

## EL DISCO DEL PLUS 3

=====

*El empleo de una unidad de disco no resulta muy difícil, en relación con las posibilidades que reporta; sin embargo, como todo, requiere un aprendizaje. Vamos a hacer un estudio en profundidad del sistema de disco del Plus 3 para que, en lo sucesivo, nadie pueda decirnos que no sabe qué hacer con él.*

=====

Jesús Alonso Rodríguez

La aparición en el mercado del Spectrum Plus 3 viene a fijar definitivamente un standard en cuanto al formato de disco para Spectrum. Es evidente que la elección del formato de 3 pulgadas -no solo en el Spectrum, sino en todos los ordenadores de la gama Amstrad, excepto los compatibles PC por motivos obvios- obedece más a condicionantes de tipo económico que a la búsqueda de un sistema de almacenamiento masivo de gran calidad.

Las unidades de disco de 3 pulgadas pasan por ser las de menor calidad (exceptuando casos realmente folclóricos como el "Tritón"); especialmente las empleadas por Amstrad con una sola cara y capacidad para 40 pistas exclusivamente (180 Ks sin formatear). No obstante, no nos engañemos; el Spectrum es, probablemente, el ordenador más barato del mercado; dotarlo de una unidad de discos de 3 pulgadas y media con doble cara y 80 pistas trabajando en doble densidad (720 Ks sin formatear), hubiera supuesto un incremento de precio superior a las 10,000 pesetas, que significaría la pérdida de competitividad en su segmento del mercado.

A pesar de todo, el sistema de disco del Plus 3 resulta suficientemente rápido y fiable para las aplicaciones a que está destinado. Cae fuera de toda duda que supera con creces las características del absurdo "Microdrive" de Sir Clive (más folclórico aún que el "Tritón") y, por supuesto, no tiene nada que ver con el empleo de un cassette-audio como sistema de almacenamiento masivo. Tal vez su principal defecto sea la baja capacidad de almacenamiento, unido al hecho insólito de que se requiera dar la vuelta físicamente al disquete para acceder a la segunda cara. Indudablemente, le cabe a Amstrad el mérito de haber encontrado un sistema de almacenamiento masivo de prestaciones aceptables, a un precio que permite incorporarlo incluso en los ordenadores más baratos. Cosa de la que Sinclair no fue capaz y que explica, en gran parte, el fracaso de la "Sinclair Research".

Desde el punto de vista del usuario de Spectrum, acostumbrado a trabajar con cassette -ó a lo sumo, con Microdrive-, el Plus 3 abre todo un espectro de posibilidades en cuanto a programación; y no solo en lo que se refiere a las denominadas "aplicaciones serias". Sin embargo, para muchos supondrá su primer contacto con un Sistema Operativo de Disco y tal vez cree cierta confusión a los no iniciados.

En este artículo, vamos a hacer un estudio en profundidad del disco del Plus 3 y, sobre todo, de su Sistema Operativo. En primer lugar, empezaremos por aclarar algunos conceptos generales relativos a los sistemas de almacenamiento en disco.

### Sistemas de disco

Casi cualquier usuario tiene bastante claro cómo se almacenan los bytes en un cassette. Los bits van uno detrás de otro y se graban como tonos de baja frecuencia en la cinta, de una manera puramente secuencial. Si salvamos un bloque de 40 Ks, se graba todo de una vez, y no es posible leer una parte aislada del mismo sin leerlo todo.

En un sistema de disco, sin embargo, la información se fracciona en bloques de longitud uniforme (normalmente, 256 ó 512 bytes por bloque) y cada bloque no se almacena, necesariamente, a continuación del que le precede. El hecho de hacerlo así nos va a permitir acceder a cualquier parte de la información sin tener que leer todo lo que precede; esta posibilidad se denomina "acceso aleatorio" y, por ello, se dice que el disco es un sistema de almacenamiento masivo de acceso aleatorio.

