es necesario consultar con la función PEEK las variables del sistema. En la figura 3 se listan los bloques de la RAM con la dirección fija, o con la variable del sistema a leer.

MAPA DE MEMORIA DEL SPECTRUM	
0	ROM
10384	ARCHIVO DE PANTALLA
22528	ARCHIVO DE ATRIBUTOS
23296	MEMORIA IMPRESORA
23552	VARIABLES DEL SISTEMA
23734	MAPAS MICRODRIVE
CHANS	INFORMACION CANALES
PROG	PROGRAMA BASIC
VARs	AREA DE VARIABLES
E-LINE	AREA DE EDITAR
WORKS	AREA DE TRABAJO
STKOOT	PILA DE CALCULADOR
STKEND	MEMORIA DE RESERVA
SP	PILA DE MAQUINA
UDG	GRAFICOS USUARIO

Figura 3

Pero comentemos más detenidamente estas secciones:

- El archivo de pantalla.— Ocupa las direcciones 16384-22527 (6K). A su vez se divide en 3 bloques de 2K cada uno que abarcan, tercios de la pantalla.

A nivel de carácter la pantalla tiene 24 filas  $\times$  32 columnas. Normalmente, en BASIC las dos últimas líneas son reservadas para entrada de datos y mensajes de error, por ello no son accesibles con PRINT.

A nivel pixel, la pantalla puede direccionar  $256 \times 192$  puntos, pero por la misma razón que antes, PLOT solo llega hasta 176 en vez de 192.

- El archivo de atributos.— Direcciones 22528-23295 (768 bytes). Ocupa 24 filas  $\times$  32 columnas = 768 byte. Se emplea un byte por carácter para definir sus atributos.

La forma en que se calcula es:

Tinta + Papel  $\times$  8 + Brillo  $\times$  64 + Flash  $\times$  128

Cuando usamos la función ATTR (x,y) en BASIC, ésta lo que hace es devolver el valor de la memoria que define los atributos del carácter situado en las coordenadas x e y. Equivale a hacer:

LET ATTR = PEEK (22528 + X  $\times$  32 + Y)

- La memoria intermedia de la impresora.— Ocupa las direcciones 23296-23551 (256

bytes). A esta zona se la conoce más comúnmente como PRINTER BUFFER. Si el Spectrum no tiene conectada una impresora, esta zona puede ser utilizada para guardar datos, teniendo cuidado de no usar LPRINT, LLIST o COPY.

- Las variables del sistema.— Localizadas en las direcciones 23552-27333 (182 bytes), son las que usa el microprocesador para su trabajo con el sistema.

Las variables pueden ser de 1, 2 o más bytes. Cuando son de 2 bytes se almacena primero el byte "low" (de menor peso) y a continuación el byte "high" (de mayor peso). Ejemplo:

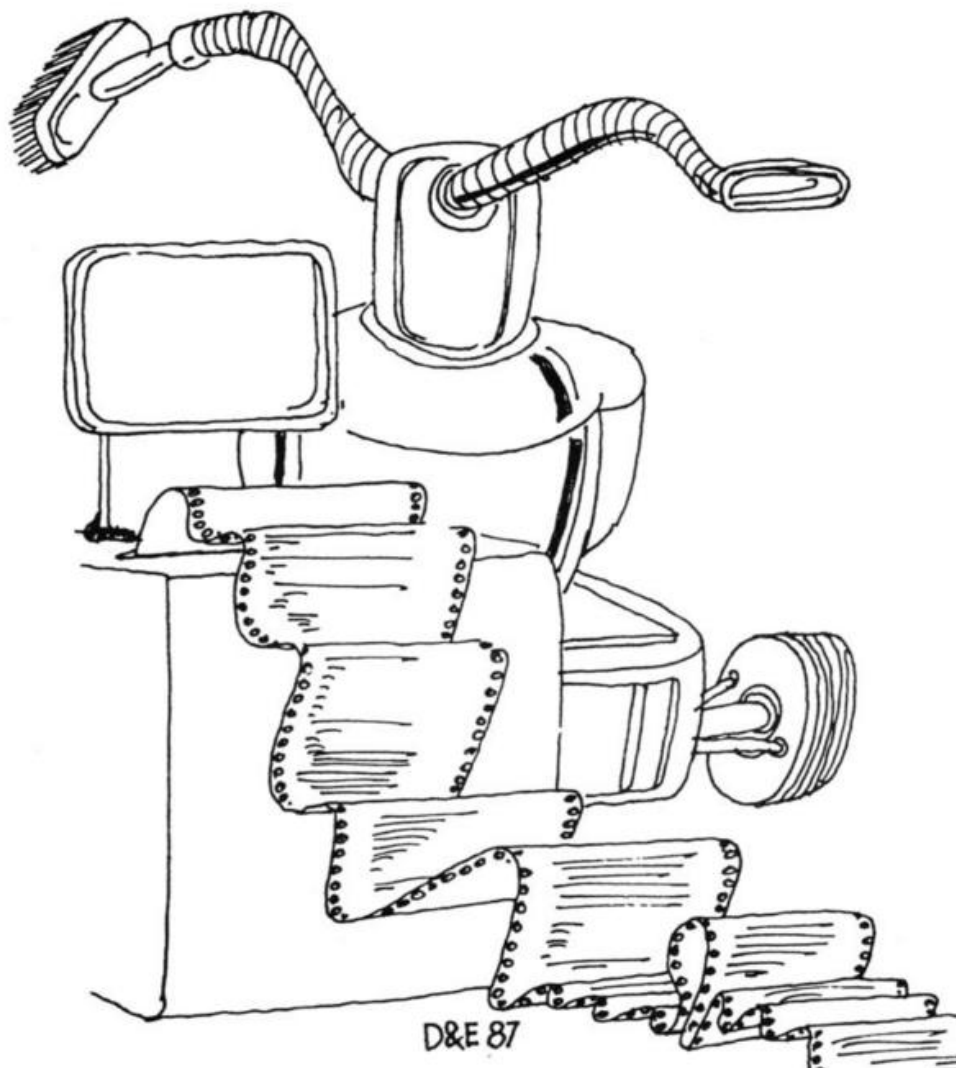
— Para saber donde comienza nuestro programa BASIC haríamos:

LET BASIC = PEEK (23635) + 256  $\times$  PEEK (23636)

De estas variables se puede sacar una valiosa información. Además podemos alterar su contenido (con conocimiento de causa) para forzar distintas situaciones. Ejemplo:

POKE 23658,5 obliga a cursor en mayúsculas.

POKE 23658,0 obliga a cursor en minúsculas



# SOFTWARE

- Mapas de Microdrive.— Comienza en la 23734 y finaliza en CHANS-1. Al conectar los microdrives, todo lo que haya por encima de los mapas de microdrives se desplazará hacia arriba tantos bytes como estos ocupan. Por esto, casi todas las zonas que siguen a este guardan su dirección de comienzo en las variables del sistema.

- Información para canales.— Comienza en CHANS (23631 y 23632). Son 4 los canales: K, S, R, P.

- El área del BASIC.— Comienza en PROG (23635 - 23636). A partir de aquí, se almacenan las líneas del programa BASIC:

- Primero el n° de línea. Cambiando el orden normalmente seguido, primero el byte de peso y luego el de menos peso.

- En segundo lugar se almacena la longitud de la línea en bytes.

- La presencia del código 13 nos marca el fin de la línea.

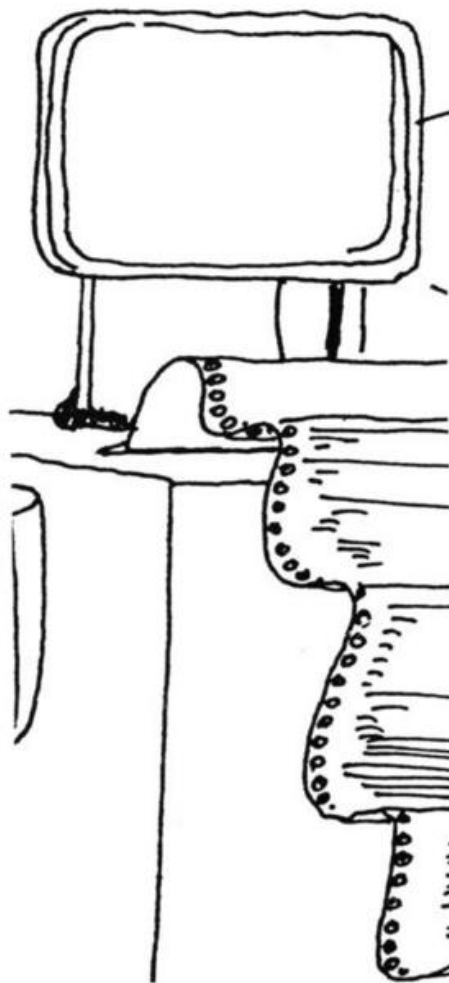
- A continuación vendría la siguiente línea, que en caso de no existir, los códigos 225, 0 marcarán el fin del programa.

- La zona de las variables.— Comienza en VARS (23627 y 23628). Aquí se almacenan las variables, ya sean numéricas o alfanuméricas, o tablas de ambos tipos, la forma en que se almacenan es la siguiente:

- Variable numérica: Código de la letra del nombre, en minúscula (1 byte) y 5 bytes que guardan el valor del número en coma flotante.

- Variable alfanumérica: Código de la letra del nombre en mayúscula (1 byte), 2 bytes con la longitud de la cadena (sólo limitada por la memoria) y los códigos de los caracteres que forman la cadena.

- Tabla alfanumérica (DIM): Se reserva en memoria el número de bytes, resultado del producto de la altura por la anchura.



- Tabla numérica: Igual que en las alfanuméricas, pero reservando 5 bytes para cada elemento.

- El área de editar.— Comienza en la varia-

ble E-LINE (23641, 23642), y contiene la línea que se va a editar, o la línea sobre la que estamos trabajando actualmente.

- El área de trabajo.— Comienza en WORKSP (23649 y 23650). Cada vez que se emplea se expande, y cuando deja de ser necesaria, se "limpia" para dejar libres las direcciones de memoria que ocupa.

- La pila del calculador.— Comienza en STKBOT (23651 y 23652) y finaliza en STKEND (23653 y 23654). Su funcionamiento es similar a la pila de máquina, el último en entrar es el primero en salir. En esta zona se guardan en coma flotantes los números que van a ser usados por el calculador.

- La memoria de reserva.— Comienza en STKEND y llega hasta la dirección donde comienza la pila de máquina, su longitud depende de la versión del ordenador.

- La pila de máquina.— Es la que usa el microprocesador para almacenar n° de 2 bytes con el uso de las instrucciones PUSH y POP.

