Diagnostics Reference Series

HOW TO USE DTOS

DTOS

DTOS

DTOS

DTOS

DTOS

DTOS

DTOS

Data General

Diagnostic Reference Series

How to Use DTOS

Data General

Notice

DGC reserves the right to make changes in specifications and other information contained in this document without prior notice, and the reader should in all cases consult DGC to determine whether any such changes have been made.

THE TERMS AND CONDITIONS GOVERNING THE SALE OF DGC HARDWARE PRODUCTS AND THE LICENSING OF DGC SOFTWARE CONSIST SOLELY OF THOSE SET FORTH IN THE WRITTEN CONTRACTS BETWEEN DGC AND ITS CUSTOMERS. NO REPRESENTATION OR OTHER AFFIRMATION OF FACT CONTAINED IN THIS DOCUMENT INCLUDING BUT NOT LIMITED TO STATEMENTS REGARDING CAPACITY, RESPONSE-TIME PERFORMANCE, SUITABILITY FOR USE OR PERFORMANCE OF PRODUCTS DESCRIBED HEREIN SHALL BE DEEMED TO BE A WARRANTY BY DGC FOR ANY PURPOSE, OR GIVE RISE TO ANY LIABILITY OF DGC WHATSOEVER.

IN NO EVENT SHALL DGC BE LIABLE FOR ANY INCIDENTAL, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING BUT NOT LIMITED TO LOST PROFITS) ARISING OUT OF OR RELATED TO THIS DOCUMENT OR THE INFORMATION CONTAINED IN IT, EVEN IF DGC HAS BEEN ADVISED, KNEW OR SHOULD HAVE KNOWN OF THE POSSIBILITY OF SUCH DAMAGES.

Ordering No. 015-000103

Printed in the United States of America
Rev. 01, March 1981

Preface

This book is a general introduction to Data General Corporation's Diagnostic Operating System (DTOS). It is written for those who have no prior experience with DTOS. It describes how to load DTOS, how to run test programs, and how to use DTOS as a troubleshooting tool.

The book does not explain the error reports produced by the test programs that run under DTOS. The text files for the programs describe the error reports.

While this book does provide a general conceptual framework for troubleshooting and some methods for deciding where to start with a system, it does not provide directions for repairing specific systems. For this information, consult the appropriate maintenance manual.

To use this book, you should be familiar with the following:

- The basic parts of a computer system.
- The basic concept of using an operating system program to load and control other programs (e.g., diagnostic, reliability, and exerciser programs).
- The operation of the switches on the front panel of the CPU (or a soft console equivalent) of the computer system you will run DTOS on.
- The use of the Command Line Interpreter (CLI) to enter commands through the keyboard of the system operator's console.

The equipment covered in this book includes the following:

ECLIPSE Series:	S/100, S/130, S/140, S/200, S/230, S/250 C/150, C/300, C/330, C/350, M/600, AP/130
NOVA Series:	NOVA 1200, NOVA 800, NOVA 2, NOVA 3, NOVA 4
MicroNOVA Series:	MicroNOVA, MP/100, MP/200

The information contained in this manual is summarized for easy reference in a pocket-sized manual entitled *DTOS Summary*, DGC No. 015-000082.

Table of Contents

Chapter 1
Introduction

In this manual, we introduce the various aspects of the DTOS system in the order that you use them when troubleshooting a system. We explain each step, idea, or procedure before proceeding to the next, and we use examples taken from actual systems whenever possible.

Organization of the Manual

Chapter 1 - Introduction describes the organization of the manual itself, defines the terms used in it and explains the conventions used in command lines and examples.

Chapter 2 - Overview of the DTOS System introduces the general structure of the DTOS media and explains the concepts you must understand to use it.

Chapter 3 - Loading, Restarting, Terminating explains how to load the DTOS system, what to do if you cannot load it, how to restart it, and how to terminate a DTOS session.

Chapter 4 - Initial Set-Up describes several features that let you control the way programs run and report errors.

Chapter 5 - Using the DTOS Media to Troubleshoot a System introduces basic troubleshooting concepts and describes in detail the commands that load test programs.

Chapter 6 - Advanced Troubleshooting Techniques explains the debugger and the ODT programs, which can be used by advanced field engineers to analyze test programs in memory.

Appendix A - Bootstrap Test (**TESTOK**) explains the outputs from the bootstrap test and the associated probable areas of failure in the hardware system.

Appendix B - DTOS System Error Messages explains the error messages you may receive on your system operator's console while running the DTOS system.

Appendix C - DTOS System Generation for Disk (**DDOS**) explains how to build the DTOS media on disk.

Appendix D - An Overview of the DTOS System Internals provides reference material for the advanced field engineer. It briefly explains the contents of the DTOS monitor, minimonitor, program control block, and the directory identification block.

Appendix E - Summary of DTOS Commands provides a summary of all the DTOS commands.

Definitions And Conventions

Throughout this manual, we use the definitions and conventions listed below.

Definitions

Halt
A stop that is built into a program. Most halts are an indication that the program has encountered an error.

Pause
A break during which the system waits for input from the operator. Most programs print a message requesting input (asking a question) before they pause.

Hang
A state in which the system cannot proceed. This may be caused by a hardware or a software malfunction. In the case of a software malfunction, the system may not have actually stopped, but may be in a loop that it cannot escape.

Circuit
The smallest grouping of components that perform a specific function. An example is a clock-generation circuit.

Module
The smallest physical assembly of circuits or components that is commonly removable. An example is a memory board.

Subsystem
One or more modules that together provide a specific capability. An example is a magnetic tape subsystem, which is made up of a controller and drive.

FRU
A Field Replaceable Unit (FRU) is usually the lowest-level unit that can be replaced in the field. Each DGC product has a defined list of FRUs. Some examples of FRUs for a disk subsystem are a disk controller board and a read/write head.

CPU
Central processing unit.

IOP
I/O processor.

Front Panel
The part of the CPU containing the ON/OFF switch and other control switches and lights.

Soft Console

Software implementation of front-panel data and function switches. At Data General Corporation, a soft console is often referred to as a virtual console (VC) or soft control panel (SCP). In a system with a soft console, you enter data through the system operator's console instead of through the front-panel switches. Where necessary, this manual instructs you with the relevant soft console procedures.

System Operator's Console

The CRT display terminal or hard-copy terminal responding to device codes 10/11 or 50/51 for TTY input/TTY output or device code 20 for MBC/1.

Conventions

We use the conventions given below throughout this manual.

Keys on the Keyboard

↵ or CR	CR (carriage return) key on the keyboard. Used to show the terminator character in examples or text.
LF	NEW LINE or LF (line feed) key on the keyboard
↑	Up arrow key on the keyboard
ESC or $	ESC (escape) key on the keyboard)
DEL	DEL (delete) or RUB OUT key on the keyboard)
↑ *x* or CTRL *x*	Control character. Depress and hold the CTRL key (represented by ↑) while you strike another key (represented by *x*).
adr	A logical address in memory (in octal).

Operator/System Dialogues

In all examples showing operator/system dalogues, operator input is shown in **BOLD FACE** capital letters while system responses are shown in ORDINARY capital letters.

Command-Line Formats

The command-line format for DTOS commands is as follows:

COMMAND *(argument [option],...)*

The conventions are as follows:

COMMAND We use upper-case bold-face letters to indicate commands and command mnemonics. These are typed into your console exactly as shown.

(*argument*) We use lower-case italic letters (usually in parentheses) to indicate that a command takes an argument. You must replace the letters or words in italics with the exact code for the argument you need. Do not enter the parentheses; they only set off the argument.

[*option*] Lower-case italics within brackets denote an optional argument. Do not enter the brackets; they only set off the argument.

... Three dots denote that you may repeat the required argument up to 15 times.

For the commands you use to run DTOS programs (see Chapter 5), the argument may take one of four different forms:

1. *program name*
   Meaning: one program only.
2. *program name,...*
   Meaning: more than one program (15 maximum).
3. *program name - program name*
   Meaning: a range of programs from the DTOS directory (15 maximum). The two names given represent the first and last programs in the range you want executed (that is, the first must come before the second in the DTOS directory).
4. *program name - program name, program name,...*
   Meaning: a range of programs along with other programs named individually (15 maximum).

Chapter 2
Overview of DTOS

What Is DTOS?

DTOS (Diagnostic Operating System) is a diagnostic operating system designed to load and run test programs on a wide range of DGC's computer systems.

The DTOS System

The diagnostic operating system is referred to as the "DTOS system" throughout the manual. It consists of the DTOS monitor and DTOS system programs.

The DTOS Monitor

The DTOS monitor controls the loading of both the system programs and the test programs. It also loads a minimonitor into the upper 1KW of the top of unmapped memory.The monitor passes DTOS system data to the minimonitor for saving before the test program overwrites the monitor. Consequently, in some modes of operation, the minimonitor will reload the monitor and restore it with the current system data.

DTOS System Programs

The DTOS system programs are a set of programs that support the monitor in performing various tasks. They are resident in memory only when being used. For example, the monitor loads the disk-builder program when you want to build the DTOS system on a disk.

Included among these system programs are the following:

- Bootstrap-load programs that load the DTOS monitor into memory from either magnetic tape or disk.
- A sizer program that determines the configuration of the computer system.
- A help program that accesses useful information concerning the DTOS system or the test programs.
- A disk-builder program that builds the DTOS media on disks.
- I/O programs that are used along with test programs for CPU-to-operator communication.

- Data channel programs that run DCH tests while other test programs are running.
- Debugger programs that allow you to display or modify memory cell contents, set breakpoints, etc.

Versions of DTOS Media

The DTOS system is grouped with test programs on either a magnetic tape or a diskette for distribution to users. The resulting group of programs is called the DTOS media.

Different versions of the DTOS media have been configured to support DGC's product line. Each version contains the DTOS system grouped with a set of test programs that test a specific CPU type and its subsystems.

Table 2.1 lists the various versions of the DTOS media along with their model and part numbers.

Product Line	Magnetic Tape		Diskette	
	DTOS Media Model No.*	Tape Number	DTOS Media Model No.*	Diskette Number
NOVA	3603M 3603H	074-000013	3603F (set of 2)	075-000004 075-000005
NOVA 4	3802M 3802H	074-000119	3802F (set of 4)	075-000071 075-000072 075-000073 075-000074
microNOVA			3650F (set of 2)	075-000009 075-000011
ECLIPSE	3604M 3604H	074-000012	3604F (set of 7)	075-000001 075-000002 075-000003 075-000012 075-000082 075-000083 075-000084
ECLIPSE and NOVA peripherals	3605M 3605H	074-000014	3605F (set of 4)	075-000006 075-000007 075-000008 075-000085
ECLIPSE M/600 CPU	3772M 3772H	074-000114		
ECLIPSE M/600 peripherals	3773M 3773H	074-000115		

Table 2.1 Versions of DTOS media

*In the model numbers, M = magnetic tape (800 BPI), H = high-density magnetic tape (1600 BPI), and F = floppy disk (diskette).

In addition to receiving the DTOS media on magnetic tape (or diskette for microNOVA), you can build it on other disks by following the procedures given in Appendix C. The resulting disk-based DTOS system is called DDOS (Diagnostic Disk Operating System).

The Test Programs

The DTOS monitor selects a test program (or programs) for loading and running from a group contained on the DTOS media. The test programs are not part of the DTOS system programs. However, since they are resident on the DTOS media, the test programs are dependent on the DTOS monitor for loading.

Directories of Test Programs

The DTOS media contains four directories, each with a unique name. A directory lists the names of test programs and related information such as the revision number, starting address, and location on the tape or disk. Consequently, the DTOS monitor looks in a specific directory to locate a test program before loading and starting it.

Each directory contains a different group of test programs. Each group tests a specific section of the computer system. Table 2.2 lists and describes the four directories.

Directory	Description
HOST	The HOST directory contains test programs that test the CPU and its subsystems. In computer systems without I/O processors (IOPs), the HOST directory is the only one that contains programs. The other directories are listed on the DTOS media, but they have no contents.
IOP	The IOP (I/O Processor) directory contains test programs that test the IOP and its subsystems. When loaded, these test programs run in the IOP, not the CPU.
HIOP	The HIOP (HOST/IOP) directory contains test programs for both the IOP and the HOST CPU. The test programs are arranged in pairs and run in parallel; that is, when one program runs in the HOST CPU, a corresponding program runs in the IOP.
HINT	The HINT (HOST/INTerface) directory contains test programs that test the interaction between the IOP and the CPU. These programs run in the CPU and test the IOP as a device attached to it. Selecting programs from the HINT directory allows you to test all the IOPs in a system.

Table 2.2 DTOS directories

NOTE: *Some programs may appear in more than one directory, because they test more than one area of the computer system. For example, when a program tests the host CPU, it loads from the HOST directory and, when it tests the IOP, it loads from the IOP directory.*

The directory to which you have access at any one time (via the DTOS monitor) is called the working directory. When you load the DTOS system, the HOST directory is the initial working directory. Whenever you select another directory (using a directory command explained in Chapter 4, "Initial Set-up"), the new directory becomes your working directory. Usually you have only one working directory at a time, and can access only the programs listed in that directory. However, you can make all the directories your working directory by issuing the ALL command. Using all the directories at once is valid only in auto mode to allow the DTOS monitor to access every program on DTOS media.

Definitions of Test Programs

The test programs in the directories can be of the following types:

Diagnostic — Diagnostic programs detect and isolate hard faults on subsystems and modules. They can also help isolate a problem to the failing circuit by providing functions such as looping on error. Their error reports indicate the probable failing FRUs.

Reliability

Reliability programs detect hard, intermittent, and interactive faults on subsystems and lower levels. They are used to verify operation in a simulated "user" environment, and their error reports provide information about the environment in which the error is is detected.

Exerciser

Exerciser programs detect and isolate hard, intermittent, and interactive faults in systems and distributed systems. Their error reports may indicate either the possible failing subsystem or FRU.

Verification

Verification programs are used to prove that a new product performs according to its functional specification.

Formatter

Formatter programs initialize magnetic media.

Timing

Timing programs provide help in calibrating equipment to meet timing specifications.

Alignment

Alignment programs help in aligning the heads of magnetic recording devices.

Features of the DTOS System

The features of the DTOS system are introduced below. They will be described in more detail in later chapters.

Bootstrap-Loading Test

The bootstrap-loading test runs whenever you bootstrap load the DTOS system from either tape or disk. It is a system program that quickly checks the basic functions of the computer hardware (excluding peripherals). When the bootstrap-loading test runs without finding any faults, it prints the word TESTOK. See Appendix A for more information about this test.

DTOS Commands

There are a number of DTOS commands that you can input when the DTOS monitor prompts with an asterisk (*). The commands are presented throughout the manual in the order that they are normally used when using the DTOS system.

The various DTOS commands allow you to perform activities related to the following features of the DTOS system.

Equipment Table

The equipment table is a list of the subsystems in the computer system that you are testing. The equipment-table commands allow you to display or change the contents of the equipment table.

Switch Register

The switch register is a word in the DTOS-monitor code that defines how the test programs are to run and to report errors. A switch-register command allows you to display and change the value of the DTOS switch register.

Help Files

The help files contain useful information about the DTOS system and some test programs. A help command accesses these files for displaying or printing.

Directories

Directory commands access and display the contents of the different test-program directories.

Loading Test Programs

Loading commands load the test programs for running in four different modes: auto, semi-auto, manual, and debug. The modes are described in Chapter 5, "Using DTOS to Troubleshoot a System."

Some loading commands load a DCH exerciser program at the same time that you load a test program.

Exiting DTOS

An exiting command performs an orderly end to the DTOS session.

Types of DTOS Errors

You may encounter either DTOS-system errors or test-program errors while using the DTOS media for troubleshooting a computer system.

DTOS-System Errors

DTOS-system errors occur when the monitor or one of the system programs encounters a problem while running. The program that finds an error *may* print an error message, or the program may halt or hang. Appendix B lists and explains the error messages produced by the DTOS system.

If the system program does not print an error message before halting or hanging, try bootstrap-loading the DTOS media again. If an error still occurs, report the error to DGC by sending in a Diagnostic Trouble Report (DTR) that explains in detail the problem and conditions in which it occurs.

Test-Program Errors

When a test program detects an error, it prints an error report. The information reported may vary from one test program to another, since testing methods and areas of failure vary. However, many programs produce easy to understand error reports. When you need additional information, see the text file of the test program.

Chapter 3
Loading, Restarting, Terminating

Loading Procedures

You must load the DTOS system into memory before you can run test programs. The loading procedure is commonly called *bootstrap loading*. You can load the DTOS system from either a magnetic tape or from a DDOS disk. The correct procedure for each is described below.

New releases of the DTOS media include a bootstrap-loading test that runs as part of the bootstrap-loading procedure. The test quickly checks the basic functions of the system hardware (excluding peripherals). When the test runs successfully, it displays the word TESTOK on the system operator's console. Otherwise it halts the CPU before displaying the entire word. See Appendix A for information about the TESTOK test.

During the loading process, the DTOS system programs may ask several questions that appear on the system operator's console. When a system program asks a question, it pauses and cannot continue until you have typed in the answer.

When successfully loaded, the DTOS monitor issues a prompt (*), which indicates that it is ready to execute commands.

Example 3.1 shows the messages and questions displayed from initial loading of the DTOS system to the issuing of the prompt (*).

```
TESTOK
ENTER CPU TYPE AND CR
(0 = MICRONOVA, 1 = NOVA, 2 = ECLIPSE) 2 ↵
CPU TYPE IS ECLIPSE. ENTER CPU SUBTYPE AND CR
0 = S200,C300; 1 = S230,C330; 6 = S140: 1 ↵
DOES THE HOST CPU HAVE AN ARRAY PROCESSOR?
TYPE (0 = NO, 1 = YES) AND CR 0 ↵
TOP OF MEMORY = 077777
HIDTOS REV 1.0
*
```

Example 3.1 Sample DTOS session on an ECLIPSE C/330

Loading the DTOS Media from Magnetic Tape

1. Power up the CPU and the magnetic tape drive.
2. Set the SELECT switch on the tape drive to 0.

   NOTE: *The DTOS media will load only from a magnetic tape drive with a primary device code.*

3. Remove the write-enable ring from your DTOS tape. Mount and thread the tape.
4. Press the LOAD switch and then the ON LINE switch on the tape drive.
5. Perform this step only if the system uses a soft console. Otherwise, skip to Step 6.

   a. Enter 100022L through the keyboard.
   b. Go to step 8.

