ADVANCED DIAGNOSTICS

The Information contained in this manual is subject to change without notice

Millers Graphics shall not be liable for technical or editorial errors or omissions contained herein; nor for incidental or consequential damages resulting from the furnishing, performance, or use of this material or product described by this manual.

Manual written by Craig G. Miller and W.R. Moseid

Advanced Diagnostics Software and Associated Manual
Printed in the United States of America

Millers Graphics
1475 W. Cypress Ave.
San Dimas, CA 91773

TABLE OF CONTENTS

GETTING STARTED

THE COMMANDS

FLOPPY DISK CONTROLLERS

APPENDIXES

START UP INSTRUCTIONS

>> IMPORTANT WARNINGS << This utility program can execute a number of different WRITE operations to your diskettes! Read through THE COMMANDS section of the manual before attempting to execute any of the Advanced Diagnostics commands or you may inadvertently DESTROY part of a good diskette!! ——— ALWAYS keep the write protect notch covered on the Advanced Diagnostics diskette. You should only remove it to change the DIAGCONFIG Command File and then immediately cover it back up. (see Command File in THE COMMANDS section of this manual for more information on the DIAGCONFIG file.)

Advanced Diagnostics comes with two different loaders on the disk. One for Extended Basic and one for the Editor Assembler or Mini Memory module.

NOTE: The Advanced Diagnostics disk MUST be in Drive 1 when loading.

EXTENDED BASIC

To load Diags with the Extended Basic module place the Advanced Diagnostics disk in drive 1 and select Extended Basic from the menu. There is a file named LOAD on the disk that will Auto Boot and Run Advanced Diagnostics.

EDITOR/ASSEMBLER

To load Diags with the Editor/Assembler module place the Advanced Diagnostics disk in drive 1 and select Editor/Assembler from the menu. Next select 3 LOAD AND RUN from the Editor/Assembler menu.

```
        When this prompt appears        * LOAD AND RUN *
                                          FILENAME?
                  Type in                 DSK1.DIAGS        and press ENTER
```

Advanced Diagnostics will then load and Auto Run.

MINI MEMORY

To load Diags with the Mini Memory module place the Advanced Diagnostics disk in drive 1 and select Mini Memory from the menu. Next select 1 LOAD AND RUN from the Mini Memory menu.

```
        When this prompt appears        * LOAD AND RUN *
                                          FILENAME?
                  Type in                 DSK1.DIAGS        and press ENTER
```

Advanced Diagnostics will then load and Auto Run.

QUICK REFERENCE - ADVANCED DIAGNOSTICS COMMAND LIST

Name —— Abbreviation ——— Syntax ——————————————————— Example

BEEP - BE BEEP BE

Sounds a Beep tone. Mainly for use in Command Files.

CHANGE COLORS - CC CHANGE COLORS fore back border ^char (1-16) CC 16 5 16 8

Sets the foreground (alphanumerics), background (screen), border (boxes) and ^ char (special character) colors.

CHECK DISK - CD CHECK DISK CD

Check the diskette in the Selected Drive for sectors that are mapped as bad. It then reports the disk Free, Used and Bad, and displays the sector numbers that are bad.. After listing out the Bad sector numbers, on a key press, this routine will list out all of the fractured files on the selected disk

COMMAND FILE - CF COMMAND FILE DSKx.filename or CF REDO CF DSK1.DIAGCONFIG

This command loads and runs a DIS/VAR 80 text file from the indicated device and executes the commands in the file as if they had been typed in from the keyboard.

CHECK MEMORY - CM CHECK MEMORY CM

Preforms complete Refresh and Bit Shift memory tests of Scratch Pad RAM, VDP RAM, Memory Expansion RAM and Mini Memory RAM.

CONVERT NUMBER- CN CONVERT NUMBER no. or >no. or Bno. (0-65535) CN >5A0

Converts the specified number (decimal, hexadecimal or binary) into the other two number bases and displays all three number bases on the screen.

COPY READ - CR COPY READ start_sector(0-?) #_of_sectors(1-36) CR 0 36

Reads 1 to 36 sectors from the Selected Drive, starting with the specified sector, into a copy buffer for use with COPY WRITE to copy diskettes or parts of diskettes.

COPY WRITE - CW COPY WRITE start_sector(0-?) #_of_sectors(1-36) CW 0 36

Writes 1 to 36 sectors from the copy buffer onto the diskette in the Selected Drive starting with the specified sector . (see COPY READ)

DISK DIRECTORY- DD DISK DIRECTORY DD

Reads and displays the file names on the Selected Drive.

EDIT SECTOR - ES EDIT SECTOR sector#(0-?) or ES REDO ES 719

Positions the head to the specified sector and reads the data into an editing buffer. The data can then be edited in ASCII or Hex and written back out. (see WRITE SECTOR)

FORMAT DISK - FD FORMAT DISK #sides(1-2) density(S-D) diskname FD 1 S TEST

Formats the diskette in the Selected Drive in the format specified and places a standard header on sector zero with the specified diskname.
NOTE: All data previously on the disk will be WRITTEN OVER!

FIND FILE - FF FIND FILE filename FF DIAGCONFIG

This command finds the file header of the specified file. The Track, Sector and Side indicators in the Status block point to the file header location. The location(s) and number of sectors of the file are displayed under the Status block along with the file size and type.

HEAD STEP - HS HEAD STEP step_time HS 10

Sets the head step time at the specified time in milliseconds for all disk operations.
TI = 6, 12, 20 or 30 CC = 3, 6, 10 or 15

QUICK REFERENCE - ADVANCED DIAGNOSTICS COMMAND LIST Continued

Name —— Abbreviation ——— Syntax ——— Example

MOTOR SPEED - MS MOTOR SPEED sample_rate Hz(50-60) MS 10 60

Displays the RPM of the Selected Drive based on the sample rate (# of index pulses) specified. High sample rate returns drives average RPM. The Hz is the AC cycles per second for Foreign (50) or U.S. (60) power.

OUTPUT DEVICE - OD OUTPUT DEVICE device_name OD PIO

Sets the output device for FCTN 0 screen dumps. Any valid output device may be used. ie: Disk, Thermal Printer, etc.

OUTPUT WIDTH - OW OUTPUT WIDTH line_length(1-80) OW 40

Changes the line length of the DIS/VAR output file for FCTN 0 screen dumps from the default of 80 to the specified line length.

PAUSE - PA PAUSE PA

Halts execution, displays 'Press any key to continue' and waits for a key press. Mainly for use in Command Files to allow diskette changes or a chance to abort.

READ TRACK - RT READ TRACK track#(0-79) RT 3

Steps the read/write head to the specified track and reads every sector, Gap and Address byte of the track into a page-able buffer.

SEEK TRACK - ST SEEK TRACK track#(0-79) ST 38

Steps the read/write head to the specified track. Track numbers 0 through 39 are on side one and 40 through 79 are on side two.

SELECT DRIVE - SD SELECT DRIVE drive#(1-5) SD 2

Selects the drive number for all commands except Command File and Output Device.

TIME DELAY - TD TIME DELAY #_of_seconds TD 3

Implements a time delay between instructions. Mainly for use in Command Files to allow diskette changes and/or the drive motor to time out (stop turning).

USE DSR - UD USE DSR UD

Toggles the DSR ROM on and off for using the sector and format routines in the DSR ROM. This allows testing of the DSR ROM routines or the use of Diags with a true RAM Disk.

WRITE SECTOR - WS WRITE SECTOR sector#(0-7) WS 45

Positions the read/write head to the specified sector and Writes the data that is in the Edit Sector buffer and verifies that the data was written out properly.
NOTE: Any data previously in the sector will be WRITTEN OVER!

WRITE TRACK - WT WRITE TRACK track#(0-79) density(S-D) WT 6 S

Steps the head to the specified track and Writes default format data to every sector on the track in the specified density.
NOTE: All data previously on the track will be DESTROYED!

——— NOTE ———

Even though the Examples listed only show the abbreviated command, you may also use the full command name. (see THE COMMANDS - General Rules)

ie: Format Disk 1 S TEST performs the same as FD 1 S TEST

QUICK REFERENCE - FCTN & CTRL KEY LIST

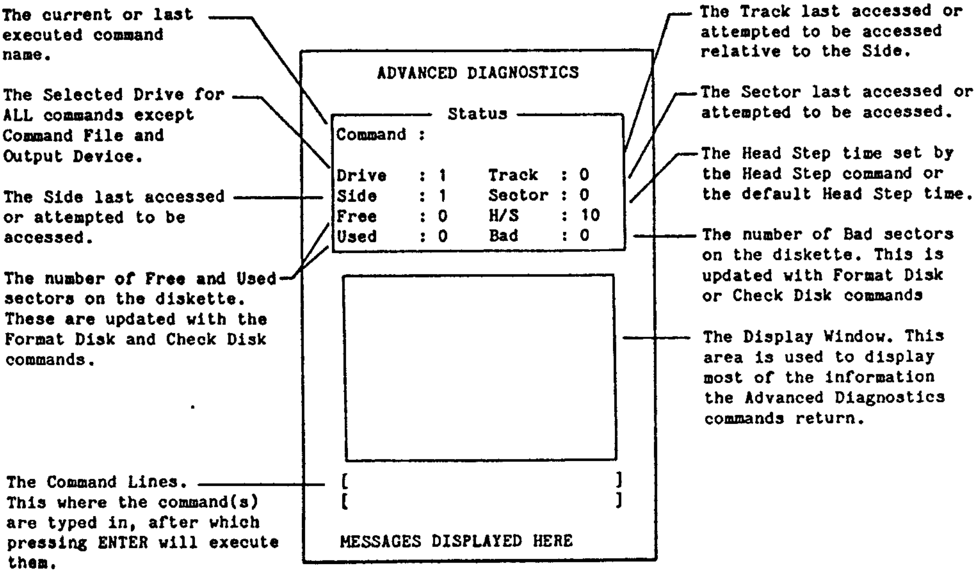
FCTN Key	Key-code	Description
1 DELETE	[3]	Deletes a character from the command line or sector being edited.
2 INSERT	[4]	Inserts a space character into the command line or the sector being edited.
3 ERASE	[7]	Erases the entire command line or sector being edited. (see Edit Sector)
4 ROLL UP	[2]	Brings the next higher numbered sector/track into the Edit/Track buffer. Note: Any changes to the sector that was previously in the edit buffer will be lost. You MUST perform a WRITE SECTOR to save your changes.
5 BEGIN	[14]	Takes you from any screen to the program's title screen. From the program's title screen FCTN 5 BEGIN will take you out of the program to the computers title screen. This key also Aborts most of the Diags commands.
6 ROLL DOWN	[12]	Brings the next lower numbered sector/track into the Edit/Track buffer. Note: Any changes to the sector that was previously in the edit buffer will be lost. You MUST perform a WRITE SECTOR to save your changes.
7 AID	[1]	Windows in the help screen. CTRL E & CTRL X allow you to page through the help section any other key press will abort Help.
8 REDO	[6]	Re-checks a memory location that is bad during the Check Memory command.
9 BACK	[15]	Brings you back to the main Command Screen from Edit Sector or Read Track.
0 SCRN DUMP	[188]	Dumps the contents of the screen to the Output Device selected. (see Output Device)
= ASCII/HEX	[5]	Toggles the Edit buffer and Track buffer displays between ASCII characters and their hexadecimal values.
E UP ARROW	[11]	Moves the cursor up a line. If the cursor is on the top line of the display it will home the cursor.
X DOWN ARROW	[10]	Moves the cursor down a line.
S LEFT ARROW	[8]	Moves the cursor left one character.
D RIGHT ARROW	[9]	Moves the cursor right one character.

CTRL Key	Key-code	Description
E UP ARROW	[133]	Page Up is used during FCTN 7 HELP and Read Track to display the next Help screen or track buffer.
X DOWN ARROW	[152]	Page Down is used during FCTN 7 HELP and Read Track to display the previous Help screen or track buffer.

Special Characters For Use In A Command File

[253]	Delay. Executes 1 second delay in a Command File
[254]	Executes a Command File in Slow speed.
[255]	Executes a Command File in High speed.
[]	End Of File marker for Command Files.

QUICK REFERENCE - ADVANCED DIAGNOSTICS COMMAND SCREEN



```
           ADVANCED DIAGNOSTICS

      ---------- Status ----------
Command :

Drive   : 1    Track  : 0
Side    : 1    Sector : 0
Free    : 0    H/S    : 10
Used    : 0    Bad    : 0




[                                   ]
[                                   ]

MESSAGES DISPLAYED HERE
```

The current or last executed command name.

