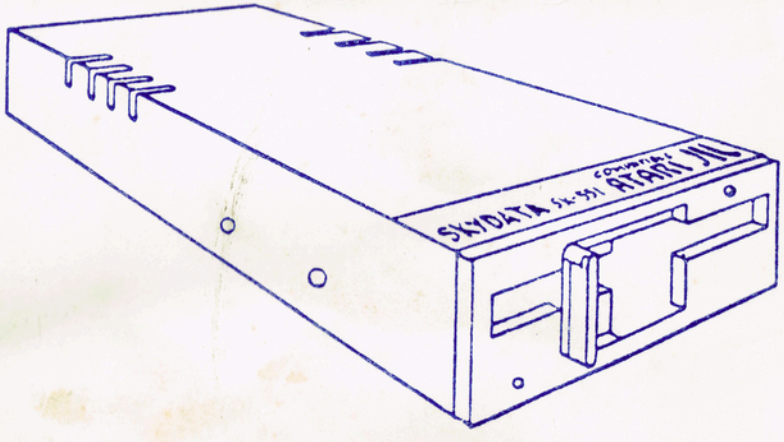
UNIDAD DE DISCO

SKYDATA SK - 551



COMPATIBLE

ATARI®

SK - 551

SKYDATA S.A.

UNIDAD DE DISCO COMPATIBLE PARA COMPUTADORAS ATARI

INDICE DE TEMAS

SU UNIDAD DE DISCO SK-551 3

LO QUE HACE EL DOS 3

SECCION I 5

LA CONEXION DE SU UNIDAD DE DISCO 4

COMO CONECTAR MAS DE UNA UNIDAD DE DISCO 8

SELECCION DEL NUMERO DE UNIDAD DE DISCO 9

EL CUIDADO DE LOS DISCOS 10

SECCION II 12

LOAD 13
SAVE 14
LIST 15
ENTER 15
RUN 16
BLOQUES DE CONTROL DE ENTRADA/SALIDA 17
OPEN 18
CLOSE 19
INPUT 20
PRINT 22
NOTE 23
PUT 24
GET 24
STATUS 25

SU UNIDAD DE DISCO
S K - 5 5 1

La unidad de disco SK-551 es un dispositivo de memoria de alta velocidad, extremadamente eficiente, que aumenta en gran medida la capacidad de su Sistema de Computación Personal ATARI. La memoria de su computador personal ATARI retiene la información y las instrucciones que Ud. ingresa a través del teclado. Pero la misma tiene su limite en tamaño y sin un dispositivo de almacenamiento como el SK-551, su contenido se borra cada vez que Ud. apaga el computador.

Su unidad de disco SK-551 le permite almacenar y administra grandes cantidades de información en archivos separados en diskettes. Con su SK-551 Ud. puede llamar sus archivos por su nombre, copiarlos o borrarlos y administrarlos de muchas formas.

LO QUE HACE EL DOS

Para almacenar información en los discos, Ud. necesita software que permita a su computador y a la unidad de disco comunicarse en torno a sus archivos. Es aqui donde hace su aparición el Sistema Operativo de Disco (DOS). El DOS es un programa, que le permite a su computador y a su unidad de disco trabajar en conjunto, para almacenar recuperar y en general administrar los archivos en disco. El DOS a su vez está organizado en archivos contenidos en

el disco maestro.

Ud. debe cargar el DOS en el computador antes de que pueda trabajar con su unidad de disco. Algunos programas computacionales comerciales ya contienen una versión del DOS, ahorrándole el trabajo y tiempo de cargarlos por separados. Pero en otros casos, especialmente en el de los programas en cartuchos, es necesario cargar el DOS junto con el programa, si UD. piensa usar una unidad de disco durante su ejecución.

En todo caso, necesitará el DOS para muchas tareas esenciales, por ejemplo: para preparar discos virgenes, para almacenar sus archivos y para poder hacer copias de respaldo de archivos y dicos importantes.