Existen, básicamente, dos sistemas de disco empleados en micro-ordenadores: el disco duro ó "Winchester" y el disco flexible ó "Floppy". Evidentemente, la diferencia relevante entre ambos no estriba en que el disco sea duro ó flexible, sino en que sea fijo ó intercambiable. Un Winchester suele ser un disco de gran capacidad -de 10 a 60 Mega-bytes- (un Mega-byte son 1024 Ks) incorporado en el ordenador y que no se puede cambiar; mientras que un Floppy es un disco de mucha menos capacidad -de 0.1 a 1 Mega-byte- pero que puede ser extraído de la disquetera (disk drive) y reemplazado por otro (al igual que una cassette puede ser extraída del magnetófono e intercambiada por otra). De esta forma, el usuario dispone de una capacidad de almacenamiento ilimitada (puede tener tantos disquetes como quiera) pero solo una pequeña parte es accesible en cada momento.

Esto nos lleva al concepto de almacenamiento "en-línea" (on-line) que es la capacidad de almacenamiento accesible al sistema en un momento dado, es decir, sin cambiar de disquete. En la configuración básica del Plus 3, la capacidad de almacenamiento "on-line" es de 244 Ks; 180 Ks en el disco flexible y 64 Ks en el disco virtual ó RAM-Disk. Si se añade una segunda unidad, por ejemplo de 80 pistas, doble cara, doble densidad, la capacidad "on-line" subiría a 964 Ks (720 Ks de la segunda unidad).

En realidad, estas capacidades son menores, puesto que antes de utilizar el disco, es necesario formatearlo, operación que lleva aparejada la escritura de ciertos datos en el disco (denominados "estructuras de volumen") que ocupan cierto espacio. Una vez formateado, la capacidad de un disco de 3" del Plus 3 es de 173 Ks, mientras que la del RAM-Disk es de 58 Ks (evidentemente, el RAM-Disk se formatea automáticamente al arrancar el sistema). En el Plus 3, cada floppy vale por dos, puesto que si le damos la vuelta al disco, dispondremos de otros 173 Ks de almacenamiento. Pero, ¿cómo se almacena realmente la información en un disco?.

### Pistas y sectores

Podemos imaginar un disco cómo si estuviera dividido en circunferencias concéntricas -en realidad son coronas circulares de muy poca anchura-. El disco gira a una velocidad de 300 revoluciones por minuto y es posible desplazar la cabeza de forma radial, es decir, siguiendo el radio del disco desde fuera

hacia dentro y viceversa. El movimiento de la cabeza está gobernado por un motor paso-a-paso. A cada impulso, la cabeza se desplaza una fracción del radio disponible del disco. El número de posiciones intermedias que puede adoptar la cabeza nos determina el número de circunferencias concéntricas que tendrá el disco. Cada una de estas circunferencias concéntricas se denomina "pista". En el Plus 3 disponemos de 40 pistas, numeradas desde 0 hasta 39 de forma consecutiva, siendo la "0" la más exterior. Evidentemente, el desarrollo (la longitud) de cada pista es diferente; la pista 0 es casi el doble de larga que la pista 39; por tanto, y dado que todas las pistas almacenan el mismo número de datos, éstos estarán más "apretados" en las pistas interiores que en las exteriores (técnicamente, las pistas interiores tienen mayor densidad). Algunas unidades de disco compensan este problema grabando las pistas interiores con menos intensidad que las exteriores.

El Plus 3 (y casi todos los sistemas de disco) vá llenando el disco de fuera a dentro. Es bueno saber esto porque los últimos ficheros que metamos en un disco se grabarán en las pistas más interiores y serán, por tanto, los más susceptibles de producir errores. Hay que tener en cuenta que, si se han borrado datos, los que se graben a continuación ocuparán los lugares que dejaron libres los que se borraron. Como el Plus 3 lista los ficheros en orden alfabético, no es posible saber qué ficheros ocupan las pistas más interiores, si se ha escrito y borrado mucho en un disco.

Cada pista está dividida en una serie de tramos denominados "sectores". Cada uno de ellos almacena un número fijo de bytes. Cuando hay que almacenar un bloque de bytes (fichero) en un disco, se fracciona en pequeños bloques cada uno de los cuales cabe en un sector. Cada sector hay que grabarlo (ó leerlo) de una vez y un mismo sector no puede contener datos correspondientes a más de un fichero; por tanto, si al final del fichero queda un bloque de menos bytes que los necesarios para llenar un sector, el espacio sobrante de ese sector queda vacío.