The Selected Drive for ALL commands except Command File and Output Device.

The Side last accessed or attempted to be accessed.

The number of Free and Used sectors on the diskette. These are updated with the Format Disk and Check Disk commands.

The Command Lines. This where the command(s) are typed in, after which pressing ENTER will execute them.

The Track last accessed or attempted to be accessed relative to the Side.

The Sector last accessed or attempted to be accessed.

The Head Step time set by the Head Step command or the default Head Step time.

The number of Bad sectors on the diskette. This is updated with Format Disk or Check Disk commands

The Display Window. This area is used to display most of the information the Advanced Diagnostics commands return.

NOTES

The commands COPY WRITE, FORMAT DISK, WRITE SECTOR and WRITE TRACK will all WRITE data to the diskette in the Selected Drive. Make sure the drive indicator in the Status Window displays the proper drive number before executing ANY of these commands!!

The Command Line will not erase itself after the command is completed, so you can execute a command over and over by just pressing Enter. The cursor does not need to be at the end of the command list to execute the entire Command Line. After a command has been executed the cursor will home itself. To clear the entire Command Line just press FCTN 3 ERASE.

The Status Window is updated at the beginning and end of every command. So with some commands, such as Check Disk or Find File, with a lot of file clusters, the Status Window will only display correct information after ALL of the "Press Any Key To Continue" functions have been completed. Most commands do not affect all of the Status indicators. In the COMMANDS section of this manual, where the screens are displayed, the indicators that have a "nu" after them are not used by the command being documented. These "nu" indicators will display the information from the last command that updated them.

The Status Window on the Edit Sector and Read Track Screens (not show here) display correct information with every Roll Up or Roll Down. If however you are using the DSR ROM routines (see USE DSR command) the Side and Track indicators will not be updated.

THE COMMANDS

General Rules

1. All commands and command parameters are delimited with a space character. Every command must end with a space character or Carriage Return (ENTER) and it must have at least 1 space inbetween each parameter. If you use the full command name there must NOT be more than one space inbetween the first and second word of the command. If you use the abbreviated form of the command then there must NOT be any spaces inbetween the first and second letters of the command.

```
          Correct                                   Incorrect
     Select Drive    1                         Select     Drive 1
     Change Colors 16 13  12   4               Change Colors16 1312 4
     SD 1 HS 10                                S  D 1   H S10
```

2. The commands and parameters can be in upper case or lower case or both.
   All of the following are Correct:

```
select DRIVE 1     Select Drive 1     sElect driVe 1     SD 1     sd 1     Sd 1
```

3. The numeric parameters can be input in either Decimal or Hexadecimal numbers. Hexadecimal numbers must be proceeded by the > sign. Some of the commands, such as Edit Sector, do not have an upper limit, however the Status Window will only display 4 digits. This was done to allow access to future devices, like a RAM Disk.

```
          Correct                                   Incorrect
      Edit Sector >3F                           Edit Sector 3F
      Edit Sector 271  or  ES >10F              Edit Sector 10F
```

BEEP or BE

Generates a Beep tone. Mainly for use in Command Files to indicate a PAUSE or the end of the Command File however, it can also be used in the Command Lines.

Example: `SD 1 HS 10 BEEP PA Format Disk 1 S TEST`

This will Select Drive 1, set the Head Step time to 10 milliseconds, BEEP, wait for a key press (PA) and then Format the diskette in drive 1 as single sided single density with the disk name of TEST.

CHANGE COLORS foreground background border ▪char   or   CC foreground background border ▪char

Allows you to set the foreground (alphanumeric characters), the background (screen), the border (box lines) and the ▪ character (special ▪ character) colors to suit your monitor. The ▪ character is used on the Edit Sector and Read Track Screens. It is used on these screens in ASCII display mode to indicate that the value in that byte location is out of the normal 32 - 127 ASCII display range.

Color Codes

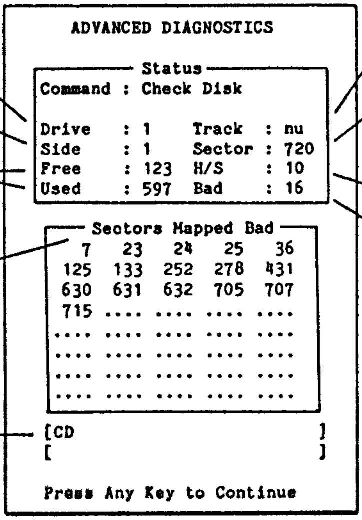
1	Transparent	5	Dark Blue	9	Medium Red	13	Dark Green
2	Black	6	Light Blue	10	Light Red	14	Purple
3	Medium Green	7	Dark Red	11	Dark Yellow	15	Gray
4	Light Green	8	Cyan	12	Light Yellow	16	White

Example: `CC 16 13 12 4`

This will set the text color to White, the screen color to Dark Green, the box lines to Light Yellow and the ▪ character to Light Green.

CHECK DISK or CD

The Check Disk command checks the diskette for sectors that have been mapped bad in the Bit Map Allocation Table in sector 0 and it also checks the diskette for Fractured Files. This command will also update the Free, Used and Bad indicators in the Status Window.



```
          ADVANCED DIAGNOSTICS

  ----------- Status -----------
  Command : Check Disk

  Drive   : 1     Track  : nu
  Side    : 1     Sector : 720
  Free    : 123   H/S    : 10
  Used    : 597   Bad    : 16

   ---- Sectors Mapped Bad ----
      7    23    24    25    36
    125   133   252   278   431
    630   631   632   705   707
    715  ....  ....  ....  ....
   ....  ....  ....  ....  ....
   ....  ....  ....  ....  ....
   ....  ....  ....  ....  ....
   ....  ....  ....  ....  ....

  [CD                          ]
  [                            ]

  Press Any Key to Continue
```

The Drive where the Checked Diskette is.

Side last accessed or attempted to be accessed.

Total number of Free and Used sectors on this diskette

Sector numbers found to be mapped as BAD in the Bit Map Allocation Table.

Typing in CD and pressing ENTER will execute the Check Disk command.

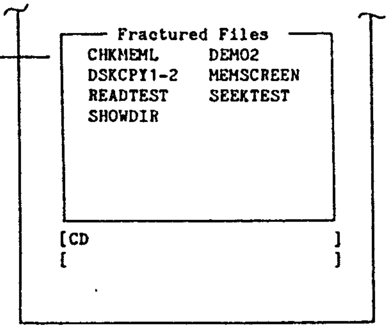
The Track indicator is not used with this command.

Indicates the Total number of sectors formatted on this diskette.

The Head Step Time used.

The total number of Bad sectors on this diskette.

After it has displayed the bad sector numbers, if there are any, it displays "Press Any Key To Continue" at the bottom of the screen. After you press a key it will then display the file names of the files that are found to be fractured. A Fractured File is a file whose data does not reside in numerically contiguous sectors. Part of the file may be in sectors 34 through 42 and the rest of the file may be in sectors 56 through 61. If a file is badly fractured (more that three noncontiguous sections) it may slow down the access time for reading and writing to the file because of all the head stepping required to get to the other sectors. To unfracture files simply use the Disk Manager's file copy function to copy the whole disk onto a new disk. Sector copiers will NOT unfracture files. (see FIND FILE for displaying the files location(s))



```
  ------ Fractured Files ------
   CHKMEML      DEMO2
   DSKCPY1-2    MEMSCREEN
   READTEST     SEEKTEST
   SHOWDIR




  [CD                          ]
  [                            ]
```

Fractured File Names are displayed here. If there are more than 16 of them the "Press Any Key..." message will be displayed at the bottom of the screen

THE COMMANDS Continued

COMMAND FILE DSKx.filename or CF DSKx.filename or COMMAND FILE REDO or CF REDO

A Command File is a collection of Advanced Diagnostic commands, key-codes and key strokes which are created with any editor or word processor that saves its files in DIS/VAR 80 format. Two such utilities are the Editor/Assembler editor and the TI Writer word processor programs. When the Command File DSKx.filename command is executed, Advanced Diagnostics will go to the specified drive and load the specified file into memory. After it is loaded Diags will then execute the commands in the file as if they were typed in from the keyboard only much faster. This advanced feature allows the creation of some powerful Command Files that will automatically perform repetitive functions and tasks for you. Command Files may be any size up to 2048 bytes and they can contain ANY of the commands, key-codes and keystrokes known to Diags. A Command File can also load and run another Command File, as the last command in the file, so you can chain multiple Command Files together to perform large tasks.

Command File Tips & Rules

1. 2048 bytes is 2K or approx 50 lines with 40 columns of characters. This is a file size of about 9 disk sectors, 1 for the File Header and 8 sectors of data. If a Command File is larger than 2K Advanced Diagnostics will NOT execute it.

2. Each of the two Command Lines is 26 characters long. This is the 3rd tab position in the Editor Assembler and the 4th tab position in TI Writer. If you use TI Writer to create your Command Files, save your files with the Print File command and not the Save File command. The Print File command does not save the TI Writer tabs as the last record of the file. No matter which editor you use make sure the last characters in your Command File are []. This is the End of File marker for Diags.

3. To have Diags execute the FCTN & CTRL keys in your Command File just place the key-code inside the bracket characters. ie: [9] will move the cursor right one character. See the Quick Reference - FCTN & CTRL Keys page for the various key-codes. You can also use these brackets for any character value. ie: [32] prints a space character. If you forget to close an open bracket, [8[32], Diags will stop execution at that point. The following characters have a special meaning to the Command File interpreter:

   [253] Generates a 1 second delay before executing the next item in your Command File. The delay time is the same for high speed or slow speed Command File execution.

   [254] Tells the interpreter to execute the items that follow it in Slow speed. This speed is good for testing and debugging your Command Files. You can abort a slow speed Command File at any time by pressing any key on the keyboard.

   [255] Tells the interpreter to execute the items that follow it in High speed. You can change the speed as often as you wish within the Command File. The only time you can abort a high speed Command File is when the PAUSE command is executed and you press FCTN 5 BEGIN.

   [] Tells the interpreter that this is the End of the Command File so that it will not try to execute any commands left over from a previous Command File.

   NOTE: ALL key-codes and character values must be in decimal and have only 1 value per bracket. ie: [8 32] is not allowed, change it to [8][32].

4. Some of the Advanced Diagnostics commands require one or more key presses to complete the command. Check Disk, Disk Directory, Motor Speed and Find File when there are a lot of clusters, are a few such commands. When you place these in a Command File you should include a few extra space characters after the [13] key-code to compensate for this. Otherwise Diags might use a few of the characters from the next command, which will cause an error.

5. Since the Command Files are saved in a DIS/VAR 80 format, all space characters on the end of the lines are stripped off. If you want to retain some space characters just use the [32] key-code as the last space character on the line.

COMMAND FILE Continued

6. Whenever a Command File is loaded, or executed with Command File REDO, the ASCII/Hex display mode automatically defaults to Hex. This way you always know what display mode you are in for Edit Sector or Read Track. The [5] key-code will toggle the display mode in your Command Files, this is the same as pressing FCTN =.

7. When a Command File is loaded and starts to run, the previous command(s) are still on the Command Line. The [7] key-code in your Command Files will erase this area, so it is a good idea to place this at the beginning of your Command Files. This is the same as pressing FCTN 3 ERASE.

Command Files can easily be created to perform a variety of tasks and tests. As you become familiar with Diags and Command Files we think you'll find this to be a truly advanced and powerful feature. There are 7 Command Files on the Advanced Diagnostics disk that should help you understand the structure and simplicity of a Command File. These files perform the follow tasks and tests:

DIAGCONFIG — This Command File automatically loads and runs when ADVANCED DIAGNOSTICS first starts up. It currently contains the commands Change Colors and Output Device. You can modify this Command File to have Diags boot up with your own Output Device, Head Step time, Colors etc. to match your system's configuration. If DIAGCONFIG is not on the diskette in drive 1 Diags will return the "Bad Open Attribute" error message, which means it could not be found. (see Appendix A - FDC and DSR Error Messages)

BOXFORMAT — This file will format a box of ten floppy diskettes in the Selected Drive as single sided, single density with the disk names of DISK-01 through DISK-10. This file can easily be modified for double sided and/or double density.

DSKCOPY1-2 — This will format the disk in drive 2 as single sided, single density and then copy the entire single sided, single density disk in drive 1 to drive 2.

DISKHEADER — This file will Edit Sectors 0 and 1 in the Selected Drive and change them to a standard, freshly formatted, disk header for a single sided, single density diskette. This file was included to show you how to use the Edit Sector command in a Command File.

