

# QL Disk Manual

Erratum - use FLP1 - Not FLD1 -

# PRELIMINARY OPERATION MANUAL - QL DISK

## CHAPTER 1 INTRODUCTION

QL DISK provides a number of advantages over the QL's native microdrive system. Perhaps the most important of these is the greatly reduced tendency to media corruption.

Tape media of any kind is prone to stretch-distortion. Microdrive cartridges are, like any mechanical device, also susceptible to wear. When this occurs, the message "CHANGED OR BAD MEDIUM" announces the probable destruction of valuable data. While this can also occur with a disk-based system, it is a relatively rare occurrence, the source of which can usually be found in a source other than the media itself.

Another advantage of the QL DISK system is that loading and search times are usually greatly reduced.

When the microdrive system is accessed, finding the required file, if it occurs immediately after the sector currently at the head position, can take up to 15 seconds for the system to point to the beginning of a file.

Further, if the file does not occur sequentially (from the system's point of view), it will continue running until file loading is completed. This can require several minutes of running time, while the tape passes the head several times looking for successive sector numbers.

In this sense, microdrives are single dimensional serial devices. An improvement in loading time can be made if, when you are satisfied with the content of a cartridge, you then copy the entire cartridge to another. This will place the copied files in an order more consistent with loading requirements.

QL DISK, however, being a two-dimensional device, allows a search to be made, according to sector header information, both in terms of sector and track location. Loading time improvements of better than 600% are not uncommon.

### DESCRIPTION SUMMARY

In order to obtain the maximum benefit of your QL DISK system, a few very important points should be observed. QL DISK is a precision piece of equipment and certain precautions should be observed in using it:

- \* Because QL Disk is precision engineered electro-mechanical device, under no circumstances should the unit be subjected to shock or vibration.
- \* When moving the QL and QL Disk, the drive unit should be disconnected from the interface. Failing to do this may damage the unit's connector or distort the alignment of the drive with the interface.
- \* Do not leave a disk in the drive when the power is switched off. Besides the distinct possibility of corrupting you disk, it is also possible that the head transport mechanism could be damaged.
- \* Both the disk drive and recorded media are very sensitive to outside magnetic fields. As a security precaution, we strongly recommend that both QL Disk as well as any media susceptible to corruption (disks or microdrive cartridges) are kept no closer than 30cm from your television or monitor.

\*\*\*\*\*  
Figure 1 - Assembly illustration  
\*\*\*\*\*

# PRELIMINARY OPERATION MANUAL - QL DISK

## A: Starting out

To transfer a file (data or program) from microdrive to the QL DISK system.

1. Place a disk in the drive type in `FORMAT FLD1_diskname`. Upon pressing `<ENTER>`, the formatting will commence (in the same way that cartridges are formatted).

At the bottom of your screen you will see the side being formatted and the progressive record of the track count.

When the information "720/720" is printed on screen, the formatting is completed and you can then begin to transfer a file.

Enter `"COPY MDV1_filename TO FLD1_filename"` and the transfer will be made from cartridge to disk.

Once completed and the flashing cursor is on your screen, input `"DIR FLD1_"` to verify that the copy has been made. As with a microdrive directory, directory information will include the number of sectors remaining followed by the full sector count of the disk.

QL DISK will hold about three microdrive cartridges of information. One of the principal advantages of disk drives over the cartridge system there is greater security in reliability. Also, programmes load faster.

Immediately beneath this will be the name of your file.

The next step is to type in `"LOAD FLD1_filename"`.

## B.1.0 SUGGESTIONS FOR FILING CONVENTIONS

### B.2.0 VARIOUS TYPES OF FILES

File type	Suggested Naming convention	Nature of file
BASIC	BAS	
BINARY	BIN or CODE	A program written in machin code or a compiled BASIC program.
SYSTEM	SYS	
DATA	DAT	
SCREEN	SCR or SCN	
TEXT	TXT or EDT	
COMMAND	CMD	
MATRIX	ARY	

## C.1.0 GENERAL INFORMATION REGARDING DISKS

Each track is divided into sectors, each with a 512 byte capacity:

Layout of the disk formatted in QL Disk is:

- 80 tracks per side.
- 9 sectors per track.
- 512 bytes per sector.

