DX10 Operating System Operations Guide

Volume II

Part No. 946250-9702 *D
1 September 1982

TEXAS INSTRUMENTS

MANUAL REVISION HISTORY

DX10 Operating System Operations Guide (Volume II) (946250-9702)

Original Issue	15 August 1977
Revision	15 October 1978
Revision	15 December 1979
Revision	15 April 1981
Change 1	1 October 1981
Revision	1 September 1982

The total number of pages in this publication is 550.

Preface

This manual provides information on how to initialize the DX10 operating system, use the terminal, maintain DX10, and use the System Command Interpreter (SCI). It is intended to guide the first-time user from first contact with the terminal to operating the DX10 system. You can use the manual as a guide to terminal operation. You can also find detailed information about maintaining your DX10 system. Additionally, the manual contains an introduction to SCI, your interface with DX10. Then, it describes the functional groups of the SCI commands and presents a convenient guide to the commands by describing them in alphabetical order. (The Text Editor and Link Editor commands are not included in this manual, but can be found in their respective manuals. Debugger commands are in Volume III.)

This manual is one of a set of six volumes that describe the operational characteristics and features of DX10. In addition to the six volumes, several support manuals are available for DX10 functions. Also, each language supported by DX10 has its own associated manuals.

Become acquainted with this manual and related manuals in the DX10 family of manuals in order to initialize DX10, use the terminal, maintain DX10, and use the SCI commands. The following paragraphs contain a brief comment regarding the contents of the other volumes in the set. (The full titles and part numbers of all manuals associated with the DX10 operating system are provided in a list at the end of this preface.)

Concepts and Facilities (Volume I) includes features, concepts, and general background information describing the DX10 operating system. It also contains a master subject index to help you find the information you need.

The *Application Programming Guide* (Volume III) contains information required by the application programmer to prepare, modify and execute application programs on DX10. Much of the material is relevant to both high-level language programmers as well as assembly language programmers, since it concerns program structure, program operation, file structure and file I/O. The SCI programming language is included, since it is a major part of constructing applications under DX10. Complete descriptions for nonprivileged SVCs and the DX10 Debugger are included for assembly language programs.

The *Text Editor Manual* (Volume IV) includes operating instructions, examples, and exercises for the interactive Text Editor provided on DX10. The SCI commands and error messages related to the Text Editor are included.

The *Systems Programming Guide* (Volume V) includes information required by the system programmer to maintain or extend a computer system running under DX10. The disk build procedure required for building your initial system disk, and the system generation procedure and troubleshooting guide required for system start-up are located in this manual. Also included is support of nonstandard devices, and privileged SVCs available on DX10.

The *Error Reporting and Recovery Manual* (Volume VI) describes each error message you can receive while operating DX10, and gives suggested procedures for recovery. It documents task errors, command errors, SVC errors, SCI errors, magnetic tape and other I/O errors. Also included are sections on system crash analysis, and system troubleshooting.

NOTE

Additional, in-depth descriptions related to specific languages including FORTRAN, COBOL, BASIC, RPG II, TI Pascal, Assembly Language, and Query are found in manuals dedicated to the appropriate programming language. A Link Editor manual is provided as a separate volume that describes the application of the link edit function in a DX10 environment. Separate manuals describe the use of an optional Sort/Merge package and the DBMS package.

The following documents contain information related to the DX10 operating system:

Title	Part Number
DX10 Manuals	
DX10 Operating System Concepts and Facilities (Volume I)	946250-9701
DX10 Operating System Application Programming Guide (Volume III)	946250-9703
DX10 Operating System Text Editor Manual (Volume IV)	946250-9704
DX10 Operating System Systems Programming Guide (Volume V)	946250-9705
DX10 Operating System Error Reporting and Recovery Manual (Volume VI)	946250-9706
DX10 Operating System Release 3.5 System Design Document	939153-9701
Link Editor Reference Manual	949617-9701
Language Manuals	
990/99000 Assembly Language Reference Manual	2270509-9701
Model 990 Computer DX10 Operating System COBOL Programmer's Guide	2270521-9701

Title	Part Number
TIBOL Programmer's Guide	2263354-9701
Model 990 Computer FORTRAN Programmer's Reference Manual	946260-9701
Model 990 Computer Report Program Generator (RPG II) Programmer's Guide	939524-9701
Model 990 Computer TI 990 BASIC Reference Manual	2250304-9701
Model 990 Computer DX10 TI Pascal Programmer's Guide	2270528-9701
TI Pascal Reference Manual	2270519-9701
Related Software Manuals	
Model 990 Computer DX10 Data Base Management System Programmer's Guide	2250425-9701
Model 990 Computer Data Base Administrator User's Guide	2250426-9701
Operator's Guide Business System 300 (International)	2533318-9701
DX10 3270 Interactive Communications Software (ICS) User's Guide	2250954-9701
Model 990 Computer DX10 Remote Terminal Subsystem (RTS) System Generation and Programmer's Reference Manual	2272054-9701
Model 990 Computer DX10 Remote Terminal Subsystem (RTS) Operator's Guide	2272055-9701
Model 990 Computer DX 3780/2780 Emulator User's Guide	946289-9701
Model 990 Computer DX10 Poller Operations Manual	2302679-9701
Model 990 Computer DX10 HDLC Communications Package User's Guide	2270526-9701
Model 990 Computer TIFORM User's Guide	2250374-9701
Model 990 Computer DX10 Query-990 User's Guide	2250466-9701

Title	Part Number
Model 990 Computer Sort/Merge User's Guide	946252-9701
Model 990 Computer DX Sort/Merge Programmer's Reference Card	946253-9701
Related Hardware Manuals	
Model 990 Computer Model 911 Video Display Terminal Installation and Operation	945423-9701
Model 990 Computer Model 913 CRT Display Terminal Installation and Operation	943457-9701
Model 990 Computer Model 940 Electronic Video Terminal (EVT) Installation and Operation Manual	2250368-9701
Model 820 KSR Terminal Operator's Manual	2208225-9701
Model 781 RO Terminal Operator's Manual	2265935-9701
Model 783 KSR Terminal Operator's Manual	2265936-9701
Model 785 Communications Terminal Operator's Manual	2265937-9701
Model 787 Communications Terminal Operator's Manual	2265938-9701
Model 743 KSR Terminal Operator's Manual	984030-9701
Model 745 Portable Terminal Operator's Manual	984024-9701
Models 763/765 Operating Instructions	2203664-9701
Models 763/765 Memory Terminals Systems Manual	2203665-9701
Universal ROM Loader User's Guide	2270534-9701