R&WTEST — This file performs a complete single sided, single density Read and Write Test on the Selected Drive. This test is not destructive since it rewrites the same data that was Read.

READTEST — This file performs a complete single sided, single density Read Only test on the Selected Drive.

SEEKTEST — This file performs a complete single sided Seek Track test on the Selected Drive with the Head Step time that was previously set.

——— Example DIAGCONFIG Command File ———

```
[255][7]                       Sets up high speed execution and clears the Command Line.
CC 16 13 12 4     [13][7]      Changes the Colors and clears the Command Line.
OD RS232.BA=4800  [13][7]      Sets up the Output Device and clears the Command Line.
Beep              [13][7]      Generates a Beep tone and clears the Command Line.
      Welcome to        [32]   Displays part of the message.
   ADVANCED DIAGNOSTICS [253]  Displays the rest of the message, waits 3 seconds and then
[253][253][7][]                clears the Command Line and ends the Command File.
```

CHECK MEMORY or CM

The Check Memory command performs complete Refresh and Bit Shift tests on Scratch Pad RAM, VDP RAM, Memory Expansion RAM and Mini Memory RAM, if it is present. These tests are conducted on the entire area of RAM in 4K segments. During these tests ADVANCED DIAGNOSTICS moves a number of 4K blocks of memory around to preserve the Diags environment. Because of this some of the Diags functions are turned off until the test is completed or aborted when a error is found. One such function is the FCTN 0 Screen Dump to the Output Device.

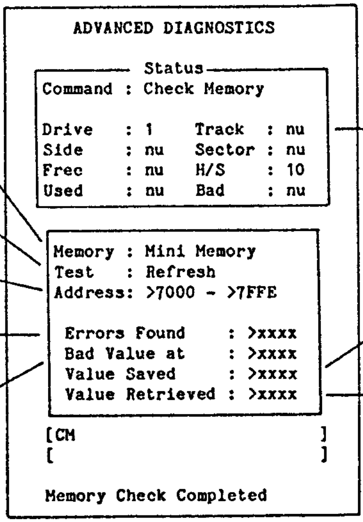
The Bit Shift test writes a bit value to every bit location in every RAM word in the test and then checks it. The Refresh test writes a value to every word location in the RAM areas. After these Refresh values have been written to the 4K segment Diags will wait approximately 3 seconds and then go back and check to make sure that the value written is still intact. Since there are 8 bits in every byte and Diags writes to every byte 9 times, once for refresh and 8 times for Bit Shift and then reads every byte 9 times, there are 18 accesses made to every byte of RAM. This means that the Check Memory test performs 4,608 accesses on Scratch Pad RAM (18 x 256). 294,912 accesses on VDP RAM (18 x 16,384). 589,824 accesses on Memory Expansion RAM (18 x 32,768) and 73,728 accesses on Mini Memory RAM for a total memory check of 963,072 memory accesses, not counting all of the 4K block moves that take place to preserve the Diags environment.

NOTE: If an area of memory is found to be bad Advanced Diagnostics will halt the test and ask if you want to continue. At this time you can press FCTN 8 REDO to retest that memory location. It is possible that this bad area of memory is pertinent to the proper operation of Check Memory and as such Diags will not perform properly after the test.

EXAMPLE: Check Memory or CM will execute the Check Memory test.

PA Check Memory or PA CM will execute the test after a key press.

NOTE: All values displayed in the Display Window are in hexadecimal for this command.



```
ADVANCED DIAGNOSTICS

         Status
Command : Check Memory

Drive   : 1    Track  : nu
Side    : nu   Sector : nu
Free    : nu   H/S    : 10
Used    : nu   Bad    : nu

Memory : Mini Memory
Test   : Refresh
Address: >7000 - >7FFE

 Errors Found     : >xxxx
 Bad Value at     : >xxxx
 Value Saved      : >xxxx
 Value Retrieved  : >xxxx

[CM                      ]
[                        ]

Memory Check Completed
```

The Current area of Memory that is being tested.

The type of test being performed.

The address range being tested.

Total number of Errors found so far.

Current or last address where an error was found.

The Status indicators are not used during the Check Memory command.

The value written to the bad memory address.

The value that was read from the bad memory address.

THE COMMANDS Continued

CONVERT NUMBER no. or >no. or Bno. or CN no. or >no. or Bno.

Convert Number enables you to convert a decimal, hexadecimal or binary number from its current base into the other two number bases which will be shown in the Display Window on the Command Screen. The valid input and display range for the different bases are: 0 - 65535 for decimal, >0 - >FFFF for hexadecimal and B0 - B1111111111111111 for binary.

```
    EXAMPLES:   Convert Number  239    or   CN  239    converts the decimal number 239
                Convert Number >2FE    or   CN >2FE    converts the hexadecimal number >2FE
                Convert number B0110   or   CN B0110   converts the binary number 0110
```

COPY READ start_sector(0-?) #_of_sectors(1-36) or CR start_sector(0-?) #_of_sectors(1-36)

With this command you can copy up to 36 sectors at a time into a copy buffer from the Selected Drive, starting with the specified start_sector number. After the sectors are in the the buffer the Copy Write command will write them back out. This allows you to move sectors around on the diskette or to copy part or all of one diskette to another.

36 sectors equals 4 full tracks on a single density diskette or 2 full tracks on a double density diskette. Since these sectors are read into a buffer you can easily switch diskettes in a single drive system or select another drive before issuing the Copy Write command.

```
    EXAMPLES:   Copy Read  0 36     reads sectors  0 through 35, inclusive, into the buffer.
                Copy Read 36 36     reads sectors 36 through 71, inclusive, into the buffer.
                       CR 34  2     reads sectors 34 and 35 into the buffer.
```

NOTE: The buffer used for Copy Read and Copy Write is also used by the Check Disk, Check Memory, Format Disk, Disk Directory and Read Track commands. You must issue a COPY WRITE command before any of these commands or the data in the buffer will be DESTROYED! Edit Sector does not destroy the copy buffer.

COPY WRITE start_sector(0-?) #_of_sectors(1-36) or CW start_sector(0-?) #_of_sectors(1-36)

Copy Write is the complementary command to Copy Read. This command writes part or all of the copy buffer out to the Selected Drive starting at the specified start_sector for the number of sectors specified. If the copy buffer is empty or if you have issued any of the commands in the above "NOTE:" issuing this command will most likely write invalid data out to your diskette. You SHOULD issue this command immediately after the Copy Read command.

The start_sector for Copy Write does not have to be the same as Copy Read. This allows you to move sectors around to different locations. However, for disk copying purposes the start_sector and #_of_sectors must be the same on both commands for an exact copy.

```
    EXAMPLES:   SD 1 CR 0 36  SD 2 CW  0 36   Copies sectors 0 through 35 from drive 1 to sectors
                                              0 through 35 on drive 2.

                SD 1 CR 0 18   SD 2 CW 18 18  Copies sectors 0 through 17 from drive 1 to sectors
                                              18 through 35 on drive 2.

                CR 2 32    PA     CW 300 33   Copies sectors  2 through 33 on the Selected  Drive
                                              to  sectors 300 through  332 on the  same drive but
                                              the PA  allows you  to switch  diskettes before  it
                                              writes out the sectors.
```

THE COMMANDS Continued

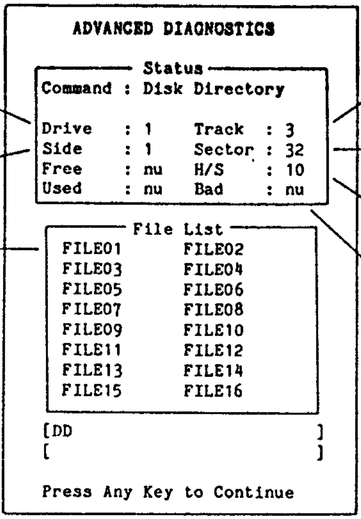
DISK DIRECTORY or DD

The Disk Directory command will list the file names on the diskette, in the Selected Drive, in the Display Window area of the Command Screen. This command will display 16 files names at a time and then wait for a key press to continue.

If you want to know the location of a file or its size, type etc. use the Find File command.

EXAMPLE: **Disk Directory** or DD will list the files names in the Display Window.

NOTE: Since the Disk Directory command usually requires a number of key presses to complete the command, you should included a few space characters in your Command File(s) after the DD command to compensate for this.



```
ADVANCED DIAGNOSTICS

------------- Status -------------
Command : Disk Directory

Drive   : 1     Track   : 3
Side    : 1     Sector  : 32
Free    : nu    H/S     : 10
Used    : nu    Bad     : nu

---------- File List ----------
  FILE01       FILE02
  FILE03       FILE04
  FILE05       FILE06
  FILE07       FILE08
  FILE09       FILE10
  FILE11       FILE12
  FILE13       FILE14
  FILE15       FILE16

[DD                             ]
[                               ]

Press Any Key to Continue
```

The Drive from where the Directory was read.

The Side where the last File Header was located.

The File Names are displayed in this area 16 at a time. The "Press Any Key....." message will be displayed at the bottom of the screen until the command is completed.

The Track where the last File Header was located.

The Sector where the last File Header was located.

The Head Step time used.

If you want the Free, Used and Bad sectors on this diskette use the Check Disk command.

THE COMMANDS Continued

EDIT SECTOR sector#(0-?) or ES sector#(0-?)

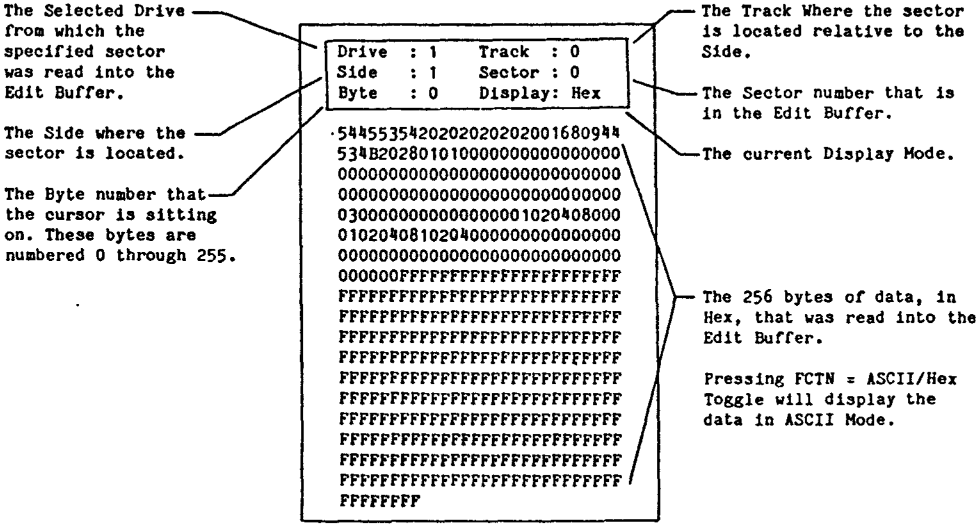
The Edit Sector command allows you to edit the contents of any valid sector on the disk. After the command has read the sector into the Edit Buffer it will be displayed on the screen along with a Status Window. The Status Window has indicators for Drive, Side, Byte, Track, Sector and Display Mode. The Drive, Side, Track and Sector indicators display information about the location of this sector. The Display Mode indicator indicates the display mode, which is selected by the FCTN = ASCII/Hex toggle. The Byte indicator is updated as you move the cursor around the screen and it indicates which byte number the cursor is sitting on.

The FCTN E, S, D and X keys move the cursor around. FCTN 9 BACK takes you back to the Command Screen. FCTN 1 DELETE and FCTN 2 INSERT will Delete and Insert characters into the sector. FCTN 3 ERASE will erase the contents of the Edit Buffer. Pressing FCTN 3 ERASE in ASCII Mode fills the Edit Buffer with space characters and in Hex Mode it fills the Buffer with zeros. (see Quick Reference - FCTN & CTRL Keys for a complete listing of the the other functions). In Hex Mode the active keys for inputting or changing data are 0 through 9 and UPPER CASE A through F. In ASCII Mode any of the keys on the keyboard, except the function keys in the Quick Reference section of the manual, may be used to input or change data.

EXAMPLES: Edit Sector 34	reads sector number 34 into the Edit Buffer and displays the following screen for editing the data.
Edit Sector >22	reads sector number >22 (34) into the Edit Buffer for editing.

NOTE: After you have edited a sector you must press FCTN 9 BACK and then execute the WRITE SECTOR command to write your sector changes to the diskette.