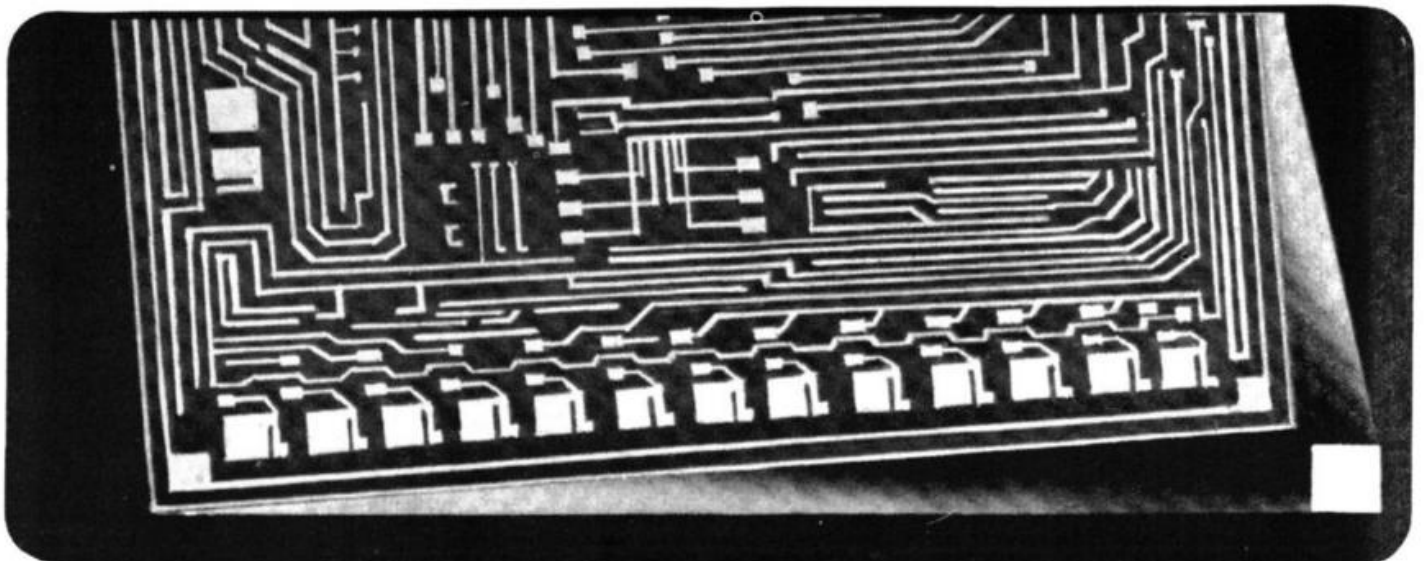
- La pila de GOSUB.— En esta zona, se almacenan los saltos que se realizan en BASIC mediante la instrucción GOSUB, para que al volver con el RETURN, el programa continúe en su sitio.

- La zona de los U.D.G.S. (gráficos definibles por el usuario).— Permite la redefinición de 21 gráficos. Comienza en UDG (23675, 23676) y tiene una longitud de 168 bytes =  $21 \times 8$ .

Es posible cambiar en la variable del sistema UDG, la dirección de comienzo de esta zona, para disponer así de más bancos de gráficos.

El próximo mes volveremos con otro tema de interés que nos permita profundizar en el conocimiento de nuestro Spectrum.

¡¡¡Buen aprendizaje!!!





## EL DISCIPLE

**EL DISCIPLE** es un interface de reciente aparición en el mercado, que facilita la conexión directa y simultánea de un Spectrum (sea cual sea el modelo) a:

- Uno o dos joysticks standard
  - Una o dos unidades de disco que pueden ser de 5 1/4", 3 1/2" ó 3", de 400 de 80 pistas, simple o doble cara y simple o doble densidad. Conexión standard tipo: Shugart SA400.
  - Una impresora con conexión paralela Centronics.
- Además dispone de:
- Un conector para los restantes periféricos.
  - Red local multiusuario, compatible con la Zx Net del Interface 1 de Sinclair, utilizable por 64 usuarios.
  - Transfer incorporado: un pulsador permite almacenar en disco el contenido íntegro de la memoria.

Este completo interface se acompaña con un cassette que contiene el programa UTILITIES PROGRAM, que nos permite adaptar cuestiones sobre el número de unidades de disco (1 ó 2), el tipo (40 ó 80 pistas y simple o doble cara) y el step-rate (retardo por paso) del disco. Nos pregunta también si vamos a emplear la

impresora, en caso afirmativo no permite configurarlo, si deseamos que parpadee el borde durante las operaciones en disco, y si se va a usar la red local. Si se va a usar nos pregunta si es la maestra, y si no, el número de estación. También nos permite formatear discos en simple o doble densidad, con o sin sistema.

Por último nos ofrece las opciones de repetir, para corregir errores cometidos en la introducción de datos, y grabar el programa utilities en disco.

**EL DISCIPLE OFRECE  
TODO UN MUNDO DE  
POSIBILIDADES**

## LOS JOYSTICKS

Para conectar los joysticks disponemos de dos tomas tipo ATARI. Kempston en la derecha, y Sinclair para la izquierda y derecha.



## LAS UNIDADES DE DISCO

Para empezar a trabajar con ellas es necesario cargar el sistema que hemos configurado con el UTILITIES PROGRAM. La forma de hacerlo es pulsar RUN y ENTER tras haber inicializado el Spectrum.

Para seleccionar la unidad escribiremos: D1 para la unidad de disco A, D2 para la B o D \* para la actual. Se puede emplear también la sintaxis del microdrive, donde D1 es reemplazado por \*"m";1;.

Podemos obtener dos tipos de directorio, uno normal con CAT 1 o CAT 2, y otro simplificado con CAT 1! o CAT 2!.

La información proporcionada en el primer caso es:

El número de orden del programa, el nombre (10 caracteres como máximo), los Kbytes que ocupa, y el distintivo de tipo.

En cambio en el segundo caso solo se listan los nombres.

Las operaciones que se pueden realizar en el disco son:

**Grabación**, con SAVE D1 "Nombre", que admite autoejecución con LINE xxxx.

**Verificación**, con VERIFY D1 "Nombre"

**Carga**, con LOAD D1 "Nombre"

**Renombrar un fichero**: ERASE D1 "ejemplo 1" TO "ejemplo 2"

**Copiar ficheros**: COPY D1 "ejemplo 1" TO "copia". Los ficheros Snapshot de los que hablaremos más adelante no pueden ser copiados así. Las copias se pueden realizar dentro del mismo disco, o de un disco a otro.

**Borrar ficheros**: ERASE D1 "fichero 1"

**Crear ficheros autoejecutables**: ERASE D1 "fichero" TO "Autoload"

**Formatear un disco**: FORMAT D1 para hacerlo con doble densidad. FORMAT SD1 para hacerlo en simple densidad.

Esta operación necesita unos 8000 bytes a partir de la dirección 49.152.

Para copiar el sistema:

SAVE D1 "System" CODE 0, 6144.

## LA RED COMUNICA HASTA 64 SPECTRUM

**Salvar ficheros con el botón Snapshot**: Es posible grabar un programa en el pto. actual en que se encuentra, pulsando este botón. Sólo se pueden grabar en el DISCO 1, y han de ser programas de 48 K (no son posibles los de 128 K). Los ficheros se grabarán como SNAP A el primero, SNAP B el segundo, etc.

Para cargar un fichero SNAP se debe teclear S después del nombre del fichero. Igualmente se hará si se usa la sintaxis de microdrive.

La pulsación del BREAK interrumpe la ejecución del programa BASIC, pero las operaciones en disco continúan hasta completarse.

Se dispone de una sintaxis abreviada. LOAD P1, carga el programa numerado como 1 en el directorio. VERIFY Y MERGE también se pueden utilizar.

Tenemos además la posibilidad de usar símbolos comodines para COPY, CAT y ERASE.

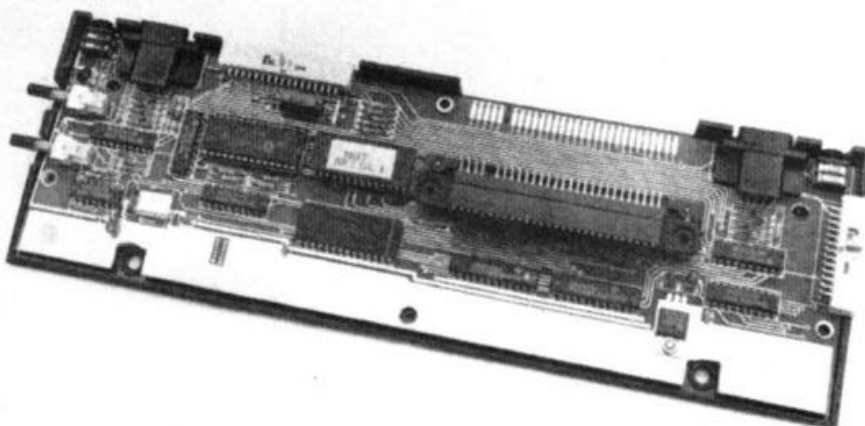
Al introducir un asterisco (\*), da igual el resto del nombre. Por ejemplo, ERASE D1 "n\*" borra el directorio todos los programas cuyo nombre empiece por "n".

La interrogación (?) es igual que el asterisco, pero para el carácter donde se encuentra. Ejemplo:

ERASE D1 "?YO?TU\*", borrará los programas donde sea "YO" la segunda y tercera letra, y "TU" la quinta y sexta.

Si reseteamos nuestro spectrum, para restablecer el sistema operativo habrá que teclear: OUT 123,0.

Junto al botón Snapshot, el interface DISCIPLE lleva el botón de inhibición. Se utiliza si otro periférico no es compatible con el DISCIPLE. Con el botón en su posición más in-



# HARDWARE.

terior e introduciendo OUT 31,0, se desactivan las unidades de disco y la ROM. La salida de impresora y las entradas de los joysticks quedan activas.

Para volver a activar la ROM del DISCIPLE habrá que pulsar de nuevo el botón y teclear OUT 31,16.

## COMO USAR LA IMPRESORA

La impresora debe tener la conexión paralela CENTRONIC.

Los comandos que se pueden utilizar son:

LPRINT: Que admite los comandos AT y TAB

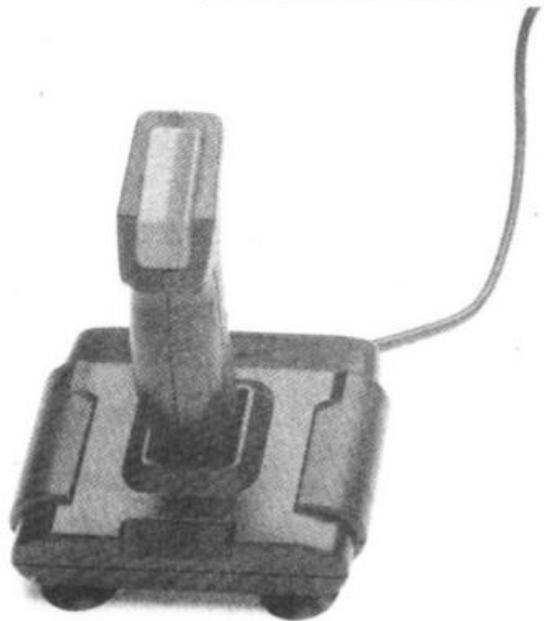
LLIST: Lista el programa por impresora.

### SNAPSHOT VUELCA PROGRAMAS A DISCO Y PANTALLAS A IMPRESORA

COPYSCREEN\$: Copia la imagen de la pantalla por impresora.

Para enviar caracteres de control a la impresora, es necesario precederlo del CHR\$ (27), uno para cada código a enviar.

Es posible imprimir con el botón Snapshot. Si se pulsa CAPS SHIFT y manteniéndolo pulsado se pulsa el botón Snapshot, la pantalla es volcada a la impresora.