Title	Part Number
Model 990/10 Computer System Hardware Reference Manual	945417-9701
Model 990 Computer Remote Terminal Subsystem (RTS) Hardware Installation Manual (European Edition)	2272053-9701
Model 990 Computer Models 306 and 588 Line Printers Installation and Operation	945261-9701
Model 990 Computer PROM Programming Module Installation and Operation	945258-9701
Model 990 Computer Communications System Installation and Operation	945409-9701
Model 990 Computer Communications System Software	946236-9701
Model 990 Computer Terminal Executive Development System (TXDS) Programmer's Guide	946258-9701
Model 990 Computer Model FD800 Floppy Disc System with International Chassis Installation and Operation Manual	2250697-9701
Model 990 Computer Model FD1000 Flexible Disk System with International Chassis Installation and Operation	2250698-9701
Model 990 Computer Model 733 ASR/KSR Data Terminal Installation and Operation	945259-9701
Model 990 Computer Model 804 Card Reader Installation and Operation	945262-9701

Contents

4 — Operating DX10 from a Terminal

5 — Maintaining Your System

8 — DX10 SCI Commands

Appendixes

Index

Illustrations

Tables

1

DX10 General Description

1.1 INTRODUCTION

DX10 is a general-purpose, multitasking operating system designed to operate with the Texas Instruments 990/10, 990/12, and 990/10A minicomputers using the memory mapping feature. DX10 is a versatile disk-based operating system capable of supporting a wide range of commercial and industrial applications. It features a powerful file management package, which includes support for key indexed files (KIFs). DX10 is interactively oriented, although a batch mode is also supported. It is also a multiterminal system capable of making each of several users appear to have exclusive control of the system. DX10 can also be configured during system generation to support specialized real-time applications.

In addition to providing multiterminal application support, DX10 features advanced program development support. A text editor is provided to enter source programs or data into the system. A macro assembler is provided for assembly language programs. Higher-level language support includes FORTRAN, COBOL, RPG II, Pascal, and BASIC. A link editor and extended debugging facilities are provided to further support program development. A variety of utility programs, a data base management package, and a comprehensive sort/merge package are supported.

DX10 is an international operating system designed to meet the commercial requirements of the United States, most European countries, and Japan. DX10 supports a complete range of international data terminals that permit users to enter, view, and process data in their own language. These terminals are currently available for the following countries: Austria, Belgium, Denmark, Finland, France, Germany, Japan, Norway, Sweden, Switzerland, Spain, Mexico, the United Kingdom, and the United States.

DX10 supports three types of files: sequential, relative record, and key indexed. Refer to Volume III for more information on file types.

DX10 supports a variety of display terminals and hard-copy devices. For more information on these devices and other hardware, refer to Section 2 of this manual and Volume I.

Volume I consists of two parts: concepts and facilities. The concepts part introduces readers of all levels to DX10; the facilities part introduces the more technically oriented features of DX10. Volume I also contains a glossary, a word index, and the master subject index. Volume III discusses program development and debugging. Volume IV explains the Text Editor. Volume V gives information on the maintenance and extension of DX10. Volume VI functions as a reference manual to error messages and as a troubleshooting guide to identify and correct error conditions.

1.2 SYSTEM COMMAND INTERPRETER (SCI) CAPABILITIES

SCI is a powerful software package that comes with the DX10 operating system. It is the means by which you, the user, communicate with the DX10 operating system. Volume III details the SCI programming language.

SCI has several capabilities. You can use it to communicate interactively (that is, you and DX10 have an exchange similar to a conversation). You can use it as a programming language to perform a batch of operations that you would not want to perform by typing in each command at a terminal. You can also use the SCI programming language to create new SCI commands that you can execute interactively or in a batch of operations.

1.3 GUIDE TO USING THIS MANUAL

This manual contains eight sections. Section 1 gives an introduction and defines a few terms. Section 2 describes some of the characteristics for the various hardware supported. Section 3 describes what you should do if you want to activate a DX10 system. It describes how to initialize the system and how to shut it down.

Section 4 explains how you can activate SCI from any terminal and begin interacting with DX10 using SCI commands. Section 5 discusses general operating procedures such as backing up the system and changing disk volumes.

Section 6 provides a detailed description of SCI and presents a table of all of the SCI commands in the DX10 software package. Section 7 discusses the SCI commands in functional groups. Section 8 describes each of the SCI commands in alphabetical order.

1.4 DEFINITION OF TERMS

The following paragraphs describe some of the important terms and concepts referred to throughout this manual.

1.4.1 Default Values

SCI often provides default values to speed up the entry of responses to prompting messages. This is especially true for optional user responses. A default value is a value that SCI automatically supplies in response to a prompting message if there is no initial value. To enter the default value for a prompting message (where a default value exists), you should press the RETURN key without entering any other data. Such an entry is called a null response.

NOTE

Throughout this manual, the key descriptions apply to the Model 911 VDT. Refer to Table 4-2 for the equivalent keys on other keyboards.

1.4.2 Initial Values

An initial value is a value that SCI automatically displays as a response to some prompting messages. You can accept the initial value by pressing the RETURN key, or you can erase the initial value by pressing the ERASE FIELD key. You can reject the initial value by entering a different value.

The initial values for some prompting messages are fixed, and the same value always appears as the initial value for that prompting message. In other cases, SCI saves values entered with one command for later display as initial values for a later entry of the same command or for the entry of a related command. SCI also saves some variable initial values from one terminal session to another.

1.4.3 Display of Error Messages

When an error occurs, DX10 ensures that you see the error message by displaying the message at the bottom line of the screen (or the next print line of the TTY device) and by inhibiting further operation until you acknowledge the message by pressing the CMD key or RETURN key.

1.4.4 Log-On Control at Terminals

You can control log-on at terminals by modifying the Initialize System (IS) command to require log-on. The IS command initially defines station parameters (assigns characteristics to terminals) within the system. You can assign a parameter that requires users to log on at predefined terminals. When log-on is required, you must enter a user ID and a passcode before operating the terminal.