When a Command File is first executed it will automatically set the Display Mode to Hex. That way you know, when you execute Edit Sector from a Command File, which Display Mode you are in and you can easily change it with the [5] key code. It will then stay in that Mode until you run another Command File, execute CF REDO or execute the [5] key code.



```
Drive   : 1    Track  : 0
Side    : 1    Sector : 0
Byte    : 0    Display: Hex

·5445535420202020202001680944
534B202801010000000000000000
0000000000000000000000000000
0000000000000000000000000000
0300000000000000001020408000
0102040810204000000000000000
0000000000000000000000000000
000000FFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFF
```

The Selected Drive from which the specified sector was read into the Edit Buffer.

The Side where the sector is located.

The Byte number that the cursor is sitting on. These bytes are numbered 0 through 255.

The Track Where the sector is located relative to the Side.

The Sector number that is in the Edit Buffer.

The current Display Mode.

The 256 bytes of data, in Hex, that was read into the Edit Buffer.

Pressing FCTN = ASCII/Hex Toggle will display the data in ASCII Mode.

THE COMMANDS Continued

FORMAT DISK #sides(1-2) density(S-D) diskname or FD #sides(1-2) density(S-D) diskname

The Format Disk command formats the entire diskette in the SELECTED DRIVE with the number of sides, density and disk name as specified in the command. This format routine formats a diskette that is fully compatible with the Disk Manager, the various TI Languages and other Command Modules.

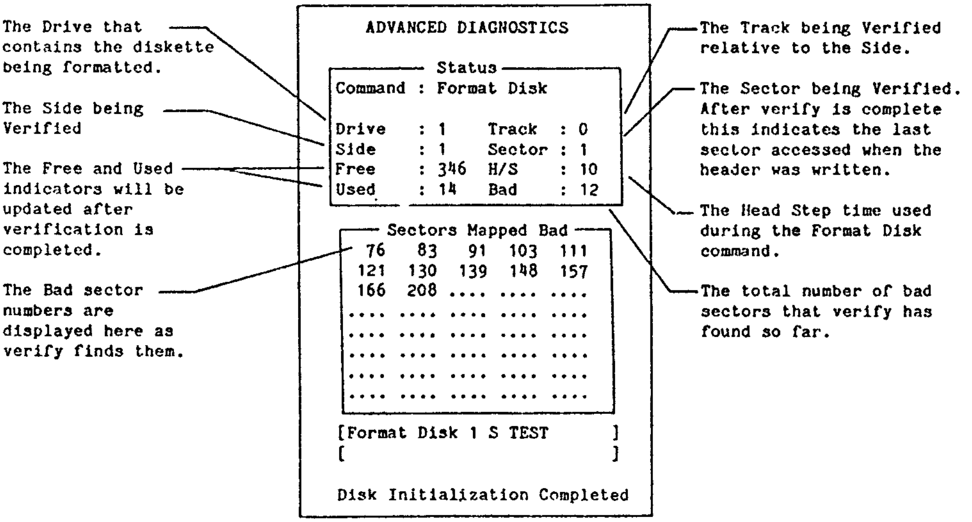
Provisions have also been implemented through USE DSR (see) to allow formatting with other Disk Controller cards or other devices that have DSR headers that incorporate a Format subprogram. If USE DSR is not turned on then Diags will use its own internal format routine otherwise it uses the format routine in the Disk Controller card.

You can format just selected tracks on a diskette by using the Write Track command and then with the Edit Sector command you can build a proper header for the disk on the sectors 0 and 1. (see Write Track and Command File - DISKHEAD)

EXAMPLES:		
	FD 1 S TEST	Formats a single sided, single density, 360 sector diskette in the SELECTED DRIVE with the disk name of TEST.
	FD 1 D DISK-01	Formats a single sided, double density, 720 sector diskette in the SELECTED DRIVE with the disk name of DISK-01.
	FD 2 D XBFILES	Formats a double sided, double density, 1440 sector diskette in the SELECTED DRIVE with the disk name of XBFILES.

NOTE: Pressing any key during Verify will abort the verify and a standard header will NOT be written out to the disk.

Double Density disk accesses can only be done with a Double Density Disk Controller Card. The TI Disk Controller Card will NOT access Double Density.



```
          ADVANCED DIAGNOSTICS

   ------------ Status ------------
   Command : Format Disk

   Drive   : 1      Track  : 0
   Side    : 1      Sector : 1
   Free    : 346    H/S    : 10
   Used    : 14     Bad    : 12
   --------------------------------
     ----- Sectors Mapped Bad -----
      76   83   91  103  111
     121  130  139  148  157
     166  208 .... .... ....
     .... .... .... .... ....
     .... .... .... .... ....
     .... .... .... .... ....
     .... .... .... .... ....
     .... .... .... .... ....

   [Format Disk 1 S TEST          ]
   [                              ]

   Disk Initialization Completed
```

The Drive that contains the diskette being formatted.

The Side being Verified

The Free and Used indicators will be updated after verification is completed.

The Bad sector numbers are displayed here as verify finds them.

The Track being Verified relative to the Side.

The Sector being Verified. After verify is complete this indicates the last sector accessed when the header was written.

The Head Step time used during the Format Disk command.

The total number of bad sectors that verify has found so far.

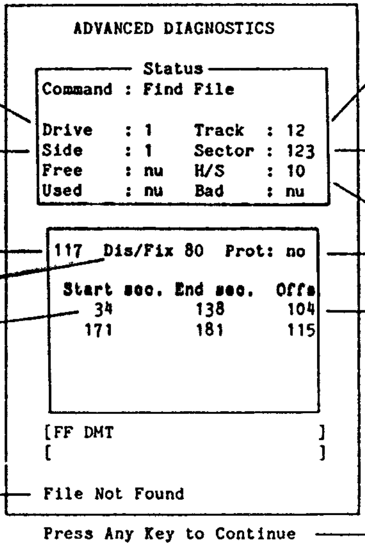
FIND FILE filename or FF filename

The Find File command displays the file's size, type, write protection, sector locations and total offset in the Display Window. Upon completion of the command, the Side, Track and Sector indicators point to the location of the File Header (File Descriptor Record) on the Selected Drive.

If the file does not exist on the Selected Drive this command will display the "File Not Found" error message at the bottom of the screen. If you would like a directory listing of the files on the Selected Drive use the Disk Directory command.

EXAMPLES: Find File LOAD will display the information for the file named LOAD.

FF SEEKTEST displays the information for the file named SEEKTEST.



```
ADVANCED DIAGNOSTICS

---------- Status ----------
Command : Find File

Drive   : 1    Track  : 12
Side    : 1    Sector : 123
Free    : nu   H/S    : 10
Used    : nu   Bad    : nu

117  Dis/Fix 80  Prot: no

 Start sec. End sec.  Offs
    34        138      104
   171        181      115

[FF DMT                        ]
[                              ]

File Not Found

Press Any Key to Continue
```

The Selected Drive to look on for the file.

The Side where the File Header was found.

The Catalog size.

The File Type.

Sector cluster positions and the files total offset.

This is displayed if the File can not be found.

The Track where the File Header was found relative to the Side.

The Sector where the File Header was found.

The Head Step time used.

Write Protection (yes/no)

If a File is Fractured it will have more than one sector cluster, such as this one.

This is displayed if the File is badly Fractured and all of the sector clusters can not be shown in one Display Window.

THE COMMANDS Continued

HEAD STEP step_time or HS step_time

The Head Step command sets the head step time in milliseconds for all commands that require the head to step in or out, such as Edit Sector, Seek Track, Format Disk etc. This Head Step time is used whenever USE DSR is turned OFF (Not Using DSR). When use DSR is ON, the head step time used is determined by the Disk Controller card setting. A good test for proper Head Step time is included on the Advanced Diagnostics disk. To test different times, input the desired time and run the Command File named SEEKTEST with the command "Command File DSK1.SEEKTEST". If the time selected is too fast diags will display error messages that end with ".... or Bad H/S". If this happens selected a slower Head Step time and try again with the command "Command File REDO". Once you have found a good time for your drives you might want to include it in your DIAGCONFIG file. Also see "Intro to Floppy Disk Controllers" for more information on Head Step times.

TI Head Step Times: 6, 12, 20 or 30 CorComp Head Step Times: 3, 6, 10 or 15

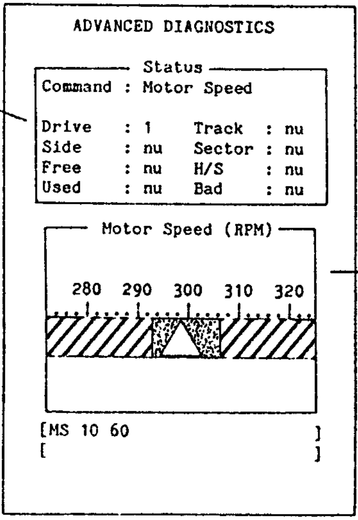
MOTOR SPEED sample_rate Hz(50-60) or MS sample_rate Hz(50-60)

The Motor Speed command will check and display the disk drive's RPM for the Selected Drive on a Bar Graph type scale. This command has two parameters, the first one, sample_rate, allows you to select the number of index pluses for calculating the RPM. A high sample rate number will give you the average RPM for the drive. A small sample rate will update the pointer more often. There are approx 5 index pluses per second, so if you choose a large sample rate Diags will not check for a key press to abort, very often. In a Command File you should place a few [255] or [254] key-codes after the [13] to keep this test going. You should also have an extra space character after these key-codes to abort this test. The second parameter allows Diags to handle the cycles per second, Hz, for the different AC currents around the world. For the United States use 60 for this parameter, for other countries use 50 or 60 depending on your AC current.

EXAMPLES: Motor Speed 10 60 will execute the Motor Speed test for U.S. current, 60 cycles per second, at a sample rate of 10 index pluses.

Motor Speed 8 50 will execute the Motor speed test for 50 cycles per second AC current and at a sample rate of 8 index pluses.

The Drive that the Motor Speed command is testing.



```
ADVANCED DIAGNOSTICS

Status
Command : Motor Speed

Drive   : 1     Track  : nu
Side    : nu    Sector : nu
Free    : nu    H/S    : nu
Used    : nu    Bad    : nu

Motor Speed (RPM)

[MS 10 60                      ]
[                              ]
```

The Drives RPM is pointed to by the white arrow. If the point of the arrow stays in the Green area of the Bar, your drive is OK. If it is in the Red, your drive needs its Motor Speed adjusted to perform properly.
(300 is the optimum RPM)

THE COMMANDS Continued

OUTPUT DEVICE device_name or OD device_name

The Output Device command sets the device to dump the screen out to with the FCTN 0 SCREEN DUMP function. Any valid device may be used such as DSK, RS232, PIO, TP etc. (except CS1 & 2). Once the Output Device has been set all FCTN 0's will dump the contents of the screen to it. If you choose a disk file (DSK1.filename) then every FCTN 0 will dump the screen to the same file, which will write over whatever was in the file, so choose your DSK filenames carefully. The FCTN 0 SCREEN DUMP dumps the screen in a DIS/VAR format. (see Output Width)

EXAMPLES: Output Device RS232.BA=1200.DA=8 sets the output device as RS232 port 1 with a baud rate of 1200 and 8 data bits.

Output Device DSK1.SCREEN sets the Output Device as a DIS/VAR file named SCREEN on drive 1.

OUTPUT WIDTH line_length(1-80) or OW line_length(1-80)

The Output Width command allows you to change the output line length from the default of 80. The valid range for this command is 1 through 80. If you select a narrow width, such as 16, the FCTN 0 screen dump function will only dump the left hand half of the screen. The main purpose for this command is to allow screen dumps to a narrow device like the TI Thermal Printer. You may want to include this command in your DIAGCONFIG command file if your Output Device has a narrower line length than 80.

EXAMPLE: Output Width 40 sets the line length to 40 for a DIS/VAR 40 type file when the FCTN 0 Screen Dump is activated.

PAUSE or PA

The Pause command prevents the commands that follow it from being executed until a key is pressed. When the Pause command is executed the message "Press Any Key to Continue" will appear at the bottom of the screen. This command is very useful in command files for two reasons. First it allows you stop the command file for disk swapping purposes or verification to continue. Secondly, it is the only way to halt and abort a command file that is running in high speed.

If you want to abort a multiple command line or a Command File when the Pause command is executed just press FCTN 5 BEGIN. This will take you to the title screen and abort command processing. From the title screen, when you press any key, you will return to the command screen and your command line and/or Command File will still be intact. If you want to re-execute your command line just press ENTER. If you want to rerun your Command File type in COMMAND FILE REDO or CF REDO and press ENTER.