SECCION I
SU UNIDAD DE DISCO
S K - 5 5 1

Cuando Ud. desempaque su unidad de disco SK-551, verifique que contenga los siguientes items:

- Unidad de disco SK-551

- Adaptador de alimentación de corriente alterna.

- Disco Maestro DOS-SK

- Manual de Uso

- Su tarjeta y registro de garantia debidamente sellado por su distribuidor.

Si cualquiera de estos items faltára o la misma se encontrara con roturas, tome contacto con su distribuidor. Guarde el material de embalaje para el caso de que Ud. requiera transportar la Unidad de disco SK-551 o enviarla por correo.

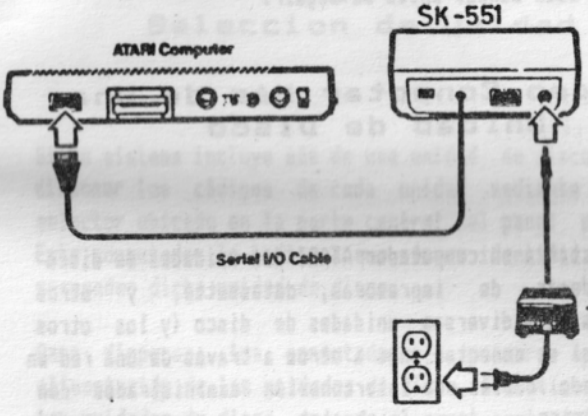
LA CONEXION DE SU UNIDAD DE DISCO
S K - 5 5 1

Siga atentamente estos pasos para conectar su SK-551 a su computador personal ATARI.

1.- Apague todos los componentes de su Sistema.

2.- Verifique que el interruptor de alimentación de su SK-551(ubicada en el panel posterior de su unidad de disco) este en la posición APAGADO).

3.- Primero enchufe el cordón con el conector más pequeño del adaptador de alimentación de corriente alterna en la perforación en el panel posterior de su Unidad de Disco. A continuación enchufe el adaptador de corriente alterna en un tomacorriente.



4.- Enchufe el extremo del cable de entrada/salida serie en el conector marcado PERIPHERAL de su computador.

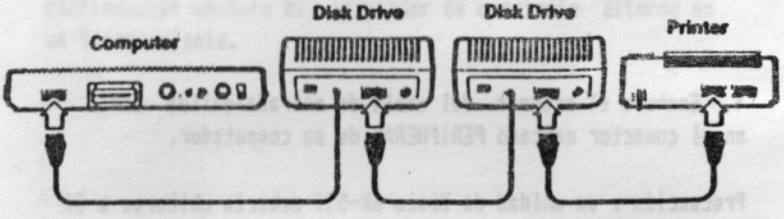
Precaución : su unidad de disco SK-551 deberia ubicarse a 30 cms o más de distancia de su televisor. Ya que este crea un fuerte campo magnético que podria afectar la información que se grabe en su disco.

5.- Prenda su unidad de disco. Se encenderán tanto la luz de encendido(en la parte inferior izquierda del panel frontal) y la luz de trabajo(parte superior derecha del panel frontal). Una vez que se apague la luz de trabajo, Ud. estará en condiciones de insertar un disco.

Ahora su unidad de disco está preparada para recibir el disco maestro SK-DOS. Es necesario que Ud. lea la próxima sección de este manual antes de seguir.

Cómo Conectar Más de Una Unidad de DIsco

Puede conectar a su computador ATARI dos unidades de disco SK-551, además de impresoras, datassette, y otros periféricos. Las diversas unidades de disco (y los otros periféricos) se conectan unos a otros a través de una red en serie, usando cables de interconexión suministrados con cada periférico.