### NOTE: (ATTENTION)

The command `FORMAT` will, in common with the QL's microdrive system, completely erase all data contained on a disk, so use the command with care.

### C.1.1 DISK MEDIA

## PRELIMINARY OPERATION MANUAL - QL DISK

Syntax : DNAME FLDn\_media\_name  
Action :  
Parameters :

### 1.4.0 File catalogue

Command : DIR  
Syntax : DIR [#CH,] FLDn\_  
Action : Will list a complete catalogue of files from the named drive, on the named media, in either the default window (#1) or a named window.

System response is:

Name of volume  
Number of used sectors/number of free sectors  
File names.

Parameters : You can pause the file listing by using <CTRL><F5>. Press any other key to continue the listing. The listing can be stopped by using the break keys - <CTRL><SPACE>.

### 1.5.0 Volume statistics

Command : CAT  
Syntax : CAT FLDn\_  
Action : Similar to DIR, using this command will produce a catalogue of files on-screen. In addition, each file will include also include its length in bytes and the most its most recent date of saving.

To stop the listing momentarily, press <SPACE>. To discontinue the listing, press <ENTER>.

### 1.6.0 COPY commands.

#### 1.6.1 Copy a single file from one drive to another.

Command : DUPLIC  
Syntax : DUPLIC FLDn\_filename\_A TO FLDn\_filename\_B  
Parameter : filename\_A is the source file.  
filename\_B is the destination file.

#### 1.6.2 Copy all files from drive to another.

Command : BKP  
Syntax : BKP FLDn\_ TO FLDn\_

## 2.0.0 FILE HANDLING

### 2.1.0 Store BASIC Programs

#### 2.1.1 SAVE (\*\*)

Syntax : SAVE FLDn\_filename  
Action : A standard SuperBASIC command, this will save a filename provided there is no other identical name on the media.

#### 2.1.2 SAVE\_OVER

Syntax : SAVE\_OVER FLDn\_filename  
Action : Will over-write a file of the same name.

### 2.2.0 Store binary files

#### 2.2.1 SBYTES (\*\*)

Syntax : SBYTES FLDn\_filename,from\_address,file\_length

#### 2.2.2 SBYTES\_OVER

Syntax : SBYTES\_OVER FLDn\_

## CHAPTER TWO

## MAKING USE OF SUPER-TRAN-DOS (Disk Operating System).

In general, the structure of command entry is the same as it is for microdrive. The input of commands has been largely determined by the QL's operating system, QDOS. Several points, only briefly documented elsewhere, should be observed:

1. Ordinarily, filenames must start with an alpha character. Likewise, filenames must not contain spaces between individual components of the name.

"SAVE FLD1\_fred\_bas" is correct.

"SAVE FLD1\_fred bas" is incorrect.

However, in some circumstances, it is permissible to utilise otherwise unconventional filenames by enclosing the drive and filename within inverted commas:

SAVE 'FLD1\_123 FRED BAS'

will produce a file of that name. Conversely, when the file is next loaded, the drive and filename must be enclosed in inverted commas again.

## 1.0.0 OPERATION OF THE DISC DRIVE

## NOTES:

1. It is suggested that, until you become familiar with the operation of Dattel QL Disk, you should perform the various operations and commands on a new disk.

2. n = the drive number (either 1 or 2).

3. media\_name is the name given in the formatting of a disk volume.

4. file\_name is the name given to each file saved or transferred to a disk drive.

5. The convention followed throughout this manual:

[#ch] is an optional command.

(\*\*) indicates a command which is analogous to the SuperBASIC.