En el Plus 3, cada pista se divide en 9 sectores numerados del 1 al 9. Cada uno de ellos puede almacenar 512 bytes. Dado que el disco gira de forma continua, de alguna manera hay que saber donde empieza el sector 1. Para ello, existe un pequeño orificio cerca del centro del disco que se corresponde con el inicio del sector 1. A este orificio se le denomina: "índice".

En realidad, en un sector caben más de 512 bytes pero, además de los datos del usuario, se almacenan algunos más que necesita el sistema. Los principales son: el número de sector, el indicador de si está libre u ocupado, y el CRC de los datos que contiene.

El CRC ("Cyclic Redundancy Check" ó Comprobación de Redundancia Cíclica) es un dato que se obtiene operando entre sí todos los restantes datos que contiene el sector, y sirve para comprobar su integridad. Su función es similar a la del último byte de un bloque grabado en cassette (byte de paridad).

La operación de formatear un disco consiste en marcar todos los sectores con su número, comprobar que no hay errores (escribiendo un dato en cada uno de ellos y volviéndolo a leer), inicializar su contenido y marcarlos como sectores vacíos; asimismo, se reserva un area en las pistas más exteriores del disco, para contener el directorio del mismo; es decir, el nombre de cada fichero almacenado y las pistas y sectores que

ocupa.

Afortunadamente, no es necesario que el usuario se preocupe, cada vez que tenga que guardar un fichero, de fragmentarlo en bloques, buscar sectores vacíos en el disco, grabar cada bloque en un sector y apuntar en qué sector está cada uno de ellos; ni de buscar y reunir sus trozos cuando, luego, desee cargarlo. Para llevar a cabo estas tareas existe un conjunto de rutinas escritas en código máquina y grabadas en una de las ROMs del ordenador. Este conjunto de rutinas conforman un programa que gestiona el disco de forma transparente para el usuario y que se denomina: "Sistema Operativo de Disco" (se suele abreviar cómo "DOS", iniciales del Inglés: "Disk Operating System").

#### Sistema Operativo de Disco

Cuando apareció el primer micro-procesador integrado en una sola pastilla y con prestaciones suficientes para utilizarlo cómo CPU de un ordenador (el Z-80 de Zilog), los fabricantes de sistemas informáticos se lanzaron a fabricar ordenadores de precio y tamaño relativamente pequeños que resultaran asequibles a las pequeñas empresas que no podían permitirse el lujo de contar con un gran ordenador. Cada fabricante incorporaba en sus máquinas, su propio Sistema Operativo, con lo que los programas escritos para un ordenador no podían correr en otro.

Pronto se hizo evidente la necesidad de desarrollar un Sistema Operativo que, con independencia de la máquina donde corriera, asegurara la compatibilidad entre ordenadores de distintos fabricantes. Así surgió el primer Sistema Operativo standard al que se denominó: "CP/M" y estaba escrito para el micro-procesador Z-80.

Posteriormente, fueron apareciendo micro-procesadores más potentes y, con ellos, los sistemas operativos correspondientes. Son muy conocidos el MS-DOS (verdadero standard debido a que fué adoptado por el gigante IBM para su gama de PCs), UNIX, System/2 (el nuevo de IBM), etc.

Con la aparición de micro-procesadores de 16 y 32 bits capaces de trabajar a velocidades de hasta 20 Megahercios, las máquinas basadas en Z-80, que funcionaban con CP/M, fueron quedando relegadas de las aplicaciones profesionales. No obstante, los fabricantes de ordenadores pronto encontraron un mercado sustitutorio y con unas potencialidades de expansión impresionantes: el del "ordenador doméstico". Volvieron a aparecer máquinas basadas en Z-80 (Spectrum, Amstrad CPC y PCW, etc.) a precios realmente irrisorios y, cuando estas máquinas fueron dotadas de disco, se rescató el CP/M, ya que permitía correr magníficos programas ya escritos para otros ordenadores (Multiplan, DBase II, etc.).