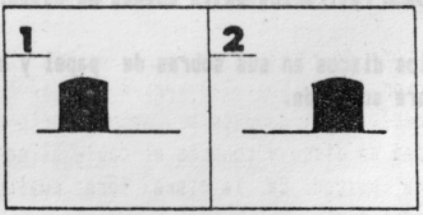
Hay un conector de entrada/salida (I/O Connector) en el panel posterior. Para instalar varias unidades de disco conecte el cable en el computador, seguidamente acomode la segunda unidad de disco y conecte el cable al conector I/O de la primera unidad. De la misma forma cualquier otro periferico adicional.

Disponiendo los conmutadores de Seleccion de Unidad

Si su sistema incluye más de una unidad de disco, Ud. debe disponer los códigos de cada unidad mediante un pequeño selector ubicado en la parte central del panel posterior. Este conmutador le indica a que número identificatorio debe responder dicha unidad de disco.

Para disponer los conmutadores, primero apague la alimentación de las unidades de disco. A continuación, gire las unidades de disco, dejando el panel posterior frente a Ud., de manera tal que pueda observar facilmente la ventana de selección de unidad (ubicada en la parte central de dicho panel). En el interior de la ventana se observa un selector negro, el cual está indicando unidad de disco número 1, colocándola en la posición contraria selecciona la número 2.

Usando un pequeño destornillador o un lápiz disponga los conmutadores como se observa en el gráfico.



Una vez dispuestos los conmutadores de cada unidad, recuerde poner una etiqueta con el número para no confundirse una con otra mientras use el DOS.

El cuidado de los Discos

La superficie de los discos está cubierta con un material magneticamente sensible, que almacena sus datos. Para asegurar una larga vida y confiabilidad de los discos, Ud. los debe tratar con limpieza y cuidado.

Cada disco se encuentra permanentemente incluido en un sobre protector negro; éste a su vez, normalmente se guarda en uno de papel. La mayoria de los discos tiene una pequeña muesca de protección de escritura a un lado del sobre protector negro. Cubriendo esta muesca con una etiqueta autoadhesiva, puede evitar el accidental borrado o grabación de la información ya escrita sobre el disco.

Reglas para el cuidado y uso de los discos:

* Nunca apague su unidad de disco mientras en su interior se encuentre un disco, y nunca deje un disco en su interior con la unidad de disco apagada.

* Nunca humedezca ni lave un disco.

* No doble sus discos; deben poder girar libremente dentro del sobre protector. Tómelos con mucho cuidado al introducirlos o retirarlos de la unidad de disco.

* Almacene los discos en sus sobres de papel y manténgalos apoyados sobre su borde.

* Almacene sus discos lejos del televisor. Los fuertes campos magnéticos que producen el televisor, pueden borrar datos almacenados. Mantenga sus discos lejos de todo dispositivo eléctrico, incluyendo el teléfono.

* No guarde sus discos bajo la luz directa del sol. Manténgalos lejos del calor excesivo.

* Como los dicos giran al interior de sus sobres, un daño al sobre puede tener como consecuencia el daño del disco.

* No escriba sobre los discos con un lápiz de grafito o un boligrado. Su punta dura puede dañar la superficie del mismo.

* No toque nunca un disco en la parte expuesta a través del sobre protector. Las huellas digitales pueden dañar el medio magnético.

Como etiquetar sus discos

La mayoria de los discos vienen con una etiqueta en una de las esquinas del sobre protector negro.
Asegurese de etiquetar cada disco con su correspondiente contenido.

SECCION II
EL USO DE COMANDOS BASIC CON DOS

Comando Basic Usados con Dos

Antes de aprender los comando Basic, que se usan don Dos debe saber como actuarán estos comandos sobre los programas que se estén grabando y recuperando. Los siquientes párrafos explican los dos tipos de archivos que pueden contener programas BASIC.

Archivos Codificados y No codificados.