1.4.5 Message Handling

You can create a message at a terminal by using the Create Message (CM) command and send that message to any terminal in the system. The message appears at the destination terminal when SCI determines that it is available for receipt of the message. You can also use the MSG (message) primitive for displaying messages.

1.4.6 News File

The news file is a sequential file that you can use to notify system users of changes to or problems with the system. You can also use it to provide any information that requires display to all system users. SCI displays the news file immediately after you log on at a terminal that requires log-on or immediately after you bid SCI at a terminal that does not require log-on. (SCI displays the news file only if the terminal is in VDT mode. See Section 4 for a description of VDT mode.) When you press the command (CMD) key, SCI displays the top-level menu. The news file has the special name .S$NEWS.

1.4.7 Privileged Commands

Privileged commands are commands limited by privilege level. You cannot enter some commands if the privilege level of your user ID or terminal is not high enough. The privilege level is associated with a user ID if you are operating at a terminal that requires log-on, or the privilege level is associated with the terminal itself if you are operating at a terminal that does not require log-on.

1.4.8 Suspend SCI (I/O-Only Terminals)

You can implement terminals under DX10 strictly as I/O devices that do not interact with SCI. SCI executes as a task in DX10 and is bid for each terminal you want to use interactively.

You can terminate SCI by entering the Quit (Q) command. DX10 also automatically suspends SCI when you enter an Execute Task and Suspend (XTS) command. The suspend function of the XTS command prevents SCI from interfering with input that is destined for an interactive program.

1.4.9 Synonyms

DX10 allows you to define a string of characters to substitute for another string of characters. The substitute string is referred to as a synonym and is usually shorter than the string it represents. (Typically, a synonym stands for the first part of a pathname.) This saves keystrokes whenever you enter the synonym in place of the value it represents. At terminals that require log-on, the synonyms that you define are associated with your user ID and are available whenever you log on any terminal. Note that the synonyms are saved when you log off. If the DX10 operating system is shut down before you log off, the synonyms you created since log-on are not saved. Synonyms that you define at terminals that do not require log-on are lost when you terminate SCI at the terminal.

These synonyms are always available to foreground tasks. Background tasks are given a copy of the foreground synonyms at the time the background task is initiated. However, synonyms assigned in background are not available in foreground, and any synonyms defined in foreground after the copy is made for the background task are not passed to the background task. Refer to Volume V for detailed information about synonyms.

1.4.10 Terminal Local File (TLF)

The TLF is the default output file to which SCI sends the results of an operation if you specify no other file or device as the destination of output. The TLF is associated with SCI when you activate SCI at the terminal you are using. The TLF is automatically displayed upon the completion of each command execution. If the output takes up more than one display screen when the terminal is in VDT mode, you can examine the other screens by pressing the roll up (F1) key or the roll down (F2) key. If the terminal is in TTY mode, the entire file is displayed in an automatic scrolling mode until the end of the file is reached.

User programs should never directly access the TLF. They should use S$ routine interfaces (refer to the SCI description in Volume III).

1.4.11 Other Significant Terms

You should refer to the glossary in Volume I for the definition of other terminology that you discover when reading this manual.

2

Introduction to Terminal Operation

2.1 GENERAL

DX10 supports a wide variety of display and hard-copy terminals. This section explains basic terminal operation. It presents pictures of the display terminals and shows you the controls for power to the terminal, brightness of the display, and volume of the audible alarm. It also describes the position at which you can enter data.

2.2 DISPLAY TERMINAL CHARACTERISTICS

DX10 supports the following display terminals:

- Model 911 Video Display Terminal (VDT)
- Model 913 VDT
- Model 940 Electronic Video Terminal (EVT)
- Business System Terminal

The 911 and 913 VDTs, 940 EVT, and Business System Terminal display information on cathode-ray tube (CRT) display screens. The terminals display information that you enter from the keyboard and information that the computer returns.

The video display of 80 characters in a line includes a marker that indicates the current position within a line where you can enter data. The marker fills that character position with a bright square of light. This marker is referred to as a cursor. When you type a character, the cursor automatically advances one character position. You can set the cursor to any position in any displayed line by moving the cursor backward, forward, up, or down. You can accomplish cursor movement by pressing the appropriate cursor movement keys. (Section 4 describes the cursor movement keys. All key descriptions in this manual apply to the 911 VDT keyboard. If you are using a different type of terminal, you can refer to the equivalency table in Section 4 to see which key to use on your keyboard.)

You can enter a command at the terminal by pressing the keys that correspond to the characters of the command and then pressing the RETURN key to signal DX10 that you entered the command. DX10 displays an error message at the terminal if a command is unacceptable.

NOTE

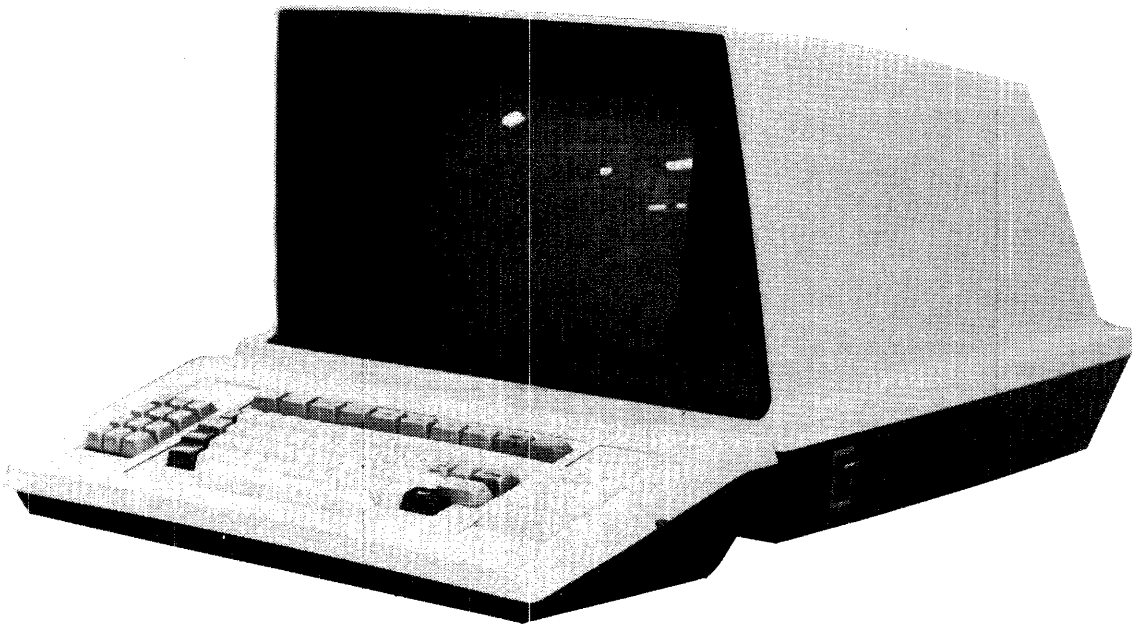
The terminals allow for uppercase and lowercase characters. However, you must enter all SCI commands and SCI field prompt responses in English in uppercase characters unless the lowercase-to-uppercase mapping option is specified by the .OPTION primitive in the M$00 procedure. The .OPTION primitive in the M$00 procedure optionally enables lowercase-to-uppercase mapping for each terminal. You must specify the .OPTION primitive for each terminal individually.