Using the general commands:

## 1.1.0 Formatting a new disk:

Command : FORMAT  
 Syntax : FORMAT FLDn\_media\_name  
 Action :  
 Parameters :

## 1.2.0 Initialisation

Command : INIT  
 Syntax : INIT FLDn\_media\_name  
 Action :  
 Parameters :

## 1.3.0

Command : DNAME

2.2.3 SEXEC (\*\*)

Syntax : SEXEC FLDn\_filename

2.3.0 Loading files

2.3.1 BASIC programs - to load and not run.

Command : LOAD (\*\*)  
Syntax : LOAD FLDn\_filename

2.3.2 BASIC programs - to load and run

Command : LRUN (\*\*)  
Syntax : LRUN FLDn\_filename

2.3.3 BASIC programs - to load and join files

Command : MERGE (\*\*)  
Syntax : MERGE FLDn\_filename

Command : MRUN (\*\*)  
Syntax : MRUN FLDn\_filename

2.4.0 Removing unwanted files

Command : DELETE  
Syntax : DELETE FLDn\_filename

2.5.0 Changing file names

Command : RENAME  
Syntax : RENAME FLDn\_filenameA TO FLDn\_filenameB  
Parameters : filenameA = the original filename,  
filenameB = the new filename  
Media must not be write-protected.  
Action : The named file will be re-named.

2.6.0 Searching for a particular file

Command : SEARCH  
Syntax : SEARCH FLDn\_filename  
Action : This command will cause a search of the entire media to the given filename. If present, a prompt will be given "File already exists"; if not, the prompt given is "Not found".

3.0.0 Error handling

3.1.0 To prevent an interruption of the DOS with an error

Syntax : ERSET  
Action :

3.2.0

Syntax : EROFF  
Action : The inverse of ERSET.

3.3.0

Syntax : var = ERROR (0)

Action :  
Example :  
10 REMark Loading a file  
20 ERSET  
30 INPUT "Name of file " ; a\$  
40 LOAD a\$  
50 ERR = ERROR (0)  
60 IF ERR <> 0 : GO TO 110  
70 PRINT "File " ; a\$ ; " loading"  
80 INPUT "To continue, press 0 " ; 0\$

## PRELIMINARY OPERATION MANUAL - QL DISK

```
90 IF o$ == "0" : GO TO 30
100 STOP
110 IF ERR = -7 : GO TO 200
120 PRINT "Retry"
130 GO TO 30
140 PRINT "The file does not exist."
150 GO TO 120
```

NOTE: The error code -7 corresponds to "NOT FOUND".  
A table of error codes has been included in appendix A2.

### 4.0.0 FILES

#### 4.1.0 Files, in general

##### 4.1.1 OPEN\_NEW

#### 4.2.0 Written file formats

- \* ASCII format
- \* Internal QL format
- \* Byte format

##### 4.2.1 ASCII format

##### 4.2.2 Internal QL format

##### 4.2.3 Byte format

#### 4.3.0 File opening

##### 4.3.1 Creating a new file

Command : OPEN\_NEW (#f)

##### 4.3.2 Opening a file in which alteration may occur

Command : OPEN (\*\*)

##### 4.3.3 Opening a file for reading only

Command : OPEN\_IN (\*\*)

##### 4.3.4 Opening a file which may be over-written by a new file

Command : OPEN\_OVER

Syntax : OPEN\_OVER #ch, FLDn\_filename

Parameter :

Action :

#### 4.4.0 File closing

In general, once writing to a file has been completed by one of the above commands, the SuperBASIC command CLOSE is used. Where a c number has been specified, the command is CLOSE #ch.

### 5.0.0 Random access of files

#### 5.1.0 Random reading of files

##### 5.1.1 ASCII format - INPUT command

Syntax : INPUT #ch, [list of variables]

Parameters : #ch :

[list of variables] : var 1, var 2 ..... , var

NOTE: Variables are each separated by commas.

