

DIGITAL MICROSYSTEMS, INC.

DSC 3/4

DIAGNOSTIC MANUAL

01-01-81

VER 1.01

Notice

Digital Microsystems, Inc. reserves the right to make improvements in the products described in this manual at any time without notice.

All rights reserved. No part of this publication may be reproduced without the prior written permission of Digital Microsystems, Inc. Please call (415) 532-3686.

(c) 1980 Digital Microsystems, Inc.

Table of Contents

- 1.0 Introduction
- 2.0 Floppy disk testing (FDTEST) (TFORMAT)
- 3.0 Memory testing (MEMTEST) (MEMSHORT) (SUBMIT)
- 4.0 Parallel port testing (PORTTEST)
- 5.0 RS232 port testing (RS232)
- 6.0 CP/M testing (SUBMIT) (TEST.SUB) (REMEMBER)
- 7.0 Network testing (NETTEST)

1.0 Introduction

This manual describes the programs on the Digital Microsystems DSC 3/4 Diagnostic diskette. (DMS Part # ??????????)

All programs on the DSC 3/4 Diagnostic Diskette are designed to run under the standard operating system that DMS provides to each customer. Neither the operating system nor CP/M utilities is provided on the diagnostic diskette.

All tests, with the exception of TEST.SUB and RS232, have 'in progress' commands. These commands allow instantaneous command summaries, error listouts, error summaries, and/or optional quitting of the program. Except for system failures, this almost eliminates the need for a unit to have a dedicated console. This allows one console to service many units without the loss of any information regarding transient errors.

These programs make up the contents of the DSC 3/4 Diagnostic Diskette:

TFORMAT	Floppy disk test format utility
*FDTEST	Floppy disk test program
*MEMTEST	Memory test program
MEMTEST.SUB	Submit file for MEMTEST
*MEMSHORT	Short memory test program
MEMSHORT.SUB	Submit file for MEMSHORT
PORTTEST	Parallel Port test program
RS232	RS232 Port test program
*TEST.SUB	CP/M utility test submit file
REMEMBER	TEST.SUB utility program
HDTEST3M	Memorex Hard disk test program
HDTEST3S	Shugart Hard disk test program

(*) The programs preceded by an asterisk either save, or optionally save, information on the diskette. This allows tests to proceed without any need of a dedicated console—even should the test (and/or system) fail completely. All programs write to the same file, CONFILE.TXT. Upon completion or termination of any of these test programs, information can be retrieved by entering TYPE CONFILE.TXT after a system reset. If the information in CONFILE.TXT must be saved, the file should be renamed, otherwise one of the other tests could easily destroy it.

2.0 Multiple Floppy disk testing

FDTEST

This test must be preceded by a FORMAT or (preferably) a TFORMAT. This program writes to formatted disks in three successive patterns: Linear, Random, and fixed. In the linear write mode, the disk is written to a linear, track by track fashion. In the random write mode, each disk is written to in randomly selected tracks. In the fixed write mode, a non-changing byte (stored at 103h) is written linearly onto the disk. This is used for testing bit-patterns most likely to cause errors on the type of drive being tested. After each write is performed it is read-tested in two successive patterns: Linear and Random. (The read patterns are identical in format as the above listed write patterns.) This test is usually run for approximately 24 hours. Since valuable error messages might be lost if the power fails, or if a terminal error occurs, a disk-write-of-console-output option is provided.

In general testing environments, the formatting of the diskette and the testing of the diskette are performed on different drives. This tests the compatibility of both of the drives with each other. A complete drive-compatibility test for a double drive system therefore consists of two 24-hour tests.

TEST 1 - TFORMAT disk on A, FDTEST disk on B. Duration: 24 hours
TEST 2 - TFORMAT disk on B, FDTEST disk on A. Duration: 24 hours

If no errors are detected from the first test in 24 hours quit (Q) the test and initiate the second test. If once more no errors occur in a 24 hour period, quit (Q) the second test. The drives should be considered to be compatible with each other.