## LA RED DE RECURSOS COM—PARTIDOS

Es posible unir hasta 63 ordenadores para lo cual se necesita un cable apantallado con un jack standar de 3,5 mm en cada extremo, formando redes de 3 tipos:

—Red de recursos compartidos: Existe una estación Master con uni-

dad de disco e impresora, y 62 posibles estaciones esclavo que se numeran con FORMAT n, donde n es un número de 2-63. Operaciones posibles:

- Master envia fichero a todas las estaciones:

SAVE NO a la master

LOAD NO a la esclavo

- Esclavo envia fichero a todas las estaciones:

SAVE NO a la esclavo

LOAD NO a el resto

- Master envia fichero a una estación:

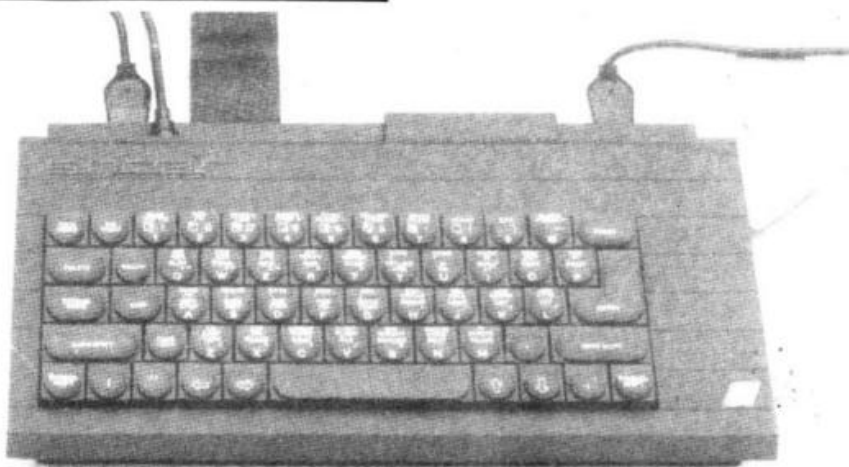
SAVE N6 Master

LOAD N1 Esclavo





# HARDWARE



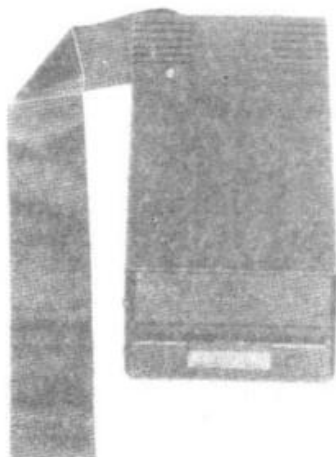
- Un esclavo envía un fichero a una estación:

SAVE N6 Master

LOAD N4 estación receptora

- El master inspecciona una pantalla: LOAD N4 Screen\$. Se detiene el trabajo del esclavo hasta que la pantalla es transferida a la unidad Master.

- El esclavo puede consultar el directorio del disco Master (CAT D1), cargar un fichero del disco Master (LOAD D1 "Nombre"), enviando (SAVE D1 "Nombre".0) a la impresora. Si ésta está ocupada, el esclavo ha de esperar a que termine la tarea actual antes de continuar. Si varios usuarios acceden a la vez tiene preferencia el de menor número de estación.



## RED DE ESTACIONES INDEPENDIENTES

Cada estación dispone de unidad de disco e impresora, y debe configurarse como unidad Master. Para numerarse se emplea FORMAT N con el número de estación (FORMAT N3). Operaciones posibles:

—enviar ficheros a todas las estaciones.—

El emisor SAVE NO

El receptor LOAD NO

—enviar fichero a una sola estación.—

El emisor manda núm. estación receptora

SAVE N4

El receptor identifica estación emisora y teclea LOAD N3

## REDES MIXTAS:

Es una red de recursos compartidos con varias estaciones Master.

Recorta o fotocopia y envía a: Mundo Spectrum. C/ Tomás López, 36º, 28009 MADRID

Deseo recibir más información sobre EL DISCIPLE

Nombre .....

Dirección .....

Población .....

Provincia .....

Modelo ordenador .....

Tel. .... C.P. ....





[illegible]

que te perseguirá por todo el  
cementerio.

Para poder moverte utiliza las teclas siguientes:

q.- arriba, a.- abajo, p.- derecha, o.- izquierda.

## Comentario Programa

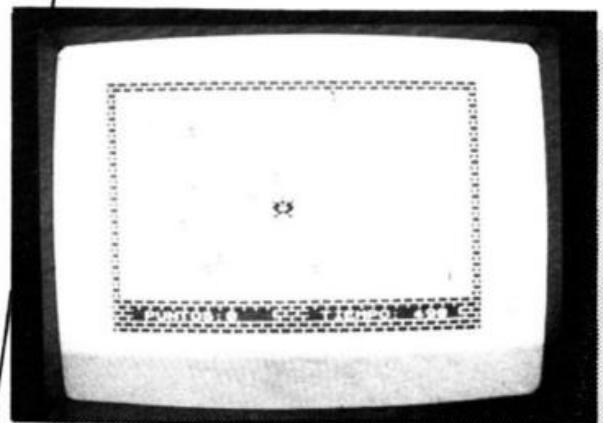
En la línea 10 junto con las líneas que van de la 8000 a 8070 se encuentran los Datos de los UDGs del juego.

De las líneas 25 a la 58 se realiza un pequeño bucle en el que se imprime una pequeña pantalla de presentación antes de empezar el juego.

En la línea 80 se efectúa un Poke con la intención de mantener en minúsculas el cursor del ordenador.

En la línea 90 se realiza un CLS de la pantalla y se colorea border, tinta y papel según la necesidad del juego.

De la línea 100 a la línea 130 se inicializan todas las variables del programa, las cuales se co-



```

1445 IF T<0 THEN LET T=0: GO TO
2500
1447 IF P<0 THEN LET P=0
1450 IF P<100 THEN PRINT PAPER 1
1455 IF P<1000 THEN PRINT PAPER
1470 PRINT PAPER 1: INK 7: AT 20,
10: P; AT 20, 25: T
1480 IF I=1 THEN GO SUB 1500
1490 IF I=2 THEN GO SUB 1500
1500 GO TO 1000
1501 REM ***** OBJETOS *****
1502 REM ***** IMPR. *****
1510 LET S=0: BEEP .005,30: BEEP
1520 IF 1+INT (RAND*18) THEN PRINT INK 4: AT
N: LET S=S+1: BEEP .005,30: BEEP
1530 IF S>11 THEN PRINT INK 4: AT
M: LET S=0: BEEP .005,30: BEEP
1540 RETURN
1550 REM ***** MANZANAS COMIDAS *****
1560 PRINT AT V,W: *****
1610 LET V=V: LET W=W
1620 PRINT V,W: INK 4: AT V,W
1630 IF ATTR=2 THEN IF V,W="N"
1,4+1)=50 OR ATTR=50 OR ATTR=50
1640 TO 2500 ATTR=50 OR ATTR=50
1650 IF V,W=18 THEN PRINT AT V,W
NEXT S: PRINT AT V,W: BEEP .005,30
1660 LET W=S+INT (RAND*20): LET V=V
1670 RETURN
1680 REM ***** COLISIONES *****
1690 IF ATTR=50 OR ATTR=50 OR ATTR
+1,4)=50 OR ATTR=50 OR ATTR
2520 IF ATTR=50 THEN GO TO 2500
+1,4)=50 OR ATTR=50 OR ATTR=50
ATTR=50 OR ATTR=50 OR ATTR=50
LET 4=4: FOR S=1 TO 20: LET X=X
: NEXT S: BORDER 5: LET P=P-10
1700 RETURN
1710 IF ATTR=1 THEN IF ATTR (X,4)=5
1,4+1)=50 OR ATTR=50 OR ATTR=50
1720 TO 2700 ATTR=50 OR ATTR=50
1730 RETURN
1740 FOR S=1 TO 20: BEEP .005,30
DER S: INT (RAND*6): NEXT S: BOR
1750 LET P=P+20
1760 LET W=1: LET W=S+INT (RAND*2
1770 RETURN
1780 REM ***** FINAL *****
1790 REM *****

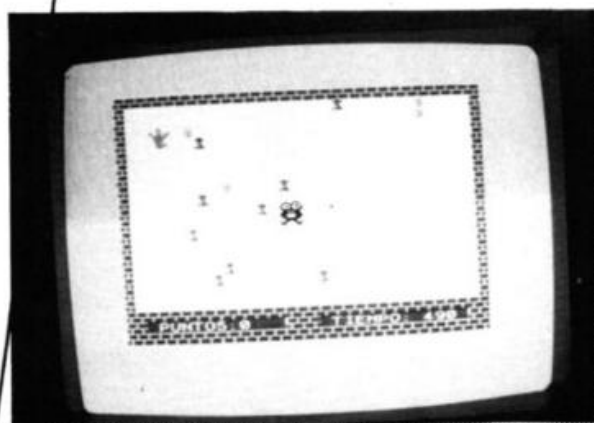
```

responden de la siguiente manera: **c.**- contador general; **x,y.**- movimiento del traga manzanas; **xb,yb.**- borrado del come manzanas; **i.**- contador parcial; **e,f.**- movimiento fantasma; **eb,fb.**- borrado fantasma; **v,w.**- variables manzana; **vb,wb.**- borrado manzana; **P.**- puntos; **T.**- contador tiempo; **g.**- contador manzanas; **h.**- contador lápidas; **ei,fi.**- variables del movimiento del fantasma, como incremento y decremento.

De la línea 200 a la línea 230 se imprime en pantalla el paisaje o contorno y los objetos.

De la línea 1000 a la línea 1080, aparece el bucle principal y se realiza la impresión y el movimiento de nuestro muñeco.

