

QUIKDISK™

HIGH SPEED FLOPPY DISK

FOR COMMODORE 64

USER'S MANUAL

Published by cgrs MICROTECH, INC.
P.O. Box 102
Langhorne, Pa.
19047

215-757-0284

The Quikdisk Manual is fully protected under the Federal Copyright Laws. No part of this manual may be copied or duplicated in any way without the prior written permission of cgrs MICROTECH, INC.

Second Edition
Copyright 1984 cgrs MICROTECH, INC.

cgrs
MICROTECH

P.O. BOX 102 • LANGHORNE, PA 19047

The Quikdisk Manual has been carefully prepared and reviewed. It is believed to be free from errors. MICROTECH, however, cannot assume any responsibility for inaccuracies. MICROTECH makes no representation of any kind about the suitability for intended use and will accept no consequential, incidental, or other liability arising from the use of this product.

The Quikdisk High Speed Floppy Disk System is covered by a limited 90 day warrenty.

TABLE OF CONTENTS

1. GETTING STARTED

A.	General Information.....	pg.5
	Compatibility.....	pg.6
	Hardware.....	pg.7
	System Diagram.....	pg.7
	Specifications.....	pg.8
	Software.....	pg.9
B.	Unpacking.....	pg10
C.	Installation.....	pg10
D.	Operation	
	Diskette Handling.....	pg13
	Start up.....	pg14
	Display Directory.....	pg15
	Formatting (New) a disk....	pg16
	Duplicating your master....	pg18
	Transferring a program	pg20
	(1541 to QUIKDISK)	

2. USER'S GUIDE

A.	DISK COMMANDS.....	pg22
	Introduction.....	pg22
	LOAD.....	pg25
	FILE NAME LOADING.....	pg25
	WILD CARD CHARACTERS.....	pg26
	STRING VARIABLE LOADING....	pg26
	SAVE.....	pg28
	DATA FILE HANDLING.....	pg29
	OPEN.....	pg30
	INPUT#.....	pg33
	PRINT#.....	pg35
	GET#.....	pg37
	CLOSE.....	pg38

1. GETTING STARTED

A. General Information

Congratulations on your purchase of Quikdisk, the high speed floppy disk system designed especially for the Commodore 64 series computers. It is part of the Pedisk Series of floppy disk systems and is optimized to provide extremely high speed and reliable operation. The Quikdisk System consists of a small disk controller module, a cable assembly, and a standard disk drive assembly.

The controller will interface to three inch, five and one quarter inch, or eight inch disk drives. The Quikdisk controller module plugs into the cartridge slot of the computer and a flat cable connects to the drive. The interface is the same as that used on IBM PC's and the QUIKDISK drives can also be used as expansion drives for those computers.

QUIKDOS software emulates a Commodore disk drive by intercepting the disk commands from the machine. Quikdisk operates, however, by transferring data directly from the diskette to the computer memory. With a data transfer rate of 250,000 bits per second, over ten times faster than the serial bus, Quikdisk provides emulation at the fastest possible speed. A full set of disk utilities is also provided.

B.	PROGRAMMING THE DISK CONTROLLER	
	Changing device number.....	pg39
	Disk Kernal Routines.....	pg40
	Disk Organization.....	pg49
	Directory Format.....	pg50
	Error codes.....	pg54
	Troubleshooting.....	pg57
	Memory Map.....	pg57
	Subroutines.....	pg58
3.	DISK HELPER	
A.	LOADING.....	pg59
B.	COMMANDS.....	pg60
	BLOCK DISPLAY.....	pg60
	DUPLICATE.....	pg61
	GO.....	pg63
	HELP.....	pg63
	KILL.....	pg63
	LOAD.....	pg64
	MONITOR.....	pg64
	NEW (FORMAT).....	pg64
	QUIT TO BASIC.....	pg65
	RENAME.....	pg66
	SAVE.....	pg66
	UNKILL.....	pg67
	VALIDATE.....	pg68
	EXECUTE.....	pg68
	\$-DIRECTORY.....	pg69
4.	QUICK REFERENCE COMMAND SUMMARY.	pg70
5.	PROBLEM REPORT SHEET.....	pg72
6.	INDEX.....	pg73

COMPARE THE SPEED!

The loading time for "Lemonade.C3" from the Business I public domain diskette is 32 seconds on the 1541 Disk Drive. With Quikdisk 580-1, the time taken to load the same program is reduced to 2.5 seconds.

1541/QUIKDISK COMPATIBILITY

Quikdisk was designed to emulate the 1541. There are some differences which should be noted. Quikdisk uses computer memory in the (\$C000) block. Any program which uses this area will most likely interfere with Quikdisk operation. Check with the software manufacturer if you are not certain. Quikdisk also provides helper programs for disk maintenance in lieu of the disk commands provided by 1541. Commands not implimented include B-R, B-W, U1, U2, V, I, etc.

Quikdisk uses an industry standard physical recording format defined by IBM. It does not use the format defined by Commodore and used on the 4040/1541. This means that you cannot purchase programs recorded for a 1541 disk drive.

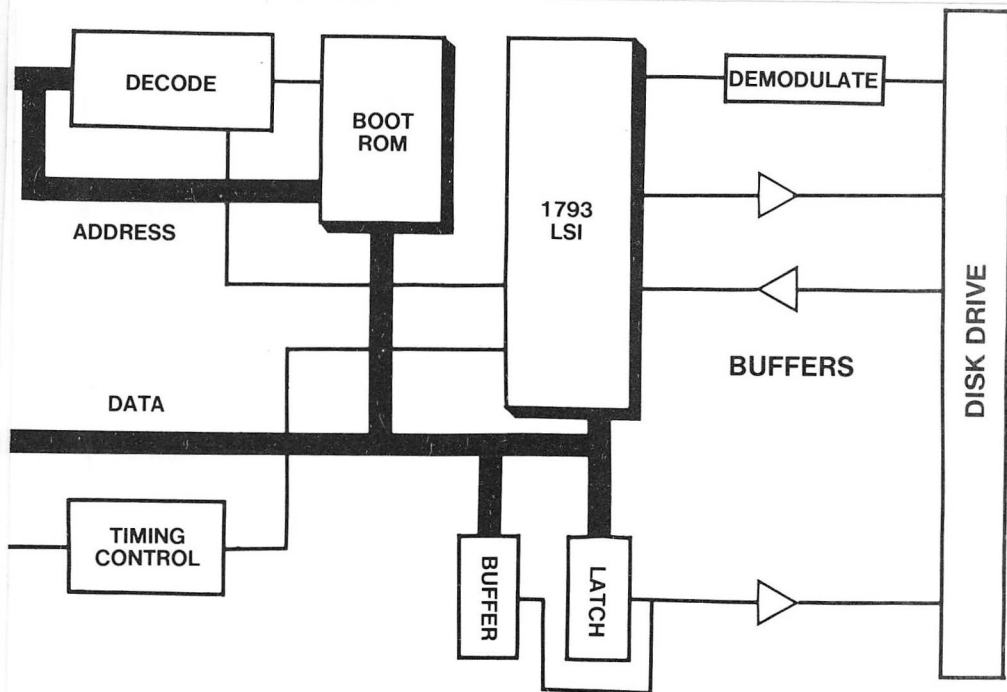
You can, however, transfer such programs if both a 1541 and a Quikdisk are connected to the same computer. Programs that are copy-protected will not transfer. Contact the manufacturer of the software to purchase a Quikdisk version. You may also contact cgrs MICROTECH to inquire.

HARDWARE

The Quikdisk controller module is a 3.5 by 3.5 "L" shaped circuit board with cover that plugs into the cartridge expansion slot of the computer.