If errors are detected, it is advised that the errors be noted, (or saved on the disk-if that option was used-) and the test re-executed. If the errors re-appear, closer investigation is called for. This will be described later.

If a certain drive is suspected to be faulty, the formatting of the diskette and the testing of the diskette should be performed on the same drive. This tests the drives compatibility with itself. Again, 24 hours should be a sufficient test length.

STARTING FDTEST

Reset the system (cold boot) and await the CP/M prompt.

```
CP/M 1.4xx
A>TFORMAT
```

Floppy Disk Format program 1.xx

```
Enter drive number (0 - 7) :1      [0=A drive, 1=B drive, etc]
Single or Double density:D        [S=single dens, D=double dens]
Type Return to start. <cr>       [Hit RETURN when disk is in B]
```

FORMAT COMPLETE

```
A>FDTEST
```

Floppy Disk Test 2.xx for the DSC3 and DSC4

Write buffer to disk or HiNet? Y

```
Disk name: (A,B,C,D) :?A
WAIT-creating file on disk
```

Test which disks (carriage return to terminate)?

```
Drive number (0-7)? 1
Drive number (0-7)? <cr>
Type Return <cr>
```

A command summary is now printed out and the test is started. With the options we selected, most of the console output will be written to disk. There are three console commands that can be entered at any time while the test is in progress. The 'in progress' commands are as follows:

F - The F (fixed track testing) option is described below.

R - Read out the 'error' buffer. The program keeps track of all write patterns and loop numbers, by writing them to a buffer located at 7000h. All errors are written to this buffer, also. When the buffer fills past 8000h the buffer is destroyed and re-initialized to 7000h.

S - Print the current status to console. This informs the user immediately as to what function is currently being performed, the number of errors that have been detected, track and sector under test, as well as the time and date.

T - Restart the test. New test drives can now be selected.

Q - Optionally quit the program. The program then asks the user if an exit is really desired. If a 'Y' is entered the test will terminate, otherwise the test will resume as if not interrupted.

? - If any other character is received from the console a menu of acceptable commands is printed out.

FDTEST Options

(1) Fixed byte.

The third write stage in any given loop is that of a fixed byte. This byte provides a bit pattern most likely to cause a failure on a Shugart model 800 floppy drive. This byte is defaulted to B6h can be changed easily through DDT. Simply change the byte at 103h to the desired value.

(2) Maximum number of errors (per drive)

The maximum number of any one type of error is stored at 104h. It is set to 10 (0Ah). As in the case of the 'fixed byte' above, it also can be easily set to any desired value through DDT. When a drive reaches its maximum capacity of errors, it is automatically logged out of the test. The log-out feature helps maintain testing efficiency.

(3) Disk write option.

This writes the contents of the R buffer (the error buffer which begins at 7000h) to a user-selected disk. When 128 bytes are collected the sectors worth of data is written to the next record of CONFILE.TXT. This will continue forever or until the disk is full or the test stopped. If desired, any valid CP/M file name may be entered on the command line. This file will receive the data instead of the default file CONFILE.TXT.

(4) Fixed track testing option

If an 'F' is entered as an 'in-progress' command the fixed track test mode option is selected. In fixed track testing mode the user can read or write continuously on any one track. After each read or write is performed, the BIOS error code (hex) returned to the user. If the number returned, is 00, a successful read/write was made. If the BIOS error code is not 00, an error has occurred. The program will now print out the NEC765 error buffer for inspection. The fixed track testing mode has its own commands.

- R - Read from the track.
- W - Write to the track.
- T - Test a new track. The user selects a new track.
- D - Done testing. Return to regular testing.
- X - Finished with the program. eXit to the system.

(5) Time option.

FDTEST records the time of the start of each test loop and the time of each error. If a system reset is made before the test is initiated the time will be set to 00:00:00. This can be a direct indicator of how long FDTEST has been running. If preferred, the time can be set to the current time thus indicating real time instead of elapsed time.

