

Talent MSX

Controlador de discos

Telemática S.A.



CONTROLADOR DE DISCOS FLEXIBLES

TALENT TDC-600

MANUAL DEL USUARIO

Producido en Argentina por Telemática S.A.

**Controlador de Discos Flexibles
Talent TDC-600
Manual del Usuario**

(c) 1989 Telemática S.A.
Chile 1347 - Tel.37-0051 al 54
(1098) Buenos Aires - Argentina

INTRODUCCION

INTRODUCCION

Acerca de este manual.....	7
Convenciones Tipográficas.....	8
El paquete del Controlador TDC-600	9
Requerimientos	10

TEORIA E INSTALACION

Características del Controlador TDC-600	13
Instalación.....	14
Archivos	15
Metacaracteres.....	16
Drives físicos y lógicos.....	18

MSX-DOS

Sistema Operativo	23
El MSX-Disk Operating system	24
Archivos del diskette de MSX-DOS.....	24
Cargando el MSX-DOS	25
El Procesador de Comandos del MSX-DOS.....	27
Generando una Copia del Diskette de Sistema	27
Desconectando el Sistema	27
Utilización de los Comandos	28
Opciones de Comandos.....	29
Información común a todos los Comandos.....	30
Procesamiento por lotes (BATCH).....	31
Parámetros	32
Como reducir las necesidades de memoria.....	34
Comandos del MSX-DOS.....	35
Duplicar diskettes.....	40
Creación de archivos y el CONFIG.SYS.....	52
Teclas especiales de edición en MSX-DOS.....	54
Caracteres de Control	56
Errores de disco	57

MSX DISK BASIC

Especificación de archivo	61
Comandos del Msx-Disk Basic	62
Archivos Secuenciales.....	86
Archivos de acceso directo.....	87
Funciones del MSX Disk BASIC.....	96

APENDICES

Características Técnicas.....	109
Notas para programadores.....	110
Descripción del FCB (File Control Block).....	110
Llamadas al BDOS	110
Modificación en la velocidad de motor	119
En caso de falla.....	122
MSX-DOS - Guía de referencia rápida.....	123
MSX Disk BASIC - Guía de referencia rápida	124
Códigos y mensajes de error de MSX Disk BASIC.....	127

ACERCA DE ESTE MANUAL

Este texto se organiza, para su mejor comprensión, de la siguiente manera.

La Introducción presenta el producto, los elementos que lo componen y los requerimientos para su uso.

El segundo capítulo presenta las características del producto y el procedimiento de instalación, seguidos de una aproximación teórica a la metodología de trabajo utilizando diskettes.

El tercer capítulo analiza las características y utilización del sistema operativo MSX-DOS. Encontrará información detallada de cada uno de sus comandos, la forma de utilizarlos, y la descripción de los mensajes de error.

El cuarto capítulo está dedicado al MSX-Disk BASIC y el manejo de archivos mediante sus comandos y funciones.

Los Apéndices están destinados a ampliar algunos aspectos de la información brindada en el manual. En ellos encontrará información más detallada, pero que requiere mayores conocimientos, sobre el uso del Controlador, sus características técnicas y las guías de referencia rápida del MSX-DOS y MSX-Disk BASIC.

NOTA:

Las menciones a la norma MSX en este manual, salvo donde se indique expresamente, se referirán tanto a MSX y MSX2 como a futuras expansiones que mantengan la compatibilidad.

CONVENCIONES TIPOGRAFICAS

A continuación se describen los tipos utilizados en este manual para diferenciar textos especiales, aclarando el significado de cada uno.

Monoespaciado	Esta tipografía representa el texto tal como aparece en la pantalla o texto que usted deberá ingresar. (Ej: PRINT HEX\$(76))
[]	Corchetes en comandos o instrucciones encierran entradasopcionales. (Ej: DISPLAY [ALL])
[opción opción]	Indica que debe escogerse alguna de las opciones pero no ambas.
{ }	Las llaves encierran comandos o instrucciones opcionales que pueden repetirse indefinidamente. (Ej: LIST [campo {, campo}])
<i>Cursiva</i>	Indica nombres de variables que deben ser reemplazados por el usuario. (Ej: OPEN "nombre-de-archivo")
<tecla>	Los signos "Mayor que" y "Menor que" indican el nombre de una tecla de la consola. (Ej: <RETURN>)
<tecla> + <tecla>	Se mencionan dos o más teclas unidas con un signo + para indicar que debe mantenerse pulsada la primera mientras se pulsa la segunda. (Ej: <SHIFT> + <TAB>)

EL PAQUETE DEL CONTROLADOR TDC-600

Antes de comenzar verifique los componentes del Controlador.

Una carpeta conteniendo:

Cartucho TDC-600

Este Manual

Un diskette con los archivos:

MSXDOS.SYS

COMMAND.COM

CONFIG.SYS

REQUERIMIENTOS

Para utilizar el controlador TDC-600 debe contar con una configuración mínima compuesta por:

Una computadora MSX con 64 kbytes de memoria RAM.

El cartucho TDC-600 (no requiere alimentación externa).

Uno o dos drives de 5 1/4 o 3 1/2 pulgadas, en cualquier combinación, con fuente de alimentación.

TEORIA E INSTALACION

CARACTERISTICAS DEL CONTROLADOR TDC-600

El cartucho TDC-600 es un controlador de drives desarrollado íntegramente por Telemática S.A., que permite conectar hasta dos drives a su computadora MSX.

Por las especificaciones de su software interno puede aceptar el trabajo con los siguientes tipos de diskettes:

5,25 pulgadas:

Simple lado (180 Kbytes)*

Doble lado (360 Kbytes)

3,50 pulgadas:

Simple lado: (360 Kbytes)

Doble lado: (720 Kbytes)

Pueden utilizarse simultáneamente hasta dos drives de la misma o diferente clase.

* Puede ser leído y escrito, pero el software del controlador no contempla un formateo de

INSTALACION

La instalación de su cartucho TDC-600 es sumamente sencilla, pero requiere de la estricta observación de los siguientes puntos:

Verificar que la computadora esté apagada: *Jamás* inserte el cartucho con la computadora encendida. Hacerlo puede significar severos daños en los circuitos del cartucho o en los de la computadora.

Seguidamente se procederá a la conexión del o los drives en el conector posterior del cartucho TDC-600.

Los drives Talent TPF-725 (para diskette 5,25) y Talent TPF-723 (para diskette 3,50) vienen provistos con el cable y ficha de conexión necesarias. Si ya de alguno de estos drives siga las instrucciones del manual a efectos de conectarlo debidamente.

Si debe conectar dos drives Talent deberá valerse del cable de conexión TPF-724, el que se provee con las correspondientes instrucciones de uso.

En caso de usar drives de otras marcas necesitará que estas dispongan de un cable especialmente construido para poder conectarlas al cartucho TDC-600. Si necesita construir o hacer construir dicho cable, remítase al apéndice Especificaciones Técnicas de este manual, en el que encontrará la información de la distribución de señales en el conector del cartucho TDC-600.

Terminada la conexión del o los drives puede recién encender su computadora y comenzar el trabajo.

ARCHIVOS

Un archivo es un conjunto de información relacionada, grabada en un diskette con un nombre que lo identifica.

Todos los programas, datos y textos de su diskette están almacenados en archivos y cada archivo tiene un nombre único. Para referirse a un archivo se utiliza ese nombre.

El nombre de un archivo se divide en dos partes:

La primera de ellas, el nombre propiamente dicho, puede tener una longitud de 1 a 8 caracteres. La segunda, llamada extensión, puede tener de 1 a 3 caracteres y se separa del nombre por un punto. Por ejemplo, el archivo:

```
+-----nombre  
| +---extensión  
| |  
MSXDOS.SYS
```

Para evitar confusiones, a lo largo del manual, al conjunto de estas dos partes se lo llama:

especificación de archivo.

Puede definirse una *especificación de archivo* sin extensión, pero no sin nombre.

Se puede ingresar el nombre de un archivo en mayúsculas o minúsculas, el software del controlador lo transforma automáticamente a mayúsculas.

Los nombres o extensiones de sus archivos pueden componerse de letras, números y los siguientes caracteres:

\$ & # % ' () - @ / ^ { } ~ ' !

En muchos casos, debe especificarse también el drive en que se encuentra el archivo seleccionado, para ello, la *especificación de archivo* en su forma más extensa es:

[drive:] nombre [.extensión]

donde *drive* es la letra que especifica el drive, seguido de dos puntos.
Por ejemplo:

A:CARTA.DOC

B:NUMEROS.MP

METACARACTERES

En una *especificación de archivo* se pueden utilizar dos caracteres especiales: el asterisco (*) y el signo de interrogación (?).

Un signo de interrogación (?) dentro de un nombre o extensión indica que cualquier carácter puede ocupar esa posición.

Por ejemplo, PRUE?AS.COM responderá a todos los archivos del directorio que tengan siete caracteres de nombre, comiencen con PRUE, terminen con AS y tengan una extensión .COM. Por ejemplo:

PRUE1AS.COM

PRUEBAS.COM

PRUE\$AS.COM

Un asterisco (*) dentro de un nombre o extensión indica que cualquier carácter o grupo de caracteres pueden ocupar desde esa posición en adelante.

A PRUE*.COM corresponden a todos los archivos cuyos nombres comienzan con PRUE y tienen una extensión .COM, por ejemplo:

PRUEBA.COM
PRUE1A.COM
PRUEBAS.COM
PRUE.COM

La representación con los caracteres ** representa a todos los archivos de un diskette.

El software del controlador utiliza nombres reservados de tres letras para designar internamente a ciertos dispositivos y por lo tanto son nombres de archivos no permitidos.

No debe utilizar los siguientes nombres:

AUX	Se utiliza para referirse a entrada o salida de datos desde o hacia un dispositivo auxiliar.
CON	Se utiliza para referirse a ingreso por teclado o salida a la pantalla.
LST ó PRN	Se utiliza para referirse al dispositivo impresora.
NUL	Se utiliza cuando no desea crear un archivo específico, pero el comando requiere un nombre de archivo de ingreso o egreso.

DRIVES FISICOS Y LOGICOS

El controlador TDC-600 es capaz de manejar hasta 2 drives.

Cada diskettera conectada es considerada un drive *físico*.

Pero el controlador es capaz de simular dos drives aunque sólo se haya conectado uno.

Esta capacidad de simular la existencia de dos drives nos introduce en un nuevo concepto: el de drive *lógico*.

El Drive Lógico es una entidad que puede guardar o no relación con los Drives físicos.

En una configuración con dos drives, cada uno lógico se corresponde con uno físico. El drive lógico A y el drive físico A son el mismo.

Pero si se dispone de un sólo drive físico, éste se comportará de acuerdo a cada caso como drive A o drive B.

En un sistema de un solo drive, Ud. ingresa los comandos como si tuviera dos unidades. Por ejemplo:

COPY A:COMMAND.COM B:

Este comando indica al Sistema Operativo que debe copiarse el archivo COMMAND.COM del drive lógico A al drive lógico B.

Cuando llega el momento de usar el drive B aparecerá en pantalla el mensaje:

**Insert diskette for drive B:
and strike a key when ready**

Ese es el momento de cambiar los diskettes, colocando el que debería estar en el drive B. Al completar el proceso inmediatamente observará el mensaje:

```
1 File(s) copied  
A>
```

De haber muchos archivos que copiar se irá pidiendo alternativamente que coloque los diskettes correspondientes a los drives A y B.

En este ejemplo el drive lógico al finalizar la operación es el B (drive lógico donde se terminó de hacer la copia). El sistema queda preparado para hacer una nueva operación en él, por lo tanto, si ahora se escribe un comando o programa cualquiera que deba ejecutarse sobre el drive A, la computadora le pedirá que vuelva a reemplazar el diskette.

Este comando es ejecutado en cada uno de los canales de los 16
que contiene el controlador. Sección 16. Sección 16. Sección 16.

Sistema Controlador

Este sistema observable en el tiempo con ciertas condiciones iniciales
de la envíos en el establecimiento entre el sistema de
control y el sistema de control. El controlador se divide en dos secciones:
una sección de control de la red y otra sección de control de los
motores. Una de las secciones de control de los motores se encarga
de controlar el movimiento de los motores de acuerdo con las órdenes
que se le envían. La otra sección de control de los motores se encarga
de controlar el movimiento de los motores de acuerdo con las órdenes
que se le envían.

MSX-DOS

SISTEMA OPERATIVO

El sistema operativo es un programa que actúa como interfaz entre el hardware de la computadora y el software del usuario. Su función consiste en crear un entorno estandarizado, posibilitando que el software se independice del hardware específico de la computadora.

Sus funciones principales son:

- Administrar la memoria secundaria (drives conectados a la consola).
- Definir la distribución de la memoria principal.
- Administrar los canales de Entrada y Salida.
- Permitir la carga y ejecución de programas.

EL MSX-DISK OPERATING SYSTEM

MSX-DOS es el sistema operativo de diskette diseñado por Microsoft para las computadoras MSX. MSX-DOS significa MSX-Disk Operating System (sistema operativo de disco MSX).

MSX-DOS le permite crear y actualizar archivos, cargar y ejecutar programas, y acceder a dispositivos externos (como por ejemplo un drive o una impresora) que se encuentren conectados a su computadora, facilitando la administración de los recursos del sistema.