PRELIMINARY OPERATION MANUAL - QL DISK

Action :  
The command INPUT

5.1.2 ASCII format - PRINT command  
 Syntax : PRINT #ch, [list of variables]  
 Parameters :  
 Action :  
 Example :  
 10 OPEN\_NEW #6, FLD1\_test  
 20 PRINT #6, "12345" \ "abcde" \ 5.12  
 30 REWIND #6  
 40 INPUT #6, a, b\$, c  
 50 PRINT a, b\$, c  
 60 REWIND #6  
 70 PRINT #6, "12345"; "abcde"; \ 5.12  
 80 REWIND #6  
 90 INPUT #6, b\$, c  
 100 PRINT b\$, c  
 110 CLOSE #6

5.2.0 Internal QL format - TAKE\_S command  
 Syntax : TAKE\_S #ch, [list of variables]  
 Parameters :  
 Action :  
 Example : TAKE\_S #6, a\$, b, t%

<u>LL</u>	<u>CCCC</u>	<u>XXXXXX</u>	<u>YY</u>
length of string	character string	variable ?? 6 bytes	complete variable 2 bytes
Variable string			

5.2.1 Internal QL format - WRITE\_S command  
 Syntax : WRITE\_S #ch, [list of variables]  
 Parameters :  
 Action :  
 NOTE :

5.3.0 Byte format - BTAKE\_S command  
 Syntax : BTAKE\_S #ch, [list of variables]  
 Parameters :  
 Action :

5.3.1 Byte format - BWRITE\_S command  
 Syntax : BWRITE\_S #6, [list of variables]  
 Parameters :  
 Action :

5.4.0 Pointers

5.4.1 Command : REWIND  
 Syntax : REWIND #ch  
 Action : Repositions the file pointer in the channel number to the beginning of the file.

5.4.2 Command : FIND  
 Syntax : FIND #ch, VOCKET  
 Parameters :



- Action :
- 5.4.3 Command : APPEND  
 Syntax : APPEND #ch  
 Parameters :  
 Action :
- 6.0.0
- 7.0.0 Matrix(?) Operations
- 7.1.0 Saving  
 Command : SARRAY  
 Syntax : SARRAY namtab TO FLDn\_filename  
 Parameters : namtab = Name of the variable table or matrix(?) of the BASIC program  
 Action :  
 NOTE :
- 7.2.0 Loading  
 Command : LARRAY  
 Syntax : LARRAY FLDn\_filename to namtab  
 Parameters : namtab = Name of variable table or matrix(?) of the BASIC program.  
 Action :  
 NOTE :

8.0.0 DIRECT ACCESS OF DISK SECTORS

- \* The number of the drive (n) - 1 or 2
- \* The number of the side (ns) - 0 or 1
- \* The number of the track (nt) - 0 to 79
- \* The number of the sector (np) - 1 to 9

- 8.1.0 Command : RS - To read a disk sector  
 Syntax : RS FLDn,ns,nt,np,DB  
 Action : To transfer 512 bytes of a disk sector; define the number of the drive (n), the number of the disk side (ns), the number of the track (nt) and the number of the sector (np) into a block of memory, beginning at address DB.

Example :  
 RS FLD1,1,18,5,HEX("3E000")

- 8.2.0 Command : WS - To exclaim(?) a disk sector.  
 Syntax : WS FLDn,ns,nt,np,DB  
 Action :  
 Example :  
 WS FLD1,0,18,5,HEX("3E000")

9.0.0

10.0.0 CONTROL OF THE MULTI-TASKING ENVIRONMENT

- PRIOR Allows a change of task priority.  
 ACTIV Activates an inactive task.  
 SUSJOB Suspends a named active task.  
 RELJOB Restarts a suspended job.  
 RMVJOB Removes a named job from the list of tasks.