FDTEST errors

What follows is a list of all FDTEST error messages. All error messages, except those noted, are generated by information received from the NEC uPD765 Single/Double density floppy disk controller chip. A booklet on this chip is provided to each customer with the DSC3/4 manual. Next to each error explanation is a NEC code which references the Status Register Identification section of the NEC booklet.

DATE XXX-00-00 TIME 00-00-00
Disk X, ERROR: equipment check Track: XX [Current Function]

NEC flas - EC

If a fault signal is received from the FDD, or if the Track 0 Signal fails to occur after 77 Step pulses (Recalibrate Command) then the EC flas is set, thus triggering this error message.

DATE XXX-00-00 TIME 00-00-00
Disk X, SYNC error - CPU and FDC out of sync Track: XX [Current Function]

No NEC flas. BIOS generated error.

If the NEC765 controller chip is out of phase with the CPU chip, the DMS BIOS reports this error. This indicates that the NEC765 chip was/is misprogrammed, or that interrupts have been disabled or missed.

DATE XXX-00-00 TIME 00-00-00
Disk X, DATA CRC error Track: XX [Current Function]

NEC flas - DE,DD

When the FDC detects a CRC error in either the ID field or the data field, this flas is set. If, in addition the DD flas is also set, which indicates that the FDC detected a CRC error in the data field, this error message is printed.

FDTEST errors (cont)

DATE XXX-00-00 TIME 00-00-00
Disk X, TRAC error - head over wrong cylinder Track: XX [Current Function]

NEC flag - WC

The WC bit is related to the ND bit, and when the contents of C on the medium is different from that stored in the IDR, the WC flag is set and triggers this error message.

DATE XXX-00-00 TIME 00-00-00
Disk X, ENDT error - read beyond end of cylinder Track: XX [Current Function]

NEC flag - EN

When the FDC tries to access a Sector beyond the final Sector of a Cylinder, this flag is set and triggers this error message.

DATE XXX-00-00 TIME 00-00-00
Disk X, ID CRC error Track: XX [Current Function]

NEC flag - DE, (DD = 0)

When the FDC detects a CRC error in either the ID field or the data field, this flag is set. If the DD flag is not set, this error message is generated.

DATE XXX-00-00 TIME 00-00-00
Disk X, DMA ORUN - controller not serviced in time Track: XX [Current Function]

NEC flag - OR

If the FDC is not serviced by the main-systems during data transfers, within a certain time, the OR flag is set and this message is printed out.

FDTEST errors (cont)

DATE XXX-00-00 TIME 00-00-00
 Disk X, SECT error - Cannot find sector Track: XX [Current Function]

NEC flag - ND

During execution of READ DATA, WRITE DELETED DATA, or SCAN command, if the FDC cannot find the Sector specified in the IDR Register, this flag is set.

During execution of the READ ID Command, if the FDC cannot read the ID field without an error, then this flag is set.

During the execution of the READ A Cylinder Command, if the starting sector cannot be found, then this flag is set.

If any of the above conditions set the ND flag, then the SECT error message is printed.

DATE XXX-00-00 TIME 00-00-00
 Disk X, PROT error - disk write-protected. RESTART [cr][lf]

NEC flag - WP

This bit is used to indicate the status of the Ready signal from the FDD. If this bit is set, the PROT error message is printed and the program restarted.

DATE XXX-00-00 TIME 00-00-00
 Disk X, DENS error - wrong density (missing ID address mark) /
 Track: XX [Current Function]

NEC flag -

DATE XXX-00-00 TIME 00-00-00
 Disk X, MADR error - missing data address mark Track: XX [Current Function]

NEC flag - MD

When data is read from the medium, if the FDC cannot find a Data Address Mark or Deleted Data Address Mark, then this flag is set and the MADR error is printed out.

FDTEST errors (cont)

DATE XXX-00-00 TIME 00-00-00
Disk X, COMPARE error: Track: XX [Current Function]

No NEC flas. Program generated error.