De la línea 1100 a la línea 1300 se realiza la impresión y el movimiento del fantasma incrementándose y decremen-





```

1000 FOR #=1 TO 70: OUT #=1:
1010 INK INT (RND#5): PT X#1:
1020 NO+40: NEXT #
1030 FOR #=1 TO 15: PRINT INK #
1040 PAPER 7: INK 0: PT #,1
1050 PUNTOS OVER #: PT #,10:
1060 VENT 1: PAPER 5: PT 14,0: DTR
1070 IF INKEY#="E" THEN GO TO 25
1080 IF INKEY#="D" THEN STOP
1090 FOR #=1 TO 4: STEP -.1: PRIN
1100 INK 0: PAPER 7: PT #,1: BEEP ,
1110 NEXT #
1120 LET #=1: THEN LET X=12: LET #
1130 LET #=1: THEN LET X=17: LET #
1140 LET #=1: THEN LET X=17: LET #
1150 LET #=1: THEN LET X=17: LET #
1160 LET #=1: THEN LET X=17: LET #
1170 LET #=1: THEN LET X=17: LET #
1180 LET #=1: THEN LET X=17: LET #
1190 LET #=1: THEN LET X=17: LET #
1200 LET #=1: THEN LET X=17: LET #
1210 LET #=1: THEN LET X=17: LET #
1220 LET #=1: THEN LET X=17: LET #
1230 LET #=1: THEN LET X=17: LET #
1240 LET #=1: THEN LET X=17: LET #
1250 LET #=1: THEN LET X=17: LET #
1260 LET #=1: THEN LET X=17: LET #
1270 LET #=1: THEN LET X=17: LET #
1280 LET #=1: THEN LET X=17: LET #
1290 LET #=1: THEN LET X=17: LET #
1300 LET #=1: THEN LET X=17: LET #
1310 LET #=1: THEN LET X=17: LET #
1320 LET #=1: THEN LET X=17: LET #
1330 LET #=1: THEN LET X=17: LET #
1340 LET #=1: THEN LET X=17: LET #
1350 LET #=1: THEN LET X=17: LET #
1360 LET #=1: THEN LET X=17: LET #
1370 LET #=1: THEN LET X=17: LET #
1380 LET #=1: THEN LET X=17: LET #
1390 LET #=1: THEN LET X=17: LET #
1400 LET #=1: THEN LET X=17: LET #
1410 LET #=1: THEN LET X=17: LET #
1420 LET #=1: THEN LET X=17: LET #
1430 LET #=1: THEN LET X=17: LET #
1440 LET #=1: THEN LET X=17: LET #
1450 LET #=1: THEN LET X=17: LET #
1460 LET #=1: THEN LET X=17: LET #
1470 LET #=1: THEN LET X=17: LET #
1480 LET #=1: THEN LET X=17: LET #
1490 LET #=1: THEN LET X=17: LET #
1500 LET #=1: THEN LET X=17: LET #
1510 LET #=1: THEN LET X=17: LET #
1520 LET #=1: THEN LET X=17: LET #
1530 LET #=1: THEN LET X=17: LET #
1540 LET #=1: THEN LET X=17: LET #
1550 LET #=1: THEN LET X=17: LET #
1560 LET #=1: THEN LET X=17: LET #
1570 LET #=1: THEN LET X=17: LET #
1580 LET #=1: THEN LET X=17: LET #
1590 LET #=1: THEN LET X=17: LET #
1600 LET #=1: THEN LET X=17: LET #
1610 LET #=1: THEN LET X=17: LET #
1620 LET #=1: THEN LET X=17: LET #
1630 LET #=1: THEN LET X=17: LET #
1640 LET #=1: THEN LET X=17: LET #
1650 LET #=1: THEN LET X=17: LET #
1660 LET #=1: THEN LET X=17: LET #
1670 LET #=1: THEN LET X=17: LET #
1680 LET #=1: THEN LET X=17: LET #
1690 LET #=1: THEN LET X=17: LET #
1700 LET #=1: THEN LET X=17: LET #
1710 LET #=1: THEN LET X=17: LET #
1720 LET #=1: THEN LET X=17: LET #
1730 LET #=1: THEN LET X=17: LET #
1740 LET #=1: THEN LET X=17: LET #
1750 LET #=1: THEN LET X=17: LET #
1760 LET #=1: THEN LET X=17: LET #
1770 LET #=1: THEN LET X=17: LET #
1780 LET #=1: THEN LET X=17: LET #
1790 LET #=1: THEN LET X=17: LET #
1800 LET #=1: THEN LET X=17: LET #
1810 LET #=1: THEN LET X=17: LET #
1820 LET #=1: THEN LET X=17: LET #
1830 LET #=1: THEN LET X=17: LET #
1840 LET #=1: THEN LET X=17: LET #
1850 LET #=1: THEN LET X=17: LET #
1860 LET #=1: THEN LET X=17: LET #
1870 LET #=1: THEN LET X=17: LET #
1880 LET #=1: THEN LET X=17: LET #
1890 LET #=1: THEN LET X=17: LET #
1900 LET #=1: THEN LET X=17: LET #
1910 LET #=1: THEN LET X=17: LET #
1920 LET #=1: THEN LET X=17: LET #
1930 LET #=1: THEN LET X=17: LET #
1940 LET #=1: THEN LET X=17: LET #
1950 LET #=1: THEN LET X=17: LET #
1960 LET #=1: THEN LET X=17: LET #
1970 LET #=1: THEN LET X=17: LET #
1980 LET #=1: THEN LET X=17: LET #
1990 LET #=1: THEN LET X=17: LET #
2000 LET #=1: THEN LET X=17: LET #

```

tándose según se sitúe nuestro muñeco.

De la línea 1400 a la línea 1499 se realiza la impresión de los puntos y el tiempo en pantalla.

De la línea 1500 a la línea 1540 se realiza la impresión de manzanas y lápidas en pantalla cada cierto tiempo.

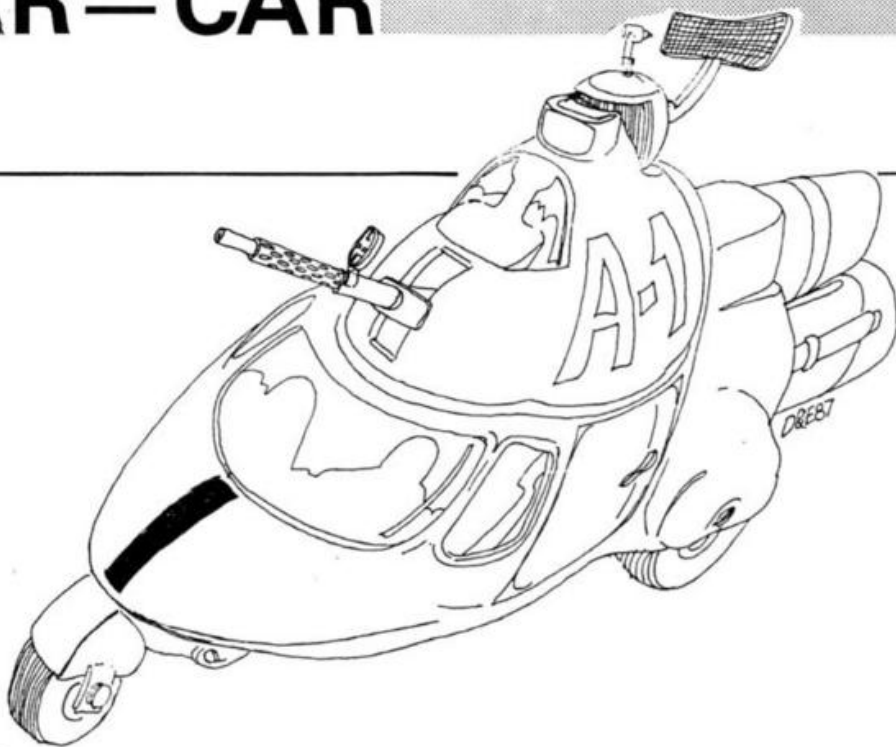
De la línea 1597 a la línea 1650 se comprueba cuando nuestro muñeco se come las manzanas.

De la línea 2500 a la línea 2533 se comprueban las colisiones con nuestro traga-manzanas.

De la línea 2600 a la línea 2750 se realiza el final del programa, y se da la opción de volver a jugar.

Dispones de dos pantallas de animación. SUERTE.

# LUNAR—CAR

[illegible]

Después de un viaje interesante por nuestra Vía Láctea has ido a aterrizar con tu nave nodriza en un planeta desconocido para el planeta tierra.

A bordo de tu LUNAR—CAR provisto de un potente laser, tendrás que hacer frente a robots y numerosos meteoritos que giran alrededor de la órbita del Planeta desconocido Guram.

Intenta acertar con tu laser a la primera sobre tu objetivo o quedarás a expensas del enemigo.

Para moverte por el planeta con el LUNAR — CAR dispones de las siguientes teclas: **Q.- SALTO, P.- DISPARO.**



[illegible]

### **Comentario del Programa.**

En la línea 50 junto con las DATAS de la 8000 a la 8070, están los datos de los UDGs del juego, los cuales no estarán introducidos en memoria hasta que no se produzca el primer RUN del programa.

En la línea 60 se inicializa la variable del Record.

En la línea 70 se colorea el  
border, papel y tinta.

Desde la línea 90 a la línea 140 se dibuja el paisaje sobre el cual se va a desarrollar el juego.

En la línea 150 hay un pequeño Poke cuyo objetivo es asegurar que el ordenador va a empezar el juego con el cursor en Mayúsculas.

Desde la línea 197 a la línea 250 se inicializan todas las variables del programa, las cua-

Continúa en pág. 21.



18 MUNDO SPECTRUM





# DON QUIJOTE DE LA MANCHA

**AVENTURA GRAFICO-CONVERSACIONAL DINAMIC**



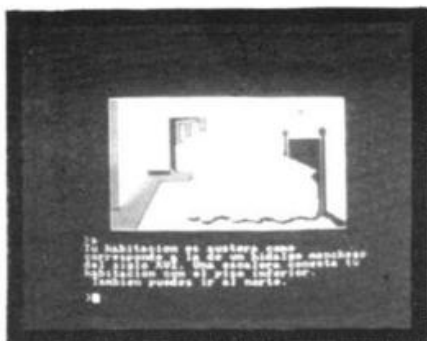
## INTRODUCCION:

La Mancha, España, abril de 1605...

Eres don Alonso Quijano, un hidalgo manchago del siglo XVI, cuya debilidad son las novelas de caballerías. Todo va más o menos bien por tu casa, hasta que un día tu locura alcanza el punto culminante: decides coger las armas y salir a correr aventuras por la Mancha. ¡Vaya ideas extrañas que tiene el hidalgo! Te consigues un escudero, Sancho Panza, que, aunque no es gran cosa, te ayu-

dará en los momentos difíciles. A partir de este momento entrarás en las más increíbles aventuras que te puedas imaginar. Ventas, colinas, bosques y posadas serán testigos de tus más locas ideas tales como atacar molinos o enfrentarte a fieros leones.

Tu objetivo final es conseguir los amores del Dulcinea del Toboso, tu dama. Aunque existe un primer objetivo, que es armarte caballero, para poder así comenzar tus desventuras.



## Comentario:

Don Quijote es un juego que pone a prueba tu ingenio y tu capacidad de observación, ya que en él tendrás que resolver muchas situaciones embarazosas, sin más ayuda que la de tu memoria y las pistas que el programa te va dando ocultas o, disfrazadas a veces, entre el texto o diálogo que el ordenador mantiene contigo.

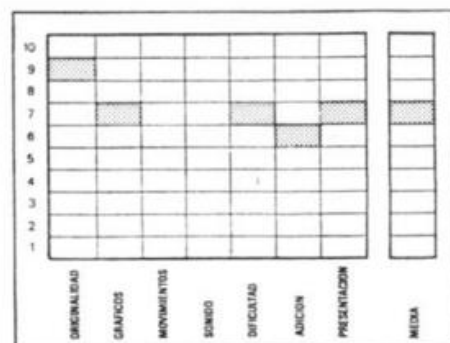
Don Quijote no es en absoluto un juego convencional y tampoco se

asemeja a las aventuras grafico-conversacionales clásicas, donde las respuestas del jugador debían ajustarse a un formato rígido y los gráficos eran pocos o de escasa calidad.

En Don Quijote por el contrario, te encontrarás con que el ordenador comprende frases ligeramente complicadas, y te sorprenderá la gran calidad de los dibujos que aparecen en cada situación. Sin embargo quizá se hechen en falta algunos sonidos ambientales, que hubieran dado mayor emoción al juego.

Como pantalla de presentación tenemos un estupendo dibujo, en el que aparecen Don Quijote y Sancho sobre sus respectivas cabalgaduras, en un paisaje típico manchego. Sin embargo el colorido no es excesivamente bueno.

Como ya hemos dicho, Don Quijote no puede ser comparado con ningún juego convencional ya que su temática es totalmente distinta, sin embargo podemos asegurar sin temor a equivocarnos que es la mejor aventura gráfico-conversacional que se ha realizado en lengua española.



# DUSTIN

## AVENTURA DINAMIC.

### ARGUMENTO:

Año 1989.

La cárcel de alta seguridad de WAD—RAS tiene un preso muy especial, se trata de KID SAGUF, más conocido como DUSTIN, el ladrón de joyas y obras de arte más famoso de su tiempo, que finalmente y tras largas persecuciones de la policía ha sido capturado.

Toda la preocupación de sus guardianes está volcada en envitar que DUSTIN pueda escapar.