ARCHIVOS DEL DISKETTE DE MSX-DOS

Su cartucho TDC-600 viene acompañado de un diskette que contiene los siguientes archivos:

<i>ARCHIVO</i>	<i>FUNCION</i>
MSXDOS.SYS	Sistema operativo MSX-DOS.
COMMAND.COM	Procesador de comandos MSX-DOS.
CONFIG.SYS	Archivo de configuración del MSX-DOS.

CARGANDO EL MSX-DOS

Encienda el o los drives y coloque el diskette del sistema* en el A.

Durante el proceso de carga, el sistema verifica que esté disponible en el diskette el archivo denominado **MSXDOS.SYS**. Si no está presente, toma el control el MSX Disk BASIC. En la sección correspondiente se describe qué sucede en este caso.

Una vez cargado el MSX-DOS, el sistema busca el archivo **COMMAND.COM** en el diskette del MSX-DOS y lo carga en memoria (si lo encuentra). El archivo **COMMAND.COM** es un programa que ejecuta los comandos del sistema operativo que usted ingresa. También se lo denomina procesador de comandos.

Cuando se carga y se ejecuta el procesador de comandos, verá la siguiente presentación en la pantalla:

```
MSX-DOS Versión X.XX
Copyright 1984 by Microsoft
```

```
Command versión X.XX
```

```
Current date is Sun 1-01-1984
Enter current date:_
```

De esta manera, el MSX-DOS pide que ingrese la fecha de hoy** en el formato **mm-dd-aa**, donde **mm** es el mes, **dd** es el día del mes, y **aa** el año.

Los tres deben expresarse en números, separados por guiones (-) o barras (/). El año puede expresarse con los cuatro dígitos o los dos últimos.

* El diskette de sistema es el que se incluye con el cartucho controlador o una copia del mismo.

** Si su computadora es MSX2, MSX-DOS no pedirá la fecha porque la toma directamente del reloj de tiempo real.

Por ejemplo, 4-13-86 ó 04/13/1986 son respuestas válidas para el ingreso de fechas. Si ingresa una fecha inválida o con formato incorrecto, el sistema la rechazará y se la pedirá nuevamente.

En caso de no querer modificar la fecha, simplemente pulse

<RETURN>

EL PROCESADOR DE COMANDOS DEL MSX-DOS

Luego de responder al pedido de fecha, se muestra por pantalla lo siguiente:

A>

Esta señal es el indicador (prompt) del procesador de comandos MSX-DOS.

La letra A es el indicador del drive sobre el que van a efectuarse los comandos.

GENERANDO UNA COPIA DEL DISKETTE DE SISTEMA

Antes de comenzar, genere una copia de respaldo del sistema operativo.

Para copiar diskettes, refiérase a los comandos COPY y FORMAT de MSX-DOS.

DESCONECTANDO EL SISTEMA

No existe ningún comando para *desconectar* el MSX-DOS. Para finalizar sus tareas con el sistema, abra la puerta del drive y saque los diskettes. Luego, simplemente apague su computadora.

IMPORTANTE: Siempre recuerde quitar sus diskettes de los drives antes de apagar el equipo.

UTILIZACION DE LOS COMANDOS

Existen dos tipos de comandos del MSX-DOS:

COMANDOS INTERNOS: Son aquellos que integran el Procesador de Comandos (**COMMAND.COM**). Su ejecución se realiza en forma directa sin que deban ser cargados del drive. Estos son:

BASIC
COPY
DATE
DEL
DIR
FORMAT
MODE
PAUSE
REN
TYPE

COMANDOS EXTERNOS: Se trata de programas grabados en el diskette con extensión **.COM** ó **.BAT**. Para pedir su ejecución deben darse dos condiciones:

- Que el comando se encuentre en el diskette.
- Que se lo invoque a través de su nombre (la extensión no es necesaria).

OPCIONES DE COMANDOS

Algunos comandos tienen opciones que pueden indicarse por medio de *selectores*. Están compuestos por la barra (/) y una letra, y se escriben a continuación del comando. Por ejemplo:

DIR /W

Si no se ingresa ninguna opción, el MSX-DOS asume ciertos valores supuestos para cada caso.

En la descripción individual de cada comando y sus valores supuestos se indican las distintas opciones que cada uno puede aceptar.

El formato que utilizamos para describir los comandos del MSX-DOS es:

Comando [opciones...]

Donde

Comando: Se refiere a cualquier nombre de comando válido y

Opciones: Controlan a los comandos del MSX-DOS. Están precedidos de una barra (por ejemplo, /P).

INFORMACION COMUN A TODOS LOS COMANDOS

- Los comandos pueden ser seguidos por una o más opciones.
- Los comandos y sus opciones pueden ingresarse indistintamente en mayúsculas o minúsculas, o una combinación de ambas.
- Los comandos y sus opciones deben separarse por medio de espacios.
- Para cancelar la ejecución de un comando, utilice las teclas <CTRL>+<STOP> ó <CTRL>+<C>.
- Todos los comandos se ejecutan luego de pulsar <RETURN>.
- Los nombres de dispositivos (por ejemplo, PRN ó CON) no pueden utilizarse como nombres de comandos.
- Cuando un comando produce una gran cantidad de información por pantalla, ésta sale en forma ininterrumpida. Para poder leer dicha información, se podrá introducir una pausa en la ejecución del comando pulsando las teclas <CTRL>+<S>. Para continuar, pulse cualquier tecla.
- Las teclas de función y edición pueden utilizarse para ingresar comandos del MSX-DOS. Más adelante se detallan cada una de sus funciones.
- Cuando no se especifica el drive donde buscar el comando o los archivos a los que este afecta, el MSX-DOS busca en el drive señalado por el *prompt*.

PROCESAMIENTO POR LOTES (BATCH)

A menudo se realiza reiteradamente una secuencia determinada de comandos. El MSX-DOS permite almacenar estos comandos en un tipo de archivo especial llamado *Batch*.

Estos se ejecutan ingresando el nombre del archivo (la extensión puede obviarse). Los comandos se van ejecutando uno a uno como si se ingresaran por teclado.

Se rotulan con la extensión .BAT.

Pueden crearse archivos batch utilizando algún editor de texto o con el comando COPY, como se explica más adelante.

Existen dos comandos del MSX-DOS diseñados expresamente para archivos batch: REM y PAUSE.

REM permite incluir comentarios y advertencias en su archivo batch sin que se los tome como comandos ejecutables.

PAUSE muestra por pantalla un mensaje que le indica que debe pulsar cualquier tecla para continuar.

Si se presiona <CTRL>+<C> ó <CTRL>+<STOP> durante la ejecución de un archivo batch, MSX-DOS pregunta si quiere terminar el proceso:

Terminate batch file (Y/N)?

Si pulsa <Y>, los comandos pendientes del archivo batch se ignoran y el sistema vuelve al modo comando.

Si Ud. pulsa <N>, el proceso continúa con el siguiente comando del archivo batch.

Si retira el diskette que contiene el archivo batch durante la ejecución de algún comando, MSX-DOS indica, al finalizar su ejecución, que debe volver a colocar ese diskette:

Insert disk with Batch file and
Strike a key when ready

El último comando de un archivo batch puede ser el nombre de otro archivo batch, permitiendo encadenarlos unos con otros.

El archivo AUTOEXEC.BAT es un caso especial dentro de los archivos batch. Si está grabado en el diskette que se instala al encender la computadora es ejecutado automáticamente.

Este archivo permite ejecutar un comando (o conjunto de comandos) cada vez que cargue el MSX-DOS.

PARAMETROS

En ocasiones el usuario puede verse en la obligación de crear un archivo Batch que sirva para ser utilizado con datos variables.

Se pueden crear archivos batch con parámetros que se designan como %0 a %9, y se reemplazan por los valores reales que se suministren en el momento de su ejecución.

El siguiente ejemplo presenta un archivo llamado LOTE.BAT con el siguiente contenido:

```
COPY %1.MAC %2.MAC
TYPE %2.PRN
TYPE %0.BAT
```

Los parámetros %1 y %2 son reemplazados secuencialmente por los parámetros provistos en la invocación al comando. Por ejemplo, al invocarlo de la siguiente manera:

A>LOTE A:NUEVO B:OTRO

La variable %1 será reemplazada por el primer parámetro ingresado (A:NUEVO). La variable %2 será reemplazada por el segundo (B:OTRO).

La variable %0 es reemplazada siempre por el nombre del archivo batch invocado (En este ejemplo: LOTE). Esta variable incluye también, si fué especificado, el drive.

NOTAS: Se admiten hasta 10 parámetros (%0 a %9).

Si necesita el signo porcentaje (%) como parte de un nombre de archivo dentro del archivo batch, deberá indicarlo dos veces. Por ejemplo, para especificar el archivo ABC%.COM deberá escribir ABC%%.COM.

El archivo AUTOEXEC.BAT no admite parámetros.

*CUANDO NO ESTA
DISPONIBLE EL
DISKETTE DEL
MSX-DOS*

Como se dijo anteriormente, si no se encuentra disponible el archivo MSX-DOS durante el proceso de carga, toma el control el MSX Disk Basic. En este caso, se verá lo siguiente en pantalla:

Enter date (M-D-Y):

Se deberá ingresar la fecha como ya se indicó anteriormente. Luego el sistema busca un archivo llamado AUTOEXEC.BAS (que deberá ser un programa Basic). Si no lo encuentra, entra directamente en el MSX Disk Basic.

COMO REDUCIR LAS NECESIDADES DE MEMORIA DEL SISTEMA

Cuando se está efectuando la carga del sistema, pueden reducirse o cancelarse las necesidades de memoria requeridas por el uso del drive.

Para ello, basta con presionar la tecla <SHIFT> cuando se enciende la consola, para que el(los) drive(es) queden deshabilitados (como si no estuvieran conectados). Toda llamada a los comandos, sentencias o funciones de diskette generará el error "Illegal Function Call".

Análogamente, la tecla <CTRL> deshabilita la simulación de los drives lógicos, liberando memoria del sistema operativo. En este caso las referencias al drive B quedan deshabilitadas.

COMANDOS DEL MSX-DOS

En esta sección se desarrolla cada uno de los comandos que integran el MSX DOS. Son ellos:

BASIC

COPY

DATE

DEL Ó ERASE

DIR

FORMAT

MODE

PAUSE

REM

REN Ó RENAME

TYPE

BASIC

SINTAXIS:

BASIC [*especificación-de-archivo*]

PROPOSITO:

Sale del trabajo directo en MSX-DOS y entra al MSX-DISK BASIC.

COMENTARIO:

Este comando invoca al MSX-DISK BASIC desde el MSX-DOS.

Si se incluye la *especificación-de-archivo* correspondiente a un programa BASIC, éste será cargado y ejecutado inmediatamente.

Para volver al MSX-DOS desde BASIC, utilice la sentencia

CALL SYSTEM (ó _SYSTEM)*

COPY

SINTAXIS:

COPY *especif-de-archivo-origen* *tespecif-de-archivo-destino*

PROPOSITO:

Copia uno o más archivos.

COMENTARIO:

Si no se indica la *especificación-de-archivo-destino*, la copia se hará en el drive activo y el nombre del archivo copiado será el mismo del archivo origen.

Si en la *especificación-de-archivo-origen* es asignado el drive activo y en la *especificación-de-archivo-destino* no se indica dato alguno, el comando **COPY** se aborta, pues no se puede copiar un archivo sobre sí mismo. En este caso MSX-DOS nos devuelve el mensaje de error:

```
File cannot be copied onto itself  
0 files copied
```

La *especificación-de-archivo-destino* puede tomar tres formas:

Sólo el drive

El archivo origen se copia con el mismo nombre en el drive indicado.

(Ej: **COPY COMMAND.COM B:**)

Sólo el nombre de archivo

El archivo origen se copia en el drive activo bajo el nombre indicado.

(Ej: **COPY MANUAL1.DOC MANUAL.TXT**)

El drive y el nombre

El archivo origen se copia en el drive indicado con el nombre especificado.

(Ej: **COPY NOTAS.DBF B:ALUMNOS.DBF**).

El comando **COPY** también permite encadenar archivos mientras se copian. Para hacerlo, basta indicar los nombres de dichos archivos como *especificación-de-archivo-fuente*, unidos por el signo *más* (+). Por ej:

```
COPY PROG1.BAS+PROG2.BAS+PROG3.TXT GRANARCH.CRP
```

Este comando encadena los archivos PROG1.BAS, PROG2.BAS y PROG3.TXT en el archivo GRANARCH.CRP.

No ingrese un comando **COPY** encadenado donde alguno de los archivos a copiar tenga el mismo nombre que el archivo destino. Por ejemplo, el siguiente comando es un error si **TODOS.LST** existe en el diskette:

```
COPY *.LST TODOS.LST
```

Este error no es detectable por la computadora. El nuevo archivo **TODOS.LST** se graba encima del existente. En estos casos la destrucción del archivo anterior hace imposible la recuperación de la información que contenía.

Si los nombre de fuente y destino son iguales, el archivo fuente se saltea, y se muestra el mensaje de error "Content of destination lost before copy" (el contenido del destino se perdió antes de copiarlo).

Los siguientes encadenamientos continúan normalmente. Esto permite *sumar* archivos, como en el siguiente ejemplo:

```
COPY TODOS.LST+*.LST
```

Este comando agrega todos los archivos *.LST, excepto el mismo TODOS.LST, al final de TODOS.LST. Este comando no produce error y es la forma correcta de agregar varios archivos a continuación de otro.