If the data from the write buffer does not match the data in the read buffer, then this error message is generated.

DATE XXX-00-00 TIME 00-00-00
Disk X, INDETERMINABLE error: Track: XX [Current Function]

NEC flas -

If the NEC765 Controller chip reports an error, and it does not match any of the previously listed errors, this message is printed.

Could not write data to file

If the program cannot write the latest console data successfully to CONFILE.TXT, or to the file noted on the command line, this error message is printed.

ERROR OVERFLOW - LOGGING OUT DISK

If a disk has reached the maximum amount of errors, it is logged out of the test. The maximum number of errors allowed is stored in a byte at 104h. See FDTEST Options for more information.

Please note: Both CP/M and BIOS calls are used. The disk write feature uses CP/M file management, while the test reads and writes are performed through BIOS calls.

3.0 Memory Testing

MEMTEST
MEMSHORT

The Transcient Program Area (TPA) memory can be tested using MEMTEST or MEMSHORT. These programs are identical, except that MEMSHORT uses a quicker test algorithm. These programs write all the console output to disk, specifically CONFILE.TXT. MEMTEST and MEMSHORT have one 'in progress' command. Any console input will trigger the message. The message reports the following:

```
MEMTEST is in progress
Current phase is: XX
Current location under test: XXXX
```

Both MEMTEST and MEMSHORT can be used to test the entire TPA by using the supplied submit files MEMTEST.SUB and MEMSHORT.SUB.

During PHASE 01 of MEMTEST or MEMSHORT, memory is initialized to zeroes. During PHASE 02 of the test, a 1 is walked through a field of zeroes. Next, memory is re-initialized to ones. During PHASE 03 a 0 is walked through a field of ones.

Starting MEMTEST (or MEMSHORT)

Reset the system. MEMTEST must be on the boot-up disk, and must be write enabled.

```
CP/M 1.4xx
A>ERA CONFILE.TXT          <<----Erase any previous data
A>MEMTEST
** MEMORY DIAGNOSTIC **
BEGINING TEST LOCATION (IN HEX): XXXX <<----Starting addr in TPA
TOTAL TEST LENGTH IN (HEX) BYTES: XXXX <<----Test length in hex
TEST INCREMENT (IN HEX):XXXX
RECORD EACH ERROR ON CONSOLE? (Y OR N)Y <<----Always yes
REPEAT TEST? (Y OR N)X      <<----Yes for indefinite test
                             <<----No for one pass only

PHASE 01
PHASE 02
PHASE 03
```

```
0000 0000 0000 0000 0000
0000 0000 0000 0000 0000
0000 0000 0000 0000 0000
0000 0000 0000 0000 0000
0000 0000 0000 0000 0000
0000 0000 0000 0000 0000
```

MEMTEST (MEMSHORT) OPTIONS

```
A>MEMTEST          This option is described above.

A>MEMTEST 1       This option will automatically start MEM-
TEST on a 3000h test which starts at 100h.

A>MEMTEST 4       This option will automatically start MEM-
TEST on a 4000h test which starts at 4000h.

A>MEMTEST 8       This option will automatically start MEM-
TEST on a 4000h test which starts at 8000h.

A>MEMTEST C       This option will automatically start MEM-
TEST on a 2000h test which starts at C000h.

A>SUBMIT MEMTEST  This option will erase CONFILE.TXT and set
up a submit file to run the four previous
options sequentially.
```

Note: All options, excluding the last one, do not erase CONFILE.TXT. This is the responsibility of the test technician.

Note: All above options can be used with MEMSHORT. Merely substitute MEMSHORT for MEMTEST

4.0 Parallel Port Testing

PORTTEST

This program outputs an 8 bit pattern (starting with 00 and ending with FFh) out of the first parallel port, then reads it in to the second parallel port, and compares the two byte patterns. If they are not identical, an error message is printed to the console. Next, the bit pattern is sent out of the second, and read in from the first. Again, if the patterns are not identical, an error message is printed to the console. Next, the pattern is incremented and the test repeated. The test will repeat indefinitely, but all patterns will be completely tested in under sixty seconds.