Por esa razón todo el centro penitenciario ha sido rediseñado para que nada escape al control de su Director; sin embargo, KID SAGUF ha estado durante meses cavilando, analizando horarios, memorizando el funcionamiento interno del centro y ya tiene su plan para conseguir de nuevo las portadas de todos los pe-r-i-ódicos.



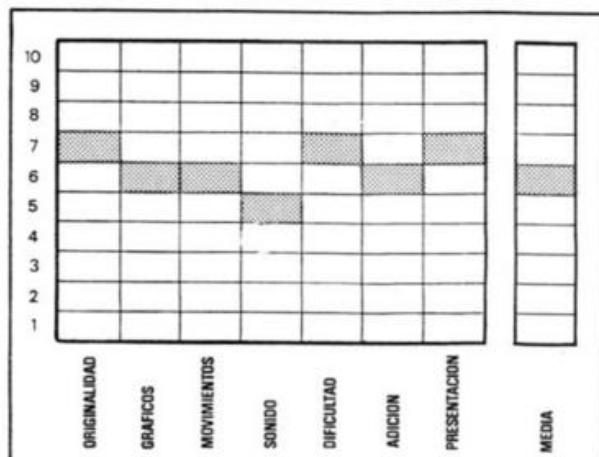
Para no olvidar ningún detalle, ha escrito su desarrollo completo en un cuaderno al que hemos tenido acceso y del que aquí no revelamos su información más interesante.

### COMENTARIO:

Dustin es un juego que trata un tema bastante original y tiene un alto grado de dificultad, dado el gran número de posibilidades o diferentes situaciones en las que puedes encontrarte, lo cual hace que una vez que has empezado a jugar te sea difícil retirarte de tu ordenador hasta ver si puedes conseguir avanzar un poco más en tu aventura.

El sistema de carga incorpora como ya viene siendo habitual en los programas de SPECTRUM un contador que efectúa una cuenta atrás de manera que podemos saber cuanto falta para que empiece el juego. La portada que acompaña la carga es sumamente atractiva y presenta una perspectiva tridimensional que la hace muy realista.

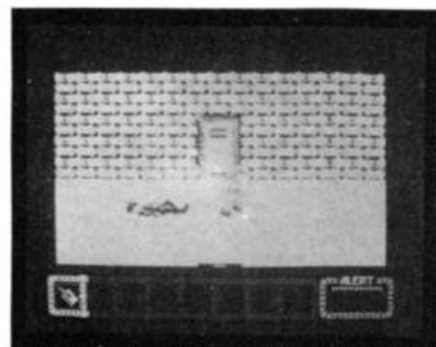
En las celdas y pasillos que constituyen los gráficos del paisaje, se ha realizado también un efecto tridimensional que les da cierta profundidad. Los personajes se mueven con rapidez y Dustin responde inmediatamente a las instrucciones del teclado. En la parte inferior de la pantalla aparecen, en forma de iconos, los diferentes objetos que vamos obteniendo a lo largo del juego, así como un pequeño marcador en la parte



derecha que nos indica el estado de alerta en que nos encontramos.

Los sonidos, si bien existe una bonita melodía durante la presentación, no han sido demasiado trabajados y únicamente nos indican cuándo golpeamos o somos golpeados en nuestros enfrentamientos con los guardias de la cárcel.

Dustin es un juego que como ya hemos dicho presenta un alto grado de dificultad, pero esta dificultad está en avanzar a lo largo de las diferentes etapas de la aventura y no en el manejo del protagonista, que es sumamente sencillo.





```

2520 IF MOUP=3 THEN PRINT INK 0;
AT 20,0;"ABCABCABCABCABCABC";AT 12,0;"DEFEFEFEFEFE"
2530 LET MOUP=MOUP+1: IF MOUP>3
THEN LET MOUP=1
2540 RETURN
2997 REM *****
2998 REM ** IMPRESSION ROBOT **
2999 REM *****
3000 PRINT AT 18,RXB;"*****";AT 19,
RXB;" "LET RXB=RX
3010 PRINT INK 1;AT 18,RX;"EQ";A
T 19,RX;"E3"
3020 LET RX=RX-1
3040 IF RX<0 THEN PRINT AT 18,RX
B;" ";AT 19,RXB;" ":LET ROBOT
=0
3050 RETURN
3497 REM *****
3498 REM *** CHOQUE DISPARO ***
3499 REM *****
3500 IF ATTR (DX,DY+2)=57 OR ATT
R (DX,DY+3)=57 OR ATTR (DX,DY+4)
=57 OR ATTR (DX,DY+5)=57 THEN GO
TO 3530
3505 IF ATTR (DX,DY+2)=58 OR ATT
(DX,DY+3)=58 OR ATTR (DX,DY+4)
=58 OR ATTR (DX,DY+5)=58 THEN GO
TO 3530
3510 RETURN
3520 LET DY=9; FOR A=1 TO 10: PR
INT AT DX,DY;" ";PRINT INK IN
(RND*5);AT 18,RXB;"UUU";AT 19,R
;"UUU": BEEP .004,A: LET DY=DY+
1
3530 IF DY>RX THEN LET DY=9
NEXT A
3540 PRINT AT 18,RXB;" ";AT 19,
" "LET R=R+20
3550 LET ROBOT=0: LET DISP=0
RETURN
3560 LET DY=9; FOR A=1 TO 10: PR
INT AT DX,DY;" ";PRINT INK IN
(RND*5);AT 18,RXB,ROYB;"UUU": BEE
P .004,A: LET DY=DY+3
3570 IF DY>ROY THEN LET DY=9
NEXT A
3580 PRINT AT ROXB,ROYB;" ";LE
T ROCA=0: LET DISP=0
RETURN
3590 REM *****
3600 REM ** IMPRESSION ROCA **
3610 REM *****
3620 PRINT AT ROXB,ROYB;"*****";LE
T ROY=ROY
3630 PRINT INK 3;AT ROX,ROY;"MN"
3640 AT ROY=ROY-1: IF ROY<0 THE
N AT ROXB,ROYB;" ":LET R
TURN
3650 RETURN
3660 M *****
3670 M *** CHOQUE CONTRA CAR ***
3680 M *****
3690 M ATTR (X,9)=57 OR ATTR (X
,8)=57 THEN GO TO 5000

```

les se corresponden de la siguiente manera: **X.-** movimiento del Car, **XB.-** borrado del Car, **C.-** contador general, **SALTO.-** 0 significa que no existe y 1 significa que sí esta saltando, **DISP.-** 0 no hay disparo y 1 se está produciendo el disparo, **DX y DY.-** coordenadas del disparo, **DB.-** borrado disparo, **MOVP.-** movimiento del paisaje, a 1 se mueve a 2 no se mueve, **ROBOT.-** 0 no existe y 1 si existe, **RX.-** coordenada movimiento robot, **RXB.-** borrado del robot, **P.-** puntos, **CONT.-** contador parcial, **ROCA.-** 0 no existe y 1 si existe, **ROX y ROY.-** coordenadas movimiento roca, **ROXB y ROYB.-** coordenadas de borrado de la roca.

De la línea 300 a la línea 330 se te dá una opción a una pausa antes de empezar el juego.

De la línea 997 a la línea 2000 se hace la impresión y movimiento del LUNAR-CAR, a la vez que es donde aparecen todas las opciones de disparo y salto de este. (Bucle General).

De la 2097 a la línea 2150 se realiza el bucle de disparo de nuestro CAR.

De la línea 4997 a la línea 5060 se produce el final del juego, en el cual se imprime en pantalla el clásico Game Over.





```

5 FOR A=0 TO 111: READ E: POK
E USR "B"+A,B: NEXT A
10 BORDER 0: PAPER 0: INK 7: C
15 PRINT INK 5; AT 1,0; " " M
20 PRINT INK 6; AT 5,0; " " M
JK JK JK JK JK JK JK JK JK JK
JK JK JK JK JK JK JK JK JK JK
JK JK JK JK JK JK JK JK JK JK
JK JK JK JK JK JK JK JK JK JK
30 PRINT INK 4; AT 12,0; " " JK JK
JK JK JK JK JK JK JK JK JK JK
JK JK JK JK JK JK JK JK JK JK
JK JK JK JK JK JK JK JK JK JK
JK JK JK JK JK JK JK JK JK JK
32 LET M=2
34 FOR A=0 TO 20
36 LET M=3-M: IF M=1 THEN PRINT
T INK 7; AT 19,A; "ABC"; AT 20,A; "D"
38 IF M=2 THEN PRINT INK 7; AT
19,A; " " AT 20,A; "GHI"
40 BEEP .01, .05, 10: BEEP .005, 12
41 IF INKEY$<>" " THEN GO TO 50
42 PRINT AT 19,A; " " AT 20,A
44 NEXT A
46 GO TO 34
50 BORDER 7: PAPER 7: INK 3: C
52 LET A=0
54 REM *****
56 REM ***** PAISAJE *****
100 FOR A=0 TO 20 STEP 2: PRINT
INK 1; AT A,0; "JKJKJK"; AT A+1,0;
"JKJKJK"
110 FOR A=0 TO 20 STEP 2: PRINT
INK 1; AT A,22; "JKJKJKJKJK"; AT A
+1,22; "JKJKJKJKJK": NEXT A

```

Para poder mover el Super-Muelle de un lado hacia otro, se han seleccionado las siguientes teclas en la consola de tu ordenador: **P.- derecha, Q.- izquierda.**

```

120 PRINT INK 1; AT 20,6; "JK.JK.JK
JK.JK.JK.JK"; AT 21,5; "JK.JK.JK
JK.JK.JK.JK"
130 PRINT INK 1; AT 0,6; "JK.JK
JK.JK"; AT 1,5; "JK.JK
JK.JK"
140 PRINT INK 3; AT 0,10; "LLLLL
LL"; AT 1,10; "LLLLLLL
150 PRINT PAPER 5; INK 0; AT 4,2
4; "PUNTOS"; AT 6,24; "INK 0; AT 4,2
4; "PAPER RECORD"; AT 12,24;
"ER 5; AT 10,24; "PAPER 2; INK 7; AT 10,2
4; "BOLA"; AT 10,24; "
190 REM *****
191 REM ***** VARIABLES *****
192 REM *****
2000 LET Y=10; LET YB=Y
210 LET A$="ABC"; LET B$="DEF"
220 LET Z=3; LET XI=1; LET ZI=1
230 LET X1=3; LET Z1=13; LET XB
1=X1; LET X1=3; LET Z1=13; LET XB
ZI=1; LET ZB1=Z1; LET XI1=1; LET
240 LET X2=3; LET Z2=13; LET XB
250 LET ZB2=Z2; LET XI2=1; LET
260 LET P=0; LET BO=1
270 LET T=0
280 REM *****
290 REM ***** SALIDA BOLA *****
300 REM *****
310 FOR A=17 TO 10 STEP -1: PRI
320 AT 0,15; BEEP .005,10; PRI
330 AT 1,15; BEEP .004,20; NEXT A
340 FOR A=17 TO 10 STEP -1: PRI
350 AT 0,15; BEEP .005,10; PRI
360 AT 1,15; BEEP .004,20; NEXT A
370 IF BO=1 THEN LET CO=2
380 IF BO=0 THEN LET CO=4
390 IF BO=0 THEN LET CO=3
400 BEEP .03,20; PRINT INK CO;
410 NEXT A
420 PRINT INK CO; AT 3,13; "M"
430 PRINT INK 1; AT 10,12; "TECLA
440 BEEP .003,INT (RAND*5); AT
450 IF INKEY$<>" THEN PRINT AT
460 A: GO TO 10; BEEP .05,A: N
470 PRINT INK 0; AT 10,12; "
480 GO TO 990
490 FOR A=10 TO 17: PRINT INK 3
500 AT 0,15; BEEP .005,10; BEEP
510 FOR A=10 TO 17: NEXT A
520 BEEP .004,20; PRINT INK 3
530 BEEP .05,10; NEXT A
540 REM ***** IMPRES. MUELLE *****
550 PRINT *****
560 IF A$<>"ABC" THEN PRINT INK
570 AT 10,Y; B$

```

### Comentario del Programa.

En la línea 50 junto con los DATA de las líneas 8000 a la línea 8060, están los datos de los UDGs del juego, que se se introducirán en memoria cuando des el primer RUN del programa.