NOTE: If you use the Pause command in a Command File you must press a key from the keyboard. The characters and key codes that follow it in the Command File do not act as a key press.

FCTN 7 HELP and FCTN 0 SCREEN DUMP are active and they will not act as a key press for continuing the execution of the commands.

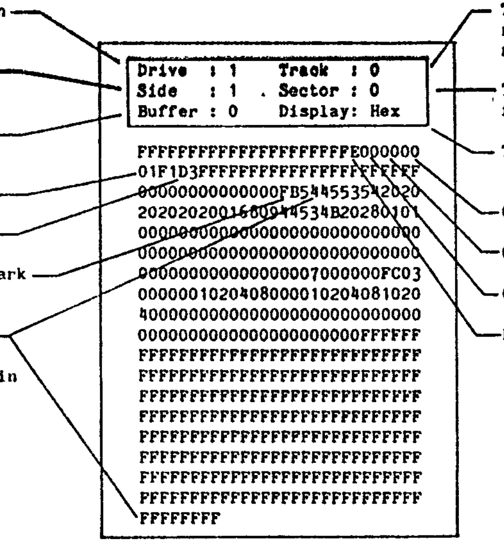
EXAMPLE: PAUSE SD 1 FD 1 S DISK1 PAUSE SD 2 FD 1 S DISK2 this multiple command line will wait for a key press, format the diskette in drive 1 and then wait for another key press before formatting the diskette in drive 2.

READ TRACK track#(0-79) or RT track#(0-79)

The Read Track command allows you to analyze the information that is written to the specified track when it is formatted and resides inbetween the sectors. This information includes the Gap, Sync, Address, Data and Cyclic Redundancy Check (CRC) Bytes. The items of most interest are the Address bytes which include the ID Address Mark, Track Number, Side Number, Sector Number, Sector Length and 2 CRC Bytes. The ID Address Mark byte is an >FE followed by the other bytes. This byte tells the Floppy disk Controller Chip (FDC) that the next 6 bytes contain the address information. The 2 CRC Bytes are used by the Floppy Disk Controller Chip for checking the integrity of the data it has just read in.

You may notice when Rolling Up, FCTN 4, and Rolling Down, FCTN 6 through the tracks that some of the Gap and Data information displayed on the screen is incomplete. This happens because of the state that the FDC is in during a Read Track and it can also be caused by noise or write-splice time synchronization. In this state, if the FDC detects certain byte values it interprets these as being an ID Address Mark or other special Mark and assembles or collects the bytes following it as something other than data. Do not be concerned if this occurs during a Read Track since this command is NOT used during normal operation of the TI 99/4A disk system it is only used for diagnostic purposes. Most of the information displayed on your screen, except the 256 bytes of data, is used by the FDC for synchronization, error checking and, identification and verification of the location of the Read/Write head. (see Introduction to Floppy Disk Controllers, Seek Track and Write Track for more information).

EXAMPLE: Read Track 0 — steps the Read/Write head to track 0, reads the data on that track into the Diags track buffer. CTRL E and CTRL X can be used to page through the buffer to display different portions of the track data. FCTN = will toggle the display between ASCII and Hex.



```
Drive  : 1     Track  : 0
Side   : 1   . Sector : 0
Buffer : 0     Display: Hex

FFFFFFFFFFFFFFFFFFFFFE000000
01F1D3FFFFFFFFFFFFFFFFFFFFFF
00000000000000FB544553542020
2020202001680944534B20280101
0000000000000000000000000000
0000000000000000000000000000
000000000000000007000000FC03
00000010204080000010204081020
4000000000000000000000000000
0000000000000000000000FFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFFFFFFFFFFFFFFFFFFFFFF
FFFFFFFF
```

The Drive from which the track was read.

The Side where the track is located.

The current Buffer being displayed.

01 = Sector Length

F1D3 = CRC Bytes

FB = Data Address Mark

5445 through FFFF is part of the data contained in Sector 0 on Track 0 of Side 1.

The Track number that was read into the buffer relative to the Side.

The starting sector number for this track.

The current Display Mode.

00 = Sector Address

00 = Side Number

00 = Track Address

FE = ID Address Mark

NOTE: The Data Displayed here is from a single sided single density Test diskette. For information on double density format see Introduction to Floppy Disk Controllers.

SEEK TRACK track#(0-79) or ST track#(0-79)

The Seek Track command instructs the Selected Drive to step the Read/Write head to the specified track on the diskette. Tracks 0-39 access side 1 and tracks 40-79 access side 2. This command tests the stepping function of your disk drives at the selected Head Step time. Upon completion of this command the Side, Track and Sector indicators in the Status Window are updated.

The Side indicator displays the side number where the track is located. The Sector indicator displays the lowest, or starting, sector number, relative to the density, on the track. . The Track indicator displays the track number relative to the side. Both sides of the diskette contain tracks that are numbered from 0 to 39 so specifying a track number greater than 39 will step the head to a track on side 2 of the diskette. If you want to step to a track on side two just add 40 to the valid track number. (see Introduction to Floppy Disk Controllers, Read Track and Write Track for more information).

EXAMPLES: **Seek Track 23** steps the head out to track 23 on side 1.

Seek Track 63 steps the head out to track 23 on side 2.

NOTE: Also see Command File - SEEKTEST

SELECT DRIVE drive#(1-5) or SD drive#(1-5)

The Select Drive command sets the disk drive number for all commands to operate on except Command File DSKx.filename and FCTN 0 Screen Dump. The valid range for this drive selection is 1 through 5. The 5th drive is to allow for possible future expansion of your system with a RAM Disk that is known as Drive 5. (see USE DSR)

EXAMPLE: Select Drive 2 sets the drive for the various commands to Drive 2.

NOTE: Once the drive is selected all commands that execute a Write operation, such as Format Disk, Write Sector and Write Track will write to that drive. Be VERY CAREFUL of your drive selection before executing any of these commands or you many inadvertently write over an important diskette.

TIME DELAY #_of_seconds or TD #_of_seconds

The Time Delay command allows you to halt command execution for the specified number of seconds. This command can be used to allow for diskette swaps in a single drive system or it can be used to allow the disk drive motor to time out (stop turning).

EXAMPLES: **Time Delay 4** causes a delay of 4 seconds.

TD >A cause a delay of >A (10) seconds.

NOTE: For a time delay in a Command File, without executing the Time Delay command, you can include the [253] delay function. (see Command File for more information.)

USE DSR or UD

The Use DSR command instructs Advanced Diagnostics to use the Sector Read/Write and Format subroutines that are built into the Disk Controller or other device's DSR ROM. This command is the only command that has a toggling action. The first time it is executed the message "Using DSR ROM" will be displayed at the bottom of the screen. The next time it is executed the message "Not Using DSR ROM" will be displayed.

This command serves two purposes. First it allows testing of the DSR ROM and a couple of its subroutines. And, secondly it allows Advanced Diagnostics to work with any future device, such as a RAM Disk, that has a standard DSR Header with the Sector I/O and/or Format subroutines in it for that device.

When Use DSR is turned ON, "Using DSR ROM", some of the Status Window indicators are not updated such as Track and Side. This is because a RAM Disk does not have Tracks or Sides only sections of memory addressed as Sectors, so these indicators would be invalid.

EXAMPLE: Use DSR toggles the DSR ROM ON or OFF.

WRITE SECTOR sector#(0-?) or WS sector#(0-?)

The Write Sector command is the complimentary command to Edit Sector. This command writes whatever is in the Edit Buffer to the specified sector number on the Selected Drive. Be VERY CAREFUL when you issue this command and make sure that the SELECTED DRIVE is the drive you want to write to!

EXAMPLE: Write Sector 0 writes whatever is in the Edit Buffer onto sector 0 in the Selected Drive.

NOTE: The Edit Buffer is also used by the Format Disk and Read Track commands. You must issue a WRITE SECTOR command before any of these commands or the data in the buffer will be DESTROYED! Copy Read does not destroy the Edit Buffer.

WRITE TRACK track#(0-79) density(S-D) or WT track#(0-79) density(S-D)

The Write Track command writes standard format information to the specified track number in the specified density. Tracks 0-39 access side 1 and tracks 40-79 access side 2. This command can be used to format just part of a diskette. Since it does not write a standard header to sector 0, you will have to generate a header that maps the formatted sectors into the Allocation Bit Map.

Part of a diskette can be formatted in single density and part of it in double density, if you have a double density disk controller card. If you should do this keep in mind that all of the relative addressing that the Sector I/O commands and Disk Controller DSR subroutines perform will be out of order. This happens because the track location of a sector is computed by dividing the number of sectors per track (9 for single density and 18 for double density) by the sector number specified.

For example, lets say you format tracks 0 and 1 as single density, track 2 as double density and track 3 as single density. Sectors 0 through 17 will be on tracks 0 and 1 and sectors 27 through 35 will be on track 3, like they are suppose to be. However, because of the relative addressing that takes place in the sector I/O routines, sectors 36 through 53 will be on track 2. The relative addressing says that track 2 is double density so the first sector on that track must be 36 (18 x 2 = 36). This will create havoc with the Disk Controller when sectors are searched for in numerical order. You can however Edit these sectors even though they are in the wrong location relative to the other tracks, but you can not Roll Up or Roll Down to them.

EXAMPLE: Write Track 1 S formats track 1 as single density.

INTRODUCTION TO FLOPPY DISK CONTROLLERS

The TI Disk Controller can control up to three 5 1/4 inch floppy disk drives. Each drive may be single or double sided. Since this controller works in single density there are 40 tracks (0-39) per side and 9 sectors (0-8) per track on the diskette. The CorComp Disk Controller can control up to 4 drives and each drive may also be single or double sided. This controller will work in either single or double density. In double density, with this controller, there are 40 tracks (0-39) per side and 18 (0-17) sectors per track on the diskette.

With both controllers each sector is 256 bytes in length and is commonly referred to as 1 Allocation Unit. An allocation unit is the smallest unit of Data that can be Read or Written from the diskette. The position of the tracks on the diskette is controlled by the Floppy Disk Controller chip and with these two controllers track 0 is nearest the outside and track 39 is nearest the center of the diskette. This is true for both sides of the diskette.

These two Disk Controllers use a technique called soft sectoring when the disk is formatted. This approach uses software and an single index pulse to control the placement of data on a diskette. On a hard sectored diskette there are as many index holes as there are sectors and this determines the placement and size of each sector. On our soft sector systems the location of sectors is determined by an ID Address field. This field contains the Track Number, Side Number, Sector Number (0-8 or 0-17), Sector Length and CRC error checking bytes. (see Single and Double Density Format on pages 24 & 25).

Following the ID Address field are some separation bytes and then the Data field. These separation bytes serve two purposes. First they allow for slight variations in disk drive motor speed and secondly they prevent the read/write head's magnetic field from disturbing the ID Address field on sector writes. The Data field is also followed by some separation bytes. These two fields, ID and Data, along with the separation bytes are written to the track for each sector. So even though a single sided, single density diskette has 90K of DATA storage it actually contains many more bytes than that.

The Floppy Disk Controller Chip (FDC) controls the disk head movement, searches for the specified track and/or sector, converts parallel information from the console to serial data for the disk drive and converts serial data from the disk drive to parallel data for the Console. Since the disk drive head only reads and writes a bit at a time it is up to the FDC to assemble these bits into bytes. On a read 8 bits are assembled by the FDC and then transferred to the Console as a byte. When data is written to the diskette, a byte is passed to the FDC which in turn writes it out to the diskette a bit at a time. So, to read or write one sector of Data 2048 bits are transferred plus some additional bits for the Sync, Address Mark and CRC bytes.

When a read or write command is issued the FDC looks for an ID Address Mark. When found, the track address and sector address values are scanned for a match. If they do not match the search continues. If a match is found, a CRC check is made. If the CRC check fails the search continues. If the CRC check is OK then the FDC looks for the Data Address Mark. When found the data is read and the information is sent to the console a byte at a time. After the data is read a CRC check is made. If it fails an error message is sent to the console. In either case the command terminates and the controller tells the console the command is complete.

When a write is requested, the FDC, after finding the specified ID Address Mark field, will wait for the ID/DATA Separator field to pass and then write the sync bytes followed by the Data Address Mark, Data and CRC bytes. On a verify, Diags will read the last sector written and compare it to the buffer it was written from. If everything is OK Diags continues with the next operation. If is not OK it will try to verify it again, if it can't verify it Diags will return the "Bad Compare on a Write Verify" error message.