NOTE: A special connector is needed for this test. (DMS PART # ?????????).

Starting PORTTEST

Install the special parallel port interconnecting cable.
Reset the system.

A>PORTTEST
PORTTEST VER 1.xx

Testing pattern
xxxxxxxx

<<---- Bit pattern of the byte currently
being test is printed out here.

Console 'in progress' commands are as follows:

- S - Print out a summary of all errors
that have occurred thus far. (Starts at 500h)
- Q - Optionally quit the program
- ? - Any other character generates a command summary

PORTTEST Error Message

ERROR PORT xx xxxxxxxx sent, but received xxxxxxxx

^ ^

portnumber |

|

Pattern sent from port xx

|

|

^

Pattern received from the other port

5.0 RS-232 port testing

RS232

This program outputs ascii characters to RS232 ports 0, 2, and 3. The three ports are also checked for input. If a character is ready, it is read, and if it is a valid printing ascii character it is echoed indefinitely (or until another valid character is found) to all three ports.

The program is self-starting. It defaults to the character 'A' (ascii 41h) if no character has been entered on the command line.

Ports 0, 2, and 3 are set to RS232, 9600 baud ports at the start of the program. This is necessary because certain versions of the DSC 3/4 BIOS initialize the ports differently.

Starting RS232

Reset the system.

CP/M 1.4xx

A>RS232 B <<----The 'B' is optional
RS232 VER 1.xx

BB.....(The B's print out continuously.)

The test is now in progress. Check ports 0, 2, and 3 for identical output. Input can be made to any of these ports, also. Any valid ascii character that is input will change the output to that ascii character.

RS232 has no error messages. If all port I/O do not produce proper and identical results.....

6.0 CP/M Utility Testine

TEST.SUB

This is a test of CP/M disk I/O, CCP, and utility programs. The purpose of this test is to exercise the system in a CP/M environment with a high duty cycle. It is usually used to gain confidence in the system after any repair.

The entire file is listed on the following page. TEST.SUB sets up a perpetual loop that should only terminate when the disk is full. In the loop are commands that exercise DIR, STAT, ASM, PIP, ERA, and also DMS utility TIME. This test saves the current function it is performing on the disk. Upon system failure, the disk-write function allows the test technician to see what function was being performed when the system failed. Under normal circumstances, the system failure should occur during the PIP function. It requires a CP/M disk with the following files:

```
ASM.COM    TIME.COM    SUBMIT.COM  MEMTEST.ASM
TIME.COM   PIP.COM     STAT.COM    REMEMBER.COM
```

In addition, another formatted diskette (use FORMAT) is required.

The test is run through the CP/M utility SUBMIT. If the reader is unfamiliar with SUBMIT, it is highly recommended the CP/M Manuals be consulted before starting the test.

Please note the frequent appearance of the file name REMEMBER. This trivial program writes the contents of the command line (which CP/M puts in a buffer starting at 80h) to CONFILE.TXT. If no command line is entered, CONFILE.TXT is erased. This allows the test technician to check what function was in progress before a system failure occurred.

This test has no 'in progress' error messages. If the console shows that the submit test is still running, no fatal errors have occurred. When MEMSUM.HEX finally overloads the A diskette, the test will ungracefully terminate with a CANNOT CLOSE FILE or NO DISK SPACE message from PIP. An examination of the console, or CONFILE.TXT should reveal the last utility that the test performed. If it was PIP, and a STAT command shows no read/write space available, the test terminated in an expected manner. There is a possibility that REMEMBER tried to write to the full disk, thus causing the failure, but since it writes only one record, the chance is slim.

If another command caused the system termination, closer examination is needed. If the system failed during the execution of ASM, STAT, or SUBMIT, quite possibly a bad copy of the file exists on the diskette. If the system failed during the execution of one of the CCP commands (DIR, ERA, or TYPE), quite possibly a bad copy of CP/M and/or BIOS is on the diskette.