En la línea 10 se pone de la pantalla después de realizar un CLS el color del border, papel y tinta según lo requieran las circunstancias.

Desde la línea 15 a la línea 48 aparece un For-Next, en el cual se dibuja una pantalla de presentación antes de la iniciación del juego. Dentro del For-Next, en la línea 41 se da la opción de salir de él al pulsar cualquier tecla del ordenador.

En la línea 50 procedente de un GOTO de la 41, se realiza un CLS de la pantalla del TV y se colorea según lo deseado.



```

1030 IF A$="ABC" THEN PRINT INK
0;AT 18,Y;A$;AT 19,Y;B$
1040 IF X<18 OR X1<18 THEN LET A
$="ABC": LET B$="DEF" THEN LET A
1050 IF INKEY$="P" OR INKEY$="P"
THEN LET Y=Y+1: IF Y>19 THEN LE
T Y=19
1060 IF INKEY$="O" OR INKEY$="Q"
THEN LET Y=Y-1: IF Y<6 THEN LET
Y=6
1100 GO SUB 2000
1110 PRINT PAPER 6; INK 0;AT 6,2
5;P
1120 PRINT PAPER 5; INK 0;AT 12,
25;R
1130 PRINT PAPER 2; INK 7;AT 18,
25;BO
1140 IF T>400 THEN GO TO 1010
1150 IF T=200 THEN FOR A=1 TO 20
: OUT 254,A: NEXT A: BORDER 7: L
ET BO=2: LET T=T+1: GO TO 910
1160 IF T=400 THEN FOR A=1 TO 20
: OUT 254,A: NEXT A: BORDER 7: L
ET BO=3: LET T=T+1: GO TO 910
1170 LET T=T+1
1200 GO TO 1010
1200 REM *****
1200 REM *** MOVIMIENTO BOLA 1 ***
1200 REM *****
1200 PRINT AT XB,ZB;": LET XB=
X: LET ZB=Z
2010 PRINT INK 2;AT X,Z;"M"
2015 IF BO>1 THEN GO SUB 3000
2020 LET X=X+XI: LET Z=Z+ZI
2030 IF Z<=6 THEN LET ZI=1: BEEP
.005,10
2035 IF Z>=21 THEN LET ZI=-1: BE
EP .005,10
2040 IF X<=2 THEN LET XI=1: BEEP
.005,10
2045 IF X=18 AND Z>Y-1 AND Z<Y+3
THEN GO TO 2100
2050 IF X>=20 THEN GO TO 2200
2097 REM *****
2098 REM *** CHOQUE BOLAS ***
2099 REM *****
2100 LET A$="": LET B$="GHI"
2110 BEEP .005,10: BEEP .05,5: B
EEP .005,20
2120 IF X>=17 THEN LET XI=-1
2125 IF X1>=17 THEN LET XI1=-1
2130 IF X2>=17 THEN LET XI2=-1
2140 LET P=P+5
2197 REM *****
2198 REM *** FINAL PROGRAMA ***
2199 REM *****
2200 IF X>=17 THEN PRINT AT XB,Z
B;"N": BEEP .5,5: BEEP .2,10: BE
EP .1,15: PRINT AT XB,ZB;"N"
2205 IF X1>=17 THEN PRINT AT XB1,
ZB1;"N": BEEP .5,5: BEEP .2,10:
BEEP .1,15: PRINT AT XB1,ZB1;"N"
2207 IF X2>=17 THEN PRINT AT XB2,
ZB2;"N": BEEP .5,5: BEEP .2,10:
BEEP .1,15: PRINT AT XB2,ZB2;"N"
2210 PRINT INK 7; PAPER 2;AT 8,1
2;"GAME":AT 10,12;"OVER"

```

En la línea 60 se inicializa la variable R correspondiente al record.

Desde la línea 90 a la línea 150 se dibuja el paisaje sobre el cual se va a desarrollar el juego, a la vez que se han impreso los puntos, el record, y el número de bolas que existen.

Desde la línea 190 a la línea 240 se inicializan todas las variables existentes en el juego o programa, correspondiéndose de la siguiente manera: Y.- movimiento lateral del muelle; YB.- borrado del muelle; A\$,B\$.- variables para impresión del muelle; X,Z.- variables x,y para movimiento de la primera bola; XB,ZB.- borrado de la primera bola; XI,ZI.- incremento o decremento de la primera bola; X1,Z1.- variables x,y para movimiento de la segunda bola; XB1,ZB1.- borrado de la segunda bola; X11,Z11.- incremento o decremento de la segunda bola; X2,Z2.- variables x,y para movimiento de la tercera bola; XB2,ZB2.- borrado de la tercera bola; X12,Z12.- incremento o decremento de la tercera bola; P.- puntos; BO.- contador de las bolas que hay en pantalla; T.- tiempo.

Desde la línea 900 a la línea 994 se produce la apertura de la puerta superior y salida de las bolas correspondientes al transcurso del juego.

Desde la línea 996 a la línea 998 se vuelve a cerrar la puerta superior.



```

0215 PRINT INK 0; AT 18,YB; "ABQ";
0220 IF YB="DEF" THEN GOTO 0225
0225 IF R<P THEN LET R=P
0230 IF INKEY$="5" OR INKEY$="3"
0240 IF INKEY$="N" OR INKEY$="D"
0250 GO TO 0230
0260 REM *****
0270 REM ** MOVIMIENTO BOLA 2 **
0280 PRINT AT XB1,ZB1; "": LET X
0290 IF B0>2 THEN GO SUB 03100
0300 LET X1=X1+XI1: LET Z1=Z1+ZI
0310 IF Z1<=5 THEN LET ZI1=1: BE
0320 IF Z1>=21 THEN LET ZI1=-1:
0330 IF X1<=2 THEN LET XI1=1: BE
0340 IF X1>=18 AND Z1>Y-1 AND Z1<
0350 THEN GO TO 02100
0360 RETURN
0370 REM *****
0380 REM ** MOVIMIENTO BOLA 3 **
0390 PRINT AT XB2,ZB2; "": LET X
0400 IF B2=X2+XI2: LET Z2=Z2+ZI
0410 IF Z2<=5 THEN LET ZI2=1: BE
0420 IF Z2>=21 THEN LET ZI2=-1:
0430 IF X2<=2 THEN LET XI2=1: BE
0440 IF X2>=18 AND Z2>Y-1 AND Z2<
0450 THEN GO TO 02100
0460 RETURN
0470 REM *****
0480 REM ** DATOS UDGS *****
0490 DATA 127,255,127,1,127,1,127,1
0500 DATA 127,255,127,1,127,1,127,1
0510 DATA 127,255,127,1,127,1,127,1
0520 DATA 127,255,127,1,127,1,127,1
0530 DATA 127,255,127,1,127,1,127,1
0540 DATA 127,255,127,1,127,1,127,1
0550 DATA 127,255,127,1,127,1,127,1
0560 DATA 127,255,127,1,127,1,127,1
0570 DATA 127,255,127,1,127,1,127,1
0580 DATA 127,255,127,1,127,1,127,1
0590 DATA 127,255,127,1,127,1,127,1
0600 DATA 127,255,127,1,127,1,127,1
0610 DATA 127,255,127,1,127,1,127,1
0620 DATA 127,255,127,1,127,1,127,1
0630 DATA 127,255,127,1,127,1,127,1
0640 DATA 127,255,127,1,127,1,127,1
0650 DATA 127,255,127,1,127,1,127,1
0660 DATA 127,255,127,1,127,1,127,1
0670 DATA 127,255,127,1,127,1,127,1
0680 DATA 127,255,127,1,127,1,127,1
0690 DATA 127,255,127,1,127,1,127,1
0700 DATA 127,255,127,1,127,1,127,1
0710 DATA 127,255,127,1,127,1,127,1
0720 DATA 127,255,127,1,127,1,127,1
0730 DATA 127,255,127,1,127,1,127,1
0740 DATA 127,255,127,1,127,1,127,1
0750 DATA 127,255,127,1,127,1,127,1
0760 DATA 127,255,127,1,127,1,127,1
0770 DATA 127,255,127,1,127,1,127,1
0780 DATA 127,255,127,1,127,1,127,1
0790 DATA 127,255,127,1,127,1,127,1
0800 DATA 127,255,127,1,127,1,127,1
0810 DATA 127,255,127,1,127,1,127,1
0820 DATA 127,255,127,1,127,1,127,1
0830 DATA 127,255,127,1,127,1,127,1
0840 DATA 127,255,127,1,127,1,127,1
0850 DATA 127,255,127,1,127,1,127,1
0860 DATA 127,255,127,1,127,1,127,1
0870 DATA 127,255,127,1,127,1,127,1
0880 DATA 127,255,127,1,127,1,127,1
0890 DATA 127,255,127,1,127,1,127,1
0900 DATA 127,255,127,1,127,1,127,1
0910 DATA 127,255,127,1,127,1,127,1
0920 DATA 127,255,127,1,127,1,127,1
0930 DATA 127,255,127,1,127,1,127,1
0940 DATA 127,255,127,1,127,1,127,1
0950 DATA 127,255,127,1,127,1,127,1
0960 DATA 127,255,127,1,127,1,127,1
0970 DATA 127,255,127,1,127,1,127,1
0980 DATA 127,255,127,1,127,1,127,1
0990 DATA 127,255,127,1,127,1,127,1
1000 DATA 127,255,127,1,127,1,127,1

```

Desde la línea 1000 a la línea 1200 aparece el bucle principal, donde se imprime el muelle y se te da la opción de poderlo mover hacia los lados. También se imprimen puntos y récord.

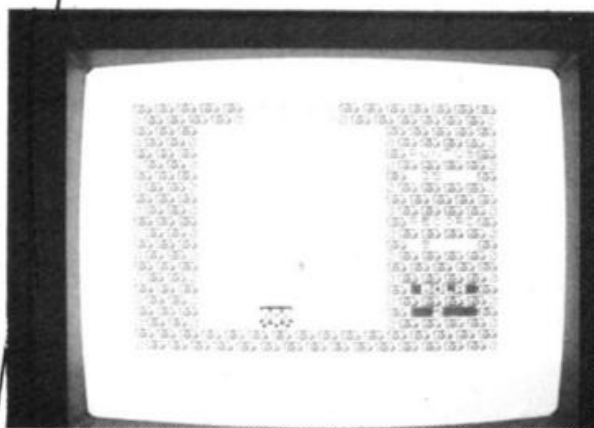
Desde la línea 1997 a la línea 2060 se produce la impresión y el movimiento de la primera bola.

Desde la línea 2097 a la línea 2140 se comprueban los choques de las bolas contra el suelo.