6. Perform this step only if the system has the program-load option. Otherwise, skip to Step 7.

   a. Put 100022_8 on the front-panel data switches (0-15).
   b. Press the RESET switch and then the PROGRAM LOAD switch on the front panel.
   c. Go to step 8.

7. Perform this step only for a system with no soft console or program-load option.

   a. Press the RESET switch; then use the front-panel data switches (0-15) to load
      060122_8 into memory location 000376
      000377_8 into memory location 000377.
   b. Put 000376_8 on the front-panel data switches.
   c. Press the START switch on the front panel.

8. Observe that the word TESTOK is displayed to indicate that the bootstrap-loading test has successfully run. If the CPU halts and part or none of the word is displayed, go to Appendix A for information about the possible failure.
9. When the entire TESTOK word is displayed, the DTOS sizer program may ask questions about the CPU type and subtype. Respond by choosing the appropriate number from the question's prompt. If your answer is incorrect, either the DTOS sizer program will not continue or the DTOS monitor eventually will run test programs for the wrong CPU.
10. When loaded correctly, the DTOS monitor displays the following message:

    TOP OF MEMORY = XXXXX
    HIDTOS REV XX.XX

    When the monitor prompts with the asterisk (*), enter any desired DTOS command.

Loading the DTOS System from Disk or Diskette (except for microNOVA)

1. Power up the CPU and the disk or diskette drive.
2. Insert the disk cartridge, pack, or diskette into the unit 0 disk drive.
3. Recalibrate the drive. (See Table 3.1 for procedure.)

Type of Drive	Recalibration Procedure
6060/61/67	Press the RESET switch on the console.
Diskette	Open and close the drive door. Check to make sure the READY and TRACK 0 lights are on.
Other drives (systems with soft console)	Using keyboard, enter following octal values into memory locations and accumulators specified: Location Octal Value 000000 063033 (primary) 063073 (secondary) 000001 065333 (primary) 065373 (secondary) 000002 063077 AC0 000000 AC1 003400 Enter 000000R followed by a CR.
Other drives (systems without soft console)	Press the RESET switch and, using console data switches (0-15), enter octal values given above into memory locations and accumulators specified. Put 000000 on the console data switches. Press the START switches on the console.

Table 3.1 Recalibration procedures, disk drives

4. Perform this step only if the system uses a soft console. Otherwise, skip to Step 5.

   a. For 6060/61/67 disk drives, enter through the keyboard

      100027 L (primary) or 100067 L (secondary).

      For all other disk drives, enter through the keyboard

      100033 L (primary) or 100073 L (secondary).

   b. Go to Step 7.

5. Perform this step only if the system has the program-load option. Otherwise skip to step 6.

   Press the RESET switch; then:

   a. For 6060/61/67 disk drives, enter through the front-panel data switches (0-15):

      100027 (primary) or 100067 (secondary).

      For all other disk drives, enter through the data switches:

      100033 (primary) or 100073 (secondary)

   b. Press the PROGRAM LOAD switch on the front panel.
   c. Go to Step 7.

6. Perform this step for a system with no soft console or program-load option.

   a. Press the RESET switch; then use the front-panel data switches (0-15) to enter the appropriate octal values given in Table 3.2 into the specified locations.

Location	Value	
	For 6060/61/67 disk drives	For all other disk drives
000376	060127 (primary) 060167 (secondary)	060133 (primary) 060173 (secondary)
000377	000377	000377
ACO	000056 (primary) 000156 (secondary)	000066 (primary) 000166 (secondary)

Table 3.2 Loading without program-load option (step 6)

b. Enter 000376_8 through the front-panel data switches (0-15).

c. Press the START switch on the front panel.

7. Observe that the word TESTOK is displayed, which indicates that the bootstrap-loading test has successfully run. If the CPU has halted and part or none of the word is displayed, go to Appendix A for information about the possible failure.
8. When the entire TESTOK word is displayed, the DTOS sizer program may ask questions about the CPU type and subtype. Respond by choosing the appropriate number from the question's prompt. If your answer is incorrect, either the DTOS sizer will not continue or the DTOS monitor will run test programs for the wrong CPU.
9. The DTOS monitor displays the following message when loaded correctly.

```
TOP OF MEMORY = xxxxx
DDOS REV xx.xx
DISK I.D.#:xxxxx
```

10. When the monitor prompts with the asterisk (*), enter any desired DTOS command.

Loading the DTOS System from Diskette for microNOVA

1. Power up the CPU and the diskette drive.
2. Insert the diskette into the drive jumpered as logical unit 0 and observe that the READY light is on.
3. Select one of the system configurations in Table 3.3 and perform the procedure specified.

System Configuration	Procedure
microNOVA with ODT ROM or soft console	Enter XXL into terminal when you get prompt (!) from ODT in ROM. XX represents one of the following device codes for diskette: Device Code Diskette Subsystem 33 First 73 Second 30 Third 70 Fourth
microNOVA with hand-held console	Press RESET key and then CLR D key. Enter device code (see above). Press PR LOAD key.
microNOVA with automatic program-load	Press PL/START switch on processor (diskette must be in drive specified by jumpers for program-load option).

Table 3.3 Loading DTOS for microNOVA (step 3)

4. Observe that the word TESTOK displays to indicate that the bootstrap-loading test has successfully run. If the CPU has halted, and part or none of the word has displayed, go to Appendix A for information about the possible failure.
5. The microNOVA DTOS monitor displays the following message when it is loaded correctly.

```
TOP OF MEMORY = xxxxxx
DDOS REV. xx.xx
DISK I.D.#:xxxxx
```

NOTE: *TOP OF MEMORY is the top of RAM memory, not necessarily the top of the addressing space. ROM may be at the top of the addressing space.*

6. When the DTOS monitor prompts with an asterisk (*), enter any desired DTOS command.

Restart Procedure

Certain commands do not return the DTOS monitor at the end of their execution. To reaccess the monitor when this happens, you may either reload the DTOS system or use the following restart procedure.

1. Either use the value displayed as the top of memory during loading or choose an octal value corresponding to the maximum memory size from Table 3.4. The values in the table may not be accurate for microNOVA systems, however.

Switch Setting	Maximum Memory Size
007777	4K
017777	8K
027777	12K
037777	16K
047777	20K
057777	24K
067777	28K
077777	32K or greater

Table 3.4 Switch settings for restart procedure

2. Perform the following steps for systems with a soft console. Otherwise, skip to step 3.

   a. Enter the octal value (from step 1) through the keyboard followed by **R**.
   b. Go to step 4.

3. Perform the following steps for systems without a soft console:

   a. Press the RESET switch; then enter the octal value (from step 1) through the front-panel data switches (0-15).
   b. Press the START switch on the front panel.

4. When the monitor prompts with an asterisk (*), enter any DTOS command. If DTOS does not issue a prompt, press RESET and then repeat the above steps. If you still do not get a prompt, you must bootstrap load the DTOS system again.

   NOTE: *The restart procedure will not work if the test program was loaded with an auto-mode or a semiauto-mode command.*

Terminating Procedure

When you have finished using the DTOS system, use the **EXIT** command to leave DTOS. You may enter this command after receiving a DTOS prompt (*). **EXIT** will rewind the DTOS tape (or recalibrate the DDOS disk) and halt. After using **EXIT**, you can reaccess the DTOS monitor with the restart procedure described above.

What to Do if You Cannot Load the DTOS System

If your initial attempt to load the DTOS system fails (i.e., you do not get a prompt), any of a variety of system failures may be the cause. Check first for the following problems:

- Is the CPU power on?
- Is the CPU unlocked?
- Is the system operator's console on line?
- Is the magnetic tape properly threaded and the tape drive on line? (If loading from a disk, is the disk drive on line?)
- Are all of the cables in good condition and properly plugged in?

If your answer to any of these questions is no, correct the problem before loading the DTOS system a second time.

If the DTOS system still fails to load, try loading a third time by using the manual-load procedure (even if you have done so in your previous attempts). (See the Loading Procedures, above.)

This time, watch carefully the magnetic tape drive, the system operator's console, and the CPU's front-panel lights. Then, answer the questions below in the order given. In each case, if the question does not apply to your problem, skip to the next question.

Soft Console Operation

Were all of your inputs echoed on the system operator's console?

If no, the CPU may not be receiving information input through the keyboard of the system operator's console or the output path may be faulty. Check the system operator's console and its interface to the CPU.

Table 3.5 gives octal inputs which comprise a test that checks for the proper operation of both the TTI and TTO functions of the system operator's console. If either the keyboard or the display is faulty, the test will not output zeroes to the display.

Location	Value	Instruction
100	061111	DOAS 0,TTO
101	063511	SKPBZ TTO
102	000101	JMP 101
103	000100	JMP 100
AC0	000060	(ASCII value of 0)

Table 3.5 Testing for bad operator's console

Start the CPU at location 100 (input 000100R).

If no characters appear on the system operator's console, you probably have a faulty console or console controller board. Consequently, the CPU may not be receiving your load command.

If zeroes are printed on the system operator's console, the console is working properly.

Tape/Disk Operation

Did the tape move? (If loading from a disk, and the disk drive has front-panel lights, did those lights flicker?)

If no, try again to load DTOS manually, verifying each number that you enter.

If DTOS still fails to load, you may have a bad copy on the tape or disk. Try loading from the alternate medium. (If you have been loading from a tape, use a disk pack; if you have been loading from a disk, use a tape.)

DTOS System Operation

Was only part of the word TESTOK displayed on the system operator's console?

Or, if the system has the system operator's console on device codes other than 10 and 11:

Did the CPU halt and the PC appear in the front-panel lights?

Or, if the system has a soft console:

Did the bootstrap test output an octal number and the ODT prompt?

The TESTOK message is an output from the bootstrap test. The test outputs the entire TESTOK word when it has found no errors after quickly checking the basic hardware system (excluding peripherals) during bootstrap loading. If the test detects an error, however, it outputs only part of the word to indicate how far it went before encountering the error.

If only part of the word displays (or your answer to one of the other two questions above is yes), go to Appendix A for an explanation of the TESTOK bootstrap test.

If all of the TESTOK word was displayed (or the CPU is not halted and the program did not display an octal number and the ODT prompt), the bootstrap test ran without finding any errors. Continue with the next question.

If no part of the word displays (after 30 seconds), the following are probable areas of failure:

- Bad system operator's console, or console controller
- Bad CPU board
- First 256 to 512 locations of memory bad
- A JMP, JSR, or STA instruction failed
- An indirect JMP or JSR instruction failed
- TTO busy could not be cleared

Did the tape rewind (if a tape load)? Is it back at the "Beginning Of Tape" (BOT) marker?

If no, try to load DTOS from a different tape or from a disk.

If DTOS still does not load, call the next level of technical assistance available to you.

Did a question (asking for CPU type) appear on the system operator's console?

If no, follow the restart procedure (see above).

If DTOS fails to restart, check that the CPU is unlocked. (A locked CPU cannot be reset, thereby preventing the restart procedure from working.)

If DTOS still does not issue any questions, call the next level of technical assistance available to you.

Did an error message appear on the system operator's console?

If yes, see Appendix B for an explanation of the error message. If the error message indicates a failing magnetic tape unit, try loading DTOS from a disk.

If no, follow the restart procedure (see above).

If this does not produce a DTOS prompt, repeat the entire loading process again, starting from the beginning of this chapter.

If you reach this point a second time, call the next level of technical assistance available to you.

Chapter 4
Initial Set-Up

After you load the DTOS system and before you run any test programs, you should set up both of the following:

1. **The equipment table,** which lists the devices and subsystems in the system you are testing.
2. **The switch register,** which defines how programs run and report errors.

Adjusting both of these *before* you run test programs saves time and makes your troubleshooting session much easier. In this chapter we explain how to use the appropriate DTOS commands to make these adjustments. We also explain several other DTOS commands that can help you during your troubleshooting session.

The Equipment Table

A system-sizer program runs as one of the first DTOS system programs. It creates a CPU equipment table, which lists the equipment on your computer system. The sizer program recognizes (autosizes) most of the peripherals and CPU options in your system and automatically adds them to the CPU equipment table.

In large ECLIPSE systems (such as an ECLIPSE M/600) that have one or more I/O processors (IOPs), the sizer program can find IOPs on device codes 60 thru 67. If an IOP is present on the system, the DTOS monitor loads and starts the sizer program in the IOP. The sizer program then creates an equipment table for each IOP in the system.

When adding to or deleting from the equipment table, it is necessary to specify which equipment table you want to change. Consequently, the DTOS monitor assigns each equipment table a unique number. The identification number for the CPU equipment table is 0. The identification number of an IOP equipment table is equal to 1 plus the last digit (x) of the IOP device code ($6x$, where $x = 0$ thru 7). For example, the equipment table number for an IOP on device code 65 is $1 + 5 = 6$.

Not all of the equipment may appear in the equipment table, however, because the sizer program does not autosize some peripherals and CPU options. Also, if a peripheral is turned off or is not on line, the sizer program may not "see" it. If you want these devices to be tested, you should add them manually to the equipment table by using one of the equipment-table commands.

Later during the DTOS session, when you ask the DTOS monitor to run sets of tests on the entire system (in auto mode), the monitor uses the information contained in the equipment table to decide which test programs to run. Thus, you must make sure that the equipment table is complete and accurate by taking an inventory of the system and, if you have any questions, by asking the person in charge about the configuration of the system.

The CPU and the memory do not appear in the equipment table. These are considered to be the *system* that the devices in the equipment table are connected to. The DTOS monitor recognizes and runs tests on the CPU and the first 32K of memory even if the equipment table is empty.

> **CAUTION:** *If your hardware system contains any devices not sold by Data General Corporation, you must remove them* before *using the DTOS system. Otherwise you may get spurious or misleading error reports.*

Tables 4.1, 4.2, and 4.3 list device-code assignments for each DGC computer family (ECLIPSE, NOVA , and microNOVA, respectively). If a device code is shared by at least two families, it is considered to be a *common device code.* Assignments for these are listed in Table 4.4.

As you will see from studying the tables, there is no DTOS mnemonic for most of the device codes. Instead, the symbol printed consists of three dashes (---) or three stars (***). The dashes indicate that only one device is assigned to this device code by DGC. The stars mean that more than one device is assigned to this device code. However, only one device can be present in the system on a specific device code at any one time.

The tables indicate whether or not the DTOS sizer program automatically checks (autosizes) for the presence of the device. Normally, the device must be on line and ready before the sizer program can recognize it. The DTOS monitor uses the information gained during autosizing later, when running in auto mode. It runs a set of test programs for the autosized devices, but does not run any programs that require operator input.

Device Code[1]	Device Name	Assembler Mnemonic	DTOS Mnemonic	Autosized by DTOS?
1	Writable control store (C/300 and C/330)	WCS	---	No[2]
2	Error checking and correction	ERCC	***	Yes
3	ECLIPSE MAP (MMPU)	—	***	Yes
3	Memory allocation and protection (MMPU1)	MAP	---	Yes
4	Host, SP MAP (S/250)	ISPO	***	No
4	Demand-paging MAP (MMPU2) (C/150, M/600, AP/130)	DPM	***	No
4	IOP MAP, IDP timer (C/150, M/600, AP/130)	IOPI	***	No
4	Parity for ECLIPSE	—	***	Yes
5	Burst multiplexor channel	BMX	***	Yes
26	Fixed-head DG/disk (except C/300 and C/330)	DKB	***	Yes
27	DG/disk storage subsystem (except C/300 and C/330)	DPF	***	Yes
34	First cassette subsystem	CAS	***	Yes
34[3]	Data control unit (S/140, S/250, C/150, M/600, AP/130)	DCU	***	No
34	Multiline asynchronous controller	MX1	***	No
35	Multiline asynchronous controller	MX2	---	No
43	Digital I/O timer	DIOT	***	Yes
43	Programmable interval timer (S/140, S/250, C/150, M/600, AP/130)	PIT	***	Yes
60-67[4]	Satellite processor (S/250) I/O processor (M/600)	SP0-SP7 IOP	*** ***	Yes Yes
65	Host to IOP Interface (S/140, C/150, M/600, AP/130)	IOPI	***	Yes
66	Second fixed-head DG/disk subsystem (except C/300 and C/330)	DKB1	***	Yes
67	Second DG/disk storage subsystem (except C/300 and C/330)	DPF1	***	Yes
75	Commercial ECLIPSE	—	***	Yes
76	DCU-to-host interface (S/140, S/250, C/150, M/600, AP/130)	DPU	***	No
76	Floating-point option	—	***	Yes
77	AP option	—	***	Yes

Table 4.1 Device list for ECLIPSE computers

[1] See also Table 4.4 for common device codes 0, 6-24, 30-33, 36-42, 44, 46-64, and 70-75. Device codes 25 and 45 are unused (not autosized).

[2] When running in auto mode, the DTOS monitor automatically selects test programs that test this device if you have added it to the equipment table.

[3] The DCU is normally assigned device code 34, but it can be set up with any unused device code between 1 and 76. However, the DTOS sizer program will not autosize it on any device code.

[4] Device code 65 is primary for satellite or I/O processors. Note that device codes 60-63 may also be used as common device codes (see Table 4.4).

Device Code[1]	Device Name	Assembler Mnemonic	DTOS Mnemonic	Autosized by DTOS?
1	Multiply/divide unit (except NOVA 3 series)	MDV	---	Yes
2	Memory management unit (NOVA 3 series)	MMU	***	Yes
3	Memory management unit (NOVA 3 series)	MMU1	---	Yes
4	Parity option (NOVA 3 series)	PAR	***	Yes
2	Memory management unit (NOVA 4 series)	MAP	***	Yes
3	Memory management unit (NOVA 4 series)	MAP1	---	Yes
2	Memory management and protection unit (NOVA 800 series)	MMPU	***	Yes
34	First cassette subsystem	CAS	***	Yes
34	Multiline asynchronous controller	MUX 8	***	No
35	Same	CRC	---	No
64	Alternate location for floating-point unit (except NOVA 4 series)	FPU1	***	No
65	Same (except NOVA 4 series)	FPU2	***	No
66	Same (except NOVA 4 series)	FPU	***	No
74	Second cassette subsystem (except NOVA 4 series)	CAS1	***	Yes
74	Second multiline asynchronous controller (except NOVA 4 series)	—	***	No
75	Second multiline asynchronous controller (except NOVA 4 series)	—	***	No
74	Floating-point unit	FPU1	***	No
75	Same	FPU2	***	No
76	Same	FPU	***	Yes

Table 4.2 Device list for NOVA computers

[1] See also Table 4.4 for common device codes 0, 6-24, 30-33, 36-44, 46-63, 70-73, and 77. Device codes 5, 25-27, 45, and 67 are unused (not autosized). Device code 4 is unused except by NOVA 3 series computers.