Dado que son los archivos ASCII los que usualmente se encadenan, este comando interpreta un CTRL z (caracter 1AH) como indicador de fin de archivo.

Por lo tanto, se necesita el selector "/B" para utilizar el final de archivo físico (la longitud del archivo que se muestra con el comando DIR), para encadenar archivos binarios. Por ej:

COPY/B A.COM+B.COM

donde el archivo B.COM se agrega a continuación del archivo A.COM, siendo el nombre del archivo destino A.COM.

Este selector tiene efecto para el archivo que se indica y los subsiguientes. Para volver a poner el modo ASCII, debe utilizarse "/A".

Cuando no se encadena, los selectores "/A" y "/B" también son válidos, y el tipo de archivo por omisión es binario. "/A" termina la copia cuando encuentra el primer CTRL Z.

DUPPLICAR DISKETTES

El comando COPY permite duplicar el contenido de todos los archivos de un diskette en otro.

El comando para duplicar un diskette es:

COPY A:.* B:

Si tiene sólo un drive, el controlador leerá todos los archivos que entren en memoria (irá mostrando los nombres), y luego pondrá en pantalla el mensaje:

**Insert diskette for drive B:
and strike a key when ready**

Indicando que debe colocar el diskette destino (el que debería estar en el drive B) y pulsar una tecla. Si la copia no terminó de completarse se desplegará el mensaje

**Insert diskette for drive A:
and strike a key when ready**

Indicando que debe volver a poner el diskette fuente y pulsar una tecla. Leerá la información restante y volverá a pedir el cambio de diskettes tantas veces como sea necesario hasta terminar la copia.

NOTA: Como al terminar el proceso el último diskette colocado fué el destino, MSX-DOS supone que el drive activo es el B. Por eso, al efectuar el siguiente comando, por ejemplo, DIR, MSX-DOS pedirá que coloque el diskette del drive A:.

DATE

SINTAXIS:

DATE [*mm-dd-aa*]

PROPOSITO:

Consulta o cambia la fecha del sistema. Esta fecha se grabará en la entrada de directorio de todos los archivos que a partir de ese momento se generen o modifiquen.

COMENTARIOS:

Podrá cambiar la fecha desde su consola o a través de un archivo batch. (MSX-DOS no pide el ingreso de fecha cuando utiliza un archivo AUTOEXEC.BAT, y por lo tanto Ud. podría desear incluir un comando DATE en ese archivo).

Si Ud. ingresa DATE, el comando responderá con este mensaje:

Current date is <fecha>

Enter new date:

(La fecha actual es... Ingrese nueva fecha:)

Pulse <RETURN> si no desea modificar esta fecha.

También puede ingresar una nueva fecha con el siguiente formato:

DATE <*mm-dd-aa*>

En este caso, MSX-DOS no hace la pregunta "Enter new date:"

La nueva fecha debe ingresarse utilizando únicamente números; no están permitidas las letras. Las opciones permitidas son las siguientes:

<*mm*> de 1-12 (el mes)

<*dd*> de 1-31 (el día)

<*aa*> de 0-79 (significa 2000 a 2079)

ó 80-99 (significa 1980 a 1999)

ó 1980-2099

Las entradas de día, mes y año pueden estar separadas por guiones (-), barras (/), o puntos (.). El MSX-DOS valida los meses y años, según el mes tenga 31, 30, 29 ó 28 días o los años sean bisiestos.

Si los datos o los separadores no son válidos, DATE nos muestra el siguiente mensaje

```
Invalid date  
Enter new date:
```

esperando que reingrese una fecha válida.

DEL

SINTAXIS:

DEL *especificación-de-archivo*

SINONIMOS:

DELETE - **ERASE**

PROPOSITO:

Borra los archivos que correspondan a la *especificación-de-archivo*.

COMENTARIOS:

Si *especificación-de-archivo* es ***.***, MSX-DOS muestra el mensaje

Are you sure?

(¿Está seguro?)

Si pulsa **<Y>**, seguida de **<RETURN>**, la respuesta se toma como afirmativa.

EJEMPLOS:

DEL PROG1.BAS

DEL *.BAS

DELETE PROG3.*

DEL PROG???.B??

ERASE PROG5

DIR

SINTAXIS:

DIR [*especificación-de-archivo*] [/P] [/W]

PROPOSITO:

Lista los archivos de un diskette.

COMENTARIOS:

Si Ud. ingresa únicamente DIR, se listan los archivos en el drive activo con su tamaño en bytes, la hora y el día de su última modificación.

Si se ingresa una especificación de drive (por ejemplo: DIR B:), el listado se realiza sobre el drive especificado.

Si *especificación-de-archivo* menciona sólo un nombre sin extensión, se listarán todos los archivos con ese nombre.

Los metacaracteres (?) y (*) pueden utilizarse en la opción *especificación-de-archivo*.

Los siguientes formatos de DIR son equivalentes:

DIR	ó	DIR *.*
DIR nombre	ó	DIR nombre.*
DIR .ext	ó	DIR *.ext
DIR .	ó	DIR *.

Se pueden especificar dos selectores.

Con /P el directorio hace una pausa cuando se completa la pantalla. Para continuar debe pulsarse cualquier tecla.

Con /W se muestran sólo los nombres de archivo, sin ninguna otra información, tantos como sea posible por cada línea.

FORMAT

SINTAXIS:

FORMAT

PROPOSITO:

Inicializa (o formatea) un diskette para que acepte archivos MSX-DOS.

COMENTARIOS:

Este comando inicializa el directorio, las tablas de ubicación de archivos (File Allocation Tables) y el *boot sector**. Todo diskette nuevo debe formatearse antes de poder ser utilizado. Si se formatea un diskette con información, los archivos que contenga se destruyen.

Al invocarlo, MSX-DOS pregunta

Drive name? (A,B)

Pidiendo que seleccione el drive. Pulse <A> ó y verá el mensaje:

- 1) 3.5" Simple lado (360 Kb)
- 2) 3.5" Doble lado (720 Kb)
- 3) 5.25" Doble lado (360 Kb)

En esta ocasión se le pide que indique el tipo de drive físico donde se ha alojado el diskette a formatear. La elección se realiza pulsando el número correcto. Luego emite el mensaje

Strike a key when ready

deteniéndose hasta que inserte el diskette a formatear y pulse una tecla.

El formateo tarda entre 50 y 90 segundos. Al terminar, MSX-DOS informa

Format complete

* No es necesario que el usuario conozca el significado de estas estructuras.

MODE

SINTAXIS:

MODE *ancho*

PROPOSITO:

Selecciona el ancho de la presentación en pantalla.

COMENTARIOS:

ancho indica la cantidad de caracteres por línea. Debe ser un valor entre 1 y 40 para MSX, ó entre 1 y 80 para MSX2. Si *ancho* es menor o igual que 32 se selecciona automáticamente el modo de pantalla 1 (SCREEN 1 del MSX BASIC), de lo contrario, el modo es 0.

El valor por defecto puede fijarse en MSX2*, y en MSX es 37.

* Sobre cómo hacerlo, refiérase al comando SET SCREEN en las Extensiones al MSX2 BASIC.

PAUSE

SINTAXIS:

PAUSE [*comentario*]

PROPOSITO:

Detiene momentáneamente la ejecución de un archivo batch.

COMENTARIOS:

Cuando el procesador de comandos encuentra PAUSE, muestra por pantalla

Strike a key when ready...

(Pulse una tecla cuando esté listo...)

Si pulsa <CTRL>+<C>, MSX-DOS pregunta

Terminate batch file (Y/N)?

(¿Finaliza archivo batch?)

Si pulsa "Y" la ejecución del resto del archivo batch se abortará, y retomará el control el MSX-DOS.

El *comentario* es opcional y puede ingresarse en la misma línea que PAUSE.

EJEMPLO:

PAUSE Cambie el diskette

REM

SINTAXIS:

REM [*comentario*]

PROPOSITO:

Muestra en pantalla los comentarios durante la ejecución de un programa batch.

COMENTARIOS:

Los únicos separadores permitidos en el comentario son espacio, tabulador y coma.

EJEMPLO:

REM Programa auxiliar

REN

SINTAXIS:

REN *especificación-de-archivo* *nueva-especificación-de-archivo*

SINONIMO:

RENAME

PROPOSITO:

Cambia el nombre del archivo indicado en *especificación-de-archivo* por el que figure en *nueva-especificación-de-archivo*

COMENTARIO:

Cualquier designación de drive que se indique en *nueva-especificación-de-archivo* es ignorada. El archivo permanecerá en el drive dónde reside.

Los metacaracteres pueden utilizarse en cualquiera de los dos *especificaciones-de-archivo*.

Por ejemplo, para cambiar la extensión .LST por .PRN en todos los archivos:

REN *.LST *.PRN

Para renombrar el archivo ABODE en el drive B a ADOBE:

RENAME B:ABODE ?D?B?

Si intenta renombrar un archivo con un nombre que ya existe en el directorio, se emitirá el mensaje "Rename error".

TIME

SINTAXIS:

TIME [hh [:mm [:ss]]]

PROPOSITO:

Muestra y modifica la hora del sistema en computadoras MSX2.

COMENTARIOS:

Si se ingresa el comando TIME sin ningún argumento, se muestra el siguiente mensaje:

Current time is <hora>

Enter new time:

Hora actual es ...

Ingresar nueva hora:

Pulse <RETURN> si no desea cambiar la hora. También puede ingresarla escribiendo:

TIME 8:20

Deben ingresarse solamente números. Los rangos válidos son:

hh = 00 - 24

mm = 00 - 59

ss = 00 - 59

Deben estar separados por dos puntos (:).

MSX-DOS valida la hora ingresada. Si los valores o los separadores no son válidos emite el mensaje

Invalid time

Enter new time:

y espera a que ingrese una hora válida.

TYPE

SINTAXIS:

TYPE *especificación-de-archivo*

PROPOSITO:

Muestra por pantalla el contenido de un archivo.

COMENTARIOS:

Utilice este comando para examinar el contenido de un archivo.

El único formateo que ejecuta TYPE es la expansión de los caracteres de tabulación a espacios que coinciden con las tabulaciones de pantalla cada ocho columnas. Aplique este comando solamente a archivos en formato ASCII.

CREACION DE ARCHIVOS Y EL CONFIG.SYS

Dentro del entorno MSX-DOS es necesario muchas veces crear pequeños archivos en forma rápida, como en el caso de los archivos Batch o el CONFIG.SYS.

Esto se puede lograr copiando información desde la consola hacia un archivo, mediante el comando:

COPY CON *espec-de-archivo*

donde CON se refiere al dispositivo lógico consola, y *espec-de-archivo* indicará el archivo donde se desea grabar el resultado. Al ingresar este comando y pulsar <RETURN>, el cursor pasará a la línea de abajo sin mostrar el prompt A>. Ingrese el texto del archivo línea por línea. Para finalizar, pase a una última línea (pulsando <RETURN>), y como único carácter pulse <CTRL>+<Z> y oprima <RETURN>. Esto indica el fin del archivo. MSX-DOS lo grabará en el diskette y mostrará el mensaje:

1 File(s) copied

Algunos ejemplos:

A>COPY CON AUTOEXEC.BAT	<ENTER>
MODE 40	<ENTER>
BASIC MI-PROG.BAS	<ENTER>
^Z	<ENTER>

De la misma manera puede crearse el archivo de configuración CONFIG.SYS. Este es buscado por el MSX-DOS en el momento de cargarse y permite fijar ciertos parámetros del sistema, como el color de la pantalla, el contenido de las teclas de función, y en casos especiales, la inclusión de un programa controlador (driver)* para dispositivos especiales.

* Como ejemplo puede citarse el driver para el cartucho de expansión de 80 columnas.

Ejemplo:

```
A>COPY CON CONFIG.SYS
COLOR 15,1,1
KEY 1=DIR/
KEY 2=BASIC/
KEY 3=MODE 40
KEY 4=MODE 80
^Z
```

Aquí se definen el color de pantalla (letras blancas, fondo y borde negro), y las primeras 4 teclas de función. En las dos primeras, la barra al final de la línea simboliza el carácter 13 (RETURN).

TECLAS ESPECIALES DE EDICION EN MSX-DOS

No es necesario ingresar por teclado las mismas secuencias repetidamente, ya que la última línea de comando queda almacenada en un área denominada *plantilla*.

Si se equivoca en una línea de comando, Ud. podrá corregirla y reingresarl sin tener que volver a escribir el comando completo.

El siguiente es el listado de teclas de edición:

- | | |
|------------------------------|---|
| <CURSOR DERECHA> | Copia un carácter desde la plantilla a la línea de comando. |
| <CURSOR ABAJO> | Copia todos los caracteres restantes en la plantilla a la línea de comando. |
| | Saltea (no copia) un carácter en la plantilla. |
| <CURSOR ARRIBA> | Vacia la línea de comando. Deja la plantilla sin cambios. |
| <CURSOR IZQUIERDA>
Ó <BS> | Borra el último carácter de la línea de comando. |
| <INS> | Conmuta entre el modo inserción o reescritura. |

EJEMPLOS:

Si escribe

DIR COMMAND.COM

MSX-DOS muestra información referida al archivo COMMAND.COM. La línea de comando se almacena también en la plantilla. Para repetir el comando, simplemente pulse <CURSOR ABAJO> y <RETURN>.