Screen information regarding the job list is detailed and includes:

## PRELIMINARY OPERATION MANUAL - QL DISK

```

JOB NAME           : SuperBASIC
IDENTIFIER         : 00000000
LENGTH            : 00000000      (in bytes)
ENTRY POINT       : 00000000      (address)
OWNER JOB         : 00000000
PRIORITY          : 20             (0 to 127)
WAITING JOB       : 00
STATUS            : 0000
EXCEPTION VECTOR  : 00000000
    
```

```

10.1.0 Command    : JOBS
      Syntax      : JOBS [#ch]
      Action      : Lists the parameters of each job currently in the QL.
    
```

```

10.2.0 Command    : PRIOR
      Syntax      : PRIOR jobid, PR
      Parameters  : jobid = task number obtained by JOBS.
                   PR = new priority assignment of the task.
      Action      : Changes priority of defined job.
    
```

```

10.3.0 Command    : ACTIV
      Syntax      : ACTIV jobid, PR, TMOUT
      Parameters  : jobid = task number obtained by JOBS.
                   PR = new priority assignment of the task.
                   TMOUT =
    
```

```

10.4.0 Command    : SUSJOB
      Syntax      : SUSJOB jobid, flagA, TMOUT
      Parameters  : jobid = task number obtained by JOBS:
                   PR = new priority assignment of the task.
                   flagA =
                   TMOUT = length of task suspension, in units of 20ms
      Action      : Suspends an active job.
    
```

```

10.5.0 Command    : RELJOB
      Syntax      : RELJOB jobid
      Parameter   : jobid = task number obtained by JOBS.
      Action      : To re-activate jobid previously suspended.
    
```

```

10.6.0 Command    : RMVJOB
      Syntax      : REMJOB jobid, err_number
      Parameters  : jobid = task number obtained by JOBS.
                   err_number = Error code
      Action      :
    
```

### 11.0.0 TIME

```

11.1.0 Command    : CLOCK_ON
      Syntax      : CLOCK_ON
      Parameters  : None
      Action      :
      NOTE       :
    
```

```

11.2.0 Command    : CLOCK_OFF
      Syntax      : CLOCK_OFF
      Parameters  : None
      Action      :
      NOTE       :
    
```

### 12.0.0 SYSTEM PARAMETER CONTROLS

## PRELIMINARY OPERATION MANUAL - QL DISK

12.1.0 Command : GETBV  
Syntax : Variable = GETBV (jobid)  
Parameter : jobid = task number obtained by JOBS.  
Action :  
If jobid = 0, GETBV returns the address of the beginning of the SuperBASIC variable tables.  
If jobid = 0, GETBV returns the address of the end

12.2.0 Command : GETSV  
Syntax : Variable = GETSV ( ?? )  
Parameter :  
Action :  
Example :

```
100 a6 = GETBV(0)
110 ap_nibas = a6 + HEX("20")
120 ap_nlp = a6 + HEX("2d")
130 nibas = a6 + PEEK_L(ap_nibas)
140 nlp = a6 + PEEK_L(ap_nlp)
150 FOR ptl = nibas TO nlp
160 l = PEEK(ptl)
170 FOR ptn = ptl + 1 TO ptl + 1
180 PRINT chr$(PEEK(ptn));
190 END FOR ptn
200 PRINT
210 ptl = ptl + 1
220 END FOR ptl
```

### 13.0.0 CONVERSION UTILITIES

13.1.0 Command : HEX  
Syntax : HEX (character string)  
Parameters : Character string to be contained within "....." defined as a variable string terminated with "\$".  
Example : a = HEX("3E000")

13.1.1 Command : HEX\$  
Syntax : HEX\$(numerical value)  
Parameters : Can be any numerical value consistent with the QL SuperBASIC.  
Example : PRINT HEX\$(255) will print "FF".

13.2.0 Command : BIN  
Syntax : BIN(character string)  
Parameters : The character string to be contained within "....." defined as a variable string terminated with "\$". 1 character string will be composed exclusively of binary numbers - either "0" or "1".  
Example : PRINT BIN("0101") will print "5".

13.2.1 Command : BIN\$  
Syntax : BIN\$(numerical value)  
Parameters : Can be an numerical value consistent with the QL SuperBASIC.  
Example : PRINT BIN\$(5) will print "101".