Device Code[1]	Device Name	Assembler Mnemonic	DTOS Mnemonic	Autosized by DTOS?
2	High-speed channel tester		***	Yes
4	Hand-held Console	HHC	***	Yes
5	PROM Programmer	PROG	***	Yes
13	Unused	PTP	PTP	Yes
20	microNOVA board computer (MBC1)	MBC	***	Yes
21	First Model 4223 analog-to-digital interface	ADCV	---	No
23	First Model 4224 digital-to analog interface	DACV	---	No
26	First Model 6101 integrated rigid disk subsystem (12.5 megabyte + 1.2 megabyte)	DEP	***	Yes
26	First Model 6102 rigid disk subsystem (12 megabyte)	DEP	***	Yes
27	First Model 6095 cartridge disk subsystem (10 megabyte)	DPH	***	Yes
30	Third Model 6038/6039 diskette subsystem	DPX2	***	Yes
33	First Model 6038/6039 diskette subsystem	DPX	***	Yes
34	Asynchronous/synchronous line multiplexor	ASLM	***	No
35	Cyclic redundancy checker	CRC	---	No
42	Model 4222 digital input/output interface	DIO	---	No
52	Second paper-tape reader	PTR1	***	No
57	Second line printer	LPT1	***	No
61	Second Model 4223 analog-to-digital interface	ADCV1	***	No
63	Second Model 4224 digital-to-analog interface	DACV1	***	No
66	Second Model 6101 integrated rigid disk (subsystem)	DEP1	***	Yes
66	Second Model 6102 rigid disk subsystem	DEP1	***	Yes
67	Second Model 6095 cartridge disk subsystem	DPH1	***	Yes
70	Fourth Model 6038/6039 diskette subsystem	DPX3	***	Yes
73	Second Model 6038/6039 diskette subsystem	DPX1	***	Yes
74	Second asynchronous/synchronous line multiplexor	ASLM1	***	No
75	Second cyclic redundancy checker	CRC1	***	No

Table 4.3 Device list for microNOVA computers

[1] See also Table 4.4 for common device codes 0, 10-12, 14, 17, 50-51, 54, and 77. Unused device codes (not autosized) are 1, 3, 6, 7, 15, 16, 22, 24, 25, 31, 32, 36-41, 43-47, 53, 55, 56, 60, 62, 64, 65, 71, 72, and 76.

Device Code	Device Name	Assembler Mnemonic	DTOS Mnemonic	Autosized by DTOS?
0	I/O test board	IOBD	IOBD	Yes
6	Multiprocessor adapter transmitter	MCAT	---	No
7	Multiprocessor adapter receiver	MCAR	---	No
10	First data terminal in	TTI	TTI	Yes
11	First data terminal out	TTO	TTO	Yes
12	First paper-tape reader	PTR	PTR	Yes
13	First paper-tape punch	PTP	PTP	Yes
14	First real-time clock	RTC	RTC	Yes
15	First incremental plotter	PLT	PLT	Yes
16	First card reader	CDR	CDR	Yes
17	First line printer	LPT	LPT	Yes
20	First fixed-head disk subsystem	DSK	***	Yes
21	First 4120 series analog/digital conversion subsystem	ADCV	---	No
22	First magnetic tape subsystem	MTA	MTA	Yes
23	First 4180 series digital/analog conversion subsystem	DACV	---	No
24	Data communications multiplexor	DCM	---	No
30	First asynchronous hardware multiplexor	QTY	***	Yes
30	First synchronous line adapter	SLA	***	No
31	First IBM 360/370 interface	IBM1	---	No
32	First IBM 360/370 interface	IBM2	---	No
33	First moving-head disk subsystem	DKP	***	Yes
36	Interprocessor bus - half duplex	IPB	---	No
37	IPB watchdog timer	IVT	---	No
40[1]	IPB full-duplex input	DPI	---	No
41[2]	IPB full-duplex output	DPO	---	No
40	Synchronous communication receiver	SCR	---	No
41	Synchronous communication transmitter	SCT	---	No
42	Digital I/O	DIO	---	No

Table 4.4 Device list for common device codes

[1] The DPI is normally assigned device code 40, but it can also be set up with any unused device code above 40. However, the DTOS sizer program will not autosize it on any device code.

[2] The DPO is normally assigned device code 41, but it can be set up with any unused device code above 41. However, the DTOS sizer program will not autosize it on any device code.

Device Code	Device Name	Assembler Mnemonic	DTOS Mnemonic	Autosized by DTOS?
43	Digital I/O timer	DIOT	***	Yes
44	Modem control for MX1/MX2	MXM	---	No
46	Second multiprocessor transmitter	MCAT1	---	No
47	Second multiprocessor receiver	MCAR1	---	No
50	Second data terminal in	TTI1	***	Yes
51	Second data terminal out	TTO1	***	Yes
52	Second paper-tape reader	PTR1	***	Yes
53	Second paper-tape punch	PTP1	***	Yes
54	Second real-time clock	RTC1	---	No
55	Second incremental plotter	PLT1	***	Yes
56	Second card reader	CDR1	***	Yes
57	Second line printer	LPT1	***	Yes
60	Second fixed-head disk subsystem	DSK1	***	Yes
61	Second 4120 series analog/digital conversion subsystem	ADCV1	***	No
62	Second magnetic-tape subsystem	MTA1	***	Yes
63	Second 4180 series digital/analog conversion subsystem	DACV1	***	No
70	Second asynchronous hardware multiplexor	QTY1	***	Yes
70	Second synchronous line adapter	SLA1	***	No
71	Second IBM 360/370 interface	—	---	No
72	Second IBM 360/370 interface	—	---	No
73	Second moving-head disk subsystem	DKP1	***	Yes
74	Second cassete subsystem	CAS1	***	Yes
74	Second multiline asynchronous controller	—	***	No
75	Second multiline asynchronous controller	—	***	No
77	Central processing unit	CPU	***	No

Device list for common device codes, cont.

Displaying the Equipment Table

The EQUIP Command

As soon as the DTOS monitor first issues the prompt, examine the equipment table by issuing the **EQUIP** command. The DTOS monitor then displays all the equipment tables in numerical order on the system operator's console. The format is

EQUIP

This command takes no arguments.

Example 4.1 illustrates the use of the **EQUIP** command.

```
*EQUIP )
DEVICES FOUND BY THE SYSTEM HOST EQUIPMENT TABLE
EQUIPMENT TABLE # 0
NAME  CODE
---    3
---    5
TTI   10
TTO   11
RTC   14
MTA   22
---   27
IOP   65
***   75
IOP EQUIPMENT TABLE IOP # 65
EQUIPMENT TABLE # 6
NAME  CODE
---    0
---   77
*
```

Example 4.1 Equipment tables for HOST CPU and an IOP

Changing The Equipment Table

The **ADD**, **DELETE**, and **ASSIGN** commands allow you to make changes in the equipment table. Once you change the equipment table, your changes remain in the table until you either make more changes or reload the DTOS system.

The ADD Command

To add a device to the equipment table, use the ADD command. The format for this command is

ADD,*n* *(device-mnemonic, device-code),...*

where *n* is the number of the equipment table.

The **ADD** command accepts up to 15 devices at a time. If you want to add more than 15 devices, just use the **ADD** command again until all the devices are added. Tables 4.1-4.4 give the proper DTOS mnemonic to use for each device code.

If you do not know the DTOS mnemonic and do not have Tables 4.1-4.4 available for easy reference, you can just input one or more dashes (---) in place of the DTOS mnemonic. However, you must always include the device code. For example,

ADD,0 ---,22

is permitted, but

ADD,0 MTA,---

is not permitted.

The DTOS monitor will automatically report the correct mnemonic when you input the **EQUIP** command to display the equipment table.

There is one exception to the using the dashes, however. When adding a device to one of the device codes from 60 thru 67 on an ECLIPSE system, you must use two different inputs. If the device is not an IOP, use the dashes. If the device is an IOP, use **IOP** as the mnemonic for the command argument.

Example 4.2 illustrates the use of the **ADD** command. Note that we first use the **EQUIP** command to display the equipment table. We see that the sizer program did not "find" our disk. We check the disk drive and, finding it off line, we load a scratch pack and put the disk drive on-line. Then we add it to the equipment table with the **ADD** command. Using the **EQUIP** command a second time, we check to make sure that the disk drive is part of the equipment table.

```
*EQUIP ↵
DEVICES FOUND BY THE SYSTEM HOST EQUIPMENT TABLE
EQUIPMENT TABLE # 0
NAME  CODE
TTI    10
TTO    11
RTC    14
LPT    17
MTA    22
*ADD,0 ***,33 ↵
*EQUIP ↵
DEVICES FOUND BY THE SYSTEM HOST EQUIPMENT TABLE
EQUIPMENT TABLE # 0
NAME  CODE
TTI    10
TTO    11
RTC    14
LPT    17
MTA    22
***    33
*
```

Example 4.2 Effect of **ADD** command on equipment table

The DELETE Command

To delete devices from the equipment table, use the **DELETE** command. The format is

DELETE,*n* *(device-mnemonic, device-code),...*

where *n* is the number of the equipment table.

The **DELETE** command accepts up to 15 devices at a time. If you want to delete more than 15 devices, use the **DELETE** command again. You may use dashes to replace the DTOS mnemonic in the command argument if you do not know what the DTOS mnemonic is for the device.

Example 4.3 illustrates the use of the **DELETE** command. First, we examine the equipment table. We then use the **DELETE** command to delete both the paper-tape reader and the paper-tape punch. Using the **EQUIP** command a second time, we check to make sure these devices are no longer included in the equipment table.

```
*EQUIP ↲
DEVICES FOUND BY THE SYSTEM HOST EQUIPMENT TABLE
EQUIPMENT TABLE # 0
NAME  CODE
TTI   10
TTO   11
PTR   12
PTP   13
RTC   14
LPT   17
MTA   22
***   33
*DELETE,0 PTR,12,PTP,13 ↲
*EQUIP ↲
DEVICES FOUND BY THE SYSTEM HOST EQUIPMENT TABLE
EQUIPMENT TABLE # 0
NAME  CODE
TTI   10
TTO   11
RTC   14
LPT   17
MTA   22
***   33
*
```

Example 4.3 Effect of **DELETE** command on equipment table

The ASSIGN Command

To delete the entire equipment table and replace it with a new one, use the **ASSIGN** command. The format is

ASSIGN,*n* *(device-mnemonic, device-code),...*

where *n* is the number of the equipment table.

The **ASSIGN** command accepts up to 15 devices at a time. If the equipment table you are building contains more than 15 devices, you can add additional ones with the **ADD** command. You may use dashes to replace the DTOS mnemonic in the command argument if you do not know what the DTOS mnemonic is for the device.

Example 4.4 illustrates the use of the **ASSIGN** command. After examining the table we want to reconstruct, we use the **ASSIGN** command to construct a new one. When we issue the **EQUIP** command a second time, we see the new table, built from the devices listed in the **ASSIGN** command.

```
*EQUIP ↲
DEVICES FOUND BY THE SYSTEM HOST EQUIPMENT TABLE
EQUIPMENT TABLE # 0
NAME  CODE
TTI   10
TTO   11
PTR   12
PTP   13
RTC   14
LPT   17
MTA   22
*ASSIGN,0 --,3,--,4,LPT,17,TTI,10,TTO,11 ↲
*EQUIP ↲
DEVICES FOUND BY THE SYSTEM HOST EQUIPMENT TABLE
EQUIPMENT TABLE # 0
NAME  CODE
---   3
---   4
TTI   10
TTO   11
LPT   17
*
```

Example 4.4 Using **ASSIGN** command to reconstruct an equipment table

> **NOTE:** *Issuing the* **ASSIGN** *command with no devices listed after the equipment table number results in an empty equipment table.*

If, after altering the equipment table, you want to restore it to its original state, you can do so by running the sizer program again, either by reloading the DTOS system or by issuing the command,

LOAD.1 IOSIZER.

The Switch Register

After you have examined and modified the equipment table, examine and set the switch register of the DTOS monitor. This is a 16-bit word in memory.The bits of the word are numbered from 0 to 15, and each bit has one meaning when it is set to 1 and another meaning when it is reset to 0.

Switch 0 (the left-most bit) is a flag bit that indicates whether or not you have set any of the other switches (set = 1, reset = 0). The DTOS monitor sets this switch whenever you alter the switch register.

Each of the other switches affects in some way how the DTOS monitor runs programs or how the program reports information. Table 4.5 shows the effect of each switch.

Switch No.	Bit Value	Description
0	0 1	Switches in default condition (all zeroes). Switches not in default condition.
1	0 1	Loop on error. Do not loop on error.
2	0 1	Print out to console. Do not print out to console.
3	0 1	Do not print percent of failure. Print percent of failure.
4	0 1	Print pass count. Do not print pass count.
5	0 1	Do not print out to line printer. Print out to line printer.
6	0 1	Do not halt on error. Halt on error.
7	0 1	Do not print summary or passing of each subtest. Print summary or passing of each subtest.
8	0 1	Print only first error. Print every error.
9-11	—	Reserved for future use.
12-15	—	Used only by individual diagnostic programs. Refer to text files of specific diagnostic program for meaning of these switches (if used).

Table 4.5 Standard settings, switch register

Setting the Switch Register (the SWREG Command)

The SWREG command displays the switch register and enters a mode that lets you change its contents. The format is

SWREG

This command uses no arguments.

After issuing a **SWREG** command, the DTOS monitor displays the current value (octal) of the switch register. You may then just examine the value and exit with no changes, clear the switch register (reset the value to 000000), or change the settings of the switches (either set or reset them). These choices are accomplished as follows:

- To examine the switch register and make no changes, strike the LF key without typing in a new value.
- To clear the switch register, strike the CR key without typing in a new value.
- To set or reset switches, type in the new value (an octal number) followed by a CR. (Use Table 4.6 to find the correct value. Leading zeroes may be omitted.)

Switch	Value (Octal)	Switch	Value (Octal)
1	040000	9	000100
2	020000	10	000040
3	010000	11	000020
4	004000	12	000010
5	002000	13	000004
6	001000	14	000002
7	000400	15	000001
8	000200		

Table 4.6 Setting switches in the switch register

If you want to change the setting of only one switch, just input the octal value associated with the switch from Table 4.6. However, if you want to change the setting of more than one switch, add their associated octal values together and use the resulting number as your input. For example, when you want to change the settings of switches 6, 7, and 8, add 001000, 000400, and 000200 together to get 001600.

Example 4.5 illustrates how to use the **SWREG** command to change the setting of switches 6, 7, and 8. First, we input the **SWREG** command. The DTOS monitor responds by directing us to input our change in octal and by displaying the contents of the switch register. We see that the register is in its default state (all switches set to zero). We now input 001600, which is the number that we have previously calculated, followed by a CR.

Using the **SWREG** command a second time shows us the new value of the switch register (101600). Note that Switches 0, 6, 7, and 8 are now set to 1. Switch 0 has been automatically set by the DTOS monitor to indicate that the switch register is not in its default state (all zeroes). We press the LF key to end the session, because this time we do not wish to make any changes.

```
*SWREG ↵
INPUT IN OCTAL AND CR
SWREG= 000000 001600↵
*SWREG ↵
INPUT IN OCTAL AND CR
SWREG= 101600 LF
*
```

Example 4.5 Setting the switch register

Example 4.6 shows how to clear the switch register. We enter the SWREG command, and the DTOS monitor displays the contents of the register. We press the CR key without adding any new value. Inputting the SWREG command a second time, we see that the register has been cleared.

```
*SWREG ↵
INPUT IN OCTAL AND CR
SWREG= 102000 CR
*SWREG ↵
INPUT IN OCTAL AND CR
SWREG= 000000 LF
*
```

Example 4.6 Clearing the switch register

Using the Switch Register to Direct Outputs to the Line printer

Switches 2 and 5 determine where the DTOS monitor will output information such as error reports, help files (see the **HELP** command description), and lists of programs in a directory (see the **DIR** command description). Table 4.7 shows the effect of these two switches on the location of outputs.

Switches		**Information is displayed or printed on:**
2	**5**	
0	0	System operator's console but not printer
0	1	System operator's console and printer
1	0	Neither system operator's console nor printer
1	1	Printer but not system operator's console

Table 4.7 Effects of switches 2 and 5 on information displayed and printed out.

Before you can use switch 5, the line printer must be in the equipment table and on line and ready. Otherwise, the DTOS monitor pauses while waiting for the correct conditions to exist.

Changing the Switch Register While Running a Program

Some test programs on the DTOS media have their own built-in switch register. This switch register is distinct from the one in the DTOS monitor. However, the DTOS monitor passes the value of its switches to the program switch register each time it loads the test program. Consequently, every switch that was set in the DTOS-monitor switch register of the DTOS monitor is also set in the switch register of the test program.

You can examine and alter the switch settings in the switch register of the test program while the program is running. However, these settings are only temporary. They are lost when another program loads and runs. For this reason, you should set the most used switches in the DTOS monitor's switch register to allow the values to be passed to all test programs whenever they are loaded.

To examine the switch settings while the test program is running, input the letter **M**. If the test program has a switch register, the test program will display the current value of the switch register in binary format, with the bits labeled. Otherwise, the program will ignore the **M** command.

To change a switch setting at any time while the program is running, simply enter a single-digit *key* associated with the switch that you want to change. Table 4.8 shows the keyboard key that is associated with each switch in the switch register. Each input complements (changes) the setting of the associated switch. A switch set to 1 becomes reset to 0; a switch reset to 0 becomes set to 1.

Switch	Key	Switch	Key
0	0	8	8
1	1	9	9
2	2	10	A
3	3	11	B
4	4	12	C
5	5	13	D
6	6	14	E
7	7	15	F

Table 4.8 Switch/Key association

The key for switch 0 opens a *switch-modification mode* (and also complements switch 0) that stops the test program while you complement the switches whose settings you want to change. To leave the switch-modification mode and restart the program, type a CR.