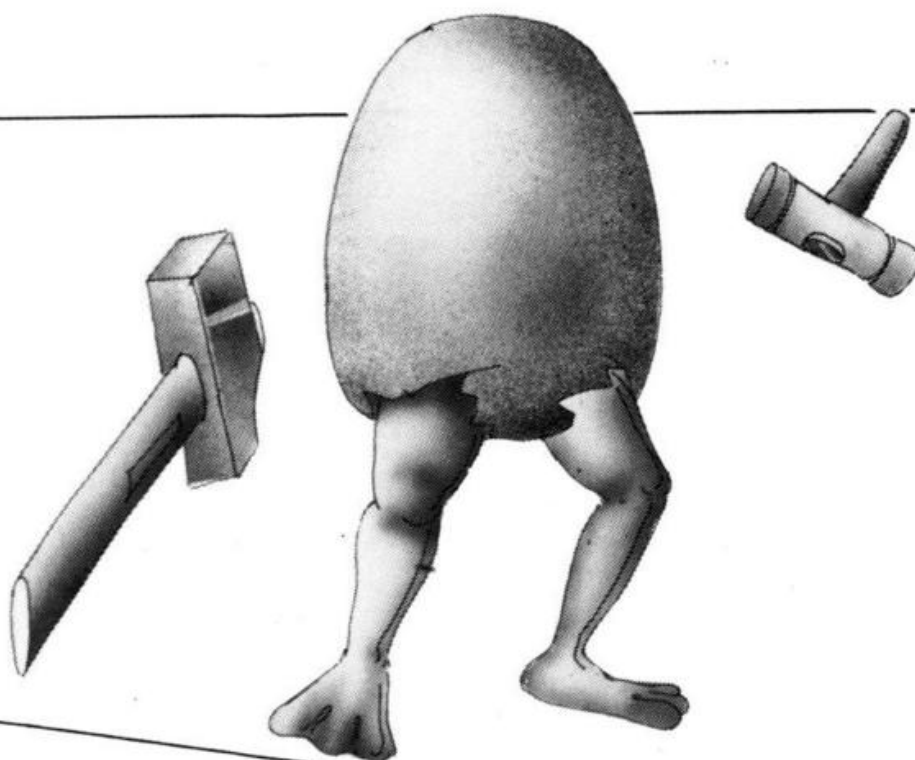
Desde la línea 2197 a la línea 2250 se realiza el final del programa donde explota la bola y se te imprime Game Over al no tener posibilidad de más vidas.

Desde la línea 2997 a la línea 3070 se realiza la impresión y el movimiento de la segunda bola.

Desde la línea 3097 a la línea 3180 se realiza la impresión y el movimiento de la tercera bola.



# MISTER- HUEVO



En este programa tan especial por su contenido, se te anima como a un pollo que al intentar salir del cascarón solamente consigue sacar las patas.

Sin poder ver nada a consecuencia de la cáscara del huevo andas de un lado hacia otro buscando a mamá gallina para que te ayude.

Pero no con mucha suerte, pues te has metido en una fábrica donde se les caen a los obreros martillos, que deberás esquivar para que no te dañen. Pero cuidado con las cintas transportadoras que te harán andar aunque tú no quieras.

Tendrás que moverte de una puerta a otra constantemente, con las siguientes teclas: **P.-derecha, Q.-izquierda.**

```

1 FOR b# = 0 TO 167: READ b: POK
E USR "a": b: NEXT b: INK 7: C
LS 3 BORDER 0: PAPER 0: INK 7: C
4 POKE 23858, 0
PRINT "AT 9, 0;"
PRINT INK 0; AT 9, 0;"
PRINT INK 0; AT 9, 0;"
PAUSE 10: PAUSE 0
LET t = 0
BORDER 0: PAPER 0: INK 7: C
ET 30 LET s# = "GH": LET b# = "KL": L
40 u = 1: LET u = 4: LET u1 = 1: LET
40 LET r = 0: LET c = 1: LET t = 5
50 LET m = 9: LET p = 1: LET p = 1
55 LET g = 12: LET h = 17: LET mb = m
60 LET h = h
e: LET f = 14: LET f = 23: LET e =
REM ***** IMPROV. DIAGRAMS *****
PRINT INK *****
PAPER *****
***** AT 0, 0

```

# MISTER HUEVO



```

1300 REM *****
1301 REM ** MOVIM. MARTILLOS **
1302 REM *****
1310 PRINT AT mb,nb;"*****"
1320 LET mb=m: LET nb=n
1330 PRINT INK 6; AT m,n;"U"
1340 LET m=m+1: IF m>18 THEN PRI
NT INK 6; AT mb,nb;"T": FOR a=1 T
AT mb,nb".005,a: NEXT a: PRINT
NT (RND*9): LET m=9: LET n=S+I
1400 PRINT AT eb,fb;" "
1410 LET eb=e: LET fb=f
1420 PRINT INK 3; AT e,f;"U"
1430 LET e=e+1: IF e>18 THEN PRI
NT INK 3; AT eb,fb;"T": FOR a=1 T
AT eb,fb".005,a: NEXT a: PRINT
NT (RND*9): LET e=9: LET f=17+
1500 PRINT AT gb,hb;" "
1510 LET gb=g: LET hb=h
1520 PRINT INK 4; AT g,h;"U"
1530 LET g=g+1: IF g>18 THEN PRI
NT INK 4; AT gb,hb;"T": FOR a=1 T
AT gb,hb".005,a: NEXT a: PRINT
NT (RND*20): LET g=11: LET h=6+
1600 IF g=17 OR e=17 OR m=17 THE
N GO SUB 3000
1700 PRINT INK 7; PAPER 1; AT 1,9
1710 LET t=t-1: IF t<0 THEN GO T
O 3500
1730 IF t=99 THEN PRINT PAPER 1;
AT 1,26;" "
1735 IF t=9 THEN PRINT PAPER 1;A
T 1,27;" "
2000 GO TO 1010
2007 REM *****
2008 REM ** CAMBIO PUERTAS **
2009 REM *****
2010 FOR e=1 TO 30: OUT 254,e: B
EEP .004,a: NEXT a: BORDER 0
FOR a=18 TO 18: THEN IF pa=1 THEN
NK 1; PAPER 5; AT a,0;" "
O; AT a,0;" "
a: LET pa=2: BEEP .009,2: NEXT
+10: LET re=re+1: RETURN
2120 IF pa=0 THEN IF pa=2 THEN
FOR a=18 TO 18: PRINT FLASH 1; I
NK 1; PAPER 5; AT a,0;" "
O; AT a,0;" "
a: LET pa=1: BEEP .009,2: NEXT
+10: LET re=re+1: LET pa=1: LET p=p
2150 RETURN
2200 REM *****
23001 REM ** CHOQUE MARTILLOS **
23002 REM *****
23010 IF m=17 THEN IF n>4-1 AND n
<4+2 THEN GO TO 3500
23020 IF e=17 THEN IF f>4-1 AND f
<4+2 THEN GO TO 3500
23030 IF g=17 THEN IF h>4-1 AND h
<4+2 THEN GO TO 3500
23040 RETURN
2400 REM *****
2401 REM ** FINAL PROGRAMA **
2402 REM *****

```

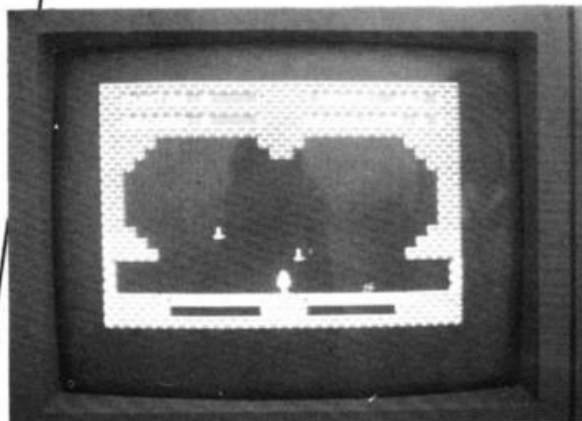
tro del programa y que darán movimiento a los distintos objetos. Estas vienen relacionadas de la siguiente manera: Las variables **a\$** y **b\$** se las iguala a los gráficos del muñeco, gracias a lo cual, cambiando en el juego éstas, se hará que parezca que el muñeco se mueve andando de un sitio a otro; **y**.- coordenada y del muñeco o huevo; **y1, y2, y3**.- se utilizan como contadores para realizar el movimiento del huevo; **re**.- número de recogidas o lo que es lo mismo las veces que se ha pasado de una puerta a otra; **co**.- contador para el movimiento de cintas transportadoras; **t**.- variable del tiempo; **p**.- variable de los puntos; **pa, pae**.- contadores generales para la caída de los martillos, **m, n**.- coordenadas x e y de un martillo para su movimiento; **mb, nb**.- coordenadas de borrado del primer martillo; **g, h**.- coordenadas de borrado del segundo martillo; **gh, hn**.- coordenadas de borrado del segundo movimiento; **e, f**.- coordenadas de borrado del tercer martillo.

Desde la línea 97 a la línea 160 se realiza la impresión de toda la pantalla incluyendo Puntos y Record, sobre la que se desarrollará el juego.

Desde la línea 1000 a la línea 2000 aparece el bucle central del programa, en el cual se realiza el movimiento de todos los

A B C D E F G H I J K  
L M N O P Q R S T U  
V W X Y Z

Desde la línea 3497 a la línea 3550 aparece el final del programa, donde se imprimirá en pantalla el clásico GAME OVER, y se dará la opción de volver a jugar otra vez o de dejar el juego.





Blanco y negro es una corta rutina en C/M que ocupa solamente 212 bytes, que transforma una pantalla en color en la equivalente en blanco y negro.

Esta rutina actúa sobre una pantalla normal en color, transformando estos colores a tonalidades de grises. La ventaja que nos proporciona, es que podemos hacer un COPY en impresora, de pantallas en color, sin obtener los "fallos" que se ocultan tras los atributos.

La rutina tiene dos entradas





# BLANCO Y NEGRO

10000  
 10001  
 10002  
 10003  
 10004  
 10005  
 10006  
 10007  
 10008  
 10009  
 10010  
 10011  
 10012  
 10013  
 10014  
 10015  
 10016  
 10017  
 10018  
 10019  
 10020  
 10021  
 10022  
 10023  
 10024  
 10025  
 10026  
 10027  
 10028  
 10029  
 10030  
 10031  
 10032  
 10033  
 10034  
 10035  
 10036  
 10037  
 10038  
 10039  
 10040  
 10041  
 10042  
 10043  
 10044  
 10045  
 10046  
 10047  
 10048  
 10049  
 10050

INVER  
 BU3  
 BU2  
 NORMAL  
 HL, 15384

posibles, una en la dirección a partir de la cual se ensambla, en nuestro caso 40.000, y que produce el efecto descrito, y la otra, tres posiciones de memoria más adelante que hace lo mismo, ofreciendo el resultado invertido, es decir los puntos que estaban de tinta quedan como puntos de papel y los de papel como puntos de tinta.

Si deseáis ver la rutina en acción, tendréis que introducir el listado 1 con un ensamblador, o si no disponéis de esta herramienta, os proporcionamos un cargador BASIC que realiza la misma operación en el listado 2.

—Pero... ¿Cómo funciona?

— La transformación se realiza carácter a carácter. Para ello se emplean tres bucles, el primero, es el que indica el tercio de la pantalla. El segundo, recorre los 256 caracteres de cada bloque (8 filas  $\times$  32 columnas). Es aquí donde se toma el atributo correspondiente de la pantalla, se separa el color del papel y el de la tinta, y se almacenan en PT1 y PT2 los punteros que se calculen de las mascararas de color. El tercer bucle es el que transforma cada fila de carácter (1 byte) empleando las más-



```

1060      LD      BC, 0144H
1070      LD      A, (HL)
1080      XOR     A, BC
1090      LD      (HL), A
1100      INC     HL
1110      DEC     BC
1120      LD      A, C
1130      OR      B
1140      JR      NZ, BU4
1150

```

**LISTADO 2 -Cargador-**

```

10 REM * CARGADOR C/M * COPY *
20 REM
30 CLEAR
40 LET S=0: FOR A=4E4 TO 40211
50 READ B: LET S=S+B: POKE A,B
60 NEXT A
70 IF S<>21819 THEN PRINT "ERR
80 STOP
90 REM
100 REM * GRABACION *
110 SAVE "COPY"CODE 40000,212
120 REM
130 REM * DEMOSTRACION *
140 REM
150 FOR A=0 TO 7: PRINT PAPER A
160 RANDOMIZE USR 40000
170 PAUSE 0
180 RANDOMIZE USR 40000
190 STOP

```

caras de color, en el byte resultado.

La operación realizada es la siguiente; del puntero de tinta (PT1) se extrae la máscara de tinta y se realiza un AND con el byte de pantalla. Luego hacemos lo mismo con el papel (para saber que puntos de papel son los coloreados, invertimos el byte). Una vez tenemos el nuevo aspecto de la tinta y el de papel, los unimos con OR para obtener el byte que sustituirá al original en la pantalla.

Tras completarse los tres bucles, se cambian todos los atributos de la pantalla al color: papel 7, tinta 0, brillo 0, flash 0.

Si se desean cambiar las tramas que definen cada color, habrá que modificar los datos contenidos en las líneas 80 a 270 del listado ensamblador. Cada 8 datos conforman la trama de cada color, como si de un U.D.G. se tratase. El orden en que se almacenan es del más oscuro al más claro, del Negro (0) al Blanco (7).

La única desventaja, es que al simular los colores mediante tramas de grises, pequeños detalles con poco contraste de color se pierden. Aún así esperamos que os sea de utilidad.



# LIBRERIA

## FICHEROS EN BASIC

Autor: C. Delanoy. Editorial: Paraninfo. Páginas: 173



Esta obra les resultará muy útil a todos aquellos usuarios que quieran aprender a crear sus propios ficheros, partiendo de unos conocimientos mínimos o que quieran profundizar en este interesante tema de la gestión.

En el libro se ofrecen una serie de explicaciones sobre fi-

chero manual e informático, y dentro de éste el acceso secuencial, directo e indexado.

En los Ficheros Secuenciales, se da una amplia información sobre cómo se crea un fichero directorio, profundizando en conceptos tales como grabación de información, la lectura, los modos separados, etc... Todo ello acompañado de una gran variedad de ejemplos de diversa complejidad.

En sucesivos capítulos se explican los ficheros de acceso directo, los indexados, así como el modo de crearlos y ventajas y desventajas con respecto al secuencial.

En todo caso, habrá que tener en cuenta que los programas demostrativos han sido escritos en un Basic muy general y que algunos por tanto habrán de ser adecuados al Basic del Spectrum.

## EL ORDENADOR EN LA EDUCACION BASICA

Autor: Tony Mullan. Editorial: Gustavo Gili. Páginas: 165



El libro está dirigido a padres y profesores que se encuentran en la necesidad de dar respuestas a problemas relacionados con los microordenadores en el ámbito de la Educación General Básica.

Así pues, se pretende dotar a los educadores de unos conocimientos mínimos con fines educativos, y sobre todo, mentalizarles en la necesidad

de que sus hijos y/o alumnos aprendan a programar.

La obra comienza con una introducción al ordenador, explicando someramente su historia, su rápido desarrollo y algunas de sus características más significativas, y continúa con unas explicaciones acerca de la importancia del ordenador en la actualidad, como herramienta indispensable de trabajo y estudio en las clases.

En otros capítulos trata en profundidad sobre el software, sobre su importancia a nivel educacional, sobre los modos de utilización de los programas, afin de sacarle el mayor rendimiento posible, siempre desde un punto de vista educacional, y sobre cómo han de ser planteados los ejercicios a los alumnos para ayudarles a introducirse de forma coherente y grata en el mundo del ordenador.

Mundo Spectrum



## BOLETIN DE SUSCRIPCION TODO UN AÑO DE PROGRAMAS E INFORMACION POR SOLO **2000** Pts.

### SUSCRIBETE AHORA Y OBTENDRAS

Ahorro de 400 Pts. SOBRE EL PRECIO DE VENTA EN TU KIOSCO un equivalente a 12 revistas por el precio de 10. Además tienes la seguridad de no perderte ningún número, aunque se agote.

Deseo que me envíen los doce próximos números del Mundo Spectrum por sólo 2.000 Pts. ahorrándome 400 Pts. sobre el precio de portada.

El primer número que deseo recibir es el .....

NOMBRE ..... APELLIDOS .....

DOMICILIO ..... CODIGO POSTAL .....

POBLACION ..... PROVINCIA ..... TELEFONO .....

#### FORMA DE PAGO:

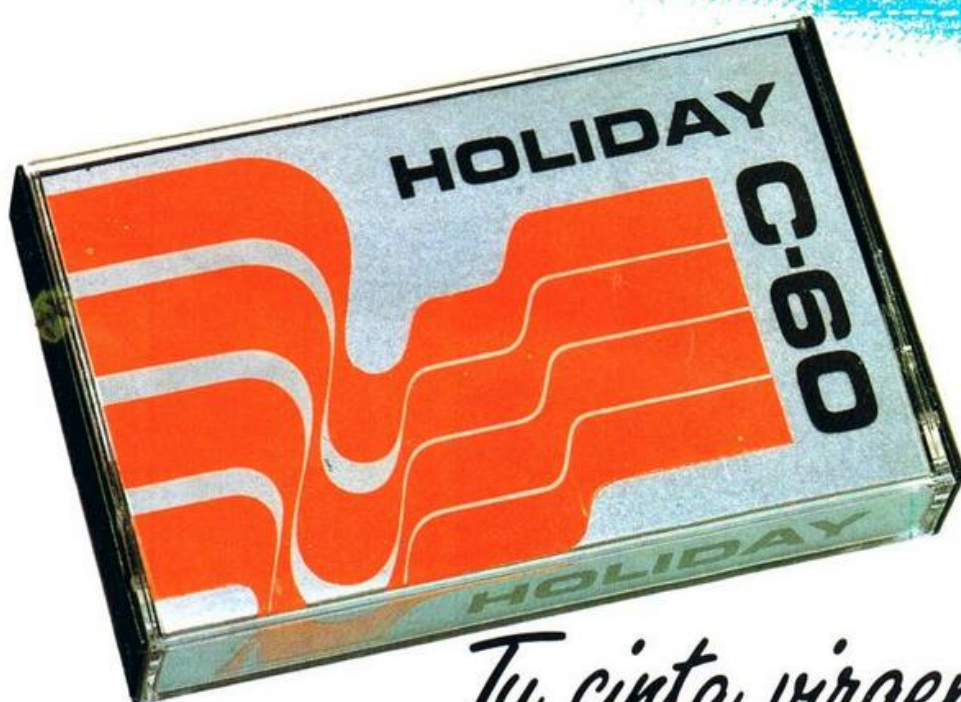
- ☐ Contra reembolso al recibir el primer número.
- ☐ Giro postal n°.....
- ☐ Cheque adjunto n°.....

FOTOCOPIA, COPIA O RECORTA ESTE CUPON Y ENVIALO A GENESIS PUBLICACIONES, Tomás López, 3 - 6° - 28009 MADRID



*¡Tu cinta para grabar guay!*

# HOLIDAY



*Tu cinta virgen  
de 40, 60 y 90*



Fabricada por IBEROFON, S.A.

Avenida de Fuentemar, 35 - Polígono Industrial de Coslada - MADRID

Teléfonos 671 22 00-04-08-12 - Télex 42797 FONO E - Telefax (91) 671 39 09



Direct Metal  
**dmm**  
Mastering

COMPACT  
**disc**  
DIGITAL AUDIO

**discoflex**<sup>®</sup>



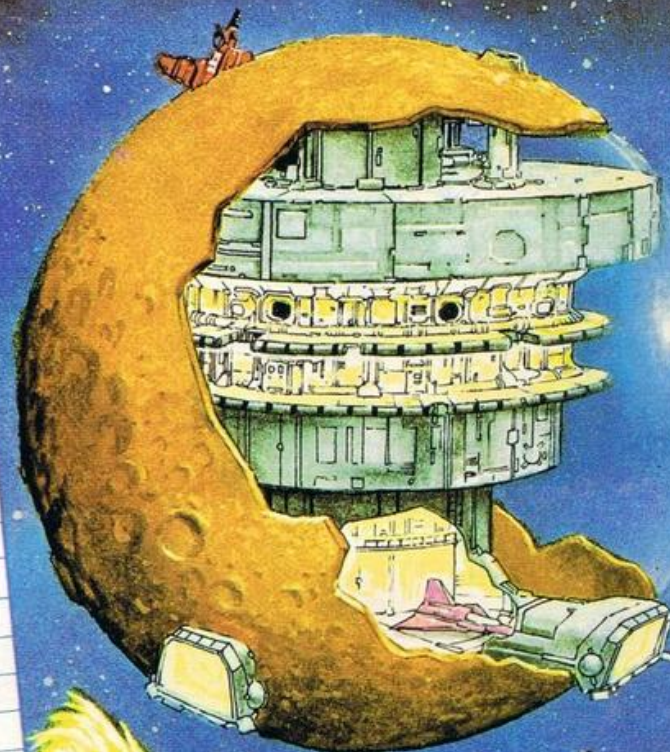
# FREDDY HARDEST

## FICHA TECNICA

- Dominio de las artes marciales: Patada - Puñetazo - Protección.
- Turbo-Láser de carga atmosférica.
- Salto controlable en altura y dirección.
- 30 Pantallas de Multiscroll.
- 64 Pantallas conectadas por ascensores y túneles.
- 16 Terminales de ordenador para hacer de Hacker.
- Enemigos inteligentes que se protegen de tu Láser.
- 5 Especies de animales hostiles.
- 3 Tipos de Androides asesinos.

## FX DOBLE CARGA

Significa dos programas en uno, el doble de acción, el doble de aventura, el doble de gráficos y máxima adictividad. Prueba un FX DOBLE CARGA de DINAMIC, te sorprenderá.



**FREDDY HARDEST**, agente secreto y refutado playboy, se encuentra en un satélite enemigo con su nave averiada. Para escapar, deberá llegar hasta la base enemiga y robar un cozo.

**FREDDY** demostrará su destreza saltando, agarrándose a las argollas que encuentre, trepando por cuerdas, disparando su láser.

**FREDDY** es todo un número uno, pero sobre todo, quiere escapar vivo.



SPECTRUM



COMMODORE



AMSTRAD CPC



MSX

## DINAMIC

DINAMIC SOFTWARE. PZA. DE ESPAÑA, 18. TORRE DE MADRID, 29-1. 28008 MADRID. TELEX: 44124 DSOFT-E

TIENDAS Y DISTRIBUIDORES: (91) 314 • 18 • 04. PEDIDOS CONTRA REEMBOLSO: (91) 248 • 78 • 87.

875