El primer tipo de archivo llamado no codificado, contiene caracteres de texto ATASCII normales, de modo que los archivos parecen listas impresas de un programa BASIC. Estos programas no mantienen sus tablas simbólicas cada vez que se cargan o segraban. Las tablas simbólicas asocia el nombre de una variable con la ubicación de memoria en la cual se almacena el valor de esa variable. Para grabar y recuperar un archivo en su forma no codificada, use los comando LIST y ENTER.

Un archivo codificado es una versión condensada del programa BASIC. Tiene códigos de un Byte en vez de caracteres ATASCII para representar los comando Basic. Ud. puede mover hacia uno y otro lado los programas codificados desde y hacia la unidad de disco y la memoria del computador, usando los comando SAVE y LOAD. Las versiones codificadas de los archivos generalmente son más cortas que las versiones no codificadas. Por esta razón, muchos programadores prefieren archivar sus programas en forma codificada.

Generalmente, durante el desarrollo de un programa, la tabla simbólica se llena de nombre de variables que finalmente no se usan. Si Ud. usa una variable en una linea de programa y a continuación cambia su nombre o elimina la linea, el nombre original permanece en la tabla simbólica. Use el siguiente procedimiento para eliminar los imbolos de todos los nombres que ya no usa.

1 Cargue (LOAD) su programa

2 Listelo (LIST) a disco

3 digite NEW

4 Entre (ENTER) su programa desde el disco. La tabla de simbolos ahora contiene solamente aquellos nombres de variables realmente presentes en el programa.

LOAD (LO.)

Formato :LOAD esparchivo
Ejemplo : LOAD "D1:EJEMPLO1.BAS"

Este comando se usa para cargar un archivo desde un disco, hacia el area de memoria de programas. Antes de que Ud. pueda usar este comando para cargar un archivo llamado ejemplo1.bas, el archivo debe haber sido creado previamente usando el comando SAVE. Este comando solamente carga la versión codificada de un programa.

Se puede usar también este comando para concatenar programas. Si un programa es demasiado largo para poder ejecutarlo en la memoria, puede usar el comando LOAD para repartir el programa a través de dos archivos. Simplementeejecute la sentencia LOAD como ultima linea del archivo del primer programa, tal como lo muestra el ejemplo que sigue a este parrafo. Cuando el programa encuentre la sentencia LOAD, automaticamente leerá la próxima parte del programa desde el disco. El segunda archivo del programa debe ser capaz de funcionar solo sin depender de ninguna variable ni dato en RAM perteneciente al archivo del primer programa.

El programa cargado no será ejecutado mientras Ud. no digite RUN y presione RETURN, instante en el cual el programa anterior y todas sus variables serán borradas de la memoria.

```
100 REM CONCATENACION DE PROGRAMAS
110 LOAD "D:EJEMPLO2.BAS"
```

SAVE (S.)

Formato : SAVE esparchivo
Ejemplo : SAVE "D:EJEMPLO2.BAS"

Este comando instruye al sistema computacional para que grabe en el disco un programa con el nombre de esparchivo designado en el mismo programa. SAVE es el complemento de LOAD y almacena los programas en forma codificada.

LIST (L.)

Formato : LIST esparchivo, nolinea,nolinea,dispositivo
ejemplo : LIST "D1:ARDAT.LST"
LIST "P:"
LIST "P:",100,200

Uno de los usos del comando LIST es muy similar al comando SAVE. Puede tomar un programa del area de memoria del usuario y grabarlo a través de una unidad en particular con el nombre que Ud. quiera asignarle. Sin embargo, el programa se almacena en formato ATASCII normal y no como codigo.
Ciertas diferencias en el formateo del almacenamiento de datos también le permiten al LIST ser más flexible que el SAVE. Como se indica en los ejemplos, puede especificar un dispositivo determinado o puede especificar los números de lineas que deben listarse al dispositivo designado.

ENTER (E.)