Si se desea mostrar información sobre un archivo denominado COMMAND.ASM, pulse <CURSOR DERECHA> hasta que aparezca en pantalla

DIR COMMAND.

y luego escriba ASM.

CARACTERES DE CONTROL

Los caracteres de control son una combinación de teclas a pulsar que guardan un significado especial para el MSX-DOS. Ejemplos ya vistos en este manual son <CTRL>+<C> y <CTRL>+<S>.

Recuerde que cuando pulsa un carácter de control, como ser <CTRL>+<C>, debe pulsar <CTRL> y luego <C>, sin soltar la anterior.

A continuación se describen las funciones de los caracteres de control.

<CTRL>+<P> Envía un eco de lo que sale en la pantalla a la impresora.

<CTRL>+<N> Cancela el eco a impresora.

<CTRL>+<C> Interrumpe el comando en ejecución.

<CTRL>+<S> Suspende el envío de datos a la pantalla. Pulse una tecla para continuar.

ERRORES DE DISCO

Si se genera un error de diskette cuando se está ejecutando un comando o programa. MSX-DOS reintenta la operación tres veces.

Si la operación no puede completarse con éxito, MSX-DOS retorna un mensaje de error con el siguiente formato:

tipo error *acción* in drive *X*

Abort, Retry, Ignore ?

Donde *tipo* puede ser uno de los siguientes mensajes:

Write protect (Protección contra grabación)

Not ready (No está preparado)

Disk (Disco)

acción (Entrada o Salida) puede ser alguna de las siguientes:

Read (Lectura)

Write (Escritura)

X indica el drive donde el error ha ocurrido.

MSX-DOS espera que Ud. ingrese una de las siguientes respuestas:

Abort

(Abortar)

Pulsando la tecla <A>.

Finaliza la ejecución del comando o programa que pedía la lectura o grabación de diskettes.

Ignore

(Ignorar)

Pulsando la <I>.

Ignora el la lectura o escritura fallida y sigue funcionando como si el error no hubiese ocurrido (no recomendable).

Retry

(Reintentar)

Pulsando la <R>.

Reintenta la operación.

Otro mensaje de error que se puede relacionar con una lectura o escritura de diskette es:

Bad FAT

Este mensaje significa que la copia en memoria de una de las Tablas de Ubicación de Archivos (FAT) tiene punteros a bloques inexistentes.

Si este error persiste, el diskette no puede utilizarse y deberá formatearlo antes de utilizarlo nuevamente.

MSX DISK BASIC

Con la instalación del cartucho TDC-600 se adicionan al MSX BASIC de su computadora un conjunto de comandos y funciones que hacen posible el manejo de todas las operaciones con disco.

En este capítulo veremos en detalle cada uno de estos nuevos términos Basic. Para un mejor estudio los agrupamos en dos categorías:

- 1) Comandos
- 2) Funciones

ESPECIFICACION-DE-ARCHIVO

Algunas de estas nuevas palabras Basic requieren de la mención explícita de un archivo, o un conjunto de ellos.

Para nombrar a un archivo que se encuentre grabado en un disco, o que se deba grabar en él, hace falta respetar el formato del Sistema Operativo.

Este tema fue tratado con detenimiento en el capítulo Teoría e Instalación del manual. Si no lo ha leído aún, le recomendamos hacerlo antes de continuar la lectura.

Sólo nos queda agregar que en el entorno de trabajo MSX Disk Basic la *especificación-de-archivo* se escribe necesariamente entre comillas, ya que reciben por parte del intérprete el mismo tratamiento que los demás datos alfanuméricos.

Por esta misma razón es posible incluir la *especificación-de-archivo* como contenido de una variable alfanumérica. Es lo mismo entonces:

10 FILES "A:*.BAS"

que

```
10 A$="A:*.BAS"
20 FILES A$
```

COMANDOS DEL MSX-DISK BASIC

La lista completa de los comandos agregados es:

BLOAD
BSAVE
CALL FORMAT
CALL SYSTEM
CLOSE
COPY
DSKO\$
FIELD
FILES
GET
INPUT#
KILL
LFILES
LINE INPUT #
LOAD
LSET
MAXFILES
MERGE
NAME
OPEN
PRINT #
PRINT # USING
PUT
RSET
RUN
SAVE

BLOAD

SINTAXIS:

BLOAD *especificación-de-archivo* [,OFFSET] [,R]

BLOAD *especificación-de-archivo* [,OFFSET] [,S]

PROPOSITO:

Permite la carga de un programa en código de máquina o una imagen de pantalla

COMENTARIOS:

El programa se carga en la dirección de memoria que fue indicada en el momento de la grabación (ver BSAVE).

[,OFFSET] es un número que indica la cantidad de bytes que se desea desplazar la dirección inicial de carga de un archivo. Con un valor positivo esta dirección se desplaza en sentido creciente y con un valor negativo en sentido decreciente.

[,R] indica que una vez cargado el programa se debe iniciar su ejecución

[,S] el archivo es cargado en la Video-Ram. Se usa para cargar imágenes de pantalla grabadas en disco.

EJEMPLOS:

```
BLOAD "A:ZEN.BIN"  
BLOAD "ZEN.BIN",R  
BLOAD "MAXI.PIC",S  
BLOAD "PROG",-1000,R  
BLOAD "PROG2",&H50
```

NOTA: Para uso de BLOAD en archivos grabados en cassette remítase al manual de MSX BASIC

BSAVE

SINTAXIS:

BSAVE *especificación-de-archivo , inicio, fin [,ejecución]*

BSAVE *especificación-de-archivo , inicio, fin [, s]*

PROPOSITO:

Graba porciones de memoria RAM o de Video RAM.

COMENTARIO:

La grabación de programas en código de máquina requiere la indicación de las tres direcciones (*inicio, fin y ejecución*)

La grabación de un sector de Video-RAM requiere de las dos primeras direcciones y de la opción *[,s]*

Las direcciones deben ser indicadas por medio de expresiones numéricas.

EJEMPLO:

```
BSAVE "MAIN",32768,40000,32768
BSAVE "MAIN".&H8000,XX,&H8000
BSAVE "FUT.PIC",0,&H1030,S
BSAVE "PROG",A,B,C
```

NOTAS: Si se utilizan variables para indicar el valor de la dirección *ejecución* se debe tener cuidado de no emplear una con nombre ese (*s*), que será confundida con una referencia a VideoRAM.

Para grabar en cassette usando BSAVE remitirse al manual de MSX BASIC.

CALL FORMAT

SINTAXIS:

CALL FORMAT
FORMAT

PROPOSITO:

Inicializar (formatear) un nuevo disco.

COMENTARIOS:

Todo disco nuevo requiere del proceso de formateo, que consiste en realizar una serie de grabaciones preliminares que permitan posteriormente a la computadora identificar los diferentes lugares físicos del disco donde leer o grabar información. De haber información en el diskette, será destruida.

Ver detalles de uso en la segunda sección destinada al MSX-DOS, en la página correspondiente al comando **FORMAT**.

CALL SYSTEM

SINTAXIS:

CALL SYSTEM
_SYSTEM

PROPOSITO:

Abandonar el trabajo en MSX Disk Basic y cargar el MSX DOS.

COMENTARIOS:

Esto es posible solamente cuando el BASIC haya sido invocado desde el MSX-DOS.

Con este comando los archivos que hubiere abiertos son cerrados y el programa y los datos alojados en memoria son abandonados.

Tome la precaución de grabar los programas o datos que desee conservar antes de ejecutar este comando.

Al invocarlo debe estar presente en el drive un diskette con el MSX-DOS.

CLOSE

SINTAXIS:

CLOSE [[#] *número-de-canal*] , [#] *número-de-canal*)

PROPOSITO:

Finaliza las operaciones de entrada y salida desde y hacia un archivo en disco, cerrando el canal de comunicación previamente establecido por medio de una instrucción OPEN.

COMENTARIOS:

Si no se especifica número de canal la instrucción CLOSE cierra todos los canales que hubiera abiertos.

El número de canal puede antecederse por el carácter #. Su uso es optativo.

Para cerrar más de un canal se deben indicar todos los números de canal a cerrar separados con comas.

Si en la memoria transitoria (buffer) hay datos, estos son grabados.

El número de canal puede indicarse por medio de una variable o expresión Basic.

EJEMPLO:

```
CLOSE  
CLOSE 1  
CLOSE #1  
CLOSE A+1  
CLOSE 1,3,5  
CLOSE #1,#3,#5
```

NOTAS: Las comandos NEW, CLEAR y END realizan, en forma automática una sentencia CLOSE, cerrando todos los archivos que estuviesen abiertos.

Para consultar sobre el uso de esta sentencia en archivos de cassette o en manejos de pantalla gráfica remítase al manual de MSX BASIC

COPY

SINTAXIS:

COPY especif-de-archivo-origen to especif-de-archivo-destino

PROPOSITO:

Copia uno o varios archivos.

COMENTARIOS:

Cuando se realiza una copia el archivo-origen puede estar en un drive y el destino en otro.

La *especificación-de-archivo-origen* debe indicarse con la precisión necesaria como para identificar él o los archivos que se desea copiar. Pueden utilizarse metacaracteres.

Cuando lo que se desea es copiar uno o varios archivos en otro disco sin cambiarles el nombre bastará con que la *especificación-de-archivo-destino* sea simplemente la mención del drive de destino.

Las *especificaciones-de-archivo* (tanto *origen* como *destino*) pueden ser asignadas como contenido de variables alfanuméricas.

EJEMPLOS:

```
COPY "A:PROG.BAS" TO "B:"  
COPY A$ TO B$  
COPY "PROG.BAS" TO "PROG.BAK"  
COPY "A:PROG.BAS" TO "B:PROG.BAK"
```

NOTAS: La copia de un drive a otro (por ejemplo de A: a B:) es posible aunque sólo se disponga de una diskettera. En este caso El único drive se comportará alternativamente como A: y B:, debiendo el usuario reemplazar los discos cada vez que se lo solicite. Vea la sección Drives Físicos y Lógicos en la Introducción.

La cantidad de reemplazos es menor cuando se usa el comando COPY desde el MSX-DOS, por lo tanto, si desea copiar archivos muy extensos recomendamos hacerlo desde ese entorno.

DSKO\$

SINTAXIS:

DSKO\$ *número-de-drive, número de sector-lógico*

PROPOSITO:

Transfiere información desde la memoria RAM hacia un sector determinado del diskette.

COMENTARIOS:

La dirección inicial de memoria debe estar contenida en las direcciones &HF351 y &HF352.

Los posibles valores de *número-de-drive* son:

- 0 - Para el drive activo.
- 1 - Para el drive A:
- 2 - Para el drive B:

número-de-sector-lógico, comenzando desde 0, puede incluir hasta el máximo sector disponible en el disco (719 para 5 1/4" y 1439 para 3 1/2" doble lado). No se realiza ninguna verificación sobre la validez de este número.

EJEMPLO:

DSKO\$ 0,5

NOTA: Recuerde que la posición inicial de memoria a grabar puede calcularla usando la fórmula:

PEEK (&HF351) + 256 * PEEK (&HF352)

FIELD

SINTAXIS:

FIELD [#] *número-de-canal* , *ancho-del-campo AS variable-alfanumérica*
[, *ancho-del-campo AS variable-alfanumérica*]

PROPOSITO:

Cuando se utilizan archivos de acceso directo, reserva espacio en una memoria transitoria (buffer) para los datos correspondientes a un registro del archivo. Al mismo tiempo permite la definición de la longitud de los campos del registro.

COMENTARIOS:

Es conveniente habituarse a utilizar FIELD como instrucción inmediata a la apertura (OPEN) de un archivo de acceso directo.

La relación entre programa y archivo, cuando este es de acceso directo, se efectúa usando la memoria transitoria (buffer).

Cuando se efectúa una operación de lectura los datos son leídos del diskette y depositados allí (ver GET). Luego son extraídos para su posterior utilización.

Cuando se desea grabar los datos deben depositarse en el buffer (ver LSET y RSET) para luego ser grabados (ver PUT).

numero-de-canal debe indicar el mismo número que se ha empleado en la instrucción OPEN. Puede ir precedido del carácter #.

La secuencia *largo-del-campo AS variable-alfanumérica* se repetirá tantas veces como campos tenga un registro.

Se debe tener la precaución de que la suma de las longitudes de todos los campos definidos no exceda la longitud del registro indicada en la instrucción OPEN. Si así ocurre se producirá un error del tipo FIELD OVERFLOW (sobrepasamiento en los campos).

EJEMPLOS:

```
FIELD 1, 10 AS D1$  
FIELD #2, 20 AS D1$, 30 AS S$, 25 AS A2$
```

NOTA: Si las variables usadas en el comando FIELD son usadas posteriormente en el programa, se pierden las referencias a los campos del registro.

FILES

LFILES

SINTAXIS:

FILES *especificación-de-archivo*

LFILES *especificación-de-archivo*

PROPOSITO:

Muestra los nombres de los archivos existente en un diskette.

COMENTARIOS:

FILES y LFILES son iguales en su función y en su sintaxis con la diferencia que FILES envía la información a la pantalla y LFILES a la impresora.

