Data General

Customer Documentation

How to Generate and Run DG/RDOS

How to Generate and Run DG/RDOS

093-000470-03

For the latest enhancements, cautions, documentation changes, and other information on this product, please see the Release Notice (085-series) supplied with the software.

Ordering No. 093-000470

Printed in the United States of America
Rev. 03, November 1990

Notice

How to Generate and Run DG/RDOS
093-000470-03

Revision History:	Effective with:
Original Release - August, 1987	DG/RDOS Rev. 2.00
First Revision - May, 1988	DG/RDOS Rev. 2.20
Second Revision - September, 1989	DG/RDOS Rev. 2.40
Third Revision - November, 1990	DG/RDOS Rev. 2.60

A vertical bar in the margin of a page indicates substantive technical change from the previous revision. (The exception is Chapter 5, which contains entirely new material.)

Preface

This manual has two main parts. The first part, chapters 2 through 9, explains how to generate a DG/RDOS system on a DESKTOP GENERATION® computer, a DG/500™ system, an MV/1000™ DC computer, an MV/1400™ DC computer, an MV/2000™ DC system, an MV/2500™ DC system, an MV/3500™ DC system, an MV/5000™ DC Series system, an MV/7800™ XP system, an MV/9500™ system, an MV/15000™ system, or an MV/18000™ system. This process involves both loading system software onto your computer and running the tailoring programs SYSGEN and CONFIG to build a system that suits your configuration and your applications. The remaining chapters discuss managing the system: starting up and shutting down DG/RDOS, backing up and transferring files, running the disk initializer, setting up and using a RAM disk, and making changes to the system.

The manual assumes that you are familiar with computers, but not necessarily with RDOS or DG/RDOS. If you are acquainted with these operating systems, you can use the manual's syntax descriptions, examples, and sample dialogs as reference tools. If you are not, you should find the more detailed explanations of how to generate and run the system useful.

Organization of the Manual

This manual is organized as follows:

Chapter 1 provides an overview of how to generate and run a DG/RDOS system.

Chapter 2 explains how to load DG/RDOS system software on DESKTOP GENERATION computers.

Chapter 3 discusses loading DG/RDOS on DG/500 systems.

Chapter 4 explains how to load DG/RDOS on MV/1000 DC, MV/1400 DC, MV/2000 DC, and MV/2500 DC computers.

Chapter 5 explains how to load DG/RDOS on MV/3500 DC and MV/5000 DC Series computers.

Chapter 6 discusses loading DG/RDOS on MV/7800 XP, MV/9500, MV/15000, and MV/18000 systems.

Chapter 7 tells you how to use the SYSGEN utility to generate a DG/RDOS operating system that supports all of the devices in your configuration.

Chapter 8 discusses using the CONFIG utility to do such things as reserve memory for system use, indicate where your $ devices are attached and what their characteristics are, and set the baud rate and a variety of other characteristics for your QTY lines.

Chapter 9 deals with applying patches to DG/RDOS system files.

Chapter 10 discusses how to start up and shut down your system and how to handle an abnormal shutdown.

Chapter 11 provides an overview of the DG/RDOS utilities you use to back up your files and to transfer files between systems.

Chapter 12 describes the FCOPY utility, which is designed specifically for backing up diskette-based material.

Chapter 13 explains how to use the utility program BURST to perform disk-image backups.

Chapter 14 explains how to use the IMOVE utility to perform full or incremental backups.

Chapter 15 covers the LABEL utility, which you use to read and write diskette and tape labels.

Chapter 16 explains how to run the disk initializer DKINIT.

Chapter 17 discusses setting up and using a RAM disk.

Chapter 18 deals with making changes to your system, such as adding new hardware or installing a new revision of DG/RDOS.

Appendix A provides important information about the disk, diskette, and tape drives supported by DG/RDOS.

Appendix B tells you what to do if a new bad block develops on your disk.

Notational Conventions

This manual uses these conventions for command formats:

```
COMMAND required [optional]...
```

Where	Means
COMMAND	You must enter the command or its accepted abbreviation as shown.
required	You must enter some argument (such as a filename). Sometimes, we use the form

$$\left\{\begin{array}{l}\text{required-1}\\\text{required-2}\end{array}\right\}$$

In this case, you should enter one of the arguments. Do not enter the braces; they only set off the choice.

[optional] — You have the option of entering this argument. Do not enter the brackets; they only set off what is optional.

... — You can repeat the preceding entry or entries. The command explanation will tell you exactly what you can repeat.

In addition, this manual generally uses

```
ALL UPPERCASE LETTERS AND THIS FONT TO INDICATE ITEMS
THAT YOU SHOULD ENTER AT THE KEYBOARD
```

and

```
A Combination of Uppercase and Lowercase Letters and a Typewriter
Font to Indicate System Prompts and Messages
```

This manual also uses two other special symbols: <NL> and the vertical bar (|). The symbol <NL> indicates that you should press the New Line key. A vertical bar in the margin of a page highlights information that is new or that has changed since the last revision of this manual. (The exception is Chapter 5, which contains entirely new material.)

Contacting Data General

- To order any Data General manual, notify your sales representative and supply the manual title and order number.
- If you have software problems, please notify your Software Support Center or submit a Software Trouble Report (STR) to the Software Trouble Report Processing Center.
- If you have hardware problems, please notify the Field Engineering Dispatch Center.

-End of Preface-

Contents

Chapter 1 Generating and Running DG/RDOS: An Overview

Chapter 2 Loading DG/RDOS on a DESKTOP GENERATION System

Chapter 3 Loading DG/RDOS on a DG/500 System

Chapter 4 Loading DG/RDOS on an MV/1000 DC, MV/1400 DC, MV/2000 DC, or MV/2500 DC System

Chapter 5 Loading DG/RDOS on an MV/3500 DC or MV/5000 DC Series System

Chapter 6 Loading DG/RDOS on an MV/7800 XP, MV/9500, MV/15000, or MV/18000 System

Chapter 7 Running the SYSGEN Utility

Chapter 8 Running the CONFIG Utility

Chapter 9 Applying Patches

Chapter 10 Starting Up and Shutting Down Your System

Chapter 11 Backing Up and Moving Files

Chapter 12 FCOPY: Backing Up Diskette-Based Files

Chapter 13 BURST: Performing Disk-Image Backups

Chapter 14 IMOVE: Making Full and Incremental Backups

Chapter 15 LABEL: Reading and Writing Diskette and Tape Labels

Chapter 16 Using the Disk Initializer

Chapter 17 Using a RAM Disk

Chapter 18 Making Changes to Your System

Appendix A Disk, Diskette, and Tape Drives

Appendix B Handling New Bad Blocks

Index

Related Manuals

Tables

Table

Chapter 1

Generating and Running DG/RDOS: An Overview

This chapter summarizes what is involved in generating and running a DG/RDOS system, discusses special keyboard characters (and sequences of characters) that you will use as you work with the system, and describes QTY lines, foreground consoles, and background consoles. These subjects are covered in the following sections:

- What Is DG/RDOS?
- Generating a DG/RDOS System
- Running the System
- Using the Console's Special Keys

The final section, "Before You Proceed," describes what you should know about QTY lines, foreground consoles, and background consoles.

What Is DG/RDOS?

DG/RDOS is a version of Data General's Real-Time Disk Operating System (RDOS) that runs on DESKTOP GENERATION®, DG/500™, MV/1000™ DC, MV/1400™ DC, MV/2000™ DC, MV/2500™ DC, MV/3500™ DC, MV/5000™ DC Series, MV/7800™ XP, MV/9500™, MV/15000™, and MV/18000™ systems. In each of these hardware environments, the operating system supports all of the computer's physical address space, 64 Kbytes of logical address space, and 16-bit accumulators.

Other major features of DG/RDOS include the following:

- DG/RDOS is a small operating system that is completely memory resident; therefore, the system is fast.
- DG/RDOS supports dual-ground processing. That is, two programs can reside in memory—one in the background and one in the foreground—and appear to execute simultaneously. The programs share such system resources as CPU time and I/O devices.
- DG/RDOS enables you to write multitask applications and, thus, to make maximum use of CPU time.

The operating system also possesses a number of unique technical features such as support for user-defined devices.

Generating a DG/RDOS System

Building a new DG/RDOS system involves two steps. First, you must move the operating system files from the release diskettes or tape Data General provides to your master disk so that you can boot the system from that disk. Then, once you have the starter system running, you use the utility programs SYSGEN and CONFIG to tailor DG/RDOS for your particular hardware configuration and applications.

The basic procedure for loading the operating system onto your disk is outlined below:

1. Load into memory the bootstrap code that resides on your first release diskette or tape.
2. Bootstrap the disk initializer DKINIT, which is also on the first release diskette or tape. This utility allows you to check your master disk for bad blocks, set up a bad block table, define a remap area, and establish your disk's frame size.
3. Load the disk bootstrap routine DISKBOOT from your first release diskette or tape into memory, and install a bootstrap root on your master disk.
4. Bootstrap the DG/RDOS starter system from your first release diskette or tape, set up a file system on your master disk, and move the operating system files from your release medium to your disk.

At this point you can run DG/RDOS from your master disk; however, the system on that disk is almost certainly not the ideal system for your hardware configuration and your applications. For example, the starter system does not support multiplexor lines, and it may not include enough stacks, cells, and buffers for you to run your programs efficiently. To build your optimal system, you can run the tailoring programs SYSGEN and CONFIG.

The purpose of the SYSGEN utility is very simple—the program allows you to specify what devices are and are not in your configuration. The utility then builds a system that supports your peripherals and does not include unneeded drivers.

The CONFIG utility, on the other hand, lets you define a variety of characteristics for your system. For instance, the program allows you to set up the capability to run a foreground program that uses a foreground console; it lets you decide how much memory to reserve for DG/RDOS use; and it allows you to define a set of characteristics for each multiplexor line you have.

Running the System

After you have generated your DG/RDOS system, you must determine how to run it. In some instances, you will have to consult other manuals to find the information you need. For instance, you can find complete information about using the Command Line Interpreter in *Using the DG/RDOS Command Line Interpreter*, and you can read about developing assembly-language programs in *RDOS/DOS Assembly Language and Program Utilities*. The manual you are reading now, however, covers most aspects of running the system, including

- Starting the system, shutting it down, and handling an abnormal shutdown
- Backing up your programs and data
- Initializing new disks and maintaining old ones
- Setting up and using a RAM disk
- Adding new hardware or software to your system

Using the Console's Special Keys

While generating your system, and afterwards, you will use your system console extensively. You can make most effective use of this console if you are familiar with the uses of the special keys and sequences of keys shown in Table 1-1. These keys allow you to correct typing mistakes, interrupt the execution of a program, and regulate scrolling.

Table 1-1 Special Keys and Key Sequences

Keys	Purpose
\ (backslash)	If you are in the CLI, pressing the backslash key normally erases the current line and allows you to retype your command line. If SCREENEDIT mode is on, typing a backslash erases all characters from the beginning of the line to the position of the cursor.
Cmd-Brk	Pressing these two keys simultaneously returns you to the virtual console (DESKTOP GENERATION and DG/500 systems) or the System Control Program (MV/Family systems). On some machines, you must type Cmd-Brk three times to achieve this effect. You can resume processing at the point where you typed Cmd-Brk by typing P (DESKTOP GENERATION and DG/500 systems), TTY<NL> (MV/3500 DC and MV/5000 DC Series systems), or CONTINUE<NL> (other MV/Family systems).
Ctrl-C Ctrl-A	The Ctrl-C Ctrl-A sequence interrupts the execution of a program or CLI command.
Ctrl-C Ctrl-B	If you type this sequence while a program is running, the system waits for task I/O to complete, terminates the program, and writes the current memory image to a breakfile called BREAK.SV or FBREAK.SV.
Ctrl-C Ctrl-F	Typed from the system console, this sequence terminates a foreground program.
Ctrl-Q	If you have suspended terminal output by typing Ctrl-S, you can cancel that suspension by entering Ctrl-Q.
Ctrl-S	Typing Ctrl-S suspends terminal output.
Delete	Pressing the Delete key erases the last character you typed.
New Line	You use the New Line key to terminate a command line.

Before You Proceed

Read this information before you begin loading DG/RDOS on your system.

QTY Lines

If your system includes one or more asynchronous communications line controllers, see Chapter 7 for a discussion of which controller DG/RDOS considers to be the first, which it considers to be the second, and so on. Then consult the list below to find the relationship between individual asynchronous lines and QTY filenames.

- For your first controller, the first physical line has the DG/RDOS filename QTY:0; the second line is QTY:1, and so on.
- For your second controller, the first physical line has the DG/RDOS filename QTY:*n*, where *n* is the number of lines on the first controller. The second physical line is QTY:(*n*+1), and so on.
- For your third controller, the first physical line has the DG/RDOS filename QTY:*n*, where *n* is the number of lines on the first two controllers. The second physical line is QTY:(*n*+1), and so on.
- For your fourth controller, the first physical line has the DG/RDOS filename QTY:*n*, where *n* is the number of lines on the first three controllers. The second physical line is QTY:(*n*+1), and so on.

The only exception is on DESKTOP GENERATION systems. When numbering QTY lines, DG/RDOS does not differentiate between one- and four-line USAM cards. The first line on the second card has the filename QTY:4, even if the first card is a one-line card. The first line on the third card has the filename QTY:8, and the first line on the fourth card has the filename QTY:12.

QTY line numbering is not affected by any use of the SYSGEN and CONFIG utilities to link the background console, the foreground console, or extra linked $ devices to QTY lines. However, you can no longer refer to those lines with the filename QTY:*n*. You can still use the CONFIG utility to define the baud rate and other characteristics for any such lines.

Background Console

A background (BG) console is the console that the operating system refers to by the filenames $TTI and $TTO. All DG/RDOS operating systems have a BG console. When you bring up an operating system, this console is used by the initial CLI (which runs in the background). On systems with a dedicated master console, the BG console and the master console are normally the same.

A master console is the console that uses device codes 10 and 11. Most systems have a dedicated master console.

MV/1000 DC, MV/1400 DC, MV/2000 DC, MV/2500 DC, MV/3500 DC, and MV/5000 DC Series systems do not have a dedicated master console; these systems do provide master console emulation, however. This emulation can be used when the master console is the only asynchronous line on the system board that will be used.

Stand-alone programs like DKINIT, DSKED, and the disk and tape bootstraps use the master console. Many systems use the master console for power-up messages or menus.

On MV/1000 DC, MV/1400 DC, MV/2000 DC, and MV/2500 DC systems with LAC QTY lines or with no QTY lines, use master console emulation. In this case, your BG console and master console are the same. You cannot use the other asynchronous lines on the system board.

On MV/1000 DC, MV/1400 DC, MV/2000 DC, and MV/2500 DC systems with DUART QTY lines, master console emulation cannot be used. Run the SYSGEN utility and, before trying to boot your new operating system, use the CONFIG utility to link the BG console to a QTY line. Use QTY:1, since that is the line used for master console emulation.

On MV/3500 DC and MV/5000 DC Series systems, use the asynchronous lines on the system board as LAC QTY lines. In this case, master console emulation cannot be used. Run the SYSGEN utility and, before trying to boot your new operating system, use the CONFIG utility to link the BG console to a QTY line. Use the last QTY line on the system board; that is the line used for master console emulation.

If you choose not to use the lines on an MV/3500 DC or MV/5000 DC Series system board as LAC QTY lines, you can use master console emulation. In this case, your BG console and master console are the same. You cannot use the other asynchronous lines on the system board.

Foreground Console

A foreground (FG) console is the console that the operating system refers to by the filenames $TTI1 and $TTO1. The SYSGEN utility lets you choose whether to have a FG console. On systems with a secondary console, the FG console and the secondary console are normally the same.

A secondary console is the console that uses device codes 50 and 51. DG/500 and DESKTOP GENERATION model 10 and 10-SP systems have a dedicated secondary console. Most other systems can have a secondary console as an option board; however, MV/1000 DC, MV/1400 DC, MV/2000 DC, MV/2500 DC, MV/3500 DC, and MV/5000 DC Series systems cannot.

If you have QTY lines, you can have a FG console even if you don't have a secondary console. Run the SYSGEN utility and request support for your QTY lines and a FG console. Then, before trying to boot your new operating system, use the CONFIG utility to link the FG console to a QTY line.

On MV/1000 DC, MV/1400 DC, and MV/2000 DC systems with DUART QTY lines, use QTY:3, since that is the line that previous revisions of DG/RDOS used for the FG console.

On MV/2500 DC systems with DUART QTY lines, use QTY:0 for the same reason.

On MV/3500 DC and MV/5000 DC Series systems that use the asynchronous lines on the system board as LAC QTY lines, use the second to last QTY line on the system board.

If you need further information about slot and line numbers, consult your hardware installation manual.

-End of Chapter-

Chapter 2

Loading DG/RDOS on a DESKTOP GENERATION System

Read this chapter when you have set up a new DESKTOP GENERATION computer and want to install DG/RDOS from diskettes.

Before you can run the tailoring programs discussed in Chapters 7 and 8, you must load the DG/RDOS system files onto your first, or only, fixed disk or onto a system diskette (DESKTOP GENERATION Models 10 and 10SP only). The following sections cover the steps that constitute this loading process:

- Preparing Your Hardware
- Initializing Your Master Disk
- Installing a Bootstrap Root on Your Disk
- Transferring the Operating System Files to Your Disk

Preparing Your Hardware

You must perform several steps before you can boot a program from your release diskettes. The following instructions are based on the assumption that your system is shut down when you begin.

1. Turn on power to the system console using the switch on the back of your monitor.
2. Turn on each terminal, printer, and plotter in the system, and make sure that these devices are on-line. Also, turn on any remote units such as an external tape drive.
3. Turn on power to the computer itself using the switch at the upper right corner of the power supply module. The On Line status light at the top of each keyboard should glow. If a terminal's light does not glow, press the Cmd and On Line keys simultaneously to put the terminal on-line.

NOTE: If you have a Model 10 or 10SP with a printer or terminal on its CPU printer port, that device must be on-line. Also, if the device is a printer, it must have paper. When you power up a Model 10 or 10SP with a device on the CPU printer port, the computer tests the device. If the device fails the test—which happens if the device is off-line—the computer does not complete its power-up tests.

On a Model 10 or 10SP system, the computer runs a system power-up test and displays the following message:

```
TESTING...

ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
BLINK DIM REVERSE UNDERLINE ALL NORMAL

TEST PASSED.
```

If your Model 10 or 10SP has a fixed disk, you also see the message `SELECT LOAD DEVICE: 20H (FOR DISKETTE) OR 26H (FOR DISK)`. Then the virtual console prompt (`!`) appears.

On a Model 20 or 30, the virtual console prompt appears immediately.

5. If you have a Model 10 or 10SP with a German, Swiss/German, or Swiss/French keyboard, you must now load a terminal emulator. Insert the diskette labeled "D211 Bootable Emulator" into drive DJ0 (the right-hand drive), and at the virtual console prompt, type `20H`. (Note that you do not type a line-termination character, such as New Line, after entering a virtual-console command.) When the emulator program has been loaded into memory, the virtual console prompt returns and you can remove the diskette.

You are now ready to software format your system disk or diskette.

Initializing Your Master Disk

Your terminal should now be displaying the virtual console prompt (`!`).

1. Insert your first DG/RDOS release diskette in drive DJ0, the right-hand drive. At the virtual console prompt, type `20H`. At this point, you see a `Filename?` prompt.

2. Type `DKINIT<NL>` within approximately 30 seconds to start the disk initializer program. During system generation, you use this program to write identifying information on block 3 of your disk, to check your disk for bad blocks, to set up a bad block table, to establish a remap area, and to define your disk's frame size.

3. The program announces itself with the following message:

```
Disk Initializer - Revision xx.xx

  *--> Help can be obtained to any question
  *--> simply respond to the question with HELP or ?

Disk drive model number?
```

To answer DKINIT's first question, supply the model number of your first, or only, fixed disk (unless you are working on a diskette-based system, in which case you should insert a hardware-formatted diskette in drive DJ1 and enter the model number of your diskette drive). If you do not know this model number, consult Appendix A or use DKINIT's help facility.

```
Disk drive model number? 6336<NL>

Disk unit?
```

To identify the disk you want it to examine, DKINIT also needs to know the disk's or diskette's unit name. This name is DE0 for the first (or only) fixed disk and DJ1 for a diskette in your left-hand drive. Type in the appropriate identifier and press New Line.

```
Disk unit? DE0<NL>

6336   DEn    70 MB  5-1/4" sealed moving head disk

Command?
```

DKINIT displays a line of information about your disk and then asks for a command.

4. Since you are working with a new disk, issue DKINIT's FULL command and then answer the questions the utility poses. For complete information about running this part of the disk initializer program, consult the subsections "Coresident Diagnostics" and "The Rest of the FULL Command's Dialog" in Chapter 16. Or if you want to initialize your disk in a fairly standard way, use the sample dialog below as a guide.

```
Command? FULL<NL>

Command destroys any previous RDOS disk structure
RDOS INIT/F must be done on the disk after the command
Type CONTROL-A now to abort without loss

Do you want a coresident diagnostic area? NO<NL>
```

```
Number of patterns to run (1-5)? 5<NL>

*** Pattern # 1 (155555) ***
Checksum error - Bad block = 001154
Disk error - Bad block = 007107

*** Pattern # 2 (133333) ***

*** Pattern # 3 (066667) ***

*** Pattern # 4 (155556) ***

*** Pattern # 5 (125252) ***

*** All patterns run ***

Do you wish to declare any blocks bad
that are not already in the bad block table? NO<NL>

Default remap area size is x block(s) long
It needs to be at least y block(s) long
Remap area size (type RETURN for default) ? <NL>

Remap area start block number (type RETURN for default) ? <NL>
Remap area will start at block x

Default frame size is x, min is 1, and max is y
Frame size (type RETURN for default) ? <NL>

Full disk init complete

Command?
```

5. When the utility has fully initialized your disk and asks for a new command, type STOP<NL> to leave DKINIT. As the program terminates, DESKTOP GENERATION systems display the virtual console prompt.

Installing a Bootstrap Root on Your Disk

By initializing your master disk, you have prepared it to hold its first few blocks of system software: the bootstrap root. After you have moved all the operating system files onto your disk, you can instruct your system to load this bootstrap code into memory and to transfer control to it. At that point the bootstrap root will enable you to read in and execute either a DG/RDOS operating system or a stand-alone utility such as DKINIT.

1. Make sure that your first release diskette is inserted in drive DJ0. At the virtual console prompt, type 20H. At this juncture, the bootstrap root on your release diskette begins executing, and you see the Filename? prompt.

2. Type DISKBOOT<NL> within approximately 30 seconds to bring up the disk bootstrap program. You use this program to install a bootstrap root on the disk or diskette you just initialized.

3. As the bootstrap program begins execution, it prompts you for the unit name of the disk or diskette on which you want to install a bootstrap root:

```
DISKBOOT Rev xx.xx

Bootstrap Device Specifier ?
```

   Respond to this question by entering the same unit name that you used in your dialog with DKINIT.

```
Bootstrap Device Specifier ? DE0<NL>

Install Bootstrap (Y or N) ?
```

   The program now asks whether you want to install a bootstrap root on this disk. Type Y (a New Line is not necessary). Doing so causes DISKBOOT to write bootstrap code to the first few blocks of your disk and then to display the message Done.

4. After installing a bootstrap root, DISKBOOT again prompts you for a unit name. This time enter the unit name of the drive in which you inserted your first release diskette (DJ0) and type a New Line. Then, when the program asks whether you want to install bootstrap code on this disk, answer N. Following these steps takes you back to the Filename? prompt.

```
Bootstrap Device Specifier ? DJ0<NL>

Install Bootstrap (Y or N) ? N

Filename?
```

Transferring the Operating System Files to Your Disk

To finish installing DG/RDOS, you need to bring up the starter system you received on your first release diskette, create master allocation and system file directories on your system disk or diskette, and then transfer the DG/RDOS system files to your newly formatted disk.

Bringing Up the Starter System

As you begin this step, you should have your first release diskette in drive DJ0. Your system console may be displaying the prompt `Filename?` If it is not displaying that prompt, step 1 has been done automatically; go on to step 2.

1. Type New Line to start the operating system named SYS.SV, which is linked to DGRDOS.SV. This system announces itself with the message `DG/RDOS Rev xx.xx`.

2. You must now set the system date and time. When the system console displays the prompt `Date (m/d/y) ?`, enter the date (followed by a New Line) in one of the following formats:

   ```
   4 28 88
   4,28,88
   4-28-88
   4:28:88
   4/28/88
   APR 28 88
   APR,28,88
   28-APR-88
   ```

   The system then prompts you for the time. After this prompt, enter the time (followed by a New Line) in one of the following formats:

   ```
   14 30 00
   14,30
   14-30
   14:30
   14/30
   2 30 PM
   2:30 P
   ```

3. On DESKTOP GENERATION Model 10 and 10SP systems, after a short pause you see the message `Welcome to the DG Desktop Generation`. The system now asks if you would like to install a terminal emulator file:

   ```
   EMULATOR FILE NOT FOUND: xD211yyyy.TX
   DO YOU WISH TO LOAD THE EMULATOR (Y or N) ?
   ```

   You need to install this file on your system disk later, but for now answer `N` (pressing a New Line is not necessary).

4. At this point, the system console displays the DG/RDOS Command Line Interpreter prompt, R. The DG/RDOS starter system is up and running.

Setting Up a File System on Your Master Disk

Once the starter system is up, you can use a CLI command to establish a new master allocation directory (MAP.DR) and system directory (SYS.DR) on your system disk.

1. Use the command INIT/F followed by the disk unit name that you used earlier in your dialogs with DKINIT and DISKBOOT. For instance, you might enter this command:

   ```
   R
   INIT/F DE0<NL>
   ```

2. Since the CLI's full initialization effectively destroys all existing files on a disk, the system prompts you to confirm the command you just issued. Respond by typing Y (a New Line is not necessary).

   ```
   Confirm? Yes
   ```

   NOTE: The system adds the letters es.

Your system disk is now ready to hold the operating system files that you received on your release diskettes.

Transferring the DG/RDOS Files to Your Disk

Transferring the DG/RDOS system files to your newly formatted disk is a two-stage process. Since the system files on your first release diskette are in an executable format, you transfer these files to your disk using the CLI command MOVE. The remaining system files are in IMOVE (dump) format; therefore, you must boot the DG/RDOS starter system from your system disk and then use the IMOVE utility to load these files into your master directory.

1. With your first release diskette in drive DJ0, issue a MOVE command that follows this format.

   MOVE/V device_name

   For example, to move all of the nonpermanent files on your first release diskette to the first fixed disk on a DESKTOP GENERATION system, you would type

   ```
   MOVE/V DE0<NL>
   ```

   As the system moves these files, it displays their names on your system console:

   ```
   DGRDOS.SV
   DGRDOS.OL
   CLI.SV
   IMOVE.SV
   ...
   R
   ```

2. Now, before you can load the remainder of the DG/RDOS system files, you must shut down the starter system that is currently running and restart DG/RDOS from your master disk. If you are working on a DESKTOP GENERATION Model 10 or 10SP system and building your system on a diskette, make sure that this diskette is still in drive DJ1. Then, to shut down the current operating system and boot the system on your master disk, use the CLI's BOOT command:

   BOOT disk-name:SYS

   For instance, to boot the starter system from the first fixed disk on a DESKTOP GENERATION system, you would type

   ```
   BOOT DE0:SYS<NL>
   ```

   In response to this command, the current system displays the message `Master device released`, and then the new starter system begins to come up. Set the date and time when you are prompted to do so.

   If you are working on a DESKTOP GENERATION Model 10 or 10SP system, the system now asks if you would like to install a terminal emulator file:

   ```
   EMULATOR FILE NOT FOUND: xD211yyyy.TX
   DO YOU WISH TO LOAD THE EMULATOR (Y or N) ?
   ```

   If you are working on one of these systems, type `Y` (pressing a New Line is not necessary), and you will see the following message:

   ```
   INSERT EMULATOR DISKETTE IN DRIVE DJ0; TYPE C TO CONTINUE
   ```

   As the directions above indicate, you should now insert the diskette labeled "DG/RDOS D211 Emulator" in drive DJ0 and type `C`. When you have performed these steps, the system loads the emulator into memory and also copies an emulator file to your master directory. At the end of this process, your terminal should display the following messages:

   ```
   ...Emulator load completed
   xD211yyyy.TX is being copied to the master directory
   ```

   Now remove your emulator diskette from DJ0. In the future, each time you start up your system, a program called LOADEM will read in the emulator automatically.

   At this juncture, on all systems, the master console displays the DG/RDOS Command Line Interpreter prompt, R.

3. From this point, you issue a single IMOVE command to load the remaining DG/RDOS files. The format for this command is

   IMOVE/A/F/V diskette-name

   Thus, to load the remaining files on a DESKTOP GENERATION system, you normally enter the command

   ```
   IMOVE/A/F/V DJ0<NL>
   ```

   The IMOVE utility responds to this command by prompting you to insert the first diskette in the IMOVE dump file—that is, your second release diskette—and then to enter a New Line. Follow these instructions, and IMOVE will begin loading files onto your master disk and displaying the names of those files on your system console.

   ```
   Please insert labeled disk 1, then press NEW LINE. . .<NL>
   30-APR-89 16:01:22
     BURST.SV
     CONFIG.SV
     DO.SV
     ...
   ```

   IMOVE continues until it has loaded all of the files present on your second release diskette. Then, if the files in IMOVE format occupy more than one diskette, the utility prompts you to insert the second diskette in the IMOVE dump file—your third release diskette.

   ```
   This disk has been exhausted
   Please insert labeled disk 2, then press NEW LINE. . .<NL>
   ```

   This process goes on until IMOVE has loaded onto your disk all of the DG/RDOS system files that are not on the bootable release diskette.

   If you are building your DG/RDOS system on a diskette, you will not be able to load all of the files in IMOVE format onto your system diskette. Therefore, you must decide which files you need and load them using an IMOVE command that follows this format:

   IMOVE/A/F/V diskette-name filename ...

   In the *diskette-name* slot, enter the unit name of the drive in which you plan to insert your release diskettes, and in the *filename* ... slot, list the names of the files you want to include on your system diskette.

Once you have transferred the DG/RDOS files to your disk, the basic installation process is complete. However, you still need to run the SYSGEN and CONFIG programs to tailor your operating system to your particular hardware configuration. If you want to run these programs now, turn to Chapters 7 and 8. Otherwise, consult Chapter 10 for instructions on how to shut down your system.

-End of Chapter-

Chapter 3

Loading DG/RDOS on a DG/500 System

Read this chapter after (1) you have set up a new DG/500 computer and (2) you have powered up your system for the first time according to the instructions in your hardware startup manual. From this point, you can install DG/RDOS on your first, or only, fixed disk by following the steps detailed in the following sections:

- Initializing Your Master Disk
- Installing a Bootstrap Root on Your Disk
- Transferring the Operating System Files to Your Disk
- Installing Power-Up Diagnostics

Initializing Your Master Disk

If you followed the directions in your hardware startup manual for powering up your DG/500 system for the first time, you now have your System Media diskette in your upper, or only, diskette drive (DJ0), and your screen is displaying the "Start from a Different Device" menu.

```
Start from a Different Device

1 Hard disk
2 Diskette
3 Tape


To exit from this menu, press the Cancel/Exit key (F11) or ESC
For assistance, press the Help key (SHIFT-F1) or H


Start from which device? [1]
```

The instructions below indicate how you should proceed to software format the disk that will hold your DG/RDOS operating system.

NOTE: Before you proceed with the steps below, make sure that if DJ0 (your upper diskette drive) is a 48 TPI drive, your DG/RDOS release diskettes are 48 TPI diskettes. If DJ0 is a 96 TPI drive, your release diskettes must be 96 TPI diskettes.

1. Remove your System Media diskette from its drive, and in its place insert your first DG/RDOS release diskette. Then type 2<NL> to boot that release diskette. Shortly, you will see the prompt `Filename?`

2. Type `DKINIT<NL>` within approximately 30 seconds to start the disk initializer program. During system generation, you use this program to write identifying information on block 3 of your disk, to check your disk for bad blocks, to set up a bad block table, to establish a remap area, and to define your disk's frame size. You can also use the program to reserve space for coresident diagnostics.

3. The program announces itself with the following message:

```
Disk Initializer - Revision xx.xx

  *--> Help can be obtained to any question
  *--> simply respond to the question with HELP or ?

Disk drive model number?
```

To answer DKINIT's first question, supply the model number of your first, or only, fixed disk. If you do not know this model number, consult Appendix A or use DKINIT's help facility.

```
Disk drive model number? 6508<NL>

Disk unit?
```

To identify the disk you want it to examine, DKINIT also needs to know the disk's unit name. Normally, this name is DE0, so type in that information and press New Line.

```
Disk unit? DE0<NL>

6508   DEn        70 MB  5-1/4" sealed moving head disk

Command?
```

DKINIT displays a line of information about your disk and then asks for a command.

4. Since you are working with a new disk, issue DKINIT's FULL command and then answer the questions the utility poses. For complete information about running this part of the disk initializer program, consult the subsections "Coresident Diagnostics" and "The Rest of the FULL Command's Dialog" in Chapter 16. Or if you want to initialize your disk in a standard way, use the sample dialog below as a guide.

```
Command? FULL<NL>

Command destroys any previous RDOS disk structure
RDOS INIT/F must be done on the disk after the command
Type CONTROL-A now to abort without loss

Do you want a coresident diagnostic area? YES<NL>
```

NOTE: You must declare a coresident diagnostic area if you want to avoid loading power-up diagnostics from diskette every time you start up your system. The appropriate size for this diagnostic area is specified below.

```
The default coresident diagnostic area size is 5000 blocks
How big would you like your diagnostic area to be? 720<NL>
```

```
Number of patterns to run (1-5)? 5<NL>

*** Pattern # 1 (155555) ***
Checksum error - Bad block = 001154
Disk error - Bad block = 007107

*** Pattern # 2 (133333) ***

*** Pattern # 3 (066667) ***

*** Pattern # 4 (155556) ***

*** Pattern # 5 (125252) ***

*** All patterns run ***

Do you wish to declare any blocks bad
that are not already in the bad block table? NO<NL>

Default remap area size is x block(s) long
It needs to be at least y block(s) long
Remap area size (type RETURN for default) ? <NL>

Remap area start block number (type RETURN for default) ? <NL>
Remap area will start at block x

Default frame size is x, min is 1, and max is y
Frame size (type RETURN for default) ? <NL>

Full disk init complete

Command?
```

5. When the utility has fully initialized your disk and asks for a new command, type STOP<NL> to leave DKINIT. As the program terminates, DG/500 systems display the virtual console prompt (!).

Installing a Bootstrap Root on Your Disk

By initializing your master disk, you have prepared it to hold its first few blocks of system software: the bootstrap root. After you have moved all the operating system files onto your disk, you can instruct your system to load this bootstrap code into memory and to transfer control to it. At that point the bootstrap root will enable you to read in and execute either a DG/RDOS operating system or a stand-alone utility such as DKINIT.

1. Make sure that your first release diskette is inserted in drive DJ0. At the virtual console prompt, type 20H. The bootstrap root on your release diskette begins executing, and you see the `Filename?` prompt.

2. Type `DISKBOOT<NL>` within approximately 30 seconds to bring up the disk bootstrap program. You use this program to install a bootstrap root on the disk you just initialized.

3. As the bootstrap program begins execution, it prompts you for the unit name of the disk or diskette on which you want to install a bootstrap root:

```
DISKBOOT Rev xx.xx

Bootstrap Device Specifier ?
```

   Respond to this question by entering the same unit name that you used in your dialog with DKINIT.

```
Bootstrap Device Specifier ? DE0<NL>

Install Bootstrap (Y or N) ?
```

   The program now asks whether you want to install a bootstrap root on this disk. Type `Y` (pressing a New Line is not necessary). Doing so causes DISKBOOT to write bootstrap code to the first few blocks of your disk and then to display the message `Done`.

4. After installing a bootstrap root, DISKBOOT again prompts you for a unit name. This time enter the unit name of the drive in which you inserted your first release diskette (DJ0) and press New Line. Then, when the program asks whether you want to install bootstrap code on this disk, answer `N`. Following these steps takes you back to the `Filename?` prompt.

```
Bootstrap Device Specifier ? DJ0<NL>

Install Bootstrap (Y or N) ? N

Filename?
```

Transferring the Operating System Files to Your Disk

To finish installing DG/RDOS, you need to bring up the starter system you received on your first release diskette, create master allocation and system file directories on your system disk, and then transfer the DG/RDOS system files to your newly formatted disk. As you begin this step, you should have your first release diskette in drive DJ0, and your system console should be displaying the prompt `Filename?` If it is not displaying that prompt, step 1 has been done automatically; go on to step 2.

1. Press New Line to start the operating system named SYS.SV, which is linked to DGRDOS.SV. This system announces itself with the message `DG/RDOS Rev xx.xx`, displays the date and time, and then displays the DG/RDOS Command Line Interpreter prompt, R. The DG/RDOS starter system is up and running.

2. Once the starter system is up, you can use a CLI command to establish a new master allocation directory (MAP.DR) and system directory (SYS.DR) on your system disk. Use the command INIT/F followed by the disk unit name that you used earlier in your dialogs with DKINIT and DISKBOOT:

   ```
   R
   INIT/F DE0<NL>
   ```

   Since the CLI's full initialization effectively destroys all existing files on a disk, the system prompts you to confirm the command you just issued. Respond by typing `Y` (pressing a New Line is not necessary).

   ```
   Confirm? Yes
   ```

   NOTE: The system adds the letters `es`.

   Your system disk is now ready to hold the operating system files that you received on your release diskettes.

3. Transferring the DG/RDOS system files to your newly formatted disk is a two-stage process. Since the system files on your first release diskette are in an executable format, you transfer these files to your disk using the CLI command MOVE. The remaining system files are in IMOVE (dump) format; therefore, you must boot the DG/RDOS starter system from your system disk and then use the IMOVE utility to load these files into your master directory.

   To carry out the first of these two steps, ensure that your first release diskette is in drive DJ0, and then issue a MOVE command that follows this format:

   ```
   MOVE/V device_name
   ```

For example, to move all of the nonpermanent files on your first release diskette to the first fixed disk on a DG/500 system, you type

```
MOVE/V DE0<NL>
```

As the system moves these files, it displays their names on your system console:

```
DGRDOS.SV
DGRDOS.OL
CLI.SV
IMOVE.SV
...
R
```

4. Now, before you can load the remainder of the DG/RDOS system files, you must shut down the starter system that is currently running and restart DG/RDOS from your master disk. To do this, use the CLI's BOOT command:

   ```
   BOOT DE0:SYS<NL>
   ```

   In response to this command, the current system displays the message `Master device released`, and then the new starter system begins to come up. At the end of this step, you should see the CLI's R prompt.

5. From this point, you issue a single IMOVE command to load the remaining DG/RDOS files. The format for this command is

   IMOVE/A/F/V diskette-name

   Thus, to load the remaining files on a DG/500 system, you normally enter the command

   ```
   IMOVE/A/F/V DJ0<NL>
   ```

   The IMOVE utility responds to this command by prompting you to insert the first diskette in the IMOVE dump file—that is, your second release diskette—and then to press New Line. Follow these instructions, and IMOVE begins loading files onto your master disk and displaying the names of those files on your system console.

   ```
   Please insert labeled disk 1, then press NEW LINE. . .<NL>
   30-APR-89 16:01:22
     BURST.SV
     CONFIG.SV
     DO.SV
     ...
   ```

IMOVE continues until it has loaded all of the files present on your second release diskette. Then, if the files in IMOVE format occupy more than one diskette, the utility prompts you to insert the second diskette in the IMOVE dump file—your third release diskette.

```
This disk has been exhausted
Please insert labeled disk 2, then press NEW LINE. . .<NL>
```

This process goes on until IMOVE has loaded onto your disk all of the DG/RDOS system files that are not on the bootable release diskette.

Once you have transferred the DG/RDOS files to your disk, the basic installation process is complete. However, it is a good idea at this point to install your system's power-up diagnostics on your hard disk.

Installing Power-Up Diagnostics

To avoid having to load power-up diagnostics from your System Media diskette each time you power on your computer, install the diagnostics on your system disk.

NOTE: This procedure will work only if you established a coresident diagnostic area when you ran DKINIT.

Before you can install the power-up diagnostics, you must shut down your system and then boot your System Media diskette or tape. To shut down the operating system, you need to boot your master disk, at the end of which step you will see the `Filename?` prompt.

```
R
BOOT %MDIR%<NL>

Master device released

Filename?
```

At this point, hold down the Command key and press the Brk key in order to reach the virtual console.

To install the diagnostics, make sure that your System Media diskette is in drive DJ0, and then type 20H. Shortly, the system console displays the following message:

```
Copyright (C) Data General Corporation, 1988.

Do you want to install power-up code on your hard disk?

If power-up code is not installed on the hard disk, you will need to
insert this System Media diskette each time you power up. The disk
on which power-up code will be installed must have a coresident
diagnostic area reserved by the operating system software formatter.

Install power-up code (Y or N)?:
```

You want to install the power-up code, so type Y<NL>. The diagnostics installer now copies the power-up diagnostics from the System Media diskette to the coresident diagnostic area on your hard disk. In the future, the power-up code will run from this disk automatically, so you will not need to use your System Media diskette.

After the power-up code has been installed, the Automatic Program Load Menu appears. Press New Line to reach DG/RDOS's `Filename?` prompt, and then either press New Line again or wait for the automatic timeout feature to start the operating system.

Installation is now complete. However, you still need to run the SYSGEN and CONFIG programs to tailor your operating system to your particular hardware configuration. If you want to run these programs now, turn to Chapters 7 and 8. Otherwise, consult Chapter 10 for instructions on how to shut down your system.

-End of Chapter-

Chapter 4

Loading DG/RDOS on an MV/1000 DC, MV/1400 DC, MV/2000 DC, or MV/2500 DC System

Read this chapter when you have set up a new MV/1000 DC, MV/1400 DC, MV/2000 DC, or MV/2500 DC computer and want to install DG/RDOS from release diskettes or a release tape.

Before you can run the tailoring programs discussed in Chapters 7 and 8, you must load the DG/RDOS system files onto a fixed disk. The following sections cover the steps that constitute this loading process:

- Preparing Your Hardware
- Initializing Your Master Disk
- Installing a Bootstrap Root on Your Disk
- Transferring the Operating System Files to Your Disk
- Installing Microcode and Power-Up Diagnostics

The final section, "Installing DG/RDOS on a Second Fixed Disk," deals with the special case of loading DG/RDOS on disk DA1 on an MV/2000 DC or MV/2500 DC system.

Preparing Your Hardware

Here are the steps you must perform before you can boot a program from your release diskettes or tape. These instructions are based on the assumption that your system is shut down when you begin.

1. Supply power to all of the devices in your configuration that are not controlled by the power switch on your computer.
2. If you received your System Media on a diskette, insert that diskette in your diskette drive (DA10). If you received your System Media on a tape, insert the tape in your tape drive (UT0 or UT10). This diskette or tape contains your system's microcode and power-up diagnostics.

3. Press the computer unit's power button to turn the unit on. The button should illuminate and stay in. The On Line light on each terminal's keyboard should also glow. If any terminal's On Line light is not glowing, hold down the Cmd key and press the On Line key to put the terminal on-line.

   After you supply power to the computer, it begins a series of power-up tests and displays diagnostic messages similar to these:

```
TESTING...

        Model # 8936; System Processing Unit (SPU) Level A
                 ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789, PASSED

        -- Standard Hardware Tests Complete --

        -- Optional Hardware Tests Beginning --

        Model # 4560; Slot A; Async Communications Board
                 ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789, PASSED

        -- Optional Hardware Tests Complete --

        -- Memory Size is 4 Megabytes --
```

   Then the Automatic Program Load Menu appears.

```
                                        DD-MMM-YY HH:MM:SS

            Automatic Program Load Menu

1 Continue immediately with preset values
2 Change preset values

Loading with preset values will continue automatically
unless you respond within 45 seconds.

The default device is hard disk

For assistance, press the Help key (SHIFT-F1) or H


Enter choice [1]:
```

4. Type 2<NL> to select the menu that allows you to change the system's preset values.

```
 Change Preset Values Menu

  1 Continue the powerup
  2 Change the system date or time
  3 Start from a different device
  4 Change the default device
  5 Change the time-out delay for the Auto Program Load Menu
  6 Enter the SCP CLI
  7 Change the system console
  8 Select diagnostics sequence
  9 Configure parallel printer port
 10 Select positional tracking device for tablets
 11 Select system clock type

 For assistance, press the Help key (SHIFT-F1) or H
 To exit from this menu, press the Cancel/Exit key (F11) or ESC

 Enter choice [1]:
```

5. Type 2<NL> to set the system date and time. The screen displays the following prompt:

   ```
   Date [DD-MMM-YY]:
   ```

   Type the date after the prompt and press New Line. You can enter the date in any of the following formats:

   ```
   28 APR 86
   28/APR/86
   28:APR:86
   28-APR-86
   ```

   The system then prompts you for the time.

   ```
   Time [HH:MM:SS]:
   ```

   Enter the time using the 24-hour format and press New Line. You can enter the time in several different ways:

   ```
   15:04:32
   17:49
   9:0
   ```

   If you omit seconds, the system assumes zeros.

   Finally, the system prompts you for the difference between local time and Greenwich Mean Time. DG/RDOS does not support this offset feature, so you may skip this question by pressing New Line. At this point, the Change Preset Values menu reappears.

6. If you have a parallel printer connected to the system board, you should now configure the printer port. To do so, select option 9 from the Change Preset Values Menu. The following menu appears:

```
Configure the Parallel Printer Port

 1 Positive strobe, VFU option, tab memory
 2 Positive strobe, no VFU option, tab memory
 3 Positive strobe, no VFU option, no tab memory
 4 Negative strobe, VFU option, tab memory
 5 Negative strobe, no VFU option, tab memory
 6 Negative strobe, no VFU option, no tab memory

 To exit from this menu, press the Cancel/Exit key (F11) or ESC
 For assistance, press the Help key (SHIFT-F1) or H

 Enter choice [1]:
```

Refer to the hardware startup manual for your system and your printer manual to determine which settings to use. Then enter your selection and press New Line. After you make this selection, the system console again displays the Change Preset Values Menu.

7. Type 6<NL> to enter the SCP CLI.

You are now ready to software format your system disk.

Initializing Your Master Disk

This section explains how to bring up and run the DG/RDOS disk initializer, DKINIT. During system generation, you use this program to write identifying information on block 3 of your disk, to check your disk for bad blocks, to set up a bad block table, to establish a remap area, and to define your disk's frame size. You can also use the program to reserve space for coresident diagnostics.

The instructions below are based on the assumption that when you begin, your system console is displaying the SCP CLI prompt.

Bringing Up DKINIT

The procedure for starting the disk initializer from a diskette is different from that for booting the utility from tape. So refer to the appropriate subsection below.

Starting DKINIT from a Diskette

1. Remove the System Media diskette from your diskette drive (DA10), and insert your first DG/RDOS diskette. Then type `BOOT 64<NL>` to execute the bootstrap code on that diskette. After a moment, you will see the prompt `Filename?`
2. To start DKINIT, type the command `DKINIT<NL>` within approximately 30 seconds.

Starting DKINIT from Tape

1. Remove the System Media tape from your tape drive (UT0 or UT10), and insert your DG/RDOS release tape. Then type `BOOT` *nn*`<NL>` where *nn* is the device code of the tape drive on which your release tape is mounted (see Appendix A). After a moment, you will see the prompt

   ```
   Number of tape file to boot? File =
   ```

2. DKINIT is in XFER format on tape file 4, so type `4<NL>` to start the disk initializer program.

Software Formatting Your Disk

Whether you started DKINIT from a diskette or tape, you should now see the following message:

```
Disk Initializer - Revision xx.xx

  *--> Help can be obtained to any question
  *--> simply respond to the question with HELP or ?

Disk drive model number?
```

To software format your disk, perform the steps listed below.

1. To answer DKINIT's first question, supply the model number of your fixed disk. If you do not know this model number, consult Appendix A or use DKINIT's help facility.

   ```
   Disk drive model number? 6329<NL>

   Disk unit?
   ```

   To identify the specific disk you want it to examine, DKINIT also needs to know the disk's unit name. Normally this unit name is DA0. Type in this identifier and press New Line.

   ```
   Disk unit? DA0<NL>

   6329   DAn      120 MB  5-1/4" sealed moving head disk

   Command?
   ```

   DKINIT displays a line of information about your disk and then asks for a command.

2. Since you are working with a new disk, issue DKINIT's FULL command and then answer the questions the utility poses. For complete information about running this part of the disk initializer program, consult the subsections "Coresident Diagnostics" and "The Rest of the FULL Command's Dialog" in Chapter 16. Or if you want to initialize your disk in a fairly standard way, you can use the sample dialog below as a guide.

   ```
   Command? FULL<NL>

   Command destroys any previous RDOS disk structure
   RDOS INIT/F must be done on the disk after the command
   Type CONTROL-A now to abort without loss
   ```

```
Do you want a coresident diagnostic area? YES<NL>
```

NOTE: You must declare a coresident diagnostic area if you want to avoid loading microcode and power-up diagnostics from diskette or tape every time you start up your system.

```
The default coresident diagnostic area size is 5000 blocks
How big would you like your diagnostic area to be? <NL>
```

NOTE: If you intend to install both System Media and User Friendly Diagnostics, you should request 10000 blocks instead of accepting the default.

```
Number of patterns to run (1-5)? 5<NL>

*** Pattern # 1 (022445) ***

*** Pattern # 2 (173366) ***

*** Pattern # 3 (170761) ***

*** Pattern # 4 (116234) ***

*** Pattern # 5 (144711) ***

*** All patterns run ***

Do you wish to declare any blocks bad
that are not already in the bad block table? NO<NL>

Default remap area size is x block(s) long
It needs to be at least y block(s) long
Remap area size (type RETURN for default) ? <NL>

Remap area start block number (type RETURN . . .) ? <NL>
Remap area will start at block x

Default frame size is x, min is 1, and max is y
Frame size (type RETURN for default) ? <NL>

Full disk init complete

Command?
```

3. When the utility has fully initialized your disk and prompts you for a new command, type STOP<NL> to leave DKINIT. After the program terminates, your master console will display the System Control Program prompt.

Installing a Bootstrap Root on Your Disk

By initializing your master disk, you have prepared it to hold its first few blocks of system software: the bootstrap root. After you have moved all the operating system files onto your disk, you can instruct your system to load this bootstrap code into memory and to transfer control to it. At that point the bootstrap root will enable you to read in and execute either a DG/RDOS operating system or a stand-alone utility such as DKINIT.

The exact instructions for installing a bootstrap root vary depending on whether you are loading DG/RDOS from diskette or tape. So read the appropriate subsection below.

Running the Disk Bootstrap Program from a Diskette

1. Make sure that your first release diskette is inserted in drive DA10. At the SCP prompt, type `BOOT 64<NL>`. The bootstrap root on your release diskette begins executing, and you see the `Filename?` prompt.

2. Type `DISKBOOT<NL>` within approximately 30 seconds to bring up the disk bootstrap program. You use this program to install a bootstrap root on the disk you just initialized.

3. As the bootstrap program begins execution, it prompts you for the unit name of the disk on which you want to install a bootstrap root:

   ```
   DISKBOOT Rev xx.xx

   Bootstrap Device Specifier ?
   ```

   Respond to this question by entering the same unit name that you used in your dialog with DKINIT.

   ```
   Bootstrap Device Specifier ? DA0<NL>

   Install Bootstrap (Y or N) ?
   ```

   The program now asks whether you want to install a bootstrap root on this disk. Type `Y` (pressing a New Line is not necessary). Doing so causes DISKBOOT to write bootstrap code to the first few blocks of your disk and then to display the message `Done`.

4. After installing a bootstrap root, DISKBOOT again prompts you for a unit name. This time enter the unit name of the drive in which you inserted your first release diskette (DA10) and press a New Line. Then, when the program asks whether you want to install bootstrap code on this disk, answer N (pressing a New Line is not needed). Following these steps takes you back to the Filename? prompt.

```
Bootstrap Device Specifier ? DA10<NL>

Install Bootstrap (Y or N) ? N

Filename?
```

Running the Disk Bootstrap Program from Tape

1. Make sure that your release tape is inserted in drive UT0 or UT10. At the SCP prompt, type BOOT *nn*<NL> where *nn* is the device code of the tape drive on which your release tape is mounted (see Appendix A). The bootstrap root on your release tape begins executing, and again you see the prompt

```
Number of tape file to boot? File =
```

2. To install a bootstrap root on the disk you just initialized, you must run a disk bootstrap program. This program, DISKBOOT.SV, is in XFER format on tape file 5, so to start the program type 5<NL>.
3. As the bootstrap program begins execution, it prompts you for the unit name of the disk on which you want to install a bootstrap root:

```
DISKBOOT Rev xx.xx

Bootstrap Device Specifier ?
```

Respond to this question by entering the same unit name that you used in your dialog with DKINIT.

```
Bootstrap Device Specifier ? DA0<NL>

Install Bootstrap (Y or N) ?
```

The program now asks whether you want to install a bootstrap root on this disk. Type Y (pressing a New Line is not necessary). Doing so causes DISKBOOT to write bootstrap code to the first few blocks of your disk and then to display the message Done.

4. After installing a bootstrap root, DISKBOOT again prompts you for a unit name and will install another bootstrap root if you ask it to. However, since the next step in the installation process is to boot the DG/RDOS starter system from tape, you need to leave the disk bootstrap program. Type Cmd-Brk to return to the SCP CLI. On some machines you must type this break sequence three times.

Transferring the Operating System Files to Your Disk

To finish installing DG/RDOS, you need to bring up the starter system you received on your first release diskette or your release tape, create master allocation and system file directories on your disk, and transfer the DG/RDOS system files to your newly formatted disk. If you are loading DG/RDOS from diskettes, consult the first subsection below, and if you are installing the system from tape, consult the second.

Loading the DG/RDOS System Files from Diskettes

As you begin this phase of the installation process, you should have your first release diskette in drive DA10. Your system console may be displaying the prompt `Filename?` If it is not displaying that prompt, step 1 has been done automatically; go on to step 2.

1. Type New Line to start the operating system named SYS.SV (linked to DGRDOS.SV), or wait for approximately 30 seconds and it will start automatically. The system announces itself with the message `DG/RDOS Rev xx.xx`. The system console then displays the date and time and the DG/RDOS Command Line Interpreter prompt R. The DG/RDOS starter system is now up and running.

2. Once the starter system is up, you can use a CLI command to establish a master allocation directory (MAP.DR) and system directory (SYS.DR) on your system disk. Use the command INIT/F followed by the disk unit name you used earlier in your dialogs with DKINIT and DISKBOOT. For instance, you might enter this command:

   ```
   R
   INIT/F DA0<NL>
   ```

   Since the CLI's full initialization effectively destroys all existing files on a disk, the system prompts you to confirm the command you just issued. Respond by typing Y (a New Line is not necessary).

   ```
   Confirm? Yes
   ```

   NOTE: The system supplies the letters es.

   Your system disk is now ready to hold the operating system files you received on your release diskettes.

3. Transferring the DG/RDOS system files to your newly formatted disk is a two-stage process. Since the system files on your first release diskette are in an executable format, you transfer these files to your disk using the CLI command MOVE. The remaining system files are in IMOVE (dump) format; therefore, you must boot the DG/RDOS starter system from your system disk and then use the IMOVE utility to load these files into your master directory.

To carry out the first of these two steps, ensure that your first release diskette is in drive DA10, and then issue a MOVE command that follows this format:

MOVE/V device_name

For example, to move all of the nonpermanent files on your first release diskette to your first, or only, fixed disk, you would type

```
MOVE/V DA0<NL>
```

As the system moves these files, it displays their names on your system console:

```
DGRDOS.SV
DGRDOS.OL
CLI.SV
IMOVE.SV
...
R
```

4. Now, before you can load the remainder of the DG/RDOS system files, you must shut down the starter system and restart DG/RDOS from your master disk. To do this, use the CLI's BOOT command:

   BOOT disk-name:SYS

   For instance, to boot the starter system from your first, or only, fixed disk, you would type

   ```
   BOOT DA0:SYS<NL>
   ```

   In response to this command, the current system displays the message `Master device released`, and then the new starter system comes up. At the end of this step, you should see the CLI's R prompt.

5. From this point, you issue a single IMOVE command to load the remaining DG/RDOS files:

   ```
   IMOVE/A/F/V DA10<NL>
   ```

   The IMOVE utility responds to this command by prompting you to insert the first diskette in the IMOVE dump file—that is, your second release diskette—and then to enter a New Line. Follow these instructions, and IMOVE begins loading files onto your master disk and displaying the names of those files on your system console.

```
Please insert labeled disk 1, then press NEW LINE . . .<NL>
30-APR-89 16:01:22
  68000RS.SV
  ALPHARS.SV
  BURST.SV
  ...
```

IMOVE continues until it has loaded all of the files present on your second release diskette. Then, if the files in IMOVE format occupy more than one diskette, the utility prompts you to insert the second diskette in the IMOVE dump file—your third release diskette.

```
This disk has been exhausted
Please insert labeled disk 2, then press NEW LINE . . .<NL>
```

This process goes on until IMOVE has loaded onto your disk all of the DG/RDOS system files that are not on the bootable release diskette.

Once you have transferred the DG/RDOS files to your disk, the basic installation process is complete. However, it is a good idea at this point to install your system's CPU instruction microcode and power-up diagnostics on your hard disk.

Loading the DG/RDOS System Files from Tape

As you begin this phase of the installation process, you should have your release tape in drive UT0 or UT10, and your system console should be displaying the SCP CLI prompt.

1. To execute the bootstrap code on tape file 0, type `BOOT` *nn*`<NL>` where *nn* is the device code of the tape unit on which your release tape is mounted (see Appendix A). TAPEBOOT.SV, the tape bootstrap program, responds by printing the message

   ```
   Number of tape file to boot? File =
   ```

2. You now supply the tape file number of the save file for the starter system: `2<NL>`. Then DG/RDOS comes up and asks which one of three operations you want it to carry out.

   ```
   Number of tape file to boot? File = 2<NL>

   Full (F), Restore (R), or Partial (P or -RETURN-)?
   ```

   If you have run DKINIT on your disk and installed a bootstrap root, you can use the `Full` option to install DG/RDOS for the first time. During this operation, the system establishes a master allocation directory and system directory on the target disk; it loads the contents of tape file 1 onto that disk; and it brings up the operating system and the CLI with the target disk as the master directory.

You use the other options—Restore and Partial—on disks that already contain DG/RDOS. The Restore operation initializes a disk for I/O without destroying the existing file system and then loads onto that disk the contents of tape file 1. If a file exists both on tape file 1 and on disk, the disk file is deleted and then the file is restored from tape. You might use this procedure to restore an operating system file that was deleted accidentally or to load a new revision of the operating system while preserving the other files on the disk. The Partial operation also initializes a disk for I/O without destroying the existing file system, but it does not load the contents of tape file 1.

3. Since you are initializing your disk and installing the operating system for the first time, choose a full initialization by typing F<NL>. DG/RDOS then asks you which disk to initialize. In answer, supply the unit name of the disk on which you just installed a bootstrap root, usually DA0.

```
Full (F), Restore (R), or Partial (P or -RETURN-)? F<NL>

Initializing what disk? DA0<NL>
```

The system now displays the date and time, loads the DG/RDOS files present on tape file 1 onto your disk, and establishes DA0 as your master directory. When you see the CLI prompt, R, you can use the LIST command to see the names of the files that have been moved:

```
R
LIST<NL>
DGRDOS.SV          31774  SD
DGRDOS.OL          26624  C
CLI.SV             38400  SD
. . .
```

4. The remaining operating system files are in IMOVE format on tape file 6. To load them onto your master disk, issue an IMOVE command similar to the following one:

```
IMOVE/F/T/V UT0:6<NL>
```

Once you enter this command, IMOVE begins loading files onto your master disk and displaying the names of those files on your system console.

```
30-APR-89 16:01:22
  68000RS.SV
  ALPHARS.SV
  BURST.SV
    . . .
```

When IMOVE finishes executing, the basic installation process is complete. However, it is a good idea at this point to install your system's CPU instruction microcode and power-up diagnostics on your hard disk.

Installing Microcode and Power-Up Diagnostics

To avoid having to load microcode and power-up diagnostics from your System Media diskette or tape each time you power on your computer, install the microcode and diagnostics on your system disk.

NOTE: This procedure will work only if you established a coresident diagnostic area when you ran DKINIT.

Before you can install the microcode and power-up diagnostics, you must shut down your system and then boot your "System Media" diskette or tape. To shut down the operating system, you need to boot your master disk. This causes the `Filename?` prompt to appear.

```
R
BOOT %MDIR%<NL>

Master device released

Filename?
```

At this point, hold down the Cmd key and press the Brk key (you may have to type this break sequence three times) in order to reach the SCP CLI.

To install the diagnostics, make sure that your System Media diskette or tape is in the appropriate drive, and then type `BOOT` *nn*`<NL>` where *nn* is the device code of the diskette or tape unit on which your release media is mounted (see Appendix A). The system console will display a message like the following:

```
Do you want to install powerup diagnostics on your hard
disk? If these diagnostics are not installed on the hard
disk, you will need to insert this media each time you
power up.  For the diagnostics to work, the disk on which
they will be installed must have a Diagnostic Area reserved
by the operating system's software formatter.

Install powerup diagnostics (Y or N)? [Y]
```

You want to install the diagnostics, so choose the default answer, yes, by pressing New Line. The diagnostics installer now copies the power-up diagnostics from the System Media diskette or tape to the coresident diagnostic area on your hard disk. In the future, the diagnostics will run from this disk automatically, so you will not need to use your System Media diskette or tape.

After the diagnostics have been installed, the Automatic Program Load Menu reappears. Wait 45 seconds, and you will see DG/RDOS's `Filename?` prompt. Either press New Line to start DG/RDOS immediately or wait for approximately 30 seconds for DG/RDOS to start automatically.

Installation is now complete. However, you still need to run the SYSGEN and CONFIG programs to tailor your operating system to your particular hardware configuration. If you want to run these programs now, turn to Chapters 7 and 8. Otherwise, consult Chapter 10 for instructions on how to shut down your system.

Installing DG/RDOS on a Second Fixed Disk

If you have an MV/2000 DC or MV/2500 DC system that includes two fixed disks, you can, if you desire, install DG/RDOS on the second disk (mnemonic DA1). To do so, follow the directions provided in the first five sections of this chapter, with the following exceptions:

1. Assuming that you have installed microcode and power-up diagnostics in a coresident diagnostic area on DA0, you need to perform only the following steps from the first section, "Preparing Your Hardware":

   a. Turn on your peripherals, and put them on-line if necessary.

   b. Turn your computer on, and monitor the screen until the Automatic Program Load Menu appears.

   c. Type 2<NL> to reach the Change Preset Values Menu.

   d. Type 6<NL> to enter the SCP CLI.

   Steps a through d above correspond to steps 1, 3, 4, and 7 in the section "Preparing Your Hardware."

2. While performing the steps explained in the section "Initializing Your Master Disk," make sure that you format DA1, not DA0, and answer NO<NL> when DKINIT asks whether you want to set up a coresident diagnostic area.

3. Ask DISKBOOT, which is discussed in the section "Installing a Bootstrap Root on Your Disk," to write bootstrap code to DA1 instead of DA0.

4. Follow the directions in the section "Transferring the Operating System Files to Your Disk" up to the point where you have initialized your disk and moved the starter system to it. Of course, you should initialize and move the files to DA1.

   If you are installing the operating system from a release tape, you are now running the starter system from DA1, so you can go ahead and use the IMOVE utility to load the remaining system files.

   If you are installing DG/RDOS from release diskettes, the procedure is a bit more involved. While you are running the CLI from DA10, type BOOT DA10<NL>. Shortly, you will see the prompt Filename? To bring up the starter system on DA1, type DA1:SYS<NL>. Then, when you see the CLI prompt R, use the IMOVE utility to load the remaining system files onto DA1.

5. Skip the fifth section, "Installing Microcode and Power-Up Diagnostics," altogether. Your system's microcode and power-up diagnostics need to reside on DA0 only.

For instructions on how to bring up a DG/RDOS system you have installed on the second fixed disk of an MV/2000 DC or MV/2500 DC system, see Chapter 10.

-End of Chapter-

Chapter 5

Loading DG/RDOS on an MV/3500 DC or MV/5000 DC Series System

Read this chapter after (1) you have set up a new MV/3500 DC or MV/5000 DC Series computer and (2) you have powered up your system for the first time according to the instructions in your hardware startup manual. From this point, you can install DG/RDOS on a fixed disk by following the steps detailed in the following sections:

- Initializing Your Master Disk
- Installing a Bootstrap Root on Your Disk
- Transferring the Operating System Files to Your Disk
- Installing Microcode

NOTE: For the following procedures to work when installing from tape, your tape drive must be unit 0 on its device code.

Initializing Your Master Disk

This section explains how to bring up and run the DG/RDOS disk initializer, DKINIT. During system generation, you use this program to write identifying information on block 3 of your disk, to check your disk for bad blocks, to set up a bad block table, to establish a remap area, and to define your disk's frame size. You can also use the program to reserve space for coresident diagnostics.

By following the directions in your hardware startup manual for powering up your system for the first time, you have

1. Powered up your system.
2. Used the Change Preset Values menu to set the date and time and to configure your parallel printer.
3. Inserted and booted a System Media diskette or tape and then used the Microcode Update and Installation menu to load microcode into memory.
4. Removed the System Media diskette or tape, inserted your first DG/RDOS release diskette or your DG/RDOS release tape, and then used the Microcode Update and Installation menu to boot your DG/RDOS release media.

Bringing Up DKINIT

The procedure for starting the disk initializer from a diskette is different from that for booting the utility from tape. So refer to the appropriate subsection below.

Starting DKINIT from a Diskette

1. You should see the prompt

   ```
   Filename?
   ```

2. To start DKINIT, type the command DKINIT<NL> within approximately 30 seconds. If you do not respond quickly enough, you will see

   ```
   DG/RDOS Rev xx.xx
   Date (m/d/y) ? mm/dd/yy
   Time (h:m:s) ? hh:mm:ss

   R
   ```

   In this case, type the command BOOT DKINIT<NL>.

Starting DKINIT from Tape

1. You should see the prompt

   ```
   Number of tape file to boot? File =
   ```

2. DKINIT is in XFER format on tape file 4, so type 4<NL> to start the disk initializer program.

Software Formatting Your Disk

Whether you started DKINIT from a diskette or tape, you should now see the following message:

```
Disk Initializer - Revision xx.xx

  *--> Help can be obtained to any question
  *--> simply respond to the question with HELP or ?

Disk drive model number?
```

To software format your disk, perform the steps listed below.

1. To answer DKINIT's first question, supply the model number of your fixed disk. If you do not know this model number, consult Appendix A or use DKINIT's help facility.

   ```
   Disk drive model number? 6662<NL>

   Disk unit?
   ```

   To identify the specific disk you want it to examine, DKINIT also needs to know the disk's unit name. Normally this unit name is DA0. Type in this identifier and press New Line.

   ```
   Disk unit? DA0<NL>

   6662   DAn      330 MB  5-1/4" sealed moving head disk

   Command?
   ```

   DKINIT displays a line of information about your disk and then asks for a command.

2. Since you are working with a new disk, issue DKINIT's FULL command and then answer the questions the utility poses. For complete information about running this part of the disk initializer program, consult the subsections "Coresident Diagnostics" and "The Rest of the FULL Command's Dialog" in Chapter 16. Or if you want to initialize your disk in a fairly standard way, you can use the sample dialog below as a guide.

   ```
   Command? FULL<NL>

   Command destroys any previous RDOS disk structure
   RDOS INIT/F must be done on the disk after the command
   Type CONTROL-A now to abort without loss
   ```

```
Do you want a coresident diagnostic area? YES<NL>
```

NOTE: You should request a coresident diagnostic area into which you can install User Friendly Diagnostics. The appropriate size for this diagnostic area is specified below.

If you choose not to install User Friendly Diagnostics, do not request a coresident diagnostic area. You don't need a coresident diagnostic area to automatically load your microcode file each time you power up your system.

```
The default coresident diagnostic area size is 5000 blocks
How big would you like your diagnostic area to be? 10000<NL>
```

You should request 10000 blocks instead of accepting the default.

```
Number of patterns to run (1-5)? 5<NL>

*** Pattern # 1 (022445) ***

*** Pattern # 2 (173366) ***

*** Pattern # 3 (170761) ***

*** Pattern # 4 (116234) ***

*** Pattern # 5 (144711) ***

*** All patterns run ***

Do you wish to declare any blocks bad
that are not already in the bad block table? NO<NL>

Default remap area size is x block(s) long
It needs to be at least y block(s) long
Remap area size (type RETURN for default) ? <NL>

Remap area start block number (type RETURN . . .) ? <NL>
Remap area will start at block x

Default frame size is x, min is 1, and max is y
Frame size (type RETURN for default) ? <NL>

Full disk init complete

Command?
```

3. When the utility has fully initialized your disk and prompts you for a new command, type STOP<NL> to leave DKINIT. After the program terminates, your master console will display the System Control Program prompt.

Installing a Bootstrap Root on Your Disk

By initializing your master disk, you have prepared it to hold its first few blocks of system software: the bootstrap root. After you have moved all the operating system files onto your disk, you can instruct your system to load this bootstrap code into memory and to transfer control to it. At that point the bootstrap root will enable you to read in and execute either a DG/RDOS operating system or a stand-alone utility such as DKINIT.

The exact instructions for installing a bootstrap root vary depending on whether you are loading DG/RDOS from diskette or tape. So read the appropriate subsection below.

Running the Disk Bootstrap Program from a Diskette

1. Make sure that your first release diskette is inserted in your diskette drive. At the SCP prompt, type `BOOT` *dc n*`<NL>` where *dc* is your diskette's device code and *n* is its unit number. The bootstrap root on your release diskette begins executing, and you see the `Filename?` prompt.

2. Type `DISKBOOT<NL>` within approximately 30 seconds to bring up the disk bootstrap program. You use this program to install a bootstrap root on the disk you just initialized.

3. As the bootstrap program begins execution, it prompts you for the unit name of the disk on which you want to install a bootstrap root:

   ```
   DISKBOOT Rev xx.xx

   Bootstrap Device Specifier ?
   ```

   Respond to this question by entering the same unit name that you used in your dialog with DKINIT.

   ```
   Bootstrap Device Specifier ? DA0<NL>

   Install Bootstrap (Y or N) ?
   ```

   The program now asks whether you want to install a bootstrap root on this disk. Type `Y` (pressing a New Line is not necessary). Doing so causes DISKBOOT to write bootstrap code to the first few blocks of your disk and then to display the message `Done`.

4. After installing a bootstrap root, DISKBOOT again prompts you for a unit name. This time enter the unit name of the drive in which you inserted your first release diskette and press a New Line. Then, when the program asks whether you want to install bootstrap code on this disk, answer N (pressing a New Line is not needed). Following these steps takes you back to the Filename? prompt. In this example, the diskette drive is device code 24, unit 1:

```
Bootstrap Device Specifier ? DA1<NL>

Install Bootstrap (Y or N) ? N

Filename?
```

Running the Disk Bootstrap Program from Tape

1. Make sure that your release tape is mounted, and at the SCP prompt, type BOOT *dc*<NL> where *dc* is the device code of the tape drive on which you have mounted your release tape. At this juncture, the bootstrap root on your release tape begins executing, and again you see the prompt

```
Number of tape file to boot? File =.
```

2. To install a bootstrap root on the disk you just initialized, you must run a disk bootstrap program. This program, DISKBOOT.SV, is in XFER format on tape file 5, so to start the program type 5<NL>.

3. As the bootstrap program begins execution, it prompts you for the unit name of the disk on which you want to install a bootstrap root:

```
DISKBOOT Rev xx.xx

Bootstrap Device Specifier ?
```

   Respond to this question by entering the same unit name that you used in your dialog with DKINIT.

```
Bootstrap Device Specifier ? DA0<NL>

Install Bootstrap (Y or N) ?
```

   The program now asks whether you want to install a bootstrap root on this disk. Type Y (pressing a New Line is not necessary). Doing so causes DISKBOOT to write bootstrap code to the first few blocks of your disk and then to display the message Done.

4. After installing a bootstrap root, DISKBOOT again prompts you for a unit name and will install another bootstrap root if you ask it to. However, since the next step in the installation process is to boot the DG/RDOS starter system from tape, you need to leave the disk bootstrap program. Type Cmd-Brk to return to the SCP CLI. On some machines you must type this break sequence three times.

5. At this point, the CPU is still running, so enter the SCP-CLI command HALT<NL> or RESET<NL>.

Transferring the Operating System Files to Your Disk

To finish installing DG/RDOS, you need to bring up the starter system you received on your first release diskette or your release tape, create master allocation and system file directories on your disk, and transfer the DG/RDOS system files to your newly formatted disk. If you are loading DG/RDOS from diskettes, consult the first subsection below, and if you are installing the system from tape, consult the second.

Loading the DG/RDOS System Files from Diskettes

As you begin this phase of the installation process, you should have your first release diskette in your diskette drive. Your system console may be displaying the prompt `Filename?` If it is not displaying that prompt, step 1 has been done automatically; go on to step 2.

1. Type New Line to start the operating system named SYS.SV (linked to DGRDOS.SV), or wait for approximately 30 seconds and it will start automatically. The system announces itself with the message `DG/RDOS Rev xx.xx`. The system console then displays the date and time and the DG/RDOS Command Line Interpreter prompt R. The DG/RDOS starter system is now up and running.

2. Once the starter system is up, you can use a CLI command to establish a master allocation directory (MAP.DR) and system directory (SYS.DR) on your system disk. Use the command INIT/F followed by the disk unit name you used earlier in your dialogs with DKINIT and DISKBOOT. For instance, you might enter this command:

   ```
   R
   INIT/F DA0<NL>
   ```

   Since the CLI's full initialization effectively destroys all existing files on a disk, the system prompts you to confirm the command you just issued. Respond by typing `Y` (a New Line is not necessary).

   ```
   Confirm? Yes
   ```

   NOTE: The system supplies the letters `es`.

   Your system disk is now ready to hold the operating system files you received on your release diskettes.

3. Transferring the DG/RDOS system files to your newly formatted disk is a two-stage process. Since the system files on your first release diskette are in an executable format, you transfer these files to your disk using the CLI command MOVE. The remaining system files are in IMOVE (dump) format; therefore, you must boot the DG/RDOS starter system from your system disk and then use the IMOVE utility to load these files into your master directory.

To carry out the first of these two steps, ensure that your first release diskette is in your diskette drive, and then issue a MOVE command that follows this format:

MOVE/V device_name

For example, to move all of the nonpermanent files on your first release diskette to your fixed disk, you might type

```
MOVE/V DA0<NL>
```

As the system moves these files, it displays their names on your system console:

```
DGRDOS.SV
DGRDOS.OL
CLI.SV
IMOVE.SV
...
R
```

4. Now, before you can load the remainder of the DG/RDOS system files, you must shut down the starter system and restart DG/RDOS from your master disk. To do this, use the CLI's BOOT command:

   BOOT disk-name:SYS

   For instance, you might type

   ```
   BOOT DA0:SYS<NL>
   ```

   In response to this command, the current system displays the message `Master device released`, and then the new starter system comes up. At the end of this step, you should see the CLI's R prompt.

5. From this point, you issue a single IMOVE command to load the remaining DG/RDOS files. In this example, the diskette drive is device code 24, unit 1:

   ```
   IMOVE/A/F/V DA1<NL>
   ```

   The IMOVE utility responds to this command by prompting you to insert the first diskette in the IMOVE dump file—that is, your second release diskette—and then to enter a New Line. Follow these instructions, and IMOVE begins loading files onto your master disk and displaying the names of those files on your system console.

   ```
   Please insert labeled disk 1, then press NEW LINE . . .<NL>
   30-APR-89 16:01:22
     68000RS.SV
     ALPHARS.SV
     BURST.SV
     ...
   ```

IMOVE continues until it has loaded all of the files present on your second release diskette. Then, if the files in IMOVE format occupy more than one diskette, the utility prompts you to insert the second diskette in the IMOVE dump file—your third release diskette.

```
This disk has been exhausted
Please insert labeled disk 2, then press NEW LINE . . .<NL>
```

This process goes on until IMOVE has loaded onto your disk all of the DG/RDOS system files that are not on the bootable release diskette.

Once you have transferred the DG/RDOS files to your disk, the basic installation process is complete. However, it is a good idea at this point to install your system's CPU instruction microcode on your hard disk.

Loading the DG/RDOS System Files from Tape

As you begin this phase of the installation process, you should have your release tape in the tape drive, and your system console should be displaying the SCP CLI prompt.

1. To execute the bootstrap code on tape file 0, type `BOOT` *dc*`<NL>` where *dc* is the device code of the tape unit on which your release tape is mounted (see Appendix A). TAPEBOOT.SV, the tape bootstrap program, responds by printing the message

   ```
   Number of tape file to boot? File =
   ```

2. You now supply the tape file number of the save file for the starter system: `2<NL>`. Then DG/RDOS comes up and asks which one of three operations you want it to carry out.

   ```
   Number of tape file to boot? File = 2<NL>

   Full (F), Restore (R), or Partial (P or -RETURN-)?
   ```

   If you have run DKINIT on your disk and installed a bootstrap root, you can use the `Full` option to install DG/RDOS for the first time. During this operation, the system establishes a master allocation directory and system directory on the target disk; it loads the contents of tape file 1 onto that disk; and it brings up the operating system and the CLI with the target disk as the master directory.

You use the other options—Restore and Partial—on disks that already contain DG/RDOS. The Restore operation initializes a disk for I/O without destroying the existing file system and then loads onto that disk the contents of tape file 1. If a file exists both on tape file 1 and on disk, the disk file is deleted and then the file is restored from tape. You might use this procedure to restore an operating system file that was deleted accidentally or to load a new revision of the operating system while preserving the other files on the disk. The Partial operation also initializes a disk for I/O without destroying the existing file system, but it does not load the contents of tape file 1.

3. Since you are initializing your disk and installing the operating system for the first time, choose a full initialization by typing F<NL>. DG/RDOS then asks you which disk to initialize. In answer, supply the unit name of the disk on which you just installed a bootstrap root.

```
Full (F), Restore (R), or Partial (P or -RETURN-)? F<NL>

Initializing what disk? DA0<NL>
```

The system now displays the date and time, loads the DG/RDOS files present on tape file 1 onto your disk, and establishes DA0 as your master directory. When you see the CLI prompt, R, you can use the LIST command to see the names of the files that have been moved:

```
R
LIST<NL>
DGRDOS.SV          31774  SD
DGRDOS.OL          26624  C
CLI.SV             38400  SD
. . .
```

4. The remaining operating system files are in IMOVE format on tape file 6. To load them onto your master disk, issue an IMOVE command similar to the following one:

```
IMOVE/F/T/V UT0:6<NL>
```

Once you enter this command, IMOVE begins loading files onto your master disk and displaying the names of those files on your system console.

```
30-APR-89 16:01:22
  68000RS.SV
  ALPHARS.SV
  BURST.SV
  . . .
```

When IMOVE finishes executing, the basic installation process is complete. However, it is a good idea at this point to install your system's CPU instruction microcode on your hard disk.

Installing Microcode

To avoid having to load microcode from your SCP System Media tape each time you power on your computer, you must install a microcode file on your system disk. To do this, follow the procedure described in the chapter on powering up for the first time in your hardware manual.

This is a summary of the procedure. Remove the DG/RDOS release tape or last DG/RDOS release diskette and insert the appropriate SCP System Media tape or diskette in the relevant drive. The microcode file is in IMOVE format; if you are using a tape, it is file 1, so you would enter a command similar to the following one:

```
IMOVE/F/T/V UT0:1<NL>
```

If you are using a diskette, issue a command similar to this:

```
IMOVE/F/U/V DA<i>n</i><NL>
```

where *n* is the unit number.

If the name of the microcode file is a legal DG/RDOS filename, IMOVE simply loads the file and returns you to the CLI. On the other hand, if the name of the file is not a legal DG/RDOS filename, IMOVE displays a message similar to the following one:

```
Illegal file name: MV35.MCF
Enter new filename (NL to skip):
```

Simply enter a legal filename, such as MV35.MF, and press New Line. IMOVE now loads the file and returns you to the CLI.

The final step in installing your microcode is to create a link file that links the name SYS.MF to your microcode file. Each time you power up your system, the disk bootstrap looks for SYS.MF and, if it finds the link, tries to load your microcode from the disk file. To create the link, enter a command that follows this format:

```
LINK  SYS.MF  microcode-file-name
```

If you don't want to install User Friendly Diagnostics, installation is now complete. However, you still need to run the SYSGEN and CONFIG programs to tailor your operating system to your particular hardware configuration. If you want to run these programs now, turn to Chapters 7 and 8. Otherwise, consult Chapter 10 for instructions on how to shut down your system.

If you do want to install User Friendly Diagnostics, first you should run the SYSGEN and CONFIG programs (see Chapters 7 and 8) to tailor your operating system to your particular hardware configuration.

Then, shut down your system and return to the SCP-CLI. Refer to "Shutting Down Your System" in Chapter 10 for instructions on how to do this.

Finally, consult the documentation that came with your User Friendly Diagnostics for installation information.

-End of Chapter-

Chapter 6

Loading DG/RDOS on an MV/7800 XP, MV/9500, MV/15000, or MV/18000 System

Read this chapter when you have set up a new MV/7800 XP, MV/9500, MV/15000, or MV/18000 computer and want to install DG/RDOS from the release tape.

Before you can run the tailoring programs discussed in Chapters 7 and 8, you must load the DG/RDOS system files onto a fixed disk. The following sections cover the steps that constitute this loading process:

- Preparing Your Hardware
- Initializing Your Master Disk
- Installing a Bootstrap Root on Your Disk
- Transferring the Operating System Files to Your Disk
- Installing Microcode

Preparing Your Hardware

Here are the steps you must perform before you can boot a program from your release tape. These instructions are based on the assumption that your system is shut down when you begin.

1. Supply power to all of the devices in your configuration that are not controlled by the power switch on your computer.
2. Mount the tape labeled SCP System Media in your tape drive. This tape contains your system's microcode.
3. Press the computer unit's power button to turn the unit on. The On Line light on each terminal's keyboard should glow. If any terminal's On Line light is not glowing, hold down the Cmd key and press the On Line key to put the terminal on-line.

   After you supply power to the computer, it runs a series of power-up tests and then one of two things happens. You may see the Automatic Program Load Menu shown below. In this case you can skip the rest of step 3 and proceed with step 4.

Or, you may see the System Control Program prompt (SCP-CLI>, SCP-CLI/Jp0>, etc.). If you see this prompt enter the following command:

```
SCP-CLI> FLAGS AUTO YES<NL>
```

Switch the power to your computer off and back on. After the computer completes its power-up tests, you should see a menu similar to the one shown below:

```
                                        DD-MMM-YY HH:MM:SS

            Automatic Program Load Menu

1 Continue immediately with preset values
2 Change preset values

Loading with preset values will continue automatically
unless you respond within 45 seconds.

The default device is 24

Enter choice [1]:
```

4. Type 2<NL> to select the menu that allows you to change the system's preset values.

```
 Change Preset Values Menu

  1 Continue the powerup
  2 Change the system date or time
  3 Start from a different device
  4 Change the default device
  5 Change the time-out delay
  6 Enter the SCP CLI


 Enter choice [1]:
```

5. Type 2<NL> to set the system date and time. The screen displays the following prompt:

   ```
   Date [DD-MMM-YY]:
   ```

   Type the date after the prompt and press New Line. You can enter the date in any of the following formats:

   ```
   28 APR 86
   28/APR/86
   28:APR:86
   28-APR-86
   ```

   The system then prompts you for the time.

   ```
   Time [HH:MM:SS]:
   ```

   Enter the time using the 24-hour format and press New Line. You can type the time in several different ways:

   ```
   15:04:32
   17:49
   9:0
   ```

   If you omit seconds, the system assumes zeros.

   Finally, the system prompts you for the difference between local time and Greenwich Mean Time. DG/RDOS does not support this offset feature, so you may skip this question by pressing New Line. At this point, the Change Preset Values menu reappears.

6. Type 4<NL> to set the default device. The system prompts you for a device code. Respond with the device code of the disk onto which you will be installing DG/RDOS.

7. Now type 6<NL> to enter the SCP-CLI so that you can boot your system's microcode. When you see the SCP's prompt, enter a command similar to the following one. The device code you supply on the command line should be the device code of the tape unit on which you have mounted your tape. See Appendix A for a complete list of device codes.

   ```
   SCP-CLI> BOOT 22<NL>
   ```

   You will see messages indicating that your system's microcode is being loaded, and then you will be returned to the SCP-CLI.

You are now ready to software format your system disk.

Initializing Your Master Disk

This section explains how to bring up and run the DG/RDOS disk initializer, DKINIT. During system generation, you use this program to write identifying information on block 3 of your disk, to check your disk for bad blocks, to set up a bad block table, to establish a remap area, and to define your disk's frame size. You can also use the program to reserve space for coresident diagnostics.

The instructions below are based on the assumption that when you begin, your system console is displaying the SCP CLI prompt.

Bringing Up DKINIT

1. Remove the SCP System Media tape from your tape drive and insert your DG/RDOS release tape. Then type `BOOT` *nn*`<NL>` where *nn* is the device code of the tape drive on which your release tape is mounted. After a moment, you will see the prompt `Number of tape file to boot? File =`.
2. DKINIT is in XFER format on tape file 4, so type `4<NL>` to start the disk initializer program.

Software Formatting Your Disk

You should now see the following message:

```
Disk Initializer - Revision xx.xx

  *--> Help can be obtained to any question
  *--> simply respond to the question with HELP or ?

Disk drive model number?
```

To software format your disk, perform the steps listed below.

1. To answer DKINIT's first question, supply the model number of a fixed disk. If you do not know this model number, consult Appendix A or use DKINIT's help facility.

   ```
   Disk drive model number? 6581<NL>

   Disk unit?
   ```

   To identify the specific disk you want it to examine, DKINIT also needs to know the disk's unit name. Type in this identifier and press New Line.

```
Disk unit? DA0<NL>

6581   DAn       500 MB  R.A.M.S. disk

Command?
```

DKINIT displays a line of information about your disk and then asks for a command.

2. Since you are working with a new disk, issue DKINIT's FULL command and then answer the questions the utility poses. For complete information about running this part of the disk initializer program, consult the subsections "Coresident Diagnostics" and "The Rest of the FULL Command's Dialog" in Chapter 16. Or if you want to initialize your disk in a fairly standard way, you can use the sample dialog below as a guide.

```
Command? FULL<NL>

Command destroys any previous RDOS disk structure
RDOS INIT/F must be done on the disk after the command
Type CONTROL-A now to abort without loss

Do you want a coresident diagnostic area? NO<NL>
```

NOTE: You should request a coresident diagnostic area only if you plan to load coresident diagnostics on your disk. You will not write your system's microcode file in such an area.

```
Number of patterns to run (1-5)? 5<NL>

*** Pattern # 1 (022445) ***

*** Pattern # 2 (173366) ***

*** Pattern # 3 (170761) ***

*** Pattern # 4 (116234) ***

*** Pattern # 5 (144711) ***

*** All patterns run ***

Do you wish to declare any blocks bad
that are not already in the bad block table? NO<NL>

Default remap area size is x block(s) long
It needs to be at least y block(s) long
Remap area size (type RETURN for default) ? <NL>
```

```
Remap area start block number (type RETURN . . .) ? <NL>
Remap area will start at block x

Default frame size is x, min is 1, and max is y
Frame size (type RETURN for default) ? <NL>

Full disk init complete

Command?
```

3. When the utility has fully initialized your disk and prompts you for a new command, type STOP<NL> to leave DKINIT. After the program terminates, your master console displays the System Control Program prompt.

Installing a Bootstrap Root on Your Disk

By initializing your master disk, you have prepared it to hold its first few blocks of system software: the bootstrap root. After you have moved all the operating system files onto your disk, you can instruct your system to load this bootstrap code into memory and to transfer control to it. At that point the bootstrap root will enable you to read in and execute either a DG/RDOS operating system or a stand-alone utility such as DKINIT.

To install the bootstrap root, follow the directions below:

1. Make sure that your release tape is mounted, and at the SCP prompt, type BOOT *nn*<NL> where *nn* is the device code of the tape drive on which you have mounted your release tape. At this juncture, the bootstrap root on your release tape begins executing, and again you see the prompt `Number of tape file to boot? File =.`

2. To install a bootstrap root on the disk you just initialized, you must run a disk bootstrap program. This program, DISKBOOT.SV, is in XFER format on tape file 5, so to start the program type 5<NL>.

3. As the bootstrap program begins execution, it prompts you for the unit name of the disk on which you want to install a bootstrap root:

```
DISKBOOT Rev xx.xx

Bootstrap Device Specifier ?
```

Respond to this question by entering the same unit name that you used in your dialog with DKINIT.

```
Bootstrap Device Specifier ? DA0<NL>

Install Bootstrap (Y or N) ?
```

The program now asks whether you want to install a bootstrap root on this disk. Type Y (pressing a New Line is not necessary). Doing so causes DISKBOOT to write bootstrap code to the first few blocks of your disk and then to display the message Done.

4. After installing a bootstrap root, DISKBOOT again prompts you for a unit name and will install another bootstrap root if you ask it to. However, since the next step in the installation process is to boot the DG/RDOS starter system from tape, you need to leave the disk bootstrap program. Type Cmd-Brk to return to the SCP CLI. On some machines you must type this break sequence three times.

5. At this point, the CPU is still running, so enter the SCP-CLI command HALT<NL> or RESET<NL>.

Transferring the Operating System Files to Your Disk

To finish installing DG/RDOS, you need to bring up the starter system you received on your release tape, create master allocation and system file directories on your disk, and transfer the DG/RDOS system files to your newly formatted disk.

As you begin this phase of the installation process, you should have your release tape in the tape drive, and your system console should be displaying the SCP CLI prompt.

1. To execute the bootstrap code on tape file 0, type BOOT *nn*<NL> where *nn* is the device code of the tape unit on which your release tape is mounted. TAPEBOOT.SV, the tape bootstrap program, responds by printing the message Number of tape file to boot? File =.

2. You now supply the tape file number of the save file for the starter system: 2<NL>. Then DG/RDOS comes up and asks which one of three operations you want it to carry out.

```
Number of tape file to boot? File = 2<NL>

Full (F), Restore (R), or Partial (P or -RETURN-)?
```

   If you have run DKINIT on your disk and installed a bootstrap root, you can use the Full option to install DG/RDOS for the first time. During this operation, the system establishes a master allocation directory and system directory on the target disk; it loads the contents of tape file 1 onto that disk; and it brings up the operating system and the CLI with the target disk as the master directory.

You use the other options—Restore and Partial—on disks that already contain DG/RDOS. The Restore operation initializes a disk for I/O without destroying the existing file system and then loads onto that disk the contents of tape file 1. If a file exists both on tape file 1 and on disk, the disk file is deleted and then the file is restored from tape. You might use this procedure to restore an operating system file that was deleted accidentally or to load a new revision of the operating system while preserving the other files on the disk. The Partial operation also initializes a disk for I/O without destroying the existing file system, but it does not load the contents of tape file 1.

3. Since you are initializing your disk and installing the operating system for the first time, choose a full initialization by typing F<NL>. DG/RDOS then asks you which disk to initialize. In answer, supply the unit name of the disk on which you just installed a bootstrap root.

```
Full (F), Restore (R), or Partial (P or -RETURN-)? F<NL>

Initializing what disk? DA0<NL>
```

The system now displays the date and time, loads the DG/RDOS files present on tape file 1 onto your disk, and establishes DA0 as your master directory. When you see the CLI prompt, R, you can use the LIST command to see the names of the files that have been moved:

```
R
LIST<NL>
DGRDOS.SV           31774  SD
DGRDOS.OL           26624  C
CLI.SV              38400  SD
. . .
```

4. The remaining operating system files are in IMOVE format on tape file 6. To load them onto your master disk, issue an IMOVE command similar to the following one:

```
IMOVE/F/T/V MT0:6<NL>
```

Once you enter this command, IMOVE begins loading files onto your master disk and displaying the names of those files on your system console.

```
30-APR-89 16:01:22
  68000RS.SV
  ALPHARS.SV
  BURST.SV
  ...
```

When IMOVE finishes executing, the basic installation process is complete. However, it is a good idea at this point to install your system's CPU instruction microcode on your hard disk.

Installing Microcode

To avoid having to load microcode from your SCP System Media tape each time you power on your computer, you must install a microcode file on your system disk. To do this, remove the DG/RDOS release tape from your tape drive and mount your microcode tape. The microcode file is in IMOVE format and is file 1 on the tape, so enter a command similar to the following one:

```
IMOVE/F/T/V MT0:1<NL>
```

If the name of the microcode file is a legal DG/RDOS filename, IMOVE simply loads the file and returns you to the CLI. On the other hand, if the name of the file is not a legal DG/RDOS filename, IMOVE displays a message similar to the following one:

```
Illegal file name: MV_15000_20.MCF
Enter new filename (NL to skip):
```

Simply enter a legal filename, such as MV1500020.MF, and press New Line. IMOVE now loads the file and returns you to the CLI.

The final step in installing your microcode is to create a link file that links the name SYS.MF to your microcode file. Each time you power up your system, the disk bootstrap looks for SYS.MF and, if it finds the link, tries to load your microcode from the disk file. To create the link, enter a command that follows this format:

```
LINK  SYS.MF  microcode-file-name
```

Installation is now complete. However, you still need to run the SYSGEN and CONFIG programs to tailor your operating system to your particular hardware configuration. If you want to run these programs now, turn to Chapters 7 and 8. Otherwise, consult Chapter 10 for instructions on how to shut down your system.

- End of Chapter -

Chapter 7

Running the SYSGEN Utility

Read this chapter when

- You have installed the DG/RDOS starter system from diskettes or tape and want to begin tailoring your system to match your hardware and application
- You have installed a DG/RDOS update or a new revision of the operating system and need to rebuild your tailored system
- You have installed a new piece of hardware and want to generate a system that supports that hardware
- You have removed a device from your configuration and want to build a system that will not think it supports a device that does not exist

The next three chapters discuss the steps involved in tailoring your DG/RDOS system to suit your hardware and application. This chapter explains how you use the SYSGEN utility, an interactive program that allows you to define which devices are in your configuration and then builds a system that supports those devices. Chapter 8 deals with the CONFIG utility, which lets you define a variety of characteristics for your system. For instance, the CONFIG utility allows you to set up the capability to run a second program in the foreground; it lets you decide how much memory to reserve for DG/RDOS use; and it allows you to define a set of characteristics for each multiplexor line you have. Chapter 9 discusses how to to use the PATCH utility to apply patches to DG/RDOS system files.

The information in this chapter is presented under the following headings:

- An Overview of SYSGEN
- Starting SYSGEN
- The SYSGEN Dialog
- SYSGEN's Error Messages

An Overview of SYSGEN

Once you have the DG/RDOS starter system—or a previously generated system—running, you can invoke the SYSGEN program. This utility asks you a series of questions designed primarily to ascertain which of the devices that your computer can support are actually in your configuration.

The program also asks whether you want your operating system to include some special features, such as the capability to produce a memory dump. Once you have answered these queries, SYSGEN builds a system that meets your specifications.

As it builds this system, SYSGEN produces several files:

- *system-name.SV*. This file contains your tailored DG/RDOS system.
- *system-name.OL*. This file contains the system overlays.
- *system-name.SG*. If you include on your SYSGEN command line the argument/local-switch combination *system-name.SG/V*, the utility records its questions and your answers in this dialog file. Such a file can serve as a useful record of the devices and features your operating system supports, and you can also use the file as input to SYSGEN when you run the utility in the future. If you do use the file as input to SYSGEN, you can either ask the program to build a system according to the specifications recorded in the dialog file or request that the utility run interactively using the values in your dialog file as the default answers to its questions.
- *system-name.LM*. If you include on your SYSGEN command line the argument/local-switch combination *system-name.LM/L*, SYSGEN writes to a load map file the names and values of all the entry points in your system and overlay files. It is a good idea to create this file because you must have a load map file to use the PATCH utility.

The next section explains how to start the SYSGEN utility and to request such options as dialog and load map files.

Starting SYSGEN

You can start the system-generation utility from the Command Line Interpreter by typing SYSGEN<NL>. However, chances are that you will want to use one or more of the program's switches on your command line to define the exact operation you want to perform. These switches are described in Tables 7-1 and 7-2.

Table 7-1 SYSGEN's Global Switches

Switch	Description
/H	Displays a help message that describes SYSGEN's global and local switches and shows some sample command lines.
/N	Instructs SYSGEN not to build a new system after you and the utility complete your dialog.
/R	Causes the program to build a system without an overlay file.

Table 7-2 SYSGEN's Local Switches

Argument/Switch	Description
system-name.SG/A	Tells SYSGEN to build a system according to the specifications recorded in a dialog file *(system-name.SG)* from a previous SYSGEN session. If you use this argument/local-switch combination, the utility does not run interactively. NOTE: Do not use the /A and /E switches on the same command line.
system-name.SG/E	Instructs SYSGEN to use the values recorded in a dialog file *(system-name.SG)* from a previous SYSGEN session as the default answers to the questions it will pose during the current session. NOTE: Do not use the /E and /A switches on the same command line.
system-name.LM/L	Causes SYSGEN to list in a load map file the names and values of all the entry points in your system and overlay files.
system-name.SV/S	Indicates the name you want to give to the new operating system. If you do not supply this argument, SYSGEN uses the name SYS000. NOTE: Unless you want to delete your DG/RDOS starter system, do not name the new system DGRDOS.SV.
system-name.SG/V	Instructs SYSGEN to record your dialog with the utility in a file named *system-name.SG*. During later SYSGEN sessions, you can use the values recorded in this dialog file as input to the program (see the descriptions of the /A and /E switches above).

If you are running SYSGEN on your system for the first time, you will probably use a command line similar to the following one:

```
SYSGEN system-name.<SV/S,LM/L,SG/V><NL>
```

This command ensures that SYSGEN will produce save and overlay files for your tailored system, a load map, and a dialog file. It also causes the program to begin the dialog that is explained in the next section.

The SYSGEN Dialog

As SYSGEN executes, it presents you with a series of questions about your configuration and the operating system you want to build. The utility lists the valid responses to each question in parentheses and presents the default answer in square brackets. To choose the default answer, simply press New Line. To select another valid answer, type in your response followed by New Line. If you are unsure how to answer any question, type ?<NL> to display help information about the subject of the question.

The following list describes in sequence the subjects that SYSGEN inquires about and makes some recommendations about how you should answer its questions.

1. The program first asks you to identify the type of computer you are generating a system for:

   ```
   Currently supported processors include:
   DG/10, DG/10-SP, DG/20, DG/30, DG/500
   MV/1000-DC, MV/1400-DC, MV/2000-DC, MV/2500-DC
   MV/3500-DC, MV/5500-DC
   MV/7800-XP, MV/9500
   MV/15000-8, MV/15000-10, MV/15000-20, MV/18000-1, MV/18000-SX
   Processor type ? [DG/500]
   ```

   To answer this question, enter one of the codes listed under the heading of currently supported processors and press the New Line key. The first four codes that begin with the letters DG represent DESKTOP GENERATION systems.

2. The utility then asks about the disk controllers and drives in your configuration.

   If you are working on a **DESKTOP GENERATION** or **DG/500** system, SYSGEN asks whether your system includes DEn controllers (for a list of the various disk, diskette, and tape drives, refer to Appendix A):

   ```
   Any DEn disk controllers (Yes or No) ? [Yes]
   ```

   If your system does not have a fixed disk, answer No. If it has one or two such drives, answer Yes. If you type Yes, the program goes on to ask how many drives are connected to your controllers:

   ```
   Number of units on first controller (0-2) ? [1]
   Number of units on second controller (0-2) ? [0]
   ```

   On DESKTOP GENERATION and DG/500 systems, all drives are on the first controller, so your answer for the number of units on the second controller should be zero.

SYSGEN then asks if your system includes any DJn controllers:

```
Any DJn diskette controllers (Yes or No) ? [Yes]
```

If your machine does not have a diskette drive, answer No, and if it has one or two drives, answer Yes. If you answer Yes, the utility also asks how many drives are connected to your diskette drive controllers:

```
Number of units on first controller (0-2) ? [2]
Number of units on second controller (0-2) ? [0]
```

Respond by entering the number of diskette drives in your configuration. On DESKTOP GENERATION and DG/500 systems, all drives are on the first controller, so your answer for the number of units on the second controller should be zero.

If you are working on an **MV/1000 DC**, **MV/1400 DC**, **MV/2000 DC**, **MV/2500 DC**, **MV/3500 DC**, or **MV/5000 DC Series** system, SYSGEN begins by asking if there are any DAn controllers in your system (for a list of the various disk, diskette, and tape drives, refer to Appendix A):

```
Any DAn disk(ette) controllers (Yes or No) ? [Yes]
```

If you answer No, you are telling SYSGEN that you do not want support for any DAn disk controllers. In this case, you must answer Yes when the utility asks if you want support for a RAM disk (discussed below). You can then boot DG/RDOS from tape and install the operating system on your RAM disk.

If you answer Yes, SYSGEN then asks these questions, regardless of the CPU type:

```
Number of units on first controller (0-8) ? [1]
Number of units on second controller (0-8) ? [1]
Number of units on third controller (0-8) ? [0]
Number of units on fourth controller (0-8) ? [0]
```

On MV/1000 DC, MV/1400 DC, MV/2000 DC, and MV/2500 DC systems, the first controller manages the hard disk drives in the main chassis, the second manages the diskette drive, and the third manages the hard disk drives in any expansion chassis. On MV/3500 DC and MV/5500 DC systems, the first controller manages all disk and diskette drives.

Answer 0 for controllers that are not in your configuration. For example, an MV/2500 DC configuration can have two DAn units on the first controller, one on the second, four on the third, and none on the fourth; you should answer 0 to the question about the fourth controller.

NOTE: If you plan to install DG/RDOS on a second fixed disk (DA1), you must request support for two fixed disks even if you will not access your first fixed disk (DA0) from DG/RDOS.

If you are working on an **MV/7800 XP**, **MV/9500**, **MV/15000**, or **MV/18000** system, SYSGEN begins by asking if there are any DAn controllers in your system (for a list of the various disk, diskette, and tape drives, refer to Appendix A):

```
Any DAn disk(ette) controllers (Yes or No) ? [Yes]
```

If you answer Yes, SYSGEN then asks these questions, regardless of the CPU type:

```
Number of units on first controller (0-8) ? [1]
Number of units on second controller (0-8) ? [1]
Number of units on third controller (0-8) ? [0]
Number of units on fourth controller (0-8) ? [0]
```

SYSGEN then asks if your system includes any DJn controllers:

```
Any DJn diskette controllers (Yes or No) ? [Yes]
```

If your machine does not have a 5.25-inch diskette drive, answer No; otherwise, answer Yes. If you answer Yes, the utility also asks how many drives are connected to your diskette drive controllers:

```
Number of units on first controller (0-2) ? [2]
Number of units on second controller (0-2) ? [0]
```

Respond by entering the number of diskette drives in your configuration.

Next, SYSGEN asks if your system includes any DZn controllers:

```
Any DZn disk controllers (Yes or No) ? [Yes]
```

If you answer Yes, the utility asks how many drives are connected to your controllers:

```
Number of units on first controller (0-4) ? [1]
Number of units on second controller (0-4) ? [0]
```

3. Unless you used the global /R switch on your SYSGEN command line, the utility now asks whether you would like to set aside a portion of system memory that DG/RDOS will treat as if it were a disk or a diskette. You can request only one RAM disk.

```
A DRn RAM disk controller (Yes or No) ? [Yes]
```

Type New Line if you want support for a RAM disk, and answer No if you do not. If you opt to use some of your system memory in this way, you will be asked to specify the number of pages you wish to reserve for this purpose when you run the CONFIG program. (If you are not sure whether you want support for a RAM disk, see Chapter 17.)

4. Next, SYSGEN asks a question designed to help it judge how much memory needs to be set aside to hold information about the bad blocks on the disks and diskettes in your system.

```
Bad block pool size in blocks (1-512) ? [50]
```

To answer this question, first recall the number of bad blocks present on each disk or diskette in your system, and then add these numbers. Your response should be a number equal to or, preferably, larger than this total. Using a slightly larger number will protect you from having to run SYSGEN again if one or more of your disks develops a new bad block or you one day use a diskette that has a lot of bad blocks. If you supply a number smaller than the total you calculated, you will not be able to initialize all the disks and diskettes in your configuration at the same time.

5. The utility then asks whether you want the operating system you are building to support one or more tape drives.

If you are working on a **DESKTOP GENERATION** or **DG/500** system, SYSGEN displays this prompt:

```
Any MTn tape controllers (Yes or No) ? [Yes]
```

If you answer Yes, SYSGEN then asks the following:

```
Number of units on first controller (0-8) ? [1]
Number of units on second controller (0-8) ? [0]
```

If you have a **DG/500** system, SYSGEN will also ask this question:

```
Any STn tape controllers (Yes or No) ? [No]
```

If you answer Yes, you are asked

```
Number of units on first controller (0-7) ? [1]
Number of units on second controller (0-7) ? [0]
```

If you are working on an **MV/1000 DC, MV/1400 DC, MV/2000 DC, MV/2500 DC, MV/3500 DC,** or **MV/5000 DC Series** system, the utility asks

```
Any UTn tape controllers (Yes or No) ? [Yes]
```

If you answer Yes, SYSGEN then asks these questions:

```
Number of units on first controller (0-8) ? [1]
Number of units on second controller (0-8) ? [0]
```

The first controller uses unit names starting at UT0, and the second uses unit names starting at UT10.

If you are working on an **MV/7800 XP**, **MV/9500**, **MV/15000**, or **MV/18000** system, SYSGEN displays the following:

```
Any MTn tape controllers (Yes or No) ? [Yes]
```

If you answer Yes, SYSGEN then asks the following:

```
Number of units on first controller (0-8) ? [1]
Number of units on second controller (0-8) ? [0]
```

You are then asked about UTn controllers:

```
Any UTn tape controllers (Yes or No) ? [Yes]
```

If you answer Yes, SYSGEN asks

```
Number of units on first controller (0-8) ? [1]
Number of units on second controller (0-8) ? [0]
```

6. At this point, the utility poses a series of questions that deal with support for multiplexors. The program first asks if you want your system to support QTY lines and, if you do want such support, how many lines you plan to use. Then, if you request support for QTY lines, SYSGEN asks whether you need modem support and how you plan to pass characters to your program's interrupt handler.

   NOTE: Do not request support for more than 64 QTY lines.

   a. If you are working on a **DESKTOP GENERATION** system, the first question SYSGEN poses concerns the number of Universal Synchronous/Asynchronous Multiplexors (USAMs) in your configuration:

      ```
      Number of USAM controllers (0-4) ? [0]
      ```

      If your system contains USAM boards and you plan to use one or more of them to support DG/RDOS QTY lines, enter the number of boards you will use for this purpose. If no USAM boards are present, or if you plan to introduce your multiplexors to DG/RDOS using the .IDEF system call, enter 0.

For example, if you intend to run a Business BASIC system that uses DG/RDOS's multiplexor driver, enter a number 1 through 4. On the other hand, if you plan to generate a Business BASIC system that includes its own multiplexor driver, enter 0.

Unless you answered 0 to the previous question, SYSGEN then asks several questions about your multiplexors. First, the utility prompts you for the device code for each USAM board; for example, to get the device code for your first board, SYSGEN asks,

```
Device code for first controller (1-76) ? [34]
```

You select this device code using switches on the board; then you should respond to the question by entering a value that corresponds to the switch settings on one, and only one, of the boards. You will be asked a similar question for each USAM board in the system.

If you are building an operating system that will run on a **DG/500** computer, SYSGEN begins by asking you how many asynchronous controllers you want DG/RDOS to support.

A DG/500 system can support up to four controllers, with eight lines on each. One controller (device code 34) resides on the system board, while another (device code 74) is a Model 4618 option card. The remaining two controllers must be Model 5618 option cards, and their switches must be set carefully; refer to the hardware documentation provided with the option card for switch information.

```
Number of async controllers (0-4) ? [0]
```

An answer of 0 tells the program that you do not want DG/RDOS support for any QTY lines.

Unless you answered 0 to the previous question, SYSGEN then asks several questions about your multiplexors. First, the utility prompts you for the device code of each controller; for example, to get the device code for your first controller, SYSGEN asks this question:

```
Device code for first controller (34-74) ? [34]
```

Respond with the device code of one of your controllers. You will be asked a similar question for each controller in the system.

If you are working on an **MV/1000 DC**, **MV/1400 DC**, **MV/2000 DC**, **MV/2500 DC**, **MV/3500 DC**, or **MV/5000 DC Series** system, SYSGEN'S first prompt asks whether your system includes any Local Asynchronous Controllers:

```
Number of LAC controllers (0-4) ? [0]
```

These controllers, also referred to as LMC cards, are intelligent devices that control asynchronous communications lines. If you have such boards and plan to use them to support DG/RDOS QTY lines, enter the number of boards you will use for this purpose. If no LACs are present, or if you plan to introduce your multiplexors to DG/RDOS using the .IDEF system call, enter 0.

NOTE: Treat the lines on an MV/3500 DC system board as an LMC-8_2 on device code 42. Treat the lines on an MV/5000 DC Series system board as a LAC-16 on device code 42.

If you don't request this support for your MV/3500 DC or MV/5000 DC Series system, the master console is the only asynchronous line on the system board that you will be able to use. If you do request this support, you must use CONFIG to link the background console to a QTY line.

If you provide an answer other than 0, SYSGEN asks you to enter a device code for each multiplexor board; for instance, for your first controller, the program asks,

```
Device code for first controller (40-53) ? [40]
```

The device code must be one of these values: 40, 41, 42, 46, 47, 50, 51, 52, or 53. To determine which device code you should use, see the "I/O Device Codes" appendix in the *Principles of Operation Supplement* for your system or see the documentation that came with your hardware.

SYSGEN then asks you to specify the controller type:

```
Type (LMC-8_1, LMC-8_2, LAC-12, LAC-16, or LAC-32)? [LAC-12]
```

If, on the other hand, you answer 0 to SYSGEN's question about LACs and you have an **MV/1000 DC**, **MV/1400 DC**, **MV/2000 DC**, or **MV/2500 DC** system, the utility inquires whether you would like to use the available Dual Universal Asynchronous Receiver/Transmitter ports on the system board as QTY lines:

```
Use DUART lines as QTY lines (Yes or No) ? [No]
```

If you respond NO, the master console is the only asynchronous line that you will be able to use.

If you respond YES, you must use CONFIG to link the background console to a QTY line.

If you are working on an **MV/7800 XP**, **MV/9500**, **MV/15000**, or **MV/18000** system, SYSGEN'S first prompt asks whether your system includes any IAC controllers:

```
Number of IAC controllers (0-4) ? [0]
```

If your answer is greater than zero, SYSGEN asks for the device code and type of each controller:

```
  Device code for first controller (30-74) ? [52]
    Controller type (MCP-1, IAC-8_3, or IAC-24) ? [IAC-24]
```

If you answered zero to the IAC question, SYSGEN asks this question:

```
Number of ALM/ULM controllers (0-2) ? [0]
```

If your response is greater than zero, SYSGEN asks for the device code, type, and number of lines for each controller:

```
  Device code for first controller (34-74) ? [34]
    Controller type (ALM or ULM) ? [ALM]
    Number of lines on this controller (4-64) ? [16]
```

b. If you have requested any kind of support for QTY lines, from this point on SYSGEN's prompts are the same for all computers and types of QTY devices. The utility first asks whether you want your system to include modem support:

```
  Modem support desired (Yes or No) ? [No]
```

If you intend to use a modem, answer YES<NL>, and the program inquires,

```
    Standard modem timer (Yes or No) ? [Yes]
```

You should respond YES to this question. This ensures that DG/RDOS will wait five seconds after receiving a signal that indicates a connection has been established or broken—to make sure the signal does not disappear or reappear—before acting on that signal.

Finally, if your system will support QTY lines, the program asks you whether you want to define two interrupt characters or use Ctrl-C key sequences to pass characters to your program's interrupt handler:

```
Use ^C^x interrupt character sequences (Yes or No) ? [Yes]
```

If you respond NO, the system asks you to specify two single-byte interrupt characters:

```
First interrupt character (1-127 or None) ? [1]
Second interrupt character (1-127 or None) ? [3]
```

Answer NO only if your application depends on single-byte interrupt characters, for example, if you define Function Key Header <36> as an interrupt character. Even then, you should plan to enhance your program to work with Ctrl-C character sequences because only these character sequences will be supported in the future.

If you respond YES, DG/RDOS passes the second byte of any valid Ctrl-C character sequence to your program's interrupt handler. To be valid the second byte must be in the range Ctrl-A to Ctrl-Z. You can choose to have the operating system map the ESC character to the Ctrl-C Ctrl-A sequence by opening a QTY line with the characteristics bits DCESC and DCNI set to 1.

7. The utility then asks if you want support for a foreground console:

   ```
   Foreground console (Yes or No) ? [No]
   ```

   Answer Yes if one of the following conditions applies:

   a. Your computer has a secondary console controller and you intend to use it, either for a foreground console or for a serial printer.

      A secondary console controller uses device codes 50 and 51. On a DG/10, DG/10-SP, or DG/500 system, a secondary console controller is built-in.

      On an MV/1000 DC, MV/1400 DC, MV/2000 DC, MV/2500 DC, MV/3500 DC, or MV/5000 DC Series system, a secondary console controller cannot exist.

      On other systems, a secondary console controller can exist on an option board.

   b. You intend to use the CONFIG utility to link the foreground console to a QTY line.

8. SYSGEN next inquires whether you want support for one or more parallel printer controllers:

```
Number of line printer controllers ?
```

On a **DG/500**, **MV/1000 DC**, **MV/1400 DC**, **MV/2000 DC**, **MV/2500 DC**, **MV/3500 DC**, or **MV/5000 DC Series** system, the possible answers are 0 and 1. Answer 1 only if you plan to connect a parallel printer to your system's printer port.

On a **DESKTOP GENERATION**, **MV/7800 XP**, **MV/9500**, **MV/15000**, or **MV/18000** system, the range of possible answers is 0 to 2. If you have a parallel printer interface card and plan to connect a parallel printer to it, answer 1. If you have two such cards and two parallel printers, answer 2. Otherwise, answer 0.

NOTE: An MCP-1 includes a parallel printer controller as well as its eight asynchronous lines.

If you have an **MV/7800 XP**, **MV/9500**, **MV/15000**, or **MV/18000** system and you indicate that you will have a printer on a parallel printer port, the utility asks if your controller uses a data channel interface:

```
Primary printer using DCH interface (Yes or No) ? [No]
```

Answer No if your controller uses a programmed I/O interface.

9. If you requested any kind of support for QTY lines, the utility asks

```
Number of extra linked $ devices (0 - 4) ? [0]
```

Respond with the number of QTY lines that you want to redefine as $ devices. These $ devices support spooling, Form Feed on open, and other features that QTY lines do not support. CONFIG lets you redirect background and foreground consoles to QTY lines, but do not include those consoles here.

By redefining a QTY line to be a $ device, you can make a printer attached to QTY hardware behave as if it were attached to a secondary console or parallel printer controller. For example, if you don't have $LPT hardware, you can redirect $LPT to QTY hardware.

Initially, linked $ devices are named $TTI2/$TTO2, $TTI3/$TTO3, and so on, and are linked to the first few QTY lines. CONFIG lets you rename linked $ devices and link them to other QTY lines.

Once you redefine a QTY line to be a $ device, you can no longer access that line using the filename QTY:*n*.

10. SYSGEN now asks about your system's clock:

```
Clock (0=none, 1=10Hz, 2=50Hz, 3=60Hz, 4=100Hz, 5=1000Hz)
(0-5) ? [1]
```

DG/RDOS uses a real-time or architectural clock to maintain the current date and time while the system is running and to schedule the various activities taking place in the system. You can also access the clock through your own programs by using system calls such as .DUCLK. Unless you want a special frequency for programs you code yourself, answer 1 for a 10 Hz clock.

11. Next, SYSGEN asks for the maximum number of user-defined devices your system will contain at any one time:

```
Number of user defined devices (0-64) ? [10]
```

Enter the maximum number of devices that your software could introduce to DG/RDOS through the .IDEF system call. Some languages and communications software products require that you allocate user-defined devices via this question.

12. Finally, the utility asks whether you want your operating system to be able to dump the contents of memory:

```
Memory dump (device name or None) ? [None]
```

To see what device names you can choose from, type ?<NL>.

If you do not want this capability, press New Line to accept the default answer, and if you do, enter the unit name of the drive to which you plan to write the memory dump.

NOTE: If you want your system to support writing a memory dump to diskettes, refer to Chapter 10 for instructions on how to prepare a special set of diskettes to hold this dump. If your computer panics or hangs before you prepare these diskettes, you will be unable to take a memory dump at that point.

Once this dialog is concluded, SYSGEN writes an RLDR command line to the file CLI.CM. The utility then instructs DG/RDOS to execute CLI.CM so that the Relocatable Loader builds your tailored system. When the R prompt reappears, you can boot your newly created system (BOOT system-name<NL>); however, you may need to run the CONFIG utility (Chapter 8) and apply any optional patches before you do so.

SYSGEN's Error Messages

Below is a list of error messages you may receive from SYSGEN. Each message is accompanied by an explanation of what problem has occurred and of how you should respond.

`Error: Invalid answer`

If you use the argument/local-switch combination *system-name.SG/A* on your command line, SYSGEN does not run interactively, but builds a system according to the specifications recorded in the dialog file *system-name.SG*. If this file contains an invalid line, like "Number of LAC controllers (0-4) ? 5," SYSGEN does not know how to answer the question, so it prints this message and aborts.

`Error: Invalid dialog file`

If you use the argument/local-switch combination *system-name.SG/A* on your command line and the file *system-name.SG* was not created during an earlier SYSGEN session, if the file was created by an earlier revision of SYSGEN and the questions have changed, or if the file has been corrupted, you will receive the message shown above.

`Error: Question not found in dialog file`

You may have used the argument/local-switch combination *system-name.SG/A* on your command line and are trying to create a system with overlays. If you created *system-name.SG/A* while you were building an operating system without overlays—that is, you used the /R switch—you will see this message because the dialog file will not contain the question `A DRn RAM disk controller (Yes or No) ?` You will also see this message if the file *system-name.SG* was created by an earlier revision of SYSGEN and the questions have changed.

`Error: Unknown or invalid argument/switch combination`

This message usually means one of two things. Either you made a typing mistake and used a nonexistent switch (all valid switches are listed in Tables 7-1 and 7-2), or you used the local /A and /E switches on the same command line.

```
Warning: Invalid answer
```

If you use the argument/local-switch combination *system-name.SG/E* on your command line, SYSGEN uses the answers you provided during an earlier SYSGEN session as the default answers for the current session. If, however, the old dialog file *system-name.SG* contains a line such as "Number of LAC controllers (0-4) ? 5," the utility returns the message shown above and then chooses a default answer on its own. There is no need for SYSGEN to abort since it will later require you to answer the question.

-End of Chapter-

Chapter 8

Running the CONFIG Utility

Follow the instructions in this chapter after you have used SYSGEN to create a DG/RDOS operating system that supports the devices in your configuration. CONFIG allows you to finish tailoring your operating system by giving you an opportunity to

- Declare how much memory your computer has and how much of that memory you want to set aside for a RAM disk and for DG/RDOS use.
- Indicate where your $ devices are attached and what their characteristics are.
- Set the baud rate and a variety of other characteristics for your QTY lines.

The information on running this utility is presented under the following headings:

- Starting CONFIG
- CONFIG: Page One
- CONFIG: $ Device Pages
- CONFIG: QTY Line Pages
- CONFIG's Error Messages

Starting CONFIG

You can use the command CONFIG/H<NL> to display help information about the utility; however, to actually begin setting up your system's characteristics, you should issue a command in the format

CONFIG [system-name.SV] [dialog-file.CF/V]

where

system-name is the name of the operating system save file that you want to configure. If you do not include this name on your command line, CONFIG prompts you for it as the program comes up:

```
Name of system to configure ?
```

dialog-file/V is an argument/local-switch combination in which the argument is the name of a file to which you want CONFIG to write the specifications you enter as you go through the program's questions.

As an example, you could start CONFIG by typing

```
CONFIG NEWSYSTEM JUL2288.CF/V<NL>
```

This command line would start the program and cause it to display the first screen full of system parameters:

```
DG/RDOS System Configuration Program                    Rev xx.xx
System name  NEWSYSTEM    fk1 for choices, shft-fk1 or ? for help
------- use arrows (up, down, left, right) to move cursor -------

Total memory (64 - 1024 pages) ?          1024
.
.
.
.
.
.
.
Number of directories accessible at one time (1 - 64) ?  10

-------------------------------------------------------------------
   (1) Go to next page           (4) Size the system
   (2) Edit parameters           (5) Exit with/without change
   (3) Go to previous page       --> Enter your choice (1 - 5) _
```

When CONFIG presents this first page of questions, your cursor is positioned in a small menu area at the bottom of the screen next to the prompt `Enter your choice`. From this point, you can do several things:

- You can type 1<NL> to move to the next page of system parameters. (There may be anywhere from two to several dozen pages in a CONFIG session depending on how many $ devices and QTY lines you request.)
- You can type 2<NL> to begin answering the questions on the current page. Your cursor will be positioned at the beginning of the field following the first question. (If you type 2 followed by an uparrow, your cursor will be placed at the beginning of the field after the last question on the page.)
- You can type 3<NL> to move to a previous page of questions.

- You can type 4<NL> at any point during a CONFIG session to find out the size, in words, of your system as it is currently configured.
- You can type 5<NL> to leave the program.

Generally, the first step in running CONFIG is to choose option 2 to position your cursor at the beginning of the field after the page's first question. If you are unsure how to answer the question, you can consult the section below that discusses this question or type SHIFT-F1 or ? to display on-line help information. Then, if you decide to change this parameter, you can usually just type a valid answer over the default answer. However, since you may want to change only a single character or may make a typing mistake, it is a good idea to know about the functions of the editing keys listed in Table 8-1.

Table 8-1 CONFIG's Editing Keys

Key	Action
Home	Positions the cursor at the beginning of a field
Ctrl-A	Positions the cursor at the end of a field
Erase EOL	Erases from the current cursor position to the end of the field
rightarrow	Moves the cursor one character to the right
leftarrow	Moves the cursor one character to the left
Del	Replaces the character preceding the cursor with a space

Once you have typed your response to one of CONFIG's queries, you register that answer and move your cursor to another part of the screen by using one of several terminators. These are listed in Table 8-2.

Table 8-2 CONFIG's Line-Termination Characters

Key	Action
<NL> or downarrow	Terminates your response and moves the cursor to the next field on the screen.
uparrow	Terminates your response and moves the cursor to the previous field on the screen.
Shift-downarrow	Terminates your response and moves the cursor to the next field on the screen. Additionally, for any QTY Line Page question other than "QTY line number?", this sequence makes your response effective on all QTY lines.
Shift-uparrow	Terminates your response and moves the cursor to the previous field on the screen. Additionally, for any QTY Line Page question other than "QTY line number?", this sequence makes your response effective on all QTY lines.
F1	Terminates your response and moves the cursor to the menu at the bottom of the screen.

These are the mechanics of using the CONFIG program. The next three sections look at the questions CONFIG asks you and offer suggestions about how you should answer them.

CONFIG: Page One

The following list presents the questions that appear on page 1 of CONFIG and also makes some suggestions about how you should answer these questions.

1. `Total memory ?`

   Your answer to this question should indicate how much memory your computer has. If your response consists of a number followed by K, KB, M, or MB, it is converted and redisplayed in terms of 2-Kbyte pages. The suffixes K, KB, M, or MB can be separated by spaces from their preceding numbers.

   For example, if you are working on a DESKTOP GENERATION system with 256 Kbytes of memory, your answer could be 128, 256 KB, or 256K. If you are building a system for an MV/1400 DC computer with 4 Mbytes of memory, you could use an answer such as 2048, 4096K, or 4M.

2. RAM disk memory ?

   If you did not request a RAM disk during your dialog with SYSGEN, CONFIG displays this question, but does not allow you to answer it. If you did ask for a RAM disk, enter the number of 2-Kbyte pages of system memory you would like to set aside for this virtual disk. (For further information about creating and using a RAM disk, see Chapter 17.)

   If you request a large amount of memory, but a relatively small RAM disk, you may receive an overflow message from CONFIG. Refer to the section "CONFIG's Error Messages" later in this chapter.

3. BG/FG shared memory ?

   This feature allows you to set aside a certain number of memory pages as a shared data area, fully accessible to both the background and foreground. Unless your software requires shared pages, answer 0.

4. Number of I/O channels: BG ? ___ FG ? ___

   A channel is a gateway to an input or output device or a file. Therefore, to run a program in the background, you must enter in the first field of this line the maximum number of files that your background program can have open at any one time. Or, if your program addresses channels explicitly, by number, you should enter the highest decimal number your program uses as a channel number. (The minimum possible answer is 16 because the CLI requires 16 channels.) Do not enter an unnecessarily large number, however, because each channel CONFIG reserves consumes about 38 words of memory whether the program actually uses the channel or not. If you plan to run a program in the foreground, you should fill in the second blank using the same criteria you used in filling in the first. If you do not plan to run a program in the foreground, type 0 in the second field.

5. Number of stacks ?

   DG/RDOS uses stacks for system calls, disk I/O, and printer operations. You need one stack for your background program, one for a foreground program, and one for your printer. If your system has a printer and more than one terminal, you should request four or five stacks. However, keep in mind that for every stack over three, CONFIG reserves two extra system buffers.

6. Number of buffers ?

   The minimum number of buffers you can request is either four or twice the number of stacks you asked for, whichever is greater. You may want to allocate some extra buffers to reduce the time DG/RDOS spends accessing the disk. The extra buffers allow disk data and overlay files, ordinarily swapped out to disk, to remain memory resident. There are drawbacks to requesting too many buffers. First, each extra buffer uses 270 words of memory, so requesting a lot of buffers reduces the amount of memory available for user programs and may even prevent the operating system from fitting in its 64 Kbytes of address space. Second, if you create a large number of buffers and then need to find things that are not in the buffers, you will spend a lot of time searching the buffers and still have to access a disk anyway.

7. Number of cells ?

   A cell is a 16-word data buffer that the operating system can use for a variety of purposes. For instance, when your program does a system call, DG/RDOS stores certain status information, such as the values of your accumulators, in a cell. Also, the system stores input from QTY lines in cells. A good rule of thumb in deciding how many cells to allocate is to select three times the number of stacks you requested if your system does not support QTY lines and to request more cells if the system does.

8. Number of directories accessible at one time ?

   DG/RDOS allows you to divide each physical disk, or primary partition, into sections called secondary partitions. And each primary or secondary partition can contain one or more subdirectories. Your answer to this question determines how many of these directories can be initialized at any one time; it does not limit the number of directories you can create.

After answering the last question on page 1, you should press New Line, downarrow, or F1 to return the cursor to the menu at the bottom of the screen. Then you can choose option 1 to display the second page of system parameters.

CONFIG: $ Device Pages

Below are the questions that appear for each system console (the background console and then, if you asked the SYSGEN utility for one, the foreground console) along with some recommendations about how you should respond to them. Note that the CONFIG utility will display a separate screen for each console.

1. `Output device name ?`

   Enter a new, unique name if you want to rename this output device. Normally, you shouldn't rename the BG console.

   You should only rename a FG console if you'll use it as a printer.

2. `Input device name ?`

   Enter a new, unique name if you want to rename this input device. Normally, you shouldn't rename the BG console.

   Again, you should only rename a FG console if you'll use it as a printer.

3. `Link to QTY line ?`

   If you did not request support for QTY lines during your dialog with SYSGEN, CONFIG displays this question, but does not allow you to answer it or question 4. The next question you can respond to is question 5.

   If you are using the lines on an MV/1000 DC, MV/1400 DC, MV/2000 DC, MV/2500 DC, MV/3500 DC, or MV/5000 DC Series system board as QTY lines, you should answer `Y`. Otherwise, answer `N` for the BG console.

   For a FG console, answer `N` if you have FG console hardware and `Y` if you do not.

   If you answered `Y` to this question, go to question 4; if you answered `N`, go to question 5.

4. `QTY line number ?`

   Enter the name of the QTY line to which you want to redirect this device. Make sure that you don't link two devices to the same QTY line. After a device is linked to a QTY line, you can't use the line as filename QTY:n. However, you can still define the baud rate and other characteristics for the line on the appropriate QTY Line page.

5. `Spooling enabled ?`

   Choose whether to have spooling enabled or disabled for this device when the system comes up. However you respond, spooling can be enabled and disabled while the system is running.

6. `FF on open ?`

   Choose whether to have the system send a Form Feed to the device each time it is opened. If you answer `Y`, this characteristic will be enabled by default, but you can disable it via ac1 on open. If you answer `N`, this characteristic can never be enabled.

7. The following group of questions are for output characteristics that affect the .WRL system call. If you answer `Y` to a question, that characteristic will be enabled by default, but can be disabled via ac1 on open. If you answer `N` to a question, that characteristic can never be enabled.

   `Lower -> uppercase ?`

   Choose whether to have lowercase characters converted to uppercase characters before they are output.

   `Limit to 80 columns ?`

   Choose whether to have the system truncate output lines longer than 80 characters. Answer `N` to this question unless the device is a printer; in that case, base your response on the width of your paper.

   `Tab  -> spaces ?`

   Choose whether to have the system convert each tab character to a number of spaces (from 1 to 8) during output. Answer `Y` unlesss you're sure your device can handle tab characters.

   `CR -> CR+NL ?`

   Choose whether to have the system append a New Line after each Carriage Return that is output.

   `FF -> CR+FF+nulls ?`

   Choose whether to have the system append several nulls after each Form Feed that is output. Answer Y if you have an older printer that can't handle more characters while it is moving to the next page.

8. `8-bit ?`

   If your system console can handle 8-bit characters and you want support for these characters, or if this device is linked to a QTY line, respond to this question by typing `Y`. Also, refer to your terminal manual to see if there are switches on the back of your terminal you must set, or menus you need to use, to allow 8-bit operation.

   Otherwise, choose `N` to have the system strip the 8th bit (the parity bit when using 7-bit characters) of each input character.

   7-bit character support means that U.S. terminals can display the entire U.S. ASCII character set and non-U.S. terminals can display their special native characters. 8-bit character support allows terminals to display an additional 128 characters, the international character set.

9. The following group of questions are for input characteristics that affect the .RDL system call. If you answer `Y` to a question, that characteristic will be enabled by default, but can be disabled via ac1 on open. If you answer `N` to a question, that characteristic can never be enabled.

   Lower -> uppercase ?

   Choose whether to have lowercase input characters converted to uppercase characters before they are echoed or passed to a .RDL system call.

   `Echo, line-edit, etc ?`

   Choose whether to enable the echoing of input characters. This characteristic also enables special interpretations of Del, \, and Ctrl-Z.

   Normally, you should choose `Y`. If you answer `N`, these and other features (CR <-> NL, and lower -> uppercase) will be disabled regardless of your other choices.

Hardcopy ?

If you choose N, the Delete character is echoed as backspace-space-backspace, the \ character is echoed as multiple backspaces and spaces, and the New Line character becomes a Carriage Return character on input. If you answer Y, the Delete character is echoed as _, and the \ character is echoed as \-Carriage Return-New Line.

CR <-> NL ?

Choose Y to change Carriage Return characters to New Line characters and to change New Line characters to Carriage Return characters on input. Answer N if you do not want to swap the characters.

Next, if you asked the SYSGEN utility for support for parallel printer controllers, you will be asked questions for each printer (up to two printers); note that the CONFIG utility will display a separate screen for each printer. These questions are described below:

1. Output device name ?

   Enter a new, unique name if you want to rename this output device.

2. Spooling enabled ?

   Choose whether to have spooling enabled or disabled for this device when the system comes up. However you respond, spooling can be enabled and disabled while the system is running.

3. FF on open ?

   Choose whether to have the system send a Form Feed to the device each time it is opened. If you answer Y, this characteristic will be enabled by default, but you can disable it via ac1 on open. If you answer N, this characteristic can never be enabled.

4. The following group of questions are for output characteristics that affect the .WRL system call. If you answer Y to a question, that characteristic will be enabled by default, but can be disabled via ac1 on open. If you answer N to a question, that characteristic can never be enabled.

   Lower -> uppercase ?

   Choose whether to have lowercase characters converted to uppercase characters before they are output.

```
Limit to 80 columns ?
```

Choose whether to have the system truncate output lines longer than 80 characters. Base your response on the width of your paper.

```
Tab  -> spaces ?
```

Choose whether to have the system convert each tab character to a number of spaces (from 1 to 8) during output. Answer Y unlesss you're sure your device can handle tab characters.

```
CR -> CR+NL ?
```

Choose whether to have the system append a New Line after each Carriage Return that is output.

```
FF -> CR+FF+nulls ?
```

Choose whether to have the system append several nulls after each Form Feed that is output. Answer Y if you have an older printer that can't handle more characters while it is moving to the next page.

Finally, if you requested support for QTY lines and requested any extra linked $ devices from the SYSGEN utility, the CONFIG utility asks the questions below for each extra linked $ device (up to four devices); note that the CONFIG utility will display a separate screen for each extra linked $ device:

1. `Output device name ?`

   Enter a new, unique name if you want to rename this output device.

2. `Input device name ?`

   Enter a new, unique name if you want to rename this input device.

4. `QTY line number ?`

   Enter the name of the QTY line to which you want to redirect this device. Make sure that you don't link two devices to the same QTY line. After a device is linked to a QTY line, you can't use the line as filename QTY:n. However, you can still define the baud rate and other characteristics for the line on the appropriate QTY Line page.

5. `Spooling enabled ?`

   Choose whether to have spooling enabled or disabled for this device when the system comes up. However you respond, spooling can be enabled and disabled while the system is running.

6. `FF on open ?`

   Choose whether to have the system send a Form Feed to the device each time it is opened. If you answer `Y`, this characteristic will be enabled by default, but you can disable it via `acl on open`. If you answer `N`, this characteristic can never be enabled.

7. The following group of questions are for output characteristics that affect the .WRL system call. If you answer `Y` to a question, that characteristic will be enabled by default, but can be disabled via `acl on open`. If you answer `N` to a question, that characteristic can never be enabled.

   `Lower -> uppercase ?`

   Choose whether to have lowercase characters converted to uppercase characters before they are output.

   `Limit to 80 columns ?`

   Choose whether to have the system truncate output lines longer than 80 characters. Answer `N` to this question unless the device is a printer; in that case, base your response on the width of your paper.

   `Tab  -> spaces ?`

   Choose whether to have the system convert each tab character to a number of spaces (from 1 to 8) during output. Answer `Y` unlesss you're sure your device can handle tab characters.

   `CR -> CR+NL ?`

   Choose whether to have the system append a New Line after each Carriage Return that is output.

   `FF -> CR+FF+nulls ?`

   Choose whether to have the system append several nulls after each Form Feed that is output. Answer Y if you have an older printer that can't handle more characters while it is moving to the next page.

8. The following group of questions are for input characteristics that affect the .RDL system call. If you answer Y to a question, that characteristic will be enabled by default, but can be disabled via ac1 on open. If you answer N to a question, that characteristic can never be enabled.

   Lower -> uppercase ?

   Choose whether to have lowercase input characters converted to uppercase characters before they are echoed or passed to a .RDL system call.

   Echo, line-edit, etc ?

   Choose whether to enable the echoing of input characters. This characteristic also enables special interpretations of Del, \, and Ctrl-Z.

   Normally, you should choose Y. If you answer N, these and other features (CR <-> NL, and lower -> uppercase) will be disabled regardless of your other choices.

   Hardcopy ?

   If you choose N, the Delete character is echoed as backspace-space-backspace, the \ character is echoed as multiple backspaces and spaces, and the New Line character becomes a Carriage Return character on input. If you answer Y, the Delete character is echoed as _, and the \ character is echoed as \-Carriage Return-New Line.

   CR <-> NL ?

   Choose Y to change Carriage Return characters to New Line characters and to change New Line characters to Carriage Return characters on input. Answer N if you do not want to swap the characters.

After you respond to the last question on the last $ Device page, your cursor returns to the menu at the bottom of the page. At this point, if you configured a system without QTY lines, you should choose option 5 to exit from the program. Otherwise, you have the option of answering another page full of questions for each QTY line on which you do not want the default characteristics. To see what those characteristics are or to begin changing the characteristics on one of your QTY lines, choose option 1 to go to the next page.

CONFIG: QTY Line Pages

Below are the questions you may answer for each QTY line in your system along with some brief recommendations about how you should respond.

For all questions but the first one, you can change a characteristic for all lines by using the combination Shift-downarrow or Shift-uparrow, as described in Table 8-2.

1. `QTY line number ?`

   There are two ways to answer this question. First, if you want to change the characteristics on all of your QTY lines, you can simply accept the default value on each page. That is, on the first page of QTY-line questions, the utility supplies a default answer of 0. On the next page, it supplies a default answer of 1, and so on. Alternatively, you can type in the number of the QTY line whose characteristics you want to alter; for instance, to work on QTY line 8, type `8<NL>`, and you will automatically be moved to the appropriate page.

2. `Baud rate code ?`

   Your answer to this question depends on the device attached to the line. For example, if there is a 1200-baud modem on a given line, answer 7 for a line speed of 1200 baud. Or if a line is connected to a terminal whose baud-rate selector is set to 9600, answer 14.

   NOTE: The only asynchronous controller that supports all 16 of the baud rates is the USAM board on DESKTOP GENERATION systems. On other systems, if you choose certain baud rates, the operating system usually maps your choice to another baud rate, often one that is close or one that is supported by the hardware but not by CONFIG.

   DG/500, LMC-8_1, LAC-12, MCP-1, and IAC-24 controllers, as well as the lines on MV/1000 DC, MV/1400 DC, MV/2000 DC, and MV/2500 DC system boards, all support the same set of baud rates. These baud rates include all but 3 of the choices offered by CONFIG; the exceptions are 50, 3600, and 7200 baud, mapped to 75, 4800, and 9600 baud, respectively.

   On LMC-8_2, LAC-16, LAC-32, and IAC-8_3 controllers and the lines on MV/3500 DC and MV/5000 DC Series system boards, DG/RDOS maps 50 to 75, 3600 to 4800, 7200 to 9600, and 135 to 38400 baud.

   On ULM controllers, 2000 baud becomes 200 baud, 7200 baud becomes 2400 baud, and 3600 baud is undefined.

ALM controllers offer four "clock" selections. You must set jumpers on these boards to define the four baud rates that will correspond to these four clocks. DG/RDOS maps 3600, 7200, and 9600 baud to clock 0; 600 and 19200 baud to clock 1; 50, 1200, 1800, 2000, and 4800 baud to clock 2; and 75, 110, 135, 150, 300, and 2400 baud to clock 3.

3. `Number of data bits ?`

   Enter the number of bits per character, not including a parity bit, you will transfer over this line. On a multiplexor line, a terminal or printer will normally work with 7 data bits per character and even parity, while communications lines almost always require 8 data bits per character and no parity.

   If you have an MV/1000 DC, MV/1400 DC, MV/2000 DC, MV/2500 DC, MV/3500 DC, or MV/5000 DC Series system, you should request 8 bits and no parity for the master console.

4. `Parity type code ?`
   `0=none 1=odd 2=even`

   In general, if you will be sending or receiving 7-bit characters over this line, answer 2 for even parity. If the line will carry 8-bit characters, select 0 for no parity.

5. `Number of stop bits ?`

   On any line, the standard device uses 1 stop bit and 1 start bit. Therefore, you should usually leave the default answer to this question unchanged. However, some devices with very low baud rates, for example 110 baud, require two stop bits.

6. `Loopback mode ?`

   Enter `Y` only if you want any output to this line to be reflected into this same line. Loopback mode can be useful for testing.

7. `Honor hardware busy ?`

   You need to answer this question only if you are setting up characteristics for one or more of the following lines:

   1. DG/500 systems: The first, second, fifth, and sixth lines on each controller.
   2. MV/1000 DC, MV/1400 DC, MV/2000 DC, and MV/2500 DC systems:
      a. If you responded YES to SYSGEN's question about using DUART lines as QTY lines:
         - Lines QTY:0, QTY:1, QTY:4, and QTY:5
      b. If you responded with an answer greater than 0 to SYSGEN's question about LAC controllers:
         - The last two lines on each LAC-12
         - All lines on each LMC-8_1 or LMC-8_2
         - The first four lines on each LAC-16
         - The first eight lines on each LAC-32
   3. MV/3500 DC and MV/5000 DC Series systems:
      - The first four and last four lines on the system board
      - All lines on each LMC-8_2
      - The first four lines on each LAC-16
      - The first eight lines on each LAC-32
   4. MV/7800 XP, MV/9500, MV/15000, and MV/18000 systems:
      - All lines on each MCP-1
      - All lines on each IAC-8_3

   If you are setting up the characteristics for one of these lines and the device on the line is a terminal or a printer employing XON/XOFF, you should press New Line to accept the default answer, no. If there is a modem on the line or any device that uses hardware busy (Clear To Send), type `Y<NL>`.

8. `Modem line ?`

   If this line will be connected to a modem or another device that uses the modem status signals DTR, RTS, and DSR, make sure this value is `Y`; otherwise, the value should be `N`.

   Also, for each USAM line connected to a modem, you must set a switch on the USAM card to tell it to notify the computer when it receives a Ring Indicator signal. The instructions for setting this switch are in the installation manual that came with your hardware.

9. `Data Terminal Ready ?`
   `Request To Send ?`
   `Data Set Ready ?`

   On lines that are not modem lines, you can leave these values as they are. On a modem line, follow these rules. For both Data Terminal Ready and Request To Send, accept the default answer if you want the operating system to assert the signal whenever you boot. If you asked for the standard modem timer when you ran SYSGEN, base your response to the last question on which signal you want your operating system to monitor, Carrier Detect or Data Set Ready.

Answer this series of questions for each QTY line in your system on which you want other than the default characteristics. You can move from page to page by answering the `QTY line number?` question or by using the menu at the bottom of the screen.

Leaving CONFIG

When you have set up all of your QTY lines, you can choose option 5 from CONFIG's menu to exit the program. As long as the system you have configured is not too large to fit in the logical address space available, the utility will respond by printing, near the bottom of the screen, the size of your new system:

```
------------ Size (words) is 28630 ------------
```

The program will also ask you to confirm that you want to make the changes that you requested earlier:

```
Do you want to install changes in the system (Y or N) ? Y
```

If you change the default value to N and press New Line, CONFIG will make no changes to your system and will not write the dialog file you requested on your command line. So to create the system you specified while running CONFIG and the accompanying log file, press New Line to accept the default answer, yes. After a few moments, the CLI prompt will reappear, indicating that CONFIG has successfully modified your system.

To run the newly configured system, you must shut down the system you are running currently and bring up the new one. You can do so very simply by typing

```
BOOT system-name<NL>
```

On DESKTOP GENERATION systems, you will have to reenter the date and time before you reach the CLI.

CONFIG's Error Messages

Below is a list of error messages you may receive from CONFIG. Each message is accompanied by an explanation of what problem has occurred and of how you should respond.

Incompatible revision level

The revision levels of your CONFIG utility and your DG/RDOS operating system do not match. For example, you might be trying to run Rev. 2.60 of CONFIG on a Rev. 2.50 operating system.

NMAX out of range

The system you built with SYSGEN is too large to execute (NMAX is greater than 76000 octal).

Not a DG/RDOS system file

You asked CONFIG to process a file other than a DG/RDOS operating system .SV file.

`Overflow (words) is nnnnn`

If you configure a system that is too large for your computer's logical address space, this message tells you by how many words you need to reduce the size of the system. Generally, you cut back the size of the system by returning to page 1 of CONFIG and requesting fewer channels, stacks, buffers, and cells than you asked for earlier.

You may receive this message if you request a large amount of memory and a relatively small RAM disk. In this case, the table that lists what pages are being used by each ground is too big to fit within the operating system. Decreasing the amount of total memory or increasing the size of your RAM disk will solve the problem.

`Please enter Y, y, N, or n`

If you make a typing mistake and respond to a yes-or-no question by typing a character other than `Y` or `N`, CONFIG issues this reminder.

`This is the first page`

You issued the command "Go to previous page" while you were on page 1 of CONFIG.

`This is the last page`

You issued the command "Go to next page" while you were on CONFIG's last page.

`Unable to access file`

This message usually indicates that the DG/RDOS system file you are trying to configure is read protected.

`Unable to open file`

You receive this message if you ask CONFIG to open a file that does not exist.

`Value is not in range`

For each question that you must answer with a value, CONFIG presents within parentheses the range of values you may select from. If you choose a value outside this range, the program prints this message.

`Value is not numeric`

You have made a typing mistake and entered a character other than a number in a field where only numbers are acceptable.

`Value is not valid`

For some questions, you can specify a number followed by K, KB, M, or MB (with or without separating spaces between the number and the suffix). If you specify a value that is not valid, the program prints this message. Examples of invalid values are 257 KB (with the KB suffix, the number must be even), 2NB (should be "MB"), and MB 2 (the number must be first).

-End of Chapter-

Chapter 9

Applying Patches

A patch is a modification to a save or overlay file that alters the way a program works. If one or more such patches are available for your revision of DG/RDOS, they will be described in your Release Notice.

If your Release Notice contains information about patches, read the descriptions of what the patches do. If you find a patch you want or need to use, make careful note of the location to be patched, the current contents of that word, and the value that you want to write at that location. You generally then use the ENPAT utility to create the patch and the PATCH utility to apply it. Both of these utilities are documented in the manual *RDOS/DOS Debugging Utilities*.

If you patch an operating system file or a stand-alone utility such as DKINIT, the change you make takes effect the next time you boot that operating system or utility. If you patch a stand-among utility, such as the CLI or BURST, the change takes effect the next time you execute that program.

If you received a DG/RDOS update at the same time that you received your operating system and this update consists of patches, consult the section "Installing a DG/RDOS Update" in Chapter 18 for instructions on how to handle the update.

-End of Chapter-

Chapter 10

Starting Up and Shutting Down Your System

Read this chapter when you want to

- Restart your system after a normal shutdown
- Shut down your system
- Recover from a panic or a hang, or a power failure

This chapter begins by detailing the steps involved in starting up and shutting down a DG/RDOS system. Then it provides instructions on how to handle an abnormal shutdown and how to produce a memory dump. These subjects are covered in the following sections:

- Starting Up Your System
- Shutting Down Your System
- Handling an Abnormal Shutdown

Starting Up Your System

Once you have generated your DG/RDOS system (Chapters 2-9), restarting that system is easy. Simply refer to the subsection "DESKTOP GENERATION Systems," "DG/500 Systems," "MV/1000 DC, MV/1400 DC, MV/2000 DC, and MV/2500 DC Systems," "MV/3500 DC and MV/5000 DC Series Systems", or "MV/7800 XP, MV/9500, MV/15000, and MV/18000 Systems," depending on the hardware you are using. Each of these subsections takes you to the point where you have the CLI running.

DESKTOP GENERATION Systems

The following instructions are based on the assumption that your system is shut down completely when you begin. Therefore, if you did not cut off the power to your peripherals and computer the last time you shut down the operating system, you can skip the initial steps listed below. If your system console is displaying the virtual console prompt, begin with step 4; if it is displaying the `Filename?` prompt, start with step 5.

1. Turn on power to your system console using the switch on the back of your monitor.

2. Turn on each disk, tape, terminal, printer, and plotter in the system, and make sure that these devices are on-line. Also turn on any remote units.

3. Turn on power to the computer itself using the switch at the upper right corner of the power supply module. If the On Line status light at the top of a keyboard does not glow, press that terminal's Cmd and On Line keys simultaneously to put the terminal on line.

   NOTE: If you have a Model 10 or 10SP system with a printer or terminal on its CPU printer port, that device must be on-line. Also, if the device is a printer, it must have paper. When you power up a Model 10 or 10SP system with a device on the CPU printer port, the computer tests the device. If the device fails the test—which will happen if the device is off-line—the computer does not complete its power-up test.

   On a Model 10 or 10SP system, the computer runs a system power-up test and displays the following message:

```
TESTING...

     ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
     BLINK DIM REVERSE UNDERLINE ALL NORMAL

TEST PASSED.
```

   If your Model 10 or 10SP system has a fixed disk, you also see the message `SELECT LOAD DEVICE: 20H (FOR DISKETTE) OR 26H (FOR DISK)`. Then the virtual console prompt (`!`) appears.

   On a Model 20 or 30 system, the virtual console prompt appears immediately.

4. Type `26H` to give control to the bootstrap routine (DISKBOOT.SV) on your hard disk. When this routine begins executing, it asks for the name of the program file you want to load:

```
Filename?
```

   NOTE: If your system does not include a hard disk, insert your system diskette in drive DJ0 and type `20H`.

5. To start an operating system that is pointed to by a link file named SYS.SV or an operating system named SYS.SV, either press New Line or wait for approximately 30 seconds and it automatically starts. (Your DG/RDOS release diskettes contain a link file, SYS.SV, that points to the starter system.) To start another operating system (or a stand-alone program), type in the program name and then press New Line; if you wait too long to type the program name, SYS.SV starts automatically. As the system comes up, it announces itself with the message `DG/RDOS Rev xx.xx`.

6. You must now set the system date and time. When the system console displays the prompt `Date (m/d/y) ?`, enter the date (followed by New Line) in one of the following formats:

```
4 28 88
4,28,88
4-28-88
4:28:88
4/28/88
APR 28 88
APR,28,88
28-APR-88
```

   The system then prompts you for the time. After this prompt, enter the time (followed by New Line) in one of the following formats:

```
14 30 00
14,30
14-30
14:30
14/30
2 30 PM
2:30 P
```

7. At this point, on an Model 10 or 10SP system, you see a welcome message and an indication that an emulator file is being loaded. Then, on all models, the CLI's R prompt appears.

DG/500 Systems

If your system is shut down completely, consult the chapter on routine powerups in the hardware manual that explains how to start and maintain a DG/500 system. By the time you reach the end of that chapter, your console should be displaying the prompt `Filename?`

If you did not cut off the power to your peripherals and computer the last time you shut down the operating system and your system console is displaying the `Filename?` prompt, go to the paragraph below. If your system console is displaying the virtual console prompt, type 26H; the bootstrap root on your hard disk begins executing, and you will see the `Filename?` prompt.

Once you reach the `Filename?` prompt, you can start an operating system that is pointed to by a link file named SYS.SV (or an operating system named SYS.SV) by either pressing New Line or waiting approximately 30 seconds for it to start automatically. (Your DG/RDOS release media contains a link file, SYS.SV, that points to the starter system.)

To start another operating system (or a stand-alone program), type in the program name and then press New Line; if you wait too long to type the program name, SYS.SV starts automatically. As the system comes up, it announces itself with the message `DG/RDOS Rev xx.xx` and displays the date and time. Then the CLI's R prompt appears.

MV/1000 DC, MV/1400 DC, MV/2000 DC, and MV/2500 DC Systems

The first subsection below, "Starting a DG/RDOS System on DA0," explains in detail how to start DG/RDOS on MV/1000 DC and MV/1400 DC computers, and on any MV/2000 DC or MV/2500 DC computer that has a DG/RDOS system on its first fixed disk.

The second subsection, "Starting a DG/RDOS System on DA1," summarizes the steps you need to take to start a DG/RDOS system that resides on the second fixed disk of an MV/2000 DC or MV/2500 DC computer.

Starting a DG/RDOS System on DA0

The following instructions are based on the assumption that your system is shut down completely when you begin. Therefore, if you did not cut off the power to your peripherals and computer the last time you shut down the operating system, you can skip the initial steps listed below. If your system console is displaying the prompt `Filename?`, just perform step 4 below. If the console is displaying the SCP-CLI prompt, first type `BOOT 24<NL>`; then, when the `Filename?` prompt appears, perform step 4.

1. Turn on each terminal, printer, and plotter in the system, and make sure that these devices are on line. Also turn on any remote units such as an external tape drive or CSS/DC expansion chassis.

2. Make sure that your diskette and tape drives are empty so that when you supply power to your computer, it will not try to load microcode and power-up diagnostics from one of those devices.

3. Press the computer unit's power button to turn the unit on. The button should illuminate and stay in. Also, the On Line light on each terminal's keyboard should glow. If any terminal's On Line light is not glowing, hold down the Cmd key and press the On Line key to put the terminal on-line.

   After you supply power to the computer, it begins a series of power-up tests and displays diagnostic messages similar to these:

```
TESTING...

     Model # 8936; System Processing Unit (SPU) Level A
        ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789, PASSED

     -- Standard Hardware Tests Complete --

     -- Optional Hardware Tests Beginning --

     Model # 4560; Slot A; Async Communications Board
        ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789, PASSED

     -- Optional Hardware Tests Complete --

     -- Memory Size is 4 Megabytes --
```

Then the Automatic Program Load Menu appears.

```
                                        9-MAR-88 12:08:17

              Automatic Program Load Menu

   1 Continue immediately with preset values
   2 Change preset values

   Loading with preset values will continue automatically
   unless you respond within 45 seconds.

   The default device is hard disk

   For assistance, press the Help key (SHIFT-F1) or H


   Enter choice [1]:
```

Simply wait for 45 seconds (or press New Line), and the system transfers control to the bootstrap routine (DISKBOOT.SV) on your hard disk. When this routine begins executing, it prompts you for the name of the program you want to load:

```
Filename?
```

4. To start an operating system that is pointed to by a link file named SYS.SV or an operating system named SYS.SV, either press New Line or wait for approximately 30 seconds and it automatically starts. (Your DG/RDOS release tape or diskettes contain a link file, SYS.SV, that points to the starter system.) To start another operating system (or a stand-alone program), type in the program name and then press New Line; if you wait too long to type the program name, SYS.SV starts automatically. As the system comes up, it announces itself with the message `DG/RDOS Rev xx.xx` and displays the date and time. Then the CLI's R prompt appears.

Starting a DG/RDOS System on DA1

Since an MV/2000 DC or MV/2500 DC computer's power-up menus do not provide a way for you to boot an operating system that resides on a second fixed disk, you must follow the steps enumerated below to start a DG/RDOS system that was installed on DA1. These instructions assume that your system is shut down completely when you begin.

1. Turn on the power to all of your peripherals; make sure that your diskette and tape drives are empty; and then turn on the power to your computer.
2. When you see the Automatic Program Load Menu, type `2<NL>` to reach the Change Preset Values Menu.
3. Type `6<NL>` to enter the SCP CLI.
4. Either insert your first DG/RDOS release diskette in your diskette drive or insert your DG/RDOS release tape in your tape drive. Then type `BOOT` *nn*`<NL>` where *nn* is the device code of the diskette or tape unit on which your release media is mounted (see Appendix A).

   If you just booted your first DG/RDOS release diskette, you will soon see the prompt `Filename?` at which point you can proceed to step 5.

   If you booted a DG/RDOS release tape, you will first see the prompt `Number of tape file to boot? File =.` In response, type `5<NL>` to start the disk bootstrap program DISKBOOT.SV. As this program begins executing, it displays the following message and prompt:

   ```
   DISKBOOT Rev xx.xx

   Bootstrap Device Specifier ?
   ```

   Type `DA1<NL>`, and when DISKBOOT asks whether you want to install a bootstrap root on DA1, answer `N` (pressing New Line is not necessary). At this point you will see the prompt `Filename?`

5. Once you reach the `Filename?` prompt, starting a DG/RDOS operating system that resides on DA1 is very simple. Just enter a response in the form `DA1:system-name<NL>`; if you booted a DG/RDOS release diskette, type your response within approximately 30 seconds. As the system comes up, it announces itself with the message `DG/RDOS Rev xx.xx` and displays the date and time. Then the CLI's R prompt appears.

MV/3500 DC and MV/5000 DC Series Systems

If your system is shut down completely, consult the chapter on routine powerups in the hardware manual for MV/3500 DC and MV/5000 DC Series computers. By the time you reach the end of that chapter, you will have done the following:

1. Turned on the power.
2. Checked the standard power-up test display.
3. Reached the Automatic Program Load (APL) menu.

When the APL menu appears, simply wait for 45 seconds (or press New Line if you don't want to wait), and the system transfers control to the bootstrap routine (DISKBOOT.SV) on your hard disk. Each time you power up your system, this routine finds SYS.MF, resolves any links, and tries to load your microcode from the disk file. While the microcode is being loaded, you may receive messages like the following:

```
Initiating microcode load...
```

When the microcode has been loaded, you receive this message:

```
Microcode load complete.
```

The bootstrap routine then prompts you for the name of the program you want to load:

```
Filename?
```

If you did not cut off the power to your peripherals and computer the last time you shut down the operating system and your system console is displaying the `Filename?` prompt, go to the paragraph below. If your system console is displaying the System Control Program prompt, type `BOOT` *dc n*`<NL>` where *dc* is your hard disk's device code and *n* is its unit number. The bootstrap root on your disk begins executing, and you will see the `Filename?` prompt.

Once you reach the `Filename?` prompt, you can start an operating system that is pointed to by a link file named SYS.SV (or an operating system named SYS.SV) by either pressing New Line or waiting approximately 30 seconds for it to start automatically. (Your DG/RDOS release media contains a link file, SYS.SV, that points to the starter system.)

To start another operating system (or a stand-alone program), type the program name and then press New Line within approximately 30 seconds. As the system comes up, it announces itself with the message DG/RDOS Rev xx.xx and displays the date and time. Then the CLI's R prompt appears.

MV/7800 XP, MV/9500, MV/15000, and MV/18000 Systems

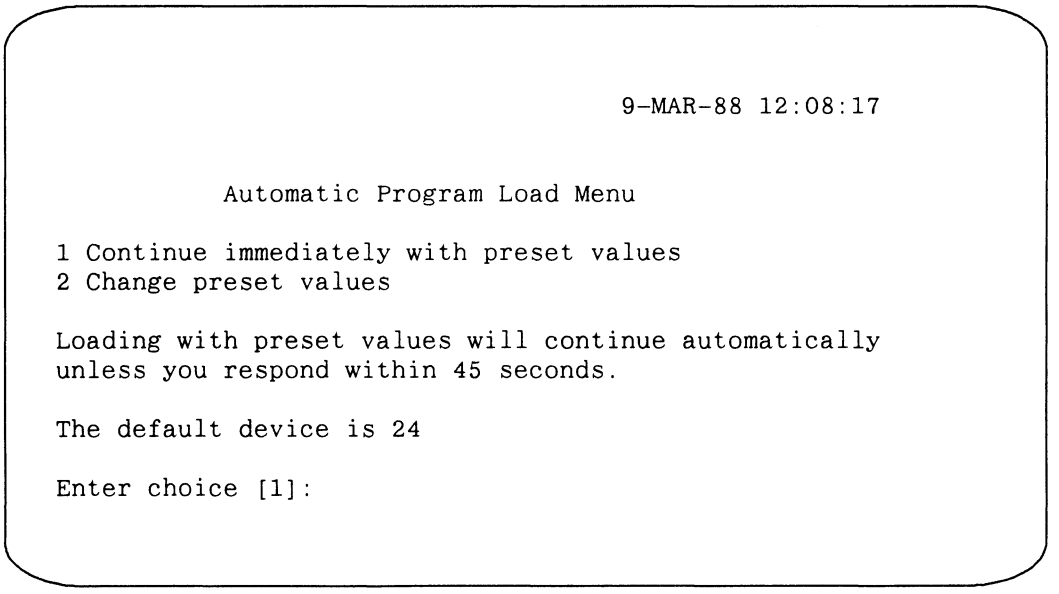
The following instructions are based on the assumption that your system is shut down completely when you begin. Therefore, if you did not cut off the power to your peripherals and computer the last time you shut down the operating system, you can skip the initial steps listed below. If your system console is displaying the prompt Filename?, just perform step 3 below. If the console is displaying the SCP-CLI prompt, first type BOOT *nn*<NL>, where *nn* is the device code of your system disk; then, when the Filename? prompt appears, perform step 3.

1. Turn on each terminal, printer, and plotter in the system, and make sure that these devices are on line. Also turn on power to your disk drive(s) and any tape drives.

2. Press the computer unit's power button to turn the unit on. The On Line light on each terminal's keyboard should glow. If any terminal's On Line light is not glowing, hold down the Cmd key and press the On Line key to put the terminal on-line.

   After you supply power to the computer, it begins a series of power-up tests. A series of messages keeps you apprised of the system's progress.

   Then the Automatic Program Load Menu appears.



Simply wait for 45 seconds (or press New Line), and the system transfers control to the bootstrap routine (DISKBOOT.SV) on your hard disk. Each time you power up your system, this routine finds SYS.MF, resolves any links, and tries to load your microcode from the disk file. While the microcode is being loaded, you may receive messages like the following:

```
Initiating microcode load...
```

When the microcode has been loaded, you receive this message:

```
Microcode load complete.
```

The bootstrap routine then prompts you for the name of the program you want to load:

```
Filename?
```

3. To start an operating system that is pointed to by a link file named SYS.SV or an operating system named SYS.SV, press New Line, or wait approximately 30 seconds for it to start automatically. (Your DG/RDOS release tape contains a link file, SYS.SV, that points to the starter system.) To start another operating system (or a stand-alone program), type the program name and then press New Line within approximately 30 seconds. As the system comes up, it announces itself with the message `DG/RDOS Rev xx.xx` and displays the date and time. Then the CLI's R prompt appears.

Shutting Down Your System

Shutting down your system involves, first of all, stopping DG/RDOS in an orderly fashion so that all your directories are released and your master allocation and system directories are updated. Second, if your system will not be used for a while, you may choose to turn off your peripherals and your computer.

CAUTION: Never turn off the power to your computer while DG/RDOS is running.

1. Terminate any non-CLI program running in the background and any program running in the foreground. As a last resort, you can terminate a foreground program by typing Ctrl-C Ctrl-F at the background console.
2. Use the DIR command to get into the master directory:

```
R
DIR %MDIR%<NL>
```

3. Use the BOOT command to shut down DG/RDOS (you can also use the macro BYE.MC, which is provided with your operating system):

```
R
BOOT %MDIR%<NL>

Master device released

Filename?
```

The message `Master device released` means that DG/RDOS has closed all directories and terminated normally.

NOTE: If you have used the MDIR command to reassign your master directory (for example, to a RAM disk) this procedure may not work. Use the name of your system disk instead of %MDIR% in the DIR and BOOT commands above.

4. At this point, DG/RDOS is shut down.

If you are working on a **DESKTOP GENERATION** or **DG/500** system, you can now use the break key sequence if you want to return to the virtual console:

```
Filename? Cmd-Brk
...
!
```

Type `I` after the ! prompt to reset your I/O devices.

If you are working on an MV/Family system, you can use the break key sequence to return to the SCP-CLI (you may have to type Cmd-Brk three times):

```
Filename? Cmd-Brk

BREAK
...
SCP-CLI>
```

Type `RESET<NL>` after the SCP-CLI prompt to reset your I/O devices.

If you will not be using your system for a long period of time, you may want to turn off the power.

Before cutting off the power, remove any diskettes or tapes from their drives. Then turn off the power to all of your peripherals that have power switches, except the master console. Turn off the power to your computer. And, finally, switch off your system console.

Handling an Abnormal Shutdown

This section tells you how to proceed if your system panics or hangs or if there is a power failure.

Restarting after a Panic or Hang

There are certain errors that DG/RDOS can recognize, but cannot fix. For example, the operating system may detect invalid data in a master allocation directory or a system stack fault. When the system encounters one of these errors, it stops what it is doing and executes a panic routine that displays on your console a series of five six-digit numbers. The first four of these are the values of the four accumulators, and the fifth is a panic code. For an explanation of what a given panic code means, consult the parameter file DGPARU.SR or the *RDOS System Reference*.

A hang also results from a serious error, but there is no panic message on the system console. Instead, DG/RDOS seems frozen. There is no response to commands you type at the system console.

If your system panics or hangs, you may want to produce a memory dump before you restart your system. And when you do start DG/RDOS again, you must reset your I/O devices and clear the use counts of files that were open when the system went down.

Producing a Memory Dump

If you selected the memory dump feature when you ran the SYSGEN utility, you can write all of memory to tape or diskettes after a panic or a hang. The instructions below tell you how to prepare the media that will receive the memory dump, how to start the memory dump routine, and how to respond to the routine's prompts.

1. If you plan to write the contents of memory to tape, there is no need to prepare special tapes; you simply need to make sure that you have extra tapes on hand. However, if you plan to dump the contents of memory to diskettes, you must prepare a set of diskettes to receive the dump before your system panics or hangs.

    a. First, make sure you have enough diskettes on hand to hold the memory dump. The memory dump routine begins dumping data to each diskette at block 25 (octal) and always dumps a multiple of 256 Kbytes per diskette. This means that if you are using 48 TPI diskettes, one diskette can hold 256 Kbytes of memory, and if you are using 96 TPI diskettes, one diskette can hold 512 Kbytes. Simply divide the amount of memory on your system by the appropriate diskette capacity to arrive at the number of diskettes you will need for a memory-dump operation.

b. Next, run DKINIT on each of these diskettes. (For information on how to run this utility, see Chapter 16.) If DKINIT reports that a diskette has a bad block, you cannot use that diskette for a memory dump. Also, be sure to accept the default answers to DKINIT's questions about the location of your remap area and your disk's frame size.

c. Use the command INIT/F to write master allocation and system directories to each diskette. For example, if you are working on an MV/2000 DC system and are preparing a diskette in drive DA10, you should type

```
R
INIT/F DA10<NL>
```

Then, when you are prompted to confirm that you want to initialize this diskette, type Y.

d. Finally, make each diskette you are preparing the current directory and then create on that diskette a contiguous file of 512 (48 TPI diskettes) or 1024 (96 TPI diskettes) blocks. On your first diskette, you might name this file FILE1; on your second diskette, name it FILE2, and so on:

```
R
DIR DA10<NL>
R
CCONT/N FILE1 1024<NL>
R
RELEASE DA10<NL>
R
```

After you create this file on each diskette, be sure to put a paper label on the diskette and to write on the label the name of the file you just created.

2. Getting your memory-dump diskettes ready is actually a preparatory step. Once a panic or hang occurs, the first thing you need to do is start the memory-dump routine. If your system panics, this routine begins executing automatically and prompts you to insert a piece of media into the device to which you plan to write the memory dump. For instance, if you are working on an MV/2000 DC system and, at SYSGEN time, selected UT0 as your memory dump device, the routine displays this message:

```
Ready UT0, then hit New Line to start the Memory Dump
```

On the other hand, if your system hangs, you must start the memory-dump routine yourself. The steps involved in this task are detailed below.

a. Since your system is frozen, first type the break key sequence (Cmd-Brk) to return to the virtual console (DESKTOP GENERATION and DG/500 systems) or the System Control Program (MV/Family systems). To reach the System Control Program, you may have to type this sequence three times.

b. Then, use a virtual-console or SCP command to reset your I/O devices. On a DESKTOP GENERATION or DG/500 system, type `I` after the ! prompt. On an MV/Family system, type `RESET<NL>` after the SCP-CLI prompt.

c. Next, determine the address at which the system's panic routine begins. If you have a printout of your system load map, you can simply look for the label PNIC and note the address at which the panic code begins. Alternatively, you can examine location 51 in memory to find the routine's starting address.

DESKTOP GENERATION or DG/500 system: Type `51/` at the virtual console prompt:

```
!51/000051 006412
```

The first number after the slash represents a memory location, and the second number is the value stored at that location. This value is the starting address for PNIC.

Press New Line to get back to the virtual console prompt.

MV/3500 DC or MV/5000 DC Series system: Type `FLAGS RADIX 8<NL>` at the System Console Program prompt to make sure that the current radix is octal:

```
SCP-CLI/Jp0> FLAGS RADIX 8<NL>
```

Then type `EXAMINE 51<NL>` after the prompt to determine the value stored at location 51:

```
SCP-CLI/Jp0> EXAMINE 51<NL>
00000000051 006412
```

The first number is a memory address and the second number is the value stored at that location. This value is the starting address for PNIC.

You are automatically returned to the SCP CLI prompt.

Any other MV/Family system: Type EXAMINE MEMORY 51<NL> after the SCP-CLI prompt:

```
SCP-CLI> EXAMINE MEMORY 51<NL>
MEMORY 00000000051 00000000051 006412
```

The first number after the word MEMORY is a logical memory address; the second number is a physical address; and the third number is the value stored at that location. Again, this value is the starting address for PNIC.

Press the carriage return key to reach the SCP-CLI prompt.

d. Now, use a virtual-console or SCP-CLI command to execute the system's panic routine. Generating this spurious panic causes DG/RDOS to run its memory-dump code just as it does after an actual panic.

To enter the panic routine on a DESKTOP GENERATION or DG/500 system, type the starting address of PNIC and R. For example, if the panic routine begins at word 6412, you would type

```
!6412R
```

On an MV/Family system, type START, the starting address of PNIC, and New Line:

```
SCP-CLI> START 6412<NL>
```

At this point, on any machine, the panic routine prints the contents of the accumulators and a fifth number, which is meaningless. Then, you see the prompt

```
Ready xxx, then hit New Line to start the Memory Dump.
```

3. Once you enter the memory-dump code, there is very little to do. Mount your first tape or diskette and press New Line. Then, unless the system encounters a problem, it writes all that it can to your first tape or diskette; it uses 2-Kbyte records for MTn devices and 16-Kbyte records for UTn devices. (If you see the message ERROR while performing the Memory Dump, start the memory dump again using a different tape or diskette.) If the memory dump requires another tape or diskette, you will see the following prompt:

```
Replace xxx, then hit New Line to continue the Memory Dump
```

Simply insert your second diskette or tape and press New Line. You may have to repeat this step, but eventually you will see the message Memory Dump is DONE and then be prompted to begin another dump. To exit from the routine, use the break key sequence. (You may have to type the break sequence three times.)

Restarting DG/RDOS

When you restart DG/RDOS after a panic or a hang, you should reset your I/O devices, determine which files were open when the system went down, and zero the use counts for any files that were open. To perform these tasks, consult the directions below:

1. If your system is hung, type the break key sequence (Cmd-Brk) to return to the virtual console (DESKTOP GENERATION and DG/500 systems) or the System Control Program (MV/Family systems). To reach the SCP CLI, you may have to type the break sequence three times. If your system is not hung, you should already see the virtual-console or SCP-CLI prompt.

2. Use a virtual-console or SCP command to reset your I/O devices. On a DESKTOP GENERATION or DG/500 system, type `I` after the `!` prompt. On an MV/Family system, type `RESET<NL>` after the SCP-CLI prompt.

3. Now, warm boot DG/RDOS by following the directions in the section "Starting Up Your System" up to the point where you have responded to the system's `Filename?` prompt. (A warm boot means that the power to your peripherals and computer was not cut off the last time the operating system went down.)

   At this point, you see the message `Partition in use - type C to continue`. As the message indicates, you should type `C` to continue. Then, on DESKTOP GENERATION systems, you will have to enter the date and time before you reach the CLI's R prompt; on other systems, the date and time are displayed automatically just before the CLI begins executing.

4. Before using your system for normal processing, you should use the CLI command CLEAR/A/D/V to clear the use count on any files that were open when the system panicked or hung. One reason you do this is that you cannot rename or delete a file that has a nonzero count. Another reason is that if you use the /V switch, the system lists on your console the names of the files that were left open. You should keep a list of these files (with the exception of SYS.DR) because files that are not closed normally may not contain current information and may even contain invalid data. Verify the integrity of these files if possible, and if a file has been ruined, delete the file and restore it from your backup media.

   NOTE: Do not clear the use count on, or delete, temporary CLI files (files with names such as CLI.C0, CLI.S0, and CLI.T0). The CLI will manage these files.

After you clear the use count of files that were left open in the master directory, you should go into each directory that was initialized when the system panicked or hung and repeat the CLEAR command:

```
R
DIR CONTRACTS<NL>
R
CLEAR/A/D/V<NL>
Cleared SYS.DR
. . .
R
```

5. Finally, after clearing the use counts on your files, you should use the DIR command with the argument %MDIR% if you changed directories. Then, reboot your operating system by typing `BOOT system-name<NL>`.

Restarting after a Power Failure

If the power to your computer fails, turn the power switches on your computer and all of your peripheral devices to the off position. Then, after the power comes back on, cold start your system according to the directions in the section "Starting Up Your System." You may see the message `Partition in use - type C to continue.` As the message indicates, you should type `C` to continue. Then, on DESKTOP GENERATION systems, you will have to enter the date and time before you reach the CLI's R prompt; on other systems, the date and time are displayed automatically just before the CLI begins executing.

Once you reach the CLI, you should use the CLEAR command to determine which files were open when the system went down and to clear the use count on those files. (The preceding section explains in detail why you need to use this command and how to use it.) Finally, make sure you are in the master directory (use the command `DIR %MDIR%` if you have changed directories) and reboot your system by typing the command `BOOT system-name<NL>`.

-End of Chapter-

Chapter 11

Backing Up and Moving Files

Read this chapter when you want to

- Find out about the uses and the relative advantages and disadvantages of the DG/RDOS backup utilities
- Plan a backup program
- Move files from one DG/RDOS machine to another or from a DG/RDOS machine to an AOS or AOS/VS machine

It is very important to use the DG/RDOS backup utilities to create archive copies of important files and of your entire disk so that you will be able to recover from the accidental deletion of a file or a serious disk problem. You also may want to use a backup program to store data in archives and then recall that data for periodic processing. In addition, the IMOVE program gives you the ability to move data between operating systems.

The remainder of this chapter explains how to create a practical backup scheme and how to transfer files from one system to another. These subjects are covered in the following sections:

- The Backup Utilities
- Planning a Backup Program
- Moving Files Between Systems

The Backup Utilities

The LABEL utility is related to the backup process in that it allows you to read and write diskette and tape labels. However, the utilities that actually let you copy data from a disk or diskette to a backup medium include FCOPY, BURST, and IMOVE. In some instances, one of these utilities provides a unique function, but in other areas the capabilities of the backup programs overlap. Therefore, it can be important to consider which utility is best to use in a given situation.

Here is a list of the functions each utility can perform that the other two cannot:

- To copy a single file from one diskette to another diskette, you must use FCOPY.

- To duplicate a diskette using one diskette drive, you again must use FCOPY.
- To trace the ownership of a disk block, you must use BURST.
- To duplicate the contents of one disk on another disk of the same type, you again must use BURST.
- Finally, to dump selected files or directories from a disk to either diskettes or tape, you must use IMOVE.

There are also two jobs—the second of which is very important—that you can accomplish in more than one way.

- To duplicate the contents of one diskette on another diskette using two drives, you can use either FCOPY or BURST.
- To dump the entire contents of a disk to either diskettes or tapes, you can use either BURST or IMOVE.

When you are performing the first of these jobs, duplicating a diskette using two diskette drives, there are several points of comparison to consider. In BURST's favor, BURST can duplicate a diskette much faster than FCOPY. On the other hand, FCOPY can duplicate any diskette with the proper hardware format, while BURST can duplicate only diskettes that have been initialized with DKINIT and INIT/F. Also, BURST requires that you software format (with DKINIT) your destination diskette whereas FCOPY requires only that the diskette be hardware formatted and have no bad blocks. Thus, you can save the time it takes to run the disk initializer by using FCOPY.

Similarly, the utilities you can use to dump the contents of a disk to diskettes or tape—BURST and IMOVE—both have their merits. The following paragraphs detail the relative advantages and disadvantages of each program so that you can decide which to use in a particular situation.

One slight disadvantage of using BURST is that BURST dumps and loads disk images, exact block-by-block copies of the original disk. As a result, any fragmentation on the original disk will also be present on the disk onto which you load your BURST dump file. This means that disk seek times may be greater than necessary on the target disk and that there may be unnecessary restrictions on the number of large contiguous files you can create. Also, because BURST copies disk images, the original and target disks must share a number of properties in common. For instance, both disks must be the same type; the remap areas on the two disks must be the same size and start at the same location; and the disks' coresident diagnostic areas (if any) must also be the same size and have the same starting address. Since IMOVE performs a logical backup, it does not pose these potential problems.

It is also important to remember that BURST decides which blocks to back up by consulting the master allocation directory (MAP.DR) in the primary partition. This means that when you create a secondary partition, all the blocks in that partition are marked as in use; therefore, if there are few or no files in the partition, BURST backs up a lot of empty blocks.

Also, if MAP.DR in the primary partition becomes corrupt, BURST will not work properly. If bits associated with unused blocks are set to 1, the utility will copy blocks that it shouldn't, and if bits associated with allocated blocks are set to 0, you will lose data.

A final limitation of BURST is that it is more time consuming to run than IMOVE if you are using diskettes for a backup or restore operation. Not only does BURST need more time than IMOVE to dump or load your data, but BURST also requires that you initialize (with DKINIT) your diskettes before you use them in a dump operation. The diskettes you use with IMOVE only need to be hardware formatted and free of bad blocks.

On the other hand, if your backup medium is tape, BURST is usually faster than IMOVE. Therefore, if consolidating your data on the target disk is not important, it may be to your advantage to use BURST to back up your disk to tapes.

Planning a Backup Program

Because the best backup plan for one site may not be the best for another, this section does not propose a standard backup program. Instead, it lists important guidelines that you should consider as you develop your own program.

- You should perform backups frequently. If you do not plan to do a full backup daily, you should back up each day at least the files that you have modified since your last full backup. To do this, use an IMOVE command line that includes the argument/local-switch combination *date*/A/M, where *date* is the date of your last full backup.

  Then, at least on a weekly basis, you should perform a full backup using either BURST or IMOVE. In general, if your backup medium is tape, you will save time by using BURST, and if your backup medium is diskettes, the opposite is true. (For a more detailed comparison of the two utilities, see the preceding section.)

  You may also want to do full backups monthly and yearly. Doing so increases your ability to recover lost data and provides you with a record of what information was on your disk at a particular time.

- For each type of backup you do—daily, weekly, and so on—you should have at least three or four generations of diskettes or tapes. That is, if each daily backup requires two tapes, you need to set aside at least six to eight tapes for daily backups. You use the first two tapes on Monday, the second two on Tuesday, and so forth until you have used all the tapes; then you reuse the first two tapes and restart the cycle. Following this guideline helps ensure that a media problem or a new disk problem does not leave you without a good backup.

- If you use BURST for the majority of your full backups, it is important to use IMOVE occasionally. One reason to use IMOVE is that your IMOVE backup gives you the ability to restore single files without loading an entire backup set. Also, by using IMOVE to dump the contents of your disk, creating a new file system on your disk (INIT/F), and then loading your IMOVE backup, you can generally improve the performance of your system. This IMOVE dump-load process reclaims any spool space that was allocated at the time of a system crash and compacts your files toward the beginning of your disk.

- Whether you use BURST or IMOVE, it is essential that you verify your backup media after every dump operation, especially if you have dumped data to a device that does not do reads after writes. If you are using BURST, you request this verification by appending the /V switch to your DUMP command; after BURST completes its dump, the program asks you to reinsert your backup media in sequence and compares the data on your backup diskettes or tapes with the data on your disk. If you are using IMOVE, you verify your media by issuing a command that includes both the /F and /N switches (for details, see Chapter 14). The utility prompts you to reinsert your backup diskettes or tapes and makes sure that the data on your media is readable. If you do not perform these verification steps, you have absolutely no assurance that your backup is restorable.

By following these guidelines, you should be able to formulate an effective backup strategy, one that will give you the protection you need against the loss of important data.

Moving Files Between Systems

This section deals with the methods you can use to transfer files from one DG/RDOS system to another or from a DG/RDOS system to an AOS or AOS/VS system. Generally, compatible mass storage devices have either the same capacity (such as two 130-Mbyte cartridge tape drives) or the same density (for example, two 1600 BPI reel-to-reel tape drives, or two 96 TPI diskette drives).

DG/RDOS to DG/RDOS

If you are working with two systems that have compatible mass storage devices, moving a file from one machine to the other is very easy. You can use any one of the following methods:

- You can use the CLI command DUMP to dump the file to a tape and then use the LOAD command to move the file onto your second system.

- You can use the CLI command MOVE to copy a file from your disk to a diskette. Then you can carry that diskette to your second system and use the MOVE command to copy the file from your diskette to that system's disk.

- You can use the CLI command XFER to copy the contents of a file to either a diskette or tape and then use the same command to transfer the file onto your second system.
- You can use the IMOVE utility to dump the file to a diskette or tape and then use IMOVE to load the file onto your destination system.
- Under certain circumstances, you can also use BURST to transfer data from one machine to another of the same type. This method works only if (1) you want to transfer everything on one disk to a disk on another system and (2) the destination disk meets the criteria spelled out in the section "Special Disk Requirements" in Chapter 13.

On the other hand, if you want to move data between systems that do not have compatible mass storage devices, your options are more limited. For instance, suppose that you have a DESKTOP GENERATION system with a Model 6270 cartridge tape drive and a 48 TPI diskette drive and an MV/2000 DC system with a Model 6351 cartridge tape drive and a 96 TPI diskette drive. Since the tape drives on the two machines are not compatible and it is not possible to read a 96 TPI diskette (MV/2000 DC system) in a 48 TPI drive (DESKTOP GENERATION system), most of the methods recommended above do not work. In fact, if you are interested in moving data from the MV/2000 DC system to the DESKTOP GENERATION computer, no methods work unless you have a 1600 BPI reel-to-reel tape drive on each system. The only other way of moving data in this direction is to transfer the data over an asynchronous line using a product such as DG/BLAST or DG/XAP.

If you want to move data on diskettes from a system with a 48 TPI diskette drive to a system with a 96 TPI diskette drive, you can use IMOVE to effect the transfer. Simply dump the file you want to move to 48 TPI diskettes and then carry those diskettes to your other system. The 96 TPI diskette drives can read 48 TPI diskettes as long as the information on those diskettes is in IMOVE format.

If you have a DG/500 system with a 60-Mbyte cartridge tape drive, you can read from and write to 22-Mbyte cartridge tapes.

If you have a 150/320/525-Mbyte cartridge tape drive, you can read from and write to 150-Mbyte cartridge tapes.

If none of the descriptions above explains your situation, the only alternative method is to transfer your data over an asynchronous line.

DG/RDOS to AOS or AOS/VS

For transfers between DG/RDOS and AOS or AOS/VS systems, you always use IMOVE from the DG/RDOS side. However, the exact IMOVE command line you use and the utility you use from the AOS or AOS/VS side depend on the type of portable media you use for the transfer. For details on these subjects, see the section "Transferring Files Between Operating Systems" in Chapter 14.

-End of Chapter-

Chapter 12

FCOPY: Backing Up Diskette-Based Files

Read this chapter when you want to

- Duplicate a diskette
- Copy a file from one diskette to another using only one drive

FCOPY is unique among the backup utilities. First, the utility is designed specifically for backing up diskette-based material. Also, while BURST and IMOVE record information in a dump format, FCOPY preserves the material's original format. As a result, you can use an FCOPY backup of a diskette file or of an entire diskette just as you would use the original.

This chapter provides specifics on when and how to use FCOPY. These subjects are covered in the following sections:

- Knowing When to Use FCOPY
- Using FCOPY Interactively
- FCOPY's Command Line Syntax
- FCOPY's Error Messages

Knowing When to Use FCOPY

If your system has two diskette drives, you do not need FCOPY to duplicate a diskette or to copy a file from one diskette to another because the CLI MOVE command can do both of these jobs. In fact, for copying individual files, MOVE is preferable to FCOPY because FCOPY allows you to copy only one file at a time. MOVE, on the other hand, lets you name multiple files on one command line or to use filename templates. For duplicating a diskette, FCOPY has the advantage over MOVE because FCOPY only requires a destination diskette to be hardware formatted and free of bad blocks. The destination diskette in a MOVE operation must have been software formatted with DKINIT and INIT/F.

If your system has only one diskette drive, then FCOPY provides the only method of copying information directly from one diskette to another.

Using FCOPY Interactively

To run FCOPY interactively, move to the master directory (DIR %MDIR%), where FCOPY.SV resides, and at the R prompt, enter either the command

```
FCOPY<NL>
```

or the command

```
CHAIN FCOPY<NL>
```

Use the first of these commands if FCOPY will not read from or write to your master directory, because in this case it does not matter at which level the utility will run. If FCOPY will access the master directory, however, you must use the second command. To access the master directory, the utility must release that directory, which means FCOPY must be running at level 0.

Whichever command you use, the utility responds by presenting the following menu, the first in a series of prompts that lead you through the copying process.

```
     DG/RDOS Diskette Transfer Utility     Rev xx.xx
Command:                    Strike function key 1 for options
------------------------------------------------------------

          1.  Copy a file

          2.  Duplicate a diskette

          3.  Exit from program

     Select desired command:
```

What you do next depends on whether you want to duplicate a diskette using two diskette drives, duplicate a diskette using one diskette drive, or copy a diskette file using one drive.

Duplicating a Diskette Using Two Drives

To copy the contents of a diskette using two drives, select option 2 on the main menu and press New Line. The program now asks you for the unit names of the source and destination diskettes and inquires whether you want to verify the correctness of the duplicate. Respond to these prompts in this manner:

```
Enter source diskette name: DJ0<NL>

Enter destination diskette name: DJ1<NL>

Do you want verification on the duplicate? (Y/N)  Y<NL>
```

By requesting verification, you can ensure that your duplicate diskette is readable.

At this point FCOPY lets you review the information you have supplied to the program and to decide whether you want to go on with the copy.

```
        Writing time will be longer due to verify option

      Source diskette name: DJ0
      Source filename: DJ0
      Source directory pathname: DJ0


      Destination diskette name: DJ1
      Destination filename: DJ1
      Destination directory pathname: DJ1



    Select Continue if the pathnames above are correct
(1) Continue  (2) Restart  (3) Return to main menu
                                Select desired option: 1
```

If you do not want to go on with the copy operation, select option 2 to reenter the unit names of your drives, or choose option 3 to return to the program's main menu. To continue with the copy, press New Line.

If you choose to continue, FCOPY requests that you insert the diskette you want to duplicate in DJ0. Once you have inserted this diskette, press New Line, and FCOPY reads as much of the diskette as available memory in your ground allows (up to 4 MB). Then the program prompts you to insert a destination diskette in DJ1. The diskette you insert must be free of bad blocks and must be hardware formatted; it need not be software formatted. Again press New Line and FCOPY writes the information it just read to the diskette in DJ1. Depending on how much memory is available to FCOPY, the utility may need to repeat the read and write steps one or more times to complete the duplication.

When the copy is complete, FCOPY displays this message:

```
          Transfer and verification completed
------------------------------------------------------------------
  (1) Continue  (2) Restart  (3) Return to main menu
                                          Select desired option: 1
```

To make another copy, press New Line and the program prompts you to insert a second source diskette. Or to return to the CLI, choose option 3; this takes you back to FCOPY's main menu, where you can select option 3 to exit from the program.

When you exit from the program, if you started the utility with the command FCOPY<NL>, you will see the CLI's R prompt. If you started FCOPY by chaining to it, you may see the message `Insert system disk; strike any key to continue.` If you see this message and you originally booted DG/RDOS from a fixed disk, press New Line to return to the CLI; if you booted the system from a diskette, insert your system diskette in your first drive and then press New Line.

Duplicating a Diskette Using One Drive

To duplicate a diskette using only one diskette drive, begin by selecting option 2 on the main menu and pressing New Line. The program now asks you for the unit names of your source and destination diskettes and asks whether you want to verify the correctness of the duplicate. Since you have only one drive to work with, your responses to the first two prompts should be the same. For instance, if you are working on a DESKTOP GENERATION system, you would answer as follows:

```
Enter source diskette name: DJ0<NL>

Enter destination diskette name: DJ0<NL>
```

By requesting verification, you can ensure that your duplicate diskette is readable.

```
Do you want verification on the duplicate? (Y/N)  Y<NL>
```

At this point FCOPY lets you review the information you have supplied to the program and decide whether you want to go on with the copy.

```
          Writing time will be longer due to verify option

        Source diskette name: DJ0
        Source filename: DJ0
        Source directory pathname: DJ0

        Destination diskette name: DJ0
        Destination filename: DJ0
        Destination directory pathname: DJ0


      Select Continue if the pathnames above are correct
(1) Continue  (2) Restart  (3) Return to main menu
                                  Select desired option: 1
```

If you do not want to go on with the copy operation, select option 2 to reenter the unit name of your drive, or choose option 3 to return to the program's main menu. To continue with the copy, press New Line.

If you choose to continue, FCOPY requests that you insert the diskette you want to duplicate in DJ0. Once you have inserted this diskette, press New Line, and FCOPY reads as much of the diskette as available memory in your ground allows (up to 4 MB). Then the program prompts you to remove your source diskette and to insert your destination diskette in DJ0. The diskette you insert must be free of bad blocks and must be hardware formatted; it need not be software formatted. Again press New Line and FCOPY writes the information it just read to the new diskette in DJ0. Depending on the storage capacity of your source diskette and on how much memory is available to FCOPY, the utility may need to repeat the read and write steps one or more times to complete the duplication.

When the copy is complete, FCOPY displays this message:

```
         Transfer and verification completed
------------------------------------------------------------
  (1) Continue  (2) Restart  (3) Return to main menu
                                    Select desired option: 1
```

To make another copy, press New Line and the program prompts you to insert a second source diskette. To return to the CLI, choose option 3; this takes you back to FCOPY's main menu, where you can select option 3 to exit from the program.

When you exit from the program, if you started the utility with the command FCOPY<NL>, you will see the CLI's R prompt. If you started FCOPY by chaining to it, you may see the message `Insert system disk; strike any key to continue.` If you see this message and you originally booted DG/RDOS from a fixed disk, press New Line to return to the CLI; if you booted the system from a diskette, insert your system diskette in your first drive and then press New Line.

Copying a File Using One Drive

To copy a file from one diskette to another using one diskette drive, first select option 1 on the main menu and press New Line. The program now asks you for the pathnames of the source and destination files and inquires as to whether you want to verify the correctness of the copy. In response to the first two prompts, enter a diskette unit name, the name of any secondary partition or subdirectory in the path to the file, and a filename. (The destination file you name should not already exist.) For example, if you are working on an MV/1400 DC system and want to copy a file named ORDERS from a subdirectory called NOVEMBER to a subdirectory of the same name on your backup diskette, you should enter the following pathnames:

```
Enter source pathname: DA10:NOVEMBER:ORDERS

Enter destination pathname: DA10:NOVEMBER:ORDERS
```

By requesting verification, you can ensure that your backup file is readable.

```
Do you want verification on the copy? (Y/N)  Y<NL>
```

At this point FCOPY lets you review the information you have supplied to the program and decide whether you want to go on with the copy.

```
          Writing time will be longer due to verify option

        Source diskette name: DA10
        Source filename: ORDERS
        Source directory pathname: DA10:NOVEMBER

        Destination diskette name: DA10
        Destination filename: ORDERS
        Destination directory pathname: DA10:NOVEMBER



      Select Continue if the pathnames above are correct
(1) Continue  (2) Restart  (3) Return to main menu
                                  Select desired option: 1
```

If you do not want to go on with the copy operation, select option 2 to reenter the pathnames of your files, or choose option 3 to return to the program's main menu. To continue with the copy, press New Line.

If you choose to continue, FCOPY requests that you insert the diskette containing your source file in DA10. Once you have inserted this diskette, press New Line, and FCOPY reads as much of the file as available memory in your ground allows (up to 4 MB). Then the program prompts you to remove your source diskette and to insert your destination diskette in DA10. (The diskette you insert must have been fully software formatted with DKINIT and INIT/F.) Again press New Line and FCOPY writes the information it just read to the new diskette in DA10. Depending on how large your source file is and on how much memory is available to FCOPY, the utility may need to repeat the read and write steps one or more times to complete the copy.

When the copy is complete, FCOPY displays this message:

```
         Transfer and verification completed
-------------------------------------------------------------
  (1) Continue  (2) Restart  (3) Return to main menu
                                    Select desired option: 1
```

To make another copy of the same file, press New Line and the program prompts you to reinsert your source diskette. To copy a different file, select option 2 and the utility prompts you for new source and destination pathnames. To return to the CLI, choose option 3; this takes you back to FCOPY's main menu, where you can select option 3 to exit from the program.

When you exit from the program, if you started the utility with the command FCOPY<NL>, you will see the CLI's R prompt. If you started FCOPY by chaining to it, you may see the message `Insert system disk; strike any key to continue`. If you see this message and you originally booted DG/RDOS from a fixed disk, press New Line to return to the CLI; if you booted the system from a diskette, insert your system diskette in your first drive and then press New Line.

FCOPY's Command Line Syntax

If you invoke FCOPY with the proper switches and arguments, you can bypass the program's first two menus and save a little time. The format of FCOPY's command line and its global switches are shown below.

FCOPY's Syntax

The arguments you use with FCOPY include a description of a source diskette or file and a description of a destination diskette or file:

[CHAIN] FCOPY source destination

If you are duplicating a diskette, *source* should be the unit name of your source diskette (DJ0, for example), and *destination* should be the unit name of your destination diskette. If you are copying a single file, *source* and *destination* must be pathnames of this form:

diskette-unit-name:[secondary-partition:][subdirectory:]filename

The destination file should not already exist.

Use the optional command CHAIN if FCOPY will access your master directory during the course of its operation. In order to access the master directory, the utility must release that directory, which means that FCOPY must run at level 0.

Table 12-1 covers the global switches you can use with the FCOPY command.

Table 12-1 FCOPY's Global Switches

Switch	Description
/C	Tells FCOPY to copy a single file from one diskette to another.
/D	Indicates that you want to duplicate a diskette.
/V	Causes FCOPY to check the copy it produces to ensure that the copy is readable. You should always use this switch.

Examples

The following examples demonstrate how you could use FCOPY's command line syntax to shorten the backup sessions described in the section "Using FCOPY Interactively."

The first of those sessions dealt with duplicating a diskette using two drives (DJ0 and DJ1). To simplify this process, you could invoke FCOPY with the following command line:

```
R
FCOPY/D/V DJ0 DJ1<NL>
```

This command gives the utility all the information that its first two menus ask for. Therefore, you enter the program at the point where FCOPY asks you to confirm the instructions you have given it.

```
        DG/RDOS Diskette Transfer Utility     Rev xx.xx
Command: Duplicate        Strike function key 1 for options
------------------------------------------------------------

        Enter source diskette name: DJ0

        Enter destination diskette name: DJ1

        Do you want verification on the duplicate? (Y/N)  Y
           Writing time will be longer due to verify option

        Source diskette name: DJ0
        Source filename: DJ0
        Source directory pathname: DJ0

        Destination diskette name: DJ1
        Destination filename: DJ1
        Destination directory pathname: DJ1



      Select Continue if the pathnames above are correct
(1) Continue  (2) Restart  (3) Return to main menu
                                  Select desired option: 1
```

From this point you can simply follow the instructions given in the section "Using FCOPY Interactively" for duplicating a diskette using two drives.

A command line for duplicating a diskette using one drive might look like this:

```
FCOPY/D/V DJ0 DJ0<NL>
```

Again, if you issue this command, FCOPY skips to the point where it asks you to make sure you entered the correct unit name for your source and destination diskettes. From that point you can proceed according to the directions for duplicating a diskette using one drive given in the section "Using FCOPY Interactively."

Last, to shorten the file-copy session covered in "Using FCOPY Interactively," you can invoke FCOPY with a command like this one:

```
FCOPY/C/V DA10:NOVEMBER:ORDERS DA10:NOVEMBER:ORDERS<NL>
```

If you do so, FCOPY begins by prompting you to confirm that the pathnames on your command line are correct. From that point you should proceed just as in the interactive session.

FCOPY's Error Messages

Below is a list of error messages you might receive from FCOPY. Each message is accompanied by an explanation of what problem has occurred and how you should respond.

Bad blocks discovered on destination file/device

Your destination diskette has one or more bad blocks. If you were trying to duplicate a diskette, you must use another destination diskette, one with no bad blocks. (You can use the diskette with the bad blocks as a DG/RDOS directory if you software format it.) If you were trying to copy a file to the damaged diskette, you can use DKINIT's PARTIAL command to enter the bad block(s) in the diskette's bad block table and then try your FCOPY operation again.

Can't initialize the directory

This message, which you receive while copying a file, could point to one of several problems. For example, if either your source or destination diskette has not been formatted as a DG/RDOS directory, you generate this error. Also, you could have mistyped the unit name of your source or destination diskette, or you might have inserted one of your diskettes incorrectly or left the door of its drive open.

`Destination block xxxxxx fails verification`

While verifying the readability of your destination diskette, the utility was unable to read the block indicated in the message. Repeat your copy or duplicate operation using a different destination diskette.

`File is not a contiguous or random file`

The file you were trying to copy is a sequential file, and FCOPY cannot copy sequential files. To get around this problem, transfer the contents of the file to a random file using the CLI's XFER command: XFER filename newfilename/R<NL>. Then try copying the file again, being sure to use the new filename in your source pathname.

`File not found`

While attempting to copy a file, you probably specified as part of your source pathname the name of a file that is not present on your source diskette. Or your source diskette may be write-protected.

`I recognize Y, N, y, or n but not this...`

When FCOPY poses a yes-or-no question such as "Do you want verification on the duplicate?", you must answer with a `Y` or `N`, or FCOPY sends you this reminder.

`Illegal input...`

This message indicates that you accidentally entered a non-numeric character while choosing a numbered option from one of FCOPY's menus.

`Incorrect diskette`

The proper destination diskette is not in the diskette drive. Insert the correct diskette and press New Line to continue.

```
Pathname must begin with the diskette name
```

When you are copying a file, FCOPY expects the source and destination pathnames you type in to conform to this format:

diskette-unit-name:[secondary-partition:][subdirectory:]filename

The message above usually indicates that you forgot to include the diskette unit name.

```
Source disk block xxxxxx is bad, 'OWNER' can find any bad file
Select Continue to skip over and continue
```

While you were trying to duplicate a diskette, FCOPY found a new bad block on your source diskette. The utility cannot duplicate this block on the destination diskette; however, if you select CONTINUE, FCOPY can skip over the bad block and duplicate the rest of the source diskette. For information on how to repair your source diskette, see Appendix B.

```
Source diskette contains marginal blocks
```

This message indicates that FCOPY was unable to read one or more of the blocks on your source diskette on the first try. In this case, your destination diskette should be reliable; however, you should treat your source diskette as if a new bad block had developed (see Appendix B).

```
Source file block xxxxxx is bad
Select Continue to skip over and continue
```

While you were trying to copy a file, FCOPY found a new bad block on your source diskette. In this instance, there is little value to selecting CONTINUE because you are only copying one file and that file is damaged. For information on how to repair your source diskette, see Appendix B.

```
Space not available
```

You asked FCOPY to copy a file to a destination diskette that is too full to accommodate the new file.

`This value is out of range:`

While selecting an option from one of FCOPY's two menus, you entered a number that was not in the range of 1 to 3.

`Unable to close the file/device`

You may have tried to copy a file from a tape to a diskette; that is, you may have entered a tape drive's unit name as the source diskette name. FCOPY will not read from or write to tape.

`Unable to delete the existing file`

If you are trying to copy a file and a file already on the destination diskette has that same name, FCOPY gives you the opportunity to delete the file on the destination diskette and then perform the copy. The message above appears if you elect to delete the file, but the file is permanent.

`Unable to exclusively access the device`

While you are duplicating a diskette, this message may appear for one of several reasons. For example, you may have made a slight mistake in entering one of your diskette names (typed `DJO` instead of `DJ0`). Or you may have inserted one of your diskettes incorrectly or left one of your drive doors open. Another possibility is that before you issued your FCOPY command in one ground, the other ground had initialized one of the diskette drives you planned to use for your duplication.

`Unable to open the file/device`

You probably tried to copy a file onto a write-protected diskette. However, you may also have asked FCOPY to read from or write to a tape during a duplicate operation. FCOPY can only duplicate the contents of one diskette on another diskette.

`Unable to read source file/device`

This message usually indicates that you tried to copy a file that is read protected. You can use the CLI's CHATR command to remove the file's R attribute and then try copying the file again.

```
Unable to write to destination file/device
```

You probably tried to duplicate a diskette, and your destination diskette was write protected.

```
WARNING: Destination file already exists
Is this the source diskette? (Y/N)
```

This message will appear while you are copying a file if the filename you include as part of your destination pathname matches the name of a file on the diskette in your destination drive. If you have the correct diskette inserted, answer `N` to FCOPY's question, and the utility will then give you the option of deleting the existing file whose name corresponds to that of the file you want to copy: `Select CONTINUE to delete`. If you do not have the correct diskette inserted, answer `Y` and correct your mistake.

-End of Chapter-

Chapter 13

BURST: Performing Disk-Image Backups

Read this chapter when

- You want to dump the entire contents of a disk to either diskettes or tapes, or restore the entire contents of a disk from BURST backup media
- You want to copy the entire contents of a disk to another disk
- You want to determine which file owns a given disk block

BURST is a stand-among utility whose main function is to dump and restore disk images. However, the utility also enables you to duplicate the contents of one disk on another of the same type and to trace the ownership of disk blocks. This chapter discusses these functions in the following sections:

- An Overview of BURST
- Special Disk Requirements
- Starting the Program
- Using the DUMP Command
- Using the LOAD Command
- Using the DUPLICATE Command
- Using the OWNER Command
- BURST's Command Line Syntax
- Creating a Stand-Alone BURST Tape
- BURST's Error Messages

An Overview of BURST

Whether you use BURST to dump or duplicate the data on a disk or to load the contents of a backup set, the utility employs a lot of the same logic. For instance, BURST always consults the master allocation directory, MAP.DR, and backs up or restores only those disk blocks that the file system indicates are in use. Also, if you request verification of the results of a backup or load operation, BURST reads the data on the source and destination media into separate buffers and then compares that data. As an example, here is what happens if you ask the utility to dump the contents of your disk.

BURST first dumps blocks 0 through 3 and 5 through 17 (octal). Then, it consults the master allocation directory MAP.DR, which begins at block 17, to determine which of the remaining blocks the system has allocated. MAP.DR is a contiguous file in which each bit corresponds to a disk block. If a bit is set to 0, BURST understands that the corresponding block is not in use and does not back it up; if a bit is set to 1, BURST does back up the indicated block. In instances where a bad block has been remapped and allocated, the operating system guides BURST to the correct block in the remap area.

If you asked BURST to verify the accuracy of this dump, the utility makes another complete pass through the disk and the archive diskettes or tapes, reading blocks from each medium into separate buffers. As BURST compares the data in the two buffers, it reports the octal block number of any disk block that was not recorded correctly.

To further ensure the reliability of the backup-and-restore process, BURST also writes diskette or tape labels during a dump operation and checks those labels when you restore your BURST backup. The most important pieces of information written in these labels are as follows:

1. A filename in the HDR1 label that consists of the unit name of the disk whose contents are being dumped, the date, and the time the dump started

2. A sequence number in the HDR1 label that indicates a diskette or tape's place in a backup set

3. A block count in the EOV1 or EOF1 label that specifies how many blocks BURST has dumped so far

When you load your BURST backup media, BURST reads and remembers the contents of the filename field in the first diskette or tape's HDR1 label and then checks to make sure that all subsequent diskettes or tapes have the same unit name, date, and time in their filename fields. In addition, the utility looks at each diskette or tape's sequence number to ensure that you are inserting your media in the proper order. Also, after loading the contents of each diskette or tape in a backup set, BURST compares the number of blocks it has loaded to the block count in the EOV1 or EOF1 label on that diskette or tape.

To read these labels, use the LABEL utility, which is discussed in Chapter 15.

If you use BURST to trace the ownership of disk blocks, the utility works like this. You supply the octal block number of one or more disk blocks—usually these are new bad blocks. BURST then begins searching the system directories, SYS.DRs, on your disk to find the descriptors of the files that might own these blocks. If the program sees a descriptor for a contiguous file, it can tell immediately whether the file owns a block you are looking for because all it needs to know is the file's starting address and size. For a random file, BURST must search the file's index structure to determine which blocks the file contains, and for a sequential file, the utility must trace the file from beginning to end.

One potential problem with this approach is that if a new bad block is part of a random file's index structure or part of a sequential file (or part of a system directory), the OWNER routine cannot run to completion. The program prints an error message and terminates. (If you encounter this problem, consult Appendix B.) If, on the other hand, each block you are interested in is a random file data block or part of a contiguous file, OWNER reports the pathname of the file that owns each block and terminates normally. If the routine completes normally but does not report that a block you inquired about belongs to a file, you can assume the system is not using that block. In such a case, it is safe to log the bad block in the bad block table using DKINIT's ENTER command.

Special Disk Requirements

Whether you use BURST to duplicate a disk or to dump the contents of a disk to tapes or diskettes so that you can later restore those contents, the ultimate destination for a BURST backup is always a disk. This disk may or may not be the same as the source disk; however, in either case, the source and destination disks must share several properties for BURST to work correctly.

- The destination disk must be the same size as the source disk.
- The remap areas on the two disks must be identical in size and have the same starting address.

  NOTE: If you software format two disks of the same type and request the default placement for the remap area on each, there is no guarantee that the two remap areas will start at the same address. DKINIT attempts to put each remap area at the end of the disk; however, if one disk has a bad block at one of its highest addresses, its remap area will have to precede that bad block.

- If the source disk has a coresident diagnostic area, the destination disk must also have such a diagnostic area, and that area must occupy the same number of blocks and begin at the same location as the diagnostic area on the source disk. If the source disk does not have a coresident diagnostic area, the destination disk cannot have one either.

- Neither the source nor destination disk should have a bad block that has not been entered in its bad block table. If BURST encounters a new hard error, it cites the location of the bad block—by unit name and block number—and then terminates.
- The source and destination disks should have the same unit number. The reason for this is that the usual source disk contains quite a few files that are links to files elsewhere on the disk. If the contents of this disk are transferred to a disk with a different unit number, these links will not work as they were intended to.

These restrictions will not concern you much if you use BURST to create an archive copy of a disk that you would restore to that same disk if a serious system failure occurred (unless you run DKINIT on that disk between the backup and restore sessions). However, if you are duplicating a disk or creating a backup set that you plan to load onto a disk on another system, it is essential to be aware of these disk requirements.

Starting the Program

Since BURST is a stand-among program, you start it from the CLI. At the R prompt, type in a command that follows this format:

[CHAIN] BURST [buffer-size/S]

Use the first optional element, the CLI command CHAIN, if BURST will access your master directory. To access the master directory, the utility must release that directory, and in order to release the master directory, BURST must be running at level 0.

The *buffer-size/S* argument/local-switch combination tells the utility what size buffers to use for a dump or load operation in place of the default 8-Kbyte buffers. The legal buffer sizes are 8192, 10240, 12288, 14336, and 16384.

You may need to experiment to find out which buffer size works best when you are dumping data to, or loading data from, a reel-to-reel tape. The performance of some cartridge tape drives is improved greatly if you request large buffers; see Appendix A for recommendations.

In any case, the utility introduces itself with the following message:

```
        Disk backup and block owner utility
  For help, enter ? as the response to any question
   or HELP as the response to the Command? question

                Command?
```

In response to the Command? prompt, either type ?<NL> to display information about BURST's commands or enter one of the commands listed in Table 13-1.

Table 13-1 BURST Commands

Commands	Description
DUMP[/V]	Dumps the contents of a disk to diskettes or tapes. If you use the /V switch, after the program performs the dump, it compares the data on your disk to the data it writes on your backup media.
DUPLICATE[/V]	Copies the contents of a disk to another disk of the same type. If you use the /V switch, the utility, after performing the copy, compares the information on your source and destination disks.
HELP	Displays information about using BURST.
LOAD[/V]	Restores the information on your BURST backup media to your disk. If you use the /V switch, after the program finishes the load operation, it compares the data on your backup diskettes or tapes to the data it writes to disk.
OWNER	Determines which file, if any, owns a given disk block.
STOP	Terminates BURST.

Using the DUMP Command

Generally, you use the DUMP command to create an archive copy of a disk. Then, if a catastrophic system failure occurs, you can restore the entire contents of your disk as it existed at the time of your last backup. Alternatively, you can take your backup diskettes or tapes to another system and load their contents onto a disk of the same type as the original. Before doing so, however, you should read carefully the section earlier in this chapter called "Special Disk Requirements."

To begin dumping the contents of your disk to either diskettes or tapes, respond to BURST's Command? prompt by typing DUMP/V<NL>. The program then asks you, in turn, to name a source disk, a primary backup device, and a secondary backup device. Answer the first two queries by typing in the unit name of the disk you want to back up and the unit name of the diskette or tape drive that will hold your backup media. Answer the third prompt only if you are dumping data to diskettes or tapes and have two diskette drives or two tape drives; in such a case, you can enter the unit name of your second drive, and BURST dumps data alternately to the two units.

For example, if you are working on an MV/2000 DC system and want to dump the contents of DA0 to diskettes, the first part of your dialog with BURST looks like this:

```
                Command?  DUMP/V<NL>
           Source disk?  DA0<NL>
Primary tape or diskette?  DA10<NL>
      Secondary diskette?  <NL>

        Dump in progress

Insert diskette # 1 in drive DA10; enter New Line to continue
```

At this point, insert in DA10 a diskette that has been initialized with DKINIT. Then press New Line. BURST begins dumping data to this diskette and, if it fills your first diskette, prompts you for a second.

```
Insert diskette # 2 in drive DA10; enter New Line to continue
```

Again, insert a diskette you have formatted with DKINIT and press New Line. This process continues until BURST has dumped the entire contents of your disk.

BURST then displays the message `Verification in progress` and prompts you to insert your backup diskettes, in order, in DA10. As you insert each diskette and press New Line, the utility compares the data on that diskette to the appropriate data on your disk to ensure that you have a good backup. When this process is complete, BURST prints the message `Command completed` and prompts you for another command.

To leave the utility, type `STOP<NL>`. If you did not chain to the utility, you will see the CLI prompt immediately. Otherwise, you may see the message `Insert system disk; strike any key to continue`. If you see this message and you originally booted DG/RDOS from a fixed disk, press New Line to return to the CLI; if you booted the system from a diskette, insert your system diskette in your first or only drive and then press New Line.

To dump the data on your disk to tapes instead of diskettes, you need to make only a couple of adjustments. Of course, you should supply the unit name of your tape drive in response to the `Primary tape or diskette?` prompt. Also, you can enter a tape file number after this unit name if you want to preserve existing files on your first backup tape. For example, if you type `UT0:4<NL>`, BURST skips over tape files 0, 1, 2, and 3 and then begins writing the dump file.

NOTE: If you supply a tape file number, be sure to make a note of it because you will have to use that number again when you load the contents of your backup tape(s).

The following dialog might occur if you were working on a DESKTOP GENERATION system and wanted to dump the data on DE0 to tapes:

```
                Command?  DUMP/V<NL>
             Source disk?  DE0<NL>
Primary tape or diskette?  MT0<NL>
          Secondary tape?  <NL>

        Dump in progress

Mount tape # 1 on drive MT0; enter New Line to continue<NL>

Mount tape # 2 on drive MT0; enter New Line to continue<NL>

Verification in progress

Mount tape # 1 on drive MT0; enter New Line to continue<NL>

Mount tape # 2 on drive MT0; enter New Line to continue<NL>

                Command completed

          Disk backup and block owner utility
   For help, enter ? as the response to any question
    or HELP as the response to the Command? question

                Command?  STOP<NL>

Insert system disk; strike any key to continue<NL>
```

Using the LOAD Command

Most frequently, you use the LOAD command to restore an archive copy of a disk following a serious system failure. However, you can also take BURST diskettes or tapes created on one system and use the command to load their contents onto another system that includes a disk of the same type as the source disk. Before using the LOAD command for the latter purpose, read carefully the section earlier in this chapter called "Special Disk Requirements."

To begin loading the contents of a set of diskettes or tapes onto your hard disk, respond to BURST's Command? prompt by typing LOAD/V<NL>. The utility then asks you, in turn, to name a destination disk, a primary backup device, and a secondary backup device. Answer the first two questions by typing in the unit name of the disk whose contents you are restoring and the unit name of the diskette or tape drive that will hold your backup media. Answer the third prompt only if you are loading data from diskettes or tapes and have two diskette drives or two tape drives; in that case, you can enter the unit name of your second drive, and BURST loads data alternately from the two units.

For instance, if you are working on a DESKTOP GENERATION system with two diskette drives and want to load the contents of a set of BURST diskettes onto DE0, the first part of your dialog with BURST looks like this:

```
                Command?  LOAD/V<NL>
        Destination disk?  DE0<NL>
Primary tape or diskette?  DJ0<NL>
      Secondary diskette?  DJ1<NL>

        Load in progress

Insert diskette # 1 in drive DJ0; enter New Line to continue
```

At this point insert your first and second backup diskettes in DJ0 and DJ1 respectively. Then type New Line. BURST begins loading data from the diskette in DJ0 and, when this diskette is exhausted, prompts you to insert your second diskette.

```
Insert diskette # 2 in drive DJ1; enter New Line to continue
```

You have already inserted this diskette, so just press New Line. Then, when BURST requests that you insert your third diskette, you can insert both your third and fourth diskettes. This process continues until BURST has loaded the entire contents of your backup set.

BURST then displays the message Verification in progress and prompts you to insert your backup diskettes, in order, in DJ0 and DJ1. As you insert each diskette and press New Line, the utility compares the data on that diskette to the appropriate data on your disk to ensure that you have a good backup.

When this process is complete, BURST prints the message Command completed and prompts you for another command.

To leave the utility, type STOP<NL>. If you did not chain to the utility, you will see the CLI prompt immediately. Otherwise, you may see the message Insert system disk; strike any key to continue. If you see this message and you originally booted DG/RDOS from a fixed disk, press New Line to return to the CLI; if you booted the system from a diskette, insert your system diskette in your first or only drive and then press New Line.

To load data from a set of backup tapes instead of diskettes, you use a procedure very similar to the one illustrated above. Simply enter the unit name of your tape drive in response to the Primary tape or diskette? prompt, and if the BURST dump file does not begin on file 0 of your first backup tape, follow this unit name with the number of the tape file at which the dump file does begin. For instance, if you type MT0:2, BURST skips over tape files 0 and 1 before it begins loading data.

The following dialog might occur if you were working on an MV/2000 DC system and wanted to load the contents of a set of cartridge tapes onto DA0:

```
                Command?  LOAD/V<NL>
       Destination disk?  DA0<NL>
Primary tape or diskette?  UT0<NL>
         Secondary tape?  <NL>

       Load in progress

Mount tape # 1 on drive UT0; enter New Line to continue<NL>

Mount tape # 2 on drive UT0; enter New Line to continue<NL>

Verification in progress

Mount tape # 1 on drive UT0; enter New Line to continue<NL>

Mount tape # 2 on drive UT0; enter New Line to continue<NL>

                Command completed

          Disk backup and block owner utility
    For help, enter ? as the response to any question
     or HELP as the response to the Command? question

                Command?  STOP<NL>

Insert system disk; strike any key to continue<NL>
```

Using the DUPLICATE Command

The DUPLICATE command allows you to copy all of the allocated blocks on one disk or diskette onto another disk or diskette. However, the destination disk in such an operation must meet all of the criteria detailed in the section earlier in this chapter called "Special Disk Requirements." If your destination disk does meet these criteria, you can start the duplication process by responding to BURST's Command? prompt with one of these commands:

```
DUPLICATE<NL>
```

or

```
DUPLICATE/V<NL>
```

If you use the /V switch, DUPLICATE not only copies the contents of your source disk onto your destination, but, after the duplication, compares the information on the two disks to ensure that there are no discrepancies.

BURST then asks for the unit names of your source and destination disks. (Refer to Appendix A for a list of disks and their unit names.) So the next part of your dialog with BURST might look like this:

```
          Source disk?  DE0<NL>
     Destination disk?  DE1<NL>
```

Once you have provided these names, the program performs the duplication and verifies the copy if you requested such verification. When the process finishes, BURST prints the message Command completed successfully and prompts you for another command.

To terminate the program, type STOP<NL>. If you did not chain to the utility, you will see the CLI prompt immediately. Otherwise, you may see the message Insert system disk; strike any key to continue. If you see this message and you originally booted DG/RDOS from a fixed disk, press New Line to return to the CLI; if you booted the system from a diskette, insert your system diskette in your first or only drive and then press New Line.

Using the OWNER Command

The OWNER command allows you to ascertain which file, if any, owns a particular disk block. Therefore, if BURST is unable to restore all the data in a backup set to your disk, you can use OWNER to find out which files were corrupted. Or if DKINIT's TEST function indicates that you have a new bad block on your disk, you can use OWNER to determine whether the flawed block belongs to a file before you enter the block in the bad block table.

To start OWNER, respond to BURST's Command? prompt with the command

```
OWNER<NL>
```

The routine responds by asking for the unit name of the disk you are working with. If you are not sure of your disk's unit name, refer to Appendix A.

```
Disk? DE0<NL>
```

Then OWNER asks for an octal block number. Normally, this is the number of a block that either DKINIT or one of BURST's error messages pointed out as being unreliable. You need not include leading zeros when you enter this number; for example, 000045 becomes 45.

```
Block number? 11554<NL>
Block number?
```

OWNER repeats this question 32 times or until you respond by pressing New Line.

```
Block number? <NL>
```

The routine then reports the pathnames of the files that own the blocks you inquired about:

```
The following blocks  are owned by the following files:
              11554  DE0:CONFIG.SV

                  Command completed

          Disk backup and block owner utility
     For help, enter ? as the response to any question
      or HELP as the response to the Command? question

                  Command?
```

To leave the utility, type STOP<NL>. If you did not chain to the utility, you will see the CLI prompt immediately. Otherwise, you may see the message `Insert system disk; strike any key to continue`. If you see this message and you originally booted DG/RDOS from a fixed disk, press New Line to return to the CLI; if you booted the system from a diskette, insert your system diskette in your first or only drive and then press New Line.

BURST's Command Line Syntax

In addition to running in interactive mode, BURST also supports a command-line interface. This interface allows you to supply a BURST command and any necessary arguments at the same time that you start the program from the CLI. The exact formats of the various command lines are shown in the following subsections.

HELP

To execute the HELP command, type

```
BURST HELP<NL>
```

or

```
BURST/H<NL>
```

The system displays a multipage help message and then returns you to the CLI.

NOTE: If you use the HELP command—or any other command—in interactive mode, BURST performs one of its operations and then prompts you for another command. If you use the command-line interface, BURST performs the operation and then returns you to the CLI.

DUMP and LOAD

For a dump or restore operation, the command line syntax is as follows:

[CHAIN] BURST [buffer-size/S] { DUMP[/V] / LOAD[/V] } disk backup-dev-1 [backup-dev-2]

Use the first optional element, the CLI command CHAIN, if BURST will access your master directory. To access the master directory, the utility must release that directory, which means that BURST must be running at level 0.

The *buffer-size/S* argument/local-switch combination tells the utility what size buffers to use in place of the default 8-Kbyte buffers. See Appendix A for recommendations.

The *backup-device-2* argument allows you to dump data to or load data from two devices alternately (two diskette drives or two tape drives).

You use the remaining command-line elements just as you would expect. For example, if you are working on an MV/2000 DC system and want to dump the contents of DA0 to diskettes and to verify that dump, you enter the command

```
CHAIN BURST DUMP/V DA0 DA10<NL>
```

To restore the contents of DA0, you would simply change the word DUMP to LOAD.

DUPLICATE

With the DUPLICATE command, you must supply two arguments: the unit name of the disk whose contents you want to copy and the unit name of the disk that is to receive the copy:

[CHAIN] BURST DUPLICATE[/V] source-disk destination-disk

For example, to copy the contents of DA0 onto DA1, you would use the command line

```
CHAIN BURST DUPLICATE/V DA0 DA1<NL>
```

OWNER

The command line syntax for the OWNER command is as follows:

[CHAIN] BURST OWNER disk-unit-name block-no-1 [block-no-2...block-no-32]

Creating a Stand-Alone BURST Tape

In the past, documentation of the BURST utility has included a section on building a stand-alone tape that you could use to rebuild your disk quickly in the event of a disk crash or the loss of important data (such as the primary partition's MAP.DR or SYS.DR). Basically, the process involved placing TBOOT on tape file 0, transferring BURST to file 1, and beginning the BURST dump on file 2. Then, if you developed a serious disk problem, you could boot BURST from tape and load the data on the tape starting at file 2.

Now that BURST runs on top of the operating system, such a tape will not work. You can still create a stand-alone, restorable BURST tape, but that tape must include an operating system. Also, since it is not possible to boot a program like BURST from tape, you must provide a mechanism for getting the program onto a disk so that you can execute it from there. The specific steps required to build such a tape follow.

1. Write bootstrap code to file 0 of your magnetic tape by running the stand-among tape bootstrap program TAPEBOOT.SV. For instance, if you are working on a DESKTOP GENERATION system, issue the command:

   ```
   R
   TAPEBOOT MT0<NL>
   ```

2. Using the CLI DUMP command, dump the files CLI.SV and BURST.SV to tape file 1:

   ```
   DUMP MT0:1 CLI.SV BURST.SV<NL>
   ```

3. Transfer an operating system .SV file to tape file 2. For example, you might use this command line:

   ```
   XFER DGRDOS.SV MT0:2<NL>
   ```

4. If the operating system requires an overlay file, transfer the .OL file to tape file 3:

   ```
   XFER DGRDOS.OL MT0:3<NL>
   ```

5. Using the BURST utility, dump the contents of your disk starting at tape file 4 (or 3 if your operating system does not need an overlay file):

   ```
   CHAIN BURST DUMP/V DEO MT0:4<NL>
   ```

If you follow these steps, you will produce a tape that gives you the same capabilities as the old-style stand-alone BURST tapes; that is, you will be able to load the contents of your BURST dump file onto a disk that has been formatted with DKINIT so that its format matches that of the original disk. If you also want the capability to load your BURST dump file onto a disk that has not been software formatted or one that needs to be reformatted, you should transfer DKINIT.SV to file 4 of your restorable tape and begin the BURST dump on file 5. (If you did not transfer an operating system overlay file to tape file 3, you should transfer DKINIT to tape file 3 and begin the BURST dump at file 4.) Then you can run the disk initializer before you set up your disk's system and master allocation directories and load your BURST backup.

In any case, once you have created your stand-alone BURST tape, restoring the contents of a damaged disk is fairly easy. Simply proceed according to the instructions listed below.

1. If you are working on a DESKTOP GENERATION system, turn on your computer. Then, when you see the virtual console prompt (!), insert your stand-alone BURST tape in drive MT0, and type 22H. At this point, the bootstrap code on your tape begins executing and displays the message

   ```
   Number of tape file to boot? File =
   ```

   If you are working on a DG/500, MV/1000 DC, MV/1400 DC, MV/2000 DC, MV/2500 DC, MV/3500 DC, MV/5000 DC Series, MV/7800 XP, MV/9500, MV/15000, or MV/18000 system, follow these directions:

   a. Turn on your computer. After the system completes its power-up tests, your console displays the Automatic Program Load Menu.

NOTE: If you are working on a DG/500, MV/1000 DC, MV/1400 DC, MV/2000 DC, or MV/2500 DC and your coresident diagnostic area has been damaged, you may have to load your System Media from a diskette or tape. In this case, after restoring the contents of your disk, you would also need to reinstall the System Media on your fixed disk.

b. Before the system continues its startup using preset values, type 2<NL>. This causes the system to present its Change Preset Values Menu.

c. Type 6<NL> to reach the virtual console (DG/500 systems) or the System Control Program's Command Line Interpreter (MV/Family systems).

d. If you have an MV/3500 DC or MV/5000 DC Series system, you must mount your System Media tape or diskette. Enter the command BOOT *n [optional-unit-number]*<NL>, where *n* is the appropriate device code of the unit on which you have mounted your System Media. (See Appendix A for a complete list of device codes.) Your console will display the Microcode Update and Installation menu; choose to load microcode into memory. After the system's microcode is loaded, choose to return to the SCP-CLI from the menu.

e. If you have an MV/7800 XP, MV/9500, MV/15000, or MV/18000 system, you must mount your System Media tape or diskette and boot your system's microcode. Enter the command BOOT *n*<NL>, where *n* is the appropriate device code of the unit on which you have mounted your System Media. (See Appendix A for a complete list of device codes.) You will see messages indicating that your system's microcode is being loaded, and then you will be returned to the SCP-CLI.

f. If you are working on a DG/500 system, insert your tape in drive MT0 and then type 22H. If you are working on an MV/Family system, mount the stand-alone BURST tape on the tape drive and type BOOT *n*<NL>, where *n* is the appropriate device code.

At this point you will see the prompt

```
Number of tape file to boot? File =
```

2. In response to the prompt `Number of tape file to boot? File =`, enter the tape file number of the program you want to boot. If you need to run the disk initializer, type the number of the file that contains DKINIT, usually file 4. Otherwise, enter a 2 to start up DG/RDOS. When the operating system begins execution, it writes the following message:

```
Full (F), Restore (R), or Partial (P or -RETURN-)?
```

3. Type F<NL> to indicate that you want DG/RDOS to perform the equivalent of an INIT/F on your disk or diskette. The system then asks you for the unit name of the disk you are working with:

   ```
   Initializing what disk?
   ```

   Supply the appropriate unit name—DE0, for example—and press New Line.

4. If you are working on a DESKTOP GENERATION system, the operating system prompts you for the date and time. Respond to its questions as you typically do when you boot DG/RDOS.

   ```
   Date (m/d/y) ? MAR 12 88<NL>
   Time (h:m:s) ? 9 09 AM<NL>
   ```

   On other systems, the time and date are simply displayed.

   On all systems, DG/RDOS then loads onto your disk the files that you dumped earlier onto tape file 1—CLI.SV and BURST.SV—and starts up the CLI.

5. At the CLI's R prompt, issue a BURST command to restore the contents of your disk. For instance, if you began your BURST dump on tape file 4, you might type

   ```
   CHAIN BURST LOAD/V DE0 MT0:4<NL>
   ```

   As BURST begins execution, it prints its startup message and then asks you to insert your stand-alone BURST tape.

   ```
   Mount tape # 1 on drive MT0; enter New Line to continue
   ```

   From this point on, the procedure for loading your backup is the same as it would have been had the BURST dump begun at tape file 0.

BURST's Error Messages

Below is a list of error messages you may receive from BURST. Each message is accompanied by an explanation of what problem has occurred and of how you should respond.

```
Block is already in list; not duplicated
```

While running the OWNER routine, you entered the same block number twice. OWNER will not include any one block number in its search list more than once.

```
Block list is full; starting search
```

OWNER can accommodate a maximum of 32 block numbers in its search list at one time. Therefore, if you give the program 32 block-number arguments, it prints this message and starts its search automatically (that is, it does not wait for you to press New Line in response to its `Block number?` prompt).

```
Directory depth exceeded
```

OWNER reports this message if it sees a directory subordinate to a subdirectory. Such a situation indicates that your disk is corrupt.

```
Error xxxxxx at location xxxxxx; aborting
```

The first number in this message is a system error code whose meaning you can look up in the parameter file DGPARU.SR. For example, if, during a dump operation, you mistyped the unit name of your source disk, you would get system error 52, which means "Illegal directory specifier."

```
Error xxxxxx; cannot restore the following portion(s) of the disk:
xxxxxx blocks starting at xxxxxx
```

The first number in this message is a system error code whose meaning you can look up in the parameter file DGPARU.SR. For instance, if BURST is verifying the accuracy of a dump and encounters one or more unreadable blocks, you receive the system error 30, which indicates a "File data error."

```
Error xxxxxx while reading from disk Dxx
Bad block =
```

The first number in this message is a system error code whose meaning you can look up in the parameter file DGPARU.SR. For example, if BURST encounters a bad block while reading your source disk, you may get the message `Error 30`. According to DGPARU.SR, error 30 is a "File data error."

```
Error during diskette initialization; try another diskette
```

You may receive this message for one of a number of reasons. One likely problem is that you tried to dump data to a diskette that had not been initialized with DKINIT. Or there could be some more obvious problem: your destination diskette is write protected; you inserted your destination diskette incorrectly; or you left the door on your destination drive open.

```
Error during tape initialization; try another tape
```

You are probably trying to write data to a write-protected tape. However, you might also be trying to load data from a tape that does not have a BURST-format VOL1 label; in this case you will also see the message `No BURST format VOL1 label found`.

```
Failure xxxxxx xxxxxx
xxxxxx xxxxxx xxxxxx
Error xxxxxx at location xxxxxx; aborting
```

On the first line, "reading from," "writing to," "opening," or "closing" indicates the action that failed, and the name of the failing device is displayed. Here is an example: `Failure reading from UT0`.

The three numbers on the second line are values that will help Data General personnel diagnose the problem.

The first number of the third line is a system error code whose meaning you can look up in the parameter file DGPARU.SR.

BURST displays this type of error message when the program cannot recover from an error involving a backup device (diskette or tape) during a DUMP or LOAD command. Examples include most errors writing to a backup device during a DUMP, or an error on one of the first few reads from a backup device during a LOAD.

`Invalid xxxxxxxxxx in xxxx label`

Chances are that while restoring the contents on a multivolume backup set, you inserted one of your diskettes or tapes out of sequence. This type of error will produce such messages as `Invalid sequence number in HDR1 label` and `Invalid record count in HDR1 label`.

`Invalid block number; not added to list`

While using the OWNER routine, you either entered a block number in decimal instead of octal or entered a number higher than the number of blocks on your disk.

`Invalid command or argument; use HELP command for help`

Most likely you mistyped a command or issued a nonexistent one. The valid commands in BURST are shown in Table 13-1.

`Invalid remap area on target disk`

You may see this message while you are trying to load a backup set or duplicate a disk. In either case, the problem is that the remap area on the original disk and that on the destination disk differ either in size or starting location. (BURST displays both the size and starting location of the remap area on each disk.) The only solution to this problem is to use DKINIT to change the size or starting address of the remap area on the destination disk.

`Invalid target disk size`

This message can appear while you are loading a backup set or duplicating a disk. For either operation to succeed, the original disk and the target disk must be exactly the same size. (BURST displays the size of both disks.) If the two disks have the same model number, perhaps one has a coresident diagnostic area and the other does not, or maybe the two disks have coresident diagnostic areas of different sizes. If the problem does lie in this area, you can use the DKINIT utility to alter the diagnostic area on the destination disk to match that on the original disk.

`Mismatch in block # xxxxxx`

This message indicates that BURST found an error while verifying the results of a DUMP, LOAD, or DUPLICATE operation. The utility was able to read both the disk and the backup media, but the data read in from those two sources was not identical.

```
No BURST format VOL1 label found
```

Most likely you are trying to load data from a tape that was not created with the BURST utility. If you know that the tape is labeled, perhaps the tape's VOL1 label was written by the LABEL utility or by IMOVE.

```
Warning: That block is owned x times
```

OWNER counts the number of files that own any one block in its search list. MAP.DR blocks may be owned multiple times: by the disk, its subpartitions, and its subdirectories. However, if OWNER finds a block besides a MAP.DR block that is owned by more than one file, the routine prints this message to warn you that your disk is probably corrupt.

-End of Chapter-

Chapter 14

IMOVE: Making Full and Incremental Backups

Read this chapter when you want to

- Dump files or directories from your disk to a single-volume diskette or tape, or restore material in IMOVE format from such media
- Dump files or directories from your disk to multi-volume diskettes or tapes, or restore material in IMOVE format from such media
- Move files between a DG/RDOS and an AOS or AOS/VS system

IMOVE is the most flexible and useful of the DG/RDOS backup utilities and commands. As with BURST, you can use IMOVE to back up an entire disk; however, IMOVE also lets you dump selected files and directories. The CLI command DUMP gives you the ability to back up selected files too, but IMOVE has major advantages over DUMP. For example, IMOVE is much faster than DUMP, and IMOVE can handle multidiskette and multitape backup sets whereas DUMP cannot. You can also use IMOVE to transfer files between DG/RDOS and other operating systems.

The rest of this chapter describes both the specific capabilities of IMOVE and how to use them. These subjects are covered in the following sections:

- An Overview of IMOVE
- IMOVE's Syntax
- Selecting Files to Move
- Using Unlabeled Diskettes and Tapes
- Using Labeled Diskettes and Tapes
- Using MMOVE-Format Diskettes
- Transferring Files Between Operating Systems
- IMOVE's Error Messages

An Overview of IMOVE

As mentioned above, IMOVE is a very flexible backup utility. For instance, IMOVE allows you to dump your whole disk or to dump only selected files or directories. IMOVE can use three formats: unlabeled (single-volume diskette or tape), labeled (multivolume diskettes or tapes), and MMOVE format (multivolume diskettes). IMOVE enables you to back up data to unlabeled, labeled, or MMOVE-format media and to create multidiskette or multitape backup sets. Also, the utility writes in dump formats that other Data General operating systems can read and is able to read backup sets written under those same systems.

To select which files IMOVE will dump, first move to the directory that contains the files you want to back up. From there, if you do not specify particular files you want to dump, IMOVE dumps all the files in your current directory, including subordinate directories and their files. Alternatively, you can name on your command line the files you want to dump; you can dump the files named in an indirect file (created with the CLI BUILD command); or you can use IMOVE's date/local-switch combinations. The latter give you the option of dumping only files last modified or opened either before or after a given date.

Whichever files you decide to back up, IMOVE will dump to either an unlabeled diskette, an unlabeled tape, labeled diskettes, labeled tapes, or MMOVE-format diskettes. Dumps to labeled diskettes, labeled tapes, and MMOVE-format diskettes can span several volumes. However, if your backup medium is an unlabeled diskette or tape, your IMOVE dump file is restricted to the capacity of one diskette or tape.

A final advantage of using IMOVE is that it writes its dumps in formats that AOS and AOS/VS can read. You can use the AOS or AOS/VS LOAD command to read an IMOVE dump file written to labeled diskettes or tapes, or an unlabeled diskette or tape. And you can use the AOS utility MMOVE to load an IMOVE dump file recorded on MMOVE-format diskettes. You can also transfer files in the opposite direction by using the DUMP command or the AOS utility MMOVE to dump files from your AOS or AOS/VS system and IMOVE to load them onto your DG/RDOS system.

IMOVE's Syntax

You invoke IMOVE from the CLI with the command IMOVE, using switches and arguments to define the exact operation you wish to perform. The following sections cover the format of the command line and all of IMOVE's switches.

The IMOVE Command Line

The format of the IMOVE command line is

IMOVE device-name [filename ...]

The device-name argument specifies the name of the diskette or magnetic tape drive to or from which you want to move files, for example DJ0 or MT0. If you are using an unlabeled magnetic tape, you must also include a tape file specifier (for example, MT0:0).

You use the optional filename argument to specify a file you wish to dump or load. You can name multiple files here, and, if you are dumping files, you can also name files outside the current directory by placing a directory specifier in front of the filename (directory-name:filename).

NOTE IMOVE does not allow you to include a pathname on your command line when you are loading files.

You cannot use filename templates to specify groups of files; however, you can fill the filename slot with the name of an indirect file that was built with filename templates. (For details, see the section below titled "Selecting Files to Move.") To invoke an indirect file as the IMOVE filename argument, use the command format IMOVE device-name @indirect-file@.

If you do not specify filenames on your command line, IMOVE behaves as follows (unless you use switches that tell it to do otherwise). In a backup operation, the utility dumps all of the files in the current directory, including subordinate directories and their files. In a restore operation, IMOVE loads the files that were in the current directory when you created your backup into the present current directory and the files that were in subordinate directories into directories subordinate to the present current directory. One exception to this rule can occur when a file on the backup media has the same name as a file already in the current or a subordinate directory. Such a file is not restored unless you use IMOVE's global /O or /R switch.

IMOVE's Switches

Tables 14-1 and 14-2 briefly describe IMOVE's global and local switches. For more detailed information about how to employ these switches, see the sections in this chapter on selecting files to move, using unlabeled, labeled, and MMOVE-format media, and transferring files between systems.

Table 14-1 IMOVE's Global Switches

Switch	Description
/A	Enables IMOVE to use labeled media. For labeled tape operations, use the global /T switch in conjunction with /A.
/C	Allows you to convert line termination characters in text files so that you can transfer files between DG/RDOS and AOS or AOS/VS. When you are dumping files, /C converts CRs to NLs; when you are loading files, it does the opposite. (There is also a local /C switch that converts the line termination characters in a single file.) NOTE: Do not use the global /C switch if you are dumping or loading binary files.
/F	Loads files from diskettes or tapes. The dump file may have been created with IMOVE, with the AOS or AOS/VS DUMP command, or with the AOS utility MMOVE.
/H	Displays help text about the IMOVE command line and switches. /H prevents any file transfer regardless of what other switches you use on your command line.
/L	Lists on your printer (filename $LPT) the names of all files moved.
/N	Prevents IMOVE from performing the operation that the rest of your command defines, but does list on your screen the files that the operation would have moved if the /N switch had not been present.
/O	Overrides IMOVE's standard practice of not restoring a file from a backup diskette or tape if a file of the same name already resides in the directory into which IMOVE is trying to load the backup file. If you use the /O switch, the file you are trying to restore overwrites the existing file. Do not use /O and /R in the same command line.
/R	Conditionally overrides IMOVE's standard practice of not restoring a file from a backup diskette or tape if a file of the same name already resides in the directory into which IMOVE is trying to load the backup file. If the file on your backup media is of more recent date than the file on your disk, IMOVE overwrites the existing file. Do not use /R and /O in the same command line.

(continued)

Table 14-1 IMOVE's Global Switches

Switch	Description
/T	Dumps files to or loads files from tape. If you do not use /T, the utility assumes that you are using diskettes.
/U	Enables IMOVE to use unlabeled media. This switch is optional for unlabeled tape.
/V	Displays on your screen the names of files as they are moved.

(concluded)

Table 14-2 IMOVE's Local Switches

Argument/Switch	Description
date/A/M	Dumps or loads only files modified on or after the date you specify. This date must be in the form dd-mmm-yy[:hh:mm:ss]; for example, `12-JAN-89:12:00:00/A/M` is a valid argument/switch combination. Any parts of the time you do not enter default to zero.
date/A/R	Dumps or loads only files last opened on or after the date you specify.
date/B/M	Dumps or loads only files last modified before the date you enter.
date/B/R	Dumps or loads only files last opened before the date you specify.
filename/C	During a dump operation, converts carriage returns to New Lines in *filename* only. During a load operation, converts New Lines to carriage returns in the file you name. If you use the global /C switch, all files undergo conversion regardless of how you use the local /C switch. NOTE: Do not append the local /C switch to the name of a binary file.

(continued)

Table 14-2 IMOVE's Local Switches

Argument/Switch	Description
filename/F	Allows you to specify a filename of from 1 to 17 characters to be included in the HDR1 record of the labeled diskette or tape you are dumping to. This switch also lets you specify what filename must be in a diskette or tape's HDR1 label before a load operation can take place. Use this switch only in conjunction with the global /A switch.
filename/L	Lists in *filename* the files dumped or loaded. The local /L switch overrides the global /L and /V switches.
buffer-size/S	Lets you determine the size of the buffers IMOVE will use for a dump or load operation. The legal buffer sizes are 2048, 4096, 6144, 8192, 10240, 12288, 14336, and 16384. By default the utility uses 8-Kbyte buffers. You may need to experiment to find out which buffer size works best when you are dumping data to, or loading data from, a reel-to-reel tape. The performance of some cartridge tape drives is improved greatly if you request large buffers; see Appendix A for recommendations.
days/X	Allows you to state the number of days, from the current date, for which data on a labeled diskette or tape should be protected from overwriting. (Note that your data is protected only from being overwritten during a subsequent IMOVE dump to labeled media.) If you do not use this switch, such protection lasts for 90 days. Use this switch only in conjunction with the global /A switch.

(concluded)

NOTE: With the exception of the local switch /C, which you may append to the name of a file you are dumping or loading, all local switches and their corresponding arguments must precede your device-name argument.

Selecting Files to Move

When indicating to IMOVE which files to dump or load, you have several options.

- You can leave the filename argument off your command line, in which case you will either dump all the files and directories in your current directory or load into your current directory all the files on your backup media. (The exceptions to this rule are noted below.)
- You can specify on your command line the names of the files and directories you wish to dump or load.
- You can substitute for the filename argument the name of an indirect file that contains the names of files and directories you wish to dump or load.
- You can use one of IMOVE's date/local-switch combinations in conjunction with one of the options mentioned above.

Each of these methods is discussed in more detail below.

Omitting the Filename Argument

If you perform a dump operation without specifying any file or directory names on your command line, you dump the entire contents of the current directory. For instance, assume that you are in a directory WEEKLYRPT.DR that contains the following files:

```
R
LIST/S<NL>
JUL0589.              11762  D
JUN0789.               9658  D
JUN1489.              13886  D
JUN2189.              10643  D
JUN2889.              11904  D
MAY3189.              12771  D
R
```

You can use the ensuing command to dump all of these files to a diskette.

```
Dx0:IMOVE/V Dxx<NL>
Please insert disk 1, then press NEW LINE to continue the DUMP.<NL>
17-OCT-89 15:49:45
  MAY3189
  JUN1489
  JUN2189
  JUL0589
  JUN2889
  JUN0789
R
```

If you were in the master directory, you could use the same command to dump the entire contents of your disk.

Similarly, when you ask IMOVE to load files from your backup media and do not supply a filename argument, the utility restores all the files in the dump. Thus, if WEEKLYRPT.DR were accidentally deleted, you could recreate the directory, make it your current directory, insert your backup diskette in Dxx, and restore your files by issuing the following command:

```
R
Dx0:IMOVE/F/V Dxx<NL>
Please insert disk 1, then press NEW LINE to continue the LOAD.<NL>
17-OCT-89 15:49:45
  MAY3189
  JUN1489
  JUN2189
  JUL0589
  JUN2889
  JUN0789
R
```

Specifying File and Directory Names

Unless you want to back up your entire disk or restore the entire contents of a backup set, it is often most convenient to name on your command line the files or directories you want to dump or load. As an example, assume that your master directory, DE0, contains the directories WEEKLYRPT.DR and CONTRACTS.DR and that you want to dump to tape the contents of both. One way to approach this task would be to make WEEKLYRPT.DR your current directory and to dump its contents using the command `DE0:IMOVE/T/V MT0:0`; then, you could make CONTRACTS.DR the current directory and issue a similar command. A better way to do this job, however, would be to move to directory DE0 and type this command:

```
R
IMOVE/T/V MT0:0 WEEKLYRPT.DR CONTRACTS.DR<NL>
18-OCT-89 14:23:53
  WEEKLYRPT.DR
    WEEKLYRPT:MAY3189
    WEEKLYRPT:JUN1489
    WEEKLYRPT:JUN2189
    WEEKLYRPT:JUL0589
    WEEKLYRPT:JUN2889
    WEEKLYRPT:JUN0789
```

```
  CONTRACTS.DR
    CONTRACTS:MASONB
    CONTRACTS:MAYBERRYL
    CONTRACTS:MOROCZL
    CONTRACTS:POAGC
    CONTRACTS:WASERD
    CONTRACTS:REDFORDS
    CONTRACTS:DRAKEC
R
```

Moreover, if you want to dump only some of the files in a directory (or load some of the files from your backup media), you have to specify the names of those files. For instance, if you wanted to dump only the files MAY3189 and JUL0589 from the directory WEEKLYRPT, you could move to that directory and then use this command:

```
R
DEO:IMOVE/T/V MT0:0 MAY3189 JUL0589<NL>
18-OCT-89 14:53:36
  MAY3189
  JUL0589
R
```

Using Indirect Files

Instead of supplying file or directory names on your IMOVE command line, you can, if you wish, supply the name of an indirect file that contains the names of the files or directories you wish to dump or load. The primary advantage of using an indirect file is that you can use template characters as you build that file. IMOVE, on the other hand, does not accept template characters.

Here's an example of how you might use an indirect file with IMOVE. Suppose that you had several years worth of monthly reports with filenames in the form mmm.yy and that they were located in a directory called MONTHLYRPT.DR. As a preliminary step in backing up your reports from 1988, you could build an indirect file (ALL88) containing the filenames of the reports from that year.

```
R
DIR MONTHLYRPT<NL>
R
BUILD ALL88 -.88<NL>
R
TYPE ALL88<NL>
APR.88,OCT.88,SEP.88,NOV.88,JUN.88,MAY.88,JUL.88,FEB.88,MAR.88,^
DEC.88,JAN.88,AUG.88^
R
```

Then, to dump those selected files to a diskette, you could employ this command line:

```
R
DX0:IMOVE/V Dxx @ALL88@<NL>
Please insert disk 1, then press NEW LINE to continue the DUMP.<NL>
18-OCT-89 15:54:58
  APR.88
  OCT.88
  SEP.88
  ...
R
```

You can also use indirect files and the names of individual files on the same command line. Thus, if you wanted to back up your monthly reports for the period November 1988 to December 1989, you could type

```
DX0:IMOVE/V Dxx @ALL89@ NOV.88 DEC.88<NL>
```

Using IMOVE's Date Switches

IMOVE has four local switches that, when used in conjunction with one of the file-selection methods mentioned above, can make performing incremental backups relatively easy. These switches allow you to move only files last opened before or after a given date (date/B/R and date/A/R) or only files last modified before or after a given date (date/B/M and date/A/M). Simply place the switch you need in front of the device-name argument on your command line.

As an example, let's say that the files in directory CONTRACTS.DR have the following last-modified dates:

```
R
DIR CONTRACTS<NL>
R
LIST/C<NL>
MASONB.          4655  D       10/18/89 10:20
MAYBERRYL.       8502  D       10/18/89 15:39
MOROCZL          9604  D       10/19/89 12:15
POAGC            5756  D       10/19/89 14:44
WASERD.         12878  D       10/25/89 11:23
REDFORDS.        6548  D       10/25/89 13:27
DRAKEC.          7704  D       10/25/89 13:30
R
```

To back up only the files in this directory that were last modified on or after October 25, 1989, you could use the local switch 25-OCT-89/A/M.

```
R
Dx0:IMOVE/T/V 25-OCT-89/A/M xT0:0<NL>
28-OCT-89 16:02:09
  WASERD
  REDFORDS
  DRAKEC
R
```

If CONTRACTS.DR were a subdirectory in DE0, you could also dump these newly revised contracts by moving to directory DE0 and issuing this command:

```
IMOVE/T/V 25-OCT-89/A/M MT0:0 CONTRACTS.DR<NL>
```

Using Unlabeled Diskettes and Tapes

If you do not use the global /A switch on your command line and you use either the /U (if you have a diskette) or /T (if you have a tape) switch, IMOVE assumes that you want to move files to or from a single-volume backup medium. You should allow the utility to make this assumption in the following cases:

- When you want to restore data from an unlabeled medium
- When you plan to create several IMOVE dump files on one tape
- When your dump files will fit on one diskette or tape

Unlabeled Diskettes

There are several pieces of general information about dumping data to an unlabeled diskette that you should be aware of.

- The files you want to back up must fit on one diskette.
- The diskette you use for backup cannot have any bad blocks.
- The diskette you move files to must be hardware formatted. (Data General diskettes are hardware formatted when you buy them; if you want to use blank diskettes that did not come from Data General, refer to the documentation you received with your Customer Mode Diagnostics or SCP System Media for instructions on how to hardware format them.) However, the diskette does not need to be software formatted. If you insert a software formatted diskette during a dump operation, IMOVE writes to it, but also destroys that diskette's directory structure and any files the diskette contains.

Once you have an appropriately formatted diskette, dumping your files to that diskette is quite easy. Just position yourself in the proper directory, insert the diskette in its drive, and issue a command in this format:

IMOVE/U/V diskette-name [filename ...]

The /V switch tells IMOVE to list on your console the names of the files it dumps, and the /U switch indicates that the diskette will be unlabeled. Everything else is taken care of by default. The absence of the /F switch indicates that you want to dump, not restore, files. The absence of the /T switch indicates that you are dumping to a diskette, not a tape. And the absence of the /A switch, as mentioned above, signals that you want a single-volume diskette.

After IMOVE reads this command, it prompts you to insert the diskette and press New Line to continue. Then the program begins dumping the files you requested it to move. After IMOVE is done, the utility returns you to the CLI.

At this point, you should verify the integrity of the dump file you just created. To do this, enter a command of the form IMOVE/F/N/U/V diskette-name.

To restore files and directories from an unlabeled diskette, you move to the directory into which you want to load the files, insert the diskette, and enter a command in this format:

IMOVE/F/U/V diskette-name [filename ...]

After IMOVE reads your command and you press New Line to continue, the program begins to load the files you requested that it move. After IMOVE is finished, the utility returns you to the CLI.

Unlabeled Tape

The most important thing to bear in mind about using an unlabeled tape is that the files you want to back up, when stored in dump format, must fit on one tape. If they do not, you receive an error message stating that IMOVE reached the end of your tape before completing its dump operation, and then the utility pops you back to the CLI. Your only recourses at this point are (1) to divide the files you want to back up and use two or more IMOVE commands to move your files to two or more unlabeled tapes, or (2) to use labeled diskettes or tapes, or MMOVE-format diskettes.

Also, be aware that you can use IMOVE to move files to or from a tape whether its drive is initialized or not; however, the status of the drive does affect how IMOVE works. If you initialize your tape drive before invoking the utility, IMOVE dumps or loads data, leaves the tape at the end of the dump file, and does not release the device. If the drive is not initialized, IMOVE initializes it, dumps or loads data, and releases the drive; thus, the tape rewinds automatically.

To begin dumping files to an unlabeled tape, first move to the directory that contains those files, mount your tape, and then issue a command in this format:

IMOVE/T/V [buffer-size/S] tape-name:file-number [filename ...]

The *buffer-size*/*S* argument/local-switch combination tells the utility what size buffers to use in place of the default 8-Kbyte buffers. See Appendix A for recommendations.

The /T switch tells IMOVE that you are moving your files to magnetic tape, and the absence of the /A switch indicates that you do not want that tape to include labels.

A :0 on the device specifier signals that you want to dump your files at the beginning of the tape, in file 0. (If there are no files currently on the tape, you must specify a dump to file 0.) To stack later dumps on the tape, you can specify file 1, 2, 3, and so on. Be sure to keep close track of your stacked dumps, however, because you must use the correct tape file number each time you use IMOVE to load material from unlabeled tape. Also, if you accidentally overwrite an existing dump file, you erase not only that file, but all subsequent tape files.

In any case, once IMOVE reads your command line, it prints the date and time and then begins dumping the appropriate files. If all these files fit on one tape, the utility simply returns you to the CLI, at which point you should release your tape drive if necessary. If the files you named on your command line do not fit on your tape, IMOVE issues the following message as it aborts:

```
File space exhausted: xTx:0
R
```

To create a multitape backup set, you must use labeled tapes (discussed below).

At this point, you should verify the integrity of the dump file you just created, especially if you are working with a tape drive that does not have read-after-write heads. To do this, enter a command of the form IMOVE/F/N/T/V [buffer-size/S] tape-name:file-number.

For a restore operation, move to the directory into which you want to load the material on your backup tape, insert your tape, and enter a command in this format:

IMOVE/F/T/V [buffer-size/S] tape-name:file-number [filename ...]

After reading this command, the utility loads the files you asked it to move and sends you back to the CLI.

Using Labeled Diskettes and Tapes

If you use the global /A switch on your command line, IMOVE understands that you want to write AOS/VS-compatible labels on your backup media. Some of the advantages of writing such labels are that

- You can use labels to store important information about your dump file, such as the date on which it was created
- Labels contain expiration dates that prevent you or anyone else from inadvertently overwriting your dump file while performing a dump to labeled diskettes or tapes
- You can transfer IMOVE-format material on labeled media to an AOS/VS system
- When you are dumping data to magnetic tape, labels make it possible for you to create a multivolume backup set

Labeled Diskettes

Before you begin dumping files to labeled diskettes, be sure that you have on hand a sufficient number of diskettes that meet the following criteria:

1. All your diskettes must be free of bad blocks.
2. The diskettes you use must be hardware formatted. (Data General diskettes are hardware formatted when you buy them; if you want to use blank diskettes that did not come from Data General, refer to the documentation you received with your Customer Mode Diagnostics or SCP System Media for instructions on how to hardware format them.) The diskettes do not, however, need to be software formatted. If you insert a software formatted diskette during a dump operation, IMOVE writes to it, but also destroys that diskette's directory structure and any files the diskette contains.

Your backup diskettes may or may not be prelabeled. (For instructions on how to prelabel diskettes using the LABEL utility, see the next chapter.) If they are not labeled, IMOVE writes a volume label (VOL1), a header label (HDR1), and an ERMAP label on each diskette. If they do have existing labels, IMOVE preserves each diskette's volume ID and then writes new HDR1 and ERMAP labels.

To begin backing up files to labeled diskettes, first position yourself in the proper directory, and then issue a command in this format:

```
IMOVE/A/V [filename/F] [days/X] diskette-name [filename ...]
```

The /A switch indicates that you want to write AOS/VS-format labels on your backup diskettes, and the optional argument/local-switch combinations allow you to place information in the HDR1 labels of those diskettes. The local /F switch enables you to enter a filename of from 1 to 17 characters in each diskette's header label; if you do not specify such a filename on your command line, IMOVE prompts you for one later. The local /X switch lets you enter the number of days, from the current date, for which you want to protect your data from being overwritten. (Note that your data is protected only from being overwritten during a subsequent IMOVE dump to labeled diskettes.) If you do not use this switch, such protection lasts for 90 days.

Once you have entered your command line, the utility prompts you to insert your first backup diskette and to press New Line. Then, if the diskette you inserted does not already have a volume label, IMOVE asks you for a volume name.

```
VOL1 does not exist.
Enter Volume ID for volume label VOL1 (1-6 char):
```

If your diskette already has a VOL1 label, IMOVE skips this request.

Next, if you did not use the local /F switch on your command line, the utility asks you to name your dump file.

```
Filename not on command line.
Enter Filename for header label HDR1 (1-17 char):
```

Once you supply this filename, IMOVE has all the information it needs to write the necessary labels on your diskette, so it begins to dump the files you want to back up. If all of these files fit on one diskette, the utility simply returns you to the CLI. If the files do not fit on a single diskette, IMOVE requests that you insert a second backup diskette.

```
This diskette has been filled: Dxx
Please insert labeled disk 2, then press NEW LINE to continue the
DUMP.
```

If the diskette you insert does not have a VOL1 label, IMOVE again prompts you for a volume ID. Then, after you have supplied this identifier, IMOVE resumes dumping files.

The program continues to request diskettes in this manner until it has completed the dump operation. Then you are returned to the CLI.

At this point, you should verify the integrity of the dump file you just created. To do this, enter a command of the form IMOVE/A/F/N/V diskette-name and then insert your backup diskettes as you are prompted to do so.

To restore files from labeled diskettes, start by moving to the directory into which you want to load the files you have backed up and entering a command in this format:

IMOVE/A/F/V [filename/F] diskette-name [filename ...]

During a restore operation, the *filename/F* option lets you specify a filename that must appear in your diskette's HDR1 label in order for IMOVE to work. If you do not use this argument/local-switch combination, the utility simply ensures that all the diskettes in a multivolume set have the same filename in their HDR1 labels.

Once IMOVE has read your command line, and you have inserted your first (or only) backup diskette and typed New Line to continue, the program starts loading the files you asked it to move. If these files are part of a single-diskette dump, IMOVE loads the files and returns you to the CLI. If they are part of a multidiskette set, IMOVE loads the appropriate files from the first diskette in the set and then prompts you for the second.

```
This diskette has been exhausted: Dxx
Please insert labeled disk 2, then press NEW LINE to continue the
LOAD.
```

The program continues to request diskettes in this manner until it has read the entire backup set.

Labeled Tape

As mentioned earlier, using labeled tape as your backup medium allows you to create multivolume backup sets, which are not possible with unlabeled tape. Other important differences between unlabeled and labeled tape operations include the following:

- Whereas many dump files can reside on a single unlabeled tape (MT0:0, MT0:1, and so on), a labeled tape or tape set can hold only one IMOVE-format file.
- While IMOVE rewinds an unlabeled tape only in cases where you do not initialize your tape drive before issuing a dump or load command, the program always rewinds a labeled tape.

As you begin a dump operation where labeled tape is the backup medium, you will confront one of two situations: your tape will either have existing labels or it won't. If your tape does not have labels, IMOVE writes a volume label (VOL1), two header labels (HDR1 and HDR2), and either an end-of-volume label (EOV1) or end-of-file labels (EOF1 and EOF2)—an EOV1 label appears on all tapes except the last in a multitape set. If your tape does have existing labels, IMOVE preserves the tape's VOL1 label and any user volume labels (UVL1-UVL9) and then writes HDR1 and HDR2 labels and either an EOV1 label or EOF1 and EOF2 labels.

NOTE: To write UVLs, you must use the LABEL program described in the next chapter.

After positioning yourself in the proper directory and inserting your tape, you can instruct IMOVE to write these labels and to dump the files you want to back up by issuing a command in this format:

IMOVE/A/T/V [buffer-size/S] [filename/F] [days/X] tape-name [filename ...]

The *buffer-size*/*S* argument/local-switch combination tells the utility what size buffers to use in place of the default 8-Kbyte buffers. See Appendix A for recommendations.

The /A switch indicates that you want to write AOS/VS-format labels on your backup tapes, and the optional local /F and /X switches permit you to place information in your label files. The local /F switch enables you to enter a filename of from 1 to 17 characters in your tape's HDR1 label; if you do not specify such a filename on your command line, IMOVE prompts you for one later. The /X switch lets you enter the number of days, from the current date, for which you want to protect your data from being overwritten. (Note that your data is protected only from being overwritten during a subsequent IMOVE dump to labeled tapes.) If you do not use this switch, such protection lasts 90 days. There is no tape file specifier in this command line (`:0`, for example) because a labeled tape set can accommodate only one dump file.

Once you have entered such a command line, IMOVE asks you to mount a tape and to press New Line. Then, if the tape you are using does not already have a VOL1 label, the program prompts you for a volume name.

```
VOL1 does not exist.
Enter Volume ID for volume label VOL1 (1-6 char):
```

If your tape has a VOL1 label, IMOVE skips this request.

NOTE: IMOVE may abort if it cannot determine whether your tape has existing labels. Either initialize your tape with the command INIT/F or write labels to it using the LABEL program.

Next, if you did not use the local /F switch on your command line, the utility asks you to name this dump file.

```
Filename not on command line.
Enter Filename for header label HDR1 (1-17 char):
```

Once you supply a filename, IMOVE has all the information it needs to write its tape labels, so it begins to perform the operation you requested. If all the files you asked the program to dump fit on one tape, the utility simply rewinds your tape and returns you to the CLI. If those files do not fit on a single tape, IMOVE prompts you to insert the next tape in the set.

```
Please mount labeled tape 2, then press NEW LINE to continue the
DUMP.
```

The program continues to request tapes until it has completed the dump operation. Then it returns you to the CLI.

At this point, you should verify the integrity of the dump file you just created, especially if you are working with a tape drive that does not have read-after-write heads. To do this, enter a command of the form IMOVE/A/F/N/T/V [buffer-size/S] tape-name and then insert your backup tapes when you are prompted to do so.

To restore files from labeled tapes, move to the directory into which you want to load the files you have backed up, mount the first tape in the backup set, and enter a command in this format:

IMOVE/A/F/T/V [buffer-size/S] [filename/F] tape-name [filename ...]

During a restore operation, the *filename/F* option lets you specify a filename that must appear in your tape's HDR1 label for IMOVE to work. In the absence of this argument/local-switch combination, the utility simply ensures that all tapes in a multivolume set have the same filename in their HDR1 labels.

After IMOVE reads your command line and you type New Line to continue, the program starts loading the files you asked it to move. If these files are part of a single-tape dump, IMOVE loads the files, rewinds your tape, and returns you to the CLI. If they are part of a multitape set, IMOVE loads the appropriate files from the first tape in the set and then prompts you for the second.

```
Please mount labeled tape 2, then press NEW LINE to continue the
LOAD.
```

The program continues to request tapes in this manner until it has read the entire backup set.

Using MMOVE-Format Diskettes

If you do not use the /A, /T, or /U switches on your command line, IMOVE assumes that you want to move files to or from diskettes in MMOVE-format. With MMOVE-format, you can create a multivolume backup.

There are several pieces of general information about dumping data to MMOVE-format diskettes that you should be aware of.

- Because IMOVE writes a sequence number on each diskette it writes to, the utility does provide for multiple-diskette dumps. Therefore, be sure that you record on a paper label each diskette's number; when you restore the files in your backup set, you must insert the diskettes that make up the set in the same order in which you inserted them during the dump operation.
- The diskettes you use for backup cannot have any bad blocks.

- The diskettes you move files to must be hardware formatted. (Data General diskettes are hardware formatted when you buy them; if you want to use blank diskettes that did not come from Data General, refer to the documentation you received with your Customer Mode Diagnostics or SCP System Media for instructions on how to hardware format them.) However, the diskettes do not need to be software formatted. If you insert a software formatted diskette during a dump operation, IMOVE writes to it, but also destroys that diskette's directory structure and any files the diskette contains.

Once you have on hand an adequate number of appropriately formatted diskettes, dumping your files to those diskettes in MMOVE format is quite easy. Just position yourself in the proper directory, insert a diskette in its drive, and issue a command in this format:

Dx0:IMOVE/V diskette-name [filename ...]

The /V switch tells IMOVE to list on your console the names of the files it dumps. Everything else is taken care of by default. The absence of the /F switch indicates that you want to dump, not restore, files. The absence of the /T switch indicates that you are dumping to diskettes, not tapes. The absence of the /A and /U switches, as mentioned above, signals that you want MMOVE-format diskettes.

After IMOVE reads this command, it prompts you to insert a diskette and to press New Line to continue. Then the program begins dumping the files you requested it to move. If all of these files fit on one diskette, IMOVE returns you to the CLI; otherwise, you receive this message:

```
This disk has been filled: Dxx
Please insert disk 2, then press NEW LINE to continue the DUMP.
```

IMOVE continues to request more diskettes until it has completed the dump operation. Then the CLI is swapped back into execution.

At this point, you should verify the integrity of the dump file you just created. To do this, enter a command of the form IMOVE/F/N/V diskette-name and then insert your backup diskettes as you are prompted to do so.

To restore files and directories from MMOVE-format diskettes, you move to the directory into which you want to load the files, insert the first (or only) diskette in the backup set, and enter a command in this format:

IMOVE/F/V diskette-name [filename ...]

After IMOVE reads your command and you type New Line to continue, the program begins to load the files you requested that it move. If these files are part of a single-volume dump, the utility simply loads the files and then returns you to the CLI. If they are part of a multivolume backup set, IMOVE loads the appropriate files from the first diskette and then prompts you to insert the next one in the series.

```
This disk has been exhausted: Dxx
Please insert disk 2, then press NEW LINE to continue the LOAD.
```

The utility continues to ask for new diskettes until it has read the entire backup series.

Transferring Files Between Operating Systems

As mentioned earlier, one advantage of using IMOVE is that the utility allows you to transfer files between DG/RDOS and AOS or AOS/VS systems. The table below summarizes which AOS and AOS/VS commands and utilities you can use in conjunction with IMOVE, which systems these commands and utilities work on, and which types of media you can use with each command (if you are transferring files between systems).

Table 14-3 File Transfer Using IMOVE

Commands	Operating Systems	Media Types
DUMP/LOAD	AOS, AOS/VS	labeled diskettes, labeled tapes, unlabeled diskette, unlabeled tape
MMOVE	AOS	MMOVE-format diskettes

For further information on how to transfer files between the two operating systems, consult the examples below.

NOTE: If you transfer a contiguous file to an AOS/VS system and then retrieve that file, the file's size may increase. The file you get back from the AOS/VS system will occupy a number of blocks that is a multiple of the file-element size under AOS/VS (usually 4).

Transferring Files on Unlabeled Diskette

The first dialog below shows how you could move files from a DG/RDOS to an AOS or AOS/VS system, and the second demonstrates how to move files in the opposite direction.

DG/RDOS to AOS or AOS/VS

At the DG/RDOS machine:

```
R
IMOVE/U/V DJ0 SAVEFILE TEXTFILE/C<NL>
Please insert disk 1, then press NEW LINE . . . <NL>
 10-JUL-89 10:01:26
  SAVEFILE
  TEXTFILE
R
```

At the AOS or AOS/VS machine:

```
) LOAD/V @DPM0<NL>
SAVEFILE
TEXTFILE
)
```

AOS or AOS/VS to DG/RDOS

At the AOS or AOS/VS machine:

```
) DUMP/V @DPM0 SAVEFILE TEXTFILE<NL>
SAVEFILE
TEXTFILE
)
```

At the DG/RDOS machine:

```
R
IMOVE/F/U/V DJ0 SAVEFILE TEXTFILE/C<NL>
Please insert disk 1, then press NEW LINE . . . <NL>
10-JUL-89 10:14:26
  SAVEFILE
  TEXTFILE
R
```

Transferring Files on Unlabeled Tape

The first dialog below shows how you could move files from a DG/RDOS to an AOS or AOS/VS system, and the second demonstrates how to move files in the opposite direction.

DG/RDOS to AOS or AOS/VS

At the DG/RDOS machine:

```
R
IMOVE/T/V MT0:0 SAVEFILE TEXTFILE/C<NL>
09-JUL-89 10:03:00
  SAVEFILE
  TEXTFILE
R
```

At the AOS or AOS/VS machine:

```
) LOAD/BUFFERSIZE=8192/V @MTA0:0<NL>
9-JUL-89 10:03:00
 SAVEFILE
 TEXTFILE
)
```

AOS or AOS/VS to DG/RDOS

At the AOS or AOS/VS machine:

```
) DUMP/BUFFERSIZE=8192/V @MTA0:0 SAVEFILE TEXTFILE<NL>
9-JUL-89 10:26:21
 SAVEFILE
 TEXTFILE
)
```

At the DG/RDOS machine:

```
R
IMOVE/F/T/V MT0:0 SAVEFILE TEXTFILE/C<NL>
09-JUL-89 10:26:21
  SAVEFILE
  TEXTFILE
R
```

Transferring Files on Labeled Diskettes

The first dialog below shows how you could move files from a DG/RDOS to an AOS or AOS/VS system, and the second demonstrates how to move files in the opposite direction.

DG/RDOS to AOS or AOS/VS

At the DG/RDOS machine:

```
R
IMOVE/A/V FILENAME/F DA10 SAVEFILE TEXTFILE/C<NL>
Please insert labeled disk 1, then press NEW LINE . . . <NL>
VOL1 does not exist.
Enter Volume ID for volume label VOL1 (1-6 char): VOL1<NL>
09-JUL-89 13:41:03
  SAVEFILE
  TEXTFILE
R
```

At the AOS or AOS/VS machine:

```
) SUPERUSER ON<NL>

*)  OPERATOR ON<NL>

*)  ACL @DPJ10 +,WARE<NL>

*)  LOAD/V @LFD:VOL1:FILENAME<NL>

Please insert a diskette if not already inserted.
Unit [@DPJ10] Volume ID [VOL1] ? [y] <NL>
 9-JUL-89 13:41:03
  SAVEFILE
  TEXTFILE

Please remove the diskette.
)
```

AOS or AOS/VS to DG/RDOS

At the AOS or AOS/VS machine:

```
) SUPERUSER ON<NL>

*)  OPERATOR ON<NL>
```

```
*)  ACL @DPJ10 +,WARE

*)  DUMP/V @LFD:VOL1:FILENAME SAVEFILE TEXTFILE<NL>

Please insert a diskette if not already inserted.
Unit [@DPJ10] Volume ID [VOL1] ? [y] <NL>
 9-JUL-89 14:26:37
  SAVEFILE
  TEXTFILE

Please remove the diskette.
)
```

At the DG/RDOS machine:

```
R
IMOVE/A/F/V FILENAME/F DA10 SAVEFILE TEXTFILE/C<NL>
Please insert labeled disk 1, then press NEW LINE . . . <NL>
09-JUL-89 14:26:37
  SAVEFILE
  TEXTFILE
R
```

Transferring Files on Labeled Tapes

The first dialog below shows how you could move files from a DG/RDOS to an AOS or AOS/VS system, and the second demonstrates how to move files in the opposite direction.

DG/RDOS to AOS or AOS/VS

At the DG/RDOS machine:

```
R
IMOVE/A/T/V FILENAME/F MT0 SAVEFILE TEXTFILE/C<NL>
Please mount labeled tape 1, then press NEW LINE . . . <NL>
VOL1 does not exist.
Enter Volume ID for volume label VOL1 (1-6 char): VOL1<NL>
09-JUL-89 16:22:44
  SAVEFILE
  TEXTFILE
R
```

From the AOS or AOS/VS side:

```
(At the OP console, type CONTROL @EXEC OPERATOR ON<NL>.)

 ) LOAD/BUFFERSIZE=8192/V @LMT:VOL1:FILENAME<NL>

(At the OP console, type CONTROL @EXEC MOUNTED @MTA0<NL>.)

  9-JUL-89 16:22:44
  SAVEFILE
  TEXTFILE
)

(At the OP console, type CONTROL @EXEC DISMOUNTED<NL>.)
```

AOS or AOS/VS to DG/RDOS

From the AOS or AOS/VS side:

```
(At the OP console, type CONTROL @EXEC OPERATOR ON<NL>.)

) XEQ LABEL @MTA0 VOL1<NL>

) DUMP/BUFFERSIZE=8192/V @LMT:VOL1:FILENAME SAVEFILE TEXTFILE<NL>

(At the OP console, type CONTROL @EXEC MOUNTED @MTA0<NL>.)

9-JUL-89 16:45:10
  SAVEFILE
  TEXTFILE
)

(At the OP console, type CONTROL @EXEC DISMOUNTED<NL>.)
```

At the DG/RDOS machine:

```
R
IMOVE/A/F/T/V FILENAME/F MT0 SAVEFILE TEXTFILE/C<NL>
Please mount labeled tape 1, then press NEW LINE . . . <NL>
09-JUL-89 16:45:10
  SAVEFILE
  TEXTFILE
R
```

Transferring Files on MMOVE-Format Diskettes

The first dialog below shows how you could move files from a DG/RDOS to an AOS system, and the second demonstrates how to move files in the opposite direction.

DG/RDOS to AOS

At the DG/RDOS machine:

```
R
IMOVE/V DJ0 SAVEFILE TEXTFILE/C<NL>
Please insert disk 1, then press NEW LINE . . . <NL>
 10-JUL-89 10:01:26
  SAVEFILE
  TEXTFILE
R
```

At the AOS machine:

```
) XEQ MMOVE/DUMP/F/V @DPM0<NL>
SAVEFILE
TEXTFILE
)
```

AOS to DG/RDOS

At the AOS machine:

```
) XEQ MMOVE/DUMP/V @DPM0 SAVEFILE TEXTFILE<NL>
SAVEFILE
TEXTFILE
)
```

At the DG/RDOS machine:

```
R
IMOVE/F/V DJ0 SAVEFILE TEXTFILE/C<NL>
Please insert disk 1, then press NEW LINE . . . <NL>
10-JUL-89 10:14:26
  SAVEFILE
  TEXTFILE
R
```

IMOVE's Error Messages

Below is a list of error messages you may receive from IMOVE. Each message is accompanied by an explanation of what problem has occurred and of how you should respond.

`Attempt to alter a permanent file`

If a file in your IMOVE backup set and a file in the current directory share the same name, you can generally get IMOVE to overwrite the disk file by using either the global /R or global /O switch on your command line. This strategy does not work, however, if the disk file has the attribute P.

`Buffer size must be a multiple of 2048`

If you use IMOVE's local /S switch on your command line, the argument to which you append that switch must be a multiple of 2048.

`Buffer size must be between 2048 and 16384`

When you use the argument/local-switch combination *buffer-size/S* on your command line, the numeric argument must be a value between 2048 and 16384.

`Conflicting local switch setting`

You may have used a date/local-switch combination on your command line and typed the date in an unacceptable format. Make sure the date is in the form dd-mmm-yy, for example, `31-MAY-89`.

`Device time out`

You probably entered a command to dump to or load from a diskette, but you do not have a diskette drive, forgot to insert your diskette, inserted it incorrectly, or left your drive door open. Also, a hardware error may have occurred.

`Directory depth exceeded`

You began your IMOVE restore operation from too low a level in your directory structure; consequently, IMOVE is trying to create a secondary partition within a secondary partition or a subdirectory within a subdirectory. First terminate the utility by typing Ctrl-C Ctrl-A. Then move up a level in your directory structure and try again to restore your files.

Directory in use

The diskette drive you named on your IMOVE command line was not released the last time it was used. Release the device, and then try performing your IMOVE operation again.

Disk format error

While trying to start a dump or load operation with diskettes, you probably included the global /T switch on your command line by mistake.

Duplicate local switch setting

No local switch, except the /C switch, can appear more than once on a command line.

File already exists

If a file in your IMOVE backup set and a file in the current directory share the same name, you cannot load the file in IMOVE format unless you use IMOVE's global /O or /R switch on your command line.

File data error

If you are working with a diskette, most likely that diskette is not hardware formatted; however, you could also be trying to write to a write-protected diskette. If you are attempting to read a tape, your tape is blank.

File does not exist

There are several reasons why you might receive this message. You could have misspelled a filename. You may have failed to include a file's filename extension. You may have forgotten to specify a tape file number while attempting to write to or read from an unlabeled tape. Or you could have attempted to access a tape drive on a system that does not include a driver for that device.

File in use

You cannot use IMOVE to overwrite a file that is open.

`File read protected`

You attempted to dump a read-protected file, that is, one with the attribute R. Use the CLI's CHATR command to remove this attribute and then try dumping the file again.

`File space exhausted`

If you are dumping files to an unlabeled tape, this message indicates that IMOVE found the tape's EOT marker or filled the diskette before it finished dumping your files. You might consider dumping the files to labeled media. If you are loading files, then IMOVE has discovered that the destination disk or a secondary partition on that disk is full.

`File write protected`

You tried to dump files to a write-protected tape.

`Filename too long.`

While using IMOVE to write labels on a diskette or tape, you requested a volume ID of more than 6 characters or a filename of more than 17 characters.

`Illegal directory name`

You may have mistyped the unit name of the diskette drive you are using; for example, you could have typed DJO (the letter O) instead of DJ0 (the digit 0). Also, if the system you are running does not include support for a diskette drive controller and you try to to write to or read from a diskette, you will receive this message. Another possibility is that you tried to move data to or from an unlabeled tape without using the global /T switch on your command line. Or while trying to write to or read from a labeled tape, you may have included on your command line a tape file specifier (UT0:0).

`Illegal file name`

This message could indicate a number of things. For example, you may have used a filename template when indicating to IMOVE which files to dump or load, and the utility does not recognize template characters. Or when typing your command line, you may have placed an argument/local-switch combination after your device-name argument; in this case, IMOVE interprets the argument/local-switch combination as a filename.

Another possibility is that you tried to load a file that was created on an AOS or AOS/VS system and has a filename that is not legal under RDOS (perhaps the name includes an underscore or is more than 10 characters long); if this is the case, you will be prompted to enter a new filename. Other possibilities are that you used the *filename/F* argument/local-switch combination on your command line, and the filename you supplied was longer than 17 characters, or you used the *filename/L* argument/local-switch combination and provided an invalid filename.

`Improper switch selection, type IMOVE/H for help`

Most likely you mistyped a global switch. The valid global switches are listed in Table 14-1.

`Incompatible dump revision`

The file you tried to load from tape was not in IMOVE format.

`Insufficient contiguous blocks`

You are attempting to load a large contiguous file from your IMOVE backup media, and there are not enough free contiguous blocks in your partition to hold the file.

`Insufficient memory to execute program`

There is not enough memory available in the ground in which you are trying to run IMOVE. You can use the CLI command SMEM to allocate more memory to that ground.

`Invalid argument value specified`

This error could occur if, for example, you used the argument/local-switch combination *days/X* on your command line and filled the *days* slot with something other than a number.

`Invalid date:`

Most likely you forgot to use hyphens to separate the day, month, and year in your date. Retype the date in the proper format, for example, `31-MAY-89`.

`Invalid dump block format`

The diskette or tape file that you are attempting to read was not written by IMOVE, or you may need to use a larger *buffer-size/S* to read this tape file.

`Invalid dump block type`

You may have tried to load data from a labeled tape using a command line appropriate for reading data from an unlabeled tape. If this is the case, you need to add the global /A switch to your command line and omit the tape file specifier after the tape drive unit name. Or, you may need to use a larger *buffer-size/S* to read this tape file.

`Invalid VOL1 label on labeled media.`

While IMOVE was trying to restore data from labeled media, you inserted media that was unlabeled or media that contained a VOL1 label that was either not in the format or not in the location that IMOVE expected.

`No such directory`

If you use a directory specifier to point to the files you want IMOVE to dump, the directory associated with that specifier must be initialized.

`Tape not ready`

You asked IMOVE to begin writing data to, or reading data from, a labeled tape, and either there is no tape in the drive you specified on your command line, or the tape is not at the BOT mark. After issuing this message, the utility does not terminate, but prompts you again to insert your tape.

`The Character Instruction Set is required`

You should never see this message. If you do, either you have a hardware problem or the CPU instruction microcode on your MV/Family system is corrupt.

`This disk contains more than 15 bad sectors`

The diskette you are trying to write to may not have been hardware formatted or it may contain bad blocks. Another possibility is that you have inserted a 96 TPI diskette in a 48 TPI drive.

`This disk is not in sequence`

During a restore operation, you must insert the diskettes that compose a backup set in the order in which you inserted them during the dump operation.

`This disk is not part of the dump being processed`

While restoring data from a multidiskette backup set, you inserted an IMOVE-format diskette that was not created at the same time as the actual diskettes in the set were.

`Unexpired media; expiration date:`

When you first dump files to a labeled tape, you set an expiration date that protects your data from being overwritten during another labeled-tape dump operation for the length of time you specify (90 days is the default). If, prior to the expiration date, you again ask IMOVE to write data to your labeled tape, you will receive this message and be asked if you want to overwrite the contents of the tape.

`Unit improperly selected`

Either you issued a command for IMOVE to write to or read from an unlabeled tape without first mounting your tape, or you forgot to include the /T switch on your command line.

`Unrecognizable local switch setting`

When you place a date or filename argument before your device-name argument, you must append one, and only one, of the following local switches to that argument: /A/M, /A/R, /B/M, /B/R, /F, or /L.

`Wrong filename:`

This error can occur if you try to load data from a labeled tape using a command line that includes the argument/local-switch combination *filename/F*. If the filename on your command line and the filename in the HDR1 label on your tape do not match, IMOVE does not read the tape.

-End of Chapter-

Chapter 15

LABEL: Reading and Writing Diskette and Tape Labels

Read this chapter when you want to

- Read or write AOS/VS- or IBM-format diskette labels, or read the diskette labels written by the BURST utility
- Examine or create AOS/VS- or IBM-format tape labels
- Initialize a labeled magnetic tape

The LABEL utility serves two primary purposes. First, it allows you to examine existing diskette or tape labels, and, second, it enables you to create new labels in preparation for an IMOVE, AOS DUMP, or AOS/VS DUMP backup session. If you plan to use BURST to back up your disk, there is no need to prelabel your diskettes or tapes because BURST overwrites existing labels.

If you plan to use IMOVE, it is not necessary to use LABEL to prepare your backup media; however, especially if your backup medium is tape, there are advantages to prelabeling each volume. For instance, while IMOVE writes only VOL1, HDR1, and HDR2 labels at the beginning of your tape, LABEL allows you to write up to nine user volume labels as well. Also, while IMOVE writes the same volume ID on all the tapes in a multivolume backup set, LABEL lets you give each tape a unique ID. This ability becomes important when you want to move files from a DG/RDOS to an AOS/VS system because AOS/VS requires that each tape in a backup set have a different volume ID.

The rest of this chapter presents LABEL's command line syntax and examples of the various uses to which you can put the utility. These subjects are treated in the following sections:

- LABEL's Syntax
- Using LABEL with Diskettes
- Using LABEL with Tapes
- LABEL's Error Messages

LABEL's Syntax

You invoke LABEL from the CLI by typing the command LABEL followed by the unit name of the diskette or tape you are working with:

```
LABEL device-name/D
```

You may also need to use one or more of the switches listed in Tables 15-1 and 15-2, depending on the exact operation you wish to perform. For more detailed information on how to employ these switches, see the sections ahead on reading and writing diskette labels, reading and writing tape labels, and initializing a labeled tape.

Table 15-1 LABEL's Global Switches

Switch	Description
/B	Reads diskette labels written by BURST.
/F	Examines existing labels.
/H	Displays a help message.
/I	Examines or writes labels in IBM format.
/L	Sends output to $LPT.
/P	Requests partial initialization of a tape. This switch has the same effect as /S.
/S	See /P.
/T	Reads from or writes to a tape. In the absence of this switch, the utility assumes that you are working with diskettes.
/V	Sends output to the console.

Table 15-2 LABEL's Local Switches

Argument/Switch	Description
device-name/D	Indicates the unit name of the diskette or tape you are working with. This argument/local-switch combination must be present on all LABEL command lines unless you are simply requesting a help message.
volid/V	Allows you to supply a volume ID of from 1 to 6 characters. To write new labels or partially initialize a tape, you must use this switch.
owner/O	Specifies an owner name to be included in a diskette or tape's VOL1 label. If you are writing an AOS/VS-format label, this name can be up to 14 characters long; if you are using the IBM format, it can be up to 10 characters long.
access/A	Lets you enter a 1-character access code in a diskette or tape's VOL1 label. CAUTION: Do not use an access code if you plan to move the files you are dumping to an AOS or AOS/VS system.
filename/L	Sends output to *filename*. This switch overrides the global /L switch.
uvltext/U	Creates a user volume label. As long as *uvltext* consists of 1 to 72 characters surrounded by double quotation marks, the utility puts this text in a label. (The semicolon and backslash are illegal in UVLs.) You can request a total of nine user volume labels.

Using LABEL with Diskettes

If you ever dump data to or restore data from labeled diskettes, you can use the LABEL utility to do two jobs. You can use it (1) to read the contents of existing labels, or (2) to write labels on previously unlabeled diskettes in preparation for a backup session. For specific instructions about how to perform these jobs, see the subsections below.

NOTE: Don't prelabel your diskettes if you plan to back up your disk using BURST because BURST overwrites any existing labels.

Examining Diskette Labels

To read the existing labels on a diskette, insert your labeled diskette in its drive and then issue a command in this format:

LABEL/F[/B][/I] device-name/D

The absence of the /T switch indicates that you are working with diskette, not tape, labels, and the /F switch signals that you want to read those labels. You may or may not need the /B and /I switches. You should use the /B switch only if you want to read diskette labels written by BURST, and the /I switch if the labels you want to read are in IBM format. The *device-name* argument is the unit name of the drive that holds your diskette.

Other switches are optional. You can use the global /L and /V switches to direct LABEL's output to either your printer or console. Or you can use the local /L switch to redirect the output. If you do not use either the global or local /L switch on your command line, LABEL sends its output to your terminal whether you use the global /V switch or not.

The following example demonstrates how you could use LABEL to read diskette labels created earlier by IMOVE.

```
R
LABEL/F Dxx/D<NL>

  LABEL revision x.xx     Reading from Dxx


VOL1 label
            Volume ID     VOLID
               Access
           Owner name
     Format indicator
        Record length     2
      Sector sequence
       Version number     3

HDR1 label
              Filename     FILENAME
         Block length     00512
         Begin extent     01001
           End extent     02009
        Record format     D
     Bypass indicator     B
Multivolume indicator
       Section number     01
        Creation date     851209
```

```
    File organization     S
      Expiration date     860309

ERMAP label
   First bad cylinder
  Second bad cylinder

Operation successfully completed
R
```

To read IBM-format labels, you would simply add the global /I switch to the command line shown above. The format of LABEL's output would be unchanged.

To read the diskette labels that BURST writes, include LABEL's global /B switch on your command line. As the following example illustrates, BURST's diskette labels do not resemble those written by IMOVE; they follow the format that BURST and IMOVE use when writing tape labels.

```
R
LABEL/B/F Dxx/D<NL>

  LABEL revision x.xx     Reading from Dxx

VOL1 label
            Volume ID     BURST
               Access
           Owner name     BURST
       Version number     3

HDR1 label
             Filename     Dxx 860521 163559
          File ID set     BURST
       Section number     0001
      Sequence number     0001
    Generation number     0001
       Version number     00
        Creation date
      Expiration date
               Access
          Block count     000000
            System ID     BURST

HDR2 label
          Record type     U
         Block length     08192
        Record length     65535
```

```
EOF1 label
             Filename      Dxx 860521 163559
          File ID set      BURST
       Section number      0001
      Sequence number      0001
    Generation number      0001
       Version number      00
        Creation date
      Expiration date
               Access
          Block count      000032
            System ID      BURST

Operation successfully completed
R
```

Writing Diskette Labels

If you request that LABEL write diskette labels, the utility creates three labels: a VOL1 label, a HDR1 label, and an ERMAP label. However, you can enter information only in the VOL1 label. For that label, you must supply a volume ID, and you may also specify an access code and an owner name.

NOTE: Don't prelabel diskettes in preparation for a BURST backup session. LABEL does not write diskette labels in the format or location that BURST uses when writing such labels, and BURST overwrites existing labels anyway.

To begin the labeling process, first insert the diskette you wish to label in its drive and then enter a command in this format:

LABEL[/I] device-name/D volid/V

The absence of the global /F switch indicates that you intend to create labels instead of reading existing ones, and the local /V switch is required since you must supply a volume identifier to be included in your diskette's VOL1 label.

Among the switches you can use at your discretion are those that control where LABEL sends its output. The global /V switch sends that output to your console; the global /L switch directs it to $LPT; and the local /L switch redirects the output.

Two other optional switches—the local /A and /O switches—allow you to enter an access code and an owner name in your diskette's VOL1 label. The access code must be a single character, and the owner name should be either 1 to 14 characters (AOS/VS format) or 1 to 10 characters (IBM format).

The following example presents a LABEL operation in which the user specifies a volume ID and owner name for his VOL1 label and requests that output be directed to his console:

```
R
LABEL/V Dxx/D VOLID/V OWNER/O<NL>

  LABEL revision x.xx     Writing to Dxx


VOL1 label
            Volume ID     VOLID
               Access
           Owner name     OWNER
     Format indicator
        Record length     2
      Sector sequence
       Version number     3

HDR1 label
             Filename
          Block length     00512
          Begin extent
            End extent
         Record format     D
     Bypass indicator      B
Multivolume indicator
        Section number
         Creation date
    File organization      S
      Expiration date

ERMAP label
   First bad cylinder
  Second bad cylinder

Operation successfully completed
R
```

In the example above, LABEL creates AOS/VS-format labels. To write IBM-format labels, add the global /I switch to your command line and make sure that your owner name is 10 characters or less. LABEL's output will look the same.

Using LABEL with Tapes

There are only a few differences between using LABEL with diskettes and with tapes:

- You must use the global /T switch on your command line if you want to read from or write to a tape.
- A magnetic tape can have from 0 to 9 user volume labels after its VOL1 label; therefore, the LABEL utility provides for reading and writing such labels.
- The global /P and /S switches enable you to partially initialize a labeled tape; there is no corresponding operation for diskettes.

For specific instructions on how to read tape labels, write tape labels, and initialize a labeled tape, see the subsections below.

Examining Tape Labels

To read the existing labels on a tape, mount your tape and then issue a command in this format:

```
LABEL/F/T[/I] device-name/D
```

The global /T switch on your command line indicates that you want to read tape, not diskette, labels. If you want to read AOS/VS-format labels, omit LABEL's /I switch; to read labels in IBM format, include that switch.

Several switches are optional. You can use either the global /L or /V switch to direct LABEL's output to your printer or to your console. Or you can use the local /L switch to redirect the output. If you do not use either the global or local /L switch, LABEL sends its output to your terminal whether you use the global /V switch or not.

The ensuing example illustrates how you could use LABEL to read tape labels that were created earlier with IMOVE. (You could use the same command to examine BURST tape labels, and the only difference in the output would be the contents of a few of the fields.)

```
R
LABEL/F/T/V xTx/D<NL>

  LABEL revision x.xx     Reading from xTx

VOL1 label
            Volume ID     VOLID
               Access
           Owner name
       Version number     3
```

```
HDR1 label
             Filename     FILENAME
          File ID set     VOLID
       Section number     0001
      Sequence number     0001
    Generation number     0001
       Version number     00
        Creation date      85344
      Expiration date      86069
               Access
          Block count     000000
            System ID     DG/RDOS

HDR2 label
          Record type     U
         Block length     08191
        Record length     65535

Operation successfully completed
R
```

To read IBM-format labels, add the global /I switch to the command line shown above.

Writing Tape Labels

If you use LABEL to write tape labels, the utility creates a VOL1 label, 0 to 9 user volume labels, and a HDR1 label. On your command line, you must supply a volume ID to be included in your tape's VOL1 label, and you may also specify an access code and owner name to appear in that label. If you choose to write user volume labels on your tape, you must enter the contents of those labels as well.

NOTE: Don't use LABEL to prelabel tapes in preparation for a BURST backup session. When BURST runs, it overwrites any existing labels.

To start the labeling process, mount your tape and then issue a command in this format:

LABEL/T[/I] device-name/D volid/V

Use the global /I switch only if you want to create IBM-format labels.

As an option, you can also use one of three switches to direct LABEL's output to a particular device. The global /V switch sends that output to your console; the global /L switch directs it to $LPT; and the local /L switch redirects the output.

Two other optional switches, the local /A and /O switches, allow you to enter an access code and an owner name in your tape's VOL1 label. The access code must be a single character, and the owner name should be either 1 to 14 characters (AOS/VS format) or 1 to 10 characters (IBM format). The local /U switch instructs LABEL to insert text in a user volume label. You can create up to nine such labels on each tape.

The following example illustrates a LABEL operation in which a user specifies a volume ID and an owner name to be inserted in his VOL1 label and supplies the text for two user volume labels. LABEL's output is directed to the user's terminal.

```
R
LABEL/T/V xTx/D VOLID/V OWNER/O "BACK UP YOUR FILES REGULARLY"/U
"OR ELSE!"/U<NL>

   LABEL revision x.xx     Writing to xTx

VOL1 label
             Volume ID     VOLID
                Access
            Owner name     OWNER
        Version number     3

UVL1 label
Contents:
BACK UP YOUR FILES REGULARLY

UVL2 label
Contents:
OR ELSE!

HDR1 label
              Filename
           File ID set
        Section number     0001
       Sequence number     0001
     Generation number     0001
        Version number     00
         Creation date      00000
       Expiration date      00000
                Access
           Block count     000000
             System ID     DG/RDOS

Operation successfully completed
R
```

Initializing a Tape

Besides reading and writing tape labels, LABEL can erase the data from any labeled tape that contains a valid VOL1 label. To initiate this process, you supply on your command line a volume ID and the text for any user volume labels you want the utility to write. Then, if the volume ID on your command line matches the volume ID in your tape's VOL1 label, LABEL takes the following steps: it preserves the existing VOL1 label, writes any user volume labels you requested, creates a HDR1 label followed by two tape marks, and writes an end-of-file label (EOF1) followed by two tape marks. At the end of this process, your tape appears to have header and trailer labels but no data.

To begin such a session, mount your tape and then enter a command in this format:

LABEL/P/T[/I] device-name/D volid/V

Use the /I switch only if your tape's current VOL1 label is in IBM format. Also, make sure that the volume ID on your command line matches that in your tape's VOL1 label.

Beyond the mandatory arguments and switches, you may include on your command line one of the three LABEL switches that direct output to a particular device. The global /V switch sends that output to your console; the global /L switch directs it to $LPT; and the local /L switch redirects the output. In addition, you may use the local /U switch to supply text for 1 to 9 user volume labels. (You use a separate /U switch for each user volume label you want to create.)

In the LABEL example that follows, a user asks the utility to initialize his tape, write a user volume label, and send all output to his console:

```
R
LABEL/P/T/V xTx/D VOLID/V "BACK UP YOUR FILES REGULARLY"/U<NL>

  LABEL revision x.xx     Partial initialization of xTx

VOL1 label
            Volume ID     VOLID
               Access
           Owner name     OWNER
       Version number     3

UVL1 label
Contents:
BACK UP YOUR FILES REGULARLY

HDR1 label
             Filename
          File ID set
       Section number     0001
```

```
      Sequence number     0001
    Generation number     0001
       Version number     00
        Creation date      00000
      Expiration date      00000
              Access
         Block count      000000
           System ID      DG/RDOS

EOF1 label
            Filename
          File ID set
       Section number     0001
      Sequence number     0001
    Generation number     0001
       Version number     00
        Creation date      00000
      Expiration date      00000
              Access
         Block count      000000
           System ID      DG/RDOS

Operation successfully completed
R
```

LABEL's Error Messages

Below is a list of error messages you might receive from LABEL. (Some of the messages appear after the more general message `Unexpected error while performing requested operation.`) Each message is accompanied by an explanation of what problem has occurred and how you should respond.

`Argument length`

You tried to write to a label a volume ID, an owner name, or an access code that contains more characters than LABEL allows. The utility's help message details how long each of these arguments can be.

`Bad tape`

LABEL encountered a problem while writing data to, or reading data from, your tape. Try using a different tape.

`Device name argument required`

You failed to specify on your command line the unit name of the device that you want to read from or write to. This argument, followed by the local /D switch, is required for all LABEL operations unless you only want to display a help message. The unit names of all the diskette and tape drives supported by DG/RDOS are listed in Appendix A.

`Duplicate switch`

You have used more than one local /D, /V, /O, /A, or /L switch on your command line. The only local switch that you can legally use more than once is the /U switch, which allows you to write user volume labels.

`Error, no ERMAP label found on diskette.`

You asked LABEL to read AOS/VS-format labels from a diskette that does not contain an ERMAP label. The diskette may be unlabeled, or it may contain labels written in a different format.

`Error, no HDR1 label found on diskette.`

You asked LABEL to read AOS/VS-format labels from a diskette that does not contain a HDR1 label. The diskette may be unlabeled, or it may contain labels written in a different format.

`Error, no VOL1 label found on diskette.`

You asked LABEL to read AOS/VS-format labels from a diskette that does not contain a VOL1 label. The diskette may be unlabeled, or it may contain labels written in a different format.

`Illegal or meaningless switch combination`

This message may indicate that you accidentally used a nonexistent switch. (All valid switches are listed Tables 15-1 and 15-2.) Or you may have used a combination of switches that makes no sense to LABEL. For example, you could have omitted the global /T switch, to indicate that you want to write labels on a tape, and then used the local /U switch to request a user volume label. The utility cannot write a UVL on a diskette.

`Not ready`

This message may indicate that you did not mount your tape properly or that your tape drive is not on-line. Another possibility is that you typed the unit name of your tape drive incorrectly on your LABEL command line.

`Tape error, not End Of File`

You may have tried to write labels on a write-protected tape. If this is the case, you will also receive the error message `Write protected`.

`Tape record not 80 bytes long`

Most likely, you attempted to read labels from an unlabeled tape.

`Too many /U switches`

On your command line, you used more than nine local /U switches, the maximum number that LABEL will write.

`Unknown label type; may be wrong format`

This message usually indicates one of two things. Either you tried to read IBM-format labels from a tape that contains AOS/VS-format labels, or you attempted to read AOS/VS-format labels from a tape with IBM-format labels.

`Volume ID mismatch with /P or /S`

When you use LABEL to initialize a tape, the volume ID on your command line must match that in your tape's VOL1 label. If the two do not match, use LABEL to read your tape's labels, note the volume ID in the VOL1 label, and then retype your command line.

`Write protected`

You tried to write to a tape that is write protected. Enable the tape for writing and then type your command line again.

-End of Chapter-

Chapter 16

Using the Disk Initializer

Read this chapter when you want to

- Software format a new (blank) disk, or reformat an old one
- Reserve space on your disk for coresident diagnostics
- Test a disk for bad blocks with the option of recording in the disk's bad block table any flawed blocks found
- Report bad blocks to be entered in the bad block table
- Select the size and location of a disk's remap area
- Set a disk's frame size
- Display complete information about a disk

The disk initialization program DKINIT is an interactive stand-alone utility whose main function is to prepare a disk or diskette to hold system or user software safely. If you brought up your own first system, you already have some experience with this program. This chapter explains the rest of its capabilities in the following sections:

- An Overview of DKINIT
- Starting the Program
- Full Initialization
- Other DKINIT Functions
- DKINIT's Error Messages

An Overview of DKINIT

The software formatting utility DKINIT is essential to the processes of creating new systems, introducing new disks, backing up data, and maintaining the integrity of your disks. For instance, DKINIT performs the following tasks for all disks:

- Bad block testing
- Bad block accounting
- Disk or diskette identification

Bad block testing detects any blocks on a disk or diskette that will not faithfully record and retain information. DKINIT performs this testing by writing to your disk and then reading up to five unique patterns, each composed of 16 bits. As a group, these patterns exercise all possible combinations of 0 and 1 in any two adjacent bits. (The patterns are shown later in Tables 16-2 and 16-3.) Every disk block that is not part of a coresident area is tested in this way. A block is considered bad if (a) DKINIT cannot write to it, (b) DKINIT cannot read it, or (c) the block contains an unexpected pattern when DKINIT reads it.

Bad block accounting enables DG/RDOS and you to avoid unusable disk blocks. DKINIT records the locations of such blocks and reserves a remap area—an area of contiguous, usable disk blocks—that DG/RDOS can access in place of the defective blocks.

DKINIT describes in a bad block table each flawed block it encounters. This table resides only in main memory until the bad block test is completed; then DKINIT writes the table to block 4 on your disk or diskette. (If block 4 is unusable, DG/RDOS cannot run on this disk.) Whenever the disk is in use, the table occupies a place in memory, providing the executing software with immediate access to its contents. In addition, you can use DKINIT to update the contents of this table whenever you discover a new bad block.

Since the system cannot use a bad block, DKINIT also provides for a remap area, a set of disk blocks to which the operating system is referred when it tries to access a bad block. By default, DKINIT bases the size of this remap area on the model number of the disk you are initializing, although the utility always supplies an area at least as large as the number of bad blocks it finds. Or you can specify the size of the remap area if you choose to. You may also select a starting address for this remap area, or you can accept the default placement among the highest addresses on the disk. DKINIT records both the size and location of the remap area in your disk's bad block table (block 4).

Disk identification involves recording on block 3 of your disk key information about that disk. To obtain this information, DKINIT begins all interactive sessions by requesting the model number and unit name of your disk. Then, using your answers, the program defines such characteristics as the number of tracks per cylinder on your disk, the number of sectors per track, the number of logical blocks on your disk, and the checksum for block 3. The frame size, which influences the efficiency with which your system accesses files, is set either by default, according to the type of disk, or by you, according to the number of files you plan to have in each directory.

Together, bad block testing, bad block accounting, and disk identification support the creation of the master allocation (MAP.DR) and system (SYS.DR) directories, without which a DG/RDOS system cannot run. Through the master allocation directory—a file of contiguous bits, each of which corresponds to a disk block—DG/RDOS monitors and controls the availability of every block on the disk.

It does so by setting to 1 those bits whose corresponding blocks are in use and setting to 0 all bits whose corresponding blocks are not in use. The system directory contains file descriptors that enable the operating system to locate files on the disk.

MAP.DR relies on DKINIT to test and account for bad blocks so that it can prevent all software from using those blocks, while SYS.DR depends on DKINIT for the hash frame size that controls ease of access to its file descriptors. Together, these directories and the disk initializer help prevent software failures and ensure the efficient operation of your DG/RDOS system.

Starting the Program

There are two ways to start DKINIT from disk. First, you can boot the disk that contains the program and then, at the `Filename?` prompt, enter the program name `DKINIT<NL>`. (For complete instructions on how to reach the `Filename?` prompt, see Chapter 10.) Or if you are running the Command Line Interpreter and are at the R prompt, simply type the command `BOOT DKINIT<NL>`. In either case the system starts the disk initializer and displays the following message:

```
Disk Initializer - Revision xx.xx

  *--> Help can be obtained to any question
  *--> simply respond to the question with HELP or ?

Disk drive model number?
```

As this message indicates, DKINIT has on-line information to help you answer any program queries you are not sure how to respond to. For example, the first question DKINIT asks is `Disk drive model number?` If you do not know the model number of the disk you wish to work with, type `?<NL>` or `HELP<NL>`, and a list containing entries similar to the following one appears:

```
6268   DJn       .37 MB  5-1/4" 48 TPI diskette (Desktop)
. . .
```

At the conclusion of this list, DKINIT repeats its question so that you can enter the appropriate model number.

```
Disk drive model number? 6268<NL>

Disk unit?
```

To identify the specific disk you want it to examine, DKINIT also needs to know the disk's unit name. (In the help screen entry shown above, this name appears just after the model number.) Type in this identifier, and press New Line.

```
Disk unit? DJ0<NL>

6268   DJn       .37 MB  5-1/4" 48 TPI diskette (Desktop)

Command?
```

DKINIT responds by displaying information about your disk and then asks for a command. To see the nine functions DKINIT performs, ask for help in answering the Command? question. A summary similar to the one in Table 16-1 appears:

Table 16-1 DKINIT Commands and Their Functions

Command	Function
FULL	Provides for total software formatting of a disk. FULL first gives you the option of establishing a coresident diagnostic area. Then the routine checks the portion of the disk not reserved for diagnostics for bad blocks using up to five scans of the disk and logs in a bad block table any flawed blocks it finds. You also have the option of declaring additional bad blocks. Finally, you may use defaults or set values for the remap area's size and physical location and for the disk's frame size.
PARTIAL	Scans your disk for new bad blocks and records any such blocks in the bad block table.
TEST	Scans your disk for new bad blocks, but does not update the bad block table.
ENTER	Allows you to declare bad blocks in addition to those found by DKINIT.
REMAP	Permits you to add bad blocks to the bad block table and then to specify a size and starting location for your disk's remap area.
FRAME	Sets your disk's frame size.
LIST	Displays on your console information about the frame size, remap area, diagnostic area, and bad blocks on your disk.
DISK	Directs DKINIT to run again from the beginning so that you can work on another disk.
STOP	Halts the disk initializer and returns you to the virtual console or System Control Program.

The next two sections in this chapter discuss these functions in greater detail.

Full Initialization

With the exception of diskettes that will hold IMOVE dump files or a diskette mirror image created with FCOPY, you must software format each of your disks with the FULL command before you use that disk for the first time. This command allows you to reserve part of your disk for coresident diagnostics; it checks the surface of your disk for bad blocks; and it combines the functions of the ENTER, REMAP, and FRAME commands, which are discussed in the next section.

NOTE: After using the DKINIT FULL command on a disk, you must start the Command Line Interpreter and use the command INIT/F to finish initializing your disk. INIT/F establishes the original master allocation and system directories on a new disk and overwrites these directories on a used one. In doing so, INIT/F effectively destroys any existing file structure.

Here is how the FULL command works. After offering you the option of reserving disk space for coresident diagnostics (discussed below), the FULL command instructs DKINIT to write one or more of the patterns shown in Tables 16-2 and 16-3 to every word of every sector on the disk (outside of any blocks you may have set aside for a coresident diagnostic area). The program writes one pattern and then reads the disk to ensure that each block has recorded the pattern accurately. Any block that fails this test is recorded in a bad block table on block 4 of the disk.

Table 16-2 Test Patterns for Disks with the Unit Name DAx

Pattern Number	Bit Configuration	Octal Value
1	0010010100100101	022445
2	1111011011110110	173366
3	1111000111110001	170761
4	1001110010011100	116234
5	1100100111001001	144711

Table 16-3 Test Patterns for All Other Disks

Pattern Number	Bit Configuration	Octal Value
1	1101101101101101	155555
2	1011011011011011	133333
3	0110110110110111	066667
4	1101101101101110	155556
5	1010101010101010	125252

Once DKINIT has run these patterns, the FULL command allows you to declare additional flawed blocks and, after all bad blocks have been accounted for, to declare a remap area size and starting address of your choice. Finally, the FULL command presents a default frame size, along with the minimum and maximum number of blocks you can set aside for the system directory, SYS.DR. (Remap areas and frame sizes are discussed in more detail later in this chapter.)

Coresident Diagnostics

When you issue the FULL command, the program tells you how to abort the initialization process and then asks if you would like to reserve space on your disk for diagnostic programs:

```
Command? FULL<NL>

Command destroys any previous RDOS disk structure
RDOS INIT/F must be done on the disk after the command
Type CONTROL-A now to abort without loss

Do you want a coresident diagnostic area?
```

If you choose to have a coresident diagnostic area, DKINIT sets aside a contiguous area at the end of your disk. Such an area is off limits to the operating system and the system utilities. It is reserved to hold both diagnostic programs that a Data General Field Engineer can load for you and microcode and diagnostics that you can load yourself. For details about the diagnostics that Field Engineering can provide, place a hardware service call.

NOTE: If you are formatting the first fixed disk on a DG/500, an MV/1000 DC, an MV/1400 DC, an MV/2000 DC, or an MV/2500 DC system, you should declare a diagnostic area even if you do not plan to install coresident diagnostics. An important step in setting up these machines is to write power-up diagnostics and CPU instruction microcode to such an area.

To establish a coresident diagnostic area, simply answer YES to DKINIT's last question. (If you are not interested in such an area, answer NO and skip the rest of this section.) A YES response produces the following message:

```
Do you want a coresident diagnostic area? YES<NL>

The default coresident diagnostic area size is 5000 blocks
How big would you like your diagnostic area to be?
```

Type New Line to accept the default size, or enter an integer between 1 and 32767 (decimal) to define the size, in blocks, of the diagnostic area. If your response would result in a disk with fewer than 128 blocks left for the operating system, the error message `Diagnostic area size is too large for this disk` appears, and DKINIT repeats its question.

Once you specify an acceptable number of blocks for your coresident diagnostics, the program subtracts the requested size from the total disk size and stores the computed value in the disk's identification block (block 3). Then the DKINIT FULL dialog continues with a question about the number of patterns you want DKINIT to use in analyzing the surface of your disk.

The Rest of the FULL Command's Dialog

Whether you requested a coresident diagnostic area or not, you should now see the following prompt:

```
Number of patterns to run (1-5) ?
```

As mentioned earlier, the disk initializer checks the surface of a disk for bad blocks by writing a 16-bit pattern to every word of a block and then reading the block to ensure that it recorded the pattern accurately. Each pattern takes some time to run; we recommend that you run all 5 patterns since an undetected bad block can result in your losing data that is not recoverable. In general, the time required to run DKINIT patterns is roughly equal to one minute per megabyte; some disks are significantly faster.

After you have typed in a number of patterns, DKINIT begins its analysis of your disk's surface and reports its progress on your console.

```
Number of patterns to run (1-5) ? 5<NL>

***  Pattern # 1 (155555) ***

***  Pattern # 2 (133333) ***

...
```

DKINIT describes each pattern as it begins writing the pattern. If it finds a bad block on any pattern, it writes to the console an error message that includes the block number of the bad block:

```
CHECKSUM ERROR - Bad block = 000526
Data Compare Error - Bad Block = 000231
```

NOTE: If DKINIT displays a lot of ADDRESS ERROR messages, you may be working with a diskette that is not hardware formatted. Or you may be trying to format a 48 TPI diskette in a 96 TPI drive (or vice versa). Stop DKINIT by typing Ctrl-A.

Eventually, the program finishes its analysis and displays this message:

```
***  All patterns run ***

Do you wish to declare any blocks bad
that are not already in the bad block table?
```

Answer NO unless you want to declare flawed blocks that DKINIT has not already detected. If you answer YES, the program requests that you supply a bad block number in octal form.

DKINIT's next prompt concerns your disk's remap area.

```
Default remap area size is x block(s) long
It needs to be at least y block(s) long
Remap area size (type RETURN for default) ?
```

DKINIT calculates *y* by adding the number of bad blocks it found while checking the surface of the disk to the number of blocks that you requested be added to the bad block table. Therefore, the number of blocks in your remap area must be equal to or greater than *y*. If most or all of the blocks on your disk are sound, we recommend that you accept the default value.

```
Remap area size (type RETURN for default) ? <NL>

Remap area start block number (type RETURN for default) ?
```

Unless directed otherwise, DKINIT locates the remap area among the highest physical addresses on your disk or diskette. We recommend that you ask for this default placement unless you have a large number of bad blocks and feel that it is important to cut down on head positioning time. In such a case, you can locate the remap area among the middle range of addresses; however, this strategy limits the number of large contiguous files you can create.

NOTE: Requesting the default placement for the remap area places that area at the very end of your disk, unless there is a bad block near the end of the disk. In such a case, the remap area has to begin at a lower disk address. This information becomes important if you use BURST to back up one disk and then attempt to restore your files on another disk of the same type. In that situation, the two disks must have remap areas that begin at the same location. If you asked for the default placement when you initialized both disks, you might assume that the remap areas are in the same place, but this may not be the case.

```
Remap area start block number (type RETURN for default) ? <NL>
Remap area will start at block x

Default frame size is x, min is 1, and max is y
Frame size (type RETURN for default) ?
```

The default and maximum frame sizes depend on the type of disk you are working with. If you are unsure about your directory structure at this time, it is safest to use the default answer. If you know that each of your directories will hold only a few files, you may want to use a value lower than the default. Conversely, if your directories will hold a great many files, you may select a frame size larger than the default.

DKINIT now tells you that your disk has been fully initialized and asks for another command.

```
Full disk init complete

Command?
```

To leave DKINIT, use the STOP command.

Other DKINIT Functions

You should use the following commands (with the exception of DISK and STOP) only after you have fully initialized your disk or diskette.

The PARTIAL Command

Like the FULL command, PARTIAL checks a disk for bad blocks and records in the disk's bad block table any flawed blocks it finds. However, PARTIAL differs from FULL in that a partial initialization involves a read-only analysis of the recording surface and does not require that the CLI's INIT/F command be issued afterward. In other words, PARTIAL allows you to deal with new bad blocks without erasing all your files.

Still, you must be careful when using the PARTIAL command because you could seriously damage your file structure if, for example, the bad block DKINIT records in the bad block table belongs to a file. (You can check to see whether a bad block belongs to a file by using the BURST utility's OWNER command.) Also, if new entries cause the bad block table to overflow, you must execute the FULL or REMAP command to increase the table's size.

The following dialog occurs during partial initialization:

```
Command? PARTIAL<NL>

*** Checking for bad blocks ***
```

If the program finds new bad blocks, it reports them in the following manner:

```
CHECKSUM ERROR - Bad Block = 001154
ADDRESS ERROR - Bad Block = 001220

New errors found on disk
Updating bad block table to include them
```

If there are no new bad blocks, PARTIAL reports that fact and prompts you for another command.

```
No new errors detected on disk

Partial init run complete

Command?
```

DKINIT is now ready to accept another command.

The TEST Command

Typically, you use TEST to check for a new bad block and, if one exists, to find its location on the disk. Then, before you record the bad block in the bad block table, you can use the BURST program to determine whether the block in question is part of a file. (See the section "Using the OWNER Command" in Chapter 13.) If the block does not belong to a file, you can use the ENTER command to log the block in the bad block table; if the block does belong to a file, consult Appendix B for instructions on how to proceed.

A typical dialog follows:

```
Command? TEST<NL>

***  Checking for bad blocks ***
```

New bad blocks are reported in the following format:

```
CHECKSUM ERROR - Bad block = 000526
```

If TEST finds no new bad blocks, it returns this message:

```
No new errors detected on disk

Partial init run complete

Command?
```

DKINIT is ready to execute another command.

The ENTER Command

You might use this routine to record a bad block you discovered with the TEST command. Or, if a BURST backup failed because of a new bad block, you could use ENTER to log that block's address in the bad block table. You need not issue the CLI command INIT/F after entering a new bad block.

As with the PARTIAL command, you should exercise caution when using ENTER. For instance, you might jeopardize all the information on your disk if you entered in the bad block table the location of a block that occurred in the middle of a program file. (See Appendix B for more information on this subject.) Also, if new bad block entries cause the bad block table to overflow, you must use either the FULL or REMAP command to enlarge the table and then follow up by issuing the CLI's INIT/F command, which in effect erases all your current files.

A sample dialog with the ENTER routine follows:

```
Command? ENTER<NL>

Bad block number (type RETURN to stop) ?
```

At this point, you enter the octal block number of the block you want to register in the bad block table.

```
Bad block number (type RETURN to stop) ? 136<NL>
Bad block entered

Bad block number (type RETURN to stop) ?
```

DKINIT returns to this question each time it enters a bad block you report. Once you have reported all the flawed blocks you want to log, type a New Line to exit from the loop.

```
Bad block number (type RETURN to stop) ? <NL>
```

```
Do you wish to declare any more blocks bad? NO<NL>

Command?
```

DKINIT is ready to execute another command.

The REMAP Command

First, like the ENTER command, REMAP lets you declare any number of new bad blocks. The routine then adds the number of blocks you declare to the number of blocks already recorded in the bad block table and asks you to specify a remap area at least as large as this total. Finally, you are asked to supply the block number (octal) at which this area will start. In most instances, you should accept the default location at the top addresses of your disk. If, however, you have a large number of bad blocks and want to cut down on head positioning time, you may benefit from locating the remap area among the middle range of addresses.

When REMAP finishes running, you must use the CLI command INIT/F; therefore, REMAP effectively destroys any existing file structure.

You may find REMAP useful if, while you are running ENTER, your new bad block entries cause the bad block table to overflow. REMAP allows you to increase the size of that table. You can also use REMAP if one of the blocks in your current remap area becomes unusable.

Here is a typical REMAP dialog:

```
Command? REMAP<NL>

Command destroys any previous RDOS disk structure
RDOS INIT/F must be done on the disk after the command
Type CONTROL-A now to abort without loss

Do you wish to declare any blocks bad
that are not already in the bad block table?
```

If you answer NO, REMAP skips directly to its question about your remap area's size. If, however, bad blocks have developed since you fully initialized your disk, answer YES and the routine will let you record those blocks in the bad block table.

```
Bad block number (type RETURN to stop) ?
```

Enter the block number, in octal, of the bad block you want to declare.

```
Bad block number (type RETURN to stop) ? 345<NL>
Bad block entered

Bad block number (type RETURN to stop) ?
```

REMAP returns to this question each time it records a new bad block. Once you have declared all the bad blocks you want to report, type a New Line to exit from the loop.

```
Bad block number (type RETURN to stop) ? <NL>

Do you wish to declare any more blocks bad? NO<NL>

Default remap area size is x block(s) long
It needs to be at least y block(s) long
Remap area size (type RETURN for default) ?
```

The number of blocks in your remap area must be equal to or greater than *y*. If most or all of the blocks on your disk are usable, it is best to accept the default value.

```
Remap area size (type RETURN for default) ? <NL>

Remap area start block number (type RETURN for default) ?
```

Specify the default answer to locate the remap area among the highest addresses on your disk. Or you can define an area elsewhere on the disk by entering a starting block number (in octal). If you choose the latter option, make sure that there are enough contiguous, usable blocks at this location to hold your remap area. REMAP now displays the number of the first block in your disk's remap area and prompts you for another command.

```
Remap area start block number (type RETURN for default) ? <NL>
Remap area will start at block 001276

Command?
```

NOTE: Requesting the default placement for the remap area places that area at the very end of your disk, unless there is a bad block near the end of the disk. In such a case, the remap area has to begin at a lower disk address. This information becomes important if you use BURST to back up one disk and then attempt to restore your files on another disk of the same type. In that situation, the two disks must have remap areas that begin at the same location. If you asked for the default placement when you initialized both disks, you might assume that the remap areas are in the same place, but this may not be the case.

The FRAME Command

You use the FRAME command to specify the number of data entry blocks you want to allocate initially for the primary partition's system directory (SYS.DR). (Later, as you create secondary partitions and subdirectories, these directories will have SYS.DRs with the same number of data entry blocks.) In these blocks the system directory stores user file descriptors (UFDs), which enable DG/RDOS to locate the files and directories on your disk or diskette. Each UFD occupies 18 words of a data block and describes the characteristics of a file—for example, its name, extension, attributes, location, and date of last access. Each block of the system directory can contain up to 14 UFDs; any overflow slows response time considerably because DG/RDOS must search for and examine one or more additional frames.

We recommend that you select a frame size smaller than the default if you plan to have many user directories composed of relatively few files. Conversely, if you will have a small number of directories with many files in each, it is a good idea to choose a frame size larger than the default. As a rule of thumb, the best frame size is approximately one-tenth the number of files that will occupy any directory on the disk.

Once you have selected a frame size and stopped DKINIT, you must enter the CLI and use the INIT/F command to set up a new system directory on your disk.

A typical FRAME dialog follows:

```
Command? FRAME<NL>

Command destroys any previous RDOS disk structure
RDOS INIT/F must be done on the disk after the command
Type CONTROL-A now to abort without loss

Default frame size is x, min is 1, and max is y
Frame size (type RETURN for default) ?
```

Specify a number between 1 and *y* (preferably a prime number), or select the default frame size by typing a New Line.

```
Command?
```

DKINIT is ready to execute another command.

The LIST Command

Here is the type of output LIST might produce for a disk with a coresident diagnostic area in which diagnostics have not been installed:

```
Command? LIST<NL>

On unit DA0:  6329   DAn      120 MB  5-1/4" Sealed Moving Head Disk

Frame size = 83    Remap area size = 85
Remap area start block number = 00000662646
Diagnostic area size = 5000
Diagnostic area start block number = 00000662773
Diagnostics have not been installed
There are no bad blocks

Command?
```

DKINIT is now ready to accept another command.

The DISK Command

The DISK command directs DKINIT to run again from the beginning so that you can work on another disk. A sample dialog follows.

```
Command? DISK<NL>

Disk Initializer - Revision xx.xx

  *--> Help can be obtained to any question
  *--> simply respond to the question with HELP or ?

Disk drive model number? 6268<NL>

Disk unit? DJ0<NL>

6268   DJn      .37 MB  5-1/4" 48 TPI Diskette (Desktop)

Command?
```

The STOP Command

STOP halts the disk initializer and returns you to the virtual console (16-bit systems) or the System Control Program (32-bit systems).

DESKTOP GENERATION or DG/500 system:

```
Command? STOP<NL>

...
!
```

MV/Family system:

```
Command? STOP<NL>


...             [The contents of the accumulators and the location
...             counter are printed here.]
SCP-CLI>        [or SCP-CLI/Jp0> on some systems]
```

From this point, you can either turn off the computer or load and run another program.

DKINIT's Error Messages

Below is a list of error messages you might receive from DKINIT. Each message is accompanied by an explanation of what problem has occurred and how you should respond. Refer to "DSKED Error Messages" in Appendix B of *RDOS/DOS Debugging Utilities* for a list of hardware-related error messages.

```
All patterns run, too many disk errors to complete
```

Your disk(ette) has too many bad blocks to be usable. If you are working with a diskette, you can try to make it usable by hardware formatting the diskette. If you are working with a hard disk, a Field Engineer must help you hardware format your disk, so place a hardware service call.

```
Bad block contained in remap area specified
Please specify another area
```

You specified a starting address for your remap area that would cause a bad block to fall within that area. To solve this problem, you can (1) request the default starting address or (2) type in successively higher or lower starting addresses until DKINIT accepts your answer.

```
Bad block is in disk bad block remap area
FULL init suggested
ABORTING
```

A new bad block has developed and is in the current remap area. You can move the location of this remap area by using either the FULL or REMAP command.

```
Bad block list is corrupt
ABORTING
```

The bad block table on your disk contains invalid data. Use the FULL command to create a new bad block list.

```
Bad block list is full
unable to enter any more bad blocks
```

You attempted to enter more bad blocks in the bad block table than that table can hold. Use the REMAP or FULL command to enlarge your disk's remap area.

```
Block is part of bad block remap area
it can only be declared bad using FULL init or REMAP
```

The block you declared bad is part of your disk's remap area, where bad blocks are not allowed. Use the FULL or REMAP command to move the remap area to a different location.

```
xxxxxx contiguous blocks are not available - please select a
smaller size
```

Because of either the size of your disk or the arrangement of bad blocks on that disk, DKINIT cannot create a remap area of the size you requested. Try successively smaller sizes until the utility accepts your answer.

```
Critical disk blocks are bad, RDOS cannot be built
ABORTING
```

During a full initialization, DKINIT discovered a problem with a block that is required for system use. You should hardware format your disk and then run DKINIT again.

```
Critical disk blocks bad - suggest FULL init
```

DKINIT's PARTIAL or TEST routine found an error in a block that DG/RDOS must use in order to run. You may be able to overcome this problem by hardware formatting your disk.

```
Data compare error
```

This error can occur while you are running the FULL routine. The message indicates that DKINIT was able to write a pattern on your disk and to read the disk; however, what the utility read from one block does not match what it wrote there.

```
Diagnostic area size is too large for this disk
```

You specified a size for your coresident diagnostic area that would leave less than 128 blocks for the operating system. Try successively smaller sizes for your diagnostic area until DKINIT accepts your answer.

```
Diagnostics may not have been installed correctly
```

DKINIT is unable to determine whether coresident diagnostics have been installed or not, which could be a signal that such diagnostics programs were installed incorrectly or that your disk is corrupt. To report this problem, place a hardware service call.

```
Disk ID is incorrect
ABORTING
```

Most likely you tried to use a command other than FULL, DISK, or STOP while working on a disk that had not been initialized with DKINIT. Another possibility is that block 3 on your disk has become corrupt.

```
Illegal diagnostic area size
```

You requested a diagnostic area of less than 1 block or more than 32767 blocks.

```
Illegal disk block number
```

You entered as a disk block number (1) a number that was not in octal, (2) the number of a block in the coresident diagnostic area, or (3) a number larger than the size of the disk in blocks.

`Illegal disk unit declaration`

In identifying your disk to DKINIT, you entered a unit name that does not match the model number you supplied previously. Consult Appendix A to find the correct unit name for your disk.

`Illegal frame size`

You requested a frame size smaller than the minimum or larger than the maximum allowed for your disk.

`Illegal number of patterns`

You accidentally requested that DKINIT run fewer than one or more than five patterns.

`Illegal remap area size`

You requested a remap area (1) smaller than the number of bad blocks DKINIT found while analyzing your disk or (2) greater than 126 blocks.

`Illegal remap area start block number`

You supplied a starting address for your remap area that would (1) cause this area to extend beyond the end of the disk or (2) place the remap area in front of the primary partition's MAP.DR.

`No room for bad block remap area due to too many bad blocks ABORTING`

Because of the arrangement of bad blocks on your disk, there is no room for a remap area of the default size. Hardware formatting your disk may help.

`Not enough room for update - suggest FULL init`

While you were executing the PARTIAL function, DKINIT found bad blocks that it was unable to add to the bad block table because that table was full. To solve this problem, use the FULL or REMAP command to increase the size of your remap area.

```
RDOS will not run with this block bad
block NOT ACCEPTED
```

You tried to enter in the bad block table a block that is required for system use. You may be able to recover your disk by hardware formatting it.

```
There is no current remap area
run FULL init to establish one
```

You used the REMAP function on a disk that does not contain a remap area; that is, your disk was never fully initialized. Run the FULL routine to set up your remap area.

```
*** Too many bad blocks - ABORTING ***
```

The utility has found more bad blocks than the bad block list can hold. You can use the FULL or REMAP command to increase the size of your remap area, or you can hardware format your disk.

```
Unknown drive type
```

The number you provided in response to DKINIT's `Disk drive model number?` question is not in the utility's table of model numbers. Check Appendix A to find the correct model number for your disk.

```
*** WARNING - RDOS cannot be run with this block bad ***
```

DKINIT has discovered that one of the blocks required for system use is unreliable. Try running a hardware formatter on the disk you are working with.

-End of Chapter-

Chapter 17

Using a RAM Disk

Read this chapter if

- You are unsure how to respond to SYSGEN's question about a RAM disk controller
- You need information on how to use a RAM disk to best advantage
- You want to back up your RAM disk

The RAM disk that DG/RDOS supports is simply a portion of system memory that you can access as if it were a disk. For example, you can make your RAM disk (mnemonic DR0) your current directory. You can create secondary partitions, subdirectories, and files on the RAM disk. You can transfer files between RAM disk, hard disks, diskettes, and tapes. And you can use most CLI commands, utilities, system calls, and user programs as if your RAM disk were a real disk. The system uses blocks of RAM-disk memory in the same manner that it uses sectors on real disks.

The remainder of this chapter will help you decide whether you should create a RAM disk, and if you do want one, how to use it. The chapter is divided into the following sections:

- Memory Requirements
- The Pros and Cons of Using a RAM Disk
- Creating and Working on a RAM Disk
- Backing Up Your RAM Disk

Memory Requirements

Before you generate a system that includes support for a RAM disk, you need to determine whether your computer has sufficient memory to accommodate such a disk. This is especially true if you are generating a DESKTOP GENERATION or DG/500 system.

On all systems, we generally recommend that you allocate at least 256 Kbytes of memory to hold DG/RDOS and your programs, and if you are working on a DESKTOP GENERATION Model 10 or 10SP, you must set aside another 32 Kbytes of system memory for a terminal emulator. Since we believe that the smallest useful RAM disk would be one about the size of a 48 TPI diskette (360 Kbytes), your computer should ordinarily have at least 768 or more Kbytes of memory before you consider generating system support for a RAM disk. (Depending on the size of your programs, however, you may want a RAM disk on a 512-Kbyte machine.) If you are working on an MV/Family system, where a minimum configuration contains at least 2 Mbytes of memory, obviously you have plenty of room for a RAM disk. However, bear in mind that a RAM disk reduces the amount of space available to background and foreground programs by the number of pages allocated for the RAM disk. If your programs are written to run more efficiently when large amounts of extended memory are available, you may want to set aside 4 or even 8 Mbytes of memory for your programs, and request the rest for your RAM disk.

The Pros and Cons of Using a RAM Disk

The primary benefit of having a RAM disk is that transferring data to or from a RAM disk is much faster than transferring data to or from a real disk. As a result, a disk-bound program that uses a RAM disk in some way can run quite a bit faster than the same program when a RAM disk is not available. This is especially true when the alternative to a RAM disk is a relatively slow hard disk. Our tests indicate that a few operations complete 10 times faster on a system with a RAM disk than they do on a system without one. More typical operations show performance gains of 10 to 300 percent.

At the same time, using a RAM disk can be dangerous. If there is a power failure, or you reboot the operating system or boot another program (you could do the latter by taking DG/RDOS down and bringing it back up, using the CLI command BOOT, or running a program that executes a .BOOT system call), **the contents of your RAM disk are lost**.

The trick, then, is to determine which files can be moved to RAM disk, and which operations can be performed there, safely. For example, moving copies of programs to a RAM disk, placing Business BASIC or Interactive COBOL swap space on a RAM disk, and making your RAM disk the master directory are all generally safe. However, placing data files that will be updated by your application on a RAM disk is not safe, unless you have a power-backup system.

Creating and Working on a RAM Disk

You generate a DG/RDOS system that supports a RAM disk by responding appropriately to the RAM-disk question posed by the SYSGEN utility and then to that asked by CONFIG. Unless your SYSGEN command line indicates that you want to build an operating system without overlays (in which case, you cannot have a RAM disk), SYSGEN asks you for the number of RAM-disk "controllers" in your configuration:

```
A DRn RAM disk controller (Yes or No) ?  [Yes]
```

Type New Line if you want support for a RAM disk, and answer NO if you do not. If you opt for a RAM disk, you define its size when you run CONFIG and see this question:

```
RAM disk memory (x-y pages) ?
```

The disk is automatically formatted by the operating system's initialization code. Then the starter CLI performs an INIT/F operation on the disk and releases it. All you do to make the RAM disk usable is initialize it (using the CLI command INIT).

Now let's look at an example of how you might use your RAM disk to enhance the performance of your system without sacrificing the integrity of your data. Suppose that you have an Interactive COBOL program that accesses a database and calls a number of Interactive COBOL subprograms. To make this application run faster, you might perform the following steps:

1. Move a copy of your runtime system, your program, your subprograms, and the CLI to your RAM disk (DR0). But leave your database on a real disk so that it will not be in danger of being lost as the result of a power outage or reboot of the operating system.

2. Release the current master directory (and the RAM disk if it is initialized). Then, using the CLI command MDIR, make DR0 your master directory.

   NOTE: You cannot boot your RAM disk or a stand-alone program residing on DR0. An attempt to do so may cause you to lose all the data stored on your RAM disk.

3. Make DR0 the current directory (using the command DIR), and chain to the CLI in the new master directory (using the command CHAIN).

   NOTE: You should chain to the CLI for this reason. During the execution of certain commands, the CLI creates and opens temporary files in the directory in which it was brought up. Therefore, if you release your original master directory and make DR0 your new master directory without chaining to a copy of the CLI on DR0, the CLI will not be able to find the temporary files it needs to operate.

4. Create link files on DR0 (using the command LINK) that point to your database files, which are still on a fixed disk.

5. Initialize the directory in which your database resides.

6. Start your Interactive COBOL program.

If you follow these instructions, your application should run significantly faster than it would if all of your files were on a hard disk. The reason for this is that when your main program calls a subprogram, an image of the calling program's .DD file is not written to a real disk, but is stored in system memory allocated for the RAM disk. Also, when control returns to the main program, the runtime system does not have to read your .DD and .PD files back in from a disk, but already has those files in RAM-disk memory.

If you are a Business BASIC user, you can follow a very similar procedure. The only difference is that instead of moving a copy of the Interactive COBOL runtime system to RAM disk, you should move your Business BASIC system files and the directories $LIB and $SYS.

Of course, not all applications will show a great increase in the speed of execution. Testing your applications with and without selected files on your RAM disk is the only way to tell for certain what you stand to gain from using such a disk.

NOTE: Remember as you experiment with putting different files on your RAM disk that you should not put on that disk an important data file that your application will update. That file would be lost if the power to your computer were interrupted or someone inadvertently rebooted DG/RDOS.

Backing Up Your RAM Disk

Before you issue a BOOT command to shut down DG/RDOS or execute the macro BYE.MC, be sure to use one of the following methods to back up the files you want to keep.

- You can use the CLI command MOVE or DUMP to copy one or more files to a magnetic storage device. If you use MOVE, you can also use the MOVE command to restore the contents of that disk after you reboot DG/RDOS. If you use DUMP, use LOAD to restore those contents when you reboot DG/RDOS.

- You can use FCOPY's "Copy a file" option to move a file from your RAM disk to a diskette or disk, and you can use the same option to move a file in the opposite direction. You cannot use FCOPY's "Duplicate a diskette" option to back up your RAM disk, even if it is the same size as a diskette.

- You can use the BURST utility's DUMP/V command to dump the contents of your RAM disk to diskettes or tapes before shutting down the operating system. Then you can use the BURST command LOAD/V to restore those contents when you reboot DG/RDOS.

- You can use the IMOVE utility to dump any or all of the files on your RAM disk to diskettes or tapes. You can also use the utility to restore the contents of that disk after you reboot DG/RDOS.

-End of Chapter-

Chapter 18

Making Changes to Your System

Read this chapter when

- You add new hardware to your system
- You receive a DG/RDOS update
- You receive a new revision of DG/RDOS

After you have run your DG/RDOS system for a while, you may decide to buy additional hardware such as a multiplexor, a disk, or a tape drive. Or you may receive from Data General a revised version of the operating system. The steps you need to perform if these situations arise are discussed in the following sections:

- Adding New Hardware
- Installing a DG/RDOS Update
- Installing a New Revision of DG/RDOS
- Updating Your Manuals

Adding New Hardware

Adding a new piece of hardware to an existing configuration involves two steps. First, of course, you must install and test the hardware according to the instructions in your hardware documentation. Then you must retailor your DG/RDOS system to support the new hardware by running either the SYSGEN program, the CONFIG program, or both.

Table 18-1 summarizes the hardware upgrades available on the systems supported by DG/RDOS and indicates which tailoring program(s) you need to run to make the operating system support each piece of hardware. Instructions on how to use SYSGEN and CONFIG are in Chapters 7 and 8.

Table 18-1 Hardware Upgrades

Hardware	Run SYSGEN?	Run CONFIG?
Winchester disk	Yes	No[1]
Diskette drive	Yes	No[1]
Tape drive	Yes	No[1]
Parallel printer	Yes	Yes[2]
Secondary console	Yes	Yes[2]
Multiplexor	Yes	Yes[2]
Terminal, serial printer, or plotter (as linked $ device)	Yes	Yes[2]
Terminal, serial printer, or plotter (as QTY line)	No[3]	Yes[2]
Memory	No	Yes[2]

1 If you changed any of CONFIG's defaults in your existing configuration, you need to run CONFIG and make the changes again.

2 After you run SYSGEN, your system is set up with the CONFIG program's defaults. If you do not want to change the defaults, you do not need to run CONFIG.

3 If you attach this device to a QTY line and your operating system does not support QTY lines, you will have to run SYSGEN.

If you add a Winchester disk to your configuration, you must also software format the new disk before you can use it. To perform this formatting, first run the utility DKINIT and issue the FULL command; then use the CLI command INIT/F. (For more information on the software formatting process, see the section called "Initializing Your Master Disk" in the chapter that deals with loading DG/RDOS on your hardware, or Chapter 16, which deals with DKINIT.)

Installing a DG/RDOS Update

Between revisions of DG/RDOS, Data General may issue an update for the operating system in order to correct a problem or provide an important enhancement. Such an update is not a complete version of the system, but consists of one or more files or patches that you use to replace or patch existing system files. Also with an update, you receive an update notice that explains the purpose of the update, and you may receive files containing documentation changes.

To install an update, first load any new system files onto your master disk or diskette. The exact command or utility you use to accomplish this can vary depending in how the information was stored on the release media, so consult the installation instructions on your update notice for specifics. Next, you may need to run the SYSGEN utility depending on which files you replaced. Then, if the update contains patches, apply the patches using the PATCH utility (this utility is documented in *RDOS/DOS Debugging Utilities)*. You may also need to run CONFIG. Again, consult your update notice for installation information specific to your update.

Installing a New Revision of DG/RDOS

Data General continually improves its software products and periodically distributes revised versions of them. Each such revision of DG/RDOS includes new versions of all the operating system's program and support files. It also includes a printed release notice and may include files containing documentation changes for some of the DG/RDOS manuals. If you are currently running DG/RDOS, the release notice contains the detailed instructions you must follow to install the revision, but the general procedures for changing revisions are covered below.

One step you should perform before actually installing a new revision is to back up your entire disk using either IMOVE or BURST. (For instructions on using these programs, see Chapters 13 and 14.) This backup is a simple matter of self-protection: if you have problems with the new revision, at least you will be able to fall back on your old system. This is also a good time to check your disk for new bad blocks. To perform such a check, use DKINIT's TEST command (see Chapter 16).

Once you have observed these precautions, the general procedure for loading a revision is to replace your old DG/RDOS program and support files with new files of the same name from your release diskettes or tape. The exact command or utility you use to accomplish this replacement depends on which command or utility Data General used to store the system files on your release media; therefore, you must consult your release notice to find out whether you should use MOVE, LOAD, or IMOVE to load the new files.

A possible alternative to backing up your disk and then overwriting your old system files with the new is to start the installation process by renaming the old DG/RDOS files. For example, you might change the name DGRDOS.SV to ODGRDOS.SV. Then you could load the revised system files onto your disk without overwriting the old. If you had any problem with the revised system, you could use your original DG/RDOS files without having to restore them from your backup media.

Anyway, after you have loaded the new system files, you finish installing a revision as follows. First, shut down the system you are running and start up the new one (Chapter 10 discusses startup and shutdown). Then, follow the directions in Chapters 7 to 9 for tailoring the new system. This last step will involve running both the SYSGEN and CONFIG programs.

Updating Your Manuals

Since not all the DG/RDOS manuals can be revised each time the software is updated or revised, updates and revisions usually include one or more text files that list documentation changes. Each such file contains the changes and additions for one manual in the document set and is named after the part number of that manual. Thus, a documentation-changes file for this manual would have the filename 093000470.03. ■

Once you have installed your update or revision, you should print these text files. When you do, you will have a series of update pages for one or more of the DG/RDOS manuals. To update a manual, simply separate the pages for that manual, three-hole punch them, and insert them in the proper place in the manual's binder.

-End of Chapter-

Appendix A

Disk, Diskette, and Tape Drives

Tables A-1, A-2, A-3, A-4, and A-5 list information about all of the storage devices that DG/RDOS supports on DESKTOP GENERATION, DG/500, MV/1000 DC, MV/1400 DC, MV/2000 DC, MV/2500 DC, MV/3500 DC, MV/5000 DC Series, MV/7800 XP, MV/9500, MV/15000, and MV/18000 configurations.

Table A-1 DESKTOP GENERATION System Storage Devices

Model	Description	Capacity	Device Code	Unit Name
6123	Reel-to-reel tape drive (1600 BPI)	20 Mbytes	22	MT0
6268	5.25" 48 TPI diskette drive	368 Kbytes	20	DJ0 (right) or DJ1 (left)
6270 [2]	Cartridge tape drive	19 Mbytes	22	MT0
6271	Winchester disk drive	15 Mbytes	26	DE0 (module) or DE1 (add-on)
6301	Winchester disk drive	38 Mbytes	26	DE0 (module) or DE1 (add-on)
6336	Winchester disk drive	70 Mbytes	26	DE0 (module) or DE1 (add-on)

[2] Do not use greater than 8192/S on BURST and IMOVE command lines.

Table A-2 DG/500 System Storage Devices

Model	Description	Capacity	Device Code	Unit Name
5512 [1]	Cartridge tape drive	60 Mbytes	22	MT0
5589, 6587	Reel-to-reel tape drive (1600 BPI)	40 Mbytes	30/31, 70/71	STn
6426 [1]	Cartridge tape drive	130 Mbytes	30/31, 70/71	STn
6506	Half-height Winchester disk drive	20 Mbytes	26	DE0 (first) or DE1 (second)
6507	Half-height Winchester disk drive	40 Mbytes	26	DE0 (first) or DE1 (second)
6508	Winchester disk drive	70 Mbytes	26	DE0
6509	Winchester disk drive	160 Mbytes	26	DE0
6510	5.25" 48 TPI diskette drive	368 Kbytes	20	DJ0 (upper) or DJ1 (lower)
6511	5.25" 96 TPI diskette drive	737 Kbytes	20	DJ0 (upper) or DJ1 (lower)
6512 [1]	Cartridge tape drive	22 Mbytes	22	MT0
6589	Reel-to-reel tape drive (800/1600/6250 BPI)	140 Mbytes	30/31, 70/71	STn
6675 [1]	Cartridge tape drive	150/320/525 Mbytes	30/31, 70/71	STn
6679 [1]	Cartridge tape drive	150 Mbytes	30/31, 70/71	STn

[1] Use 16384/S on BURST and IMOVE command lines.

Table A-3 MV/1000 DC, MV/1400 DC, MV/2000 DC, and MV/2500 DC System Storage Devices

Model	Description	Capacity	Device Code	Unit Name
6297	Half-height Winchester disk drive	40 Mbytes	24	DAn
6309	5.25" 96 TPI diskette drive	737 Kbytes	64	DA10
6310	Winchester disk drive	38 Mbytes	24	DAn
6328	Winchester disk drive	70 Mbytes	24	DAn
6329	Winchester disk drive	120 Mbytes	24	DAn
6341	Reel-to-reel tape drive (1600 BPI)	20 Mbytes	63	UT1n
6351, [1] 6444	Cartridge tape drive	22 Mbytes	23	UTn
6352, [1] 6426	Cartridge tape drive	130 Mbytes	63	UT1n
6363	Winchester disk drive	160 Mbytes	24	DAn
6446	Winchester disk drive	230 Mbytes	24,25	DAn
6491	Winchester disk drive	320 Mbytes	24,25	DAn
6539	Half-height Winchester disk drive	170 Mbytes	24	DAn
6554	Winchester disk drive	660 Mbytes	24,25	DAn
6577 [1]	Cartridge tape drive	150 Mbytes	23,63	UTn
6587	Reel-to-reel tape drive (1600 BPI)	40 Mbytes	63	UT1n
6662	Half-height Winchester disk drive	330 Mbytes	24,25	DAn

[1] Use 16384/S on BURST and IMOVE command lines.

Table A-4 MV/3500 DC and MV/5000 DC Series System Storage Devices

Model	Description	Capacity	Device Code	Unit Name
6351, [1] 6444	Cartridge tape drive	22 Mbytes	23	UTn
6352, [1] 6426	Cartridge tape drive	130 Mbytes	23	UTn
6446	Winchester disk drive	230 Mbytes	24	DAn
6491	Winchester disk drive	320 Mbytes	24	DAn
6539	Half-height Winchester disk drive	170 Mbytes	24	DAn
6554	Winchester disk drive	660 Mbytes	24	DAn
6577, [1] 6679	Cartridge tape drive	150 Mbytes	23	UTn
6587	Reel-to-reel tape drive (1600 BPI)	40 Mbytes	23	UTn
6589	Reel-to-reel tape drive (800/1600/6250 BPI)	140 Mbytes	23	UTn
6662	Half-height Winchester disk drive	330 Mbytes	24	DAn
6668 [3]	5.25" 96 TPI diskette drive	737 Kbytes	24	DAn
6675, [1] 6677	Cartridge tape drive	150/320/525 Mbytes	23	UTn
6685	Winchester disk drive	1000 Mbytes	24	DAn

[1] Use 16384/S on BURST and IMOVE command lines.

[3] Use model number 6309 in DKINIT and DSKED dialogs.

Table A-5 MV/7800 XP, MV/9500, MV/15000, and MV/18000 System Storage Devices

Model	Description	Capacity	Device Code	Unit Name
4307	Reel-to-reel tape drive (800/1600/6250 BPI)	140 Mbytes	22,62	MTn
4514	5.25" 48 TPI diskette drive	368 Kbytes	20,60	DJn
6026	Reel-to-reel tape drive (800/1600 BPI)	40 Mbytes	22,62	MTn
6027	Reel-to-reel tape drive (800 BPI)	20 Mbytes	22,62	MTn
6060	Removable disk pack	96 Mbytes	27,67	DZn
6061	Removable disk pack	192 Mbytes	27,67	DZn
6067	Removable disk pack	50 Mbytes	27,67	DZn
6122	Removable disk pack	277 Mbytes	27,67	DZn
6125	Reel-to-reel tape drive (1600 BPI)	20 Mbytes	22,62	MTn
6160	Winchester disk drive	73 Mbytes	27,67	DZn
6161	Winchester disk drive	147 Mbytes	27,67	DZn
6231 [2]	Cartridge tape drive	19 Mbytes	22,62	MTn
6236	Winchester disk drive	354 Mbytes	24,64 25,65	DAn
6239	Winchester disk drive	592 Mbytes	24,64 25,65	DAn
6299, 6300	Reel-to-reel tape drive (1600/6250 BPI)	140 Mbytes	22,62	MTn

[2] Do not use greater than 8192/S on BURST and IMOVE command lines.

(continued)

Table A-5 MV/7800 XP, MV/9500, MV/15000, and MV/18000 System Storage Devices

Model	Description	Capacity	Device Code	Unit Name
6351, [1] 6444	Cartridge tape drive	22 Mbytes	23,63	UTn
6352, [1] 6426	Cartridge tape drive	130 Mbytes	23,63	UTn
6357	Winchester disk drive	862 Mbytes	24,64 25,65	DAn
6446	Winchester disk drive	230 Mbytes	24,64 25,65	DAn
6491	Winchester disk drive	320 Mbytes	24,64 25,65	DAn
6492	Winchester disk drive	720 Mbytes	24,64 25,65	DAn
6554	Winchester disk drive	660 Mbytes	24,64 25,65	DAn
6581	Winchester disk drive	500 Mbytes	24,64 25,65	DAn
6586, 6587	Reel-to-reel tape drive (1600 BPI)	40 Mbytes	23,63	UTn
6588, 6589	Reel-to-reel tape drive (800/1600/6250 BPI)	140 Mbytes	23,63	UTn
6621	Winchester disk drive	1200 Mbytes	24,64 25,65	DAn
6631	Winchester disk drive	600 Mbytes	24,64 25,65	DAn
6656, [1] 6679	Cartridge tape drive	150 Mbytes	23,63	UTn

[1] Use 16384/S on BURST and IMOVE command lines.

(continued)

Table A-5 MV/7800 XP, MV/9500, MV/15000, and MV/18000 System Storage Devices

Model	Description	Capacity	Device Code	Unit Name
6662	Half-height Winchester disk drive	330 Mbytes	24,64 25,65	DAn
6675, [1] 6676	Cartridge tape drive	150/320/525 Mbytes	23,63	UTn
6685, 6740	Winchester disk drive	1000 Mbytes	24,64 25,65	DAn

[1] Use 16384/S on BURST and IMOVE command lines.

(concluded)

-End of Appendix-

Appendix B

Handling New Bad Blocks

You should use DKINIT's TEST command on a regular basis to check for new bad blocks, and certainly if your system reports a disk error or panics, you will want to use that command. If TEST reveals that a new bad block has developed, you must eventually enter that block in your disk's bad block table; however, before you take that step, it is important that you determine what, if anything, that block contains.

To find out how the block is being used, run the BURST utility and issue the OWNER command (see Chapter 13). Once you tell OWNER which disk to look at and the octal block number of the block you are interested in, one of three things will happen. (1) The routine will run to completion, but will not report that the block belongs to a file. (2) OWNER will display the pathname of the file that owns the block and then terminate normally. (3) The program will terminate with an error.

If the first of these possibilities occurs, you are in luck—the new bad block is not being used. In this case, you can simply use DKINIT's ENTER command to log this block in your disk's bad block table and resume normal processing.

In the second case—if OWNER reports the pathname of the file that owns the new bad block and then terminates normally—your problem is only slightly more serious. The fact that the program was able to determine which file the block belongs to and continue running indicates that the block is probably a random file data block or part of a contiguous file. In either situation, you can successfully delete the damaged file. Then use DKINIT's ENTER command to add the bad block to the bad block table and reload the missing file from your most recent logical backup.

If OWNER reports a filename but is unable to run to completion, most likely the bad block you inquired about is part of a random file's index structure or part of a sequential file (although it could also be part of a SYS.DR file). In these cases, you will be unable to delete the damaged file successfully; however, there are a few techniques you can try to make your disk usable.

First, try logging the new bad block in the bad block table using DKINIT's ENTER command. The ENTER routine will attempt to copy the contents of the bad block to the appropriate block in the remap area, and if the copy is successful, you can resume normal operations.

If the copy is not successful, you can try a second technique that involves hiding the damaged file and then reloading the file from your backup media. To hide the file, you first rename it. Then you employ the CLI's CHATR command to make the file permanent—this strategy protects against the file's being deleted. After this, just be careful not to access the damaged file at all. This technique should work unless the new bad block is part of a MAP.DR or SYS.DR file.

As a last resort, you can also try using the disk editor, DSKED, to read the bad block. Since disk blocks often give intermittent errors, it is possible that you will be able to recover the information in the block completely intact. If you can do this, you should then use DSKED to enter the data you recovered into the block in the remap area that corresponds to the block that went bad.

In any case, once you have your disk working, you should perform a full logical (IMOVE) backup immediately. Don't perform a physical (BURST) backup at this point because a physical backup is highly dependent on a disk's reliability and your disk is not reliable. Then hardware and software format your disk. Sometimes this formatting will appear to make some of your disk's bad blocks disappear; however, often the same bad blocks will reappear later. Therefore, it is a good idea to keep a list of all the blocks that have ever gone bad on your disk and to declare those blocks bad even if DKINIT does not discover them. To be completely safe, you should also check to see if there is a list of unreliable blocks on the side of your disk. If there is such a list, declare those blocks to be bad as well. At this point, you can restore the contents of your disk from your IMOVE backup set and resume normal processing.

-End of Appendix-

Index

Symbols

Numbers

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

X

Related Manuals

If you are using DG/RDOS for the first time, you should read *Introduction to RDOS* before reading this manual. The other manuals in the list below are partners to *How to Generate and Run DG/RDOS* in that they treat aspects of using the system not covered in this manual.

- *Introduction to RDOS* (069-400011) describes the fundamentals of using RDOS and summarizes the features, utilities, and capabilities of the operating system.
- *RDOS/DOS Superedit Text Editor* (069-400017) introduces the commands and concepts of the Superedit Text Editor (SPEED), which offers many powerful features for editing text.
- *RDOS/DOS Assembly Language and Program Utilities* (069-400019) discusses the Extended Assembler (ASM), Macroassembler (MAC), Extended Relocatable Loader (RLDR and OVLDR), and Library File Editor (LFE) utilities that aid in programming.
- *RDOS/DOS Debugging Utilities* (069-400020) describes five utilities that assist you in editing, debugging, and patching programs—the Symbolic Editor (SEDIT), the Symbolic Debugger (DEBUG), the Disk Editor (DSKED), and the patch utilities (ENPAT and PATCH).
- *Using the DG/RDOS Command Line Interpreter* (093-000471) discusses the user interface with the operating system. It covers the features and command mechanisms of the Command Line Interpreter (CLI) and presents instructions on how to use CLI commands.
- *RDOS System Reference* (093-400027) describes RDOS system features, system calls, and user device driver implementation for assembly language and high-level language programming.

TIPS ORDERING PROCEDURES

TO ORDER

1. An order can be placed with the TIPS group in two ways:
   a) MAIL ORDER - Use the order form on the opposite page and fill in all requested information. Be sure to include shipping charges and local sales tax. If applicable, write in your tax exempt number in the space provided on the order form.

      Send your order form with payment to:
      Data General Corporation
      ATTN: Educational Services/TIPS G155
      4400 Computer Drive
      Westboro, MA 01581-9973

   b) TELEPHONE - Call TIPS at (508) 870-1600 for all orders that will be charged by credit card or paid for by purchase orders over $50.00. Operators are available from 8:30 AM to 5:00 PM EST.

METHOD OF PAYMENT

2. As a customer, you have several payment options:
   a) Purchase Order - Minimum of $50. If ordering by mail, a hard copy of the purchase order must accompany order.
   b) Check or Money Order - Make payable to Data General Corporation.
   c) Credit Card - A minimum order of $20 is required for Mastercard or Visa orders.

SHIPPING

3. To determine the charge for UPS shipping and handling, check the total quantity of units in your order and refer to the following chart:

Total Quantity	Shipping & Handling Charge
1-4 Units	$5.00
5-10 Units	$8.00
11-40 Units	$10.00
41-200 Units	$30.00
Over 200 Units	$100.00

   If overnight or second day shipment is desired, this information should be indicated on the order form. A separate charge will be determined at time of shipment and added to your bill.

VOLUME DISCOUNTS

4. The TIPS discount schedule is based upon the total value of the order.

Order Amount	Discount
$1-$149.99	0%
$150-$499.99	10%
Over $500	20%

TERMS AND CONDITIONS

5. Read the TIPS terms and conditions on the reverse side of the order form carefully. These must be adhered to at all times.

DELIVERY

6. Allow at least two weeks for delivery.

RETURNS

7. Items ordered through the TIPS catalog may not be returned for credit.
8. Order discrepancies must be reported within 15 days of shipment date. Contact your TIPS Administrator at (508) 870-1600 to notify the TIPS department of any problems.

INTERNATIONAL ORDERS

9. Customers outside of the United States must obtain documentation from their local Data General Subsidiary or Representative. Any TIPS orders received by Data General U.S. Headquarters will be forwarded to the appropriate DG Subsidiary or Representative for processing.

TIPS ORDER FORM

Mail To: Data General Corporation
Attn: Educational Services/TIPS G155
4400 Computer Drive
Westboro, MA 01581 - 9973

BILL TO:	SHIP TO: (No P.O. Boxes - Complete Only If Different Address)
COMPANY NAME	COMPANY NAME
ATTN:	ATTN:
ADDRESS	ADDRESS (NO PO BOXES)
CITY	CITY
STATE ZIP	STATE ZIP

Priority Code _____ (See label on back of catalog)

Authorized Signature of Buyer (Agrees to terms & conditions on reverse side) | Title | Date | Phone (Area Code) | Ext.

ORDER #	QTY	DESCRIPTION	UNIT PRICE	TOTAL PRICE

A SHIPPING & HANDLING

☐ UPS	ADD
1-4 Items	$ 5.00
5-10 Items	$ 8.00
11-40 Items	$ 10.00
41-200 Items	$ 30.00
200+ Items	$100.00

Check for faster delivery

Additional charge to be determined at time of shipment and added to your bill.

☐ UPS Blue Label (2 day shipping)
☐ Red Label (overnight shipping)

B VOLUME DISCOUNTS

Order Amount	Save
$0 - $149.99	0%
$150 - $499.99	10%
Over $500.00	20%

Tax Exempt # or Sales Tax (if applicable) _____

ORDER TOTAL	
Less Discount See B	-
SUB TOTAL	
Your local* sales tax	+
Shipping and handling - See A	+
TOTAL - See C	

C PAYMENT METHOD

☐ Purchase Order Attached ($50 minimum)
P.O. number is _____. (Include hardcopy P.O.)
☐ Check or Money Order Enclosed
☐ Visa ☐ MasterCard ($20 minimum on credit cards)

Account Number | Expiration Date

Authorized Signature
(Credit card orders without signature and expiration date cannot be processed.)

THANK YOU FOR YOUR ORDER

PRICES SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.
PLEASE ALLOW 2 WEEKS FOR DELIVERY.
NO REFUNDS NO RETURNS.

* Data General is required by law to collect applicable sales or use tax on all purchases shipped to states where DG maintains a place of business, which covers all 50 states. Please include your local taxes when determining the total value of your order. If you are uncertain about the correct tax amount, please call 508-870-1600.

Form 702
Rev. 8/87

DATA GENERAL CORPORATION
TECHNICAL INFORMATION AND PUBLICATIONS SERVICE
TERMS AND CONDITIONS

Data General Corporation ("DGC") provides its Technical Information and Publications Service (TIPS) solely in accordance with the following terms and conditions and more specifically to the Customer signing the Educational Services TIPS Order Form. These terms and conditions apply to all orders, telephone, telex, or mail. By accepting these products the Customer accepts and agrees to be bound by these terms and conditions.

1. CUSTOMER CERTIFICATION

Customer hereby certifies that it is the owner or lessee of the DGC equipment and/or licensee/sub-licensee of the software which is the subject matter of the publication(s) ordered hereunder.

2. TAXES

Customer shall be responsible for all taxes, including taxes paid or payable by DGC for products or services supplied under this Agreement, exclusive of taxes based on DGC's net income, unless Customer provides written proof of exemption.

3. DATA AND PROPRIETARY RIGHTS

Portions of the publications and materials supplied under this Agreement are proprietary and will be so marked. Customer shall abide by such markings. DGC retains for itself exclusively all proprietary rights (including manufacturing rights) in and to all designs, engineering details and other data pertaining to the products described in such publication. Licensed software materials are provided pursuant to the terms and conditions of the Program License Agreement (PLA) between the Customer and DGC and such PLA is made a part of and incorporated into this Agreement by reference. A copyright notice on any data by itself does not constitute or evidence a publication or public disclosure.

4. LIMITED MEDIA WARRANTY

DGC warrants the CLI Macros media, provided by DGC to the Customer under this Agreement, against physical defects for a period of ninety (90) days from the date of shipment by DGC. DGC will replace defective media at no charge to you, provided it is returned postage prepaid to DGC within the ninety (90) day warranty period. This shall be your exclusive remedy and DGC's sole obligation and liability for defective media. This limited media warranty does not apply if the media has been damaged by accident, abuse or misuse.

5. DISCLAIMER OF WARRANTY

EXCEPT FOR THE LIMITED MEDIA WARRANTY NOTED ABOVE, DGC MAKES NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR PARTICULAR PURPOSE ON ANY OF THE PUBLICATIONS, CLI MACROS OR MATERIALS SUPPLIED HEREUNDER.

6. LIMITATION OF LIABILITY

A. CUSTOMER AGREES THAT DGC'S LIABILITY, IF ANY, FOR DAMAGES, INCLUDING BUT NOT LIMITED TO LIABILITY ARISING OUT OF CONTRACT, NEGLIGENCE, STRICT LIABILITY IN TORT OR WARRANTY SHALL NOT EXCEED THE CHARGES PAID BY CUSTOMER FOR THE PARTICULAR PUBLICATION OR CLI MACRO INVOLVED. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO CLAIMS FOR PERSONAL INJURY CAUSED SOLELY BY DGC'S NEGLIGENCE. OTHER THAN THE CHARGES REFERENCED HEREIN, IN NO EVENT SHALL DGC BE LIABLE FOR ANY INCIDENTAL, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES WHATSOEVER, INCLUDING BUT NOT LIMITED TO LOST PROFITS AND DAMAGES RESULTING FROM LOSS OF USE, OR LOST DATA, OR DELIVERY DELAYS, EVEN IF DGC HAS BEEN ADVISED, KNEW OR SHOULD HAVE KNOWN OF THE POSSIBILITY THEREOF; OR FOR ANY CLAIM BY ANY THIRD PARTY.

B. ANY ACTION AGAINST DGC MUST BE COMMENCED WITHIN ONE (1) YEAR AFTER THE CAUSE OF ACTION ACCRUES.

7. GENERAL

A valid contract binding upon DGC will come into being only at the time of DGC's acceptance of the referenced Educational Services Order Form. Such contract is governed by the laws of the Commonwealth of Massachusetts, excluding its conflict of law rules. Such contract is not assignable. These terms and conditions constitute the entire agreement between the parties with respect to the subject matter hereof and supersedes all prior oral or written communications, agreements and understandings. These terms and conditions shall prevail notwithstanding any different, conflicting or additional terms and conditions which may appear on any order submitted by Customer. DGC hereby rejects all such different, conflicting, or additional terms.

8. IMPORTANT NOTICE REGARDING AOS/VS INTERNALS SERIES (ORDER #1865 & #1875)

Customer understands that information and material presented in the AOS/VS Internals Series documents may be specific to a particular revision of the product. Consequently user programs or systems based on this information and material may be revision-locked and may not function properly with prior or future revisions of the product. Therefore, Data General makes no representations as to the utility of this information and material beyond the current revision level which is the subject of the manual. Any use thereof by you or your company is at your own risk. Data General disclaims any liability arising from any such use and I and my company (Customer) hold Data General completely harmless therefrom.

How to Generate and Run DG/RDOS

093-000470-03

Cut here and insert in binder spine pocket

Data General

Data General Corporation, Westboro, Massachusetts 01580