Formato : ENTER esparchivo
Ejemplo ENTER "D:LISTADO.LST"

Este comando induce al computador a mover un archivo en el disco bajo la esparchivo mencionado a la RAM. El programa se ingresá en forma no codificada y se interpreta a medida que se reciben sus datos. ENTER, a diferencia de LOAD. no destruirá el programa BASIC que se encuentre residente en RAM, pero refundirá el programa residente en RAM con el archivo de disco que se está cargando. Si se repiten números de lineas en los dos programas, la linea del programa residente en RAM, será reemplazado.

RUN

Formato : RUN esparchivo
Ejemplo : RUN "D2:MIARCH.BAS"

Este comando hace que del computador cargue (LOAD) y ejecute (RUN) la esparchivo definida. Constituye una combinación de los dos comandos LOAD y RUN. Sin embargo, el comando RUN se puede usar el RUN "D:D:LIST.LST".

Para concatenar programas y lograr que el segundo segmento de un archivo se cargue y se ejecute automaticamente, puede usar el RUN "D:esparchivo" como última linea del primer segmento. Sin embargo, el segundo programa debe ser capaz de funcionar por si solo, sin depender de las variables o de los datos en RAM provenientes del primer programa. Antes de ejecutar el primer segmento, verifique que lo haya grabado en un disco, por la sentencia RUN borrará de la memoria RAM, el primer segmento, cuando se cargue el segundo.

Bloques de control de Entrada/Salida

Las operaciones I/O (Input/Output-Entradas/Salidas) se controlan por un bloque de control de Entrada/Salida (IOCB). El IOCB es una especificación de la operación I/O, que consiste del tipo de I/O, del largo de la memoria de transferencia, de la dirección de la memoria de transferencia y dos variables de control auxiliares adicionales, de las cuales la segunda normalmente vale 0.
Basic ATARI dispone 8 IOCB y dedica tres de ellos como sigue:

IOCB #0 usado por BASIC pasa Ingreso/Egreso al editor (E:)

IOCB #6 usado por BASIC para I/O de pantalla (S:)

IOCB #7 usado por BASIC para los comandos Lprint, Cload y Save

Los IOCB #1 al 35 pueden usarse libremente, pero los IOCB dedicades deberian evitarse, a no ser que el programa no requiera uno de los bloques dedicados indicados más arriba. IOCB #0 nunca deben abrirse o cerrarse desde un programa basic.

Cada comando I/O debe estar asociado con un IOCB. Los comandos I/O qye pueden usarse en conexión con el DOS son:

OPEN/CLOSE
INPUT/PRINT
PUT/GET
STATUS
XIO

Uso de los comando OPEN/CLOSE

OPEN (O.)

Formato : Open #iocb, expa1,expa2,esparchivo
ejemplo : OPEN #2,8,0,"D:SK551.BAS"

La sentencia OPEN enlaza un IOCB determinado al administrados de dispositivos(DEVICE NANCLER) correspondiente, inicializa las cariables de control relacionadas con el CIO, y pasa toda opción especifica del disposotivo al administrados de dispositivo. Los parámetros en esta sentencia se definen como sigue

:caracter obligatorio que debe ingresar el usuario

IOCB :Número comprendido entre 1 y 7 que se refiere al dispositivo.

expa1:Número que determinada el tipo de operación que debe ejecutarse.

codigo:

4=operación de entrada:pone el puntero de archivo en comienzo de archivo

6=Operación de entrada de directorio de disco.

7=entrada de directorio de disco con información de SK-DOS.

8=Operación de salida; ubica el puntero de archivo al comienzo del archivo.

9=Operación de añadido de fin de archivo; ubica el puntero de archivo al fin de archivo . El código 9 permite el ingreso de programas desde el editor de pantalla sin que sea necesario que el usuario presione RETURN.

12= Operación de entrada y salida;ubica el puntero de archivo al comienzo del archivo.

expa2: Código auxiliar dependiente del dispositivo.

esparchivo: Designación de archivo especifico.