Si no se indica *especificación-de-archivo* se muestran todos los archivos existentes en el disco.

En todos los casos, cuando no se encuentra información en el disco que responda a lo pedido, o cuando no hay archivos en el disco, se emite el mensaje

FILE NOT FOUND

(archivo no encontrado).

EJEMPLOS:

```
FILES  
FILES "B:"  
FILES "*.BAS"  
LFILES "B:PROG.*"  
LFILES "PRO?.BAS"
```

NOTA: Si se pide FILES de un disco sin formatear se emitirá el mensaje DISK OFFLINE.

Esto mismo ocurre si el diskette no está en el drive o si la puerta de esta permanece abierta.

GET

SINTAXIS:

GET *número-de-canal* [, *número-de-registro*]

PROPOSITO:

Lee un registro de un archivo abierto para acceso directo y lo almacena en el correspondiente buffer.

COMENTARIOS:

número-de-canal debe ser el mismo que se utiliza en las sentencias OPEN y FIELD que se refieren al archivo que se desea leer.

número-de-registro es una expresión numérica que indica el registro a leer. Si se lo omite se leerá el registro siguiente al del GET ejecutado con anterioridad.

El máximo número posible de registros está en función de la capacidad del diskette y el largo de cada registro.

Los campos alfanuméricos que llegan al buffer pueden extraerse de él en forma directa haciendo referencia a las variables declaradas en la instrucción FIELD, mientras que los campos numéricos deben someterse a una conversión (ver funciones CVI, CVS y CVD)

EJEMPLOS:

GET #1,10

GET 1,10

GET 1,N

GET C,T

INPUT #

SINTAXIS:

INPUT # *número-de-canal* , *variable* {, *variable* }

PROPOSITO:

Lee datos desde un archivo secuencial y los asigna a la o las variables indicadas.

COMENTARIOS:

Para utilizar esta sentencia debe abrirse previamente el archivo en modo INPUT (ver OPEN).

número-de-canal debe ser el mismo que se utilizó en el momento de la apertura del archivo.

Las variables a utilizar como receptoras de los datos leídos deben coincidir en su tipo y orden con el de los datos almacenados en el diskette.

EJEMPLO:

```
INPUT #1,A$  
INPUT #3, A,B,C
```

KILL

SINTAXIS:

KILL *especificación-de-archivo*

PROPOSITO:

Borra archivos del diskette.

COMENTARIOS:

El contenido que se le dé a *especificación-de-archivo* determinará si se ha de borrar uno, varios o todos los archivos del diskette.

Debe usarse con prudencia. A diferencia del comando **DEL**, cuando se indica borrar todos los archivos (**KILL "*.*"**) no pide la confirmación (Are you shure? Y/N).

Si se pide borrar un archivo que se encuentra en uso (abierto) generará un error del tipo

File already open

(archivo ya abierto)

LINE INPUT#

SINTAXIS:

LINE INPUT # *número-de-canal*, *variable-alfanumérica*

PROPOSITO:

Lee de un archivo una línea completa de datos (hasta 254 caracteres), sin respetar otra delimitación que el fin de línea o archivo, y los asigna a una variable alfanumérica.

COMENTARIOS:

número-de-canal debe ser el mismo que se indicó en el momento de abrir el archivo

variable-alfanumérica es el nombre de la variable donde se almacenará la línea leída.

La lectura de una línea finaliza cuando en el archivo encuentra un carácter CR (*Carriage Return* - `CHR$(13)`), o la secuencia CR/LF (*Carriage Return + Line Feed* - `(CHR$(13)+CHR$(10))`)

EJEMPLO:

```
10 OPEN "DATOS" FOR INPUT AS 1
20 LINE INPUT #1,A$
30 PRINT A$
40 CLOSE
```

NOTA: Esta sentencia es una ampliación de **LINE INPUT**, que encontrará en el Manual de MSX BASIC.

LOAD

SINTAXIS:

LOAD *especificación-de-archivo* [, R]

PROPOSITO:

Lee un programa Basic del disco y lo trae a la memoria de la computadora.

COMENTARIOS:

especificación-de-archivo debe referirse a un único archivo que contenga un programa BASIC. Puede consultar los nombres por medio de la instrucción FILES (o DIR desde el MSX-DOS).

La opción ",R" ejecuta automáticamente el programa una vez finalizada la carga.

LOAD cierra todos los archivos y borra todas las variables del programa que se encuentren en la memoria.

Por lo contrario, si se utiliza la opción ",R" mientras hay archivos en uso, estos permanecerán abiertos al cargarse el nuevo programa. Esta alternativa puede resultar de interés cuando se desea *encadenar* varios programas.

EJEMPLOS:

```
LOAD "PROGRA",R  
LOAD "PROGRA.BAS",R  
LOAD "PROGRA"
```

LSET

SINTAXIS:

LSET *variable-alfanumérica* = *expresión-alfanumérica*

PROPOSITO:

Mueve datos de variables del programa en uso a la memoria transitoria (buffer) de un archivo.

COMENTARIOS:

En el trabajo con archivos de acceso directo la operación de grabación de datos se divide en dos partes. La primera de ellas consiste en transferir los datos a la memoria transitoria (buffer) por medio de las sentencias LSET o RSET. Una vez que los datos están allí recién puede procederse a la grabación (ver PUT).

La sentencia LSET se usa para instalar en el buffer la información de un solo campo por vez. Si la instrucción FIELD contiene ocho campos diferentes se requerirá, para grabar todos los datos de un registro, el uso de ocho sentencias LSET o RSET.

Con LSET la información se sitúa a partir del extremo izquierdo (Left SET) del espacio reservado para el campo. Análogamente, RSET sitúa la información en el extremo derecho. Si el dato es corto, los espacios restantes se rellenan con blancos. Si por el contrario es muy largo la información se trunca al acabarse el sitio disponible (Como ejemplo: Si el campo tiene una extensión de 6 caracteres y el dato a guardar es COMPUTADORA, sólo se conserva: COMPUT si se usa LSET y TADORA si se usa RSET).

Los datos alfanuméricos se depositan directamente, los numéricos requieren de una conversión previa (ver funciones MKI\$, MKS\$ y MKD\$).

EJEMPLOS:

```
LSET D1$=A$  
LSET D2$=MKI$(N)
```

NOTA: Esta sentencia puede utilizarse aún en situaciones que no impliquen el manejo de disco. Por ejemplo para organizar los datos en las variables de acuerdo a un formato deseado:

```
110 A$=SPACE$(20)  
120 LSET A$=N$
```

*Sí, pero debe hacerse con la
disquetera encendida.*

MAXFILES

SINTAXIS:

MAXFILES = *cantidad-de-archivos*

PROPOSITO:

Indica el número máximo de archivos que pueden abrirse simultáneamente.

COMENTARIOS:

La *cantidad-de-archivos* será cualquier valor entre 0 y 6.

MAXFILES borra el contenido de todas las variables. Se recomienda su uso al comenzar el programa, antes de una instrucción **DIM**.

MAXFILES=0 no permitirá abrir archivos.

MAXFILES se encarga, entre otras cosas, de reservar memoria para los distintos buffers. Cuanto más archivos se indiquen, más memoria consumirá y, como es lógico, menos espacio quedará para programas y variables. Cuando los programas que se utilizan sean muy extensos sea muy estricto en el uso de esta sentencia.

Si un programa no usa esta sentencia, el valor por defecto que adopta la computadora es equivalente a **MAXFILES=1**.

MERGE

SINTAXIS:

MERGE *especificación-de-archivo*

PROPOSITO:

Trae a memoria un programa Basic grabado en ASCII, manteniendo al que ya estaba alojado en memoria e intercalando sus respectivas instrucciones.

COMENTARIOS:

El programa a incorporar debe estar grabado en formato ASCII (ver SAVE *especificación-de-archivo,A*), de no ser así la operación no se realizará.

Cuando ingresa una línea cuyo número ya existe en memoria la que llega reemplaza a la existente. En los demás casos las nuevas líneas se van situando en su correspondiente orden numérico.

EJEMPLO:

MERGE "ARCHIVO"

NAME

SINTAXIS:

NAME *especif-de-archivo-actual* AS *nueva-especif-de-archivo*

PROPOSITO:

Cambia el nombre de uno o varios archivos.

COMENTARIOS:

especificación-de-archivo-actual es el nombre del o los archivos a los que se les quiere cambiar el nombre. Deben existir en el disco.

nueva-especificación-de-archivo es el nombre que se le desea dar. No debe existir en el disco (en caso contrario se produce un error).

EJEMPLO:

NAME "SUELdos" AS "HABERES"

OPEN

SINTAXIS:

OPEN *especif-de-archivo* FOR *modo AS [#] número-de-canal*
OPEN *especif-de-archivo AS [#] número-de-canal [LEN=long-del-registro]*

PROPOSITO:

Abre un canal de comunicación entre el programa y un archivo en diskette. Al mismo tiempo define el tipo de archivo que va a usarse (secuencial o acceso directo).

COMENTARIOS:

El intercambio de información entre un programa y un archivo de datos grabado en diskette se realiza usando una porción de la memoria denominada *buffer*. Allí se depositan los datos que han de ser grabados y los datos que son leídos del diskette.

La comunicación entre el programa (memoria principal), el buffer y el archivo se establece por medio de un *canal* de comunicaciones interno. Este canal se identifica con un número (puede ser 1 a 6 - Ver MAXFILES), lo que simplifica las futuras referencias a operaciones de escritura-lectura que se realicen en el programa (ver GET - PUT - FIELD - INPUT# - LINE INPUT# - PRINT# - PRINT# USING - CLOSE)

El formato utilizado define al mismo tiempo si se trata de un archivo secuencial o de acceso directo.

Veamos estos dos casos por separado.

ARCHIVOS SECUENCIALES

especificación-de-archivo es el nombre del archivo que se desea usar. Si se quiere leer datos, el archivo debe existir. En caso contrario se producirá un error.

modo indica si se desea leer o escribir, y puede ser:

INPUT : Cuando se desea leer datos.

OUTPUT: Cuando se desea escribir un archivo desde el comienzo. Si el archivo existe es borrado automáticamente y sus datos reemplazados por los nuevos que se graben.

APPEND: Para grabar a continuación del último dato existente. Es una forma de extender los archivos.

número-de-canal: Valor entre 1 y 6.

EJEMPLOS:

```
OPEN "DATOS" FOR OUTPUT AS 1  
OPEN "CLIENTES" FOR INPUT AS 2  
OPEN "VENTAS" FOR APPEND AS 1
```

ARCHIVOS DE ACCESO DIRECTO

especificación-de-archivo es el nombre del archivo que se desea utilizar. Si el archivo no existe en el disco es creado automáticamente. La operación de apertura no se interrumpe por ello.

número-de-canal: Valor de 1 a 6

largo-de-registro: Cuando este valor no se define el archivo genera registros de 256 bytes de longitud. Si la longitud a utilizar es menor deberá indicarse por medio de la opción "**LEN=**".

EJEMPLOS:

```
OPEN "DATOS" AS 1  
OPEN "CLIENTES" AS 3 LEN=30
```

NOTA: El uso del comando OPEN para operaciones de cassette o VideoRAM en modos gráficos se explica en el manual de MSX BASIC.

Todo lo relativo al uso de OPEN para operaciones de comunicación se encontrarán en el manual de la RS232C y Modem.

~~PRINT #~~

PRINT # USING

SINTAXIS~~PRINT # número-de-canal, variable (; | ,) variable~~~~PRINT # número-de-canal, USING máscara; variable~~**PROPOSITO:**

Permite la grabación de datos en un archivo secuencial. Usando la opción **USING** los datos se graban de acuerdo al formato indicado.

COMENTARIOS:

número-de-canal debe ser el mismo que se utilizó en la apertura del archivo (ver **OPEN**).

variable contiene el dato a grabar.

Debemos tener presente que la información es enviada al diskette con un formato de iguales características a como **PRINT** envía los datos a pantalla (Ver Manual de MSX-BASIC).

Se pueden incluir separadores formando parte de los datos, que luego sean tenidos en cuenta por la sentencia **INPUT #**.

PRINT # 1,A\$;" ";"B\$

o unir datos grabando comillas al comienzo y el final

PRINT # 1,CHR\$(34);A\$;" ";"B\$;CHR\$(34)

La secuencia usada en la grabación de los datos deberá respetar el orden con que se desea efectuar la posterior lectura de los mismos. Esto incluye la observación del comportamiento no sólo de la instrucción **PRINT #**, sino también el de **INPUT #** y **LINE INPUT #**.

El uso de la opción **USING** permitirá grabar los datos con un formato predeterminado. Para ver su correcta utilización remítase al manual de MSX BASIC en las páginas dedicadas a **PRINT USING**.

EJEMPLOS:

PRINT #1,A\$;B\$

PRINT#1,A\$,B\$

PRINT#1,A\$;" DOMICILIADO EN ";B\$;

PRINT#1,USING "###.##";A

SINTAXIS

Permite la impresión de datos en un espacio secuencia. Usando la opción `USING` se dispone de control de los datos impresos.

PROPOSITO

permite que el comando `PRINT` sea utilizado para imprimir datos de forma directa (ver apartado `PRINT #1`).

COMENTARIOS

Dado que tener buenas dicas de información es fundamental para comenzar a programar en MSX-BASIC.