Starting TEST.SUB

Reset the system. Put the CP/M disk in drive A, and place the second disk in drive B. (These disks must be put in their listed logical drives. However, the physical assignment of A or B can be easily changed through ASSIGN. Both diskettes must be write enabled.)

CP/M 1.4xx

A>SAVE 1 MEMSUM.HEX <<---This initializes MEMSUM.HEX

A>SUBMIT TEST

This starts the submit test. It will repeat until the file MEMSUM.HEX fills the A disk. This test has a special utility file of its own. REMEMBER.COM is used to write its command line to disk. (i.e., REMEMBER STAT will write 'STAT' to disk.) REMEMBER writes its data to a CP/M file (which it creates) called CONFILE.TXT.

If the unit under test should happen to fail, the function it was performing before the failure can be examined. This is achieved by a system reset, and entering TYPE CONFILE.TXT .

Here is the contents of TEST.SUB. It is highly recommended the user customize the file to achieve the most efficient form of testing for the individual application.

```
REMEMBER                <<--- This erases CONFILE.TXT .
REMEMBER DIR            <<--- This writes 'DIR' to CONFILE.TXT .
DIR *.COM \
DIR A*.??M \ This tests the CP/M DIR function.
DIR B: /
DIR /
REMEMBER TIME          <<---This writes 'TIME' to CONFILE.TXT .
TIME                  <<---This runs program TIME
REMEMBER STAT         <<---This writes 'STAT' to CONFILE.TXT .
STAT \
STAT B: \ This tests the CP/M STAT function.
STAT *.* /
STAT B: *.* /
REMEMBER TIME          <<---This writes 'TIME' to CONFILE.TXT .
TIME                  <<---This runs program TIME
REMEMBER ASM MEMTEST  <<---This writes 'ASM MEMTEST' to CONFILE.TXT .
ASM MENTEST.AAA       <<---This runs the CP/M assembler program and
                        creates the file MEMTEST.HEX
```

The file is continued on the next page.

We update CONFILE.TXT to show we are doing a PIP command.

REMEMBER PIP MEMSUM.HEX = MEMSUM.HEX, MEMTEST.HEX

Next we merge files MEMSUM.HEX and MEMTEST.HEX. MEMSUM.HEX increases in size each loop through the submit file. This will eventually fill the diskette, causing a system failure.

PIP MEMSUM.HEX = MEMSUM.HEX, MEMTEST.HEX

All files are now copied onto the B drive. This further tests the PIP utility.

REMEMBER PIP ALL FILES ONTO B DRIVE
PIP B:=*.*[V]

Once more the TIME utility is noted on CONFILE.TXT, and tested.

REMEMBER TIME
TIME

The erase command is now noted and tested.

REMEMBER ERA
ERA B:*.COM
ERA B:*.??M

REMEMBER TIME
TIME

Once more the TIME utility is noted on CONFILE.TXT, and tested.

SUBMIT TEST

The entire test is now restarted.

7.0 Network Testing

NETTEST

NETTEST is a Diagnostic Network Test for the DMS HiNet system. Direct BIOS calls are made, so no network master need be in operation. Each station in the test specifies his own station number, and the station number he wishes to send to. NETTEST is used to test the most basic operations of the HiNet.

Starting NETTEST

Reset the system. Ready two or more units for network interaction.

```
CP/M 1.4xx
A>NETTEST VER 1.xx
Network Test Program
```

```
Who am I? (in hex, please) - XX
To whom do I send (hex, if you will) - XX
How long a string of E5's should I send (in hex) - XX
Should I initiate the HiNet sending? (Y/N) X
Hit return when you want me to start <cr>
```

..*.*.*.*.*.*. (the print-out will continue indefinitely.)

NETTEST errors

**TIME-OUT ERROR

8.0 Hard Disk Testing

HDTEST3M

HDTEST3S