En el ejemplo OPEN #2,8,0,"D:SK551.BAS", el IOCB #2 se abre para salida a un archivo que se encuentra en al unidad 1 con el nombre SK551.BAS. Si no existe archivo alguno con ese nombre en la unidad 1, el DOS lo creará. Si existe ya un archivo con este nombre, la sentencia OPEN lo destruye y crea uno nuevo. Su ya habia abierto este IOCB, la pantalla despliega el ERROR 129 (archivo ya abieto).

CLOSE (CL.)

Formato : CLOSE #iocb
Ejemplo : CLOSE #2

El comando CLOSE libera el IOCB que se habia abierto para operaciones de I/O. El número que le sigue al # obligatorio debe ser el mismo que el número de referencia del IOCB usado en la sentencia OPEN. No puede usarse el mismo IOCB para más de un dispositivo a la vez. No recibirá un mensaje de error si Ud. cierra un archivo que ya se habia cerrado.

```
10 OPEN #1,8,0,"D:ARCHIVO.EXT"
20 CLOSE #1
```

Nota: El comando END cerrará todos los archivos abiertos (exceptuando el IOCB #0).

El uso de los Comandos INPUT/PRINT

INPUT (I.)

Formato : INPUT #iocb; vars,vars
Ejemplo : Input #2,x,y
Input #2;N$

Este comando se usa para requerir datos(ya sean numericos o alfanumericos) de un dispositivo especifico. Input es el complemento de PRINT. Cuando se usa sin #iocb, los datos se suponen provenientes del dispositivo asumido (E:). INOUT se usa I/O por registros .

En le programa demostrativo INPUT/PRINT, listado a continuación, la linea 60 permite al usuario ingresar datos en el teclado. En la Linea 140 , la sentencia INPUT lee el contenido de un string desde el archivo abierto.

```
 10 REM  *** creacion  archivos
datos **
20  REM  ** open  con  "8"  crea
archivos **
30 OPEN #1,8,0 "D:GRABA.DAT"
40 DIM FRASE$(60)
50  ?"INGRESE UNA  FRASE  DE  NO
MAS DE 60 CARACTERES"
60 INPUT FRASE$
70 REM * GRABA DATOS **
80 PRINT #1;FRASE$
90   REM   **   CERRAR   (CLOSE)
ARCHIVO **
100 CLOSE #1
110  REM  **  ABRIR  ARCHIVO  P.
LECTURA **
120   REM **  OPEN CON  4:  SOLO
LECTURA **
```

```
130 OPEN #1,4,0,"D:GRABA.DAT"
140 INPUT #1,FRASE$
150 REM ** DESPLEGAR FRASE **
160 REM ** CERRAR ARCHIVO **
180 CLOSE #1
```

PRINT (PR. O ?)

Formato :PRINT #IOCB,[EXP1]...[EXPN]
Ejemplo : 100 prnt #2;x,y

Este comando escribe una expresión (ya sea string o númerica) hacia el dispositivo abierto con el mismo número de referencia IOCB.

Si no hay especificación del iocb, el sistema escribe la expresión en la pantalla que es el dispositivo asumindo. Si la información se encamina hacia un dispositivo que no se encuentra abierto, en la pantalla se desplegará un ERROR 133.

Print realiza lo que se llama I/O por registros, los registros son conjuntos de bytes separados por un carácter de fin de linea (EOL,9B Hex). El tamaño de un registro es arbitrario. El tamaño de un registro puede estar determinado por el largo de un string enviado a un archivo de disco o el fomato de una variable aritmetica. También puede ser el largo de un string de caracteres ingresados desde el teclado y terminado por un RETURN.

La sentencia INPUT generalmente no puede leer un registro que tenga más de 255 carateres de largo. Si Ud. quiere grabar con print un registro en disco, para más tarde recuperarlos con INPUT, lo mejor será limitar el tamaño del registro a 255 caracteres o menos.

