9900
DISK CONTROLLER
REFERENCE MANUAL

I M P O R T A N T A D D E N D U M

Please Read

Only factory tested double density drives with step rates of 20 milliseconds or less are recommended for double density operation. While some single density drives may appear to function properly, unreliable operation and lost data may result from the use of these drives.

The COPY DISK function under Disk Utilities in the CorComp Disk Manager Program does a sector by sector copy. It will write over any information on the TO diskette. It is mainly intended for making fast back up copies of your diskettes. The FILE COPY function under File Utilities should be used to copy multiple diskettes onto one diskette.

The Disk Manager Diskette contains the files named FORTH and FORTHSAVE. These files should be copied to your TI-Forth system diskette with the FILE COPY function under File Utilities. This will allow you to Load and Run TI-Forth with the Load and Run (Assembly File) function under File Utilities. Your TI-Forth diskette will still Load and Run through the Editor/Assemble. Module and it will now also Load and Run through the Mini Memory Module provided you have memory expansion.

Some modules contain their own power up routines such as the Terminal Emulator II and some of the Education modules. If a module or a Basic program that accesses a module does not function properly when selected from the 9900 DISK CONTROLLER title screen then you must select them through the second menu screen. When the 9900 DISK CONTROLLER title screen appears just press the space bar twice and then make your module or TI Basic selection and they will work fine.

The EARLY version of the CorComp RS232 Card may not be compatible with the CorComp Disk Controller Card. If you can not get the 9900 DISK CONTROLLER title screen to appear on power up and you have an early version of the CorComp RS232 then remove the RS232 card and try again. If the title screen appears then try inserting the RS232 right next to the Disk Controller card in the Peripheral Box and try it again. If the title screen does not appear then please call CorComp at 714-630-2903 for further information.

---------- NOTICE ----------

The information contained in this manual is subject to change without notice.

CorComp Inc. 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 material.

This document contains proprietary information protected by copyright. All rights are reserved. No part of this document may be photocopied or reproduced in any form without prior written consent from CorComp Inc.

Copyright (C)1984
by
CorComp Inc.

Reference Manual

TABLE OF CONTENTS

9900 DISK CONTROLLER CARD

INTRODUCTION

The 9900 Disk Controller Interface is a powerful package of computer software and hardware that allows you to store and retrieve information quickly from 5 1/4 inch diskettes (sold separately). The system is made up of the following pieces:

- **9900 DISK DRIVE CONTROLLER** - The 9900 Disk Drive Controller, which can control up to four disk drives, controls the drives and tells them where and when to read and write the information properly.

- **DISK MANAGER DISKETTE** - The Disk Manager Diskette packed with the 9900 Disk Controller Interface helps you control the information on your diskettes. Initializing, naming, or renaming diskettes, renaming files, deleting files, copying files, and copying diskettes are all easily done using the Disk Manager.

- **9900 DISK DRIVE*** - The 9900 Disk Drive writes information on and reads information from diskettes. Any program or data file is quickly located as commanded by the 9900 Disk Drive Controller. The disk drive spins the diskette like your record player spins a record and directs the flow of the information to and from the diskette.

The 9900 Disk Controller Interface represents the cutting edge of a major advance over other systems. Because the control software needed is on the Disk Manager Diskette and in the Disk Controller, you DO NOT NEED a Command Module in order to perform disk memory functions. First read this manual in order to get the most from your 9900 Disk Controller Interface. Please review the appendices for any special information.

* Sold Separately

SET-UP INSTRUCTIONS

Once you've unpacked the unit, you're ready to start (save the packing material for storing or transporting the unit at a later time). The steps required to check the operation of the disk controller card, Disk Manager Diskette and optional internal and external disk drives are discussed in this section. Please read it carefully.

CAUTION

Static electricity discharges can damage electronic parts. To avoid this damage do not touch any connector contacts.

Configuring Your Disk Drives

WARNING: DISCONNECT ALL POWER CORDS FROM THE POWER OUTLETS DURING THE FOLLOWING PROCEDURE!

* **Single Disk Drive System** - A new disk drive is configured at the factory as "drive one" (1) for a single disk system. Skip over the balance of this section and proceed with "Disk Drive Head Step Settings".

* **Multiple Disk Drive System** - In a multiple drive system each disk drive must know which number it is and which drive is the last drive in the system.

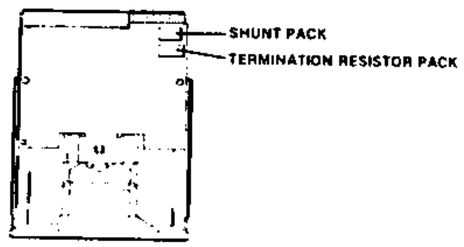
1. If you have external drives with their own power supply and cases then remove the covers using a phillips or blade screwdriver as required.

2. Decide which drive will be number one, two, three and/or four. **Note:** If you have a drive that will be installed into your TI Peripheral Box, it should be configured as drive number one.

3. Locate the main printed circuit board. On it, locate the termination resister pack, and the DIP shunt.

4. Remove the termination resister pack from all but the last drive by inserting a small blade screwdriver and gently lift the resister pack straight up. Save the resister pack as you can use it if the drive will be used later as the only drive or if it is the last disk drive.

5. Configure the DIP shunt for each drive as shown below. Note: the DIP shunt can be replaced by a plug-in DIP switch that can be purchased at most electronic stores.



6. Replace the external drive cover(s), tighten and align the screws properly, DON'T RUN ANY UNIT WITH THE COVER OFF.

- Disk Drive Head Step Settings -

The Disk Controller Interface is configured at the factory with all DIP switches set to 10 milliseconds. Try these settings with your Disk Drives and Initialize a new diskette.

When the drive heads steps to read and/or write a file, you should hear a clean sounding click as it moves in and out. Most of the Drives tested work well at 10ms - some may need a setting of 3ms or 6ms. Very few Drives require the 15ms setting. If they do not work properly, see the Head Step Setting Guide below.

If your drives are not treading and writing properly or if they are making chattering sounds - then remove the 9900 Disk Controller Interface and follow the items below and the diagrams on the next page to adjust your head step times.

If the Drive reads and writes OK but there is a little chattering when the head steps in and out - then set the head step time to the next lower value setting, faster step, and retest.

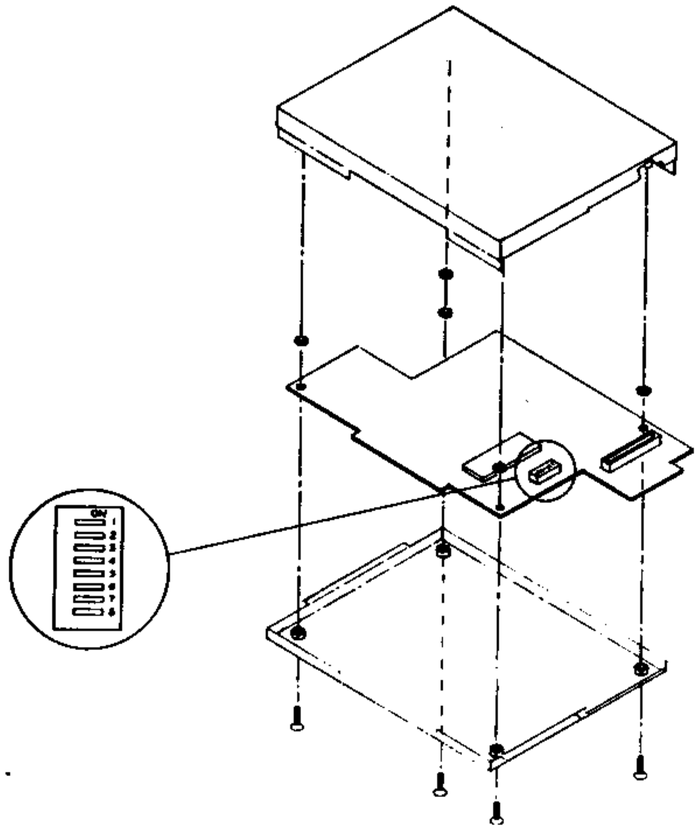
If the Drive will not read or write at all and there is a lot of chattering when the head TRIES to step - then set the head step time to a higher value setting, slower step, and retest.

If the Drive sounds OK, no chattering, but there are a lot of errors when it reads and writes - then set the head step time to the next higher value setting, slower step, and retest.

ACTION	CAUSE	ADJUSTMENT
Drive will not read or write at all. Drive chatters. Head will not step.	Head step time is set too fast.	Set the head step time to a slower setting and retest.
Drive reads and writes, but with a lot of errors. Drive sounds OK.	Head step time is set too fast.	Set the head step time to a slower setting and retest.
Drive read and writes OK, no errors. Drive sounds OK.	Head step time is set OK	----------
Drive reads and writes OK, no errors. Drive chatters a little when the head steps.	Head step time is set too slow.	Set the head step time to a faster setting and retest.

Head Step Time DIP Switch Settings

DRIVE	DIP SWITCH	3 ms	6 ms	10 ms	15 ms
DSK1	1	ON	OFF	ON	OFF
	2	ON	ON	OFF	OFF
DSK2	3	ON	OFF	ON	OFF
	4	ON	ON	OFF	OFF
DSK3	5	ON	OFF	ON	OFF
	6	ON	ON	OFF	OFF
DSK4	7	ON	OFF	ON	OFF
	8	ON	ON	OFF	OFF



INSTALLING THE 9900 DISK CONTROLLER INTERFACE

* If you have an internal drive, see "Connecting the Controller to an Internal Disk Drive."

* If you have one or more external drives, then follow the directions in "Connecting the Controller Card to External Disk Drives."

The TI Peripheral Expansion System has eight slots into which accessory cards can be inserted. The Interface Cable card must be plugged into slot one (1). (Refer to the owner's manual for set-up information) If you have an internal disk drive the 9900 Disk Controller Card must be plugged into slot number eight (8). Other cards may be plugged into slots two through seven (2-7).

1. **WARNING: TO AVOID DAMAGING ANY ACCESORY CARDS, WAIT TWO(2) MINUTES AFTER TURNING OFF THE SYSTEM FOR THE POWER TO DISCHARGE BEFORE PROCEEDING.**

2. Remove the top of the TI Box, by pressing the springs in and lifting the back edge of the top up.

3. **Plug the 9900 Disk Controller Card into slot 8.** Make sure the indicator light on the front of the card is pointing to the front of the TI Box.

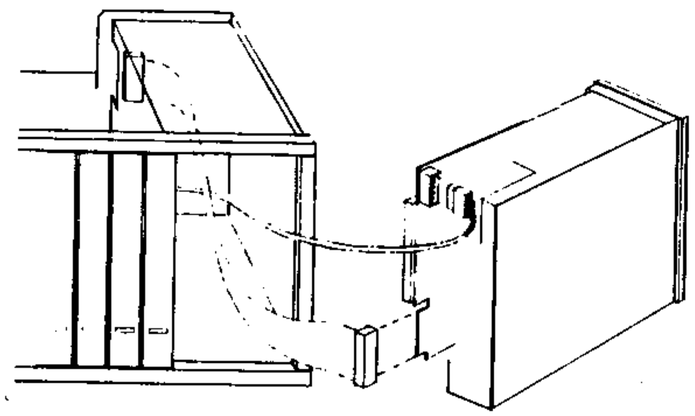
4. Now go to the appropriate drive installation section.

Connecting the Controller to an Internal Disk Drive

1. The optional cable is designed to connect one internal disk drive to the Disk Controller. Connect the small connector end of the cable to the Disk Controller card and the large end to the disk drive as shown below.

2. One four wire power cord is inside at the back of the disk drive opening on the front panel. Connect the loose end of the cord into the four-pin connector at the back of the disk drive.

   Note: If the activity light of the disk drive stays on after power is turned on, then power OFF the system and reverse the connector on the disk drive.



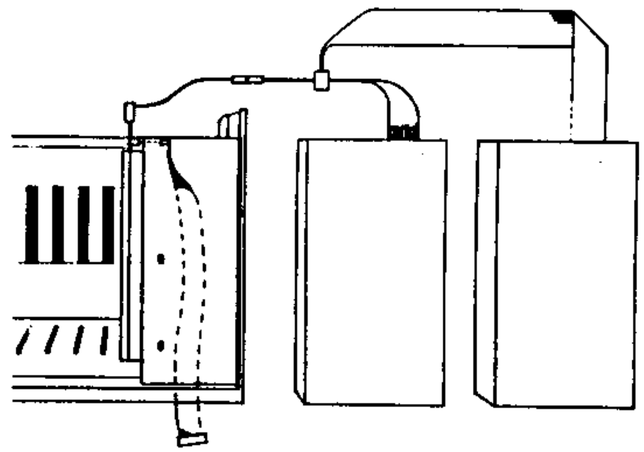
5. Connect the disk drive with the screws through the bottom and top plate of the TI Box.

6. Replace the top on the TI Box being careful to align it properly.

Note: The internal drive in the TI Box is considered as DSK1. The external drive closest to the TI Box is DSK2. The second external drive is DSK3 and the forth is DSK4.

Connecting the Controller to External Disk Drives.

1. Using the picture below as a guide, note that each external drive has an attached cable with a connector on the end of the cable which is outside of the box. Use an adapter to connect the drives together (if you are using two or more external drives) or you can use a special optional cable as shown below. Next connect the end of the adapter or special cable to the edge of the Disk Controller Card on the back of the TI Box.



CAUTION

Before moving the computer console, make sure that the TI Box is disconnected To prevent any damage occuring to your computer equipment make sure everything connected to your Home Computer is disconnected before moving anything. If you are moving a long distance make sure that you remove any cards and repack the cards and the TI Box in their packing material in which they were originally recieved.

Starting Your System

Make sure that all power to any of the devices, external drives and such, are powered off. Then ensure that all the devices are plugged into their electrical wall plugs. At this time, turn the power switches on in the following sequence:

NOTICE

Before operating your computer system always perform this power-up sequence:

1. External Disk Drives
2. TI Peripheral Box
3. Monitor/TV
4. Computer Console

Since the computer must manage the memory expansion, and the other cards in the TI Box, make sure the power is ON before the computer is turned on. If this is not done the system will not work properly. To begin again, turn the console power OFF then ON.

How to Test Your System

Make sure there are NO diskettes in any of the drives. Power up your system using the Power-up sequence on the previous page.

Select TI BASIC from the Master Screen, then enter the following program:

```
10 OPEN #1:"DSK1.TEST"
20 CLOSE #1
```

Type in "RUN" and press the "ENTER" key. Watch disk drive one (DSK1). The red light on the front of the disk drive should come on briefly, and then go out. This indicates the drive is properly connected. If the light comes on and stays on or if it does not come on at all, then recheck that all devices have power and are connected correctly.

THE CONTROLLER LIGHT COMES ON WHEN THE CONTROLLER IS TALKING TO A DISK DRIVE.

Note: Even though the drive may be working properly, since there is no diskette in the drive the following message will appear on the screen:

I/O ERROR 06 IN 10

Now, repeat the above test for disk drives 2,3 and 4 (if you have them). Be sure to change the program to say DSK2, DSK3 and DSK4 in line number 10, as you test each drive that you have.

Once you are sure the drives are working correctly, type "BYE" and press the "ENTER" key. Now, you can start using the disk portion of the system as described later in this manual.

If you have any disk system operating problems, please review the "Set-Up Instructions" section of this manual.

FEATURES OF YOUR SYSTEM

The main use of your 9900 Disk Controller Interface is saving and reading information via BASIC, Extended BASIC, TI LOGO, TI FORTH, or TI Assembly Langauge programs. Using the proper BASIC commands, you can do the following functions:

* **SAVE PROGRAMS** - Each single sided double density diskette holds over 180,000 characters of information, while a double sided double density diskette holds over 360,000 characters of information. As an example, you can hold 127 BASIC programs each about 200 lines long on a double-sided double-density diskette.

* **SAVE AND READ INFORMATION** - You can save information to various files that are created when your program is running.

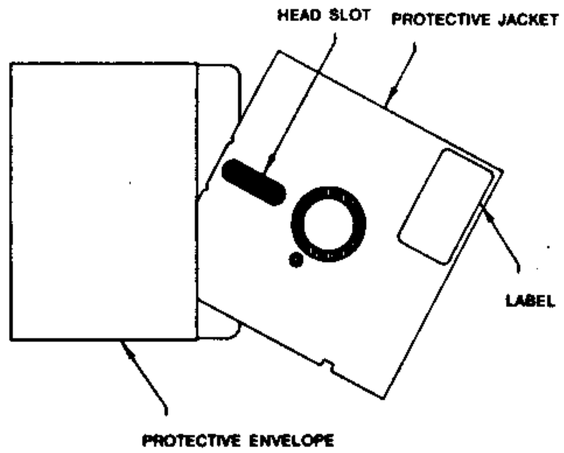
* **UPDATE DATA** - Through a BASIC program you can change and update the information in your files.

DISKETTE INFORMATION

What Are Diskettes?

Your system uses diskettes to store information. These diskettes are a round piece of flexible plastic which is coated with a magnetic film. The magnetic film is the surface upon which the information is written and read.

Note: The diskette is encased in a protective jacket which is inserted in a storage envelope (5 1/2" X 5 1/2"). Keep the diskette in the storage envelope when it is not in a disk drive.



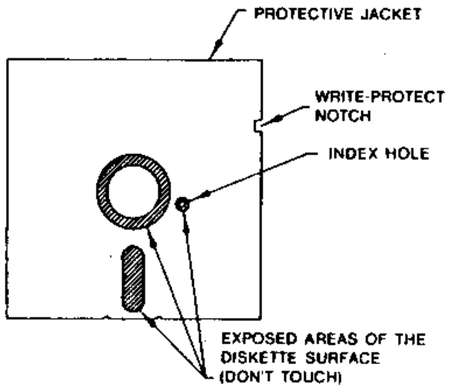
Diskettes Are...Magnetic

Just as your phonograph "spins" a phonograph record, the disk drive "spins" the diskette. The record and the diskette have information stored on them. The information on the record may be voice or music and is permanent and cannot be changed.

The diskette stores programs and files of letters or other information. Your computer can write information on the diskette and read the information. New information may also be written on the diskette. The diskette is round like your 45 RPM record, and has a magnetic coating on the round plastic material. The information is written and read just like your cassette player does with a cassette.

Diskettes Are...Flexible

A close look at the parts of a diskette is shown below:



A diskette is made of a piece of round flexible material, which is coated with a magnetic material. Just like cassettes and other magnetic material, information may be recorded on the diskette magnetic material and read back. Single sided diskettes have information on one side. Double sided diskettes have information upon both sides. Diskettes are protected by a black protective jacket. Since the diskette material is very sensitive to finger smuges, hair, smoke,etc. the jacket protects most of the diskette surface. One of the exposed areas is called the head slot. The disk drive has a read/write head which contacts the diskette surface through the head slot.

Diskettes Are...Not Forgetful

Single-Sided Single-Density diskettes which can be used with the 9900 Disk System are divided into 40 circles. The circles are called tracks. Tracks are numbered from 0 to 39, starting at the outside track and moving inward. Each track is divided into 9 sectors. In this case there are 9 sectors x 40 tracks = 360 sectors. The sectors are numbered from 0 to 359 starting with sector 0 of the first track and going to sector 359 of the last track. Each sector holds 256 characters x 360 sectors = 90K or 92,160 characters.

Single-Sided Double-density diskettes are also divided into 40 tracks, however each track is divided into 18 sectors. 18 sectors X 40 tracks = 720 sectors. Each sector holds 256 characters x 720 sectors = 180K or 184,320 characters.

Double-Sided Single-Density diskettes are divided into 40 tracks on each side of the diskette. The tracks are numbered from 0 to 39, starting at the outside track and moving inward. On the other side tracks are numbered from 40 to 79 starting from the inside track and moving outward. Each track is divided into 9 sectors. In this case there are 9 sectors x 80 tracks = 720 sectors. Each sector holds 256 characters x 720 sectors = 180K or 184,320 characters.

Double-Sided Double-Density diskettes are also divided into 40 tracks on each side and they are numbered the same as a Double-Sided Single-Density diskette. However, each track is divided into 18 sectors. In this case there are 18 sectors x 80 tracks = 1440 sectors. Each sector holds 256 characters X 1440 sectors = 360K or 368,640 characters.

NOTE: Diskette sectors 0 and 1 are automatically used during diskette initialization to store informaton about the rest of the diskette. The disk controller uses sectors 0 and 1 of the first track to make an index of file locations, if these sectors become damaged, then the 9900 Disk Controller will be unable to locate any files on the diskette.

Write-Protected Diskettes

Your diskettes have a notch in the edge of their protective container. As long as the notch is left uncovered, then the system can write on the diskette. Covering the notch with a piece of non-transparent tape prevents writing on the diskette. When you have a diskette which contains information that you want to protect, then cover the notch with a piece of tape. If you wish to later write on the diskette, simply remove the tape.

Labeling and Handling Precautions

Be very careful not to touch the diskette itself. Handle it carefully by the edges of its protective jacket.

Do not write on the protective jacket with a ballpoint pen. Use a felt marker pen or write what you wish on a label then put it on the protective jacket.

Diskette information can be affected by any magnet or magnetic field. Do not put a diskette on top of something which generates a magnetic field, such as a speaker, monitor or a television, etc. Remember that childrens toys, magnetic door latches and magnetic note holders will also affect your diskette.

CAUTION

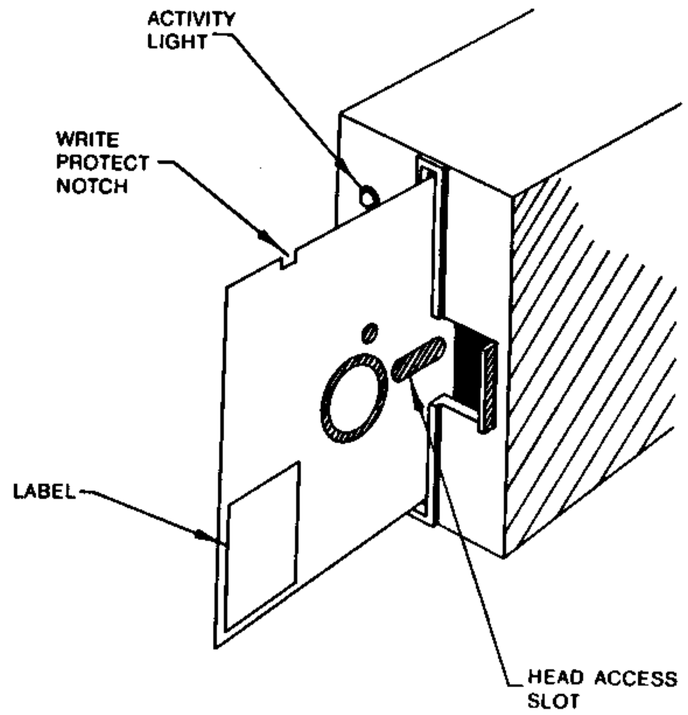
Make sure that the controller and disk drive are connected and the power is turned on. Also, make sure that you do not insert or remove a diskette when the drive light is on When installing or removing a cable never leave a diskette in a drive. Always store diskettes in their protective jackets.

In order to insert a diskette in a drive, open the disk drive door on the front of the drive. Carefully holding the diskette by the corner nearest the write protect notch, insert the diskette with the notch nearest the end of the drive which has the light.

When you wish to remove a diskette, open the disk drive door, after the light has gone out and take out the diskette. Store the diskette in its storage envelope.

Loading Diskettes

The proper way to insert (load) a floppy diskette into the disk drive is...gently. The label should be near the outside edge (where you hand holds the diskette). The "slot" should point straight in, and the diskette should slip easily but firmly into the correct place.



Note: Always turn the power ON to a disk drive before inserting a diskette. Never turn the power OFF until the diskette has been completely removed.

INITIALIZING YOUR DISKETTES

1. Power ON your system.

2. Insert the Disk Manager Diskette in any disk drive.

3. Select Disk Manager by pressing the 1 key - The Disk Manager will load and run.

4. Select Disk Utilities, from the Disk Manager screen, by pressing the 2 key.

5. Insert a blank diskette into a disk drive.

6. From the Disk Utilities screen, select Initialization by pressing the 4 key (this is shown on the following page).

Note: Diskettes can be named according to your choosing. A diskette name can be up to 10 characters long and may use any character except the period or space character.

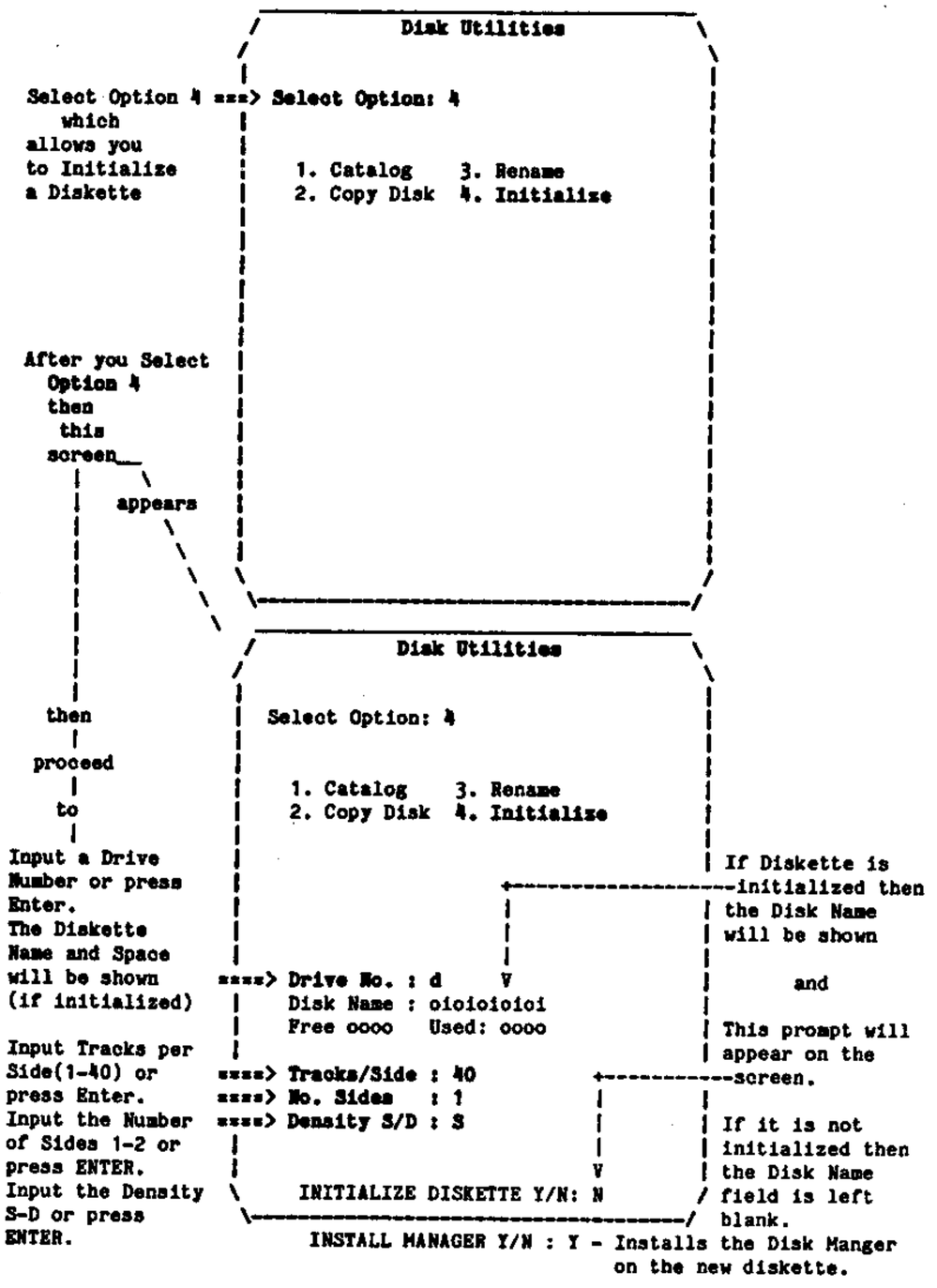
Correct names for diskettes are:

DISK10	YOURDISK	YOUR_DISK	DET_DATA
DRK_DATA	JEREMY	BCK_UP	FUD3/15/84

The following are not valid disk names:

5.268	YOUR DISK	THISNAMEISTOOLONG

Initialize Screens



Transfering Your Programs from Cassette to Diskette

1. Load the program you wish to transfer, from cassette, into computer memory with the OLD CS1 command.

2. Follow the cassette loading instructions which appear on the screen.

3. Afetr the program has been loaded, and you have the DATA OK message on the screen, save it on your diskette by typing:

SAVE DSK1.program-name and press the **ENTER** key.

4. The program name can be up to 10 characters in length, and it cannot contain the space or period characters. Also, if you SAVE two different programs under the same name on the same diskette then only the last one SAVED will be on the diskette.

5. To load your program back into the computer from diskette, type:

OLD DSK1.program-name and press the **ENTER** key

6. Continue steps 1 through 4, above, until all of your cassette programs have been transfered to diskette.

This next sequence transfers a program named "LETTER1" from the diskette in disk drive two (2) to a cassette.

TYPE: **OLD DSK2.LETTER1** and press the **ENTER** key.

(After the cursor re-appears.... then)

TYPE: **SAVE CS1** and press the **ENTER** key.

(follow the cassette instructions which appear on the screen)
The program "LETTER1" is now on the cassette.

Transfer Data from Cassette to Diskette

In order to read data from a cassette and store it on a diskette, you must tell the computer to read the data from the cassette into the computer memory and then store the data on the diskette. First determine how you are going to use the **OPEN** statement. This tells the computer how you want the data stored on the diskette.

TYPE:

```
100 OPEN #1:"CS1"
110 OPEN #2:"DSK1.filename"
120 READ #1:Z$
130 REM EOF ERROR WHEN THE CASSETTE IS DONE
140 PRINT#2:Z$
150 GOTO 120
```

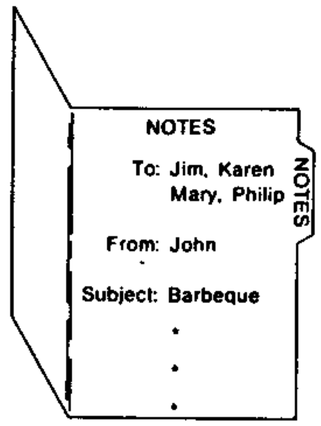
When the program is **RUN**, instructions will appear on the screen which will tell you how to use the cassette player.
Note: The **EOF** function cannot be used with cassettes.

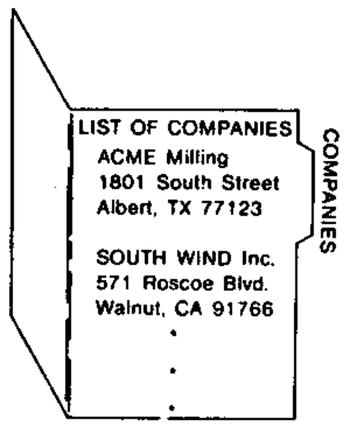
ABOUT FILES

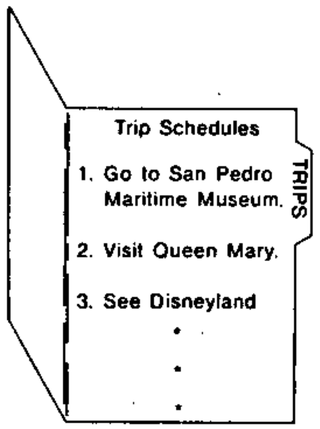
Where is important information kept? Generally, related information on a specific subject is grouped in a file.

A file may hold letters with a business associate or maybe a list of folks with whom you bowl.

As an example, here are three files:







The first file, labeled NOTES holds all the notes you have sent people.

The CUSTOMER file contains an alphabetized list of companies.

The third file TRIPS is full of information on places to vist when in Southern California.

Your computer stores information in files. As in the examples shown above, each file must have a unique name and keep related information together. Besides data files, your diskette files can also contain program information. These programs reveal the power of the 9900 Disk Controller Interface.

As an example, you may have a program in a file named LISTER, that reads the data file COMPANY and makes a third file. This file may be arranged according to geographical groupings, instead of the alphabetic arrangement.

Where Do You Store Files?

Your system has VDP memory, RAM memory and mass-storage memory. Your files are stored on the mass-memory devices, such as floppy disk drives.

Each 5 1/4 inch single-sided double-density diskette can hold about 180,000 characters of information. The double-sided double-density diskette can hold 360,000 characters of information.

How Many Files On A Diskette?

You can sore up to 127 files on a diskette. Sometimes, a smaller amount of files will fill the diskette. The reason for this is that the smaller amount of files, when added together have as many characters as the diskette can hold.

Whats In A File Name?

A file name is made up of two parts:

devicename + filename = a files name

The previous examples of files had names. Remember they had the following names:

NOTES

COMPANIES

TRIPS

These names can and are used as filenames. Try to select a file name that makes sense to you. Filenames that describe what is in the file will help you remember the file contents.

Also remember that a filename can bemade up of up to ten characters as shown below:

* Any letter of the alphabet

* Any number (0-9)

* Any symbol on the keyboard except the period and space

You may mix letters, numbers and symbols when you make up a filename. We recommend that you get in the habit of using filename extensions. Extensions always start with a colon (:) and usually have from 1-3 characters. You can use the filename extension to know what is in a file just by looking at the name. Here are some examples:

INCOME:84

NOTES:84

RECORD:TAX

TELE#:A-M

As you can see these extensions give you more information about the file contents, just by looking at them. The file INCOME:84 would hold income figures for 1984. RECORD:84 would contain receipts and other information for the year 1984.

What Can Be Wrong With Filenames?

Remember that filename cannot contain the space character or the period character. The following filenames are incorrect:

01234567891

RECORD.TAX

NOTES 84

The first name has too many characters. The second uses a bad character to make an extension (the period). The third uses a bad character (the space).

How to Manage Files

After you have written a letter or a program, you will probably want to keep it on file. The paper it is printed on may be put in a folder or envelope and stored on a shelf, but your computer allows you to store information electronically.

A disk drive is a machine similar to your record player. Just as your record player can spin the record and put (access) the record arm on different parts of the record, the disk can access different places on the diskette. Remember, when you play different songs on a record, you are accessing the record in a random way or ramdomly to play the song you wish. The disk can access the diskette to access the letter, program, etc. you wish. A diskette is a flexible piece of plastic that has a coating of magnetic material on each side. It is encased in a protective jacket that has "windows" cut in it so the disk can "read" and "write" the information. The diskette is called a "floppy disk" because of the flexible nature of the diskette material.

Bytes

The storage capacity of disks is measured in bytes. A byte is represented by one typed character.....a letter, number or symbol.

A floppy diskette that can hold 368,640 bytes (characters) is said to hold 360 kilobytes (1K=1024 Bytes). This is abbreviated as 360K. A rule of thumb (approximation) of how much information a floppy diskette can hold is to use 6 characters per word and 250 words per page.

6 characters per word X 250 words per page = 60 pages on a 90K Disk Drive
= 240 pages on a 360K Disk Drive

Remember, that each space, carraige return, line feed and control character code, etc. counts as a character. So, the actual storage capacity is about 50 pages.

A 90K diskette = Single-Sided Single-Density.
A 180K diskette = Single-Sided Double-Density.
or Double-Sided Single-Density.
A 360K diskette = Double-Sided Double-Density.

Electronic Filing

After you have written a letter or a program, you will probably want to store the information on a disk. Your computer will store information electronically The advantage to having your computer perform the filing of your data electronically is the speed and ease of storing and retrieving information.

Naming Your Files

When storing information on a floppy diskette, you are creating or updating a "file". Much like manually filling information, this becomes a record of the pieces or group of data. As with any record the file must be labeled or named.

A file may be given any name you choose, as long as it is no more than ten (10) characters long and it doesn't contain a space or a period character.

For example, a file may be named for the person that wrote or created the file. The file may be named JACK. It is a good habit to also have a back-up copy of your file in case something happens to the original file. Such a file might be named JACK:BU (BU is shorthand for backup).

Planning a weeks meals can easily be recorded on a diskette. The entire file may be named MEALS. If an entire months meals are planned, you have a collection of files. The files for each of the weeks meals could be named - MEALS1, MEALS2, etc.

Directory

A directory is on every diskette. It is exactly like the index section of a book. It contains a list of the files that are on the diskette along with the file size and type of file. Also, the space that is used on the diskette and the amount of free space remaining is shown. The 9900 Disk Controller Interface and the Disk Manager work together to maintain and show you this information. On your system the directory is called the catalog. Looking at the list of files on a diskette is called "Catologing the diskette". After you create or copy, etc. files on a diskette, the information about that file is put into the catalog by the 9900 Disk Controller Interface.

Backup Copies of Files

A lot of things can happen to your electronic files. These are things that can destroy, damage, or erase your files. Remember, the diskette material on which your files are stored is fragile. The wrong keystrokes, power failure, drinks or coffee spilled on a diskette, etc. can cause you to lose all or part of a file.

Developing a habit of "backing up" your files as you work with them cannot be too strongly encouraged. The Copy Disk Utilty of your Disk Manager and the Copy Files option of the File Utilities are the tools to use to make backup copies of your files. When using TI BASIC or other langauges that allow you to LOAD or SAVE to diskette, SAVE a backup copy of your file to another diskette.

Erasing or Deleting Files

"DISK IS FULL! screams the CRT. What to do? You must erase one file in order to save the one you have in the computer or use another diskette.

If your disk holds 90k or 180k of characters and 86k or 176k are in use, then don't start writing a long letter or program. Make sure that there is enough room on the diskette where you intend to put your file or, insert a diskette that has enough room. If your are in a position where there are no diskettes with enough space available, you may have to erase a file in order to make room on the diskette. Look for a file which is a backup file or one you haven't used in a long time.

If you are using TI BASIC or EXTENDED BASIC, remember that you cannot load the Disk Manager to look at the Catalog on the diskette. Remember LOADING the Disk Manager will wipe out your TI BASIC or EXTENDED BASIC program. What to do ? It is a good practice to keep a copy of the BASIC Catalog program, (see the Cataloging Diskette Files section), on diskettes which hold BASIC or EXTENDED BASIC programs. If you keep a copy of this program as a part of the program you are writing, then you can RUN the program and see what files are on the diskette that you can erase. Remember, that if the Catalog program is the last thing in your program then you can type in RUN <line-number> and only the Catalog function will run. In this case you can later use the DELETE statement (see User's Reference Guide) to delete the selected file to make room on the diskette.

Note: Also see the **MERGE** statement in the Extended Basic Users Guide.

OPERATING YOUR DISK SYSTEM USING TI BASIC

It is easy to use TI BASIC when operating your disk system. We will cover Saving/Loading programs and processing data using a program.

Your disk system uses some of the space in the console memory (VDP RAM) of your Home Computer. How much space used depends on how many files are **OPEN** at the same time. For TI BASIC the default number is 3. This is 2088 Bytes of VDP RAM. Remember that the CALL FILES statement can be used to change the number of files that may be OPEN. Also, if the number of files which can be OPEN is increased then the amount of memory used by the disk system is also increased.

Saving and Loading Programs

Your disk system quickly and effectively stores and reads TI **BASIC** programs About, 127 one hundred line programs can be stored on a single-sided, double-density diskette. And, 127 two hundred line programs can be stored on a double-sided diskette.

SAVE command: Used to write (or store) a program on a diskette.

TYPE: **SAVE DSK1-4.program-name** and press the **ENTER** key.

Valid program names may be up to ten characters long, and include any character except the period and space characters.

OLD command: Used to read a program from diskette

TYPE: **OLD DSK1-4.program-name** and press the **ENTER** key.

When the OLD command is typed on the keyboard, the program you specified in the **program-name** is read into the computer's memory. At this point you may edit, list or run the program.

Another way of telling your disk system how and where to get the information you want is to use the name of a diskette instead of a disk drive number. Example:

TYPE: **OLD DSK.TOMSDISK.COLORDEMO** and press the **ENTER** key.

The disk system will search all attached disk drives until the one is found which contains the diskette named **TOMSDISK**.

Naming Files

The name of a file (file-name) may be from one to ten characters long. The file-name may be any combination of characters except the space and period character. Remember, the Disk Manager Diskette recognizes BOTH upper and lower case characters which may be used to name your files. However the name **MEALS** is different then the name **meals**.

Processing a File

The **OPEN, CLOSE, EOF, PRINT, INPUT, RESTORE** and **DELETE** are the seven main TI BASIC statements which are used when working with files. Besides this, a special statement **CALL FILES** allows you to change the number of files which may be open at the same time. A discussion of how to use these powerful statements follows:

OPEN - STATEMENT

OPEN #file-number:"device.file-name"[,file-organization]
[,file-type][,open-mode][,record-type]

Note: Any file information inside of the brackets ===> [ ] is optional information

The **file-number** and **device.file-name** must be filled out in every **OPEN** statement. The information in the brackets is optional,may be in any order or omitted. For each item which is left out, the computer will assume certain defaults, as described below.

* **file-number** - The **file-number** (which is the number 1-255, or a numeric expression) is assigned to a file by the **OPEN** statement. File numbers may be assigned as you wish. But, remember that file 0 is always the keyboard and screen, and it is always **OPEN**. Also, it cannot be a number of a file which is already **OPEN**.

* **device.file-name** - The **device** is the name of a disk which contains a diskette on which the file you want is stored. The **device** is DSK1 for disk 1, DSK2 for disk 2, DSK3 for disk 3 and DSK4 for drive 4. The **file-name** indentifies a file on a diskette, it may be any name which is 1 to ten characters long. Example:

OPEN #1:"DSK1.ADDRESSES

In place of a disk drive number, you may use **DSK** followed by a period, the name of the diskette, another period,then the **file-name**. Example:

OPEN #1:"DSK.FLOPPY10.FILE:CO"

* **File-organization** - Files are composed of pieces called records. Records can be accessed (read from or written to) in two ways - in an Indian File (Sequentially) or Pick and Choose (Randomly) manner. Records accessed Randomly may be read or written in any order. Records accessed Sequentially must be read or written one after the other.

By entering **SEQUENTIAL** (one record after the other) or **RELATIVE** for random (pick and choose) accessing, you then indicate to the computer the access method you wish to use. When you ask the system to Create a file, then the access method (**SEQUENTIAL** or **RELATIVE**) may be specified. If the **file-organization** is not stated, the computer will make it **SEQUENTIAL**.

* **file-type** - files may be stored on diskettes in human readable **ASCII** characters or in machine-readable **BINARY** characters. The **ASCII** format is the better choice if the file is to be printed or displayed on the screen. However, **BINARY** format is read and written much faster by the computer and takes up less space.

To select **ASCII** format, enter **DISPLAY** as the **file-type**. To select **BINARY** format, enter **INTERNAL**. If no **file-type** is entered, the computer selects **DISPLAY** for the **file-type**.

* **open-mode** - The information in this field tells the computer:

- **INPUT**, read the file - **OUTPUT**, write to the file
- **APPEND**, to end of file - **UPDATE**, read or write

If a file is protected, it may only be read. If the **open-mode** is not entered then the computer selects the **open-mode** as **UPDATE**. The **APPEND** mode can only be set for files which have **VARIABLE** length records.

* **record-type** - File records may be different lengths (**VARIABLE**) or the same length (**FIXED**). Files with fixed length records are processed faster then files which have variable length records. However, fixed length records take up more diskette space.

The maximum record length may be set after the **record-type** field. Be sure there is a space between the end of the **record-type** field and the length. The length is a number or numeric-expression. The largest length of a **FIXED** file record is 255 characters, the largest **VARIABLE** file record is 254 characters long. If you do not set the record length, then the computer selects **record-length** as 80 characters.

The **record-type** for a **RELATIVE** file must be **FIXED**.
SEQUENTIAL files may have either **FIXED** or **VARIABLE** length records. The **SEQUENTIAL** file DEFAULT record-type is **VARIABLE**.

Examples of **OPEN** statements are shown below:

OPEN #1:"DSK1.DEMOS-FILE"

Opens or Creates a diskette file in Drive one. The file name in our example is DEMOS-FILE.

file-type = SEQUENTIAL
mode = UPDATE
format = DISPLAY
record = VARIABLE (max 80 bytes)
The above.....
DEFAULT values are set by the computer.

OPEN #3:"DSK.DEMOS-DISK.YELLOW1",
RELATIVE 99,INTERNAL,OUTPUT,FIXED 80

Opens or Creates a file named YELLOW1.

file-type = RELATIVE
mode = OUTPUT
format = INTERNAL
record-length = FIXED (80 bytes)

At first, 99 records are allocated for file growth (if space is available.)

OPEN #4:Z$,INTERNAL

Opens or Creates a diskette file in Drive four named TOM, if Z$ was equal to "DSK4.TOM"

filetype = SEQUENTIAL
mode = UPDATE
format = INTERNAL
record-length = VARIABLE (80 bytes)

CLOSE - STATEMENT

CLOSE - this statement is like putting a file on the shelf, and closing the door. It tells your computer that the program is not going to talk to the file until further notice. In order to use the file again, you must tell the computer by using the **OPEN** statement.

If a file is not closed, then you may lose the data it contains. Programs which stop because of a **BREAK** statement, or pressing **FCTN <CLEAR>**, or because of some error usually will not be closed. The following actions will properly close a file:

- Use **CLOSE #filenumber** or
- Type: **BYE** or
- Type: **NEW** or
- Enter the Edit mode (Automatically closes all files)

WARNING:

Pressing **QUIT** while in TI BASIC or Extended BASIC, may cause data loss if files were not closed. If you are doing file processing in these languages, then leave only by typing **BYE** and pressing the **ENTER** key.

INPUT - STATEMENT

INPUT - The INPUT statement is used to read data from the files on diskette. In order to use this statement, the file to be used must have previously been opened in an **INPUT** or **UPDATE** mode. **DISPLAY** files must not have over 160 characters in each record. An example is shown below:

```
INPUT #file-number[,REC record-number]:variable-list
\_/ \____ ____/\_____ _____/\_____ ____/
 V       V          V          V
Required Required  Optional   Required
```

- **file-number** - This is the number assigned to a file by a previous **OPEN** statement. A **file-number** is typed in as a # (pound sign), followed by a numeric expression or a number. The number must be between 1 and 255, and be a number of a file which was previously opened.

- **record-number** - This field refers to a record on a file which you want to use. Only with RELATIVE files.

- **variable-list** - The **variable-list** is like a row of boxes, one after the other. There is a comma between each box. The box coresponds to a space to hold a variable. As each variable is read from disk, it is put into the "box" you specify. Note that each "box" has a name. This name stands for or represents the item that you put into that "box".

Examples of **INPUT** statements are shown below:

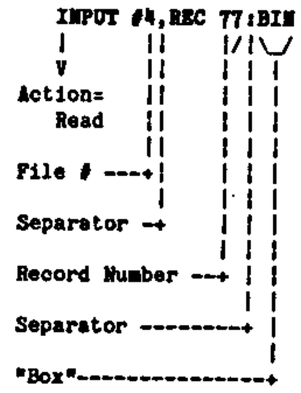
```
   INPUT #1:Z$
   |      |||/
   V      |||
Action    |||
          |||
File # ---+||
          ||
Separator -+|
           |
String -----+
```

The computer reads a variable from the diskette file and puts it into the "box" called Z$. Remember that a string may be from 1 to 255 characters long.

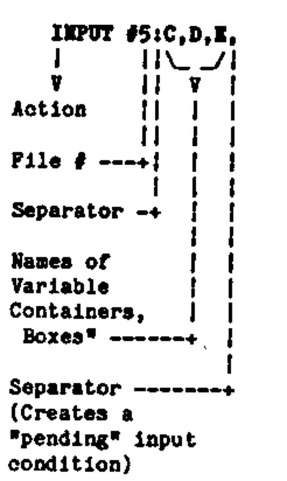
```
   INPUT #3:W,B,M$
   |      ||| / |
   V      |||/  |
Action    |||   |
          |||   |
File #  --+||   |
           ||   |
Separator -+|   |
            |   |
Names of    |   |
Variable    |   |
Containers  |   |
"Boxes" ----+   |
String ---------+
```

The computer reads 3 variables from the diskette file and puts them into the "boxes" called W,B, and M$. Remember that W and B are numbers and M$ is a string.

Further examples of INPUT statements are shown below:



The computer reads the first variable from the record number 77 and puts it into the "box" called BIN.



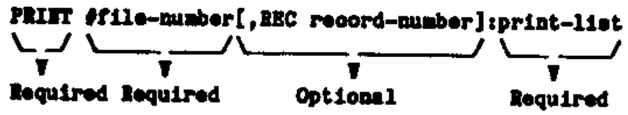
The computer reads 3 variables from the diskette file and puts them into the "boxes" called C,D, and E. Note the comma after variable E tells the computer that there is more data to be read by later statements

If the next INPUT statement has no REC number then the computer reads the next data just after the last read stopped.

If the next INPUT statement contains a REC number then the computer reads the data from the record specified by the REC number.

PRINT - STATEMENT

PRINT - The use of the PRINT statement lets you write data into a diskette file. Only files which have been opened with the OUTPUT, UPDATE, OR APPEND mode will allow you to write data to them with the PRINT statement. An example is shown below:

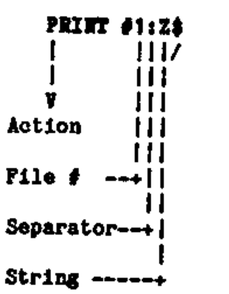


* file-number - This is the number assigned to a file by a previous OPEN statement. A file-number is typed in as a # (pound sign), followed by a numeric expression or a number. The number must be between 1 and 255, and be a number of a file which was previously opened.

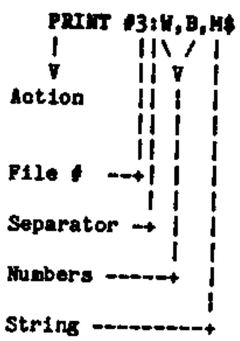
* record-number - This field refers to a record on a file which you want to use. (Only with RELATIVE files).

* print-list - The print-list is like a row of boxes, one after the other. There is a comma between each box. The box cooresponds to a space which holds a variable or expression. As each item is written to the disk, it is put into the record you specify.

Examples of PRINT statements are shown below:

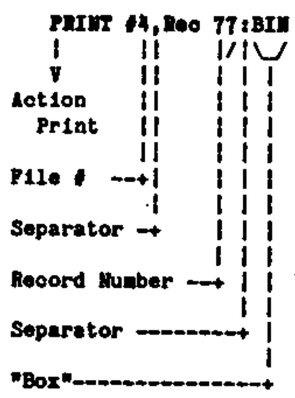


The computer prints the contents of string A$ to the diskette and puts it into the next available data "box" in the file that was opened as #1.

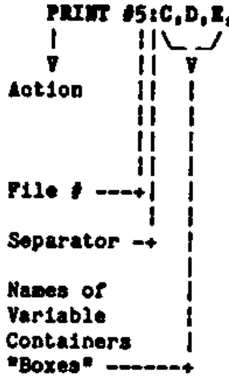


The computer writes 3 variables to the diskette file and puts them into the next 3 available data "boxes" in the file that was opened as #3. Remember that W and B are numbers and M$ is a string.

Further examples of **PRINT** statements are shown below:



The computer takes the content of the "box" called **BIN** and writes on the diskette into record number 77, in the file that was opened as #4.



The computer takes the content of "boxes" C,D and E and puts it into the next available 3 data "boxes" on the diskette file that was opened as #5. The comma after variable **E** tells the computer that there is more data to be printed by later statements.

If the next **PRINT** statement has no **REC** number then the computer places the data in the next available "box" just after the last print stopped.

If the next **PRINT** statement contains a **REC** number, then the computer places the data in the record specified by the **REC** number.

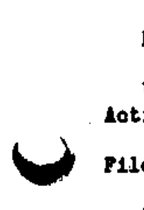
EOF - STATEMENT

EOF - The **End-Of-File (EOF)** When the computer reads the last record of a file, and there are no more records to be read from that file, this is termed the **End-Of-File (EOF)** condition. Therefore, the **EOF** function tells you when the end of the file is reached.

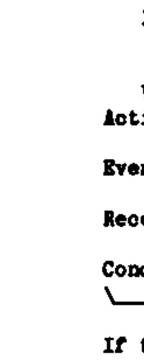
EOF (file-number)
Required Required

* **file-number** - This is the number assigned to a file by a previous **OPEN** statement. A **file-number** is typed in as a ((left parentheses), followed by a numeric expression or a number, followed by a) (right parentheses). The number must be between 1 and 255, and be a number of a file which was previously opened.

Examples of **EOF** statements are shown below:



The computer will print (to the screen) a value of 0 if #3 is not at the end of the file, or 1 if #3 is at end of file, or -1 if the diskette is full and #3 is at the end of file. Remember File #3 must have been previously **OPENED**.



If the computer is at the End Of the File (which was **OPENED** as #77) then transfer control to line 650 in the program.

If this is false, then continue execution at the next program line.

If you want to know how many records are in a relative file, then you must keep a count of the number of records. If the first record of your file is setup to hold this count, then each time you add or delete a record you must add or subtract from the count which is in the "file record count box" of that record. Usually this "box" is stored in the first record - record #0 of a relative file.

RESTORE - STATEMENT

RESTORE - The **RESTORE statement** can be used to tell the computer which record it is going to read next. The point is, the **RESTORE** statement tells the computer where to position itself, so the next **INPUT** or **PRINT** will use (access) that record.

```
RESTORE #file-number[,REC number]
   |      |           |
   V      V           V
Required Required    Optional
```

* **file-number** -This is the number assigned to a file by a previous **OPEN** statement. A **file-number** is typed in as a "#" (pound sign), followed by a numeric expression or a number, The number must be between 1 and 255, and be a number of a file which was previously opened.

* **record-number** - This field refers to a record on a file which you want to use.

Examples of **RESTORE** statements are shown below:

```
   RESTORE #7          The computer will position to
   |       |           Record 0 of the file that was
   V       |           opened as #7. Remember that
Action     |           the INPUT or PRINT statement will
           |           then read from or write to Record 0.
File # ----+
```

```
   RESTORE #4,Rec 77   The computer will position to Record 77
   |       ||    |/    of the File that was OPENED as #4.
   V       ||    |     The next INPUT or PRINT statement
Action     ||    |     will access Record 77.
           ||    |
File #  ---+|    |
           |    |
Separator --+    |
                 |
Record Number ---+
```

If you are working with **RELATIVE** files, then **RESTORE** is usually used to tell the computer to position to the beginning of that file. Remember, that with **RELATIVE** files the record you want to access may be specified in the **INPUT** or **PRINT** statement.

DELETE - STATEMENTS

DELETE - The **DELETE statement** can be used to tell the computer which files to delete from a diskette.

```
DELETE "device.file-name"
   |     |
   V     V
Required Required
```

* **device** - This is the the Disk Drive which holds the diskette on which the file is stored.

  * Disk Drive 1 = DSK1 Disk Drive 3 = DSK3
  * Disk Drive 2 = DSK2 Disk Drive 4 = DSK4

* **file-name** - The **file-name** is any valid file name. Remember, the **file-name** can be from 1-10 characters in length. It may contain any character(s) except the period and the space.

Examples of **DELETE** statements are shown below:

```
   DELETE DSK1.HELP    The computer accesses Disk Drive 1,
   |      | ||\_/      and deletes the file named HELP. If
   V      | || |       the file is not found, then an ERROR
Action    | || |       Code 7 (File does not exist) will be
          | || |       displayed on the monitor/TV
Disk Drive + || |      by the Disk Controller Interface.
             || |
Drive Number +| |      Note: A file can be deleted only if
              | |            it is not protected.
Separator ----+ |
                |
File Name ------+
```

CALL FILES - STATEMENT

CALL FILES - This subprogram allows you to set or change the number of files that can be **OPEN** at the same time. TI BASIC sets this number to three (3). You can set this number from 1 (one file can be **OPEN**) to 9 (nine files can be **OPEN**). The amount of console memory (VDP RAM) which you can use is reduced by 534 characters (bytes) plus each **OPEN** file needs an additional 518 characters (bytes).

An example of the **CALL FILES** statement follows:

```
CALL FILES(number)
     V        V
Action        |
              |
Number of     |
Files Allowed |
to be OPEN ---+
```

The computer will set aside the amount of memory necessary for this number of files to be **OPEN**.

```
*********************************
* The CALL FILES statement      *
* must be used only from the    *
* keyboard, It must also be     *
* immediately followed by a     *
* NEW statement. If you do      *
* not follow these steps to     *
* the letter, you may lose      *
* your program or diskette      *
* data.                         *
*********************************
```


Cataloging Diskette Files Using TI Basic

You can tell the computer to read the file index on a diskette, using TI BASIC. This file index is called the Catalog File. This file has no filename on the diskette.

The following is an example of an OPEN statement, that opens the catalog file on Disk Drive 1:

```
          100 OPEN #2:"DSK1.",INPUT,RELATIVE,INTERNAL
              |   |    || \/|||| \__/ \_____/ \_____/
Line # ---->  |   |    ||  | ||||  |     |       |
              |   |    ||  | ||||  |     |       |
Action -------+   |    ||  | ||||  |     |       |
                  |    ||  | ||||  |     |       |
File Number ------+|   ||  | ||||  |     |       |
                   |   ||  | ||||  |     |       |
Separator ---------+   ||  | ||||  |     |       |
                       ||  | ||||  |     |       |
Disk Drive ------------+   | ||||  |     |       |
                           | ||||  |     |       |
Drive Number ------------+ | ||||  |     |       |
                         | | ||||  |     |       |
Separator ---------------+ | ||||  |     |       |
                           | ||||  |     |       |
Delimiter ----------------+| ||||  |     |       |
                          | ||||  |     |       |
Separator ----------------+| ||||  |     |       |
                           ||||  |     |       |
OPEN mode ------------------+|||  |     |       |
                             |||  |     |       |
Separator --------------------+||  |     |       |
                              ||  |     |       |
File Organization ------------+|  |     |       |
                               |  |     |       |
Separator ---------------------+  |     |       |
                                  |     |       |
File Type ------------------------+     |       |
                                        |       |
```

The Catalog File contains 128 records (0-127). There are four variables in each record. These variables are: one string variable and 3 numeric variables. All are written in **INTERNAL** format.

Record 0:

diskette name	record type	# of diskette sectors	# of available sectors

* **diskette name** - The name of a diskette may be from one to ten characters long, and it can be any combination of characters except the space and period character. Remember, the Disk Manager Diskette recognizes both upper and lower case characters which may be used to name your diskettes.

* **record-type** - Always a zero for record 0

* **number of diskette sectors** -

	# of Sectors	# of Characters/Bytes	
Single-Sided Single-Density =	360	92,160	90K
Single-Sided Double-Density =	720	184,320	180K
Double-Sided Single-Density =	720	184,320	180K
Double-Sided Double-Density =	1,440	368,640	360K

Number of diskette sectors - Number of available sectors = used sectors

Record 1-127

file name	file type	# of file sectors	# of bytes per record

* **file-name** - The name of a file (file-name) may be from one to ten characters long. The file-name may be any combination of characters except the space and period character. Remember, the Disk Manager Diskette recognizes both upper and lower case characters which may be used to name your files.

* **file-type** -

1 = **DISPLAY/FIXED** file
2 = **DISPAY/VARIABLE** file
3 = **INTERNAL/FIXED** file
4 = **INTERNAL/VARIABLE** file
5 = **PROGRAM** or other "memory image" file

Reading the Catalog From TI BASIC

The program we have included here, will RUN under TI BASIC or TI EXTENDED BASIC. Lines 220 through 320 clear the screen, and print the program title on the screen.

```
100 REM TI-99/4A CATALOG PRO
GRAM
110 REM AUTHOR: EASY CODER
130 REM
140 REM ********************
150 REM *                  *
160 REM *     TI-99/4A     *
170 REM *                  *
180 REM * CATALOG PROGRAM  *
190 REM *                  *
200 REM ********************
210 REM
220 LIST-DEVICE=1
230 MY-PICK=1
240 CALL CLEAR
250 M$="TI-99/4A"
260 GOSUB 930
270 M$="CATALOG PROGRAM"
280 GOSUB 930
290 M$="1984"
300 GOSUB 930
310 PRINT
320 PRINT
```

Line 350 calls a subroutine to set-up the screen with the disk drive selection (for cataloging), and where you want the catalog displayed.

```
330 REM
340 REM (* GO SET UP SCREEN
*)
350 GOSUB 980
```

Lines 360 through 410 make a single dimension array of five items. This array is like a five drawer desk. Each drawer will hold one of the descriptions of each type of file (check the subroutine at line 980).

```
360 DIM TYPE$(5)
370 TYPE$(1)="DIS/FIX"
380 TYPE$(2)="DIS/VAR"
390 TYPE$(3)="INT/FIX"
400 TYPE$(4)="INT/VAR"
410 TYPE$(5)="PROGRAM"
```

Lines 420 through 460 position the cursor on the screen, next to the "Drive 1-4 ?" question, then call a subroutine to collect the answer.

```
420 ROW=12
430 COL=22
440 SEL=LIST-DEVICE
450 TOP=4
460 GOSUB 1130
```

Lines 470 through 580 OPEN the catalog file, read in the first record, call a subroutine to display the diskette name, and collect your selection of where you want the catalog displayed.

```
470 LIST-DEVICE=SEL
480 OPEN #1:"DSK"&STR$(LIST
DEVICE)&".",INPUT ,RELATI
VE,INTERNAL
490 INPUT #1:A$,J,J,K
500 MSG$="DISKNAME= "&A$
510 ROW=13
520 COL=3
530 GOSUB 1260
540 ROW=20
550 COL=16
560 SEL=MY-PICK
570 TOP=4
580 GOSUB 1130
```

Lines 590 through 680 set-up the listing device (where you wanted the catalog file to be displayed.

```
590 MY-PICK=SEL
600 ON MY-PICK GOTO 610,640
,660,680
610 F=0
620 CALL CLEAR
630 GOTO 710
640 DEVICE$="TP"
650 GOTO 690
660 DEVICE$="RS232"
670 GOTO 690
680 INPUT "WHICH DEVICE? ":D
EVICE$
```


Line 690 OPENS (gets it ready) the device you selected for the computer to send the catalog to....
Lines 700 through 720 display the column headings for the catalog file on the screen.

```
690 OPEN #2:DEVICE$,OUTPUT
700 F=2
710 PRINT #F:"DSK";STR$(I);"
- DISKNAME= ";A$:"AVAILABLE=
";K;" USED=" ;J-K
720 PRINT #F: :" FILENAME S
IZE TYPE P":"---------- ----
---------- -";
```

Lines 730 through 760 check to see if a key has been pressed.

```
730 CALL KEY(0,K,STATUS)
740 IF STATUS<>1 THEN 770
750 CALL KEY(0,K,STATUS)
760 IF STATUS<>1 THEN 750
```

Lines 770 through 890 read the catalog, then send it to the listing device.

```
770 INPUT #1:A$,I,J,K
780 IF LEN(A$)=0 THEN 850
790 PRINT #F: :A$;TAB(12);J;
TAB(17);TYPE$ (ABS(I));
800 IF ABS(I)=5 THEN 820
810 PRINT #F:K;
820 IF I>0 THEN 730
830 PRINT #F:TAB(28);"Y";
840 GOTO 730
850 CLOSE #1
860 IF F=0 THEN 880
870 CLOSE #2
```

Lines 880 through 990 check to see if you want to continue (cataloging) or return to a "Main Menu" of your own invention (Extended BASIC only Reference line 1360).

```
880 IF MY-PICK<>1 THEN 240
890 PRINT ,,"PRESS ENTER TO
CONTINUE",,
895 PRINT "PRESS FCTN <BACK>
TO RETURN TO
MAIN MENU"
900 CALL KEY(0,X,STATUS)
905 IF X=13 THEN 240
907 IF X=15 THEN 1360
910 IF STATUS=0 THEN 900
920 GOTO 900
```

Lines 930 through 970 display the text string (in M$) centered on the screen.

```
930 REM
940 REM (* CENTER TEXT *)
950 PRINT TAB((32-LEN(M$))/2
);M$
960 PRINT
970 RETURN
```

Lines 980 through 1120 are a subroutine which display the Master Disk and listing selection to indicate where you want the computer to display the catalog.

```
980 REM
990 REM (* SET UP SCREEN *)
1000 PRINT "MASTER DISK (1-4
)?"
1010 PRINT
1020 PRINT
1030 PRINT "WHERE DO YOU WAN
T LISTING ?"
1040 PRINT "1 SCREEN"
1050 PRINT "2 SOLID STATE PR
INTER"
1060 PRINT "3 RS232 INTERFAC
E"
1070 PRINT "4 OTHER"
1080 PRINT "YOUR CHOICE?"
1090 PRINT
1100 PRINT "WHILE PRINTING C
ATALOG"
1110 PRINT "PRESS ANY KEY TO
PAUSE"
1120 RETURN
```


Lines 1130 through 1250 are the subroutine which performs a "Display At" on the screen.

```
1130 REM
1140 REM (* ACCEPT AT *)
1150 CALL HCHAR(ROW,COL,30)
1160 CALL HCHAR(ROW,COL,ASC(
STR$(SEL)))
1170 CALL KEY(0,SELX,STATUS)
1180 IF STATUS=0 THEN 1150
1190 IF SELX=13 THEN 1240
1200 SELX=SELX-48
1210 IF SELX<=0 THEN 1170
1220 IF SELX>TOP THEN 1170
1230 SEL=SELX
1240 CALL HCHAR(ROW,COL,ASC(
STR$(SEL)))
1250 RETURN
```

Lines 1260 through 1350 are a subroutine that performs an "Accept At" from the screen

```
1260 REM
1270 REM (* DISPLAY AT *)
1280 FOR B1=1 TO LEN(MSG$)
1290 CALL HCHAR(ROW,COL,ASC(
SEG$(MSG$,B1,
1)))
1300 COL=COL+1
1310 IF COL<=32 THEN 1340
1320 ROW=ROW+1
1330 COL=1
1340 NEXT B1
1350 RETURN
```

Line 1360 receives control, after you press FCTN <BACK> (Extended BASIC only)

```
1360 RUN "DSK1.LOAD"
```

THE DISK MANAGER

The Disk Manager is a powerful new program that can be loaded from BASIC, Extended BASIC or the Main Title screen. With full screen editing, Configure Manager and implementation of all file utilities at the same time, this program represents a major advance in file and disk management.

* **COPY/RENAME/DELETE/CHANGE PROTECTION** - The **Disk Manager** program allows you to preform all of these functions at the same time. With full screen editing of a catalog list it is quite easy to set up your commands and sit back while the **Disk Manager** does all the work.

* **LOAD and RUN Assembly Files** - This function allows you to load Assembly Language DIS/FIX type files without using the Editor/Assembler Module.

* **CATALOG DISKETTES** - This function generates a listing of all the files on your diskette which you can easily page through since they do not scroll off the screen.

* **COPY DISKETTES** - This is a sector for sector copy and it produces an EXACT copy of the master diskette.

Note: Unlike the other Disk Manager Modules, this Disk Copy writes over any information that was on the Copy (destination) Disk.

* **RENAME DISKETTES** - With this function you can easily Rename any of your diskettes.

* **INITIALIZE DISKETTES** - Diskettes may be initialized in any format from Single-sided Single-density through Double-sided Double-density (depending on your disk drive types).

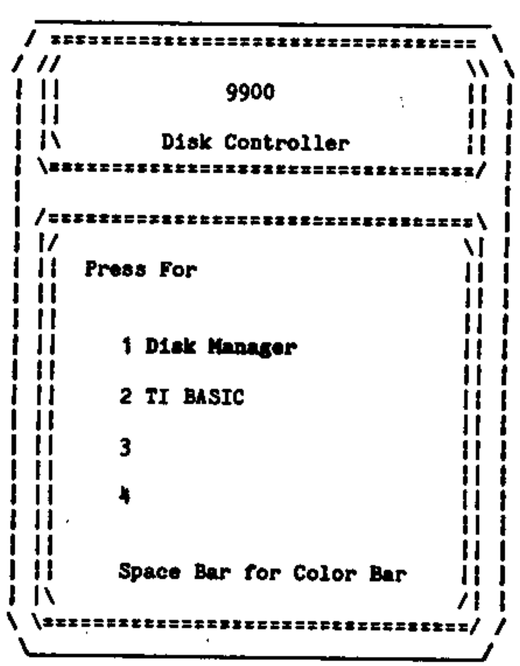
* **DISKETTE TESTS** - The **Disk Manager** offers two ways of testing diskettes in order to find out if the system or the diskette(s) are O.K. The first test is a read only non-destructive test, each sector of the diskette is verified. The second test writes test information all over the diskette, and all of your information that was on the diskette is lost. Be sure and make a backup copy of the diskette you want to test using the destructive test.

THE DISK MANAGER

This section of the manual will show you how to use the **Disk Manager** which works with the **9900 Disk Controller Interface** to provide a quick and useful way of managing your files and diskettes.

Check that the power is turned **ON** to all of the devices in your system. Insert the **Disk Manager Diskette** in drive number one (1).

The Main Tile Screen is shown below:



Pressing key 1 will load and run the **Disk Manager**. The following pages illustrate each of your **Disk Manager** screens and how to use them.

DISK MANAGER QUICK REFERENCE GUIDE

FCTN	KEY	DESCRIPTION
DEL	1	Delete a character.
INSERT	2	Insert a character.
ERASE	3	Erase the input field.
CLEAR	4	Stop inputs, clear the bottom of the screen, and move cursor to Select Option field.
BEGIN	5	Go back to Disk Manager menu screen. From the menu screen go back to the main Title Screen.
PROC'D	6	Request EXECUTE COMMANDS Y/N prompt on the COPY/DELETE/ RENAME/CHANGE PROTECTION File Utilities screen.
AID	7	Not used.
REDO	8	Move cursor back to Beginning of Selected Option (Reenter Inputs).
BACK	9	Same as FCTN 4.
DUMP	0	Dump screen to output device.
	E	Move cursor back one field.
	X	Move cursor ahead one Field.
	S	Move cursor left one character, if start of field, then move cursor to end of previous field.
	D	Move cursor right one character until end of field, then move to start of next field.

CTRL	KEY	
	E	Move cursor back one page.
	X	Move cursor ahead one page.

CONFIGURE MANAGER

The Configure Manger function of the Disk Manager allows you to set up the Disk Manger to match your system and save this information on the Disk Manager Diskete

The following items may be set up in Configure Manager:

* The type of Disk Drives in your system.
  - The number of tracks per side (1-40).
  - The number of Read/Write Sides (1 or 2).
  - Recording Density (Single or Double)
* The default FROM Drive and TO Drive for the various Copy functions.
* The Listing or Output Device (Printer, Disk, etc.)
* The Text and Background colors you wish to use.
* The Single Density Interlace.
* The Double Density Interlace.
* Enable or Disable the Turbo Option for the various Copy functions.

The following **Configure Manager** section will show you the screens the computer will display on your monitor or TV and explain what you need to enter into the computer when using the **Configure Manager Utility.**

CONFIGURE MANAGER

The Configure Manager option will allow you to specify the characteristics of the disks in your disk system and the listing device you wish for output. The listing device may be a printer or a diskette.

Note that the configuration you have specified may be saved to the Disk Manager diskette. This saving of the configuration need be done only once. The Disk Manager will then automatically use your configuration.

```
DISK MANAGER V 2.1

Select Option: 4

        1. File Utilities

        2. Disk Utilities

        3. Disk Tests

        4. Configure Manager
```

The cursor flashes in the space following the Select Option. If this is the first time you have run the Disk Manager or you wish to re-configure your system, then press the 4 key, The following screen will then appear.

CONFIGURE MANAGER

```
                    Configure Manager

                  Drive Configuration
    This is DSK1 ===> Drive 1            Drive 2
 Tracks/Side = 40 ===> Tracks/Side : 40  Tracks/Side : 40
# of Sides = 1 or 2 => No. Sides   : 2   No. Sides   : 2
Density is Single ===> Density S/D : D   Density S/D : D
       or Double

                    Drive 3             Drive 4
                    Tracks/Side : 40    Tracks/Side : 40
                    No. Sides   : 2     No. Sides   : 2
                    Density S/D : D     Density S/D : D

  This is the Drive  From               To                 This is the Drive
     to READ FROM ===> Drive No. : 1     Drive No. : 2   <== to WRITE TO
 (If you have 1 Disk                                       (If you have
  Drive, type 1)   +=> Output configuration                 1 Disk Drive
                       RS232.BA=1200                          type 1)

      Color of ===> Screen Color:  5   Text Color : 16  <== Color of
       Screen                                             Characters
    Background       TURN PAGE WITH CTRL E - CTRL X

                          COLORS

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

 See Software
 Switch Options
 (Next Page)
```

Software Switch Options

```
        RS232.BA=baud-rate.DA=x.PA=y.TW.MU.CH.EC.CR.LF
        \_/    \__ __/           \/ \/ \/ \/ \/ \/
         V        V       |     |  )  )  )  )  )  )
         |        |       |     |  |  |  |  |  |  |
         | (Default = 1200)     |  |  |  |  |  |  |
Device --+        |       |     |  |  |  |  |  |  |
                  |       |     |  |  |  |  |  |  |
Baud Rate = 110,\_|       |     |  |  |  |  |  |  |
  300,600,1200, /         |     |  |  |  |  |  |  |
  2400,4800,   /          |     |  |  |  |  |  |  |
  or 9600 ____/           |     |  |  |  |  |  |  |
(Sets the Transmission Rate)|   |  |  |  |  |  |  |
                          |     |  |  |  |  |  |  |
Data Bits = 7 or 8 <------+     |  |  |  |  |  |  |
(Sets the number of Data Bits   |  |  |  |  |  |  |
 in each character. Default = 7)|  |  |  |  |  |  |
                          |     |  |  |  |  |  |  |
Parity = Odd, Even or None \____+  |  |  |  |  |  |
       = O    E       N    /     |  |  |  |  |  |
                  (Default = O)   |  |  |  |  |  |
Two Stop Bits                     |  |  |  |  |  |
(When this switch is   \__________+  |  |  |  |  |
 on stop bits are used /             |  |  |  |  |
                     (Default = 1)   |  |  |  |  |
Nulls                                |  |  |  |  |
(When this switch is       \_______+  |  |  |  |
 on Nulls (Blanks) are used)/          |  |  |  |
                          (Default = no)|  |  |  |
Check Parity                           |  |  |  |
(When this switch is on Parity \_______+  |  |  |
 Checking is performed        /           |  |  |
                            Default = no)  |  |  |
Echo Off                                   |  |  |
(When this switch is on the Echo \_________+  |  |
 Character feature is turned off /            |  |
                           (Default = none)    |  |
CRLF Off                                       |  |
(When this switch is on the Carraige Return \__|  |
 and Line Feed Options are turned off       /     |
                                  (Default = on)  |
LF Off                                           |
(When this switch is on, then the \______________+
 Line Feed option is turned off   / (Default = on)
```


CONFIGURE MANAGER

Interlace sets up the physical location of the sectors within each track on the diskette. With the diskette spinning at 300 RPM the read/write time can be greatly improved if the sector numbers are not physicaly consequtive within the track. By the time one sector is read and the data is transfered to the computer, one or more sectors have passed under the read/write head. If the sector numbers were physicaly consequtive, the Disk Controller would have to wait until the next sector to be read came around again. This constant waiting would greatly slow down the read/write/transfer time for file accesses. The main item that determines the proper interlace is the transfer time. Some Modules and or Languages may access and transfer files faster with a different interlace than the defaults of 7 and 10. The interlace is placed on the diskette during initialization.

The Single-density interlace of 7 and the Double-density interlace of 10 produces very good read/write/transfer times for general use. However, you may want to experiment with different values for diskettes that will mainly be used with selected Modules and Languages.

```
/          Page 2 of 2                \
/                                       \
|                                       |
|   Single Density Interlace : 7        |
|   (1-10)                              |
|                                       |
|   Double Density Interlace : 10       |
|   (1-11)                              |
|                                       |
|   Enable Turbo Option Y/N  : N        |
|                                       |
|                                       |
|   SAVE CONFIGURATION TO DISK : Y      |
|                                       |
|                                       |
|                                       |
|                                       |
|                                       |
|                                       |
\   PRESS CTRL E FOR PREVIOUS PAGE      /
```

The Turbo Option is used during Copy functions. Enabling the Turbo Option turns off the Verify after Write command while using the Disk Manager. If you are Copying to freshly initialized diskettes with EVERY sector verified (no bad sectors) then this Option will greatly reduce your copying time.

Note: Do not use the Turbo Option on diskettes with bad sectors!

FILE UTILITIES

Files, as we mentioned earlier, are places where you store information. Managing the files is the job of the File Utilities. It will copy, rename, delete and change the files protection.

The File Utilities section preforms the following functions:

- Copy Files onto another diskette.
- Move Files off of one diskette onto another.
- Delete Files from a Diskette.
- Rename Files on a Diskette.
- Change Write Protection
  - Write protection marks a file so the computer will know that it should not alter that file in any way.
- Load and Run Assembly Language DIS/FIX type Files without the Editor/Assembler Module.

The following File Utilities section will show you the screens the computer will display on your monitor or TV and explain what you need to enter into the computer when using the File Utilities.

FILE UTILITIES

The File Utilities option allows you to preform various operations on your files.

```
DISK MANAGER V2.1

Select Option: 1

      1. File Utilities

      2. Disk Utilities

      3. Disk Tests

      4. Configure Manager
```

Pressing key 1 will select the File Utilities function and the following screen will appear.

File Copy/Delete/Rename/Change Protection Screens

```
                              ______________________________________
                             /           File Utilities             \
                            |                                        |
Select Option 1 ===> Select Option: 1                                |
   which                    |                                        |
 allows you                 |                                        |
  to work                   |  1. Copy / Rename / Delete             |
    with                    |     Change Protection                  |
    your                    |                                        |
diskette files              |  2. Load and Run (Assembly file)       |
                            |                                        |
After you select            |                                        |
  Option 1                  |                                        |
   then                     |                                        |
    this                    |                                        |
   screen                   |                                        |
    |  \                    |                                        |
    |   \                   |                                        |
    |    \                  |                                        |
    |    appears            |                                        |
    |       \               |                                        |
    |        \              |                                        |
    |         \              \______________________________________/
    |          \             ______________________________________
    |                       /           File Utilities             \
    |                      |                                        |
    |                      | Select Option: 1                       |
    |                      |                                        |
  then                     |                                        |
    |                      |  1. Copy / Rename / Delete             |
 proceed                   |     Change Protection                  |
    |                      |                                        |
   to                      |  2. Load and Run (Assembly file)       |
    |                      |                                        |
Input a Drive              |                                        |
Number or press      ====> Drive No.: d                             |
ENTER.                     |  Disk Name: oooooooooo                 |
                           |  Free oooo  Used: oooo                 |
The diskette Name          |                                        |
and Space will             |                                        |
be shown                   |                                        |
                           |                                        |
  and the                  |                                        |
Computer displays    ====> BUILDING FILE LIST                       |
                            \______________________________________/
```


This screen allows you to change all the following options on your files in one operation with the **Disk Manager:**

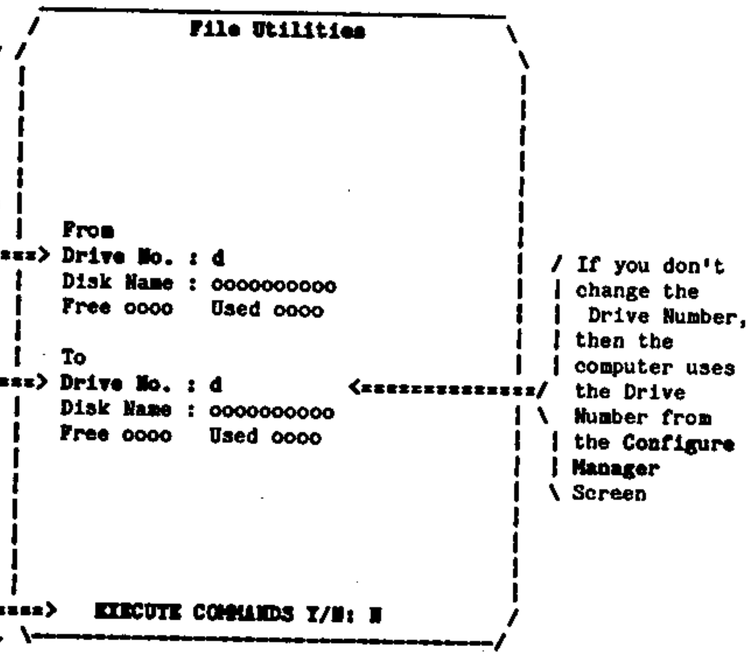
```
                                                        Pages
                        Disk        Disk     Current   Remaining
                       Number       Name      Page     | to view
                          \          |       Number--+ |
                           |     ____V___            | |
                  _________V_/___________\___________V_V____
                 /   DSKo.oooooooooo      Page 0 of 0       \
 # of Files =======> ooo Files    Free oooo  Used oooo <==== Diskette Free
                |                                         |    and Used Space
                |      Filename   Size     Type       P   |
                | CMD ---------- ---- ----------- -      |
Type in the     |  C  oioioioioi ssss ttt/ttt 111  P     | Type in the
CMD field to =======> M oioioioioi ssss ttt/ttt 111  U <===P field to
Copy, Move or   |  D  oioioioioi ssss ttt/ttt 111  T     | change a files
Delete a file.  |  N  oioioioioi ssss ttt/ttt 111  i     | protection.
                |  N  oioioioioi ssss ttt/ttt 111  i     |
C=Copy          |  N  oioioioioi ssss ttt/ttt 111  i     | P=Protect
M=Move (copy to |  N  oioioioioi ssss ttt/ttt 111  i     | U=Unprotect
  another disk  |  N  oioioioioi ssss ttt/ttt 111  i     | T=Temporary
  then delete it)| N  oioioioioi ssss ttt/ttt 111  i     |   unprotect for
D=Delete        |  N  oioioioioi ssss ttt/ttt 111  i     |   file rename
                |  N  oioioioioi ssss ttt/ttt 111  i     |   then reprotect
Type in the     |  N  oioioioioi ssss ttt/ttt 111  i     |
Filename field ========> oioioioioi ssss ttt/ttt 111  i  |
to rename a file. | N oioioioioi ssss ttt/ttt 111  i     |
                |  N  oioioioioi ssss ttt/ttt 111  i     |
                |  N  oioioioioi ssss ttt/ttt 111  i     |
                |  N  oioioioioi ssss ttt/ttt 111  i     |
                |  N  oioioioioi ssss ttt/ttt 111  i     |
                 \    EXECUTE COMMANDS Y/N : N           /
After all the     \_____________________________________/
changes have
been made
then
```

FCTN 6 <PROCEED> - Places you at the EXECUTE COMMANDS Y/N: N field.
or
The ENTER key pressed at the last field places you at the EXECUTE COMMANDS Y/N: N field.
or
CTRL <E> = Display previous page, CTRL <X> = Display next page.

Order of Operations on the Copy/Rename/Delete/Change Protection Screen

1. Unprotect all files marked with a U or T in the P field
2. Delete all files marked with a D in the CMD field.
3. Rename all files whose name you changed in the Filename field.
4. Protect all files marked with a P or T in the P field.
5. Copy all files marked with a C or M in the CMD filed.
6. Delete all files marked with a M in the CMD field.

After selelcting Option 1
and you set-up
to Copy one or
or more Files,
then this display
appears.....



The Disk Drive #
is taken from the
previous screens
Drive Number.

The Cursor is
moved to
this field.

Input a Drive
number or press
ENTER then the
Diskette name and
space will be
shown.

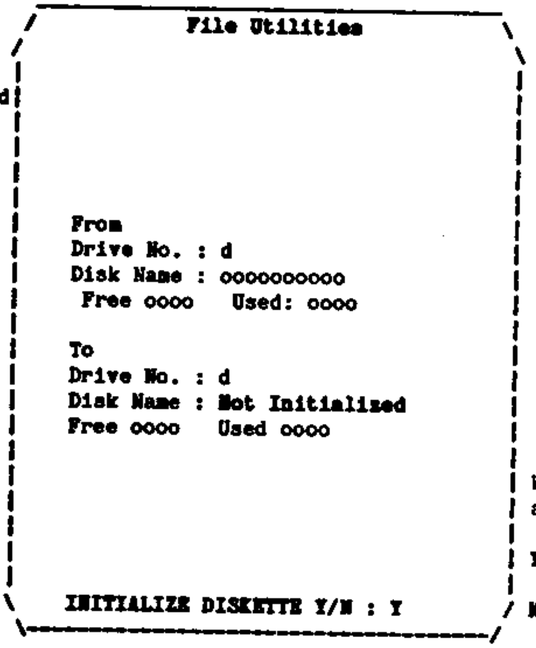
Then type Y
to start the copy.

If you don't
change the
Drive Number,
then the
computer uses
the Drive
Number from
the Configure
Manager
Screen

But...

If the TO Drive
is not initialized

Then this screen
appears



When this prompt
appears type

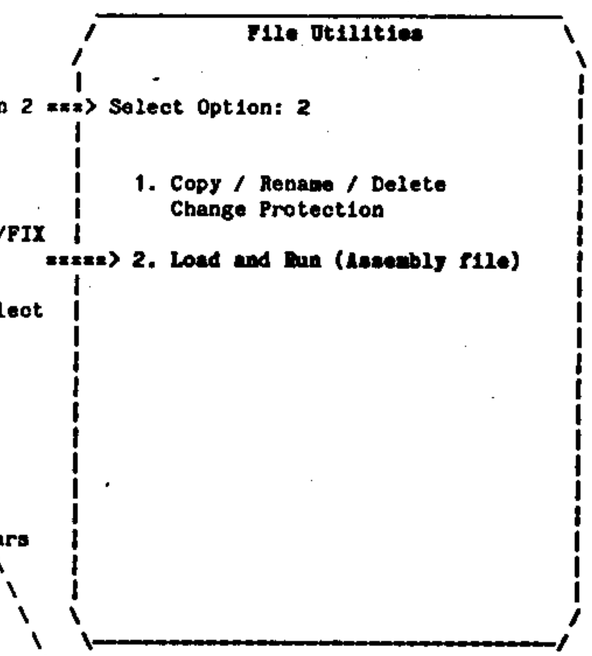
Y to Initialize

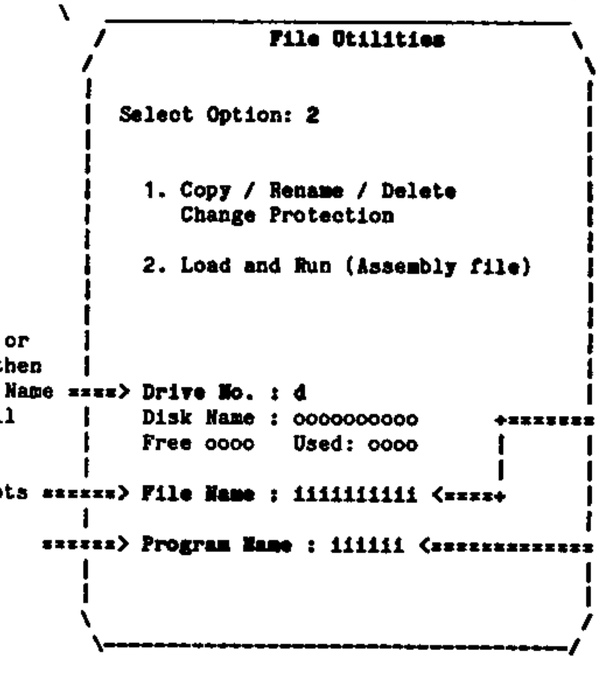
N to Abort

Load and Run Screens

Select Option 2
which allows
you to
Load and Run
a Assembly
Language DIS/FIX
type file.

After you select
Option 2
then
this
screen
appears
then
proceed
to





Input a
Drive Number or
press ENTER then
the Diskette Name
and Space will
be shown,

& these prompts
appear.

Type in the
File name and
press ENTER.

Type in the
programs start
name and press
ENTER.
If it is an
Auto-Run program
type in any
character.

DISK UTILITIES

The Disk Utilities section preforms the follow functions:

- **Catalog** - The Directory is read and the list of files on the diskette, along with the file size, protection status, file name(s) are sent to the screen. You can "turn the pages" and view the catalog. A copy of what is on the screen can be sent to the listing device (usually a printer), by pressing FCTN 0

- **Copy** the entire contents of one diskette onto another diskette.

  Note: This is a sector for sector copy and it will write over (destroy) any data on the destination diskette. To copy files onto another diskette without destroying its files use the Copy function in the **File Utilities** section.

- **Rename** a Diskette

- **Initialize** a Diskette

  Note: When you initialize a diskette the Disk Controller Interface writes selected data in each sector and inbetween each sector for timing marks. All data previously on the diskette is destroyed.

The following **Disk Utility** section will show you the screens the computer will display on your monitor or TV and explain what you need to enter into the computer when using the **Disk Utilities**.

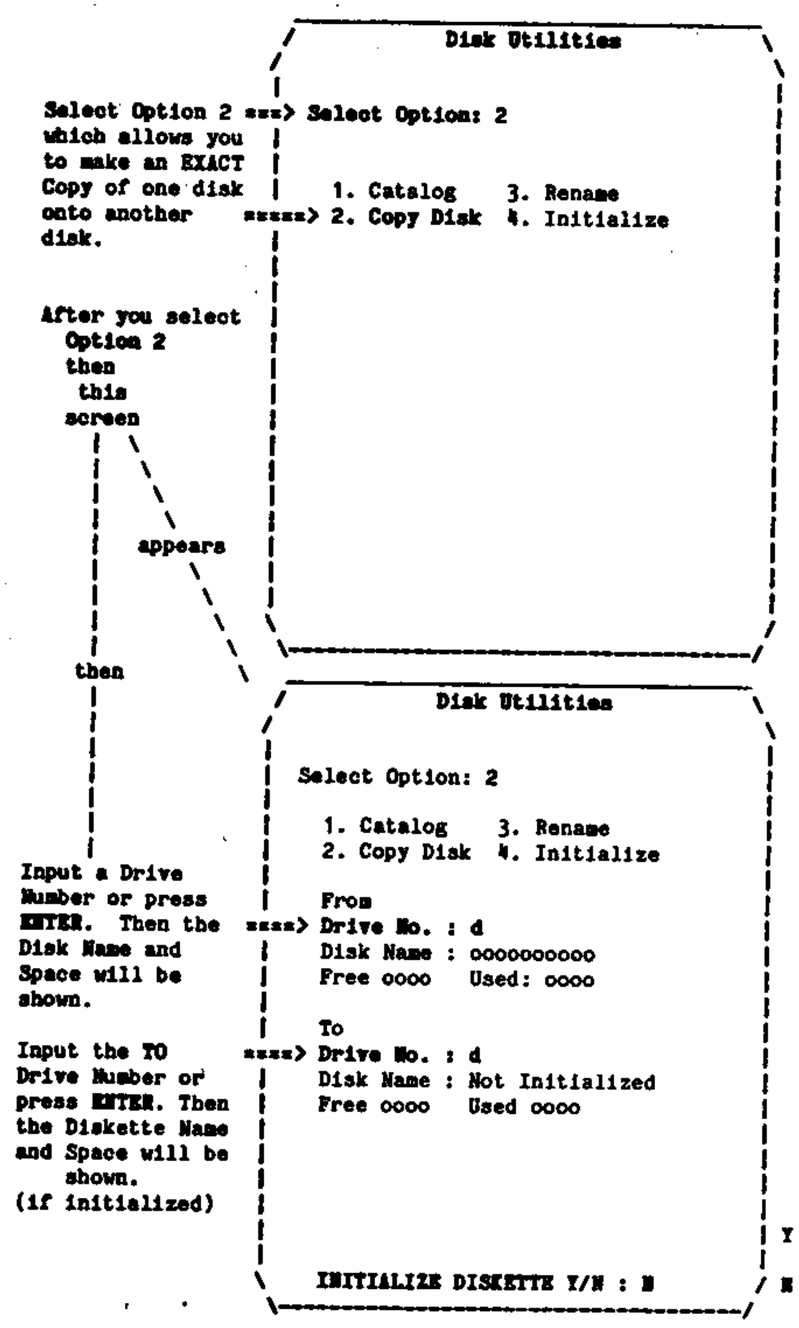
DISK UTILITES

The Disk Utilities section allows you to perform various operations on the entire diskette.

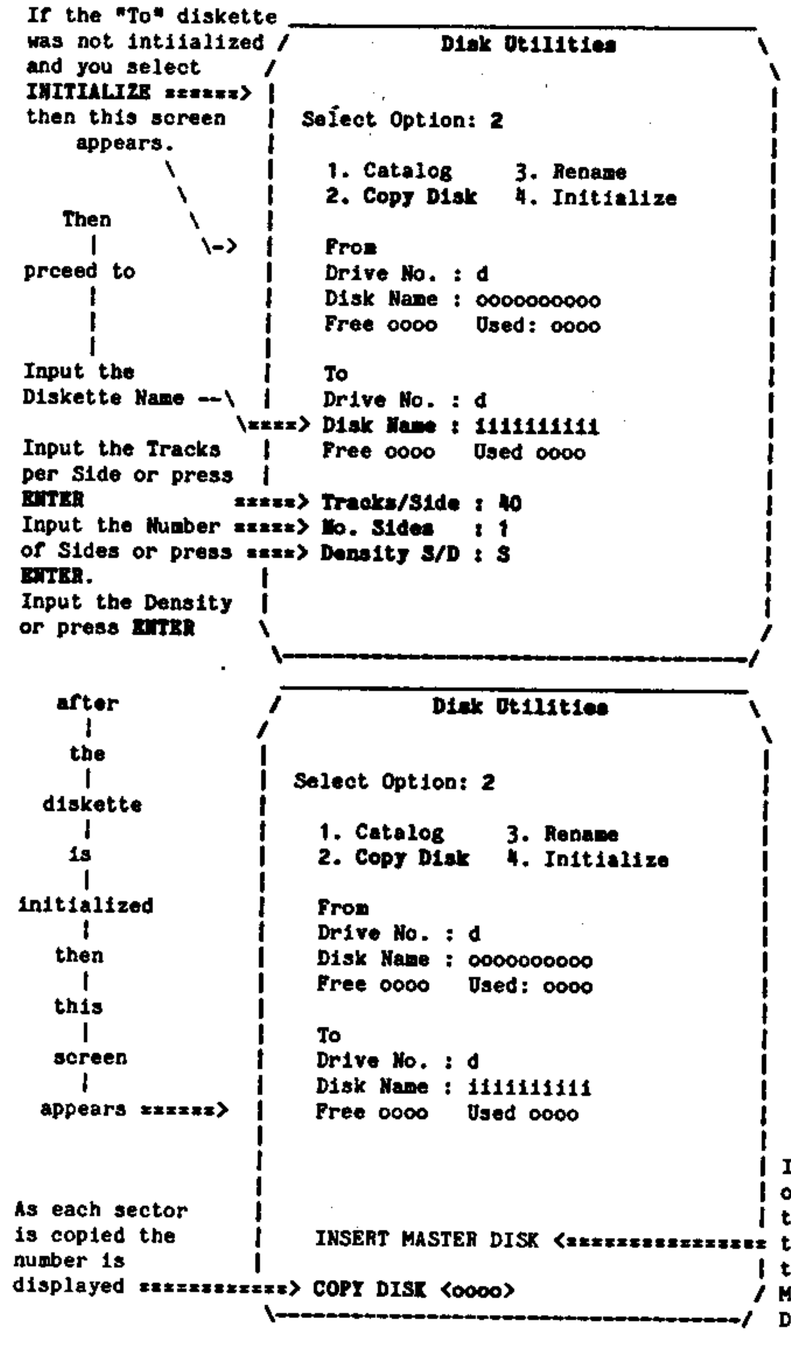
```
 D I S K  M A N A G E R  V 2.1

Select Option: 2

        1. File Utilities

        2. Disk Utilities

        3. Disk Tests

        4. Configure Manager
```

Pressing key 2 will select the **Disk Utilities** function and the following screen will appear.

Copy Disk Screens



Copy Disk Screens

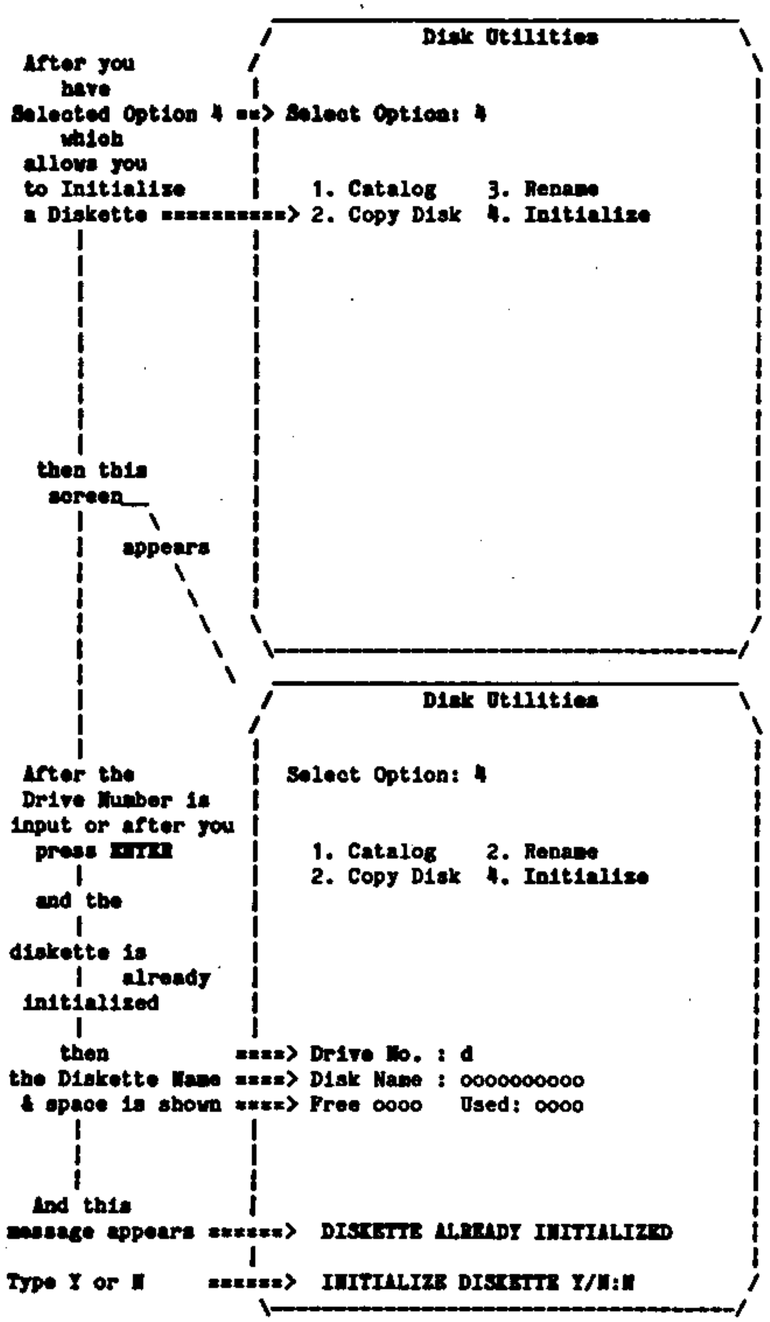


Rename Screens

```
                        /              Disk Utilities              \
                       /                                            \
                       |                                            |
Select Option 3 ===> Select Option: 3                               |
which allows           |                                            |
you to Rename          |                                            |
a Diskette ==========> 1. Catalog    3. Rename                      |
                       |  2. Copy Disk  4. Initialize               |
                       |                                            |
After you Select       |                                            |
  Option 3             |                                            |
  then                 |                                            |
  this                 |                                            |
  screen__             |                                            |
    |     \            |                                            |
    |    appears       |                                            |
    |        \         \____________________________________________/
    |         \
    |          \       /              Disk Utilities              \
    |                 /                                            \
    |                 |                                            |
    |                 |  Select Option: 3                          |
    |                 |                                            |
  then                |    1. Catalog    3. Rename                 |
    |                 |    2. Copy Disk  4. Initialize             |
 proceed              |                                            |
    |                 |                                            |
   to                 |                                            |
    |                 |                                            |
Input a Drive         |                                            |
Number or press       |                                            |
ENTER.           =====> Drive No. : d                              |
The Disk Name         |    Disk Name : oooooooooo                  |
Space is shown.       |    Free oooo    Used: oooo                 |
                      |                                            |
Input the new     ====> New Disk name: iiiiiiiiii                  |
Disk Name and         |                                            |
press ENTER.          |                                            |
                      \____________________________________________/
```


Initialize Screens

```
                         /              Disk Utilities              \
                        /                                            \
                        |                                            |
Select Option 4 ===> Select Option: 4                                |
   which                |                                            |
allows you              |                                            |
to Initialize           |  1. Catalog    3. Rename                   |
a Diskette              |  2. Copy Disk  4. Initialize               |
                        |                                            |
After you Select        |                                            |
  Option 4              |                                            |
  then                  |                                            |
   this                 |                                            |
  screen__              |                                            |
    |     \             |                                            |
    |    appears        |                                            |
    |        \          \____________________________________________/
    |         \
    |          \        /              Disk Utilities              \
    |                  /                                            \
  then                 |                                            |
    |                  |  Select Option: 4                          |
 proceed               |                                            |
    |                  |  1. Catalog    3. Rename                   |
   to                  |  2. Copy Disk  4. Initialize               |
    |                  |                                            |
Input a Drive          |                                            | If Diskette is
Number or press        |                       +--------------------+-initialized then
Enter.                 |                       |                    | the Disk Name
The Diskette           |                       |                    | will be shown
Name and Space         |                       |                    |
will be shown       ====> Drive No. : d        V                    |      and
(if initialized)       |  Disk Name : oioioioioi                    |
                       |  Free oooo    Used: oooo                   | This prompt will
Input Tracks per       |                                            | appear on the
Side(1-40) or       ====> Tracks/Side : 40             +------------+-screen.
press Enter.        ====> No. Sides   : 1              |            |
Input the Number    ====> Density S/D : S              |            | If it is not
of Sides 1-2 or        |                               |            | initialized then
press ENTER.           |                               V            | the Disk Name
Input the Density      \    INITIALIZE DISKETTE Y/N: N              / field is left
S-D or press            \__________________________________________/  blank.
ENTER.                    INSTALL MANAGER Y/N : Y - Installs the Disk Manger
                                                    on the new diskette.
```



If you type Y, then the Initialization prompts will appear and, you can proceed in the normal fashion. If you type N then it will ABORT.

DISK TESTS

Diskettes, we mentioned earlier, are made of a delicate material. There will be times when it may be necessary to test a diskette. (Or, you may wish to test a diskette to see for yourself how it is done.)

There are two types of **Disk Tests**. These tests are:

* **Read Only** - This is a non-destructive only test of the diskette. Each place on the diskette where information can be stored is read and checked. You have the option of telling the computer that you want it to log the errors it uncovers as it tests the diskette. This information can be sent to a output device (usually a printer). And the test may be run in a loop (when it finishes, it starts over).

* **Destructive** - This is a write and read test. If you run this test be sure to back up the files on the diskette you are going to test. Remember this test will erase all of your files on the diskette during this test.

The following **Disk Tests** section will show you the screens the computer will display on your monitor or TV and explain what you need to enter into the computer when using the **Disk Tests**.

DISK TESTS

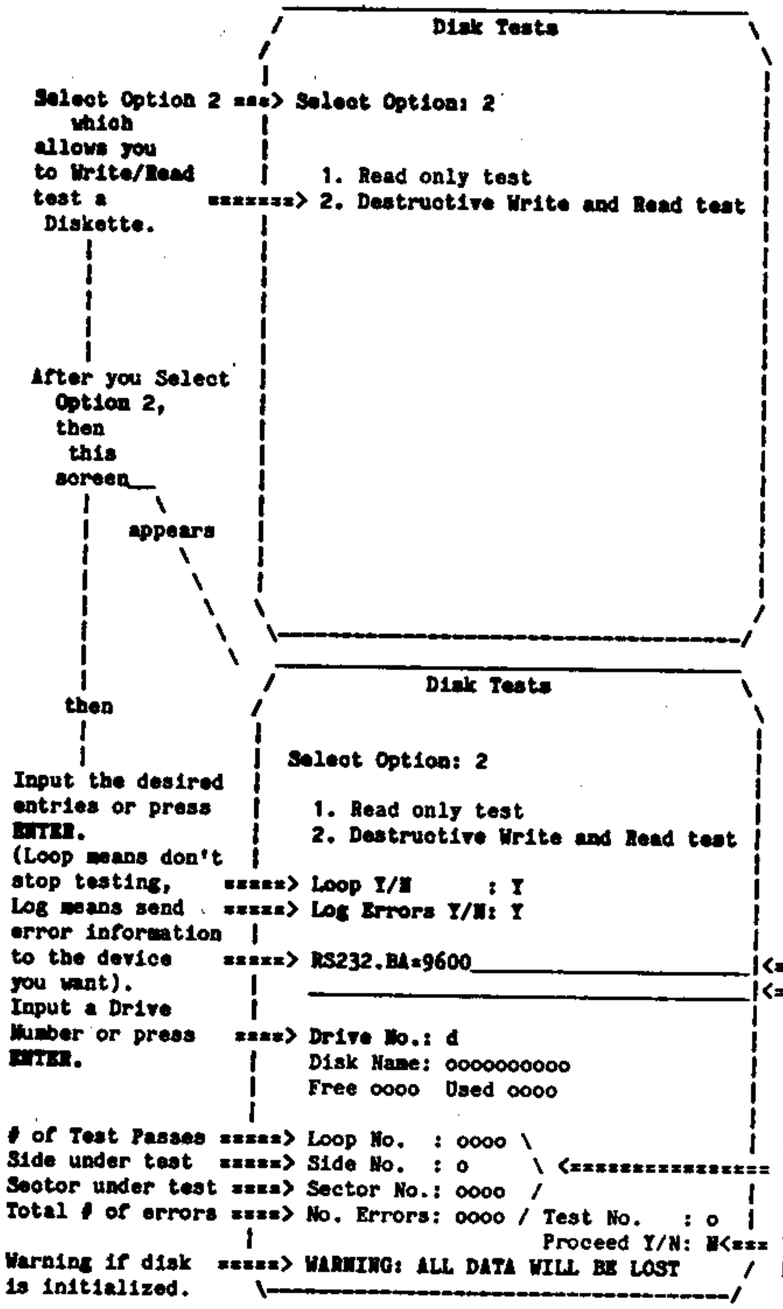
The Disk Tests section allows you to preform two types of diskette/drive tests.

```
     _______________________________
   /   D I S K  M A N A G E R  V 2.1  \
  |                                    |
  |  Select Option: 3                  |
  |                                    |
  |          1. File Utilities         |
  |                                    |
  |          2. Disk Utilities         |
  |                                    |
  |          3. Disk Tests             |
  |                                    |
  |          4. Configure Manager      |
  |                                    |
  |                                    |
   \__________________________________/
```

Pressing key 3 will select the **Disk Tests** function and the following screen will appear.


```
                        ________________________________________
                      /              Disk Tests                  \
Select Option 1 ===> Select Option: 1                             |
   which            |                                             |
allows you          |                                             |
to Read Test        |   1. Read only test                         |
a Diskette.         |   2. Destructive Write and Read test        |
    |               |                                             |
    |               |                                             |
    V               |                                             |
After you Select    |                                             |
  Option 1          |                                             |
  then              |                                             |
   this             |                                             |
  screen__          |                                             |
    |     \         |                                             |
    |   appears     |                                             |
    |        \      |                                             |
    |         \      \___________________________________________/
    |          \      ___________________________________________
    |                /              Disk Tests                   \
   then             |                                             |
    |               |  Select Option: 1                           |
    V               |                                             |
Input the desired   |                                             |
entries or          |   1. Read only test                         |
press ENTER.        |   2. Destructive Write and Read test        |
(Loop means don't   |                                             |
stop testing   =====> Loop Y/N       : Y                          |
Log means send =====> Log Errors Y/N: Y                           |     The "Log"
error information   |                                             |
to the device  =====> RS232.BA=9600_______________________________|<== Device is one
you want).          |  _______________________________________   |<== you set or it
Input a Drive #     |                                             |     is taken from
  or press ENTER, ====> Drive No.: d                              |   the Configure
then the Diskette ====> Disk Name: oooooooooo                     |   Manager screen
name and space is ====> Free oooo  Used: oooo                     |   (Only if Log=Y).
    shown.          |                                             |
# of Test Passes =====> Loop No.   : oooo \                       |
Side under test  =====> Side No.   : o     \ <=============== This data is
Sector under test ====> Sector No.: oooo   /                      |   displayed by
Total # of errors ====> No. Errors: oooo /                        |   the computer
                    |                                             |   as the Diskette
                     \___________________________________________/    is being tested.
```

Pressing **FCTN 4** - **FCTN 5** - **FCTN 8** or **FCTN 9** will stop the test at any time and return you to the approiate input prompt/screen.



Pressing FCTN 4 - FCTN 5 - FCTN 8 or FCTN 9 will stop the test at any time and return you to the approiate input prompt/screen.

THE TOOL SHED UTILITIES

The 9900 Disk Controller Interface allows you to write programs in BASIC or Extended BASIC that can access and use the powerful 9900 subprogram utilities. These utilities have all the features one would expect from such advanced functions. They are written in Assembly Language and reside in the 9900 Disk Controller Interface, which allows you maximum use of the system memory. This section provides the details for using these new commands and programming statements.

The TOOL SHED provides explanations of the following new commands:

MPEEK for reading values and characters from CPU memory.

MPOKE for writing values and characters to CPU memory.

VPEEK for reading values and characters from Video memory.

VPOKE for writing values and characters to Video memory.

WRTRG for writing values to the Video Registers.

MOVEM for moving blocks of memory around.

EXEC for executing Assembly Language subprograms by address.

MGR for loading and running the 9900 Disk Manager program.

DELETE "LD-CMDS" for loading the new Commands link names into Low Memory Expansion (Extended BASIC only)

CONVENTIONS

The following conventions are used to define the synyax of the utility commands.

* ADDR = Address in VDP or CPU memory

* BIAS = This is the value which is used with the VPEEK,MPEEK,VPOKE and MPOKE commands and it offsets the STRINGS that are passed. TI BASIC and EXTENDED BASIC use the BIAS to properly display characters on the screen. This Bais is only added to strings

  Bias of 0 = no offset added or subtracted from the string.
  Bias of 96 = a value of 96 is added to all string characters.
  bias of 160 = a value of 96 is subtracted from all characters.

* SV = String variable, such as A$ or B$, etc.

* S = Direct string, such as "HELLO" or "GOODBY".

* NV = Numeric variable, such as N or M. Where N or M stand for a numeric value.

* N = Direct number, such as 5 or 9999.

Notes:

1. When you MPEEK or VPEEK the contents of memory into a string variable, the number of bytes "peeked" into the string variable is determined by the previous length of the string.
   i.e. A$ = "123456789" CALL MPEEK(0,96,A$) will MPEEK 9 bytes.

2. String expresions and numeric expresions such as: A$&B$ or N+C or N+30 are not allowed.

3. String variables and Numeric Variables can be regular variables such as: A$ or COUNT or they can be single dimension arrays such as: A$(3) or COUNT(5). Multi-dimensioned arrays are not allowed.

The limitations in items 2 and 3 above were implimented to greatly improve the speed of execution of the new CALL routines.

CALL MPEEK

This subprogram allows you to read values and characters from Central Processor Memory, addresses 0 through 65535. This MPEEKing can return information or data that was stored into memory by the computer or by your use of MPOKE. The various memory locations are described in the CPU memory map in the appendixes.

Note: Using MPEEK on the memory mapped GROM addresses may cause your system to lock up, which will cause the loss of your program in memory. If this occurs turn off your console, wait 5 seconds and turn it back on.

Basic Syntax:

```
CALL MPEEK(addr,bias,sv,nv[,...][;addr,bias,sv,nv,...])
```

Extended Basic Syntax:

```
CALL LINK("MPEEK")(addr,bias,sv,nv[,...][;addr,bias,sv,nv,...])
```

Examples:

BASIC:

```
100 CALL MPEEK(33728,0,A,B)
110 PRINT A;B
```

Reads the Random Number Seed and puts into the numeric variables A and B
Prints the values for A and B onto the screen

Extended BASIC:

```
100 CALL INIT
110 DELETE "LD-CMDS"
120 CALL LINK("MPEEK")(33740
,0,A,B)
130 PRINT Ax256+B
```

Initializes Memory Expansion
Loads the new Commands into Low Memory Expansion
Reads the pointer to the Sound Table in VDP RAM and puts it into numeric variables A and B
prints the address of the Sound Table in VDP RAM onto the screen

CALL MPOKE

This subprogram allows you to write values and characters into Central Processor Memory, addresses 8192-16383 and 32768-65535. If you have a Mini Memory module you can MPOKE addresses 28672-32767. This MPOKEing will store information or data into memory to be used by the computer or by your use of MPEEK. The various memory locations are described in the CPU memory map in the appendixes.

Note: Indiscriminate use of MPOKE on memory addresses 32768-40959 may cause your computer to lock up.

BASIC Syntax:

CALL MPOKE(addr,bias,s,sv,n,nv[,...][;addr,bias,s,sv,n,nv,...])

Extended BASIC Syntax:

CALL LINK("MPOKE")(addr,bias,s,sv,n,nv[,...][;addr,bias,s,sv,n,nv,...])

Examples:

BASIC:

```
100 A$="This is a test"          Sets up A$ with a message
110 B$="              "          Sets up B$ to read 14 bytes
120 CALL MPOKE(12288,0,A$)       Stores the message in Low Memory Expansion
130 CALL MPEEK(12288,0,B$)       Reads the message in Low Memory Expansion
140 PRINT B$                     Print the message that was read into B$ on the
                                 screen
```

Extended BASIC:

```
100 CALL INIT :: DELETE "LD-     Initializes Memory Expansion and Loads the new
CMDS"                            Comands into Low Memory Expansion
110 A$="This should print Th     Sets up A$ with a message
is should not"
120 B$=RPT$(" ",17)              Sets up B$ to read 17 bytes
130 CALL LINK("MPOKE")(12288     Stores the message into Low Memory Expansion
,0,A$)
140 CALL LINK("MPEEK")(12288     Reads the first 17 bytes of the message that was
,0,B$)                           stored in Low Memory Expansion
150 PRINT B$                     Prints the message that was read into B$ on the
                                 screen
```


CALL VPEEK

This subprogram allows you to read values and characters from Video Processor Memory, addresses 0 through 16384. This VPEEKing can return information or data that was stored into memory by the computer or by your use of VPOKE. The various memory locations are described in the VDP memory map in the appendixes.

Note: Using VPEEK on memory addresses greater than 16384 may cause your system to look up, which will cause the loss of your program in memory. If this occurs turn off your console, wait 5 seconds and turn it back on.

BASIC Syntax:

CALL VPEEK(addr,bias,sv,nv[,...][;addr,bias,sv,nv,...])

Extended BASIC Syntax:

CALL LINK("VPEEK")(addr,bias,sv,nv[,...][;addr,bias,sv,nv,...])

Examples:

BASIC:

```
100 A$="12345678901234567890     Sets up A$ for a length of 32 characters
123456789012"
110 PRINT "This is line 2";      Places a message on line 2
: : : : : : : : : : : : : : :
: : : : : : :
120 CALL VPEEK(32,160,A$)        Reads the second line on the screen into A$
130 CALL VPOKE(384,96,A$)        Writes A$ to the 12th line of the screen
```

Extended BASIC:

```
100 CALL INIT                    Initializes Expansion Memory
110 DELETE "LD-CMDS"             Loads the new Commands into Low Memory Expansion
120 A$=RPT$(" ",32):: DISPLA     Sets up A$ for a length of 32 characters and
Y AT(2,1):"This is line 2"       displays a message on line 2
130 CALL LINK("VPEEK")(32,16     Reads the second line on the screen into A$
0,A$)
140 CALL LINK("VPOKE")(384,9     Writes A$ to the 12th line of the screen
6,A$)
```

CALL VPOKE

This subprogram allows you to write values and characters into Video Processor Memory, addresses 0 through 16384. This VPOKEing can store information or data into VDP memory to be used by the computer or by your use of VPEEK. The various memory locations are described in the VDP memory map in the appendixes.

NOTE: Using VPOKE on addresses greater than 16384 or indiscriminate use of VPOKE may cause your system to lock up.

BASIC Syntax:

CALL VPOKE(addr,bias,s,sv,n,nv[,...][;addr,bias,s,sv,n,nv,...])

Extended BASIC Syntax:

CALL LINK("VPOKE")(addr,bias,s,sv,n,nv[,...][;addr,bias,s,sv,n,nv,...])

Examples:

BASIC:

```
100 A$="THIS SHOWS ON LINE 1  Sets up A$ with a message
2"
110 CALL VPOKE(384,96,A$)     Writes A$ on the 12th line of the screen
```

Extended BASIC:

```
100 CALL INIT                 Iinitializes Memory Expansion
110 DELETE "LD-CMDS"          Loads the new Commands into Low Memory Expansion
120 A$=RPT$(" ",32)           Sets up A$ for a length of 32 characters
130 CALL LINK("MPEEK")(40960  Reads 32 bytes starting at 40960 into A$
,160,A$)
140 CALL LINK("VPOKE")(736,9  Writes A$ to the 24th line or the screen
6,A$)
```


CALL WRTRG

This utility loads the VDP Write Only Registers.

The eight VDP registers are numbered 0 through 7 and are used to control the following features:

- Register 0 - Used with register 1 to select the Grahics Mode, also controls whether an external video picture can be shown.
- Register 1 - This register and register 0 determines which Graphics Mode is selected. Screen blank control, sprite size and interrupt enable/disable are also controlled by this register.
- Register 2 - Defines the location of the screen image table.
- Register 3 - Defines the location of the color description table.
- Register 4 - Defines the location of the pattern (character) table.
- Register 5 - Defines the location of the sprite characteristics table.
- Register 6 - Defines the location of the sprite pattern table.
- Register 7 - Sets the Foreground (Text Mode only) and Background (All Modes) color.

BASIC Syntax:

CALL WRTRG(register number,value[;register #,value;.....])

Extended BASIC Syntax:

CALL LINK("WRTRG")(register number,value[;register #,value;.....])

Examples:

BASIC:

```
100 CALL WRTRG(7,1)           Writes 1 into VDP register 7 and changes the
                              screen color to Black.
```

Extended BASIC:

```
100 CALL INIT :: DELETE "LD-  Initializes Memeory Expansion and Loads the new
CMDS"                         Commands link names into Low Memory Expansion.
110 CALL LINK("WRTRG")(7,1)   Writes 1 into VDP register 7 and changes the
                              screen color to Black.
```

NOTE: Register number and value can be direct numbers or numeric variables. Also see the Appendixes for more information on the VDP registers.

CALL MOVEM

This utility will allow you to move blocks of memory from one location to another. A block of memory is one or more bytes long. A typical use for this Utility is to move new screen images from VDP or Expansion Memory RAM into the Screen Image Table in VDP RAM. Also the color tables, character definition tables, sprite definition tables, etc. may be changed by this technique.

BASIC Syntax:

CALL MOVEM(type#1-4,from-addr,to-addr,#of bytes to move[;.....])

Extended BASIC Syntax:

CALL LINK("MOVEM")(type#1-4,from-addr,to-addr,#of bytes to move[;.....])

Type of Move:
1 = Move from VDP RAM to VDP RAM
2 = Move from VDP RAM to CPU RAM
3 = Move from CPU RAM or ROM to VDP RAM
4 = Move from CPU RAM or ROM to CPU RAM

Examples:

BASIC:

CALL MOVEM(1,0,704,64)	Copies lines 1 and 2 (64 bytes) on the screen to lines 23 and 24 on the screen
CALL MOVEM(2,0,40960,768)	Copies the entire screen (768 bytes) to address 40960 (Hex >A000) in Expansion Memory
CALL MOVEM(3,40960,0,768)	Copies 768 bytes from address 40960 (Hex >A000) in Expansion Memory to the screen
CALL MOVEM(4,0,40960,8192)	Copies the entire console ROM, 8K, to Expansion Memory address 40960 (Hex >A000)

EXTENDED BASIC:

100 CALL INIT :: DELETE "LD-CMDS" 100 CALL LINK("MOVEM")(3,0,0,128)	Initializes Memory Expansion and Loads the new Commands link names into Low Memory Expansion. Copies the first 128 bytes of console ROM to the top of the screen

CALL EXEC

This utility will execute a machine langauge code routine. The routine may be in ROM or RAM. This means, for example you can execute your own machine langauge routines in Expansion Memory by calling its starting address.

Note: The Assembly routine must end with a RETURN or you may lock up your computer.

BASIC Syntax:

CALL EXEC(addr[;addr;addr;...])

Extended BASIC:

CALL LINK("EXEC")(addr[;addr;addr;...])

Examples:

BASIC:

100 CALL MPEEK(3,0,A) 110 CALL EXEC(A)	Reads the start address of the power-up routine. Executes a warm power-up and displays the TI Title Screen. Turn your console OFF and then ON to display the 9900 Title Screen.

Extended BASIC:

100 CALL INIT :: DELETE "LD-CMDS" 110 CALL LINK("MPEEK")(3,0,A):: CALL LINK("EXEC")(A)	Initializes Memory Expansion and Loads the new Commands link names into Low Memory Expansion. Reads the start address of the power-up routine and executes a warm power-up and displays the TI Title Screen. Turn your console OFF and then ON to display the 9900 Title Screen.

CALL MGR

This utility loads and runs the Disk Manager from diskette, when you are in TI BASIC or Extended BASIC. Remember that any BASIC or Extended BASIC program you have in memory will be lost, unless you SAVE it to diskette befor using CALL MGR. NOTE: A Diskette containing the Disk Manger must be in Drive 1,2,3 or 4.

BASIC syntax:

CALL MGR

Extended BASIC Syntax:

Command Mode: Program Mode:
CALL MGR

APPENDIX A (Error Codes while using TI BASIC)

Error codes which can occur when your Disk System is being used are in this appendix. Error codes that are not related to your disk system are given in the TI 99/4A User's Reference Guide.

The error code is shown as a two digit number. The digit on the left indicates the Disk Manager command or the command of the BASIC statement which was in use when the error occured. The digit on the right specifies the type of error.

LEFT DIGIT	COMMAND OR STATEMENT
0	OPEN
1	CLOSE
2	INPUT
3	PRINT
4	RESTORE
5	OLD
6	SAVE
7	DELETE
8	* not used *
9	EOF (End Of File)

RIGHT DIGIT	TYPE OF ERROR
0	The selected disk drive was not found.
1	DEVICE OR FILE WRITE PROTECTED - Remove PROTECTION using the CC Disk Manager Diskette. Or remove the tape from the write- protect slot of the diskette.
2	BAD OPEN ATTRIBUTE - One or more of the OPEN options were illegal or did not match the file's profile.
3	ILLEGAL OPERATION - This error won't occur when the Disk System is used. The error could be caused if input from a printer was requested.
4	OUT OF SPACE - The diskette is full or you have tried to OPEN more files then are allowed to be open.
5	ATTEMPT TO READ PAST END OF FILE
6	DEVICE ERROR - Will occur if the Diskette is not initialized or damaged. And, if the disk drive is not working right or power was removed while the system was printing, etc.
7	FILE ERROR - The selected file or diskette is not in a drive or you are trying to read a BASIC or EXTENDED BASIC program as if it were data.

Examples:

I/O ERROR 60	Disk Drive not connected or Drive power is OFF.
I/O ERROR 25	Trying to read too many items from the file (past EOF).

APPENDIX B (CPU Memory Map)

OVERALL SYSTEM MAP

0	**CONSOLE ROM**	8K Bytes
8191	Interrupt Vectors, XOP Vectors, GPL Interperter XMLLNK Vectors, Low-level cassette DSR etc.	
8192	**LOW MEMORY EXPANSION RAM**	8K Bytes
16383	Generally holds Assembly Language programs, not used by Extended Basic programs.	
16384	**DSR** Device service routines ROM	8K Bytes
24575	Determined by CRU bit setting Disk Controller, RS232, P-Code etc.	
24576	**CARTRIDGE PORT ROM** (& Mini-Mem RAM)	8K Bytes
32767	12K of Extended BASIC ROM. Upper 4K @ 28672 - 32767 is flipped to page in another 4K for a total of 12K	
32768	**RAM MEMORY MAPPED DEVICES VDP, GROM, SOUND & SPEECH**	8K Bytes
32768	Duplication of scratch pad ram @ 33536 - 33791	
33023		
33024	Duplication of scratch pad ram @ 33536 - 33791	
33279		
33280	Duplication of scratch pad ram @ 33536 - 33791	
33535		
33536	**CPU SCRATCH PAD RAM** 256 bytes	
33791		
33792	**SOUND CHIP**	
34816	**VDP READ DATA**	
34818	**VDP STATUS (MSBy)**	
35840	**VDP WRITE DATA**	
35842	**VDP WRITE ADDRESS** (to write set MSB of the MSBy to 01)	
36864	**SPEECH READ**	
37888	**SPEECH WRITE**	
38912	**GROM/GRAM READ DATA**	
38914	**GROM/GRAM READ ADDRESS**	
39936	**GROM/GRAM WRITE DATA**	
39938	**GROM/GRAM WRITE ADDRESS**	
40959		
40960	**HIGH MEMORY EXPANSION**	24K Bytes
	Extended Basic High Memory Usage, Free space end pointed to by CPU RAM PAD address 33670	
	Numeric values	
	Line number table	
	X-Basic program space	
65535		

APPENDIX C (VDP Memory Map)

VDP MAPS WITH THE DIFFERENT LANGUAGES

	BASIC		EXTENDED BASIC
0	SCREEN IMAGE TABLE Start PATTERN DESC TABLE	0	SCREEN IMAGE TABLE Start PATTERN DESC TABLE Start SPRITE PATTERN TABLE
	+96 Offset (Bias)		+96 Offset (Bias)
767	END SCREEN IMAGE	767	End Screen Image
768	COLOR TABLE	768	SPRITE ATTRIBUTE
799		879	
880	PATTERN DESC TABLE 768+8*character number= the decimal address 1008= Char 30 1016= Char 31 1024= Char 32 1032= Char 33 etc.	880	PATTERN DESC TABLE 768+8*character number= the decimal address 1008= Char 30 1016= Char 31 1024= Char 32 1032= Char 33 etc.
2047		1919	
2048	PABS STRINGS SYMBOL TABLES NUMERIC VALUES LINE NUMBER TABLE PROGRAM SPACE	1920	SPRITE MOTION TABLE
13783		2047	
13784	DISK FILE BUFFERS CALL FILES(3)	2048	COLOR TABLE
16383		2079	
		2080	Without MEM-EXPANSION same use as BASIC With MEM-EXPANSION PABS STRINGS SYMBOL TABLES (Numeric values, Line Number Table & Program are in High MEM-EXPANSION)
		13783	
		13784	DISK FILE BUFFERS CALL FILES(3)
		16383	

APPENDIX D (VDP Registers)

TMS 9918A VDP REGISTERS

REG #	BIT #	DESCRIPTION
0	0-5	Reserved for future use must be 0's
	6	M3 (mode bit 3)
	7	External VDP enable 0=disable 1=enable
1	0	4/16 K selection 0=4027 RAM 4K 1=4108/4116 RAM 16K
	1	Blank enable/disable 0=blank 1=display
	2	Interrupt enable 0=disable 1=enable
	3	M1 (mode bit 1) 1=Text mode
	4	M2 (mode bit 2) 1=Multicolor mode
	5	Reserved
	6	Size (sprite size select) 0=8x8 bit 1=16x16 bit
	7	Magnification 0=1x 1=2x

M1	M2	M3	Mode	
0	0	0	Graphics I mode	32 column
0	0	1	Graphics II mode	Bit-Mapped
0	1	0	Multicolor mode	64 column
1	0	0	Text mode	40 column

2	0-3	Reserved
	4-7	Screen table base address. R2 * 1024
3	0-7	Color table base address. R3 * 64
4	0-4	Reserved
	5-7	Pattern (character) descriptor table base. R4 * 2048
5	0	Reserved
	1-7	Sprite attribute table base address. R5 * 128
6	0-4	Reserved
	5-7	Sprite pattern table base address. R6 * 2048
7	0-3	Text mode forground color.
	4-7	Background (screen) color all modes.
Status	0	Interrupt Flag bit.
	1	5 or more sprites on the same row.
	2	Sprite coincidence.
	3-7	Number of the 5th sprite when bit 1 is on.

APPENDIX D - continued

EIGHT WRITE ONLY VDP REGISTERS

Bit# 0 1 2 3 4 5 6 7

Decimal Value: 128 64 32 16 8 4 2 1

0

0	0	0	0	0	0	M3 bit-map	Ext VDP

Basic = 0 Extended Basic = 0

1

4/16K RAM	BLANK SCREEN	Interrup Enable	M1 text	M2 multi	0	SIZE	MAG

Basic = 224 Extended Basic = 224

2

0	0	0	0	SCREEN TABLE BASE ADDRESS x 1024

Basic = 0 Extended Basic = 0

3

COLOR TABLE BASE ADDRESS x 64

Basic = 12 Extended Basic = 32

4

0	0	0	0	0	CHARACTER BASE ADD x 2048

Basic = 0 Extended Basic = 0

5

0	SPRITE ATTRIBUTE TABLE BASE ADDRESS x 128

Basic = 6 Extended Basic = 6

6

0	0	0	0	0	SPRITE PAT BASE ADD x 2048

Basic = 0 Extended Basic = 0

7

TEXT MODE FORGROUND COLOR	SCREEN COLOR (all modes)

Basic = 7 Extended Basic = 7

APPENDIX E (Interlace Patterns)

Interlace Selection	Sector Order within the Tracks
Single Density	
1	0 1 2 3 4 5 6 7 8
2	0 2 4 6 8 1 3 5 7
3	0 3 6 1 4 7 2 5 8
4	0 4 8 3 7 2 6 1 5
5	0 5 1 6 2 7 3 8 4
6	0 6 3 1 7 4 2 8 5
* 7	0 7 5 3 1 8 6 4 2
8	0 6 3 8 5 2 7 4 1
9	0 5 1 6 2 7 3 8 4
10	0 4 8 3 7 2 6 1 5
Double Density	
1	0 2 4 6 8 10 12 14 16 1 3 5 7 9 11 13 15 17
2	0 3 6 9 12 15 1 4 7 10 13 16 2 5 8 11 14 17
3	0 4 8 12 16 2 6 10 14 1 5 9 13 17 3 7 11 15
4	0 5 10 15 2 7 12 17 4 9 14 1 6 11 16 3 8 13
5	0 6 12 1 7 13 2 8 14 3 9 15 4 10 16 5 11 17
6	0 7 14 3 10 17 6 13 2 9 16 5 12 1 8 15 4 11
7	0 8 16 6 14 4 12 2 10 1 9 17 7 15 5 13 3 11
8	0 9 1 10 2 11 3 12 4 13 5 14 6 15 7 16 8 17
9	0 10 2 12 4 14 6 16 8 1 11 3 13 5 15 7 17 9
* 10	0 11 4 15 8 1 12 5 16 9 2 13 6 17 10 3 14 7
11	0 12 6 1 13 7 2 14 8 3 15 9 4 16 10 5 17 11

* = Default selection for the Disk Manager.

IMPORTANT ADDENDUM
Please Read

An extended memory card of at least 32K is needed to access the CorComp Disk Manager. This is because the Disk Manager is loaded into the Memory card for storage when first accessed.

On the Disk Manager Diskette are two files named FORTH and FORTHSAVE. If you use the COPY portion of the File Utilities in the Disk Manager to copy this file onto your TI FORTH diskette you will be able to Load and Run FORTH from the Disk Manager without using your Editor/Assembler Module.

This FORTH file is a revised loader, but it is still compatible with the Editor/Assembler module's Load and Run function and the Mini Memory Module provided you have memory expansion.

****
PLEASE NOTE
****

If your CorComp Disk Controller Title Screen doesn't appear after the proper power-up procedure, check to see if you have a CorComp RS232 Card in your TI Peripheral Expansion Box. If you do, remove it and again try the power-procedure. The CorComp Disk Controller Title Screen should now appear. If it doesn't appear, please call CorComp at 1-714-630-2903 for further information.

Only factory tested double density drives with step rates of 20 milliseconds or less are recommended for double density operation. While some single density drives may appear to function properly, unreliable operation and lost data may result from the use of these drives.

The COPY DISK function under Disk Utilities in the CorComp Disk Manager Program does a sector by sector copy. It will write over any information on the TO diskette. It is mainly intended for making fast back up copies of your diskettes. The FILE COPY function under File Utilities should be used to copy multiple diskettes onto one diskette.

Some modules contain their own power up routines such as the Terminal Emulator II and some of the Educational modules. If a module or a Basic program that accesses a module does not function properly when selected from the 9900 DISK CONTROLLER title screen then you must select them through the second menu screen. When the 9900 DISK CONTROLLER title screen appears just press the space bar twice and then make your module or TI Basic selection.