Cuando Amstrad estaba preparando el Plus 3, se especuló con la posibilidad de que soportara un Sistema Operativo CP/M. Lo cierto es que éste sistema requiere, entre otras cosas, 80 columnas en pantalla, por lo que no se pudo incorporar en la versión definitiva del Plus 3. En su lugar se incorporó un Sistema Operativo desarrollado por Amstrad y al que se denominó: "+3DOS". La diferencia, que puede parecer una cuestión de matices, es muy importante ya que cierra, al usuario del Plus 3, la puerta al magnífico software escrito para CP/M. No perdemos la esperanza de que algún fabricante de software se decida a lanzar un CP/M para Spectrum; entre tanto, tendremos que

conformarnos con nuestro +3DOS.

A pesar de todo, el +3DOS está fuertemente inspirado en el CP/M, la designación de ficheros es, prácticamente, igual y la mayoría de las rutinas son las correspondientes al CP/M con ligeras modificaciones. Las diferencias de sintaxis se deben, principalmente, a que el Spectrum utiliza el Sistema Operativo de disco desde Basic y, por tanto, está sujeto a las restricciones sintácticas propias del lenguaje. Por ejemplo: Si estamos utilizando la unidad de disco "A" y queremos cambiar a la "B", en CP/M teclearíamos:

B:

mientras que en el Plus 3 es necesario teclear:

SAVE "B:"

(ó LOAD "B:"). El comando del Sistema Operativo de Disco tiene que ir, siempre, precedido por un comando de Basic; lo contrario provocaría un error de sintaxis. Veremos que los programadores de Amstrad se las han arreglado de manera muy ingeniosa, para emular los comandos de un Sistema Operativo de Disco, utilizando los comandos de Basic ya disponibles y sin añadir ninguno. Es especialmente significativa la orden que se utiliza para convertir un fichero al formato del Spectrum:

COPY TO SPECTRUM FORMAT

Todo lo que se almacena en el disco es un fichero; tanto si se trata de un programa Basic, de un bloque de bytes ó de una matriz de datos, es tratado de la misma forma por el Sistema Operativo de Disco. La única diferencia frente a un DOS standard es que se le añade una cabecera para que, cuando se cargue, el Basic sepa qué hacer con él; si es un programa Basic, hay que separar programa y variables así cómo comprobar si tiene auto-ejecución; si es un bloque de bytes, hay que saber en qué dirección cargarlo, etc.

Cada fichero es identificado por un "nombre de fichero" que consta de varias partes. Antes de estudiar cada una de ellas, y para comprender mejor su función, es necesario pasar revista a algunos conceptos comunes a cualquier Sistema Operativo de Disco.

#### Volúmenes y Directorios

Un disquete introducido en una unidad de disco es un "volumen". En el momento en que introducimos un disquete en la unidad "A" estamos asociando esa unidad a un determinado volumen. Nos referiremos a ella cómo unidad "A", pero, en realidad almacenaremos los datos en el volumen introducido en ella. Es importante no confundir la unidad con el volumen, ya que el +3DOS nos permite asociar dos volúmenes a dos unidades ("A" y "B") a través de una sola disquetera. Si el disquete introducido corresponde a un volumen asociado a la unidad "A" y hacemos una referencia a la "B", el +3DOS nos pedirá que cambiemos el disco; si luego volvemos a referirnos al "A", de nuevo nos pedirá el cambio. Esto, evidentemente, solo es válido cuando se trabaja con una sola disquetera; si tenemos dos, la interna será la unidad "A" y la externa la "B". Como

particularidad del Plus 3, cada cara de un mismo disco, es un volumen diferente.

Dentro de cada volumen, podemos realizar particiones independientes, cada una de las cuales contendrá ciertos archivos. Cada una de estas particiones se denomina "directorio". Si no se crea ninguna, el Sistema Operativo asume que todo el disco es un solo directorio. Cuando hagamos un catálogo para ver los archivos de un determinado directorio, no veremos los de los demás.

En cuanto a los volúmenes, directorios y nombres de fichero, el +3DOS funciona exactamente igual que el CP/M. No existe nombre de volumen, por lo que toda referencia a él se hará mediante el nombre de la unidad a la que está asociado, seguido de dos puntos ("A:", "B:" ó "M:").