If the FDC can not find the proper ID field within 5 revolutions during a read or write it returns a Record Not Found error to Diags. Diags will then attempt the read or write 5 to 10 more times before it returns with an error. If this happens Diags will return the "Record Not Found on a Read" or "Record Not Found on a Write" error message.

Single density data is recorded on a diskette with a technique called Frequency Modulation (FM). This self clocking code resists phase shifts and clock variations. Each Data Mark (bit cell) contains a clock pulse, at the leading edge, and an optional data pulse. If a bit is 0 then the Data Mark does not contain a data pulse. If a bit is a 1 then the Data Mark contains a data pulse. (see FM diagram)

Double density data is recorded on a diskette with a technique called Modified Frequency Modulation (MFM). If a bit is 1 the Data Mark contains a data pulse. If a bit is 0 then the Data Mark does not contain a data pulse. If there are two Data Marks in a row that do not contain data pulses then a clock pulse is generated at the leading of the second Data Mark. (see MFM diagram)

The ID and Data Address Marks (see pages 24 and 25) are special bytes which do not follow the standard FM or MFM encoding technique. These data bytes have 2 clock pulses missing in the middle of each byte. They indicate to the controller that an ID or Data field has been detected and to byte synchronize the controller.

The single density disk format is shown on page 24. This information is written to the diskette when it is initialized. When a sector is rewritten (updated) only the section between Read/Write Sector and End Read/Write Sector is written. The format starts with an Index Gap of 12 bytes.

This is followed by the ID Sync bytes, ID Address Mark, Address and ID CRC Bytes. The start of the ID field has 6 zeros to synchronize the disk controller Data Separator to the bit level. These sync bytes are followed by an ID Address Mark (>FE) which specifies the start of the ID field. This field contains the track address, side number, sector address and length of the sector. This field is terminated with 2 Cyclic Redundancy Check bytes used to verify information correctness of the ID field. The ID field is followed by 11 gap bytes as a separator from the Data field. Due to disk drive motor speed variations this gap may vary when a sector is rewritten. Following the separator bytes is the Data field. This field is composed of 6 sync bytes, a Data Address Mark (>FB) and then 256 bytes of data. This field is also terminated by 2 bytes of CRC data followed by 36 bytes of >FF. Each data field is separated from the next ID field by these 36 bytes. This protects the following ID field after writing the previous sector. The information in this paragraph is written 9 times in each track, once per sector.

After the last sector, there is an additional gap of 240 >FF bytes. This gap may vary by +- 32 bytes due to motor speed variations when the diskette is initialized.

The double density disk format is shown on page 25. This format starts with an Index Gap of 32 >4E bytes.

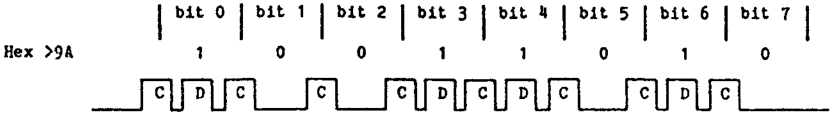
This is followed by the ID Sync bytes, ID Address Mark, Address and ID CRC Bytes. The start of the ID field has 12 zeros to synchronize the disk controller Data Separator to the bit level. These sync bytes are followed by three >A1 bytes which are necessary for MFM recording. Next comes an ID Address Mark (>FE) which specifies the start of the ID field. This field contains the track address, side number, sector address and length of the sector. This field is terminated with 2 Cyclic Redundancy Check bytes used to verify information correctness of the ID field. The ID field is followed by 22 gap bytes as a separator from the Data field. Due to disk drive motor speed variations this gap may vary when a sector is rewritten. Following the separator bytes is the Data field. This field is composed of 12 sync bytes, another three >A1 bytes, a Data Address Mark (>FB) and then 256 bytes of data. This field is also terminated by 2 bytes of CRC data followed by 28 bytes of >4E. Each data field is separated from the next ID field by these 28 bytes. This protects the following ID field after writing the previous sector. The information in this paragraph is written 18 times in each track, once per sector.

After the last sector, there is an additional gap of 190 >4E bytes. This gap may vary by +- 32 bytes due to motor speed variations when the diskette is initialized.

You can view both of these formats by using the Advanced Diagnostics Read Track command. The CTRL E and CTRL X keys will allow you the page through the different sections of the track (Buffer). FCTN = ASCII/Hex toggle will display the information in ASCII or Hexadecimal values.

INTRODUCTION TO FLOPPY DISK CONTROLLERS Continued

FM Recording

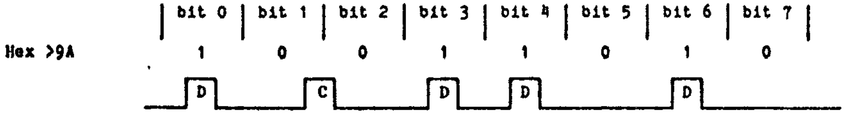


C = Clock bit   D = Data bit

Rules:

1. Write data bits at center of bit cell if the bit is on "1".
2. Write clock bits at leading edge of bit cell.

MFM Recording

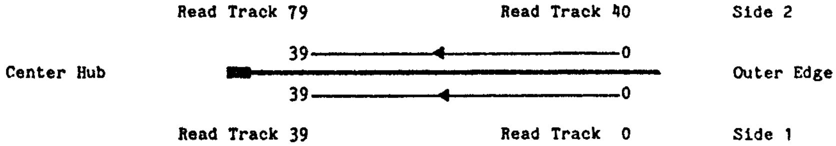


C = Clock bit   D = Data bit

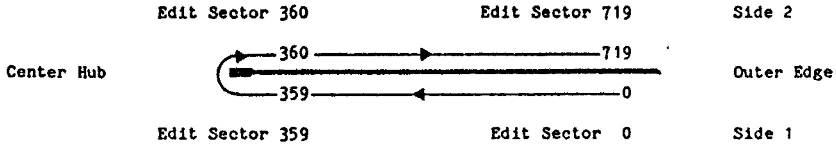
Rules:

1. Write data bits at center of bit cell if the bit is on "1".
2. Write clock bits at leading edge of bit cell IF: no data is in the previous bit cell and no data will be in the next bit cell.

Track Order on a Double Sided Disk



Sector Order on a Double Sided, Single Density Disk



SINGLE DENSITY DISK FORMAT (FM)

The following information is displayed on your monitor when a Read Track command is executed on a single density diskette by Advanced Diagnostics.

Start of Track

	Name	# of bytes	Read Track Info	Description
	INDEX GAP	12	>FF	Index mark filler.
	ID SYNC BYTES	6	>00	Synchronizes controller for ID Mark.
R	ID ADDRESS MARK	1	>FE	Identifies this as an Address Field.
e	TRACK ADDRESS	1	Track #	>00 through >27 = tracks 0 - 39.
p	SIDE NUMBER	1	Side #	>00 = side 1 >01 = side 2.
e	SECTOR ADDRESS	1	Sector #	>00 through >08 = sectors 0 - 8.
a	SECTOR LENGTH	1	>01	>01 = sector size of 256 bytes.
t	ID CRC	2	ID CRC Value	Used for error checking - ID Field.
e				
d	ID/DATA SEPARATOR	11	>FF	Used to prevent Write Sector from erasing part of the ID Field.
9	Read/Write Sector			
	DATA SYNC BYTES	6	>00	Synchronizes controller for Data Mark
T	DATA ADDRESS MARK	1	>FB	Identifies this as a Data Field.
i	DATA	256	File Data	This is the File Data - (Edit Sector)
m	DATA CRC	2	Data CRC Value	Used for error checking - Data Field
e	End Read/Write Sector			
s				
	DATA/ID SEPARATOR	36	>FF	Used to prevent Write Sector from erasing part of the ID Field.
	END OF TRACK FILLER	240	>FF	# of bytes varies with Drive Motor speed.

End of Track

325 bytes x 9 Sectors =	2925
2925 + INDEX GAP + END OF TRACK =	3177 bytes total per track
256 bytes x 9 Sectors =	2304 bytes of data storage per track

NOTE: Due to Write-splice time synchronization and/or noise the bytes in the INDEX GAP, ID/DATA SEPARATOR, DATA, DATA/ID SEPARATOR and END OF TRACK FILLER may vary from Read to Read. However, The ID ADDRESS MARK and the ADDRESS information will be correct. (Also see Read Track in the COMMANDS section of this manual)

INTRODUCTION TO FLOPPY DISK CONTROLLERS Continued

DOUBLE DENSITY DISK FORMAT (MFM)

The following information is displayed on your monitor when a Read Track command is executed on a double density diskette by Advanced Diagnostics.

Start of Track

	Name	# of bytes	Read Track Info	Description
	INDEX GAP	32	>4E	Index mark filler.
Repeated 18 Times	ID SYNC BYTES	12	>00	Synchronizes controller for ID Mark.
		3	>A1	For MFM only.
	ID ADDRESS MARK	1	>FE	Identifies this as an Address Field.
	TRACK ADDRESS	1	Track #	>00 through >27 = tracks 0 - 39.
	SIDE NUMBER	1	Side #	>00 = side 1 >01 = side 2.
	SECTOR ADDRESS	1	Sector #	>00 through >11 = sectors 0 - 17.
	SECTOR LENGTH	1	>01	>01 = sector size of 256 bytes.
	ID CRC	2	ID CRC Value	Used for error checking - ID Field.
	ID/DATA SEPARATOR	22	>4E	Used to prevent Write Sector from erasing part of the ID Field.
	Read/Write Sector			
	DATA SYNC BYTES	12	>00	Synchronizes controller for Data Mark
		3	>A1	For MFM only.
	DATA ADDRESS MARK	1	>FB	Identifies this as a Data Field.
	DATA	256	File Data	This is the File Data - (Edit Sector)
	DATA CRC	2	Data CRC Value	Used for error checking - Data Field
	End Read/Write Sector			
	DATA/ID SEPARATOR	28	>4E	Used to prevent Write Sector from erasing part of the ID Field.
	END OF TRACK FILLER	190	>4E	# of bytes varies with Drive Motor speed.

End of Track

346 bytes x 18 Sectors = 6228
6228 + INDEX GAP + END OF TRACK = 6450 bytes total per track

256 bytes x 18 Sectors = 4608 bytes of data storage per track

NOTE: Due to Write-splice time synchronization and/or noise the bytes in the INDEX GAP, ID/DATA SEPARATOR, DATA, DATA/ID SEPARATOR and END OF TRACK FILLER may vary from Read to Read. However, The ID ADDRESS MARK and the ADDRESS information will be correct. (Also see Read Track in the COMMANDS section of this manual)

INTRODUCTION TO FLOPPY DISK CONTROLLERS Continued

Head Step Times

Head Step Time is the duration of the track stepping pulses that are sent to the disk drive's head stepping motor from the floppy disk controller chip. These pulses are generally programmable in two ranges, depending on the clock rate for the chip. For a 1 MHz clock, such as the TI card, the pulses can be 6, 12, 20 or 30 milliseconds in length. For a 2 MHz clock the pulses can be 3, 6, 10 or 15 milliseconds in length.

If the pulses are too long, 15 or 30 ms, the disk drive stepping motor will chatter when it steps. This chattering can cause excessive wear on the motor and might cause it to prematurely fail. If the pulses are too short, 3 or 6 ms, the stepping motor may not step at all or may only partially step to the specified track. Usually when the step rate is too short Advanced Diagnostics will return an error message that has "..or Bad H/S" at the end of it. If this occurs use the Head Step command to select a new head step time for Diags to use.

Most disk drives will work fine at 10 ms. Some of the newer double sided half high drives will work best at 6 ms or 3 ms. The SEEKTEST Command File is a good test to see if the Head Step time selected is proper for your drives. As the Read/Write head steps in and out you should hear either a clean crisp clicking sound or very little, if any, sound for fast step times.

Set Head Step Time Guide

PROBLEM	CAUSE	SOLUTION
Drive will not Seek, Read or Write. Stepping Motor chatters a lot. Head does not step.	Head Step Time is set too fast. (3 or 6 ms)	Select a slower Head Step Time (10 or 12 ms) and try the SEEKTEST again.
Drive may Read and Write but with a lot of bad sectors on a Format Verify. Drive sounds good.	Head Step Time is set too fast.	Select a slower Head Step Time and reformat a diskette.
Drive Seeks, Reads and Writes OK - no errors. When the Head Steps the drive sounds good. no chattering.	Head Step Time is set correctly.	----------------- -
Drive Seeks, Reads and Writes OK - no errors - but the Drive chatters some when the head steps.	Head Step Time is set too slow. (15 or 30 ms)	Select a faster Head Step Time (10 or 20 ms) and try the SEEKTEST again.