### 14.0.0 MICRODRIVE EMULATION

## PRELIMINARY OPERATION MANUAL - QL DISK

### 14.1.0 Notes regarding emulation

If you have programs utilising call to microdrive, you can transfer them, unmodified, directly to disk

The command EMUL\_MDV provides a solution for file calls from within a program, without having to resort to converting a program with utilities written for this purpose. Whenever a call is made to MDV, this will be automatically translated so that the file is read from QL DISK's FLD, instead.

14.1.1 Command : EMUL\_MDV  
Syntax : EMUL\_MDV  
Parameters : None  
Action : A call for MDVn\_filename will be made to FLDn\_filename.

14.1.2 Command : EMUL\_OFF  
Syntax : EMUL\_OFF  
Parameters : None  
Action : Reverts control to normal drive names.

### 15.0.0 PROMPT LANGUAGES

QL DISK is the world's first and only bilingual DOS. Defaulting to French, prompts are also available in English.

15.0.1 Command : Francais  
Action : All prompts will be in French (the default)

15.0.2 Command : French  
Action : All prompts will be in French (the default)

15.0.3 Command : Anglais  
Action : All prompts will be in English.

15.0.4 Command : English  
Action : All prompts will be in English.

### 16.0.0 FILE COPYING

16.1.0 Command : COPY\_M  
Syntax : COPY\_M MDVn\_filename TO FLDn\_filename  
Action : Automatically converts all file call for mdv into FLD, but in their respective drive numbers.

16.2.0 Command : COPY\_MID  
Syntax : COPY\_MID MDVn\_filename TO FLDn\_filename  
Action : Automatically converts all file calls for mdv1\_ AND mdv2\_ to FLD1\_.

16.3.0 Command : CLONE  
Syntax : CLONE DRVnA TO DRVnB  
Parameters : DRV = either MDV or FLD  
nA and nB = the numbers of the source and destination drives.  
Action : Copies all files from drive A to drive B.

16.4.0 Command : CLONE\_M  
Syntax : CLONE\_M DRVnA TO DRVnB  
Parameters : DRV = either MDV or FLD  
Action : Copies all files from drive A to drive B, automatically changing file references of MDV to FLD for use on QL

# PRELIMINARY OPERATION MANUAL - QL DISK

## DISK.

16.5.0 Command : CLONE\_MID  
Syntax : CLONE\_MID DRVnA TO DRVnB  
Parameters : DRV = either MDV or FLD  
Action : Copies all files from drive A to drive B, automatically changing file references of MDV1\_ or MDV2\_ to FLD1\_ for use on a single QL DISK drive.

## APPENDIX I

### 17.0.0 MEMORY LOCATIONS

QL DISK's interface (I/F) occupies 16K of RAM

	Hex	Decimal	
BASE +	\$3FF4	16372	: Command register
BASE +	\$3FF5	16373	: Disk track register
BASE +	\$3FF6	16374	: Disk sector register
BASE +	\$3FF7	16375	: Data register
BASE +	\$3FF8	16376	: Disk side selector
BASE +	\$3FF9	16377	: Disk side selector
BASE +	\$3FFC	16380	: Disk drive one selector
BASE +	\$3FFD	16381	: Disk drive two selector
BASE +	\$3FFE	16382	: Reserved
BASE +	\$3FFF	16383	: Reserved

PRELIMINARY OPERATION MANUAL - QL DISK

APPENDIX 2

Table of Error messages

Message	Code
Not Complete	-1
Invalid Job	-2
Out of memory	-3
Out of range	-4
Buffer overflow	-5
Channel not open	-6
Not found	-7
File already exists	-8
In use	-9
End of file	-10
Drive full	-11
Bad name	-12
Xmit error	-13
Format failed	-14
Bad parameter	-15
Bad or changed medium	-16
Error in expression	-17
Overflow	-18
Not implemented	-19
Read only	-20
Bad line	-21
It's OK	0
Empty drive	1
Drive not on line	2
Read/write error	3
Not standard format	4
Double-sided media	5
File still open	6