La gestión de directorios es automática. Cada directorio viene dado por un número de usuario que precede al nombre de unidad (el manual llama a los directorios "areas de usuario", pero no es una denominación correcta). Este número puede estar comprendido entre 0 y 15. Mientras no se especifique ningún número de usuario, el Sistema asume que está en el directorio correspondiente al usuario "0". En el momento en que se especifique un número de usuario distinto de "0", se crea el directorio y se asume como directorio en curso para futuras operaciones. Lo cierto es que los números de usuario resultan de escasa utilidad en un sistema de disco que solo admite 64 ficheros en 173 Ks; su utilidad principal es para trabajar en sistemas multi-usuario (varios terminales) con discos de gran capacidad. En el +3DOS se ha respetado esta posibilidad en aras de mantener la mayor compatibilidad posible con CP/M.

A continuación del número de usuario y del nombre de unidad, debe ir el nombre de fichero. Este consta de dos partes separadas por un punto. La primera parte tiene un ancho máximo de 8 caracteres y constituye el nombre del fichero propiamente dicho. La segunda parte tiene un ancho de tres caracteres y se denomina "extensión"; suele emplearse para hacer referencia al tipo de fichero. Por ejemplo, se puede tomar cómo norma utilizar la extensión ".Bas" para todos los ficheros que contengan programas en Basic. Son frecuentes las extensiones:

- .Bas = Programa Basic.
- .Asm = Programa fuente en Assembler.
- .Obj = Código Objeto.
- .Bin = Fichero de datos en binario.
- .Txt = Fichero de texto.
- .Sys = Fichero de Sistema.
- .Exe (ó .Run) = Fichero ejecutable.
- .Cob = Programa fuente en Cobol.
- .Dat = Fichero de datos (p.ej: una matriz).
- .Pan = Fichero conteniendo una pantalla.
- .Bak = Fichero de BackUp (creado por el Sistema).
- .Ndx (ó .Ind) = Fichero de indice.
- .Mp = Fichero generado con Multiplan.

Algunos de ellos no tienen sentido en el Plus 3. (p.ej: .Sys, .Obj, .Cob, etc.). Tanto en el nombre de la unidad, cómo en el del fichero, los caracteres pueden ir en minúsculas ó en mayúsculas, pero el Sistema no hace distinción entre ellas. Los nombres: "Pepe.Dat" y "PEPE.DAT" son considerados idénticos. Es

frecuente utilizar un carácter en mayúsculas para separar dos palabras dentro de un nombre de fichero, ya que no se admiten espacios; un ejemplo podría ser: "TemaUno.Txt", "TemaDos.Txt", etc. No es necesario emplear las extensiones indicadas aquí, pero se trata de abreviaturas universalmente aceptadas. Incluso, algunos programas añaden, de forma automática, ciertas extensiones a los ficheros que crean. Veamos un fichero especificado por completo:

"7B:SuperPro.Bas"

Se trata de un fichero del volumen contenido en la unidad "B:", directorio correspondiente al usuario nº 7, cuyo nombre es "SuperPro" y que contiene un programa en Basic. No es necesario especificar los datos de unidad y directorio si el fichero se vá a grabar, cargar, borrar, etc. de la unidad y directorio en curso; sin embargo, es necesario especificar la unidad siempre que se especifique el directorio. Cada vez que se especifique una unidad, se asume como unidad en curso para futuras operaciones hasta que sea especificada otra.

Una opción que hace muy cómodo el manejo de ficheros es el empleo de "comodines" (en Inglés, "Wild-Cards"). Un comodín es un carácter que sustituye a uno ó más caracteres del nombre del fichero y/o de su extensión. La operación que incluya una especificación con comodines, afectará a todos los ficheros cuyo nombre encaje con los caracteres que no sean comodines.

Los comodines empleados por +3DOS (y por casi todos los sistemas operativos de disco) son: asterisco ("\*") e interrogación ("?"). El asterisco sustituye a varios caracteres, mientras que la interrogación sustituye a uno solo. Por ejemplo: si queremos borrar todos los ficheros de datos que empiecen por "A", podemos hacer:

ERASE "A\*.Dat"

No importa si uno se llama "Ana.Dat", otro "Area.Dat" y un tercero "Accion.Dat"; los tres serán borrados. La orden no afectará, sin embargo, a un fichero que se llame "Ana.Bas" ni a otro que se llame "Pepe.Dat". Si tuviéramos varios ficheros con distinta extensión y el mismo nombre (p.ej: "Ana.Bas", "Ana.Bin" y "Ana.Dat") podemos borrarlos con la orden:

ERASE "Ana.\*"

Logicamente, la orden:

ERASE " \*.\*"

Borrará todos los ficheros contenidos en el directorio en curso, pero pedirá confirmación antes de hacerlo; sin embargo, el comando:

ERASE "A:"

Borra todos los ficheros del disco "A:" sin pedir confirmación; se trata de un comando realmente destructivo. Una particularidad del +3DOS es que el asterisco ha de ser el último carácter del nombre ó de la extensión; NO es posible borrar

todos los ficheros acabados en "A" con la orden:

ERASE "A.\*"

El carácter "?" se puede emplear para sustituir a un carácter del nombre. Supongamos que tenemos los ficheros: "Ana1Pro.Bas", "Ana2Pro.Bas" y "Ana3Pro.Bas"; podemos borrar los tres de una vez con:

ERASE "Ana?Pro.Bas"

Se pueden utilizar tantas interrogaciones como se desee y no es necesario que sea el último carácter. Evidentemente, no es lícito emplear estos caracteres cuando se salva (SAVE) un fichero; pero se pueden emplear para borrarlos (ERASE), copiarlos (COPY), obtener un catálogo (CAT), etc.

No lo hemos dicho, pero es evidente que no pueden existir dos ficheros con el mismo nombre y extensión dentro del mismo directorio (pueden existir en el mismo disco siempre que estén en distintos directorios). Entonces, ¿qué hace el Sistema cuando vá a salvar un fichero y ya existe otro con el mismo nombre?. Simplemente, al fichero que ya existe, le cambia la extensión para que sea ".Bak" y salva el nuevo fichero. ¿Y la tercera vez, cuando ya existe el ".Bak"? En ese caso, el ".Bak" antiguo es borrado. No está de más, limpiar de vez en cuando los discos más utilizados con:

ERASE " \*.Bak"

Protección de escritura

En la parte delantera de los discos, existe un pequeño orificio tapado con una corredea de plástico blanco. Si desplazamos la corredera hacia dentro, el disco queda protegido, su contenido no puede ser alterado (escrito, renombrado ni borrado) y solo es posible leerlo -se trata de algo similar a lo que ocurre al romper la lengüeta posterior de un cassette-. En los discos de tres pulgadas, existen dos de estos orificios, uno para cada cara. En los discos de 3 1/2", la protección es una ventana cuadrada en la parte posterior, el disco está protegido cuando la ventana está abierta. En los de 5 1/4", la protección consiste en una muesca en el lateral izquierdo; el disco se protege tapando la muesca con una pegatina opaca.

Atributos del fichero

Además del nombre, el tipo y el contenido, es posible fijar unos determinados atributos para los ficheros. El +3DOS utiliza los mismos atributos que el C/PM, con la diferencia de que solo uno de ellos tiene sentido en el Plus 3. Los atributos son:

- "p" = Protegido. El fichero no se puede borrar.
- "s" = Sistema. El fichero no aparece al hacer un CATalogo.
- "a" = Archivo. No sirve para nada en el Plus 3.

Para accctivar un atributo (p.ej: "p"), se tecllea:

MOVE "nombre.ext" TO "+p"

Y para desactivarlo:

```
MOVE "nombre.ext" TO "-p"
```

Un fichero puede tener más de un atributo activo, por ejemplo, supongamos que queremos hacer que el fichero "Pepe.Dat" no se pueda borrar ni aparezca en un CATalogo; deberemos teclear:

```
MOVE "Pepe.Dat" TO "+p"
MOVE "Pepe.Dat" TO "+s"
```

La protección por atributos no resulta muy eficaz como protección de software, ya que cualquiera puede hacer:

```
MOVE "*.*" TO "-p"
MOVE "*.*" TO "-s"
MOVE "*.*" TO "-a"
```

Además, los atributos no se copian cuando se copia un fichero. La nueva copia tendrá todos los atributos desactivados.