Sector Interlace or Interleave

Sector Interlace or Interleave is the physical order of the sectors within a track. On every track the sectors are numbered 0 - 8 (>00 - >08) for single density and 0 - 17 (>00 - >11) for double density. The Interlace pattern determines where these sectors are on the track. They usually are not in numerical order (012345678) but are instead spaced around the track in a known pattern. Lets look at the Diags single density interlace pattern of 0 7 5 3 1 8 6 4 2. This pattern tells us that after reading sector 0, three sectors (7 5 3) will pass under the read write head before sector 1 is in position to be read. Our timing test have shown us that in single density this interlace pattern provides the best results. While those three sectors are passing under the read/write head the Disk Controller DSR and Console are passing the bytes from sector 0. So, after the data has been passed, sector 1 is in position and we do not have to wait for the diskette to go all the way around once more before we can read sector 1.

Diags Single Density Interlace 0 7 5 3 1 8 6 4 2

Diags Double Density Interlace 0 11 4 15 8 1 12 5 16 9 2 13 6 17 10 3 14 7

APPENDIX - A Advanced Diagnostics Floppy Disk Controller and DSR Error Messages.

When one of these Errors are generated by Advanced Diagnostics check the following items: Drive door open - No diskette in drive - Selected Drive number does not exist in your system Drive power not turned on - Not initialized or Bad floppy diskette - Ribbon and/or power cables connected to the selected drive are bad - Head Step time set wrong for the selected drive - Possible bad stepping motor in drive - Diskette not turning (bad motor or bad floppy) - Diskette turning too fast or too slow, check Motor Speed - Dirty or clogged Read/Write head - Cables, Dip Shunts, Termination Resistor packs wrong on a new drive installation.

FDC Error Message	Indicates
No Diskette/Drive or Bad H/S	The disk drive has not set the ready line to the proper state (device not ready) or the Head Step time is too fast.
Seek Error or Bad H/S	The desired Track could not be found when the read/write head was stepping in or out or the Head Step time is too fast.
Record Not Found on a Read Record Not Found on a Write	The desired Track, Sector or Side could not be found on a Read or Write operation.
Lost Data on a Read Lost Data on a Write	A data byte was lost (not ready to be transferred) on a Read Sector, Write Sector, Read Address, Read Track, Write Track command.
Crc Error on a Read	An error was found in one or more of the ID fields or an error was found in the data on a Read Sector, Read Track or Read Address command.
Write Protected Disk	The write protect notch on the diskette is covered up.

DSR Error Message	Indicates
Bad DSR Parameter(s)	An out of range value was received for a Track number, Sector number, Density or the Selected Drive was not found.
Bad Device Name	An nonexistent device name was received.
Bad Open Attribute	One of the Open Attributes (file type, file name, protected file etc.) were wrong to match any of the files on the diskette. This error usually shows up when a wrong Command File name is input.
Illegal Operation	An Command File Input was requested from an invalid device for Input such as the Thermal Printer.
Out of Table or Buffer Space	The diskette is full or there are already 127 files on it.
Bad Compare on a Write Verify	The data in the Write buffer does not compare with the data that was written to the diskette.
Device Error	The diskette is not initialized or damaged, the power has been interrupted to the Peripherals, or the drive is bad.
File Access Error	The Command File requested is not a DIS/VAR 80 type file.

APPENDIX - B Advanced Diagnostics Command Error Messages

When one of these Errors are generated by Advanced Diagnostics you should check the proper syntax, spacing and range limits for the command.

Diags Error Message	Indicates
Bad command	The spelling or abbreviation for the command is wrong.
Parameter Not Found	One or more of the parameters for the command that the cursor stopped on is missing a space or is missing entirely.
Bad Parameter(s)	One or more of the parameters for the command that the cursor stopped on is incorrect or out of range.
Bad Drive Number	The drive number Selected is out of the 1 - 5 range.
Bad Step Rate	The Head Step time selected is not a valid step time for the Disk Controller Card in your system.
Bad Color(s)	One or more of the colors selected is not valid.
Bad Line Size	The Output Width in not in the 1 - 80 range.
Number of Sectors Too Large	The number of sectors for the Copy Read or Copy Write command is not in the 1 - 36 range.
File Not Found	The file requested for Find File is not on the Selected Drive.

NOTE: Advanced Diagnostics will automatically place the cursor at the beginning of the command that caused the error.

SECTOR 0 - THE VOLUME INFORMATION BLOCK - VIB

Byte Nos.	Contents
0 - 9	**10 Character (max) Disk Name or Volume Name** Unused characters = space character
10 - 11	**Total Number of Formatted Sectors.** On the TI 4A system 1 sector is often refered to as 1 Allocation Unit or AU. >0168 = SS/SD 360 sectors >02D0 = SS/DD 720 sectors >02D0 = DS/SD 720 sectors >05A0 = DS/DD 1440 sectors
12	**Number of Sectors per Track** >09 = 9 sectors per track >12 = 18 sectors per track
13 - 15	**DSK** - The letters DSK are used to signify to the disk manager that the disk has been initialized by a TI or compatible disk controller.
16	
17	**Number of Tracks per Side** >23 = 35 tracks per side >28 = 40 tracks per side
18	**Number of Sides** >01 = single sided SS >02 = double sided DS
19	**Diskette Density** >01 = single density SD >02 = double density DD
20 - 55	These bytes are reserved for future expansion. This is currently an unused area - all 0's
56 - 255	Allocation Bit Map - each bit represents 1 sector or AU - there are enough bits in this 200 byte area to map out 1600 256 byte sectors or 400K of storage, but currently only 360K or 1440 bits are used by a Double Density disk. 56 - 100 = are used by SS/SD - sectors 0 - 359 101 - 145 = are used by SS/DD - sectors 360 - 719 101 - 145 = are used by DS/SD - sectors 360 - 719 146 - 235 = are used by DS/DD - sectors 720 - 1439 236 - 255 = are currently not used by any format. Note: After formatting, the unformatted sectors are marked as used, all bits on. (Byte no. - 56) * 8 = start of 8 sector group. ie: (56 - 56) * 8 = 0 sector numbers 0-7 (73 - 56) * 8 = 136 sector numbers 136-143 Sector No./8 + 56 = Byte no. remainder = Bit no. ie: 137/8 + 56 = 73.125 .125 * 8 = bit 1 34/8 + 56 = 4.25 .25 * 8 = bit 2 Each byte (8 bits) represents 8 sectors, one bit per sector. The bits are numbered from right to left and if the bit is on, 1, the sector is used. bit nos - 7 6 5 4 3 2 1 0 byte no 73 = 143 142 141 140 139 138 137 136 byte no 100 = 359 358 357 356 355 354 353 352 byte no 231 = 1407 1406 1405 1404 1403 1402 1401 1400

APPENDIX - D SECTOR 0 - ALLOCATION BIT MAP TABLE

SS/SD
Sectors 0 - 359

Hex Value	>80	>40	>20	>10	>08	>04	>02	>01
Bit Number	7	6	5	4	3	2	1	0
Byte No.								
56 -	7	6	5	4	3	2	1	0
57 -	15	14	13	12	11	10	9	8
58 -	23	22	21	20	19	18	17	16
59 -	31	30	29	28	27	26	25	24
60 -	39	38	37	36	35	34	33	32
61 -	47	46	45	44	43	42	41	40
62 -	55	54	53	52	51	50	49	48
63 -	63	62	61	60	59	58	57	56
64 -	71	70	69	68	67	66	65	64
65 -	79	78	77	76	75	74	73	72
66 -	87	86	85	84	83	82	81	80
67 -	95	94	93	92	91	90	89	88
68 -	103	102	101	100	99	98	97	96
69 -	111	110	109	108	107	106	105	104
70 -	119	118	117	116	115	114	113	112
71 -	127	126	125	124	123	122	121	120
72 -	135	134	133	132	131	130	129	128
73 -	143	142	141	140	139	138	137	136
74 -	151	150	149	148	147	146	145	144
75 -	159	158	157	156	155	154	153	152
76 -	167	166	165	164	163	162	161	160
77 -	175	174	173	172	171	170	169	168
78 -	183	182	181	180	179	178	177	176
79 -	191	190	189	188	187	186	185	184
80 -	199	198	197	196	195	194	193	192
81 -	207	206	205	204	203	202	201	200
82 -	215	214	213	212	211	210	209	208
83 -	223	222	221	220	219	218	217	216
84 -	231	230	229	228	227	226	225	224
85 -	239	238	237	236	235	234	233	232
86 -	247	246	245	244	243	242	241	240
87 -	255	254	253	252	251	250	249	248
88 -	263	262	261	260	269	268	267	266
89 -	271	270	269	268	267	266	265	264
90 -	279	278	277	276	275	274	273	272
91 -	287	286	285	284	283	282	281	280
92 -	295	294	293	292	291	290	289	288
93 -	303	302	301	300	299	298	297	296
94 -	311	310	309	308	307	306	305	304
95 -	319	318	317	316	315	314	313	312
96 -	327	326	325	324	323	322	321	320
97 -	335	334	333	332	331	330	329	328
98 -	343	342	341	340	339	338	337	336
99 -	351	350	349	348	347	346	345	344
100 -	359	358	357	356	355	354	353	352

SS/DD or DS/SD
Sectors 360 - 719

Hex Value	>80	>40	>20	>10	>08	>04	>02	>01
Bit Number	7	6	5	4	3	2	1	0
Byte No.								
101 -	367	366	365	364	363	362	361	360
102 -	375	374	373	372	371	370	369	368
103 -	383	382	381	380	379	378	377	376
104 -	391	390	389	388	387	386	385	384
105 -	399	398	393	396	395	394	393	392
106 -	407	406	405	404	403	402	401	400
107 -	415	414	413	412	411	410	409	408
108 -	423	422	421	420	419	418	417	416
109 -	431	430	429	428	427	426	425	424
110 -	439	438	437	436	435	434	433	432
111 -	447	446	445	444	443	442	441	440
112 -	455	454	453	452	451	450	449	448
113 -	463	462	461	460	459	458	457	456
114 -	471	470	469	468	467	466	465	464
115 -	479	478	477	476	475	474	473	472
116 -	487	486	485	484	483	482	481	480
117 -	495	494	493	492	491	490	489	488
118 -	503	502	501	500	499	498	497	496
119 -	511	510	509	508	507	506	505	504
120 -	519	518	517	516	515	514	513	512
121 -	527	526	525	524	523	522	521	520
122 -	535	534	533	532	531	530	529	528
123 -	543	542	541	540	539	538	537	536
124 -	551	550	549	548	547	546	545	544
125 -	559	558	557	556	555	554	553	552
126 -	567	566	565	564	563	562	561	560
127 -	575	574	573	572	571	570	569	568
128 -	583	582	581	580	579	578	577	576
129 -	591	590	589	588	587	586	585	584
130 -	599	598	597	596	595	594	593	592
131 -	607	606	605	604	603	602	601	600
132 -	615	614	613	612	611	610	609	608
133 -	623	622	621	620	619	618	617	616
134 -	631	630	629	628	627	626	625	624
135 -	639	638	637	636	635	634	633	632
136 -	647	646	645	644	643	642	641	640
137 -	655	654	653	652	651	650	649	648
138 -	663	662	661	660	659	658	657	656
139 -	671	670	669	668	667	666	665	664
140 -	679	678	677	676	675	674	673	672
141 -	687	686	685	684	683	682	681	680
142 -	695	694	693	692	691	690	689	688
143 -	703	702	701	700	699	698	697	696
144 -	711	710	709	708	707	706	705	704
145 -	719	718	717	716	715	714	713	712