2.2.1 Model 911 VDT Characteristics

Figure 2-1 shows the video display unit (VDU) and keyboard that comprise the 911 VDT.

Figure 2-2 shows the control panel on the side of the VDU. You can control the power to the terminal, the brightness of the display screen, and the volume of the audible alarm using the controls.

For more information describing the 911 VDT, refer to the *Model 990 Computer Model 911 Video Display Terminal Installation and Operation Manual.*



2283038

Figure 2-1. Model 911 VDT

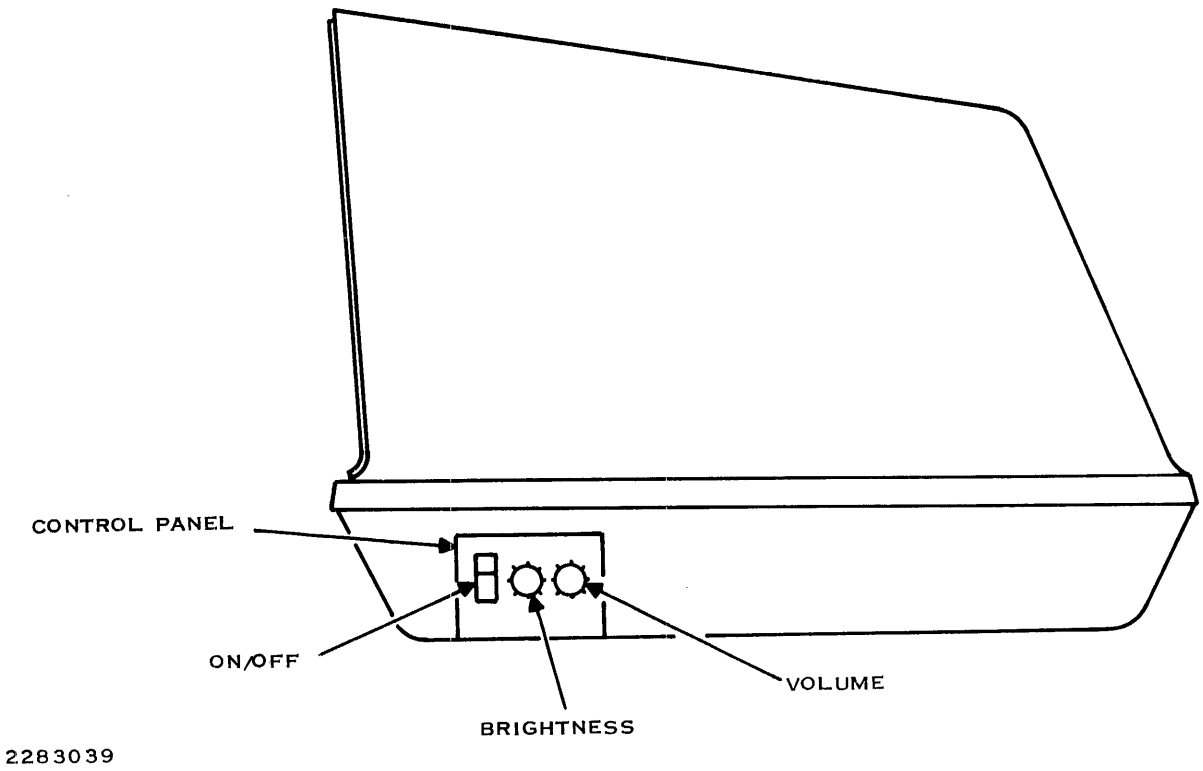


Figure 2-2. Model 911 VDT Controls

2.2.2 Model 913 VDT Characteristics

Figure 2-3 shows the VDU and attached keyboard that comprise the 913 VDT.

Figure 2-4 shows the controls on the 913 VDT. You can control brightness, contrast, and the volume of the audible alarm using the controls. The vertical hold and horizontal hold controls are on the back of the display cabinet directly behind the brightness and contrast controls.

For more information describing the 913 VDT, refer to the *Model 990 Computer Model 913 CRT Display Terminal Installation and Operation Manual.*

2.2.3 Model 940 EVT Characteristics

Figure 2-5 shows the VDU and attached keyboard that comprise the 940 EVT. Special key sequences perform the controls on the 940 EVT.

For more information describing the 940 EVT, refer to the *Model 990 Computer Model 940 Electronic Video Terminal (EVT) Installation and Operation Manual.*

2.2.4 Business System Terminal Characteristics

Figure 2-6 shows the VDU and attached keyboard that comprise the Business System Terminal. Special key sequences perform the controls on the Business System Terminal.