Se pueden incluir separadores entre los datos, de modo

PRINT #1,A\$;" ";B\$

o más ejemplos siguientes componen el final

PRINT #1,CHR(34);A\$;" ";B\$;CHR(34)

La secuencia `CHR(34)` permite que los datos se impriman sin separación entre ellos.

El uso de la opción `PRINT` permite imprimir los datos con un formato determinado. Para ver su colección utilízate la ayuda de MSX BASIC en las siguientes secciones o pínta una.

PUT

SINTAXIS:

PUT *número-de-canal* [, *número-de-registro*]

PROPOSITO:

Graba un registro de un archivo de acceso directo tomando los datos del correspondiente buffer.

COMENTARIOS:

número-de-canal debe ser el mismo que se utiliza en las sentencias **OPEN** y **FIELD** que se refieren al archivo que se desea grabar.

número-de-registro indica el número de registro a grabar. Si se lo omite se grabará el registro siguiente al del último **PUT** ejecutado.

El máximo número posible de registros que pueden grabarse dependen de la capacidad del diskette y la extensión de cada registro.

Los datos que se desean grabar deben estar previamente instalados en el buffer (ver **LSET** - **RSET** - **MKI\$** - **MKS\$** - **MKD\$**)

EJEMPLOS:

PUT #1,10
PUT 1,10
PUT 1,N
PUT C,T

RSET

SINTAXIS:

RSET *variable-alfanumérica* = *expresión-alfanumérica*

PROPOSITO: Mueve el contenido de las variables a la memoria transitoria (buffer) asociada a un archivo de acceso directo.

COMENTARIOS:

En el trabajo con archivos de acceso directo la operación de grabación de datos se divide en dos partes. La primera de ellas consiste en depositar los datos que se deben grabar en el buffer por medio de esta sentencia o su semejante LSET. Una vez que los datos están allí recién puede procederse a la grabación (ver PUT).

La sentencia RSET se usa para instalar en el buffer la información de un solo campo por vez. Por lo tanto si en la instrucción FIELD se han usado ocho campos diferentes se requerirá, para grabar todos los datos de un registro, el uso de ocho sentencias LSET O RSET.

Con RSET los datos se sitúa a partir del extremo derecho (Right SET) del espacio reservado para el campo. Si los datos no alcanzan a llenar el campo, los espacios restantes se rellenan con blancos. Si por el contrario, los datos superan la longitud del campo, se trunca (Como ejemplo: Si el campo tiene una extensión de 6 caracteres y el dato a guardar es COMPUTADORA, sólo se conserva: TADORA).

Los datos alfanuméricos se depositan directamente, los numéricos requieren de una conversión previa (ver funciones MKI\$, MKS\$ y MKD\$).

EJEMPLOS:

```
RSET D1$=A$  
RSET D2$=MKI$(N)
```

NOTA: Esta sentencia puede utilizarse aún en situaciones que no impliquen el manejo de archivos, por ejemplo, para organizar los datos en las variables de acuerdo a un formato deseado:

```
110 A$=SPACE$(20)  
120 RSET A$=N$
```

SINTAXIS:

run especiifica la ejecución de archivo (r)

COMENTARIOS:

La especiificación de archivo debe corresponder a un nombre básico

existente en el disco.

EJEMPLOS:

run "PROGRAMA"
run "PROGRAMA".B

RUN

EJEMPLOS

SINTAXIS:

RUN *especificación-de-archivo* [,R]

PROPOSITO:

Carga un programa Basic grabado en diskette y lo ejecuta.

COMENTARIOS:

La *especificación-de-archivo* debe corresponder a un programa Basic existente en el disco.

RUN equivale a LOAD con la opción ",R"

RUN cierra todos los archivos que pudieran estar abiertos, salvo que se utilice la opción ",R".

EJEMPLOS:

RUN "PROGRAM"

RUN "PROGRAM",R

~~SAVE~~

SINTAXIS

SAVE *especificación-de-archivo* [,A]

PROPOSITO

Graba en diskette un programa Basic residente en memoria.

COMENTARIOS:

Cuando la *especificación-de-archivo* que se indica ya existe en el diskette, su contenido es reemplazado por el programa cargado en memoria.

La opción ",A" debe colocarse cada vez que se desea que la grabación se realice en formato ASCII. Si no se la utiliza el programa se graba en un formato comprimido (conocido como tokenizado) que ocupa menos lugar en el diskette y disminuye el tiempo de carga.

La grabación en formato ASCII es obligatoria cuando se pretende cargar posteriormente el programa empleando el comando **MERGE**.

EJEMPLOS:

SAVE "PROG"

SAVE "PROG",A

FUNCIONES DEL MSX DISK BASIC

Las funciones intrínsecas que el MSX DISK BASIC pone a disposición del usuario pueden ser invocadas desde cualquier programa sin necesidad de definirlas previamente.

Las funciones del MSX BASIC no son autosuficientes; requieren de un comando que las llame (PRINT - LSET, etc). En caso contrario se producirá un error de sintaxis.

Las funciones devuelven un valor en base a un argumento que se indica entre paréntesis.

Las funciones del MSX Disk BASIC son:

CVD
CVI
CVS
DSKF
DSKI\$
EOF
INPUT\$
LOC
LOF
MKD\$
MKI\$
MKS\$
VARPTR

CVI
CVS
CVD

SINTAXIS:

cvI(*variable-alfanumérica*)
cvs(*variable-alfanumérica*)
cvD(*variable-alfanumérica*)

PROPOSITO:

Transforman en valores numéricos los datos que están en formato condensado.

COMENTARIOS:

En archivos de acceso directo los datos numéricos deben ser almacenados en el buffer como alfanuméricos (ver más adelante las funciones MKI\$, MKSS\$ y MKDS\$). Estas tres funciones se utilizan para reconvertir los datos al formato numérico, de acuerdo al tipo de la variable original.

Recuerde los tipos y sus largos.

Entero: Ocupa dos caracteres.

Simple precisión: Ocupa cuatro caracteres.

Doble precisión: Ocupa ocho caracteres.

A cada formato le corresponde una función.

Formato entero: cvI (ConVert Integer)

Formato Simple Precisión: cvs (ConVert Single)

Formato Doble Precisión: cvD (ConVert Doubble)

Cada una de estas funciones lleva como argumento el nombre con que se ha definido la variable en la instrucción FIELD.

EJEMPLOS:

A=CVI(D1\$)

B=CVS(D2\$)

C=CVD(D3\$)

NOTA: Las funciones que realizan la operación inversa son: MKI\$, MKS\$ y MKD\$.

DSKF

SINTAXIS:

DSKF (número-de-drive)

PROPOSITO:

Informa la cantidad de Kbytes disponibles en el disco.

COMENTARIOS:

número-de-drive debe contener

- 0 para el drive activo.
- 1 para el drive A.
- 2 para el drive B.

EJEMPLO:

PRINT DSKF(1)

DSKI\$

SINTAXIS:

DSKI\$ (*número-de-drive, número-de-sector-lógico*)

PROPOSITO:

Transfiere información de un sector lógico del diskette hacia la memoria RAM.

COMENTARIOS:

La dirección inicial de memoria se señala en las posiciones &HF351 y &HF352. Puede calcular la dirección usando la fórmula:

`PEEK(&HF351)+256*PEEK(&HF352)`

número-de-drive debe valer:

- 0 - Para el drive activo.
- 1 - Para el drive A
- 2 - Para el drive B

número-de-sector-lógico comenzando desde 0 puede incluir hasta el máximo sector disponible en el disco (719 para 5 1/4" y 1439 para 3 1/2 doble lado). No se realiza ninguna verificación sobre la validez de este número.

EJEMPLO:

`A$=DSKI$(0,5)`

NOTA: Si bien esta función no retorna un valor, requiere de un comando previo. En el ejemplo dado la variable A\$ actúa como una variable ficticia. Los datos leídos quedan en la dirección apuntada por las posiciones &HF351 y &HF352.

EOF

SINTAXIS:

EOF (*número-de-canal*)

PROPOSITO:

Es una función lógica que verifica si se ha leído el final de un archivo secuencial.

COMENTARIOS:

Por tratarse de una función lógica, devuelve dos posibles valores:

- 1: (cierto) cuando se ha leído el fin del archivo.
- 0: (falso) en caso contrario.

número-de-canal debe ser el mismo que el usado en la apertura del archivo (ver OPEN).

EJEMPLO:

Este programa lee e imprime en pantalla el contenido de un archivo llamado DATOS.

```
10 OPEN "DATOS" FOR INPUT AS #1
20 INPUT #1, A$
30 PRINT A$
40 IF EOF(1)= -1 THEN END
50 GOTO 20
```

La línea 40 puede ser también:

```
40 IF EOF(1) THEN END
```

INPUT\$

SINTAXIS:

INPUT\$ (*cantidad-de-caracteres, número-de-canal*)

PROPOSITO:

Lee de un archivo especificado por *número-de-canal*, la *cantidad-de-caracteres*.

COMENTARIOS:

cantidad-de-caracteres indica cuantos caracteres se desean leer.

número-de-canal debe ser el mismo que se utilizó en la apertura del archivo (ver OPEN).

Los caracteres son leídos secuencialmente, siguiendo el orden en que han sido grabados. La utilización de esta función puede ser adecuada para efectuar análisis de la información contenida en un archivo.

EJEMPLO:

```
10 OPEN "DATOS" FOR INPUT AS 1
20 IF EOF(1) THEN 100
30 PRINT INPUT$(1,1);
40 GOTO 20
100 CLOSE
```

NOTA: Esta función guarda relación con su homónima para lectura de teclado, de la que hallará información en el manual de MSX BASIC.

LOC

SINTAXIS:

LOC (número-de-canal)

PROPOSITO:

Indica el número del último registro que se ha leído o grabado en un archivo de acceso directo.

COMENTARIOS:

número-de-canal debe ser el mismo que se utilizó en la apertura del archivo (ver OPEN).

Si el archivo ha sido abierto, pero no se ha ejecutado ninguna lectura o grabación el valor que devuelve LOC es cero.

EJEMPLO:

```
200 IF LOC(1)>50 THEN STOP
```

LOF

SINTAXIS:

LOF (*número-de-canal*)

PROPOSITO:

Devuelve la cantidad de caracteres de un archivo.

COMENTARIOS:

número-de-canal debe ser el mismo que se utilizó en la apertura del archivo (ver OPEN).

Esta función puede usarse para controlar la extensión de un archivo, la capacidad del diskette, o calcular la cantidad de registros en un archivo de acceso directo.

EJEMPLO:

200 PRINT LOF(1)

310 ULT=LOF(1)/128

MKI\$
MKSS\$
MKD\$

SINTAXIS:

MKI\$ (*variable-numérica*)
MKSS\$ (*variable-numérica*)
MKD\$ (*variable-numérica*)

PROPOSITO:

Transforma variables numéricas a formato alfanumérico para poder ser grabadas en archivos de acceso directo.

COMENTARIOS:

Existen tres clases de formato, que se corresponden directamente con las tres clases de variables numéricas que maneja el MSX BASIC.

Entero: ocupa dos caracteres.

Simple precisión: Ocupa cuatro caracteres.

Doble precisión: Ocupa ocho caracteres.

Por lo tanto, cada dato numérico que deba ser enviado al diskette debe ser transformado a uno de estos formatos mediante las respectivas funciones:

Formato entero: **MKI\$** (MaKe Integer)

Formato Simple Precisión: **MKSS\$** (MaKe Single)

Formato Doble Precisión: **MKD\$** (MaKe Double)

variable-numérica debe contener el dato a transformar.

EJEMPLOS:

```
LSET D1$=MKI$(A)
LSET D2$=MKSS$(B)
RSET D3$=MKD$(C)
```

NOTA: Las funciones que realizan la operación inversa son: **CVI**, **CVS** y **CVD**.

VARPTR

SINTAXIS:

VARPTR ([#] *número-de-canal*)

PROPOSITO:

Devuelve la dirección de memoria donde se encuentra el bloque de control del archivo cuyo número se indica.

COMENTARIOS:

número-de-canal debe ser el mismo que se uso en la apertura del archivo (ver OPEN)

EJEMPLOS:

100 X=VARPTR(#1)

APENDICES

CARACTERISTICAS TECNICAS

El cartucho TDC-600 no requiere alimentación externa.

Dispone de un conector de 34 pines para los drives.