DS/DD
Sectors 720 - 1440

Hex Value	>80	>40	>20	>10	>08	>04	>02	>01
Bit Number	7	6	5	4	3	2	1	0
Byte No.								
146 -	727	726	725	724	723	722	721	720
147 -	735	734	733	732	731	730	729	728
148 -	743	742	741	740	739	738	737	736
149 -	751	750	749	748	747	746	745	744
150 -	759	758	757	756	755	754	753	752
151 -	767	766	765	764	763	762	761	760
152 -	775	774	773	772	771	770	769	768
153 -	783	782	781	780	779	778	777	776
154 -	791	790	789	788	787	786	785	784
155 -	799	798	797	796	795	794	793	792
156 -	807	806	805	804	803	802	801	800
157 -	815	814	813	812	811	810	809	808
158 -	823	822	821	820	819	818	817	816
159 -	831	830	829	828	827	826	825	824
160 -	839	838	837	836	835	834	833	832
161 -	847	846	845	844	843	842	841	840
162 -	855	854	853	852	851	850	849	848
163 -	863	862	861	860	859	858	857	856
164 -	871	870	869	868	867	866	865	864
165 -	879	878	877	876	875	874	873	872
166 -.	887	886	885	884	883	882	881	880
167 -	895	894	893	892	891	890	889	888
168 -	903	902	901	900	899	898	897	896
169 -	911	910	909	908	907	906	905	904
170 -	919	918	917	916	915	914	913	912
171 -	927	926	925	924	923	922	921	920
172 -	935	934	933	932	931	930	929	928
173 -	943	942	941	940	939	938	937	936
174 -	951	950	949	948	947	946	945	944
175 -	959	958	957	956	955	954	953	952
176 -	967	966	965	964	963	962	961	960
177 -	975	974	973	972	971	970	969	968
178 -	983	982	981	980	979	978	977	976
179 -	991	990	989	988	987	986	985	984
180 -	999	998	997	996	995	994	993	992
181 -	1007	1006	1005	1004	1003	1002	1001	1000
182 -	1015	1014	1013	1012	1011	1010	1009	1008
183 -	1023	1022	1021	1020	1019	1018	1017	1016
184 -	1031	1030	1029	1028	1027	1026	1025	1024
185 -	1039	1038	1037	1036	1035	1034	1033	1032
186 -	1047	1046	1045	1044	1043	1042	1041	1040
187 -	1055	1054	1053	1052	1051	1050	1049	1048
188 -	1063	1062	1061	1060	1059	1058	1057	1056
189 -	1071	1070	1069	1068	1067	1066	1065	1064
190 -	1079	1078	1077	1076	1075	1074	1073	1072

Hex Value	>80	>40	>20	>10	>08	>04	>02	>01
Bit Number	7	6	5	4	3	2	1	0
Byte No.								
191 -	1087	1086	1085	1084	1083	1082	1081	1080
192 -	1095	1094	1093	1092	1091	1090	1089	1088
193 -	1103	1102	1101	1100	1099	1098	1097	1096
194 -	1111	1110	1109	1108	1107	1106	1105	1104
195 -	1119	1118	1113	1116	1115	1114	1113	1112
196 -	1127	1126	1125	1124	1123	1122	1121	1120
197 -	1135	1134	1133	1132	1131	1130	1129	1128
198 -	1143	1142	1141	1140	1139	1138	1137	1136
199 -	1151	1150	1149	1148	1147	1146	1145	1144
200 -	1159	1158	1157	1156	1155	1154	1153	1152
201 -	1167	1166	1165	1164	1163	1162	1161	1160
202 -	1175	1174	1173	1172	1171	1170	1169	1168
203 -	1183	1182	1181	1180	1179	1178	1177	1176
204 -	1191	1190	1189	1188	1187	1186	1185	1184
205 -	1199	1198	1197	1196	1195	1194	1193	1192
206 -	1207	1206	1205	1204	1203	1202	1201	1200
207 -	1215	1214	1213	1212	1211	1210	1209	1208
208 -	1223	1222	1221	1220	1219	1218	1217	1216
209 -	1231	1230	1229	1228	1227	1226	1225	1224
210 -	1239	1238	1237	1236	1235	1234	1233	1232
211 -	1247	1246	1245	1244	1243	1242	1241	1240
212 -	1255	1254	1253	1252	1251	1250	1249	1248
213 -	1263	1262	1261	1260	1259	1258	1257	1256
214 -	1271	1270	1269	1268	1267	1266	1265	1264
215 -	1279	1278	1277	1276	1275	1274	1273	1272
216 -	1287	1286	1285	1284	1283	1282	1281	1280
217 -	1295	1294	1293	1292	1291	1290	1289	1288
218 -	1303	1302	1301	1300	1299	1298	1297	1296
219 -	1311	1310	1309	1308	1307	1306	1305	1304
220 -	1319	1318	1317	1316	1315	1314	1313	1312
221 -	1327	1326	1325	1324	1323	1322	1321	1320
222 -	1335	1334	1333	1332	1331	1330	1329	1328
223 -	1343	1342	1341	1340	1339	1338	1337	1336
224 -	1351	1350	1349	1348	1347	1346	1345	1344
225 -	1359	1358	1357	1356	1355	1354	1353	1352
226 -	1367	1366	1365	1364	1363	1362	1361	1360
227 -	1375	1374	1373	1372	1371	1370	1369	1368
228 -	1383	1382	1381	1380	1379	1378	1377	1376
229 -	1391	1390	1389	1388	1387	1386	1385	1384
230 -	1399	1398	1397	1396	1395	1394	1393	1392
231 -	1407	1406	1405	1404	1403	1402	1401	1400
232 -	1415	1414	1413	1412	1411	1410	1409	1408
233 -	1423	1422	1421	1420	1419	1418	1417	1416
234 -	1431	1430	1429	1428	1427	1426	1425	1424
235 -	1439	1438	1437	1436	1435	1434	1433	1432

SECTOR 1 - FILE DESCRIPTOR INDEX RECORD

Byte Nos.	Contents
0 - 1	Pointer to Alphabetically 1st File Descriptor Record Sector Number
2 - 3	Pointer to Alphabetically 2nd File Descriptor Record Sector Number
4 - 5	Pointer to Alphabetically 3rd File Descriptor Record Sector Number
5 - 6	Pointer to Alphabetically 4th File Descriptor Record Sector Number
.. - ..	Pointer to Alphabetically ... File Descriptor Record Sector Number

The two byte values in this table point to the sector number that contains the File Descriptor Record or File Header (File Name). These pointers are placed in this table according to the alphabetical order of the File Names. (see Sector 2 - xx description)

There is enough room in this record for 127 File Header Pointers. This is what limits the disk capacity to 127 file names. After the format routine has formatted the tracks it writes the VIB to sector 0 and then writes this sector with all 0's in it.

As files are added to the disk the pointers in this table are sorted and the sector is rewritten. When a file is deleted from the disk its pointer is removed from this table, the table is adjusted and rewritten, and the sectors that the file resided in are released from the Allocation Bit Map in sector 0. The File Header and the file's data remain intact until another file or additional data is written to the disk in the sectors that the deleted file resides in.

If you change the order of the pointers so they are out of alphabetical order some of the File Headers might not be found during the binary search for the file name (FILE NOT FOUND ERROR). If the list is out of order the File Names will also be cataloged out of alphabetical order. If you duplicate some of the pointers then the File names will also appear more than once in a catalog listing. If a disk is copied with the disk manager file by file, not sector by sector, then, since the files are copied alphabetically, the pointers will be in numerical order.

ie: >0002 >0003 >0004 >0005 >0006 >0007 >0008 >0009 >000A >000B >000C >000D
    >000E >000F >0010 >0011 etc.

Also if the first 2 bytes equal >0000 but the following bytes contain pointers to the File Headers then the file names will NOT show up when the disk is cataloged BUT the files can still be located with OLD, OPEN, LOAD AND RUN, RUN PROGRAM etc.

The easiest way to Restore a deleted file is to insert the File Header's sector pointer back into this table where it alphabetically belongs and then, using the disk manager, copy the file to another disk and then copy it back onto this disk. This will also properly restore the bits in the Allocation Bit Map to remark the sectors as used.

NOTE: This only works IF YOU HAVE NOT WRITTEN ANYTHING TO THE DISK SINCE THE FILE WAS DELETED!!

IF THIS SECTOR OR SECTOR 0 ARE MAPPED BAD DURING A FORMAT OPERATION THEN THE WHOLE DISK IS CONSIDERED TO BE BAD. THESE POINTERS MUST RESIDE IN THIS SECTOR AND THE BIT MAP MUST BE IN SECTOR 0 FOR THE DSR ROUTINE.

APPENDIX - F SECTORS 2 - xx - FILE DESCRIPTOR RECORDS - FDR

Byte Nos.	Contents
0 - 9	**10 Character (max) File Name** Unused characters = space character
10 - 11	Reserved for future expansion not used - all 0's >00 >00
12	**File Status Flags** Bit No. / On = 1 / Off = 0 >00 = Dis/Fix — 0 — Program file — Data file >01 = Program — 1 — Internal (Binary) — Display (ASCII) >02 = Int/Fix — 2 — Reserved — Reserved >80 = Dis/Var — 3 — Write Protected — Not Write Protected >82 = Int/Var — 4-6 — Reserved — Reserved If Write Prot. add >08 — 7 — Variable length — Fixed length
13	**Maximum Number of Records per Sector or AU** This byte equals the sector size (256 bytes) divided by the record length (>100/>50 = >03 or 256/80 = 3 . Program files = >00 ie: Dis/Fix 40 = >06 Dis/Fix 60 = >04 Dis/Fix 80 = >03 etc.
14 - 15	**Total Number of Sectors Used** This is the same as the directory or cataloged size minus 1. The value in this sector does not include this sector (the FDR).
16	**End of File Offset** For Variable length files and Program files this byte indicates how many bytes of the last sector of the file are used. This byte also indicates what byte number the EOF marker is at. For Fixed length files this byte = >00.
17	**Logical Record Length** ie: Fixed or Variable 40 = >28 80 = >50 163 = >A3 254 = >FE
18 - 19	**Number of Fixed Length Records or Number of Sectors Used by Variable Length Records.** Note: The bytes are reversed so >0500 = >0005 (not used by Program files)
20 -27	Reserved for future expansion - not used all 0's >0000 >0000 >0000 >0000
28 - 255	**Data Chain Pointer Blocks** 3 byte groups that point to the clusters of sectors used by a file and also indicate the highest logical offset (total number of sectors used so far). Note: The second byte in each cluster is split up into two nibbles and each nibble is joined to the other two bytes to form two three nibble pointers like this: | ss2 : ss1 | | of1 : ss3 | | of3 : of2 | gets changed to: | ss3 : ss2 : ss1 | | of3 : of2 : of1 | ie: Disk Reads — Convert to @ byte 28-30 - >2E 10 00 — >02E 001 @ byte 31-33 - >91 50 00 — >091 005 the cluster starts at sector number — with total offset of — so Find File will display Start End. Offset >02E (46) — >001 — 46 47 1 >091 (145) — >005 — 145 148 5 Note: If no clusters are indicated then this file is just a header.

MILLERS GRAPHICS - LIMITED WARRANTY

Millers Graphics warrants the Advanced Diagnostics program which it manufactures to be free from defects in materials and workmanship for a period of 90 days from the date of purchase.

During the 90 day warranty period Millers Graphics will replace any defective products at no additional charge, provided the product is returned, shipping pre-paid to Millers Graphics. The Purchaser is responsible for insuring any product so returned and assumes the risk of loss during shipping.

Ship to:

Millers Graphics
1475 W. Cypress Ave.
San Dimas, California 91773

WARRANTY COVERAGE

This ADVANCED DIAGNOSTICS program is warranted against defective material and workmanship. THIS WARRANTY IS VOID IF THE PRODUCT HAS BEEN DAMAGED BY ACCIDENT, UNREASONABLE USE, NEGLECT, TAMPERING, IMPROPER SERVICE OR OTHER CAUSES NOT ARISING OUT OF DEFECTS IN MATERIALS OR WORKMANSHIP.

WARRANTY DISCLAIMERS

ANY IMPLIED WARRANTIES ARISING OUT OF THIS SALE, INCLUDING, BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN DURATION TO THE ABOVE 90 DAY PERIOD. MILLERS GRAPHICS. SHALL NOT BE LIABLE FOR LOSS OR USE OF THE SOFTWARE OR OTHER INCIDENTAL OR CONSEQUENTIAL COSTS, EXPENSES, OR DAMAGES INCURRED BY THE CONSUMER OR ANY OTHER USE.

Some states do not allow the exclusion or limitation of implied warranties or consequential damages, so the above limitations or exclusion may not apply to you in those states.

LEGAL REMEDIES

This warranty gives you specific legal rights, and you may also have other rights that vary from state to state.

REPLACEMENT AFTER WARRANTY

After the 90 Warranty period has expired you may return any original defective diskette, along with a check for 4.00 to cover shipping and diskette costs, and we will replace it.

MILLERS GRAPHiCS
1475 W. Cypress Ave.
San Dimas, CA 91773