Acceso directo con los comandos directos NOTE/POINT

NOTE (NO.)

formato : NOTE #IOCB,vara,varb
Ejemplo : NOTE #2,A,B

Los archivos se crean en forma secuencial y normalmente se acceden desde su comienzo hasta su final. Si Ud. quiere acceder los registros de un archivo en forma no secuencial(directa), puede ya sea leer el archivo en forma secuencial y detenerse en el registro deseado, o usar en método especial de direccionamiento del registro que busca.

Como lo primero es muy lento para archivo largos, el DOS incorpora NOTE y POINT para darle la posibilidad de acceso al archivo en forma aleatoria. Para llegar a un registro , sin pasar a través de todos los registros precedentes, Ud. necesita que el computador sepa cuál es el registro que busca. Esto requiere una anotación del sector del archivo, de modo que Ud. usará un comando NOTE ante cada escritura y grabará el valor recuperado en una tabla.

Este comando obtiene el valor acutal del puntero del archivo, válido para el archivo que usa el IOCB especificado. El puntero de archivo especifica el lugar exacto del archivo, donde se leerá o se escribirá el próximo byte. Este comando alamcena el número absoluta del sector de disco en la primera vairable aritmetica y el número de byte actual en la segunda. Los números de bytes van del 0 al 124.

Los Comandos PUT/GET

PUT (PU.)

Formato : PUT #iocb,expa
Ejemplo : PUT #6,asc("a")

El comando PUT escribe un byte (un valor entre 0 y 255) al dispositivo especificado por el número de referencia del iocb. En el siguiente programa ejemplo se usa el comando PUT para escribir cada número que Ud. ingresa en un agrupamiento dimensionado como A(n). Puede ingresar hasta n numeros, cada uno de los cuales debe ser menor que 256 y también mayor o igual que 0. Este comando se usa para crear archivos de datos o para agregar datos a un archivo ya existente.

GET (GE.)

Formato :GET #iocb,var
Ejemplo : GET #2,x

Este comando lee un solo byte del byte del dispositivo especificado por el número de referencia del iocb a la variable esecificada. GET le permite recuperar cada byte almacenado con le comando PUT.

El Uso del Comando STATUS

STATUS (ST.)

Formato :STATUS #iocb,vara
Ejemplo : STATUS #5,ERROR

El comando STATUS se utiliza para determinar la condición de un archivo. Se trata de un comando CIO que comprueba por varias vias en que puede presentarse un error. El primer conjunto de posibles errores que verifica, es el siguiente:

Está disponible la memoria para el sector?	--> ERROR 161
El número de dispositivo es legal	--> ERROR 20
Nombre de archivo legal	--> ERROR 170
El archivo esta en el disco	--> ERROR 170
El archivo esta protegido	--> ERROR 167

Ud. también puede identificar errores de la linea serial de I/O como comando STATUS. Son los siguientes:

Error 138	Tiempo de dispositivo excedido.
Error 139	Dispositivo no responde.
Error 140	Error de la linea serial.
Error 141	Desborde del campo del bus serial.
Error 142	Error de verificación del bus serial
Error 144	Error del dispositivo.

Antes de poder dar un comando STATUS, debe abrirse el archivo; se recomienda usar la forma XIO (Ver referencia en el Manual Del SISTEMA OPERATIVO), ya que es mas confiable y permite asociar un nombre de archivo especifico al error que está tratando de encontrar.

N O T A :

Para mayor información dirijase a la siguiente bibliografia:

- Manual del SK-DOS

- Manual del DOS 2.5

- Una Guia a la Programación Avanzada

- Curso Todo sobre ATARI

- Manual de BASIC Avanzado

MARCELO MOJICA

(C) SKYDATA S.A. 1988

Todos los derechos reservdos por SKYDATA S.A. Hecho el depósito que marca la Ley 11.723
Primera Edición impresa en Setiembre de 1988

Edición de SKYDATA S.A.