El siguiente diagrama ilustra la distribución de señales en cada uno de ellos:

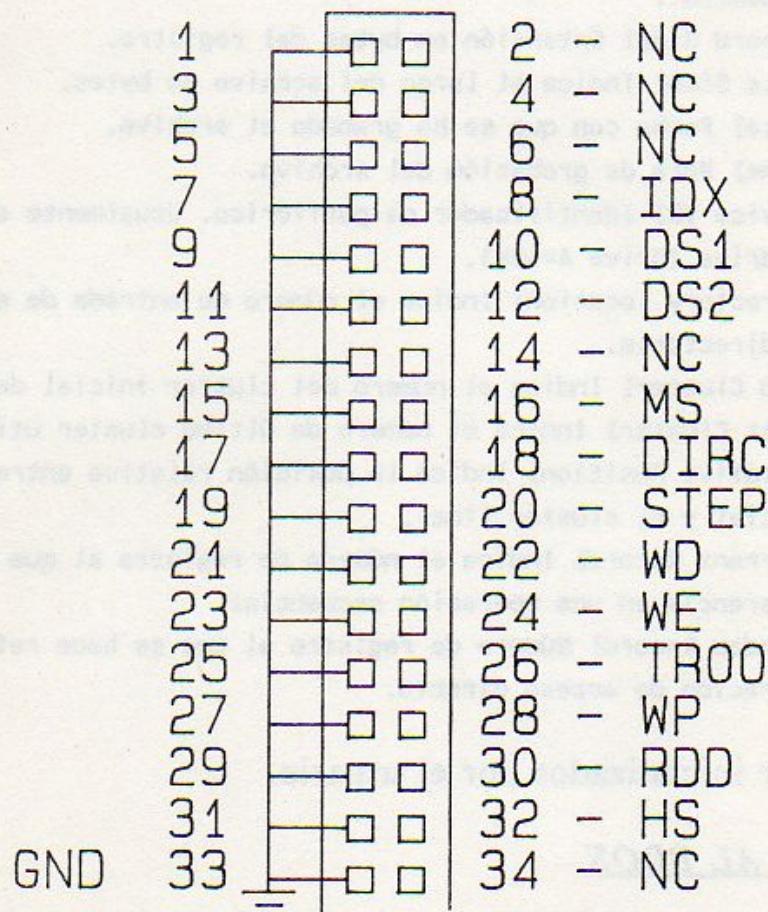


Diagrama de Pines

NOTAS PARA PROGRAMADORES

DESCRIPCION DEL FCB (FILE CONTROL BLOCK)

BYTE/S	SIGNIFICADO
* 00H:	Número de drive [0=default - 1=A - 2=B]
* 01H-08H:	Nombre del archivo.
* 09H-0BH:	Extensión del nombre.
* 0CH-0DH:	[Current Block]. Bloque al que se hace referencia en acceso secuencial.
* 0EH-0FH:	[Record Size] Extensión en bytes del registro.
10H-13H:	[File Size] Indica el largo del archivo en bytes.
14H-15H:	[Date] Fecha con que se ha grabado el archivo.
16H-17H:	[Time] Hora de grabación del archivo.
18H:	[Device ID] Identificador de periférico. Usualmente es 40H + número de drive (Drive A=40H).
19H:	[directory location] Indica el número de entrada de este archivo en el directorio.
1AH-1BH:	[Top Cluster] Indica el número del cluster inicial del archivo.
1CH-1DH:	[Last Cluster] Indica el número de último cluster utilizado.
1EH-1FH:	[Relative Position] Indica la posición relativa entre el cluster inicial y el cluster final.
* 20H:	[Current Record] Indica el número de registro al que se hace referencia en una operación secuencial.
* 21H-24H:	[Random Record] Número de registro al que se hace referencia en una operación de acceso directo.

(*) Deben ser inicializados por el usuario.

LLAMADAS AL BDOS

El valor (FUNCION) debe colocarse en el registro C antes de efectuar la llamada al sistema, mediante un CALL a la F37DH en MSX-BASIC ó 5h, en el entorno MSX-DOS.

FUNCION: 00H

NOMBRE: System Reset

REQUIERE: Nada

DEVUELVE: Nada

ACLARACION: Devuelve el control al COMMAND.COM

FUNCION: 01H

NOMBRE: Console Input.

REQUIERE: Nada

DEVUELVE: A-caracter ingresado

ACLARACION: Ingresa un caracter de teclado. Si el buffer está vacío, espera.

FUNCION: 02H

NOMBRE: Console Output.

REQUIERE: E-Caracter e enviar

DEVUELVE: Nada

ACLARACION: Envía un caracter a pantalla.

FUNCION: 03H

NOMBRE: External Input.

REQUIERE: Nada

DEVUELVE: A-Caracter ingresado

ACLARACION: Recibe un caracter del dispositivo AUX.

FUNCION: 04H

NOMBRE: External Output.

REQUIERE: E-Caracter a enviar

DEVUELVE: Nada

ACLARACION: Envía un caracter por el dispositivo AUX.

FUNCION: 05H

NOMBRE: Printer Output

REQUIERE: E-Caracter a imprimir

DEVUELVE: Nada

ACLARACION: Envía un caracter a la impresora.

FUNCION: 06H

NOMBRE: String Output

REQUIERE: DE-Dirección inicial del string.

FUNCION: 07H

NOMBRE: Direct Console Input - 1

REQUIERE: Nada

DEVUELVE: A-Caracter recibido de consola

ACLARACION: Recibe un caracter de la consola.

FUNCION: 08H

NOMBRE: Direct Console Input - 2

REQUIERE: Nada

DEVUELVE: A-Caracter recibido.

ACLARACION: Recibe un caracter de consola sin generar eco. Chequea ^C, ^P y ^N.

FUNCION: 09H

NOMBRE: String Output

REQUIERE: DE-Dirección de inicio del string

DEVUELVE: Nada

ACLARACION: Imprime en pantalla una string terminada con el caracter "\$".

FUNCION: 0AH

NOMBRE: String Input.

REQUIERE: DE - Dirección del buffer de recepción, (DE) - Cantidad máxima de caracteres a ingresar.

DEVUELVE: (DE+1)- Cantidad de bytes ingresados. (DE+2) y siguientes, caracteres ingresados.

ACLARACION: Ingresa una string desde teclado hasta que se presiona <RETURN> o se llega a la longitud máxima indicada en (DE).

FUNCION: 0BH

NOMBRE: Console status check

REQUIERE: Nada

DEVUELVE: A=0FFH cuando una tecla ha sido presionada.

A=00H en caso contrario.

ACLARACION: Controla el estado del teclado.

FUNCION: 0CH

NOMBRE: Versión number adquisition.

REQUIERE: Nada

DEVUELVE: HL=0022H

ACLARACION: Esta llamada es usada para obtener el número de version en CP/M. En MSX DOS indica siempre 0022H

FUNCION: 0DH

NOMBRE: Disk Reset

REQUIERE: Nada

DEVUELVE: Nada

ACLARACION: Esta función graba los buffers internos, setea el disco por defecto como A y la DMA en 80H.

FUNCION: 0EH

NOMBRE: Default Drive Setting

REQUIERE: E-Número de drive (A=00H, B=01H)

DEVUELVE: Nada

ACLARACION: Permite especificar el disco activo por defecto.

FUNCION: 0FH

NOMBRE: Opening Files.

REQUIERE: DE-Dirección inicial en RAM del FCB

DEVUELVE: A=0 si el archivo fue abierto.

A=OFFH en caso de error.

ACLARACION: Abre un archivo cuyos datos se encuentran en RAM en el FCB. Si la apertura se realiza el sistema operativo completa los datos.

FUNCION: 10H

NOMBRE: Closing Files

REQUIERE: DE-Dirección inicial del FCB.

DEVUELVE: A=00H si fue cerrado. A=OFFH en caso de error.

ACLARACION: Cierra un archivo.

FUNCION: 11H

NOMBRE: File search 1

REQUIERE: DE-Dirección inicial de un FCB.

DEVUELVE: A=00H si se encontró el archivo. A=OFFH en caso contrario. De hallarse, en la DMA habrá una copia de la entrada de directorio (32 bytes).

ACLARACION: Permite buscar un archivo en el directorio. Puede usarse el metacaracter "?" para componer nombre o extensión. Finaliza la búsqueda cuando encuentra el primero que cumpla la condición.

FUNCION: 12H

NOMBRE: File search 2

REQUIERE: Nada

DEVUELVE: Igual que función 11H

ACLARACION: Luego de ejecutada la función 11H, se puede continuar la búsqueda de sucesivos archivos reiterando la llamada a esta función.

FUNCION: 13H

NOMBRE: Deleting Files

REQUIERE: DE-Dirección inicial de un FCB abierto.

DEVUELVE: A=00H si fue borrado. A=OFFH en caso contrario.

ACLARACION: Borra un archivo del directorio.

FUNCION: 14H

NOMBRE: Sequential Readout.

REQUIERE: DE-Dirección inicial de un FCB. Deben especificarse los campos Current Block y Current Record

DEVUELVE: A=00H Cuando se efectuó la lectura . A=01H en caso contrario. Registro en DMA.

ACLARACION: Lee secuencialmente un archivo. Para sucesivas lecturas usando esta función los campos Current Block y Current Record se actualizan automáticamente.

FUNCION: 15H

NOMBRE: Sequential writing

REQUIERE: DE-Dirección inicial de un FCB abierto. FCB [current block] Block inicial a escribir. Deben especificarse los campos Current Block y Current Record.

DEVUELVE: A=00H si se efectuó la grabación. A=01H en caso contrario.

ACLARACION: Graba secuencialmente un archivo. Para sucesivas grabaciones usando esta función los campos Current Block y Current Record se actualizan automáticamente. No deben ser modificados.

FUNCION: 16H

NOMBRE: Creating Files

REQUIERE: DE-Dirección inicial de un FCB.

DEVUELVE: A=00H si fue creado. A=OFF en caso de error.

ACLARACION: Crea una nueva entrada de directorio.

FUNCION: 17H

NOMBRE: Renaming Files

REQUIERE: DE-Dirección inicial de un FCB no abierto. El nuevo nombre debe estar en el mismo FCB a partir del byte 18 (dec).

DEVUELVE: A=00H si el cambio fue realizado. A=OFFH en caso contrario.

ACLARACION: Cambia el nombre a un archivo. Puede usarse el metacaracter "?" para componer el nombre.

FUNCION: 18H

NOMBRE: Login Vector Adquisition

REQUIERE: Nada

DEVUELVE: HL-Información de discos en línea. Los bits de L indican si está o no en línea cada drive [bit0=A: bit1=B]

ACLARACION: Permite obtener información de la cantidad de drives lógicos que se hallan en línea.

FUNCION: 19H

NOMBRE: Defaul Drive Adquisition.

REQUIERE: Nada

DEVUELVE: A-Número de drive [00H=A: 01H=B:]

ACLARACION: Permite obtener el número de drive activo.

FUNCION: 1AH

NOMBRE: Setting DMA.

REQUIERE: DE-Dirección incial de DMA

DEVUELVE: Nada

ACLARACION: Permite fijar una nueva dirección de DMA.

FUNCION: 1BH

NOMBRE: Disk Information Adquisition

REQUIERE: E-Número de drive [00H=activo 01H=A: 02H=B:]

DEVUELVE: A-Número de sectores lógicos por cluster. BC-longitud de sector lógico. DE-cantidad de clusters. HL-cantidad de clusters no usados. IY-Dirección de inicio del DPB. IY-Dirección de inicio de la FAT en memoria.

ACLARACION: Permite obtener información de un diskette.

FUNCIONES: 1CH-20H

NOMBRE: No disponibles

FUNCION: 21H

NOMBRE: Random reading.

REQUIERE: DE-Dirección de inicio del FCB. El campo Random Record indica el número de registro a leer.

DEVUELVE: A=00H Si se efectuó la lectura.

A=01H en caso de error. Datos leídos en DMA.

ACLARACION: Realiza la lectura de un registro de un archivo ya abierto.

FUNCION: 22H

NOMBRE: Random Writing.

REQUIERE: DE-Dirección de inicio del FCB en RAM. El campo Random Record especifica el número de registro a grabar. En DMA debe haber 128 bytes de información.

DEVUELVE: A=00H si se efectuó la escritura.

A=01H en caso de error.

ACLARACION: Graba un registro de información en archivo de acceso directo ya abierto.

FUNCION: 23H

NOMBRE: File Size Acquisition

REQUIERE: DE-Dirección inicial de un FCB abierto.

DEVUELVE: A=00H si se pudo efectuar. A=OFFH en caso contrario. La longitud queda especificada en los tres primeros bytes del campo Random Record.

ACLARACION: Permite obtener la longitud de un archivo expresada en unidades de a 128 bytes cada una.

FUNCION: 24H

NOMBRE: Random Record Field Setting.

REQUIERE: DE-Dirección inicial de un FCB abierto. Campos Current Block y Current Record.

DEVUELVE: El valor calculado en el campo Random Record.

FUNCION: 25H

NOMBRE: No disponible

FUNCION: 26H

NOMBRE: Random Writing 2

REQUIERE: DE-Dirección inicial del FCB. Record Size y Random Record. HL-Número de registros a grabar. DMA-Datos a grabar.

DEVUELVE: A=00H Si se efectuó la grabación.

A=01H en caso de error.

ACLARACION: Realiza la grabación de varios registros. No es compatible con CP/M.

FUNCION: 27H

NOMBRE: Random Readout 2

REQUIERE: DE-Dirección inicial del FCB. Record Size. Random Record. HL-Cantidad de registros a leer.

DEVUELVE: A=00 Si se efectuó la lectura. A=01H en caso contrario. DMA-Los datos leídos.

ACLARACION: Realiza la lectura de varios registros. No es compatible con CP/M.

FUNCION: 28H

NOMBRE: Random writing 3

REQUIERE: DE-Dirección inicial del FCB. Random Record. DMA 128 bytes de información.

DEVUELVE: A=00H Si se efectuó la grabación. A=01H en caso contrario.

ACLARACION: Realiza la grabación de un registro.

FUNCION: 29H

NOMBRE: No disponible

FUNCION: 2AH

NOMBRE: Date adquisition.

REQUIERE: Nada

DEVUELVE: HL-Año. E-Día del mes. D-Mes. A-Día de la semana

ACLARACION: Obtiene la fecha corriente. No es compatible con CP/M.