Auto-carga

Si, al arrancar el ordenador, se elige la opción: "Cargar" ("Loader"), el Sistema busca un fichero que se llame "\*" (evidentemente, tendrá que haber sido creado en código máquina ya que no es un nombre válido del +3DOS); si lo encuentra, lo carga y lo ejecuta. Si no lo encuentra, busca un fichero que se llame "Disk" y contenga un programa en Basic; si lo encuentra, lo carga (se supone que debe haber sido grabado con auto-ejecución -LINE-). Si no existe, ó si no hay disco en ninguna unidad, solicita que se ponga en marcha el cassette y se pulse cualquier tecla, para cargar un programa desde cinta.

Todos los sistemas operativos de disco previenen la posibilidad de hacer un "boot" de un programa u otro Sistema Operativo. Un "boot" (que se puede traducir como "calzar") consiste en cargar y ejecutar automáticamente un mini-programa (en código máquina) denominado "bootstrap" que deberá estar contenido en un determinado sector del disco (normalmente, el primero) y que, una vez cargado y arrancado, cargará otro programa mayor que puede ser un juego, un Sistema Operativo, etc.

En el +3DOS, el sector de "boot" debe ser el sector 1 de la pista 0 y cara 0 (en unidades de doble cara). El programa que contenga se cargará en la dirección FE00h (65024) y se ejecutará en FE10h (65040) con las interrupciones deshabilitadas y el puntero de pila colocado en FE00h (65024) -debe ser, por tanto, un programa "auto-contenido"- . La condición para que ésto ocurra es que la suma de todos los bytes del sector debe ser tal que, al dividirlo por 256, se obtenga un resto de 3. Para conseguirlo, se puede poner el código necesario en el byte 15 (décimo-sexto byte del sector).

Manejo del Cassette

Para mantener la compatibilidad con versiones anteriores de Spectrum, el +3DOS también permite manejar el cassette. Para ello, sirven los mismos comandos de disco (excepto ERASE, MOVE, COPY y FORMAT) pero dirigidos a la unidad "T:" (inicial de

"Tape", cinta en Inglés). También es posible teclear:

```
SAVE "T:"
```

(ó LOAD "T:") para que el cassette sea asumido como unidad en curso. A partir de este momento, el cassette se puede emplear cómo si de un Spectrum normal se tratase.

Una particularidad interesante es que, si se utiliza el comando CAT dirigido al cassette (unidad "T:") actúa cómo un lector de cabeceras, es decir, muestra las cabeceras de los ficheros almacenados en la cinta. Puede resultar muy útil para pasar programas de cinta a disco.

El comando COPY

En principio, el comando COPY sirve, además de para imprimir la pantalla, para copiar ficheros; por ejemplo:

```
COPY "A:Pepe.Bas" TO "A:Perico.Bas"
COPY "A:Pepe.Bas" TO "M:"
COPY "M:*.Dat" TO "B:"
```

Con la única limitación de no copiar un fichero sobre sí mismo, ni utilizar comodines en el nombre de destino. Pero hay tres formas especiales del comando COPY que es interesante comentar:

```
COPY "nombre.ext" TO SCREEN*
```

Muestra por pantalla el contenido de un fichero, en forma de caracteres ASCII. Obviamente, solo es útil para ficheros que contengan un texto visualizable.

```
COPY "nombre.ext" TO LPRINT
```

Igual que el anterior, pero la salida se hace por impresora.

```
COPY "nombre.ext" TO SPECTRUM FORMAT
```

Copia el fichero "nombre.ext" a un "nombre.Hed" que contiene una cabecera de 128 bytes para permitir al Spectrum reconocerlo. La cabecera es creada cómo correspondiente a un bloque de bytes con el dato de longitud correctamente ajustado, pero sin información sobre la dirección de carga, que deberá ser especificada -mediante CODE dirección- cuando se cargue el fichero. La utilidad fundamental de este comando es convertir al formato del Spectrum, programas en código máquina escritos en otros ordenadores y ensamblados con un ensamblador cruzado de Z-80 (práctica habitual entre los programadores profesionales).

Con toda seguridad, nos habremos dejado algo en el tintero. Un sistema operativo de disco es algo suficientemente complejo cómo para no poder ser visto totalmente en un artículo de cuatro páginas. Esperamos, no obstante, haber proporcionado una visión global que ayude al usuario a aproximarse a su Spectrum Plus 3 y le abra las puertas de otros sistemas operativos de disco más complejos.