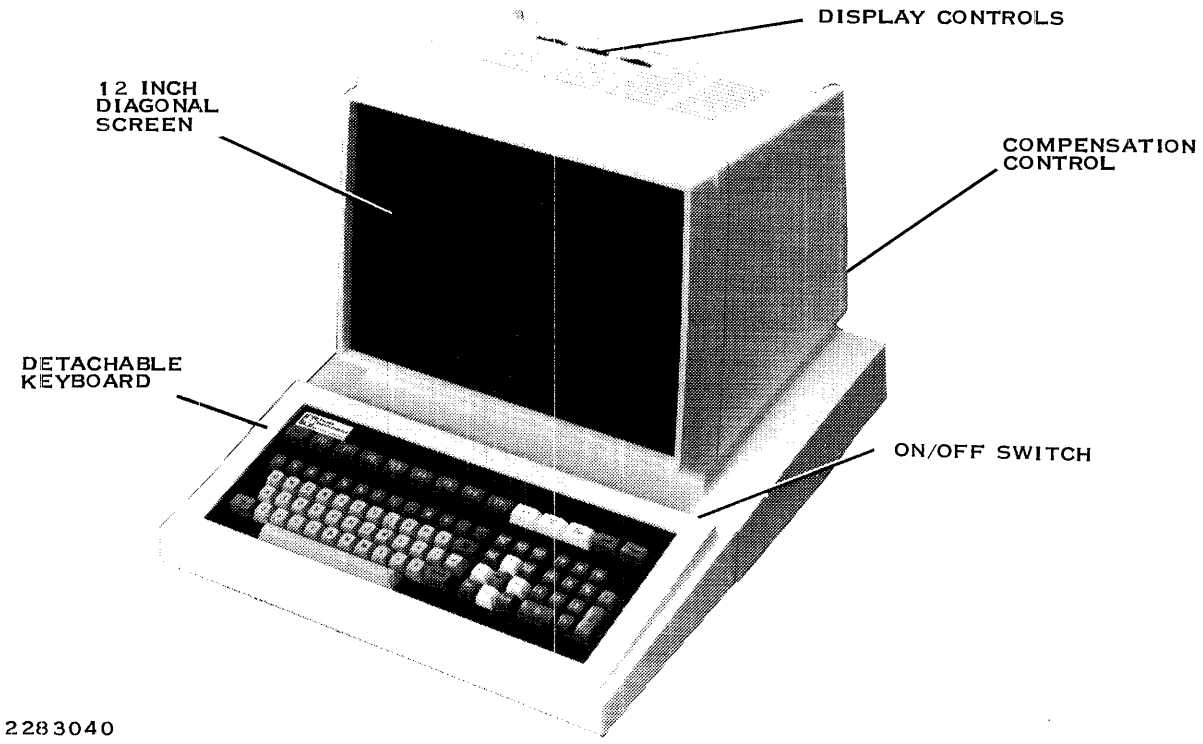


Figure 2-3. Model 913 VDT

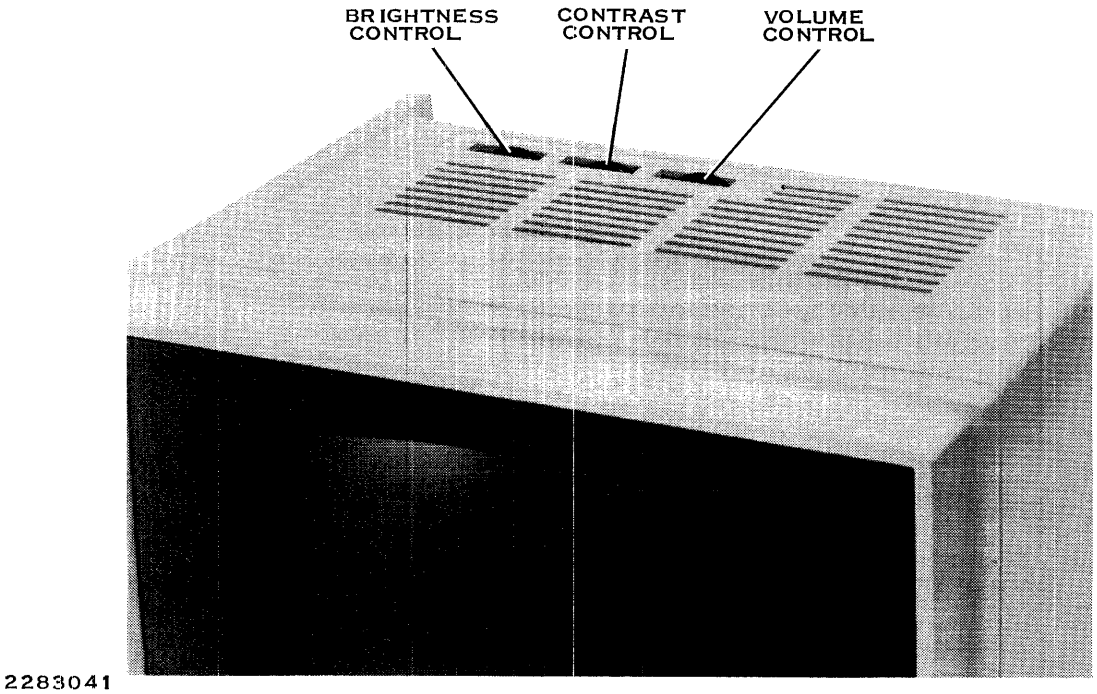
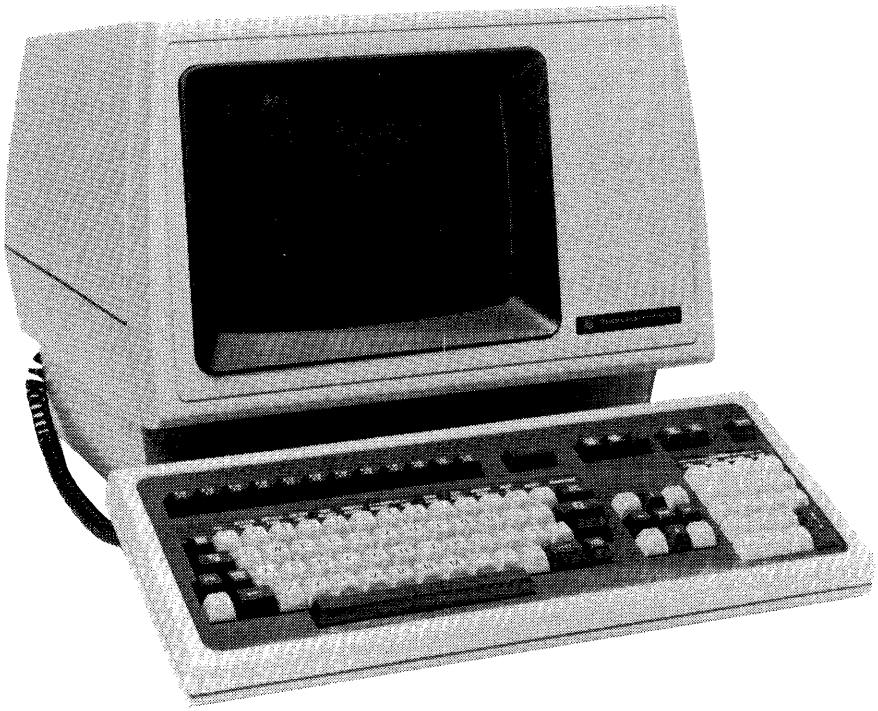
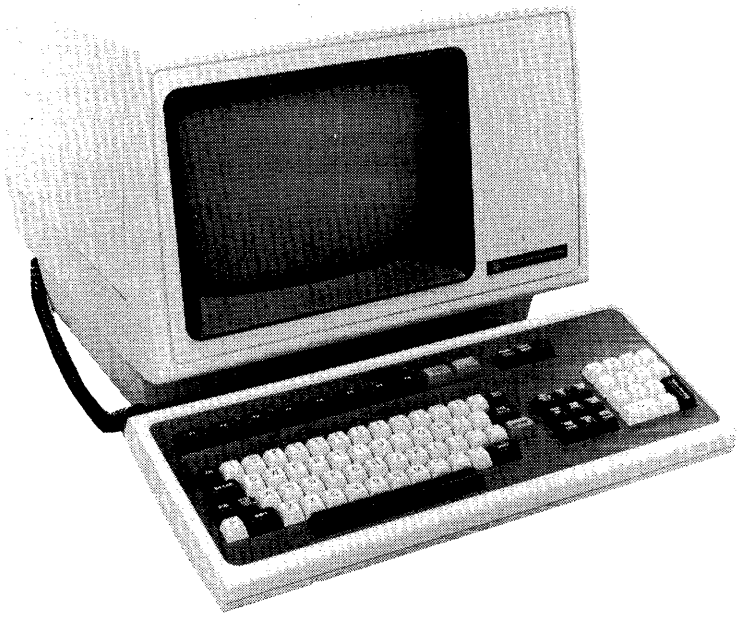