Example 4.7 illustrates how to examine and modify switch settings while a program is running. First, we use the **LOAD** command to run a test program. After the test program runs once completely (Pass 1), we decide to examine the switch register, so we type **M**.

We then decide to change the setting of switch 5, so we type **5**. Typing another **M** shows us the new value of the switch register.

We then type both a **C** to change the setting of switch 12 and another **M** to check the new value of the switch register.

Finally, we change our minds about switch 5. We type **5** again to change it back to its previous setting, and once more check the value of the switch register by typing **M**.

```
*LOAD ECLIPSE23 ↲
LOAD:
ECLIPSE23 REV. 5
ECLIPSE23 IS RUNNING
MAP TYPE = MMPU1
PHYSICAL MEMORY SIZE IN 1K BLOCKS (DEC) = 96
PASS 1
M
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
0 0 0 0 0 0 0 0 0 0 0  0  0  0  0  0
5
M
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
1 0 0 0 0 1 0 0 0 0 0  0  0  0  0  0
PASS 2
C
M
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
1 0 0 0 0 1 0 0 0 0 0  0  1  0  0  0
5
M
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
1 0 0 0 0 0 0 0 0 0 0  0  1  0  0  0
```

Example 4.7 Changing switches while a program is running

Two other commands work with test programs that have a switch register:

CTRL D Resets the switch register to its default state (all zeroes) and restarts the program at its beginning. When another program is loaded or DTOS regains control, the switch register returns to its previous state.

CTRL R Restarts the program from its beginning, but leaves the switch register in its current state.

Switches under the MORT Programs

Six multiprogramming reliability tests (MORTs) run under the DTOS monitor. These are as follows:

NMORT	For NOVA series (except NOVA3 and NOVA4)
N3MRT	For NOVA 3 and NOVA 4 N3MORT
MNMORT	For microNOVA series
EMORT	For ECLIPSE series (except M/600, S/130, S/140, S/250, C/150 and C/350)
ECLMORT	For ECLIPSE M/600, S/130, S/140, S/250, C/150 and C/350
IPMORT	For I/O processors (IOPs)

The switch registers used by these programs differ slightly from the standard switch register. The MORTs use the switches reserved for specific programs, and they use non-standard definitions for some of the other switches. See *DTOS Summary,* (DGC No. 015-000082) for the values of the switches for the various MORT programs.

Getting Documentation and Making Notes

The HELP Command

The **HELP** command has two formats. One allows you to access helpful information from the DTOS system and the other allows you to access helpful information from the test programs. To print helpful information from the DTOS system, use the format:

HELP

This command loads a system program containing the help files and asks you to input a number corresponding to the help file that you want printed (0, 1, 2, or 3) followed by a CR. Table 4.9 lists the various help topics.

Help File	Function
Command summary	Lists the DTOS commands and a brief description of each.
Diagnostic trouble report	Prints a blank diagnostic trouble report (DTR) form.
Update list	Lists all the programs that have been changed since the last DTOS media release.
Product cross reference	Lists products sold by Data General Corporation with the diagnostic, exerciser, and reliability tests for each.

Table 4.9 Help files

The help file that you select is either displayed on the system operator's console or printed on the printer, depending on the settings of switches 2 and 5 of the switch register (refer to Table 4.7). The HELP command displays each file with no stops or pauses, so print the file on a hard-copy device (teletypewriter or printer) if you want to be able to read it. Since some of the files are very long, set the switch register so that your print-out will take as little time as possible. For example, if your system has a line printer that is faster than your hard-copy system operator's console, use the line printer. To stop a print-out once it has begun, halt the CPU and either reload or restart the DTOS system.

Example 4.8 illustrates the use of the **HELP** command. When we examine the switch register, we see that Switches 2 and 5 are set to 1. This means that help files will be printed on the line printer but not on the system operator's console (see Table 4.7). We strike the LF key to leave the switch register as is and then we type in the **HELP** command. DTOS loads the program containing the help files and directs us to enter the *number* of the file we want printed. We type a 0 followed by a CR, because we want the command summary printed out.

```
*SWREG ↲
INPUT IN OCTAL AND CR
SWREG= 122000 LF
*HELP ↲
LOAD:
HELP REV. XX
TYPE A 0(CR) FOR THE COMMAND SUMMARY OR
   A 1(CR) FOR A DTR FORM OR
   A 2(CR) FOR UPDATE LIST OR
   A 3(CR) FOR CROSS REFERENCE LIST:
0 ↲
```

Example 4.8 Using the **HELP** command

To print out the help portion of specified test programs, use the format:

HELP *(program name,...)*

or

HELP ALL

The **HELP** command loads each test program in the argument and looks for the help portion of that program. The argument can be one program, a group of programs, or all of the programs that have help files. For the different ways you can list test program names in the argument, see "Conventions," Chapter 1.

The **HELP ALL** command accesses all the help portions of programs that display text files.

If a test program has a help portion, the DTOS monitor prints it. If the test program does not have a help portion (many of the older programs in the DTOS system do not), the DTOS monitor does not print any help information for the test program, but goes on to the next one in the argument. At no time does the **HELP** command cause the test programs to run.

Again, Switches 2 and 5 of the switch register determine where the text will be printed (see Table 4.7).

The ENTER Command

Occasionally you may want to add a note to the beginning or end of an error report. You can write in notes by hand, or you can use the DTOS command, **ENTER**.

The **ENTER** command opens an *input mode* which allows you to enter text. The text is not stored in memory but is printed directly on either the system operator's console or the printer. Switches 2 and 5 of the switch register determine where the text will be printed (see Table 4.7).

When you have finished entering your text, use a CTRL Z (↑Z) to exit input mode.

Example 4.9 shows how to use the **ENTER** command. Examining the switch register, we see that Switch 5 is set to 1 to make sure that the text will be printed on the line printer and the system operator's console. We press the LF key to leave the switch register as it is. We then input the **ENTER** command. We need not wait for a prompt but can enter text when the cursor moves to the left margin. After we finish entering our text, we input a CTRL Z.

```
*SWREG ↲
INPUT IN OCTAL AND CR
SWREG= 102000 LF
*ENTER ↲
TODAY'S DATE IS 5-10-83 ↲
↑Z
*
```

Example 4.9 Using the **ENTER** command

Directory Commands

DTOS provides several commands for examining the contents of directories and moving from one to another. These commands are described below.

Displaying the Working Directory (the ? Command)

To display the name of the working directory, use the DTOS command, ?. In Example 4.10, we use this command and find our working directory is the HOST directory.

```
*? ↲
HOST
*
```

Example 4.10 Finding the working directory

Displaying the List of Test Programs in a Directory

To display the names of the test programs in the working directory, use the DTOS command, **DIR**.

When a directory contains a large number of test programs, the **DIR** command displays them on your terminal in small groups called *pages*. At the end of a page, the DTOS monitor directs you to type a space if you want the next page displayed or to type any other key to abort the display of the directory contents and to return to DTOS (see Example 4.11).

To print this list on a line printer, set switch 5 of the switch register to 1 before issuing the **DIR** command. This prints the entire directory at one time instead of a page at a time, as in the example.

```
*DIR ↵
HOST PROGRAM DIRECTORY
CAT0
CAT1
CAT2
DEBUG III
EDEB III
HIODT
HELP
IOSIZER
IOMOD
HDSKMINI0
HDSKBOOT0
HDSKMON0
HDSKMINI1
HDSKBOOT1
HDSKMON1
TYPE A SPACE TO CONTINUE OUTPUT ANY OTHER TO ABORT
*
```

Example 4.11 Displaying programs in the working directory

Moving from Directory to Directory

To move from directory to directory, simply type the name of the directory you wish to enter after the DTOS prompt. For a list of the directory names, refer to Table 2.2 in Chapter 2.

In Example 4.12, we use the ? command to find our working directory. We want to change this from the HOST directory to the IOP directory, so we type in IOP. We then use the ? command again to check that we are in the IOP directory.

```
*? ↵
HOST
*IOP ↵
*? ↵
IOP
*
```

Example 4.12 Changing the working directory

The TAPE Command

You use the **TAPE** command when loading the DTOS system programs and test programs from a DDOS (Diagnostic Disk Operating System) disk. If you have chosen to incorporate more than one DTOS tape during the disk-building process, the DDOS disk contains a separate disk directory for each DTOS tape used. Consequently, the **TAPE** command allows you to move from one tape directory to another on the DDOS disk.

Example 4.13 shows how to use the **TAPE** command. When we issue the command, DTOS displays the number of the tape directory we are currently accessing and tells us to type in the number of the tape directory that we want to access next. We input 1 and then issue the **TAPE** command again to verify that we have moved to directory 1.

Note what happens when we try to move to directory 2. Since this disk contains only two directories (0 and 1), we get an error message and are returned to directory 0, the default number.

```
*TAPE ↵
CURRENT TAPE NUMBER= 000000
TYPE DESIRED TAPE NUMBER (0,1,OR 2)
1 ↵
*TAPE ↵
CURRENT TAPE NUMBER= 000001
TYPE DESIRED TAPE NUMBER (0,1,OR 2)
2 ↵
SPECIFIED TAPE NUMBER IS INACTIVE. PRESENT TAPE
NUMBER HAS BEEN SET TO 0, THE DEFAULT TAPE NUMBER.
*
```

Example 4.13 Using the **TAPE** command

Chapter 5
Using DTOS to Troubleshoot a System

General Troubleshooting Concepts

The four basic steps to any troubleshooting session are as follows:

1. Make a preliminary evaluation of the situation.
2. Run test programs to isolate the failing device.
3. Repair the system.
4. Verify the repair.

This chapter deals with steps 1, 2, and 4. For step 3, which is concerned with specific repairs and troubleshooting tips, consult the appropriate maintenance manuals for the computer system.

Preliminary Evaluation

This procedure consists of two tasks, as follows:

1. Collect the symptoms of the failure.
2. Form an initial conclusion.

Collecting the Symptoms

When possible, talk with the person who was working with the system when the failure occurred. Also, examine carefully any printouts that document the machine's state or work load at the time it failed.

After gathering all the information available on the system's failure, you should have answers to these questions:

- Is the failure intermittent, or was it a one-time event?
- Is the system still operational, or is it down?
- Is the error the same or different each time?
- Do the symptoms form a consistent picture?
- Have any hardware or software changes been made recently?

- Has the system had similar problems in the recent or distant past? What were the conclusions at that time?
- Was the solution then applied adequate, or could this be the same problem recurring?
- What does the user think the problem is? Is his theory consistent with your experience?

Forming an Initial Conclusion

If you have answers to the above questions, you are ready to form an initial conclusion:

1. The symptoms do not provide a clue to the source of the problem.
2. The symptoms point to a general area of the system but not to a specific device.
3. The symptoms point to a specific device.

Running Test Programs

Once you have completed your preliminary evaluation, the next step is to load and run test programs. You must decide what test programs to run and whether to run the data channel exerciser program along with them. Do this as follows:

1. If the symptoms do not provide a clue to the problem, use an auto-mode command to load sets of test programs that are selected by the DTOS monitor (based on the contents of the equipment table). Load the data channel exerciser along with the set of test programs unless you do not have enough time.
2. If the symptoms point to a general area of the system, use a semi-auto-mode command to run sets of test programs that will test the suspected area as completely as possible. Load the data channel exerciser along with the set of programs if you suspect faulty interaction between the memory map and the mass storage device.
3. If the symptoms point to a specific device or subsystem, run the test programs that test the suspected device or subsystem. Use either a semiauto-mode or a manual-mode command. Do not load the data channel exerciser along with the test programs.

If the first test programs that you run disprove your initial conclusion, start over and use an auto-mode command.

Modes of Operation

The DTOS monitor gives you a choice of four modes in which to run the test programs. You select a mode by the DTOS command that you use when loading a test program (or programs). Table 5.1 lists the modes and the various commands you use to select a mode. Each command is explained following a description of the modes.

Mode	Commands
Auto	ACCEPT, RUN,*n*, CRUN,*n*, TOPDOWN, CTOPDOWN
Semiauto	LOAD.*n*, CLOAD.*n*, REPEAT, CREPEAT, SELECT, CSELECT
Manual	LOAD, CLOAD, MLOAD
Debug	ODT, DEBUG

Table 5.1 Modes of operation and commands used in each

Auto Mode

When you use one of the auto-mode commands, the DTOS monitor automatically selects a set of test programs and runs them one after another. The monitor selects only those tests that do not require any operator interaction. Consequently, most test programs for peripherals cannot be run in auto mode because they require some form of operator response. To run such test programs, load them with commands for one of the other modes.

When the entire set of test programs finishes running, the DTOS monitor issues its prompt (@) to indicate that it is ready to accept more commands. When you load the programs with the **TOPDOWN** or **CTOPDOWN** commands, you will not get the prompt when the programs finish running.

Selecting auto mode may be the most efficient way to isolate a fault, especially when you are troubleshooting a system and have little or no idea of what the problem is.

Semiauto Mode

The semiauto-mode commands let you select any test program or set of programs that you want the DTOS monitor to run automatically. Once you input a command, the set of programs runs unattended, provided that a test does not require an operator response. If operator interaction is required, the program pauses and waits for the requested input.

After the specified programs have completed running, the DTOS monitor issues its prompt to indicate that it is ready to accept another command. When you load the programs with the **REPEAT** or **CREPEAT** commands, you will not get the prompt when the programs finish running.

Select semiauto-mode when you want to assume some control of the test-program selection but still keep the other features of the auto mode.

Manual Mode

The manual mode commands let you select any test program that you want to run continuously without ever returning control to the DTOS monitor.

Once you issue a manual-mode command, the test program runs unattended, provided that the test does not require an operator response. If operator interaction is required, the program pauses and waits for the requested input before continuing.

Since tests loaded in manual mode run continuously, you must use the STOP switch on the front panel (or the RESET switch for soft consoles) to stop the testing. Generally, you will have to reload the DTOS system, because the test program usually overwrites the minimonitor.

Select the manual mode when you want to load just one test program and do not want control returning to the DTOS monitor.

Debug Mode

The debug-mode commands direct the DTOS monitor to load either an octal debugging tool (ODT) or a symbolic debugger along with the test program that you want to run. This lets you perform various functions, such as displaying or changing memory cells and setting breakpoints, while the test program is resident in memory.

The Data Channel Exerciser

The data channel exerciser is a program that tests the data channel by writing to and reading from a removable disk or an I/O test board (IOBD). You can load this program along with the other test programs by prefixing one of the load commands (listed in Table 5.2) with a C. The command runs the selected test program once without the data channel exerciser. Then, on the second and each subsequent pass, the data channel exerciser runs along with the selected test program. Also, if the load command runs a set of programs, the data channel exerciser runs with each test program in the set. However, please note that some test programs do not run with the data channel exerciser.

Without DCH Exerciser	With DCH Exerciser
LOAD	CLOAD
LOAD.*n*	CLOAD,*n*
REPEAT	CREPEAT
RUN,*n*	CRUN,*n*
SELECT	CSELECT
TOPDOWN	CTOPDOWN

Table 5.2 Commands accepting C prefix

Before the data channel exerciser runs, DTOS will ask you to specify the test device. If you are using a disk, it must be on line. If you specify a test device that your system does not have or a disk that is off line, the system will hang.

> NOTE: *Since the data channel exerciser writes on the disk, you* must *put a scratch pack in the disk drive before using the data channel test. Exception: if you are running DTOS from a DDOS disk and are using the same disk as the test device, the data channel exerciser writes only on a sector of the DDOS disk meant for this purpose.*

Example 5.1 shows the use of the **CLOAD** command to load the data channel exerciser along with a test program. In this case, we are using a Model 6061 disk drive as our test device. Note that the data channel exerciser prints an **S** when it begins running and a **P** each time it makes one failure-free pass. The data channel exerciser and the test program do not necessarily complete a pass at the same time.

```
*CLOAD ECLIPSE23 ↵
INPUT DATA CHANNEL EXERCISER TO BE RUN
(0= STANDARD DISK/FLOPPY, 1=ZEBRA DISK, 2=I/O TESTER,
3=MICRONOVA FLOPPY, 4=QUAD DENSITY FLOPPY)
1 ↵
ENTER DISK DEVICE CODE
27 ↵
LOAD:
ECLIPSE23 REV. 05
ECLIPSE23 IS RUNNING
MAP TYPE = MMPU1
PHYSICAL MEMORY SIZE IN 1K BLOCKS (DEC) = 96
PASS  1   SP
PASS  2   P
PASS  3   P
PASS  4
```

Example 5.1 Data channel exerciser running with the **LOAD** command (on an ECLIPSE S/130)

Auto-Mode Commands

The auto-mode commands are **TOPDOWN**, **RUN,*n***, and **ACCEPT**. They assume complete control of your system, i.e. they select sets of test programs and run them. However, the set does not include any test programs that ask for operator input. The time required for the set of programs to run depends on the configuration of your system.

The TOPDOWN Command

Format:

TOPDOWN

or, if the series is to be run with the data channel exerciser,

CTOPDOWN

This command uses no switches or arguments.

TOPDOWN forces the working directory to be all the directories (it is the only DTOS command that selects its own directory). It runs a series of test programs, beginning with programs that test the system as a whole. Each subsequent program in the series tests a smaller portion of the system than the program before it.

If a test program detects an error, the **TOPDOWN** command selects and runs a sequence of programs that further tests the failing area. The **TOPDOWN** command continues running through the sequence until it finds a test that does not detect an error. **TOPDOWN** then runs the last program that *did* detect errors, but runs it in manual mode.

From the failing program and from the error reports generated by **TOPDOWN**, you can usually isolate a specific area of the system that is failing. You then run tests to isolate the specific device that is failing.

As shown in Example 5.2, the first program **TOPDOWN** loads is the appropriate MORT. When the MORT finishes running, **TOPDOWN** runs the next program that is appropriate for this system, then the next, etc.

```
*TOPDOWN ↲
TOPDOWN
LOAD:
NMORT S REV. 08
:
etc.
*
```

Example 5.2
TOPDOWN command
(on a NOVA 800)

The RUN Command

Format:

RUN,*n*

or, if the data channel exerciser is run with the series,

CRUN,*n*

where *n* is the number of times (in octal) you want the entire series to be run.

The **RUN** command executes every test program in your working directory that is appropriate for your system. When your working directory is all of the directories, the appropriate programs from every directory are run.

RUN uses the equipment table to decide which programs to include in the series. Even if the equipment table is empty, however, the programs that test memory and the basic CPU always run.

An internal pass counter in each program determines the number of times that program runs before control is returned to the **RUN** command. This counter may contain any number between 1 and 7.

Because the **RUN** command runs many programs, and each program may run as many as seven times, a single series can take a long time to run. You should bear this in mind when selecting the value of *n*.

If you set *n* to 0, the series will run 64K times (the maximum number possible). On an average NOVA system, a single pass through the series takes about 30 minutes. On an average ECLIPSE system, each pass takes three or four times as long. This means that, theoretically, the command, **RUN,0**, could run for several years. You can interrupt this command, or any other, simply by pressing the STOP switch on the front panel of the CPU (or the RESET switch on soft-console systems) and then reloading DTOS.

As shown in Example 5.3, the **RUN** command displays the equipment table before it loads the first program, and it prints the run count each time the entire series has run. When the count equals *n,* DTOS regains control and issues a prompt to indicate it is ready to accept another command.

```
*RUN,1 )
DEVICES FOUND BY THE SYSTEM HOST EQUIPMENT TABLE
EQUIPMENT TABLE # 0
NAME  CODE
TTI   10
TTO   11
PTR   12
RTC   14
LPT   17
MTA   22
LOAD:
8LGCTST  REV. 05
        ::
        ::
        ::
END RUN  1          - RUNALL MODE
*
```

Example 5.3 The **RUN** command (on a NOVA 800)

The ACCEPT Command

Format:

ACCEPT

This command uses no arguments and does not allow the use of the the data channel exerciser; that is, there is no **CACCEPT** form of the command.

The **ACCEPT** command runs the same series of programs as the **RUN** command. However, it runs the entire series only once and each program in the series only once. This means that the **ACCEPT** command takes much less time to run than the **RUN** command.

If you have no idea what your system's problem is, and you want to run a series of automatically selected test programs as quickly as possible, then use the **ACCEPT** command first. Remember, however, that the **ACCEPT** command, like the **RUN** command, does not run programs requiring operator response, and only programs in the *working* directory are included in the series.

Example 5.4 illustrates the **ACCEPT** command. Note that the equipment table is displayed before the first program is loaded. When the entire series has run, DTOS regains control and issues a prompt to indicate it is ready to accept another command.

```
*ACCEPT ↵
ACCEPT
DEVICES FOUND BY THE SYSTEM HOST EQUIPMENT TABLE
EQUIPMENT TABLE # 0
NAME  CODE
TTI   10
TTO   11
PTR   12
RTC   14
LPT   17
MTA   22
LOAD:
8LGCTST REV. 05
:
etc.
*
```

Example 5.4 The **ACCEPT** command (on a NOVA 800)

Running Test Programs for a Specific Subsystem

There are two reasons for running test programs for a specific subsystem:

1. Your initial conclusion or results from running with auto-mode commands point to a specific subsystem.
2. Running in auto mode did not point to any problems in the system. Now you need to test subsystems that were *not* tested in auto mode.

You can find in several places information about the test programs that test a specific subsystem. One source is the product cross reference found in the DTOS help file (use the **HELP** command to obtain this information). Another source is the maintenance manual for the system or subsystem that you want to test.

Several DTOS commands let you run test programs for a specific subsystem. Some of these commands let you run only one program at a time, while others allow sets of programs. All these commands are explained below. Refer to "Conventions," Chapter 1, if you have any questions about how to interpret the command-line formats.

Running Sets of Programs

The **SELECT** and **REPEAT** commands allow you to select a set of programs that will test a specific subsystem. **SELECT** runs in semi-auto mode and returns control to DTOS after running your set once. **REPEAT** runs in semiauto mode and runs your set over and over again until you stop it.

The SELECT Command

Format:

SELECT *(program name[.dd],...)*

or, if the series is to be run with the data channel exerciser,

CSELECT *(program name [.dd],...)*

The optional argument *.dd* represents a nonstandard device code (in octal). Use this argument only when the device to be tested by the program you have selected has a device code assignment other than the one recommended by DGC. Remember that not all test programs accept nonstandard device codes.

The **SELECT** command accepts up to 15 programs (or groups of programs) at a time and runs them in the order in which they appear in the command line. All the programs must come from your working directory. If your working directory is all of the directories, the **SELECT** command will not work.

Though the **SELECT** command runs the set you select only once, each program in the set runs as many passes as that program's internal pass counter indicates. When all the programs in the set have run, DTOS regains control and issues a prompt to indicate that it is ready to accept another command. Example 5.5 illustrates the **SELECT** command.

```
*SELECT ECLIPSE21,ECLIPSE22,ECLIPSE23 ↵
LOAD:
ECLIPSE21 REV. 05
ECLIPSE21 IS RUNNING
MAP TYPE = MMPU1
PHYSICAL MEMORY SIZE IN 1K BLOCKS (DEC) = 96
PASS 1
:
etc.
LOAD:
ECLIPSE23 REV. 05
ECLIPSE23 IS RUNNING
MAP TYPE = MMPU1
PHYSICAL MEMORY SIZE IN 1K BLOCKS = 96
PASS 1
PASS 2
*
```

Example 5.5 The **SELECT** command (on an ECLIPSE S/130)

The REPEAT Command

Format:

REPEAT *(program name[.dd],...)*

or, if the series is to be run with with data channel exerciser,

CREPEAT *(program name[.dd],....)*

where *.dd* represents nonstandard device code (in octal).

The **REPEAT** command accepts up to 15 programs (or groups of programs) and runs them in the order in which they appear in the command line. All the programs must come from the working directory. If your working directory is all of the directories, the **REPEAT** command will not work.

This command runs the set of test programs over and over until you stop it with the STOP switch on the front panel of the CPU (or the RESET switch on soft-console systems). Each program in the set runs as many times as that program's internal pass counter indicates. After you stop the set, you must reload DTOS.

Example 5.6 shows two programs running under the **REPEAT** command. Note that the pass count (determined by each program's internal pass counter) is printed after every pass. The run count is printed each time the entire series has run.

```
*REPEAT ECLIPSE21,ECLIPSE23 )
REPEAT ECLIPSE21,ECLIPSE23
LOAD:
ECLIPSE21 REV. 04
ECLIPSE21 IS RUNNING
MAP TYPE = MMPU1
PHYSICAL MEMORY SIZE IN 1K BLOCKS (DEC) = 96
PASS 1
PASS 2
LOAD:
ECLIPSE23 REV. 05
ECLIPSE23 IS RUNNING
MAP TYPE = MMPU1
PHYSICAL MEMORY SIZE IN 1K BLOCKS (DEC) = 96
PASS 1
PASS 2
END RUN 1          -REPEAT MODE
LOAD:
ECLIPSE21 REV. 04
ECLIPSE21 IS RUNNING
MAP TYPE = MMPU1
PHYSICAL MEMORY SIZE IN 1K BLOCKS (DEC) = 96
PASS 1
:
etc.
```

Example 5.6 The **REPEAT** command (on an ECLIPSE S/130)

Running Single Programs

DTOS provides the **LOAD** and **LOAD.***n* commands to run a single test program.

The LOAD Command

This is a manual-mode command; that is, it runs the selected program over and over until you stop it with the STOP switch on the CPU's front panel (RESET switch for soft-console systems). You must reload DTOS after you halt the program.

The LOAD command has several formats:

LOAD *(program name[.dd])*

or, if the program is to be run with the data channel exerciser,

CLOAD *(program name[.dd])*

or, in special cases,

MLOAD *(program name[.dd])*

where *.dd* represents a nonstandard device code (in octal).

The special cases where the **MLOAD** format is required are as follows:

- When the system has assigned a device code other than 10/11 to TTI/TTO.
- When the system has a printer with a device code other than 17.
- When the system has a data channel line printer with a device code of 17.
- When the program is running in an IOP.

> **NOTE:** *You cannot use the* **MLOAD** *command with the data channel exerciser; that is,* **CMLOAD** *is not an acceptable form of the command.*

Example 5.7 illustrates the **LOAD** command. (If the **MLOAD** command had been used, the example would appear the same.) In this case, the pass count is printed. Whether or not it is printed depends on the switch register and the program itself.

```
*LOAD ECLIPSE23 ↵
LOAD:
ECLIPSE23 REV. 05
ECLIPSE23 IS RUNNING
MAP TYPE = MMPU1
PHYSICAL MEMORY SIZE IN 1K BLOCKS (DEC) = 96
PASS 1
PASS 2
PASS 3
:
etc.
```

Example 5.7 The **LOAD** command (on an ECLIPSE S/130)

The LOAD.*n* Command

This is a semiauto-mode command; that is, it allows you to specify the number of times the test program will run. After the program has run the number of times specified, DTOS regains control and issues a prompt to indicate it is ready to receive another command.

The format of this command is

LOAD.*n* *(program name[.dd])*

or, if the program is to be run with the data channel exerciser,

CLOAD.*n* *(program name[.dd])*

where *n* is the number of times you want the program to run and *.dd* represents a nonstandard device code (in octal).

Example 5.8 illustrates the **LOAD.*n*** command. Note that, when the pass count equals *n,* DTOS regains control and issues a prompt to indicate it is ready to accept another command.

```
*LOAD.3 ECLIPSE23 )
LOAD:
ECLIPSE23 REV. 05
ECLIPSE23 IS RUNNING
MAP TYPE = MMPU1
PHYSICAL MEMORY SIZE IN 1K BLOCKS (DEC) = 96
PASS 1
PASS 2
PASS 3
*
```

Example 5.8 The **LOAD.*n*** command (on an ECLIPSE S/130)

Verifying the Repair

Once you have found the problem in your system and repaired it, verify the repair in two steps, as follows:

1. Verify that the repair corrected the specific problem by rerunning the test programs that isolated the problem. If they run without error, the problem has been solved. If the failure recurs, restore any FRUs that you changed and start the troubleshooting process again.
2. Verify that the system is now working properly by running a complete set of system exercising programs. Do this before you use the repaired system or as soon as possible. If the system exercisers disclose a new problem, start the troubleshooting process again.

Chapter 6
Advanced Troubleshooting Techniques

The DTOS commands and features that we have discussed so far are used in most troubleshooting situations. However, in addition to these, three DTOS commands have very specialized applications: **DEBUG**, **ODT**, and **PARAMETER**. Correctly used, these commands are invaluable. But if used carelessly, or for any purposes other than those recommended, the results may be unpredictable and may actually lengthen the troubleshooting process. In general, since these commands are designed as aids for the experienced field engineer, use them only when you fully understand them.

The Symbolic Debugger

The DTOS command **DEBUG** loads a symbolic debugger program along with the test program you specify. The debugger program allows you to access and change the contents of any internal register or memory location. It can also set breakpoints in the test program.

Debugger Considerations

Because the **DEBUG** command loads the debugger program first, and then loads the test program below the debugger in memory, not all of the programs on the DTOS tape are short enough to run with the debugger program. In NOVA computer systems the debugger program begins at memory location 12000_8 when there is 16 Kwords or less of memory and at location 30000_8 when there is more than 16 Kwords. In ECLIPSE computer systems, the debugger program begins at location 30000_8.

Therefore, a test program must be shorter than 12000_8 words to run with the debugger in a NOVA computer with 16 Kwords or less. A test program must be shorter than 30000_8 words to run with the debugger in a NOVA system with more than 16 Kwords or an ECLIPSE system.

There are other reasons why a test program might not run with the debugger. First, the debugger program needs locations 50 thru 57 (octal) to set breakpoints. If a test program uses or writes over these locations, the test program will not run properly with the debugger. Also, the debugger may not run when the test is mapped.

NOTE: *These exceptions are important if you are using breakpoints to monitor a program. If you are using the debugger only to insert patches in the test program, and will not be reentering the debugger, it does not matter if the test program overwrites the debugger.*

Finally, the **DEBUG** command loads and runs the debugger program only in the CPU's memory. Since programs in the IOP directory are loaded into the IOP's memory, the debugger will not work with these programs.

A possible way to tell if a specific program will run with the debugger is to read the text file of the test program.

Uses for the Debugger

One use for the debugger is to put patches into test programs. First, you must know the exact locations to be changed, their present values, and the new values required. You then insert the patch by opening each location to be changed, entering the new value for that location, and closing the location.

NOTE: *Any changes made to a program will alter the loaded program, but will not affect the copy of that program on the DTOS tape or disk.*

Another way to use the debugger is to control a test program's execution with breakpoints. (A breakpoint stops the program at a specified point.) By placing breakpoints at appropriate places in a program, you can monitor a test program's progress and examine the operation of subtests. The debugger lets you insert up to eight different breakpoints in a program.

After a test program has halted at a breakpoint, you can use any of the debugger commands to examine or change the values in memory or in certain registers. Then you can restart the test program at the breakpoint, or at any location you wish.

The DEBUG Command

The proper format for the **DEBUG** command is

DEBUG *program name[.dd]*

where *.dd* represents a nonstandard device code (in octal).

When you finish running a program with the **DEBUG** command, you can only reaccess the DTOS monitor (without reloading) if the test program or the debugger did not write over the DTOS minimonitor. Otherwise, you must bootstrap-load the DTOS medium again. Example 6.1 shows the use of the **DEBUG** command for loading the debugger along with another program.

```
*DEBUG HDSKBLD
PROGRAM STARTS = 000200
DEBUGGER STARTS= 012000
LOAD:
HDSKBLD REV. 14
(Input debugger commands now;
there is no prompt.)
```

Example 6.1 Using the **DEBUG** command

Summary Of Debugger Commands

Table 6.1 summarizes the basic debugger commands that you need to insert patches into programs and to monitor programs.

The debugger displays a **?** or a **U** when you use an invalid command.

Command	Type	Function
adr!	Display/ change memory Cells	Open location *adr*
adr/		Open location *adr* and print contents
CR		Close open location
LF		Close open location and open next location
↑		Close open location and open previous location
$A	Display/change registers	Display contents of all accumulators
n$A		Open register n (n = 0-3 for NOVA; = 0-7 for ECLIPSE)
$B	Display/set/ breakpoints	Display location of all user program breakpoints
adr$B		Insert breakpoint at location *adr*
$D	Delete breakpoints	Delete all breakpoints
n$D		Delete breakpoint n (n = 0-7)
$P	Restart program execution	Restart execution from a breakpoint with break proceed counter set to +1 (executes breakpoint once before entering debugger program)
n$P		Restart execution from a breakpoint with break proceed counter set to counter n (n = 0-7)
n$Q		Open break proceed counter n (n = 0-7)
adr$R		Restart execution at *adr* with I/O reset
=	Select output mode (Use as terminator for input or after display of cell contents.)	Display output in numeric format
:		Display output in symbolic format
;		Display output in instruction format
←		Display output in half-word format
'		Display output in ASCII format
&		Display output in byte pointer format
$=	Set output mode	Display all future outputs in numeric format
$:		Display all future outputs in symbolic format
$;		Display all future outputs in instruction format
$←		Display all future outputs in half-word format
$'		Display all future outputs in ASCII format
$&		Display all future outputs in byte pointer format
$S	Search memory	Search all memory
adr$S	(Hit any key to stop search.)	Search all memory from location 0 to *adr*
adr<$S		Search all memory from *adr* to memory limit
adr1< *adr2*$S		Search all memory from *adr1* through *adr2*
$C	Display/change special registers	Open carry/teletype output register
$F		Open floating point registers*
$L		Open location register
$M		Open mask register
$N		Open number register
$W		Open word register

Table 6.1 Debugger Commands

*Applies only to systems with floating-point option.

NOTE: *For a more complete discussion of the debugger programs, refer to either* Symbolic Debugger User's Manual *(DGC No. 093-000044) or* ECLIPSE Symbolic Debugger *(DGC No. 093-000140). The commands listed above are a subset of the commands described in these manuals.*

The Octal Debugging Tool (ODT)

ODT is another debugging program available under DTOS. With ODT you can display and change any location in memory. ODT also permits you to set a single breakpoint in a test program.

DTOS uses two different versions of ODT: one that is internal to the test programs and one that is external to the test programs. Both versions use the same set of commands, referred to as *standard* ODT commands. These are listed in Tables 6.2 through 6.6, below.

ODT Internal to the Test Program

The internal version of ODT is a feature built into most test programs. You may access it after a program has started running. When accessed, this ODT will interrupt the program and give you a prompt when it is ready to accept commands.

Most internal ODTs use the at-sign (@) as a prompt, but the internal ODT of IPMORT (the IOP MORT program) does not use the @ prompt. It uses an exclamation point (!) instead.

To access the internal ODT, enter a CTRL O (↑ O) while the test program is running (see Example 6.2).

```
*LOAD EMORT L ↵
:
etc.
↑ 0
@
```

Example 6.2 Using ↑ O to enter the ODT

ODT External to the Test Program

The external version of ODT permits you to examine and alter a test program before you run it. The most common use for this ODT is to put patches into programs. You can also use it to set a single breakpoint in a program. You can run the external ODT with any program in the DTOS system.

Access the external ODT with the DTOS command **ODT**. This command loads the external ODT along with the specified test program. The proper format is:

ODT *(program name[.dd])*

where *.dd* represents a nonstandard device code (in octal).

The **ODT** command loads both the ODT program and the test program, and then executes the ODT program. The ODT program issues the ! prompt when it is ready to accept commands. At this point you can use ODT commands to examine and alter any location in memory or set a breakpoint in the program. After the test program begins to run, you can also access the ODT that is *internal* to that program (if it has one).

Standard ODT Commands

The ODT commands listed in Tables 6.2 thru 6.6 are used by all ODT programs to display or change the contents of memory locations (cells).

See "Conventions," Chapter 1 for an explanation of the conventions that are used in the ODT command summaries.

Input	Memory Cell Opened
adr/	Cell at location *adr*
./	Last cell opened
.+*adr*/	Last cell opened plus *adr*
.-*adr*/	Last cell opened minus *adr*

Table 6.2 To open a memory cell and display its contents

NOTE: *The* adr *can be a single octal number x or an octal expression n* $(n = x_1 \pm x_2 \pm ...)$.

Input	Next Cell Opened
LF	Following cell
↑ (up arrow)	Preceding cell
/	Cell pointed to by contents
+*adr*/	Cell pointed to by contents plus *adr*
-*adr*/	Cell pointed to by contents minus *adr*

Table 6.3 To close any cell without modification, open another cell, and display its contents

Input*	Result
x CR (LF or ↑)	Substitute octal number *x* for old contents of cell.
n CR (LF or ↑)	Substitute octal expression *n* $(n = x_1 \pm x_2 \pm ...)$ for old contents of cell.
+*n* CR (LF or ↑)	Add octal expression *n* to old contents of cell.
-*n* CR (LF or ↑)	Subtract octal expression *n* from old contents of cell.
. CR (LF or ↑)	Substitute address of cell for old contents of cell.
.+*n* CR (LF or ↑)	Substitute address of cell plus octal expression *n* for old contents of cell.
.-*n* CR (LF or ↑)	Substitute address of cell minus octal expression *n* for old contents of cell.

Table 6.4 To modify the contents of a open cell

*Typing LF after an octal input closes the cell with modification and opens the following cell. Typing ↑ (up arrow) closes the cell with modification and opens the preceding cell.

Input	Internal Register Cell Opened
0A - 3A	Accumulator 0 thru 3
4A	PC of next instruction to be executed after **P** command
5A	CPU and TTO status
	BIT MEANING
	15 Status of TTO done flag
	14 Status of interrupts (ION flag)
	13 Status of carry bit
6A	Address of location having break point (if any)
7A	Instruction at break point location

Table 6.5 To open an internal register cell and display its contents

Input	Result
P	Exits ODT subroutine and restarts diagnostic program at location pointed to by value in program register accessed with 4A ODT command (P = proceed).
*adr*R	Exits ODT subroutine and starts diagnostic program at location *adr* after I/O Reset.
CR	Closes cell without modification when used alone. Closes cell with modification when pressed after typing new contents.
K	Kills string typed so far. ODT responds with ?, closes cell without modification, and prompts with @ or !.
DEL or *RUBOUT*	Deletes right-most digit of either old contents of cell or of new contents being typed.
=	Calculates and displays result of octal expression *n*. The octal number displayed is either the address (*adr*) of next cell location to be opened or the new contents of opened cell.
*adr*B	Inserts break point at location *adr*. (ODT allows only one break point at any time and it deletes break point every time you access ODT.)
D	Deletes current break point.

Table 6.6 To perform miscellaneous commands

Nonstandard ODT Commands

The IOP ODT commands listed in Tables 6.7 and 6.8 are only on the DTOS media for the ECLIPSE M/600. These commands are used mostly by the ODT external to test programs. However, a few test programs use these commands in their internal ODT program.

Input*	Equivalent Console Function	Result
0I thru 3I	Examine AC 0-3	Examine IOP accumulators 0 thru 3.
4I thru 7I	Deposit AC 0-3	Deposit data from IOP hardware switch register into IOP accumulators 0 thru 3.
10I	Deposit	Deposit contents of IOP hardware switch register into IOP location pointed to by current PC.
11I	Deposit next	Deposit contents of IOP hardware switch register into IOP location pointed to by current PC+1.
12I	Examine	Examine location specified by value in IOP hardware switch register.
13I	Examine next	Examine location pointed to by current PC+1.
14I	Start	Start IOP at location specified by value in IOP switch register.
15I	Execute	Execute contents of IOP hardware switch register.
16I	Program load	Load program from device specified by IOP hardware switch register.
17I	Continue	Start IOP at location pointed to by current PC.
20I	Instr. step	Instruction-step IOP.
21I	Reset	System-reset IOP.
22I	Stop	Halt IOP.
23I	----	No operation.
24I	----	Examine PC and Carry (bit 0) for IOP.
25I	----	Examine IOP console buffer to determine results of last console operation.
26I	----	Examine IOP address buffer to determine last IOP memory reference.
27I XXXXX	Load IOP hardware switch register	Load IOP hardware switch register with desired octal value XXXXX. For example: 27I 42653 loads 42653 into IOP hardware switch register.

Table 6.7 Console-function commands

*Note that the letter I following the number inputs stands for IOP.

Input	Result
*dd*C	Select (or change) IOP to perform all subsequent IOP ODT commands. *dd* represents device code (octal) of selected IOP.
$	Open location of IOP flag for display or change. This flag determines which processor uses ODT commands *adr*B, D, and P as follows. IOP if flag = 177777 (or any number not equal to zero) HOST if flag = 000000 To change flag, enter 0 for HOST or -1 for IOP followed by CR. To close location without change, enter CR.
*xxxxx*M *yyyyy* ↲ *zzzzz* ↲	Move *zzzzz* (octal) words from HOST memory location *xxxxx* (octal) to IOP memory location *yyyyy* (octal). (Leading zeroes are not needed.) For example: 2000M Move from 2000 (HOST) 4000 to 4000 (IOP) 100 100 words

Table 6.8 Other commands

The PARAMETER Command

The **PARAMETER** command is a specialized command that allows you to modify certain hardware parameters. The parameters that may be modified vary from system to system.

> **WARNING:** *Use of the* **PARAMETER** *command is explained in the maintenance manuals of the systems that use the command. If the maintenance manual for your system does* not *include a description of this command, DO NOT use it. While the* **PARAMETER** *command appears to work on some systems that do not use it, the results are unpredictable, and may cause spurious error reports.*

Appendix A
Bootstrap Test (TESTOK)

New releases of the DTOS media include a bootstrap-loading test that runs as part of the bootstrap-loading procedure. The test quickly checks the basic functions of the system hardware (excluding peripherals). When the test runs successfully, it displays the word TESTOK on the system operator's console. Otherwise, it halts the CPU before displaying the entire word. Thus, the portion of the displayed TESTOK word indicates the probable areas of failure, as shown in Table A.1.

Message	Meaning
None *	When no part of the word displays (after 30 seconds), the following are probable areas of failure: A JMP, JSR, or STA instruction An indirect JMP or JSR TTO busy could not be cleared
T	Indicates that either an arithmetic/logic or an indexed memory-reference instruction failed. The following are the probable areas of failure: First: CPU board Second: memory subsystem
TE	Indicates that the test could not size memory. The most probable failure is the memory subsystem.
TES	Indicates that the data=address subtest failed. The most probable failure is the memory subsystem.
TEST	Indicates that the data=complement-of-address subtest failed. The most probable failure is the memory subsystem.
TESTO	Indicates that the memory-pattern subtest failed. The most probable failure is the memory subsystem.
TESTOK	Indicates that the entire program ran successflly.

Table A.1 TESTOK message interpretation

*If your system operator's console is on device codes other than 10 and 11, see next page.

When TESTOK Is Not Displayed

In systems with the system operator's console on device codes other than 10 and 11, the bootstrap test runs, but the word TESTOK is not displayed. If the test finds no errors, the questions about the type of CPU and the DTOS prompt (*) appear on the system operator's console after an interval of approximately 90 seconds (after DTOS loads and runs the initializing programs).

If the test detects an error, it halts the CPU and displays the PC contents in either of two places. In a system with a soft console, the PC contents appear on the system operator's console followed by the ODT prompt. In other systems, the PC contents appear in the address lights of the CPU's front panel.

You can find the probable areas of failure simply by matching the PC contents to a value in Table A.2.

Contents of PC at Halt Time for Tape/Disk Versions		Meaning
Tape	Disk	
130	514	The following are probable areas of failure:
176	531	A JMP, JSR, or STA instructions
		An indirect JMP or JSR
		TTO busy could not be cleared
204	537	Indicates that either an arithmetic/logic or an indexed memory-reference instruction failed. The following are the probable areas of failure:
212	545	
214	547	
217	552	First: CPU board
223	556	Second: memory subsystem
225	560	
227	562	
231	564	
234	567	
236	571	
No halts during memory sizing.		Indicates that the test could not size memory. The most probable failure is the memory subsystem.
277	632	Indicates that the data = address subtest failed. The most probable failure is the memory subsystem.
301	634	
323	713	Indicates that the data = complement-of-address subtest failed. The most probable failure is the memory subsystem.
325	715	
346	736	Indicates that the memory-pattern subtest failed. The most probable failure is the memory subsystem.
356	746	
No halts, no errors.		Indicates that the entire program ran successfully.

Table A.2 Interpretation of PC contents

Memory Tests Within the Bootstrap Test

Table A.3 is intended to help you analyze a memory failure during the bootstrap test. When TES, TEST, or TESTO are the only parts of the TESTOK word that are displayed, look in Table A.3 to determine the failing adddress and the good and bad data. All three memory tests run with memory unmapped (first 32K only).

Test Name	ACO	AC1	AC2
Data=address (TES)	Bad data	---	Failng address (good data)
Data=complement of address (TEST)	Complement of bad data	---	Failing address (complement this number to get the good data)
Memory pattern (TESTO)	Good pattern	Bad pattern	Failing address

Table A.3 Value of ACs for memory tests

Appendix B
DTOS Error Messages

Following is a list of the error messages you may receive while running test programs under the DTOS system along with an explanation of what each one means. If you are not an experienced field engineer, you may not be able to correct the problem indicated by a particular message. You should know what the message means, however, so you can report on the situation to the next level of technical assistance available to you.

During Tape Bootstrap

Program: TPBOOT

CAN'T READ THE TAPE

The tape being used may not be the DTOS tape or it may be bad.

BAD TAPE

Either the tape drive is malfunctioning or the media is bad.

Program: HIIOMOD

IOP TIME OUT ERROR, DEVICE CODE XX

The specified IOP did not complete its sizing routine within the prescribed time limit. The HIIOMOD program will continue past this error, but once you receive a prompt, look for a hard failure in the specified IOP. (XX represents the IOP's device code.)

Program: HISIZER

IOP SIZER TIMED OUT IOP # XX

The specified IOP did not complete its sizing routine within the prescribed time limit. The HISIZER program will continue past this error, but once you receive a prompt, look for a hard failure in the specified IOP. (XX represents the IOP's device code.)

During DTOS Execution

Program: HIMON

??

You entered an illegal command. Check for correct spelling and use of punctuation marks.

ILLEGAL NAME; XXXX

DTOS did not recognize one of the device names (XXXX) used with an **ADD**, **DELETE**, or **ASSIGN** command.

DEVICE CODE NUMBER ERROR

DTOS did not recognize one of the device codes used with an **ADD**, **DELETE**, or **ASSIGN** command.

NO SUCH PROGRAM

The specifed program is not in the working directory.

PROGRAM LARGER THAN XXXXXX

The specified program requires more space than is available in memory. (XXXXXX represents the amount of memory available.)

SYSTEM CONFLICT, CANNOT FIND PROGRAM

The specified program cannot be located on the tape. This error message indicates either a bad tape or a positioning error.

FATAL STATUS= XXXXXX

XXXXXX is a 16-bit status word indicating the reason for the failure during magnetic tape operation. For the specific reason, refer to "Magnetic Tape Units, Error Checking," *Peripherals, Programmer's Reference* (DGC No. 014-000632).

SPACING ERROR, FATAL

An error has occurred in an attempt to locate a tape record. The specific cause of the failure can be determined from the status error message.

INPUT OVERFLOW

The terminal's input buffer cannot handle the number of characters you entered.

INSUFFICIENT MEMORY FOR DEBUGGER

There is not enough memory space available to load the debugger program.

NO ACTIVE IOP'S SPECIFIED, CHECK PARAMETER COMMAND

No IOP is activated (in systems with one or more I/O processors). Use the **PARAMETER** command to activate the one desired.

THE SPECIFIED COMMAND WILL NOT RUN IN "ALL" MODE

The command you specified will not run when the ALL directory is the working directory.

During Disk Building

Program: HDSKBLD

PROGRAM NOT FOUND: XXXXXX

The specified system program cannot be located on the tape. (XXXXXX represents the program name.)

DISK SEEK ERROR

The seek function of the disk drive is malfunctioning.

DISK RECAL ERROR

A recalibration error has occurred. Repeat the recalibration procedure.

DISK RD/WR ERROR, DOA= XXXXXX

XXXXXX represents the accumulator for the **DOA** that was being performed at the time the error occurred. Refer to the maintenance manual for the disk to determine if the error was a read error or a write error.

DATA COMPARE ERROR

```
GOOD        BAD
XXXXXX      XXXXXX
```

The data on the disk does not match the data from the tape. The GOOD column represents the information on the tape and the BAD column represents the same information on the disk.

SYSTEM CRASHED

The system is no longer functioning. Reload DTOS.

FATAL STATUS = XXXXXX

XXXXXX is a 16-bit status word indicating the reason for the failure during magnetic tape operation. For the specific reason, refer to "Magnetic Tape Units, Error Checking," *Peripherals, Programmer's Reference* (DGC No. 014-000632).

SPACING ERROR, FATAL

An error has occurred in an attempt to locate a tape record. See the status error message for the specific cause for the failure.

SYSTEM PROGRAM ON TAPE DOES NOT MATCH THAT ON DISK

An error occurred when the program was written from tape to disk. HDSKBLD found the error during verification of the disk copy.

PROGRAM LENGTH ON TAPE NOT EQUAL TO THAT ON DISK

When a second or third tape is added to a disk, HDSKBLD first verifies that the system programs on the tape match those already on the disk. This message indicates that the verification failed. Check the revision numbers on each tape to make sure they match. (You cannot put tapes with different revision numbers on a single disk.)

BUILDING ABORTED. MULTIPLE DTOS TAPES CAN ONLY BE BUILT ON THE SAME DISK AND, UNFORTUNATELY, YOU HAVE REACHED THE END OF THIS DISK WITHOUT WRITING ALL PROGRAMS

You have run out of room on the disk while adding a second or third tape to the disk.

BAD SECTOR FOUND UNABLE TO CONTINUE

This disk pack contains a bad sector. Use a different disk pack.

THE DISK IS WRITE DISABLED, PLEASE ENABLE

The disk may be write-disabled for a good reason. Check that the correct disk is in the drive and write enable it.

DRIVE STATUS ERROR OCCURRED, DIB= XXXXXX

An error occurred in a write operation due to a disk status error. XXXXXX represents the accumulator for the **DIB** that was being performed at the time the error occurred.

DATA TRANSFER STATUS ERROR OCCURRED, DIA= XXXXXX

A disk status error caused an error in a read operation. XXXXXX represents the accumulator for the **DIA** that was being performed at the time the error occurred.

During DDOS Execution

??

You entered an illegal command. Check for correct spelling and use of punctuation marks.

Program: HDSKMON

ILLEGAL NAME

DTOS did not recognize one of the device names used with an **ADD**, **DELETE**, or **ASSIGN** command.

DEVICE CODE NUMBER ERROR

DTOS did not recognize one of the device codes used with an **ADD**, **DELETE**, or **ASSIGN** command.

NO SUCH PROGRAM

The specified program is not in the working directory.

PROGRAM IS LARGER THAN XXXXXX

The specified program requires more space than is available in memory. (XXXXXX represents the size of memory available.)

INPUT OVERFLOW

The terminal's input buffer cannot handle the number of characters entered. (Your input line was too long.)

INSUFFICIENT MEMORY FOR THE DEBUGGER

There is not enough memory space available to load the debugger program.

SPECIFIED TAPE NUMBER IS INACTIVE

The tape number you requested with the **TAPE** command does not exist on this disk.

NO ACTIVE IOP'S SPECIFIED, CHECK PARAMETER COMMAND

No IOP is activated (in systems with one or more I/O processors). Use the **PARAMETER** command to activate the one desired.

THE SPECIFIED COMMAND WILL NOT RUN IN "All" MODE

The command you specified will not run when all the directories are the working directory.

During Data Channel Exerciser (CAT) Execution

DATA COMPARE ERROR

```
GOOD      BAD       ADDRESS
XXXXXX    XXXXXX    XXXXXX
```

This message indicates that data channel problems exist. (XXXXXX represents good and bad data and an address.)

DKP ERROR STATUS= XXXXXX

XXXXXX is a 16-bit error status word. Consult the maintenance manual for the disk for a specific definition.

Appendix C
DTOS System Generation for Disk (DDOS)

You can write DTOS tapes onto disk or diskettes to build a Diagnostic Disk Operating System (DDOS). The procedure given below does not apply to microNOVA DTOS because it is only supplied on diskette. Table C.1 lists the programs that must be on the DTOS tape to successfully generate DDOS.

Program Type	Program Name
Disk builder	HDSKBLD
Disk bootstrap	HDSKBOOT 0 and HDSKBOOT 1
Disk monitor	HDSKMON
Disk minimonitor	HDSKMINI 0 and HDSKMINI 1
Disk overlay drivers	HDSKFILES
Help program	HELP
I/O module size and load program	IOMOD
I/O bus sizer	IOSIZER
ODT program	ODT
Data channel exercisers (CATs)	CAT 0 thru CAT 5
NOVA debugger	DEBUG III
ECLIPSE debugger	EDEB III

Table C.1 Necessary DTOS tape-resident programs for disk building

Procedure To Build Disk

After bootstrap-loading the DTOS system, load the disk pack and make sure the drive is on line. You are now ready to build the DDOS disk by following the procedure outlined below. The portions labeled INPUT are what you type in (do not type the comments).

DISPLAY: * (prompt for DTOS)

INPUT: LOAD HDSKBLD

The HOST directory must be the working directory when you issue this command.

DISPLAY: DISK TYPE FOR DEVICE ON WHICH DISK WILL BE BOOTED?

INPUT: XXXX

XXXX is one of the following disk model numbers:

4047 4057 4234 6038/39 6060/61/67
4048 4231 6030 6045/50 6070

DISPLAY: UNIT # OF DRIVE ON WHICH DISK IS TO BE BUILT?

INPUT: X

X is the unit number of the disk drive used. Enter 0 for disk drives with no unit-select switches.

DISPLAY: DEVICE CODE OF DRIVE ON WHICH DISK IS TO BE BUILT?

INPUT: XX

XX equals one of the following numbers:

27 (primary) or 67 (secondary) for 6060/61/67 disk drives,

33 (primary) or 73 (secondary) for all other disk drives.

DISPLAY: ENTER REV. NUMBER OF DTOS TAPE (X.X) AND A (CR):

INPUT: X.X

You can find this number by looking on the console when the DTOS tape successfully loads. You must use the complete number.

DISPLAY: ENTER A DISK ID NUMBER

INPUT: XXXXX

XXXXX is a number you choose to identify this particular disk. The DTOS system does not use this number in any way, but displays it after a disk boot for disk identification purposes.

DISPLAY: DO YOU WANT TO COPY THE ENTIRE TAPE ONTO THE DISK OR SELECT INDIVIDUAL PROGRAMS?
ENTER: 1 (CR) TO SELECT PROGRAMS
0 (CR) TO COPY ENTIRE TAPE

INPUT: 1 or 0

Input 1 when you want to select the diagnostic and reliability programs for building on the disk.

Input 0 when you want all the programs put onto the disk.

DISPLAY (if 0): BUILDING DDOS PACK: 1 DISK DIRECTORY: *n*

NOTE: *Octal number* n *above, which represents current disk directory being built, corresponds to number of tape used during building. For example, 0 = first tape, 1 = second tape, etc.. This is the number you select disk directory with during the* **TAPE** *command.*

DISPLAY (if 1):

BUILDING DDOS PACK: 1 DISK DIRECTORY:
LOADING SYSTEM PROGRAMS ON DISK. PLEASE WAIT..

SELECT PROGRAMS NOW. (ENTER "HELP" FOR INSTRUCTIONS)
$)

INPUT: When you receive the $) prompt, input any selective building commands needed to select only the programs that you want on the disk. You exit the selective disk-building mode by inputting the **BUILD** command.

After DDOS is successfully generated (upon completion of tape when building single pack only):

DISPLAY:

DO YOU DESIRE TO INCORPORATE ADDITIONAL DTOS TAPES?
ENTER: 1 (CR) IF YES
        0 (CR) IF NO

INPUT: 1 or 0

DISPLAY (if 1): MOUNT NEXT DTOS TAPE, HIT ANY KEY

DO YOU WANT TO COPY ENTIRE TAPE ONTO THE DISK OR SELECT INDIVIDUAL PROGRAMS?

(See previous page for possible inputs in response to this question.)

DISPLAY (if 0): DDOS PACK BUILDING FINISHED

DO YOU WISH TO BUILD ANOTHER DDOS DISK PACK?
ENTER: 1 (CR) IF YES
        0 (CR) IF NO

INPUT: 1 or 0

DISPLAY (if 1):

MOUNT APPROPRIATE DTOS TAPE AND SCRATCH PACK FOR NEXT DDOS BUILD
STRIKE ANY KEY WHEN READY

DISK TYPE?

(See beginning of this section for possible inputs in response to this question.)

DISPLAY (if 0): * (DTOS prompt returns.)

For DDOS generation on diskette, when diskette runs out of space (for first DTOS tape only):

DISPLAY:	MOUNT ANOTHER DISK PACK & HIT ANY KEY ON TTI
	BUILDING DDOS PACK 2
	DDOS NOW READY FOR USE

Selective Builder

The selective builder is used when you want to select the programs that will be built on the disk.

Selective Builder Commands

Table C.1 gives a list by category, of the commands used when the selective disk builder prompts with the $). Table C.2 lists the commands alphabetically along with a description of each.

Display Directory	Select Directory	Change List	Control Printer	Get Help	End Session
DIR	HINT HIOP HOST IOP	COPY DELETE FULL	PRINT	HELP	BUILD

Table C.1 Selective builder commands

Input	Command Description
BUILD	Ends selection session. Only diagnostic programs in selection list are copied from tape to disk. Required system programs are automatically copied to disk.
COPY *(program name)* *	Adds the program or programs specified to disk build selection list. Displays following message: ADDED PROGRAM = *Program Name* REV: X
DELETE *(program name)* *	Deletes program or programs specified from disk build selection list. Displays following message: DELETED PROGRAM = *Program Name* REV: X
DIR	Displays name and contents of current directory (XXX) in following format: CONTENTS OF DIRECTORY XXX *Program name* REV: X SP *Program name* REV: X SP *Program name* REV: X ++ *Program name* REV: X etc.... Labels next to program mnemonic have following meaning concerning disk building: SP required system program ++ program selected (no label) program not selected
FULL	Clears disk build selection list, and copies all tape programs from all directories. Displays following message: ANY PROGRAM SELECTIONS ARE BEING DISCARDED. COPYING ALL TAPE PROGRAMS INTO THE CURRENT DISK DIRECTORY
HELP	Displays help file which gives a description of each selective builder command.
HINT	Selects INTERFACE directory (HINT) as working directory. Displays following message: THE NEW DIRECTORY IS: HINT
HIOP	Selects HOST and I/O PROCESSOR directory (HIOP) as working directory. Displays following message: THE NEW DIRECTORY IS: HIOP
HOST	Selects HOST directory as working directory. Displays following message: THE NEW DIRECTORY IS: HOST
IOP	Selects I/O PROCESSOR directory (IOP) as working directory. Displays following message: THE NEW DIRECTORY IS: IOP
PRINT	Enables or disables output to lineprinter. Displays following message: LINE PRINTER CONTROL. ENTER: "0" TO DISABLE "1" TO ENABLE REPLY:

Table C.2 Commands listed alphabetically

*For the various formats you can use for the argument *program name*, see "Conventions," Chapter 1. *ALL* used as an argument instead of program names will cause all of the programs in the working directory to be included in the command.

Selective Builder Messages

ALREADY EXISTS = Program Name REV: X

Program is used as argument to **COPY** command, but is already in selection list.

NEVER SELECTED = Program Name REV: X

Program is used as argument to **DELETE** command, but is not in selection list.

UNKNOWN PROGRAM = Program Name

Program is used as argument to **COPY** command, but cannot be found in directory.

SYSTEM PROGRAM = Program Name

Program is system program and you cannot select or delete it.

ILLEGAL SERIES TERMINATOR = Program Name

Program is used as last program in *program name-program name* argument to **COPY** or **DELETE** commands, but is not valid terminator because it cannot be found in directory.

PROG-PROG FORMAT ERROR. PROG OUT OF ORDER: Program Name

Program is used as last program in *program name-program name* argument to **COPY** or **DELETE** commands, but does not follow first *program name* in directory.

INPUT OVERFLOW: RE-ENTER INPUT

Input is aborted because there are too many characters for input.

PROGRAMS FOR THIS DIRECTORY NOT FOUND

The working directory does not contain any programs.

BUILD TABLE OVERFLOW ...

Selection list contains more than 292 programs. Follow instruction in rest of message.

?

Last input was not valid.

Appendix D
An Overview of DTOS Internals

This appendix contains an overview of the internal structure of DTOS. The overview is included simply as a reference for the advanced field engineer. You do not need to read or understand this material in order to use DTOS effectively.

The Parts Of The Diagnostic Operating System

DTOS is made up of five main system programs and several utility programs.

System Programs

The five main programs are as follows:

- HIMON (HDSKMON in DDOS) - This program, commonly called the DTOS monitor, provides a way for the operator to issue commands to the DTOS system. It also handles the control and program loading functions.
- HIMINI (HDSKMINI 0 and 1 in DDOS) - The HIMINI program is used to store system information, load the final 4K block of a program, start the program, and return control to the monitor when the program has finished running. This program, commonly called the minimonitor, resides in the top 200_8 locations of logical memory. The DTOS monitor transfers control to HIMINI after loading all but the first 4K of a program. HIMINI then loads this block, sets up the program's *program control block*, using information supplied by the monitor, and starts the program.

  Programs return control to the DTOS monitor by starting execution of HIMINI at the last location in logical memory. This causes the HIMINI program to reload the DTOS monitor, HIMON.
- IPMINI - This program is used to store system information, start programs in an IOP, and return control to the monitor when a program has finished running. The IPMINI program resides in each IOP's top 200_8 logical memory locations. The DTOS monitor transfers control to IPMINI after loading a program into the I/O processor. IPMINI then sets up the program's *program control block*, using information supplied by the monitor, and starts the program.

  Programs return control to the DTOS monitor by starting execution of IPMINI at the last location in the IOP's logical memory. This causes the IPMINI program to restart the DTOS monitor by executing a halt instruction sequence.

- IOMOD - This program is used to size and load the I/O modules that will handle all TTI, TTO and LPT input/output. When run by the DTOS monitor, the program sizes the CPU for TTI, TTO and LPT and loads the proper driver code and I/O modules. IOMOD then repeats this process for each IOP in the system.
- IOSIZER - This program is used to size each processor's I/O bus for all available devices. The IOSIZER uses a table of tests containing a single entry for each possible device code. When there is no test for the devices on a particular device code, the table contains a zero.

  For each device it finds, IOSIZER sets a corresponding bit in the device code block. The device code block is a temporary storage area set up by the IOSIZER program. After running the last test, IOSIZER moves the information in the device code block into the minimonitor.

  IOSIZER uses the IOP configuration word from the minimonitor to determine whether or not the system contains any IOPs, and sends a copy of the sizing routine to each IOP found. Each IOP's I/O bus is sized, and the results are stored in that IOP's minimonitor.

Data Channel Exerciser

The data channel exercisers (CAT*n*, where *n* = 0, 1, 2, etc. or A, B, C, etc.) are the DTOS utility programs that exercise the data channel (DCH) with background activity while test programs are running. The DTOS monitor selects the appropriate CAT when you load test programs with a DTOS command prefixed with C.

The data channel exerciser is made up of several different tests listed below:

Test 0 Writes 32 Kwords of memory to a disk file or an IOBD.

Test 1 Does a series of write/read transfers to a disk file, or to an IOBD using patterns of alternating zeroes and ones.

Test 2 Does a series of write/read transfers to a disk file or an IOBD using increment/swap patterns. It also checks valid data after each transfer.

Test 3 Runs only with an IOBD. Uses the hardware random number generator on the I/O test board to generate a buffer of random numbers. This buffer is then checked against numbers generated by a software random number routine, which uses the same algorithm and the same initial seed.

Each of the data channel exercisers executes an *I/O Reset* when it is first started. It then assumes complete control of the disk or I/O test board, and prevents any other program from using it.

The data channel exerciser does not save the machine state when the test program begins running. However, it does save the machine state each time it returns control to the program.

While the data channel exerciser is running, you must be careful not to cause status-change interrupts. The data channel exerciser will pass these on to the program, and the results will be unpredictable. In other words, do not turn peripheral devices (especially the disks) on or off while the data channel exerciser is running.

One pass of the data channel exerciser takes approximately 3 minutes.

I/O Modules

The I/O modules are a set of subroutines that a diagnostic program may call to provide its I/O device handling. These modules provide support for I/O devices (TTI, TTO, LPT) not supported by the standard packages, such as operator consoles using device code 50/51 and data channel line printers.

The I/O modules are loaded (by the program IOMOD) into upper memory during the initial loading procedure. If at any time the I/O modules are overwritten, you may reinstall them by running IOMOD with the **LOAD.*n*** command (**LOAD.1 IOMOD**).

Diagnostic programs use the I/O modules when loaded with any DTOS command except **LOAD**, **DEBUG** and **ODT**. The I/O modules are saved intact by the commands that use them, but may be destroyed by the **LOAD, DEBUG** or **ODT** commands.

When running a program on a system whose I/O devices (TTI, TTO, LPT) are supported only by the I/O modules, you must load the program with the **MLOAD** command instead of the **LOAD** command.

DTOS Loading Sequence

The following loading sequence occurs when you bootstrap-load the DTOS system from tape:

1. TPBOOT, the bootstrap-load program, loads the TESTOK program.
2. The TESTOK program quickly tests the basic functions of the hardware system and then loads HIMON.
3. HIMON installs HIMINI at the top of memory in the CPU and IPMINI at the top of memory in each IOP. If the system has no IOPs, DTOS does not load IPMINI.
4. HIMON loads IOMOD, which installs the I/O modules, and reloads the monitor. It also loads IOSIZER, which sizes the system and reloads the monitor.

At the end of this process the monitor (HIMON) issues a prompt (*) to show that DTOS is ready to accept commands.

DTOS in Memory

Once the various parts of DTOS are loaded, they reside in the memory locations indicated in Figure D.1.

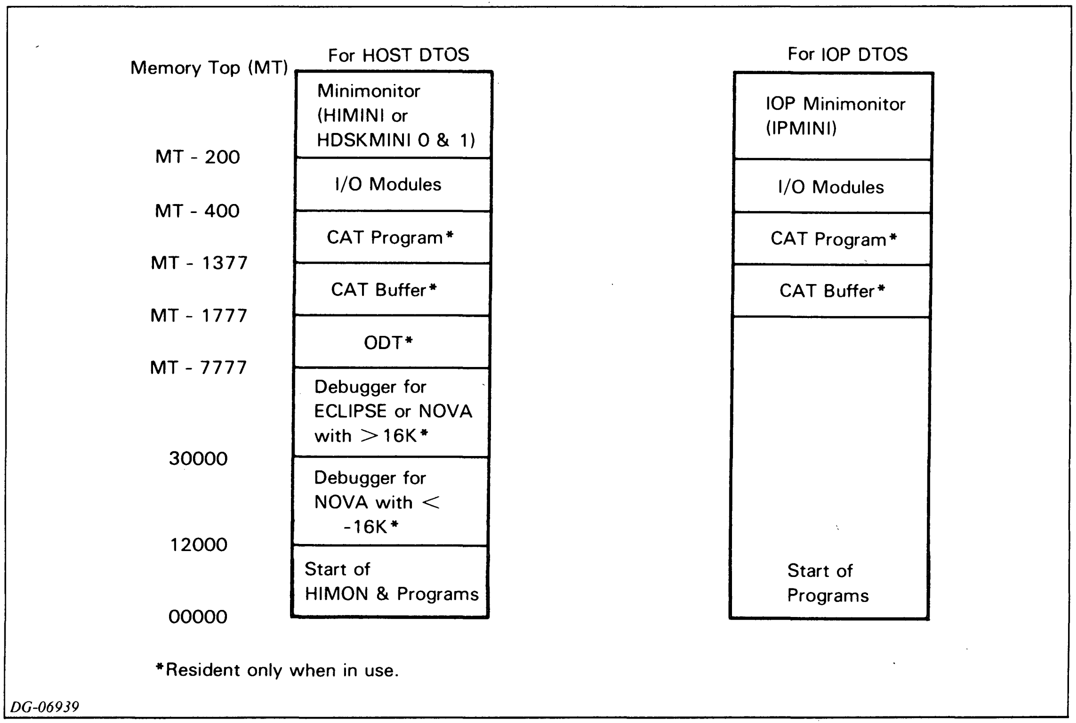

Figure D.1 Structure of DTOS program in memory

Minimonitor

The information stored in the minimonitor is for HIMINI (or HDSKMINI 0 and 1) and IPMINI. Table D.1 shows the contents of the minimonitor.

Location*	Contents
L777: X7777	Exit to monitor
L776: X7776	Schedule pointer for **SELECT/REPEAT**
L775: X7775	# passes to run this program - also pass/fail switch
X7774	Address of L777
X7773	Entry from monitor
X7772	Device code this run
X7771	Run number for auto mode and **REPEAT** command
X7770	CAT switch
X7767	Tape-loader handler (with next two words)
X7766	
X7665	
X7664	CAT device code/record number
X7663	CAT device type
X7662	Run count/command number
X7661	Used by auto-mode commands
X7660	Minimonitor length
X7657	System disk type
X7656	System device code
X7655	Switch register
X7654	CPU type and subtype
X7653	System equipment table (with next three words)
X7652	
X7651	
X7650	
X7647	Program start address
X7646	I/O module offset
X7645	IOP device code
X7644	IOP configuration
X7643	Current directory
X7642	Active IOP word
X7641	Power margin word
X7640	Offset to the schedule pointer
X7637-X7600	Schedule for **SELECT** and **REPEAT** commands. Number of entries is variable but cannot extend beyond location X7600. Entries are as follows: Program size, error switch, record # Device code, starting address Program size, error switch, record # Device code, starting address Etc. 0 (end marker for **LOAD, SELECT, RUN, ACCEPT** and **REPEAT** commands) -1 (end marker for auto mode)

Table D.1 Contents of the Minimonitor

*The X in the first digit of the location value represents an octal number from 0 to 7. Its value depends on the maximum size of unmapped memory in your system, as follows:

0 = 4K	4 = 20K
1 = 8K	5 = 24K
2 = 12K	6 = 28K
3 = 16K	7 = 32K

Communication Between DTOS and Diagnostic Programs

In order to control the programs it runs, DTOS must be able to communicate with these programs. Thus, each test program contains two data blocks that hold the information that must be transmitted between DTOS and the program.

One block contains information that is passed *to the program* by the DTOS monitor (to control the program). This is the program control block (occasionally called the EGGS block).

The other block contains information that is passed *to the DTOS monitor* by the test program (giving DTOS information about the program). This is the directory identification block (occasionally called the DIRT block).

Program Control Block

Table D.2 shows the contents of the program control block.

Word No	Name	Description
1	DTOS command number	This word has a variety of values depending on how the test program was loaded: 000000 if loaded in manual or debug mode. Non zero if loaded in auto or semiauto mode. 000001 if loaded with the **MLOAD** command. 000021 if loaded with the **HELP** command. 000022 if loaded with the **HELP ALL** command.
2	I/O module offset, device code	This word contains the I/O module offset and the device code of the device to be tested. Multiply the size of an I/O module block (200_8) by the offset value to determine the memory area occupied by the I/O modules and the minimonitor combined. For example, if this word is 233, the 2 is the offset, and 2 X 200_8 = 400_8 memory locations. The device code may be either the primary or secondary device code for this test, or the device code defined with the *.dd* extension in a command. For example, the 33 in the number 233 mentioned above is the device code.
3	CAT program switch	The DTOS monitor sets this location to -1 (177777) if the CAT program was loaded along with the test program.
4	Pass count	This internal pass count word tells the test program the number of passes it should run. This number has a variety of values depending on how the test program was loaded. If loaded with the **ACCEPT** command, the value is 000001. If loaded with any other auto or semiauto command, the value is the same as the value of Word 8 (internal pass count, bits 13-15) in the test program's DIRT block. If loaded with the **LOAD.***n* command, the value is set to the value of *n* (octal) in the command. If loaded in the manual mode, this word is not used, since the test program loops indefinitely.
5	Return	This location contains the starting address of the minimonitor. When the test program runs in auto or semiauto mode (except with the commands **TOPDOWN**, **CTOPDOWN**, **REPEAT**, and **CREPEAT**, it returns to the minimonitor (which restores the DTOS monitor) by jumping through this location. In systems with 32Kwords or more of memory, this location contains 077777. In systems with less than 32Kwords, it contains the value for the top of memory.
6	Switch register	This word contains the default value for the test program's switch register (switches 0-15). This is the word that you complement with either the DTOS **SWREG** command or the keyboard keys 0-9 and A-F.
7	Optional extra switch register	This word contains an optional extra switch register (switches 16-31) that the diagnostic programmer can define when writing a test program. Switches 16-31 cannot be changed with the DTOS **SWREG** command, but they can be complemented with the keyboard keys G-V when the test program is running. Since this switch register is part of the test program itself, the DTOS monitor does not pass a value to this register.

Table D.2 Contents of the program control block

Directory Identification Block

The directory identification block contains information that the DTOS monitor can use to identify the specific systems or devices the program will run on. Other information in this data block includes: the program's internal pass counter, the name and revision number of the program, and the program's starting address.

The contents of the directory identification block consist of the 21 words described below.

Words 0 through 5

These words comprise the program name and revision number. The program name is encoded in the first nine ASCII characters and the revision number is encoded in the tenth and eleventh ASCII characters. The twelfth ASCII character is zero.

Word 6

Word 6 is a zero word reserved for use by the DTOS system for tape record information.

Word 7

This word contains the starting address. The low byte is the normal starting address for the program. The high byte is an optional second starting address used only by some programs when loaded from the HIOP directory.

Word 8

This word comprises the CPU type and pass count. The number encoded in bits 0 thru 12 can be one of the following:

0001010000000 Refer to the values encoded in words 16 through 19 for CPU type and subtype.

0000000000000 The program runs on any CPU type and subtype. The words giving CPU type and subtype (16 through 19) are not needed and may be set to zero.

The number encoded in bits 13 - 15 is the number of passes the program will run in auto mode.

If this program is a DTOS system program instead of a test program (see Word 13, bit 2), Word 8 is redefined as the CAT word (data channel word) with the bit definitions as given in Table D.3. Words 9 thru 12 do not apply in this case; they may be set to zero.

Bit	Description
0 = 1	This file is a CAT program.
1 = 1	The CAT program must ask the operator to input the device code to be used for data channel activity.
2 = 1	The CAT program must ask the operator if data channel testing should be performed on the 6070 and 4231 disk drives.
3 = 1	The CAT program must determine if data channel testing should be performed on the system load device.
4 = 1	The device is a low speed data channel device (e.g., the standard DCH).
5 = 1	The device is a high speed data channel device (e.g., the BMC).
6 - 7	Reserved for future use by the DTOS system.
8 - 15	Identification number for this CAT program.

Table D.3 Word 8

Words 9 through 12

These four words indicate the device code bit mask. They specify the devices necessary for the program to function and, together, form a bit mask with one bit per device code. Device codes are assigned to mask bits starting with device code 00 at the left-most bit of the first word (Word 9) and going to device code 77 in the right-most bit of the fourth word (Word 12). When a device is required by the program, the corresponding bit is set (1) in the bit mask.

Words 9 thru 12 are set up as shown below when a program must be prevented from running in auto mode. This configuration prevents the program from running in auto mode by requiring too many peripherals to be present for running.

70707 = Word 9
70707 = Word 10
70707 = Word 11
70707 = Word 12

Word 13

This word consists of control bits defined as shown in Tabel D.4.

Bit	Description
0 = 1	Secondary device code possible on primary device code plus 40 (octal).
1 = 1	The program supports the HELP command.
2 = 1	The program is a DTOS-system program. When this bit is set, the definition is changed for Word 8. Refer to "Word 8," above, for the alternate definition.
3 = 1	The program is larger than 32 Kwords.
4 = 1	The program uses the ECLIPSE MV/8000 code.
5 = 1	The program is included in the TEST command sequence.
6 = 1	The program is included in the VERIFY command sequence.
7 - 9	Reserved for future use by the DTOS system.
10 - 15	Primary device code for this program.

Table D.4 Word 13

Word 14

This word is a zero word reserved for future use by the DTOS system.

Word 15

If the program identified by this DIRT block runs as part of the SCP DTOS media, this word is the load-status word. The bits are defined as shown in Table D.5.

If the program runs as part of the ECLIPSE MV/8000 DTOS media and is longer than 32 Kwords, the bits are defined as shown in Table D.6.

If the program runs on any other DTOS media, Word 15 is zero.

Bit	Description
0 = 1	The SCP program alters the ECLIPSE MV/8000 microcode if bit 1 is also set.
1 = 1	Load this program into SCP local memory.
2 = 1	Load this program into ECLIPSE MV/8000 local memory.
3 = 1	Load this program into ECLIPSE MV/8000 control store memory.
4,5,6 = 000	Do not load any microcode.
= 001	Undefined.
= 010	Load standard instruction microcode.
= 011	Load standard instruction microcode and macrocode.
= 100	Load standard FRU test microcode.
= 101	Load standard FRU test microcode and macrocode.
= 110	Load nonstandard microcode.
= 111	Load nonstandard microcode and macrocode.
7 = 1	Load additional macrocode program.
8 - 9	Reserved to define future microcode types.
10 - 15	Reserved for future use by the DTOS system.

Table D.5 Word 15 (SCP DTOS)

Bits	Description
0 - 11	Reserved for future use by the DTOS system.
12 - 13	When an ECLIPSE MV/8000 program is longer than 32 Kwords (up to 128 Kwords), a splitter program divides the program into modules of 32 Kwords. If this program is a module of a larger program, these bits encode (in binary) the section number minus one for this module.
14 - 15	The total number minus one (in binary) of the 32-Kword modules that make up the program.

Table D.6 Word 15 (ECLIPSE MV/8000 DTOS)

Words 16 through 19

These words define the CPU types and subtypes that this program runs on (0 = program does not run; 1 = program runs). The words are defined as shown in Tables D.7 through D.10.

Bit	Description
0	microNOVA 601 (standard)
1	microNOVA 602 and MP/100
2	microNOVA MP/200
3	microNOVA MBC-1
4 - 14	Reserved for future use by the DTOS system; currently set to zero.
15	Reserved for future use by the DTOS system to indicate the existence of a second word indicating microNOVA subtype; currently set to zero.

Table D.7 Word 16 (microNOVA subtype)

Bit	Description
0	NOVA 1200
1	NOVA 800
2	NOVA II
3	NOVA 3
4	NOVA 4C
5	NOVA 4S and 4X
6-14	Reserved for future use by the DTOS system; currently set to zero.
15	Reserved for future use by the DTOS system to indicate the existence of a second word indicating NOVA subtype; currently set to zero.

Table D.8 Word 17 (NOVA subtype)

Bit	Description
0	ECLIPSE S/200 and C/300
1	ECLIPSE S/230 and C/330
2	ECLIPSE S/130
3	ECLIPSE C/150
4	ECLIPSE S/250
5	ECLIPSE C/350 and M/600
6	ECLIPSE S/140
7 - 14	Reserved for future use by the DTOS system; currently set to zero.
15	Reserved for future use by the DTOS system to indicate the existence of a second word indicating ECLIPSE subtype; currently set to zero.

Table D.9 Word 18 (ECLIPSE subtype)

Bit	Description
0	ECLIPSE MV/8000
1-14	Reserved for future use by the DTOS system; currently set to zero.
15	Reserved for future use by the DTOS system to indicate the existence of a second word indicating ECLIPSE MV/8000 subtype; currently set to zero.

Table D.10 Word 19 (ECLIPSE MV/8000 subtype)

Word 20

This word is a zero word reserved for future use by the DTOS system.

Word 21

This is the **TOPDOWN** sequence number. That is, if this program is part of the automatic testing sequence accessed with the **TOPDOWN** command, Word 21 encodes its level number (first byte) and the program number (second byte) in the hierachy of programs.

If this test is not part of the **TOPDOWN** testing sequence, Word 21 is -1 (177777).

Appendix E
Summary of DTOS Commands

Table E.1 lists all the DTOS commands by usage.

Display Directory	Select Directory	Load Program (No CAT)	Load Program (& CAT)	Unload DTOS Tape	Equipment Table	Misc. Commands
DIR	ALL	ACCEPT	--	EXIT	ADD,N	ENTER
?	HINT	DEBUG	--		ASSIGN,*n*	HELP
	HIOP	LOAD	CLOAD		DELETE,*n*	PARAMETER
	HOST	LOAD.*n*	CLOAD.*n*		EQUIP	SWREG
	IOP	MLOAD	--			TAPE
		ODT	--			
		REPEAT	CREPEAT			
		RUN,*n*	CRUN,*n*			
		SELECT	CSELECT			
		TOPDOWN	CTOPDOWN			

Table E.1 Summary of DTOS commands

Table E.2 lists the commands in alphabetical order, showing the command-line format, and gives a brief description of each. See "Conventions," Chapter 1, for an explanation of the command-line formats and of the different ways you can input the *program name* argument. (Do not input the parentheses or brackets. These only set off the arguments.)

The *.dd* option shown in some of the command lines means that you can run programs on a device with a nonstandard device code (*.dd* = new device code). You cannot use a command with the *.dd* argument if ALL is your working directory.

Input	Command Description
ACCEPT	Similar to **RUN,1** command except internal pass counter for each program is forced to 1.
ADD,*n (device mnemonic, dev. code),...*	Adds specified argument(s) to equipment table number *n* (octal).
ALL	Selects all directories as working directories.
ASSIGN,*n (device mnemonic, dev. code),...*	Clears equipment table *n* (octal) and reassigns it with specified argument(s).
DEBUG *(program name[.dd])*	Loads specified program along with symbolic debugger, starts debugger, and runs until until manually stopped through console. Starting address is either 30000$_8$ for NOVA with more than 16KW memory and for any ECLIPSE or 12000$_8$ for NOVA with 16KW or less.
DELETE,*n (device mnemonic, dev. code),...*	Deletes specified argument(s) from equipment table number *n* (octal).
DIR	Displays name of current working directory and mnemonics of all diagnostic programs it contains.
ENTER	Enters input mode, which allows user to input any desired information through keyboard. This information is not stored in memory but is printed out on TTY or LPT. (Exit mode with CONTROL Z character.)
EQUIP	Displays table number and contents of current equipment table for each processor.
EXIT	Rewinds tape and halts.
HELP[1]	Displays choice of following 4 files: 1. Summary of DTOS commands. 2. Blank diagnostic trouble report (DTR) form 3. List of programs changed since last DTOS release 4. Table of cross-references between device model number and both diagnostic and reliability programs for that device.
HELP *(program name, ...)*[1]	Loads and displays text files for programs specified. (Note: Not all programs have this feature at this time.)
HINT	Selects INTERFACE directory (HINT) as working directory.
HIOP	Selects HOST and I/O PROCESSOR directory (HIOP) as working DIRECTORY.
HOST	Selects HOST directory as working directory.
IOP	Selects I/O PROCESSOR directory (IOP) as working directory.
LOAD *(program name[.dd])*[2]	Loads specified program and runs until manually stopped through console. DTOS must be reloaded for next operation.
LOAD.*n (program name[.dd])*[2]	Loads specified program and makes *n* (octal) passes.
MLOAD *(program name[.dd])*	Performs same function as **LOAD** command but is used by systems with TTI/TTO or LPT assigned secondary or nonstandard device codes. (Saves needed software I/O modules in memory.)
ODT *(program name[.dd])*	Loads specified program along with octal debugger (ODT), starts debugger at displayed address, and runs until exited with ODT commands **P** or *adr***R**.
PARAMETER	Has different meanings depending on the type of system under test. Do not use unless directed on its use by the related field level maintenance manual.
REPEAT *(program name[.dd],...)*-[2, 3]	Loads specified program(s) (15 max.) and repeats in sequence until manually stopped through console. Must reload DTOS for next operation.
RUN,*n*[2, 3]	Sequentially loads programs (in order listed in working directory) that fit current system equipment table and makes *n* (octal) passes. (If *n* equals 0, makes 65K passes.)

Table E.2 Commands listed alphabetically

Input	Command Description
SELECT *(program name[.dd],...)*[2, 3]	Loads specified program(s) (15 max.) and makes one pass in sequence.
SWREG	Enters input mode to change switches in DTOS switch register with octal input. Type linefeed if no changes, type carriage return after new input to change, and tpye carriage return with no new input to clear.
TAPE	Displays number of current disk tape-directory and allows selection of new directory with octal input (on disk system only). Number corresponds to number of tape used when building disk.
TOPDOWN[2, 3]	Runs topdown test of system to speed up isolation of problems in system.
?	Displays working directory name.

Table E.2 (continued) Commands listed alphabetically

[1] If you use the format **HELP ALL** instead of **HELP** *(program name,...)*, you can access the help files for all programs that display text files.

[2] Use the **C** prefix with this command if you want to load a data channel exerciser (CAT) along with your selected program(s).

[3] Selecting this command causes each diagnostic program loaded to set its own internal pass counter to a number from 1 through 7.

DG OFFICES

SALES AND SERVICE OFFICES

Alabama: Birmingham
Arizona: Phoenix, Tucson
Arkansas: Little Rock
California: El Segundo, Fresno, Los Angeles, Oakland, Palo Alto, Pasadena, Sacramento, San Diego, San Francisco, Santa Ana, Santa Barbara, Van Nuys
Colorado: Denver, Englewood
Connecticut: North Branford, Norwalk
Florida: Ft. Lauderdale, Orlando, Tampa
Georgia: Norcross
Idaho: Boise
Iowa: Cedar Rapids Bettendorf
Illinois: Arlington Heights, Champaign, Chicago, Peoria, Rockford, Schaumburg
Indiana: Indianapolis
Kentucky: Louisville
Louisiana: Baton Rouge, Metairie
Maine: Portland
Maryland: Baltimore
Massachusetts: Cambridge, Springfield, Wellesley, Worcester
Michigan: Grand Rapids, Southfield
Minnesota: Richfield
Missouri: Creve Coeur, Kansas City, St. Louis
Mississippi: Jackson
Montana: Billings
Nebraska: Omaha
Nevada: Las Vegas, Reno
New Hampshire: Bedford, Nashua
New Jersey: Cherry Hill, Somerset, Wayne
New Mexico: Albuquerque
New York: Albany, Buffalo, Lake Success, Latham, Melville, Newfield, New York, Rochester, Syracuse, White Plains
North Carolina: Charlotte, Greensboro, Greenville, Raleigh
Ohio: Brooklyn Heights, Cincinnatti, Columbus, Dayton
Oklahoma: Oklahoma City, Tulsa
Oregon: Lake Oswego, Portland
Pennsylvania: Blue Bell, Carnegie, Lancaster, Philidelphia, Pittsburgh
Rhode Island: Providence, Rumford
South Carolina: Columbia
Tennessee: Knoxville, Memphis, Nashville
Texas: Austin, Dallas, El Paso, Ft. Worth, Houston, San Antonio
Utah: Salt Lake City
Virginia: McLean, Norfolk, Richmond, Salem
Washington: Bellevue, Kirkland, Richland, Spokane
West Virginia: Charleston
Wisconsin: Brookfield, Madison, West Allis

INTERNATIONAL SUBSIDIARIES

Australia: Adelaide, Melbourne, New Castle, Sydney, Tasmania, Queensland, Victoria
Brazil: Sao Paulo
Canada: Calgary, Edmonton, Montreal, Ottawa, Quebec, Toronto, Vancouver, Winnipeg
France: Lille, Lyon, Nantes, Paris
Italy: Florence, Milan, Padua, Rome, Tourin
Japan: Tokyo
The Netherlands: Amsterdam, Rijswijk
New Zealand: Auckland, Wellington
Sweden: Gothenburg, Malmoe, Stockholm
Switzerland: Lausanne, Zurich
United Kingdom: Birmingham, Bristol, Chesire, Glasgow, Hounslow, London, Manchester
West Germany: Dusseldorf, Filderstadt, Frankfurt, Hamburg, Hannover, Munich, Nuremburg, Munich, Ratingen, Rodelheim, Stuttgart

REPRESENTATIVES & DISTRIBUTORS

Argentina: Buenos Aires
Bolivia: Novadata
Chile: Santiago
Columbia: Bogato
Costa Rica: San Jose
Ecuador: Quito
Egypt: Cairo
Finland: Helsinki
Guatemala: Guatemala City
Hong Kong: Hong Kong
India: Bombay
Indonesia: Jakarta
Israel: Givatayim
Korea: Seoul
Kuwait: Kuwait
Lebanon: Beirut
Malaysia: Kuala Lumpur
Mexico: Mexico City, Nuevo Leon
Morocco: Casablanca
Nicaragua: Managua
Nigeria: Ibadan, Lagos
Norway: Oslo
Paraguay: Asuncion
Peru: Lima
Philippine Islands: Manila
Portugal: Lisbon
Puerto Rico: Hato Rey
Saudi Arabia: Jaddah, Riyadh
Singapore: Singapore
South Africa: Capetown, Durban, Johannesburg, Pretoria
Spain: Barcelona, Bibao, Madrid, San Sebastian, Valencia
Taiwan: Taipei
Thailand: Bangkok
Turkey: Ankara
Uruguay: Montevideo
Venezuela: Maracaibo

ADMINISTRATION, MANUFACTURING RESEARCH AND DEVELOPMENT

Massachusetts: Cambridge, Framingham, Southboro, Westboro
Maine: Westbrook
New Hampshire: Portsmouth
California: Anaheim, Sunnyvale
North Carolina: Research Triangle Park, Johnston County
Texas: Austin

Hong Kong: Kowloon, Tai Po
Thailand: Bangkok

Data General

Engineering Publications Comment Form

Please help us improve our future publications by answering the questions below. Use the space provided for your comments.

Title: ______________________________

Document No. 015-000103-01

Yes	No			
□	□	Is this manual easy to read?	○ You (can, cannot) find things easily. ○ Language (is, is not) appropriate. ○ Technical terms (are, are not) defined as needed.	○ Other:
		In what ways do you find this manual useful?	○ Learning to use the equipment ○ As a reference ○ As an introduction to the product	○ To instruct a class. ○ Other:
□	□	Do the illustrations help you?	○ Visuals (are,are not) well designed. ○ Labels and captions (are,are not) clear. ○ Other:	
□	□	Does the manual tell you all you need to know? What additional information would you like?		
□	□	Is the information accurate? (If not please specify with page number and paragraph.)		

Name: ______________________ Title: ______________________

Company: ______________________ Division: ______________________

Address: ______________________ City: ______________________

State: __________ Zip: __________ Telephone: __________ Date: __________

CUT ALONG DOTTED LINE

Data General
Data General Corporation, Westboro, Massachusetts 01580

FOLD FOLD

STAPLE STAPLE

FOLD FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 26 SOUTHBORO, MA. 01772

Postage will be paid by addressee:

DataGeneral

ATTN: ENGINEERING PUBLICATIONS

4400 Computer Drive
Westboro, MA **01580**

Data General

Data General Corporation, Westboro, Massachusetts 01580