The controller contains a 1793 LSI circuit, memory decoding logic, a control latch, buffers and a "boot ROM".

A standard 34 wire cable connects to the drive. Jumper options are provided to control several types of drive, single or double sided. When the complete QUIKDISK package is purchased, all jumpers are preset, ready-to-go.



SYSTEM BLOCK DIAGRAM

SPECIFICATIONS

Load an 8K basic program:...2.5 seconds
Load a 40K basic program:....5 seconds
Drive access time(TYP):40 milliseconds
Maximum number files:.....
.....143(540,580)127(877)
Maximum number drives:.....4
File types:.....Seq, Random, Prg.,Usr

Total Storage Capacity

bytes per diskette

540:.....143,000
580:.....286,000
877:.....250,000
Bytes per sector:.....128

Sectors per track

540,580:.....28
877:.....26

Tracks

540:.....40
580(double sided):.....80
877:.....77

Power Requirements

Voltage:.....115+/-10%
Frequency:.....50 or 60HZ

SOFTWARE

Quikdisk software resides both on the "boot ROM" and a system diskette. It is divided into several parts: primitives, boot, initialization, load, save, open, input, print, get and utilities. The software interfaces to the computer through the "Kernal jump table." The QUIKDOS initialization routine modifies the jump table such that floppy disk functions are routed to QUIKDOS. If the Quikdisk device number is not being called, the routines revert to standard serial bus functions. This method provides two key features: transparent operation to many existing programs and the ability to operate both serial bus disk drives and Quikdisk simultaneously on one computer.

Quikdisk software provides a standard repertoire of BASIC commands including LOAD, SAVE, OPEN, INPUT, PRINT, GET, and CLOSE. These commands provide standard BASIC operation and syntax.

All BASIC commands are accomplished using "Kernal Calls." Quikdos intercepts vectors of the OPEN (\$FFC0), CLOSE (\$FFC3), CHKIN (\$FFC6), CHKOUT (\$FFC9), GETIN (\$FFE4), CLALL (\$FFE7), CHRIN (\$FFCF), CHROUT (\$FFD2), LOAD (\$FFD5), and SAVE (\$FFD8) Kernal routines.

A special Quikdisk utility package is available to provide simple, easy to use disk maintenance functions. Tasks like formatting (NEW), eliminating old files (SCRATCH), copying, and other functions are done via an easy to use menu display.

The Quikdisk initialization is accomplished with the command SYS56843. The initialization sets the Quikdisk default device number to eight. Program loading, saving, is done exactly the same as the Commodore disk drive, e.g., LOAD "PROGRAM",8.

B. UNPACKING

The following instructions show you how to connect the Quikdisk System to the Commodore 64. In addition to this manual your Quikdisk package should include the following:

1. A standard disk drive assembly
2. A small disk controller module
3. A cable assembly
4. A QUIKDOS system diskette

If any of these items are not part of your Quikdisk package, please see your dealer immediately for a replacement.

C. INSTALLATION:

1. Be sure all power is off and the on/off switch on the back of the disk drive is set to off.

2. Connect the power cable from the back of the disk drive to the appropriate voltage electrical outlet. Once this is accomplished do not attempt to open the disk drive box without unplugging the power supply as electrical shock may result.

3. Take the flat cable coming out of the back of the disk drive and connect the female socket connector at the end of the cable into the post header sticking out of the module. The red stripe on the cable is pin one. Note the pin one marking on the module. Line up the yellow dots.

4. With the computer power switch off, plug the Quikdisk module into the expansion connector slot found in the right rear of the Commodore 64. Always be sure the power is off when plugging or unplugging the module.

5. You are now ready to turn on the power switches of the disk drive and the computer. You might consider using a surge protected power strip for a one button power source. If you are using separate plugs, be sure to turn on the various devices in the proper order. Be sure that the disk drive does not contain a diskette and turn it on. Turn on the printer and monitor. Finally, turn on the computer.

6. When the power is turned on the screen should appear exactly as it normally does. If you have your disk cable on backwards, the red light on the disk drive will light up and the motor will start whirring. If the cable is connected properly, the red disk light will not be on.

7. You are now ready to use the QUIKDOS diskette supplied with your Quikdisk system.

pin 1 - 33 all odd pins common
pin 2 - optional side select
pin 4 - optional index pulse
pin 6 - ready input
pin 8 - index
pin 10 - drive select one
pin 12 - drive select two
pin 14 - drive select three
pin 16 - motor control
pin 18 - direction
pin 20 - step
pin 22 - write data
pin 24 - write gate
pin 26 - track zero input
pin 28 - write protect input
pin 30 - read data input
pin 32 - optional side select
pin 34 - optional drive select four

DISK DRIVE INTERFACE

D. OPERATION

DISKETTE HANDLING

It is very important that you abide by the following rules for handling your floppy diskettes.

1. Never handle diskettes in such a way that you touch the open slots in the diskette or the center area.

2. Never use paper clips with any diskettes.

3. Do not place the diskette near or on top of any electrical item including audio speakers, the computer, the terminal or the printer.

4. Never expose the diskette to extreme heat or cold.

5. Do not allow the diskette to come in contact with any magnets or magnetic fields.

6. Always return the diskette promptly to its protective jacket. Dirt and dust can destroy the disk surface as well as damage the disk drive unit.

7. Never insert a diskette before turning on the system. Always remove all diskettes before turning the system off.

8. Mark the diskette only with a soft tip pen. Never use a pencil or ballpoint pen as you may "mark through" to the actual disk.

9. When relabeling diskettes, write the label before placing it on the diskette. Apply the label to the upper right corner of the diskette with its own, existing label facing you. On additional copies you may make such notes as "Backup 1" and/or "Backup 2."

START UP

The first thing you should do as a Quikdisk owner is NEW a blank disk and duplicate the master. After setup and installation, initialize the system, load the HELP program, format or NEW a blank diskette, duplicate the master. Then put the master away in a safe place.

USE THE FOLLOWING STEPS:

1. Turn on power to drive and computer.

2. Take your master help diskette and insert it in the disk drive. The label should always be toward the door or lever.

3. Close the door or lever on the disk drive.

4. Enter SYS56843 and press RETURN. The red drive light will go on and the drive motor will whirr.

5. Screen should read QUIKDISK, copyright Microtech. The system is now initialized.

6. Next display the directory by typing LOAD"\$",8. The directory of drive 0 should show HELP as the first program plus many other programs beginning with exclamation marks.

NOTE: A directory loaded from the standard disk drive into the Commodore must then be listed. An advantage of the Quikdisk is that the directory is automatically listed for you when the directory is loaded. These programs are directly displayed to the screen and a basic program, if in memory already, is not wiped out.

7. Your next step should be to make a duplicate of your master diskette. This will both test your system and help to familiarize you with some of the operations of the disk drive.

8. Type LOAD"*",8 to load the first program on the diskette. This is the Help program.

You will notice that whenever the Quikdisk loads a program, the screen momentarily goes blank. This is quite normal and is done to increase the overall loading speed. The disk is so fast that the display cannot keep up with it and therefore the screen must go blank.

9. Now RUN the program by typing RUN and pressing the RETURN key. DISK HELPER will appear on the screen.

10. The DISK HELPER listing will show you all the things you can do with the DISK HELPER. This special program was written to compliment Quikdisk and to replace all the disk maintenance commands you need to use with the Commodore 64. As you can see the format is straight forward and easy to operate.

FORMATTING (NEWing) A DISK

11. Next, to NEW or FORMAT a disk, just type N and follow the instructions on the screen. This replaces the more complicated commands necessary for the standard Commodore.

A. After you have typed N with the QUICKDISK master diskette in the drive, the screen will read "DISKETTE IN DRIVE #?" Type 0 for normal applications or 1 for dual disk drives.

B. The screen will instruct you to place the diskette you want to NEW into the disk drive. If you have a single drive, remove your master diskette and put the diskette you want to NEW in drive 0. If you have a multiple drive system, place the diskette you want to NEW in the drive indicated. You do not have to remove the master diskette from drive 0 if another drive is used for the NEW procedure. Be sure the diskette you want to NEW is a disk which contains nothing you want to keep as the NEW command wipes out anything previously on the disk.

C. You will next be asked to enter the name you wish to assign to the diskette. No I.D. numbers are needed. Press RETURN key when finished with the name.

D. Answer the "Sure?" question with Y to continue or any other key to take you back to the main menu.

E. When the drive stops, the disk is NEWed.

NOTE: Any time you implement a disk Helper command your master disk helper or a copy of your master must be in drive 0. The command program is actually being loaded into memory from the disk to implement each of the functions. These programs are the ones displayed in the directory along with HELP. By keeping these programs on disk and automatically loading them when you need them, you are able to save valuable memory space. If you are using a dual disk drive, the master disk can be kept in drive 0 at all times.

After you have chosen Y in the above step, you will hear the drive clicking through the tracks formatting the diskette. The Quikdisk formats up to 80 tracks while the Commodore has 35 so the total time to format may be a bit longer. All of the base information is now being recorded on the diskette along with a blank directory and a copy of the QUIKDOS boot software. Quikdisk will now tell you if it has been successful, if not a disk error code will be displayed which you should look up in the error listings. Hit any key to continue if an error has occurred.

INSTRUCTIONS FOR DUPLICATING THE MASTER

12. Your next step should be to make a duplicate of your master diskette.

A. Remove the diskette that has been NEWED and place your master diskette in drive 0. The DISK HELPER listing is on the screen. This special program was written to compliment Quikdisk and to replace all the disk maintenance commands you need to use with the Commodore 64. As you can see the format is straightforward and easy to operate.

B. Press D. The duplicate command allows a quick and convenient way to totally duplicate a Quikdisk diskette. If you have a single disk drive system, use the following instructions. If you have a dual disk drive system, skip to the second set of instructions.

C. The screen will read "FROM DRIVE #". Enter your drive # 0.

D. The screen will next read "TO DRIVE #". Enter drive # 0.

E. Leave your master diskette in the drive and type C key to continue.

F. When prompted, remove the master diskette from the drive and place the diskette to be copied onto (the diskette NEWed in step 11) in the drive. Press C.

G. Repeat steps 8 and 9 as required when prompted. When the copy is finished, the screen will read: "Remove Disk" Hit any key to continue.

11. Press any key to continue.

12. The screen will return to the disk helper menu. Keep your newly duplicated master diskette in drive 0. Store your original master in a safe place so that it cannot be damaged.

NOTE: The RUN/STOP key will return you to the Disk Helper menu from almost any place in the helper program.

DUPLICATING A DISK ON A DUAL DRIVE SYSTEM

C. The screen will read "FROM DRIVE #". Enter your drive # 0, 1, or 2. (Enter 0 for normal usage.)

D. The screen will next read "TO DRIVE #". Enter drive # 1.

E. Leave your master diskette in drive 0 and type C key to continue.

F. Put diskette to be copied onto (the diskette NEWED in step 11) into the "copy to" drive #. When ready, press C.

G. The system will automatically duplicate the original diskette onto the copy diskette without further operator action.

H. When finished, remove original master diskette from drive 0 and store in a safe place. Put the newly duplicated master Help diskette into drive 0. Press any key to continue.

PROCEDURE FOR TRANSFERRING A PROGRAM FROM 1541 to QUIKDISK

Programs can be transferred back and forth between the regular 1541 disk drive and the Quikdisk. This is normally done by simply loading from one disk unit and saving to the other. Of course the two drive units cannot have the same device number. First change the Quikdisk device # (because it is very easy to change) and proceed with the loading and saving.

Both the Commodore 1541 and Quikdisk use the 16 character file name size. Both allow the same characters.

TRANSFERRING A PROGRAM FROM 1541

1. From BASIC (not using DISK HELPER), change Quikdisk to device 9 by typing POKE 53226,9 and hitting return.

2. Choose the program to be transferred and load it from the Commodore 1541 drive by typing LOAD"program name",8 and hitting return.

3. Save the program to Quikdisk by typing SAVE"program name",9 and hitting return

4. The transfer is complete.

5. You can display a directory of all programs on QUICKDISK by typing LOAD"\$",9 and hitting return. Verify that your program is listed.

6. For fun, type NEW (return). This erases the program from the computer memory.

7. To reload the program from QUICKDISK, type LOAD"program name",9 and hit return. See how much faster QUICKDISK loads.

NOTE: Some copy protection schemes will prevent transfer of the program. Contact the software manufacturer for information or a Quikdisk version of the software.

2. USER'S GUIDE

The QUIKDISK system must be initialized once everytime the computer is first powered on. Use the following steps:

1. Turn on power to drive and computer.

2. Take your backup master diskette and insert it in the disk drive. The label should always be toward the door or lever.

3. Close the door or lever on the disk drive.

4. Enter SYS56843 and press RETURN. The red drive light will go on and the drive motor will whirr.

5. Screen should read QUIKDISK, copyright Microtech. The system is now initialized and ready for use with BASIC.

A. DISK COMMANDS

Introduction: One of the main uses of a floppy disk is to load and save programs. Programs are saved and loaded from the disk in much the same way as they are loaded and saved from cassette. The only difference is that the device # is added at the end of the command. It is likely that the first two disk commands learned will be LOAD and SAVE. Some users need no others.

As the user gets into more advanced applications, it becomes necessary to be able to load and save more than just programs. A computer must also be able to load and save "data files." A data file is simply a collection of information like the names and addresses in a mailing list. Quikdisk provides several methods for storing data files on the disk. One type of data file that can be used is the sequential file.

Sequential files are those files that are stored and read as their name implies, sequentially from beginning to end. There are basically three different types of sequential files that can be used on the Quikdisk. The first type of sequential file is the program file. Programs from memory are actually stored sequentially on disk--beginning to end. They are abbreviated in the directory of the diskette as PRG. The program file is the only sequential file type that can be loaded. BASIC uses sequential program files when storing or reading programs.

The second and third types of sequential files are designated as sequential (SEQ) and user (USR) respectively. These two file types are for data handling. A data file can be opened for reading or writing, and three different commands can be used to actually handle the data transfer. These three commands are the PRINT#, INPUT# and GET#. The PRINT# command is used for writing to the diskette while the INPUT# and GET# are used for reading from the diskette.

In summary, the disk commands include

- LOAD-for program loading,
- SAVE-for program storing,
- OPEN-to open a data file,
- PRINT#-for writing data,
- INPUT#-for reading data,
- GET#-for reading data characters,
- CLOSE-to shut a data file.

LOAD

The purpose of the LOAD command is to get a program from the diskette into the computer memory. The LOAD must be carefully entered in order to operate correctly. In this case, the command to LOAD will transfer a program on diskette to the computer memory. The program on diskette is not altered; it's still there and can be loaded over and over.

The general syntax is:

```
LOAD"DR:program name",DD
```

where DR is the drive number,
and DD is the device number.

The program name can be a string variable or a file name that has been enclosed in quotes.

FILE NAME LOADING

To load the program "STOCKS" from disk drive 1, type LOAD "1:STOCKS",8. In Quikdisk systems, 0, 1, and 2 are valid drive numbers if your systems has multiple drives. In single drive systems, the only valid drive number is 0. If no drive number is specified, a 0 (zero) is assumed.

```
LOAD"PAYABLES",8 and  
LOAD"0:PAYABLES",8 do the exact  
same thing.
```

Loading a program always deletes any pre-existing programs in the computer.

WILD CARD CHARACTERS

The character ? can be used as a wild card or "don't care" character. This can be especially helpful when the exact name can change. If wild card characters would allow more than 1 program on disk to be loaded, the earliest in the directory is selected. For example, if you have three programs on the diskette:

"PAYABLE V1"

"PAYABLE V2" and

"PAYABLE V3" the statement

LOAD "PAYABLE V?",8 will load the first one of these three encountered. In this case, "PAYABLE V1."

STRING VARIABLE LOADING

The file name in the LOAD command can be a string variable. The string variable must have been defined in an earlier statement. LOAD B\$,8 is an example of a load command with a string variable for the program name. B\$ has been previously defined as a file that is in the directory.

For example:

B\$ = "PAYABLE V?"

LOAD B\$,8 will load the program "PAYABLE V1" from the above directory.

A string variable can also be used to specify the drive. (Remember that drive zero is assumed if no drive number is given.) For example:

```
B$="1:PAYABLE V1"
```

LOAD B\$,8 will load the program "PAYABLE V1" from drive 1.

In BASIC, string variables can be added. This could be used to allow separate variables to be used for the drive number and the program name. For example:

```
DR$="0"  
NM$="PAYABLE V1"  
B$=DR$+"":NM$
```

LOAD B\$,8 will load the program named in NM\$ from the drive specified by DR\$. In this example B\$= 0:PAYABLE V1

SAVE

SAVE "program name", 8 is the command to put a copy of the program in the computer memory onto the diskette that is in the disk drive. The syntax for the SAVE command is very similar to the LOAD command.

The general syntax is:

```
SAVE"DR:program name",DD
```

where DR is the drive number,
and DD is the device number.

The program name can be a string variable or an actual file name inside of quotation marks. The device number may be a variable or a fixed value and should be 8 unless the device number has been changed. The program will be saved from the start of the BASIC memory area to the end of the program unless a file is already on the disk that has the same file name or there is not sufficient space on the disk. If either of the last two conditions occur, an error message will be generated. Wild card characters should not be used in a SAVE command.

It is possible to save a program to disk even if that program already exists on the diskette. This is commonly required when a program is loaded, modified and then must be resaved to diskette. The user must first scratch (or kill) the old file and then save the modified program under the old name.

DATA FILE HANDLING

A floppy disk drive is typically used to store data as well as programs. These "data files" on disk are typically such things as lists of names and addresses in a mailing list program, transaction lists in an accounting program or lists of materials on hand in an inventory program.

A data file is like a notebook. To use it, it must be opened. Once open, data can be written. When finished writing, the notebook is closed. To read, open it again. It can be read over and over. It should be closed when reading is finished.

A series of BASIC commands allow the computer to manipulate data files. These are OPEN, PRINT#, GET#, INPUT# and CLOSE.

OPEN

Data files are sequential files. The sequential files must be opened in order to read or write to the file. The syntax for opening a sequential file is:

```
OPEN      "file#",      "device  #",      "data
channel", "DR:file,type,direction"
```

For example:

```
OPEN2,8,2,"O:INVENTORY,S,W"
```

```
FILE#-----2
DEVICE#-----8
DATA CHANNEL#-----2
DR# drive number----0
FILE NAME-----INVENTORY
TYPE-----SEQ
DIRECTION-----WRITE
```

```
FILE#  OPEN2,8,2,"O:INVENTORY,S,W"
```

The file number is a user assigned number between 1 and 255 or a predefined variable (usually the same number as the data channel for ease of remembering). This number is a reference number used to read or write.

```
DEVICE#  OPEN2,8,2,"O:INVENTORY,S,W"
```

The device number is usually 8 or a predefined variable.

```
DATA CHANNEL OPEN2,8,2,"O:INVENTORY,S,W"
```

The data channel is a number between 2 and 14 that the user assigns as the channel that will be used for communicating.

DR OPEN2,8,2,"O:INVENTORY,S,W"

-

DR represents the drive number which is 0 for single drives and 0, 1 or 2 for multiple drives. If the number is omitted, zero is assumed.

FILE OPEN2,8,2,"O:INVENTORY,S,W"

The file is the file name on the diskette to be opened. If the file does not exist, it will be created in the case of a write or create an error condition in the case of a read.

TYPE OPEN2,8,2,"O:INVENTORY,S,W"

-

The type represents the file type to be opened (PRG, SEQ, USR). The file type is abbreviated to its first character in the OPEN statement.

DIRECTION OPEN2,8,2,"O:INVENTORY,S,W"

-

The direction is either read or write which is designated as R or W respectively in the OPEN command. When a file has been opened, the LED relating to the particular drive will be red to indicate a file is open. The LED will remain red until all files on that drive are closed.

LENGTH OF A DATA FILE

When QUIKDOS opens a file it reserves storage space for that file on the diskette. The data file cannot grow larger than the space reserved. Therefore, it is important that the user (or the program using QUIKDOS) give consideration to the size of the file when it is opened.

QUIKDOS normally reserves a space of 200 sectors (25,600 characters) unless told otherwise. Means are provided to change this number via a special BASIC variable, LN%. To reserve a specific amount of space, simply set LN% to the number of sectors desired. Remember-each sector can hold up to 128 characters.

For example: to open a file with a reserved space for 2000 characters.

```
LN% = 2000/128  
OPEN2,8,2,"ADDRESS,S,W"
```

It may be a good idea to reserve more space than necessary just to handle contingencies. Unused space is reallocated whenever the diskette is "validated." (Reference the V command in the disk helper section.)

Any existing program that has been opened for write can replace the existing program of the same name if the program name is preceded by the "@" symbol. For instance: OPEN 8,8,8,"@0:SECOND,S,W" will replace the existing file called SECOND with the new data to be written. Be sure to use the drive number when using the "@" command.

INPUT#

After a data file is opened, records of data may be read into the computer with the INPUT command. For example:

INPUT#8,A\$ will read the next record from file 8 and put it into string variable A\$.

The first time the INPUT command is used, it will read the first record from the file and so on. The INPUT# command is used to retrieve data from the disk in groups rather than a character at a time. The file # is the file number that was used in the OPEN command. For example:

If the open is

```
OPEN2,8,2,"O:INVENTORY,S,R"
```

the input is

```
INPUT#2,X$.
```

The variable is a single variable or multiple variables separated by commas. The variables can be strings or numeric. For example:

```
INPUT#2,A$,B$,C$
```

will read one record into each of the variables A\$, B\$, and C\$.

In order to read a group of data from the diskette, it is necessary to be able to recognize the start and end of that group of data. This is accomplished through the use of separators as explained in the PRINT# section. The INPUT# command recognizes a comma, carriage return, and a colon as a separator between data fields. If a variable is to be read in from diskette, these separators will determine exactly how much data will come in each time. If the data is numeric, it is stored just as if the STR\$ function had been performed on the numeric data. Numbers are stored with positive numbers containing a space as the first character while negative numbers have a minus sign as the first character.

The PRINT# command works like the PRINT command in BASIC. Any commas that are used to separate items on a line will result in spaces being stored to the diskette. Semicolons will keep spaces from getting stored to the diskette when used to separate items on a line. The absence of either a comma or semicolon will cause a carriage return (CR) to be stored on the diskette at the end of a PRINT# command. Any data that is sent to the diskette using the PRINT# command will be stored sequentially byte by byte with all spaces, carriage returns and any other special character stored. The comma, carriage return and colon have special significance when stored to the disk. When used inside a string, they will be stored to the disk. When used as a separator between fields, the comma inserts spaces, while the carriage return and colon end the field and store a carriage return to the disk.

For example, the statement:
PRINT#8,"NO, THANK YOU" will put the comma inside the quotes onto disk. The statement: PRINT#8, "NO","THANK YOU" will cause spaces to be inserted between NO and THANK YOU instead of a comma. The importance of using the comma, carriage return and colon on the diskette becomes apparent when using the INPUT# or GET# command to retrieve the data from the disk.

PRINT#

The PRINT command is used to write records to a data file. It is similar in format to the INPUT command. For example: PRINT#8,A\$ will write the variable A\$ to next record in file 8. The PRINT# (no spaces are allowed between the PRINT and the # sign) command, is used to direct any output to the file that has been opened. The file number is that file that has been opened by the OPEN command. For example:

If the open is

```
OPEN2,8,2,"O:INVENTORY,S,W"
```

the print is

```
PRINT#2,"QUANTITY OF PC BOARDS"
```

The data may consist of variables and/or text that has been enclosed inside quotation marks.

GET#

This command can be used to get a single data byte (character) or a series of data bytes (characters). It will get data from the disk a byte at a time. From time to time, it will be desirable to get data from the disk a byte (or character) at a time. The GET# command is used for that purpose. This command is useful because all data that is on the disk is read, including the comma, carriage return and the colon. The file # is the file number that has been opened while the variable is a single variable or a list of variables separated by commas. The variables can be string or numerical variables but string variables are recommended. If string data is encountered on the diskette when a number was requested, a BASIC error statement is generated. Since a string variable can be a number or character, it can always be read. Some examples of the GET# statement are:

GET#8, A\$ and

GET#8, A\$,B\$,C\$

The GET# statement is especially useful when the actual data content or structure on the disk is not known. Since every character is received, disks that have had portions destroyed can be examined to possibly recover data that would not normally be recovered. For situations where a group of data is to be read and the data structure is known, the INPUT# statement is better suited.

CLOSE

The purpose of this command is to close the channel of communications that has formerly been opened. The file number is that file number that was used in the OPEN statement. Once a file that has been opened is no longer needed for data transfer, it must be properly closed. Closing the file causes the directory to be correctly updated. If the file is not properly closed, all data associated with that file could be lost.

PROGRAMMING THE DISK CONTROLLER

Changing device number

QUIKDOS intercepts Kernal disk commands by comparing the desired device # with a number stored in memory. Changing QUIKDISK device number is simply a matter of storing a different number to memory. In BASIC, this is done by POKE 53226,X (return) where X is the new device number.

For example:

```
POKE53226,9
```

changes the device number of QUIKDISK to nine. After this operation, all disk commands will end with ",9".

DISK KERNAL ROUTINES

When writing machine language programs, it may be convenient to use some of the special routines provided by QUIKDOS. Access is provided via jump table so that machine language programs do not have to change with various versions of QUIKDOS. The routines available are:

```
READ SECTORS.....$DE04
WRITE SECTORS.....$C207
DIRECTORY SEARCH.....$C20D
KILL A FILE.....$C210
```

To use the routines you must first do all the set up required. The routine is then called. When status parameters are checked for errors, the program can continue. Do not ignore error status messages. The remainder of your program can fail if an error occurs.

SYSTEM VARIABLES

```
PDEVIC .DE $CFA1
PTRACK .DE $CFA2
PSECTR .DE $CFA3
SCTCNT .DE $CFA6
EAL     .DE $AE
STATUS  .DE $90
BAD     .DE $CFF4
```

READ SECTORS ROUTINE

Purpose: Get data from disk sectors

Call Address: \$DE04

Input parameters:

PDEVIC-set drive #

PTRACK-set track#

PSECTR-set sector#

SCTCNT-set # of sectors

EAL -set load address

Error returns: Carry set if error
STATUS has error code
BAD has error code

Registers affected: .A, .X, .Y

Description: This routines reads data from the specified drive/track/sector into memory at the specified address. The specified quantities of sectors are read.

Example: Read 4096 bytes of data from drive 0, track 00, sector 21 into memory at \$9000.

```
PDEVIC .DE $CFA1
PTRACK .DE $CFA2
PSECTR .DE $CFA3
SCTCNT .DE $CFA6
EAL .DE $AE
STATUS .DE $90
BAD .DE $CFF4
RDSECS .DE $DE04
```

```
LDA #00
STA PDEVIC ; DRIVE NUMBER 0
LDA #00
STA PTRACK ; TRACK 0
LDA #21
STA PSECTR ; SECTOR 21
LDA #32 ; 4096/128=32
STA SCTCNT ; 32 SECTORS
LDA #00
```

```
STA SCTCNT+1 ; SECTOR COUNT HI
LDA #00
STA EAL
LDA #$90
STA EAL+1 ; LOAD ADDR $9000.;
JSR RDSECS
BCC OKLOOP
JMP ERROR1.

OK LOOP ; CONTINUE PROGRAM
```

WRITE SECTORS ROUTINE

Purpose: Record data to disk sectors

Call address: \$C207

Input parameters:

PDEVIC- set drive #

PTRACK- set track #

PSECTR- set sector #

SCTCNT- set # of sectors

EAL - write from address

Error returns: Carry set if error
STATUS has error code
BAD has error code

Registers affected: .A, .X, .Y

Description: This routine selects the proper drive, finds the correct track and sector and then records data from memory to the disk. The specified quantity of sectors will be recorded. Each sector holds 128 bytes of data.

Example: Write 2048 bytes of data to drive 1, starting at track 21, sector 1. The data is in memory at \$0801.

```

WRSECS .DE $C207
LDA #01
STA PDEVIC      ; DRIVE 1
LDA #$15        ; $15 = 21
STA PTRACK      ; TRACK = 21
LDA #01
STA PSECTOR     ; SECTOR = 1
LDA #$10        ; 2048/128 = 16 = $10
STA SCTCNT
LDA #00
STA SCTCNT + 1; COUNT = $0010;
LDA #$01
STA EAL
LDA #$08
STA EAL+1;
JSR WRSECS
BCC OK
JMP ERROR
OK ; CONTINUE WITH PROGRAM

```



```
"NEW0:FILENAME,AA"
would divide as:
PDEVIC = 0
FNTEMP = "FILENAME"
FCOMM  = N
FTTS   = AA
FSTAT  = R (default value)
```

```
The string: "1:MAILLIST,SEQ,W"
would decode as:
PDEVIC = 1
FNTEMP = "MAILLIST"
FTTS   = SE
FSTAT  = W
FCOMM  = 00 (default value)
```

Two errors can occur during string decodings,

```
BAD = 30 SYNTAX ERROR
BAD = 34 NAME MISSING
```

After the input string is decoded, the first directory block is loaded from the diskette. Names are compared with FNTEMP. If a match is found, a variable CURFILE is set and the routine is finished. If not, more directory blocks are loaded and compared till the end of the directory at track 2.

Example: Determine if a program "SORT" is on the disk.

```
DIRSCH .DE $C20D
FNADR  .DE $BB
FNLEN  .DE $B7
```

DIRECTORY SEARCH

Purpose: to search the directory of the specified disk drive to determine if the specified file is present. If present, provide details as to where the file starts, how long, etc.

Call address: \$C20D

Input parameters:

FNADR - a pointer to a text string
FNLEN - length of the text string.

Error Returns:

IF DIR FOUND - .A=\$00, carry clear
IF NO DIR FOUND - .A=\$7F, carry set
IF ERRORS - .A=\$FF, carry set

Registers affected= .A, .X, .Y

Description: The Directory Search routine does all items necessary to determine if a file described by the input text string exists on the disk. It starts by evaluating the input text string. The string is divided into sections. For example:

INPUT TEXT STRING "O:FILENAME"
into two parts,
PDEVIC = 0
FNTEMP = "FILENAME"

The string can be divided into many parts including:

PDEVIC - disk drive #
(0 if not specified)
FNTEMP - a 16 byte block holding name
FCOMM - command byte
FTTS - 2 character type code
FSTAT - status, normally R or W

A complicated string like:

```

      N1      .BY "SORT"
;
      LDA #L, N1
      STA FNADR
      LDA #H, N1
      STA FNADR ; PUT ADDR OF N1 IN FNADR
      LDA #4
      STA FNLEN ; STRING LENGTH IS 4
;
      JSR DIRSCH ; search directory.
      BCC FOUND
      JMP NOT*FOUND
FOUND ; continue with

```

KILL A FILE ROUTINE

Purpose: to remove (scratch) a file from a diskette.

Call address: \$C210

Input parameters:

FNADR - a pointer to a text string

FNLEN - length of the text string.

Error returns: carry set if error

Registers affected: .A, .X, .Y

Description: The KILL routine decodes the input string, calls DIRSCH to search the directory, and if the file found, it is marked with a scratch mark.

Example: Scratch the file "SORT".

```
KILL    .DE $C210
FNADR    .DE $BB
FNLEN    .DE $B7
N1        .BY "SORT"
;
LDA #L, N1
STA FNADR
LDA #H, N1
STA FNADR+1 ; FNADR POINTS TO N1
LDA #4
STA FNLEN
;
JSR KILL ; kill the file
BCC OK
JMP ERROR
OK ; continue with program
```

DIRECTORY FORMAT

The QUIKDISK directory is stored on disk starting at TRACK 00 SECTOR 21. For models 540 and 580, 36 sectors are reserved for directory. For Model 877, 32 sectors are reserved. The first directory entry is always the disk name.

Remaining space in the directory area is occupied by file names. Each entry is 32 bytes in length.

DISK ORGANIZATION

Several sections of each floppy disk are reserved for special functions including

1. The QUIKDOS program and
2. The DISK DIRECTORY.

A floppy disk used in QUIKDISK has 28 or 26 sectors per track. The number of tracks is 40, 80, or 77. (See SPECIFICATIONS.)

For 540 and 580 QUIKDISK:

QUIKDOS PROGRAM----TRACK 00 SECTOR
01 through TRACK 00 SECTOR 20

DIRECTORY-----TRACK 00 SECTOR
21 through TRACK 01 SECTOR 28

GENERAL STORAGE----TRACK 02 SECTOR
01 up

FOR 877 QUIKDISK:

QUIKDOS PROGRAM----TRACK 00 SECTOR
01 through TRACK 00 SECTOR 20

DIRECTORY-----TRACK 00 SECTOR
21 through TRACK 01 SECTOR 28

GENERAL STORAGE----TRACK 02 SECTOR
01 up

FORMAT OF DIRECTORY ENTRY

NAME-----BYTE 00 through 15
(\$OF)

DELETE STATUS-----BYTE 16 (\$10) 00
indicates active

END OF DIRECTOY MARK-BYTE 17 (\$11) 00
indicates NOT THE END

START TRACK-----BYTE 18 (\$12) This
is track where file starts

START SECTOR-----BYTE 19 (\$13) This
is sector where file starts

FILE TYPE-----BYTE 20 (\$14)
00=PRG 01=USR
02=REL 03=SEQ

RESERVED-----BYTE 21 (\$15)

SECTOR COUNT LO-----BYTE 22 (\$16)

 Sector count is the total quantity
of sectors used in each file. This is
lower 8 bits.

SECTOR COUNT HI-----BYTE 23 (\$17)

 Upper 8 bits of SECTOR COUNT

LOAD ADDR LO-----BYTE 24 (\$18)

LOAD ADDR HI-----BYTE 25 (\$19)

 Used for program files, this 16 bit
number indicates where the program would
normally be loaded in memory.

EXECUTE ADDR HI-----BYTE 26 (\$1A)

EXECUTE ADDR LO-----BYTE 27 (\$1B)

 Used for program files, this 16 bit
number indicates the actual beginning of
the program. Program execution begins
here. It is normally the same as the
load address but not always.

FILE CONTROL BLOCK (FCB) FORMAT

For each data file that is open in a QUIKDISK system, a "file control block" is maintained. The File Control Block (FCB) is a collection of data necessary to maintain the file. This data includes the file name, the logical address #, disk variables, and information about where characters are being stored.

NAME.....BYTE 00 through BYTE15
RESERVED VARIABLE.....BYTE 16

This is used for different functions based on file type.

CHARACTER COUNTER.....BYTE 17

This points to the position in the file buffer where the next character is stored or retrieved.

START TRACK.....BYTE 18

The file starts on this track.

START SECTOR.....BYTE 19

The file starts on this sector.

FILE TYPE.....BYTE 20

00 = PRG 01 = USR

02 = REL 03 = SEQ

ACTIVE LA#.....BYTE 21

Logical address # as opened.

SECTOR COUNT LO (USED).....BYTE 22

SECTOR COUNT HI (USED).....BYTE 23

A 16 bit number representing the total sectors used when file is written.

SECTOR COUNT LO (RESERVED).....BYTE 24

SECTOR COUNT HI (RESERVED).....BYTE 25

A 16 bit number representing the total number of sectors reserved when file is opened for writing.

BUFFER POINTER LO.....BYTE 26

BUFFER POINTER HI.....BYTE 27

A 16 bit number that points to the beginning of an 128 byte buffer used to store characters until they are recorded on disk

ACTIVE TRACK.....BYTE 28

When the character buffer is full, it will be recorded to this track during a write. When the character buffer is empty, it will be filled from this track during a read.

ACTIVE SECTOR.....BYTE 29

Used with Active Track to identify a particular point on the disk.

ACTIVE DRIVE #.....BYTE 30

Used with Active Track and Sector to identify the next sector on the disk.

STATUS.....BYTE 31

Identifies a read or write file.

FCB MEMORY MAP

QUIKDOS allows four data files to be simultaneously open. It does this by maintaining four FCBs.

FCB1 - \$CF00

FCB2 - \$CF40

FCB3 - \$CF60

FCB4 - \$CF80

The files also require character buffers.

F1BUF - \$CDOO

F2BUF - \$CD80

F3BUF - \$CE00

F4BUF - \$CE80

ERROR CODES

20 - READ ERROR ON HEADER (DIR)

Quikdisk was not able to read the directory of the diskette. Although the program or data files may be OK on the disk, QUIKDISK cannot locate the data because the location information is in the directory. Unless you have previously recorded the directory, the program or data file on this diskette is useless.

21 - READ ERROR BOOT BLOCK IN FORMAT PROGRAM

22 - SEEK ERROR AFTER 3 RETRIES

The controller board has tried three times to find a particular track on the diskette. This error is normally caused by a bad diskette.

26 - WRITE PROTECTED

The controller has been asked to record information on a disk drive with the write protect status set. The write protect status is normally set when the write protect tab is inserted on the diskette. Remove it if you wish to record data.

28 - WRITE ERROR, FORMAT-BOOT BLOCK ERROR

29 - DISK HEADER ERROR/HARD ERROR

30 - SYNTAX ERROR IN NAME

31 - NOT AN INPUT FILE.

32 - DRIVE NUMBER INVALID, SYNTAX

Only number 0, 1 and 2 are valid

33 - BUSY TIMEOUT ERROR BEGINNING OF
COMMAND

Disk Controller has not responded in time. This can be caused by an improperly seated module, a bad module, or a failure in the computer. Power down your system and make sure the module is plugged in all the way. If the code persists, try your controller/disk system on another computer.

34 - NAME MISSING ON READ

61 - FILE NOT OPEN

An attempt was made to INPUT#, GET#, or PRINT# to a file that was not opened. You must open a file to gain access to the data.

62 - NAME NOT ON DISK, READ

63 - TRY TO SAVE/ DUPLICATE NAME

A SAVE attempt was made using a file name identical to one that already exists on the diskette. You may use the "@" command to automatically scratch the old file or use a different name.

64 - FILE TYPE MISMATCH

The file type in the directory does not match the file type requested.

71 - DIRECTORY WRITE/ HARD ERROR

72 - NOT ENOUGH ROOM ON DISK/ END!

This can be caused by lack of room either on the main diskette storage area (not enough sectors free) or by a full directory. Only 151 directory entries are permitted.

74 - DRIVE NOT READY AT DRVSET

75 - DRIVE NOT RESTORED IN FORMAT

82 - FILE READ ERROR

The compress routine can't read a particular file. This usually means you have a bad diskette.

83 - FILE WRITE ERROR

The compress routine can't write a particular file. This is usually caused by a bad diskette.

84 - INDEX WRITE ERROR

The disk could not rewrite a new index after rearranging all data on disk via the compression routines. Unfortunately this means that your entire disk is not usable by the computer and all your data is lost. Be sure to duplicate your backup diskette before attempting to compress it.

SUBROUTINES INTERCEPTED BY QUIKDOS

OPEN	\$FFC0
CLOSE	\$FFC3
CHKIN	\$FFC6
CHKOUT	\$FFC9
CHRIN	\$FFCF
CHROUT	\$FFD2
LOAD	\$FFD5
SAVE	\$FFD8
GETIN	\$FFE4
CLALL	\$FFE7

Troubleshooting

More than half of the problems in a floppy disk system are caused by bad diskettes. Remember, diskettes do wear out. You must occasionally back up (duplicate) your diskettes and discard the older ones that are constantly causing errors.

If you are having trouble with a number of different diskettes the first thing to do is power down all of your equipment and start over again. If possible start with an unused backup of the Master System diskette. If several restart attempts fail, contact your dealer.

MEMORY MAPS

\$0000-\$BFFF	Normal usage by 64
\$C000-\$C1FF	Quikdos Utilities
\$C200-\$CFFF	Quikdos, data buffers
\$D000-\$DE03	Unused by Quikdisk
\$DE04-\$DFDF	Boot ROM
\$DFF4-	Control latch
\$DFF0-\$DFF3	LSI Controller
\$DFF8-\$FFFF	Unused by Quikdisk

3. DISK HELPER

LOADING THE SYSTEM DISKETTE SUPPLIED WITH YOUR QUIKDISK SYSTEM

As previously mentioned, the first thing you should do as a Quikdisk owner is to NEW a blank disk and duplicate the master. After setup and installation, initialize the system, load the HELP program, format or NEW a blank diskette, and duplicate the master. Then put the master away in a safe place.

1. Turn on the power to the drive and the computer. If you are using a power strip, this is a single switch.

2. Take your master Help diskette and insert it in the disk drive. The label should always be toward the door or lever.

3. Close the door or lever on the disk drive.

4. Enter SYS56843 and press RETURN. The red drive light will go on and the drive motor will whirr.

5. The screen should read QUIKDISK, copyright Microtech. The system is now initialized.

6. Next display the directory by typing LOAD"\$",8. The directory of drive 0 should show HELP as the first program plus many other programs beginning with exclamation marks.

7. Now load the first program on the diskette by typing LOAD"*",8. This is the HELP program.

8. Next type RUN and press the RETURN key to run the program. DISK HELPER will appear on the screen.

9. The DISK HELPER listing will show you all the things you can do with the DISK HELPER. This special program was written to compliment Quikdisk and to replace all the disk maintenance commands you need to use with the Commodore 64. As you can see the format is straight forward and easy to operate.

NOTE: The RUN/STOP key will return you to the Disk Helper menu from almost any place in the helper program.

B. COMMANDS

B-BLOCK DISPLAY

The block display command allows you to directly examine hexadecimal data anywhere on your diskette. Data is addressed by drive, track and sector.

1. With the Help Diskette in the drive, push B.

2. The screen will read "Enter Drive #". Enter the # of the drive (0 for single drive systems).

3. The screen will read "Track?". Enter any number from 00-79 for 580 series drives, 00-76 for 877 series drives.

NOTE: The RUN/STOP key will return you to the Disk Helper menu from almost any place in the helper program.

DUPLICATING A DISK ON A DUAL DRIVE SYSTEM

1. With the HELP diskette in drive 0, push D.

2. The screen will read "Copy from drive #". Enter your drive # 0, 1, or 2. (Enter 0 for normal usage.)

3. The screen will next read "Copy to drive #". Enter drive # 1 or 2.

4. Put original diskette in drive 0 and type C key to continue.

5. Put diskette to be copied onto into the "copy to" drive #. When ready, press C.

6. The system will automatically duplicate the original diskette onto the copy diskette without further operator action.

7. When finished, remove diskette and return the Help diskette to drive 0. Press any key to continue.

4. Next you will be asked for the sector number. Enter any number from 01-28. (For the 877 series of Quikdisk, the range of sector numbers is 01-26.)

A block will then be displayed on the screen. One block equals two sectors. A sector is 128 bytes. The Commodore block is 256 bytes.

D-DUPLICATE A DISK

The duplicate command allows a quick and convenient way to totally duplicate a Quikdisk diskette. If you have a single disk drive system, use the following instructions. If you have a dual disk drive system, skip to the second set of instructions.

1. The screen will read "Copy from Drive #". Enter your drive # 0.

2. The screen will next read "Copy to drive #". Enter drive # 0.

3. Put original diskette (the diskette that you wish to duplicate) in the "copy from" drive and type C key to continue.

4. When prompted, remove the original diskette from the drive and place the diskette to be copied onto in the drive. Press C.

5. Repeat steps 3 and 4 as required when prompted.

6. Remove the diskette and press any key to continue.

7. The screen will return to the disk helper menu.

G-GO TO PROGRAM

This is a special command that is not marked. This is a command that allows the computer to execute a program from any point in the memory. After the command G the screen will say "Enter Address." Enter a four digit hexadecimal number. The computer will jump to that memory location and begin executing.

H-HELP

This is a list of abbreviated instructions.

K-KILL A FILE. SCRATCH a file.

1. Press K. The screen will read "Enter name".

2. Place the diskette containing the file you want to Kill or Scratch in a drive and enter the name of the file you want to eliminate. If you wish to scratch a file in a drive other than 0, enter the drive # and a colon before the name, for example, 1:SPACEMAN will Scratch the program SPACEMAN from drive 1.

NOTE: Remember the wild card character "?" and the asterisk pattern mask character. Make sure you are scratching the correct file.

3. The screen will return to the Disk Helper and pressing \$ key will show the directory with * indicating the deleted file. (Refer to \$ instructions.)

L-LOAD DISK PROGRAM

1. Press L. The screen will read "Enter Name:". Enter the name of the program to be loaded. If you wish to load a program from a drive other than 0, enter the drive # and a colon before the program name. For example 1:PROJECT MANAGER will load the program "PROJECT MANAGER" from drive #1.

2. The screen will return to the Disk Helper menu when the program is loaded.

NOTE: Remember the wild card character "?" and the asterisk pattern mask character. You can use these to make the loading easier.

M-MONITOR

1. This is a machine language monitor public domain program. Extreme caution should be exercised with this program as severe damage to the diskette may result from misuse. This program is included for troubleshooting and testing purposes only and should not be used unless the operator is completely familiar with its use.

N-NEW OR FORMAT A DISK

Before a new diskette can be used, it must be NEWED (or formatted). This process consists of recording timing marks, identification marks, and error checking data on the disk. In addition, the QUIKDISK NEW program records a blank directory, the disk name and a copy of the system initialization software on each diskette.

R-RENAME A DISK FILE

The rename command can be used to replace the name of any program or data file on a diskette.

1. The screen will read "Old file name is --. Enter name:" If you wish to rename a file in a drive other than 0, enter the drive # and a colon before the name, for example, 1:LEDGER will rename the program "LEDGER" from drive 1.

2. Place disk with the file on it in the drive and enter the name of the old file.

3. Then enter the name of the new file. As with the other commands, if you are using a dual or triple disk drive, use the drive # and a colon before the program name.

S-SAVE A PROGRAM.

SAVE is a special command for programmers that allows you to record on disk specific areas of computer memory. It should only be used by knowledgeable users.

To save a program, you need to know where the program is located in memory in hexadecimal numbers. If you do not know the four digit hexadecimal number, you cannot use this command. If you are saving a BASIC program, use the Q command to return to BASIC and save the program with the regular Save command: SAVE"program name",8.

1. If you do know the program address, enter S.

If a new diskette is not NEWED it will give errors. Sometimes an old diskette that always gives errors can be rejuvenated by NEWing it.

1. With the Help Disk in drive 0, type N and follow the instructions on the screen.

2. The screen will read "Format Diskette in Drive #?" Type 0 for normal applications or 1 or 2 for dual or triple disk drives.

3. The screen will instruct you to place the diskette you want to NEW into the disk drive. Be sure this is a disk which contains nothing you want to keep as the NEW command wipes out anything previously on the disk.

4. You will next be asked to enter the name you wish to assign to the diskette. No I.D. numbers are needed. Press RETURN key after the name is entered.

5. Answer the Are you sure? question with Y to continue or any other key to take you back to the main menu.

6. When the drive stops, the disk is NEWED. Remove the disk and make sure the HELP disk is in drive 0.

Q-QUIT TO BASIC

This is the quit to basic command and will put you back in BASIC. Run brings back the help menu if no other program is loaded.

2. The screen will read "Enter name:"
Enter the program name. If you want to
save on any drive other than drive 0,
enter drive # and colon before the name.

3. The screen will ask for the Start
Address of the program. Enter a four
digit hexadecimal number.

4. You will next be asked for an End
Address and an Execute Address. Both of
these should be answered in four digit
hexadecimal numbers.

U-UNKILL A FILE

1. Press U. The screen will read
"Enter name".

2. Place the diskette containing the
file you want to Unkill in a drive and
enter the name of the file you want to
Unkill. If you wish to Unkill a file in
a drive other than 0, enter the drive #
and a colon before the name, for
example, 1:ACCOUNTS will Unkill the
program ACCOUNTS from drive 1.

V-VALIDATE DISK

This command compresses the files. Quikdisk stores all files contiguous and sequential. Instead of using up disk space wherever it can be found, to gain speed, Quikdisk stores files sequentially. The V command in Commodore puts such a scattered file into sequential order. The V command in Quikdisk takes all the files and compresses them into the least possible space in sequential order. The most recent program is saved at the end of the directory. Eventually you will need to validate to maximize your use of the disk.

1. The screen will read "Drive #?"
Enter 0 for normal applications or 1 or 2 for dual or triple disk drives.

2. The disk is compressed

With a directory display you will notice that files previously killed are now completely gone and cannot be UNKILLED. Also, the quantity of free sectors should be increased by the amount of space previously occupied by killed files.

X-EXECUTE

This command allows you to load and run in one step. This is why the SAVE command asks for an execute address.

1. Press X.

2. Screen will read "Enter Name:".
Enter name of program.

3. The program already SAVEed in memory will load and run and appear on the screen.

\$ - DIRECTORY

\$-is a more extensive directory. It will show where the program is located on the disk, how big it is and other information.

As you can see, the disk helper is intended for two things. It allows the novice user to do disk maintenance with less syntax. It also allows the advanced user to do a lot of advanced functions that cannot be done on the 1541.

4. QUICK REFERENCE COMMAND SUMMARY

Initialization.....SYS56843
Display Directory.....LOAD"\$",8
Load first disk program.....LOAD"*",8
Load program.....LOAD"program name",8
Save program.....SAVE"program name",8
Open.....OPEN F,D,C"DR:Name,S,R"
Print#.....PRINT# F,A\$,"HELLO"
Input#.....INPUT# F,A\$,B\$,C\$
Get#.....GET# F,A\$,B\$,C\$,D\$

Disk Helper Commands:

Block display.....	B
Duplicate.....	D
Help.....	H
Kill.....	K
Load.....	L
Monitor.....	M
New(Format).....	N
Quit to Basic.....	Q
Rename.....	R
Save.....	S
Unkill.....	U
Validate.....	V
Execute.....	X
\$-Directory.....	\$

5. PROBLEM REPORT SHEET

Name.....

Address.....

City/State.....

Serial NO.....

Version NO.....

Computer type.....

Disk type.....

Printer type.....

Date.....

State the problem.....

.....

.....

What action were you attempting....

.....

.....

Comment.....

.....

.....

Send to: cgrs MICROTECH. P.O. Box
102, Langhorne, Pa. 19047

6. INDEX

Block display.....	pg.60
Changing device number.....	pg.39
Close.....	pg.38
Command summary.....	pg.70
Commands.....	pg.60
Compatibility.....	pg. 6
Connecting.....	pg.10
Data File Handling.....	pg.29
Device Number Change.....	pg.39
Directory(\$).....	pg.69
Directory Format.....	pg.50
Disk commands.....	pg.22
Disk helper.....	pg.59
Disk Organization.....	pg.49
Diskette handling.....	pg.13
Display directory.....	pg.15
Duplicate.....	pg.61
Duplicating a disk.....	pg.18
Error codes.....	pg.54
Execute.....	pg.68
File Name Loading.....	pg.25
Formatting(New) a disk.....	pg.16
General information.....	pg. 5
Get#.....	pg.37
Getting Started.....	pg. 5
Hardware.....	pg. 7
Help.....	pg.63
Index.....	pg.73
Input#.....	pg.33
Installation.....	pg.10
Kernal Routines.....	pg.40
Kill.....	pg.63
Load.....	pg.25
Loading.....	pg.64
Memory map.....	pg.57
Monitor.....	pg.64
New(Format).....	pg.64
Open.....	pg.30
Operation.....	pg.13
Print#.....	pg.35
Programming the disk controller..	39
Problem report sheet.....	pg.72

Quick reference card.....	pg.70
Quit to basic.....	pg.65
Rename.....	pg.66
Save.....	pg.28
Saving.....	pg.66
Sequential files.....	pg.22
Software.....	pg. 9
Specifications.....	pg. 8
Start up.....	pg.14
String Variable Loading.....	pg.26
Subroutines.....	pg.58
System Diagram.....	pg. 7
Transferring a program.....	pg.20
Troubleshooting.....	pg.57
Unkill.....	pg.67
Unpacking.....	pg.10
User's guide.....	pg.22
Validate.....	pg.68
Wild Card "?".....	pg.26