FUNCION: 2BH

NOMBRE: Date setting.

REQUIERE: HL-Año. E-Día del mes. D-Mes.

DEVUELVE: A=00H Si se pudo transferir la fecha.

A=OFFH en caso de error.

ACLARACION: Informa al sistema operativo de una nueva fecha. No es compatible

FUNCION: 2CH

NOMBRE: Time adquisition.

REQUIERE: Nada

DEVUELVE: H-Hora. L-Minutos. D-Segundos. E-Centésimas.

ACLARACION: Permite obtener la hora actual. No es compatible con CP/M.

FUNCION: 2DH

NOMBRE: Time Setting.

REQUIERE: H-Hora. L-Minutos. D-Segundos. E-Centésimas.

DEVUELVE: Permite indicar una nueva hora. No compatible con CP/M.

ACLARACION:

FUNCION: 2EH

NOMBRE: Verify Flag Setting.

REQUIERE: E=00H para anular la bandera de verificación. E<>00H Para habilitar la bandera de verificación.

DEVUELVE: Nada.

ACLARACION: Según el valor del registro E equivale a VERIFY OFF o VERIFY ON. No compatible con CP/M.

FUNCION: 2FH

NOMBRE: Reading Logical sector.

REQUIERE: DE-Primer sector lógico a leer. H-Cantidad de sectores a leer. L-Drive donde leer [A:=00H B:=01H].

DEVUELVE: Datos en DMA.

ACLARACION: Lee uno o varios sectores lógicos del disco.

FUNCION: 30H

NOMBRE: Writing Logical Sector.

REQUIERE: DE-Primer sector a escribir. H-Cantidad de sectores a escribir. L-Drive donde escribir [A:=00H B:=01H]. Datos en DMA.

DEVUELVE: Nada

ACLARACION: Graba uno o varios sectores.

Cualquier consulta adicional donde se requiera información sobre la organización de slots, funcionamiento de su computadora MSX, así como puntos de entrada a rutinas de ROM, deberá referirse a:

MODIFICACION EN LA VELOCIDAD DE MOTOR PASO A PASO

El mecanismo controlador de discos realiza la lectura/escritura a través de un cabezal que está en contacto con la superficie del disco. Este cabezal se mueve accionado por un motor paso a paso (stepper motor).

Estos motores se controlan por pulsos eléctricos; a cada pulso le corresponde una cantidad de grados fijos de giro.

El tiempo de duración de cada pulso depende del tipo de motor que contiene el mecanismo controlador de discos. Este tiempo no puede ser menor que el especificado.

Debido a esto, en nuestro cartucho controlador de discos TDC-600 se ha fijado el tiempo máximo entre pulsos para contemplar el peor de los casos.

Sin embargo, para aquellos usuarios que disponen de un mecanismo que sea más rápido, está la posibilidad de modificar por software este tiempo entre pulsos, que deberá ser actualizado cada vez que se enciende la computadora.

Este tiempo entre pulsos varía desde 1 milisegundo a 15 milisegundos. Deberá verificar en el dato técnico del mecanismo controlador el tiempo mínimo entre pulsos (step rate).

A continuación damos un ejemplo de software para modificar el step rate: Este listado fuente, que llamaremos STEP.MAC, puede ser editado, luego ensamblado y linkeditado con M80.COM y L80.COM para generar el archivo STEP.COM. Este podrá ser incluido dentro del AUTOEXEC.BAT para que sea invocado cada vez que se enciende la computadora.

MSX.M-80 1.00 01-Apr-85 PAGE 1

```
.z80
.comment%
```

Este programa cambia la velocidad del motor
paso a paso de los drives controlados por

Los parámetros externos a definir son: El tiempo entre pulsos (de 1 a 15 milisegundos como 1 a 15 respectivamente), y la dirección del slot en donde está insertado el cartucho controlador. Estos parámetros se definen en los DB ubicados al pie del programa.

HUT es el tiempo de espera antes de levantar el cabezal de lectura y HLT es el tiempo de espera antes de bajar el cabezal. Estos parámetros se utilizan en aquellos drives que tiene forma de subir o bajar el cabezal de lectura/grabación. En este caso HLT está al mínimo y HUT está al máximo. HLT va de 1 a 127 y HUT va de 1 a 15.

%

000C	rdslt	equ	000ch	
0014	wrslt	equ	0014h	
8000	sts_reg	equ	8000h	
8001	dta_reg	equ	8001h	
9000	op_reg	equ	9000h	
000F	HUT	equ	0fh	;Head Unload Time
0001	HLT	equ	01h	;Head Load Time
0000' F3		di		
0001' 3E 03	ld	a,03h		;comando specify
0003' CD 0021'	call	wrtdrv		
0006' 3A 003D'	ld	a,(STEP_TIME)		
0009' ED 44	neg			
000B' E6 0F	and	0fh		
000D' OF	rrca			
000E' OF	rrca			
000F' OF	rrca			
0010' OF	rrca			
0011' 05 05	ld			;HUT

```

0013' B1          or      c
0014' CD 0021'    call    wrtdrv
0017' 3E 01       ld      a,HLT
0019' 0F          rrca
001A' F6 01       or      1
001C' CD 0021'    call    wrtdrv

001F' FB          ei
0020' C9          ret

```

;Esta subrutina escribe un byte en el
;controlador WD37C65. Antes de escribir,
;verifica si está listo para recibir datos.

```

0021'             wrtdrv:
0021' F5          push   af
0022' 3A 003E'    loop:  ld     a,(DRVSLT)
0025' 21 8000     ld     hl,sts_reg
0028' CD 000C     call   rdslt
002B' E6 C0       and    11000000b
002D' FE 80       cp     80h
002F' 20 F1       jr     nz,loop
0031' F1          pop    af
0032' 5F          ld     e,a
0033' 3A 003E'    ld     a,(DRVSLT)
0036' 21 8001     ld     hl,dta_reg
0039' CD 0014     call   wrslt
003C' C9          ret

```

;Estos son los valores que puede modificar el
;usuario, si hace falta.

```

003D' 0F          STEP_TIME: DB 15 ;Este valor es default
003E' 02          DRVSLT:    DB 02 ;Este es el slot del cartucho
                                ;para la linea Talent-MSX.

```

EN CASO DE FALLA...

SINTOMA

PROCEDIMIENTO

- | | |
|--|---|
| La computadora no reconoce el/los drives | Apague el equipo y reinstale el cartucho. Si la falla persiste al reencender la computadora puede deberse a una falla en el cartucho. Acuda a Servicio Técnico. |
| No carga el sistema operativo. | Reemplace el diskette por uno que contenga el Sistema Operativo y encienda nuevamente el equipo. |
| En una configuración con dos drives, no reconoce el B: | Revisar la conexión y encender el equipo teniendo la precaución de no pulsar la tecla <Ctrl> hasta que el sistema operativo sea cargado. |

MSX-DOS
GUIA DE REFERENCIA RAPIDA

BASIC [*especificación-de-archivo*]

COPY *especif-de-archivo-origen* [*especif-de-archivo-destino*]

DATE [*mm-dd-aa*]

DEL *especificación-de-archivo*

DIR [*especificación-de-archivo*] [/P] [/W]

FORMAT

MODE *ancho*

PAUSE [*comentario*]

REM [*comentario*]

REN *especificación-de-archivo* *nueva-especificación-de-archivo*

TIME [*hh [:mm [:ss]]*]

TYPE *especificación-de-archivo*

MSX DISK BASIC GUIA DE REFERENCIA RAPIDA

■ COMANDOS

BLOAD *especificación-de-archivo* [,OFFSET] [,R]
BLOAD *especificación-de-archivo* [,OFFSET] [,S]

BSAVE *especificación-de-archivo* ,*inicio* ,*fin* [,ejecución]
BSAVE *especificación-de-archivo* ,*inicio* ,*fin* [,S]

CALL FORMAT
_FORMAT

CALL SYSTEM
_SYSTEM

CLOSE [[#] *número-de-canal*] <, [#] *número-de-canal* >

COPY *especif-de-archivo-origen* TO *especif-de-archivo-destino*

DSKO\$ *número-de-drive*, *número-de-sector-lógico*

FIELD [#] *número-de-canal* , *ancho-del-campo* AS *variable-alfanumérica*
< , *ancho-del-campo* AS *variable-alfanumérica* >

FILES [*especificación-de-archivo*]
LFILES [*especificación-de-archivo*]

GET *número-de-canal* [, *número-de-registro*]

INPUT# *número-de-canal* , *variable* <, *variable* >

KILL *especificación-de-archivo*

LINE INPUT # *número-de-canal* , *variable alfanumérica*

LOAD *especificación-de-archivo* [, R]

LSET *variable-alfanumérica* = *expresión-alfanumérica*

MAXFILES = *cantidad-de-archivos*

MERGE *especificación-de-archivo*

NAME *especif-de-archivo-actual* AS *nuevo-especif-de-archivo*

OPEN *especif-de-archivo* FOR *modo* AS [#] *número-de-canal*

OPEN *especif-de-archivo* AS [#] *número-de-canal* [LEN=*long-del-registro*]

PRINT# *número-de-canal* , *variable* {; [*variable*] } , [*variable*] }

PRINT# *número-de-canal* , USING *formato-de-using*, *variable*

PUT *número-de-canal* [, *número-de-registro*]

RSET *variable alfanumérica* = *expresión alfanumérica*

RUN *especificación-de-archivo*

SAVE *especificación-de-archivo* [,A]

■ FUNCIONES

CVI (*variable-alfanumérica*)

CVS (*variable-alfanumérica*)

CVD (*variable-alfanumérica*)

DSKF (*número-de-drive*)

DSKI\$ (*número-de-drive , número-de-sector-lógico*)

EOF (*número-de-canal*)

INPUT\$ (*cantidad-de-caracteres,número-de-canal*)

LOC (*número-de-canal*)

LOF (*número-de-canal*)

MKI\$ (*variable-numérica*)

MKS\$ (*variable-numérica*)

MKD\$ (*variable-numérica*)

VARPTR ([#] *número-de-canal*)

CODIGOS Y MENSAJES DE ERROR DE MSX DISK BASIC

NRO/MENSAJE	ERROR
50 Field Overflow	Un comando FIELD intentó colocar más bytes de los especificados como largo del registro.
51 Internal Error	Ha ocurrido un mal funcionamiento interno. Repórtele a Telemática S.A. detallando las condiciones en que se produjo.
52 Bad File Number	Alguna instrucción se refiere a un archivo no abierto o fuera del rango especificado en la inicialización.
53 File not found	Un comando LOAD, KILL u OPEN hace referencia a un archivo que no existe en el drive indicado.
54 File already open	Una sentencia OPEN en modo OUTPUT o un comando KILL mencionan un archivo que está actualmente abierto.
55 Input past end	Una sentencia INPUT se ejecuta después de haber leído todo el archivo, o sobre un archivo vacío. Prevenga este error controlando el fin de archivo mediante la función EOF.
56 Bad File Name	Se nombra un archivo en forma ilegal (por ej: demasiados caracteres, o caracteres no válidos).
57 Direct statement in file	Un comando directo se encontró al cargar un programa BASIC en formato ASCII. La sentencia LOAD terminó de ejecutarse en ese momento.
58 Sequential I/O only	Un comando GET ó PUT se usó para un archivo secuencial.

- 59 File not open Intentó ejecutarse un comando de lectura o escritura sobre un archivo no abierto.
- 60 Bad allocation table El diskette no está formateado.
- 61 Bad file mode Se intentó usar PUT, GET ó LOF con un archivo secuencial, o LOAD con un archivo de acceso directo, o ejecutar un comando OPEN con un modo diferente de INPUT, OUTPUT ó APPEND.
- 62 Bad drive name Se especificó un drive inválido.
- 64 File still open El archivo permanece abierto.
- 65 File already exists La especificación de archivo en un comando NAME es idéntica a la de otro archivo en el mismo drive.
- 66 Disk full Se llenó toda la capacidad del diskette.
- 67 Too many files Se intentó crear un nuevo archivo (usando SAVE u OPEN) estando llenas las 255 entradas del directorio.
- 68 Disk write protected Se intentó grabar sobre el diskette estando éste protegido.
- 69 Disk I/O error Ocurrió un error en una operación de entrada/salida. Este es un error fatal, el sistema operativo no puede recuperarlo.
- 70 Disk offline El drive especificado está abierto, o no contiene diskette, o éste no está formateado.
- 71 Rename across disk Intentó ejecutarse un comando RENAME entre un drive y otro.

Telemática S.A.

Chile 1347 - Tel.37-0051 al 54

1098 - Capital - República Argentina

REPORTE DEL USUARIO

■ Datos personales

Nombre _____

Calle _____

Localidad _____ C.P. _____ Provincia _____

Tel. _____ Fecha _____

■ Datos del equipo

Producto sobre el que reporta _____

Versión _____ Vendedor _____

Computadora:

Modelo _____ Marca _____ Memoria _____ Kb

Periféricos _____

■ Categoría del reporte

Problemas de software Problemas de documentación

Problemas de hardware Sugerencias

Otros

■ Describa el problema (como reproducirlo, y su diagnóstico)

■ Para uso de Telemática

Fecha de Recepción:

Atendido por:

Fecha de Resolución:

Comentarios:
