

## CONTENTS

1. ALPHABETICAL COMMAND LIST
2. PARAMETER DOCUMENTATION
3. COMMAND DOCUMENTATION

---

### 1. ALPHABETICAL COMMAND LIST

A	*	ALLOCATE tracks
C	*	CREATE a FILE
D	*	DESTROY a FILE
F	*	FORMAT a disk
G	*	RESTART after simulated power failure
I	*	INPUT FROM a FILE if it exists
L	*	display a DIRECTORY LISTING
M		display a message
O	*	OUTPUT TO a FILE if it exists
Q		QUIT
R	*	RELEASE allocated tracks
S		display disk usage STATISTICS
T	*	display the allocation TABLE
X1	*	fixed sized file EXTENSIVE TESTING
X2	*	variable sized file EXTENSIVE TESTING
X3	*	final EXTENSIVE TESTING
0		QUIET MODE - don't display all the gory details
1		NOISY MODE - display all details for all commands

\* these commands interact with the program being tested. See below for functions tested.

#### MISCELLANEOUS DRIVER COMMANDS

Z	DRIVER TEST MODE - allows the driver to continue even on a status check error. This feature is primarily for driver debugging.
---	--

---

### 2. PARAMETER DOCUMENTATION

<u>Parameters</u>	<u>Usage</u>
address	a disk address or track number. Tracks are numbered from 0 up to 1 less than the total number of tracks.
expect	the expected status. If the command status does not agree, the test run will halt. Use the value -1 to suppress this check.
message	the rest of the line up to and including the new line (\n) character.
name	the name of a file. This must be preceded by exactly one space and must consist of exactly eight characters.
seed	a seed for the random number generator. If you don't know what this is for, you can just give it your favorite positive integer.
size	the size or average size of a file in extensive testing.
status	the return status to check against 'expect'
temp	indicates a variable for use by the driver
tracks	the number of tracks for an allocation request

---

---

### 3. COMMAND DOCUMENTATION

#### Usage

#### Program Statements

=====  
=== BASIC TESTING ===  
=====

Format a disk.

**F** <tracks>

**FormatDisk( tracks, 128 );**

Restart after simulated power fail. The number of tracks is taken from the Format (F) command.

**G**

**Restart( tracks , 128 );**

Allocate 1 or more tracks.

**A** <tracks> <expect>

**status = Allocate( &temp, tracks );**

Release 1 or more tracks.

**R** <address> <tracks> <expect>

**status = Release( address, numTracks );**

=====  
=== FILE SYSTEM TESTING ===  
=====

Create a file.

**C** <name> <tracks> <expect>

**status = CreateFile( name, tracks, temp );**

Destroy a file.

**D** <name> <expect>

**status = DestroyFile( name );**

Input from a file if it exists.

**I** <name> <expect>

**status = FindFile( name, temp1, temp2 );**  
**ReadFileTrack( temp1, temp2 )**

Output to a file if it exists.

**O** <name> <expect>

**status = FindFile( name, temp1, temp2 );**  
**WriteFileTrack( temp1, temp2 );**

Extensive tests...

X1 creates up to 5000 files of the given size, in a somewhat arbitrary order. It then deletes first the odd-numbered files, then the even numbered files. Calls to list directory are made between each block of calls. X2 uses "seed" to seed the random number generator, then creates up to 5000 files of sizes averaging the given size. The files are deleted as in X1. X3 does variable size tests with a fixed seed and a fixed average file size. In addition, X3 will generate more realistic head movements.

**X1** <size>

**X2** <seed> <size>

**X3**

```
=====
=== DISK AND FILE SYSTEM STATUS ===
=====
```

List a directory

**L** **ListDirectory();**

Display allocation table

**T** **Table();**

```
=====
=== DRIVER UTILITIES ===
=====
```

Disable detailed listing of commands

**0** **(No interactions)**

Enable detailed listing of commands (default)

**1** **(No interactions)**

Message display

**M <message>** **(No interactions)**

Quit

**Q** **(No interactions)**

Display statistics

**S** **Statistics();**

Special driver testing mode

**Z** **(No interactions)**