Figure 2-4. Model 913 VDT Controls



2280167

Figure 2-5. Model 940 EVT



2283093

Figure 2-6. Business System Terminal

2.3 TELEPRINTER DEVICES

DX10 supports a variety of hard-copy devices. These devices, which only work in TTY mode (discussed in Section 4), are as follows:

- Model 733 ASR/KSR
- Models 743/745 KSR
- Model 820 KSR
- Model 840 RO
- Models 783/785/787 KSRs
- Models 763/765 Bubble Memory Terminals

The teleprinter devices are hard-copy data terminals that print information on paper that is roll fed through the terminal. These terminals print information that you enter and information that the computer returns.

The position of the printhead on these data terminals marks the current position where you can enter data. When you type a character, the printhead advances one character position. You can move the printhead backward by pressing the CTRL and H keys simultaneously.

You can enter a command at data terminals by pressing the keys that correspond to the characters of the command and then pressing the RETURN key. (The keys that this manual describes apply only to the 911 VDT. Refer to Table 4-2 for equivalent keys on other keyboards.) DX10 prints an error message if a command is unacceptable.

3

System Initialization

3.1 WHAT SYSTEM INITIALIZATION IS

This section describes how to initialize (activate) a DX10 operating system. If DX10 is currently operating (that is, other terminals are active), then your system is already initialized and you should proceed to Section 4 for information on operating DX10 from a terminal.

In order for the DX10 operating system to function properly, all files comprising the system must reside on one disk in an online computer system. This disk can be either a fixed disk, such as the five-megabyte Winchester disk, or a removable disk, such as the DS10 cartridge or DS300 disk pack. This disk is called the system disk. Whenever you want to bring the DX10 system online, you must load the DX10 operating system into the computer memory by performing an initial program load (IPL) procedure.

If you do not yet have a system disk, that is, if your DX10 operating system software was shipped to you on flexible diskettes or tape, you must build a system disk before loading the DX10 operating system. The system disk build procedure is the first thing you should do when you receive your DX10 operating system on diskette or tape. If this has not been done, refer to Volume V for details on how to perform the disk build procedure.

The IPL procedure initiates a loader program that reads DX10 from the system disk (called loading). In most modern computer systems, the initial loader program is in read-only memory (ROM), which is composed of integrated circuits on one of the printed circuit boards in the system hardware. The initial loader program is a very short program that reads a more complex secondary loader program from an available load device, usually disk media. (However, the secondary loader program can be read from punched cards or tape if a card reader or tape device is attached to your computer system.)

It is the secondary loader program that ultimately reads the operating system into dynamic memory. Every time you perform an IPL you are loading in a new copy of the operating system into memory, and basically you are starting over. Any information left in memory, such as a program in the execution stage, is not saved. Information presented later in this section discusses how to protect hardware and data before performing an IPL.

3.2 INITIALIZING DX10 (THE IPL PROCEDURE)

The following paragraphs describe the process required to initialize the DX10 operating system. This is called the IPL procedure. First, you need to power up the computer and turn on your terminal. Also, you need to turn on any other I/O devices such as printers that you are going to use. Next, you should load DX10 from the media on which it is stored (usually a disk). There are different procedures to do this depending on whether the secondary loader program resides on the system disk or on other media. The following paragraphs also contain information about what you should do if a fault or system crash occurs during IPL. Finally, you need to execute the Initialize System (IS) command to set the current date and time and to initialize the system log.

3.2.1 Hardware Power-Up

First make sure that the hardware was tested during installation if your equipment is new and this is your first IPL. The order that you power up the various equipment is important.

1. Turn on power to the computer.

2. Perform the following steps to power up the system disk:

   a. Power up the system disk drive.

   b. Insert the system disk if it is not a fixed disk (that is, a Winchester).

   c. Make the disk spin so you can perform an IPL or read from the disk. Refer to the appropriate manual for the operating procedures for the type of disk you are using.

3. Follow these steps to power up the printer(s) that you are going to use:

   a. Power up the printer.

   b. Make sure you have paper in the printer.

   c. Place the printer online. (Refer to the appropriate manual for the type of printer you are using.)

4. Turn on the power to any terminals that you are going to use. If you are powering up from the base DX10 operating system shipped to you, it is important to know if your configuration differs from the base configuration (see your system manager). You need to know which terminals match your configuration since you cannot use the base DX10 from a terminal that does not match the base configuration.

5. Power up any other special devices that you are going to use.

3.2.2 Loading DX10 from the System Disk

If you are using a Business System Terminal, all you need to do is cycle power on the primary 940 unit to load the DX10 operating system.

If you are using the operator panel to load DX10 from the system disk, all you need to do is turn the key to LOAD.

If you are using the programmer panel to load the system from disk, perform the following steps:

1. Press the HALT/SIE switch.
2. Press the RESET switch.
3. Press the LOAD switch.

Figure 3-1 shows the programmer panel for the 13-slot chassis, and Figure 3-2 shows the programmer panel for the 17-slot chassis. If the fault indicator light remains on, a crash has occurred in the load procedure, and the procedure is not successful. Refer to Volume VI for crash analysis.

3.2.3 Loading DX10 from Other Media

You can load DX10 from a disk cartridge placed in a disk drive other than the default disk drive by changing the TILINE address value so the initial loader program can identify the device where DX10 resides. If the simplified IPL procedure presented in paragraph 3.2 does not meet your needs, refer to the *Universal ROM Loader User's Guide* for instructions on loading DX10 from several different devices.

If your secondary loader program resides on punched cards, magnetic tape, cassette, or cartridge, or if you have a stand-alone object program (such as diagnostics on magnetic tape), refer to the *Universal ROM Loader User's Guide* for the procedures to load DX10 or the object program into memory.

3.2.4 Contingencies for a Fault or System Crash During IPL

Several different events can happen during the IPL procedure. If you are using a programmer panel, you should carefully note the following conditions of the programmer panel because it is important in diagnosing the problem:

- Which lights are on. Note them by name and number.
- Which lights are flashing. Note them by name and number.

You can also note the contents of the numbered lights by converting each four-light group from binary (on equals 1; off equals 0) to hexadecimal. Then refer to Volume VI.

The Business System Terminal fault indicator lights through the top of the 940. If you are using an operator panel, see your system manager.

3.2.5 Initialize System (IS) Command

The IS command is a command procedure that performs several functions to initialize the system. On new systems, the IS command sets up two terminals to come up in VDT mode without requiring log-on. (Section 4 describes VDT mode.) You must perform a Modify Terminal Status (MTS) command to modify the terminals to require log-on.

If your IS command does not perform as described in the following procedure, someone has modified it. You need to find out the operating procedure for your system. See your system manager.

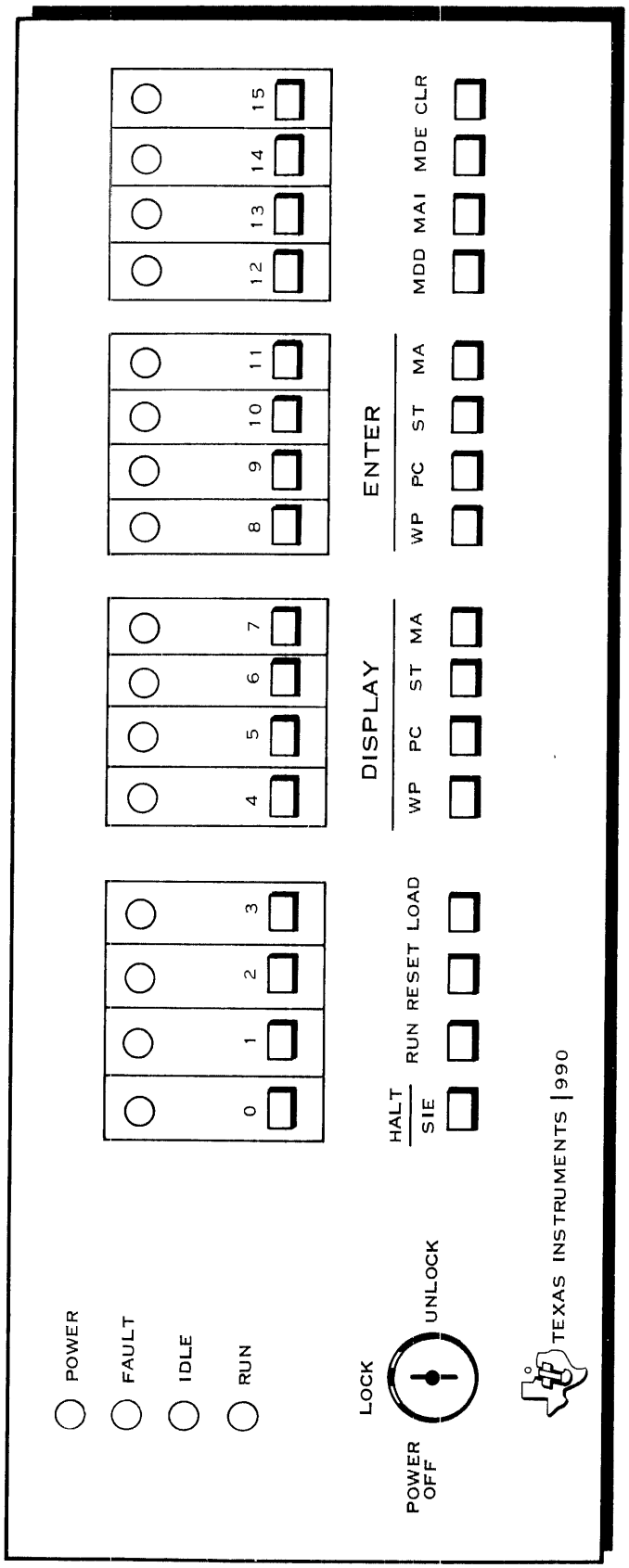
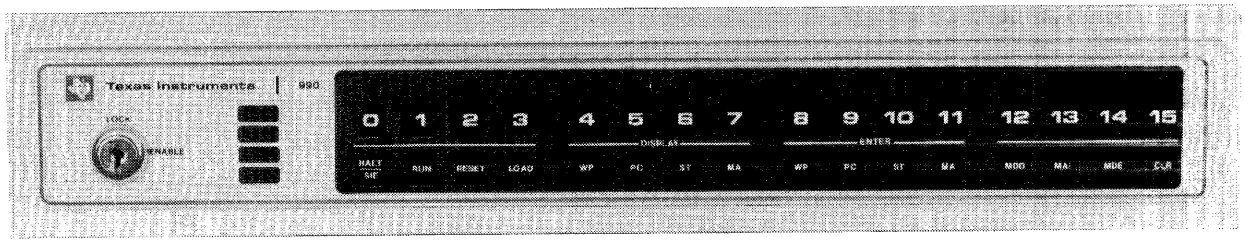


Figure 3-1. Programmer Panel for the 13-Slot Chassis



2276951

Figure 3-2. Programmer Panel for the 17-Slot Chassis

You may receive one of the following messages when you enter the SCI command mode if your patch files do not match the current release of the operating system. (That is, the patch levels applied to the various system files are not the same.) You can continue to work and apply the correct patches and perform an IPL at a later time. For information on applying patch files, refer to Volume V.

APPLY DX10 X.X.X MEMRES PATCHES AND REBOOT

APPLY DX10 X.X.X PROGA PATCHES AND REBOOT

where:

X.X.X is the release level of the operating system.

The standard procedure for executing the IS command is as follows:

1. Turn on the terminal if it is not already on.
2. Press the blank orange key on the Model 911 VDT. (Refer to Table 4-2 if you are not operating from a 911 VDT.)
3. Simultaneously press the SHIFT and exclamation point (!) keys.

NOTE

This is the SCI activation procedure. Section 4 describes this procedure in more detail.

4. Type in the characters IS in response to the SCI prompt ([ ]). The following field prompts then appear:

```
INITIALIZE SYSTEM
INITIALIZE DATE AND TIME
YEAR:
MONTH:
DAY:
HOUR:
MINUTE:
INITIALIZE SYSTEM LOG
ATTENTION DEVICE:
LOGGING DEVICE:
SYSTEM LOG PROCESSING TASK ID: 05E
ANALYSIS OUTPUT PRINTER: LP01
USER LOG PROCESSOR TASK ID: 0
```

5. Answer the prompts of the IS command as follows:

Prompt	Response
YEAR:	Enter two to four numeric characters representing the current year. For example, 82 and 1982 both mean 1982.
MONTH:	Enter one or two numeric characters representing the current month or the first three alphabetic characters of the month.
DAY:	Enter one or two numeric characters representing the current day.
HOUR:	Enter one or two numeric characters representing the current hour (according to the 24-hour clock).
MINUTE:	Enter one or two numeric characters representing the current minute.
ATTENTION DEVICE:	Enter a device name to specify a device as the receiver of attention messages produced by the system during the logging process. You should specify DUMY if you do not desire an attention device.
LOGGING DEVICE:	Enter a device name to specify a device to which the system log is written. You should specify DUMY if you do not desire logging to a device.
SYSTEM LOG PROCESSOR TASK ID:	Enter the installed ID of a task on the system program file. The task is bid whenever a log file is filled. This parameter is to be used by add-on packages to DX10. The initial value is >5E, which is the task ID of the system log analysis task for online diagnostics. You can use this value if online diagnostics are installed or if no analysis task is used.

Prompt	Response
ANALYSIS OUTPUT PRINTER:	Enter the name of a line printer used by the system log analyzer. Valid responses are LP01 through LP09. If left blank, the system log analyzer report file is used.
USER LOG PROCESSOR TASK ID:	Enter the task ID of an installed online diagnostics task on the system program file. The task is bid whenever a log file is filled. This parameter is to be used by add-on packages to DX10. Entering an ID of 0 indicates that no task is to be bid.

3.3 SHUTTING DOWN THE OPERATING SYSTEM

There is no command or command sequence used to shut down the operating system. If you press HALT and then RESET, you must perform an IPL to restart DX10. That is an effective shutdown. However, any programs, text edits in progress, and printing at a printer are also forced to terminate and must be restarted.

Make sure all programs are completed and the terminals are logged off before shutting the system down. (If you know of a quiet state for your applications where file integrity is not jeopardized, it is not necessary to log off.)

3.3.1 Hardware Precautions Before Reinitializing

Make sure the system disk is online and that it is not write protected.

3.3.2 Software Precautions Before Reinitializing

Make sure the disk from which you perform the IPL procedure is your system disk.

3.4 HARDWARE POWER-DOWN

The following list describes how to power down the hardware. It is basically a reversal of the steps described in paragraph 3.2.1. The order that you power down the various peripherals is usually not important, but it is very important that you make the various disks stop spinning before you turn off the power to the computer.

1. Follow these steps to power down the system disk:

   a. Make the disk stop spinning. Refer to the appropriate manual for the operating procedures for the disk you are using.

   b. Physically remove the disk from the disk drive.

   c. Remove the AC power from the disk drive by turning the circuit breakers off.

2. Enter Q in response to the SCI prompt ([ ]) to deactivate SCI at any terminals you are using. Then, turn off the power to the terminals.

3. Power down any other special devices that you are using.

4. Turn off the power to the printer(s).

5. Turn off the power to the computer.

NOTE

The 990 17-slot chassis power switch is on the back of the cabinet near the bottom. On 13-slot and 6-slot chassis, the power switch is at the top left corner of the back of the chassis (when viewed from the back).

4

Operating DX10 from a Terminal

4.1 INTRODUCTION

This section describes how to operate DX10 from a terminal by using the System Command Interpreter (SCI). It describes how to activate SCI and what to do if the activation procedure does not work. Next, a table describes the functions of the 911 VDT keys that DX10 supports. Another table shows the equivalent keys for each keyboard device that DX10 supports. Throughout the manual, the key descriptions apply to the 911 VDT so you should refer to this table if you are using a different type of keyboard. Then, the section describes how to enter commands to DX10 through SCI from a video display terminal (VDT) or from a TTY device. (For a detailed discussion of SCI, refer to Section 6.) The section then discusses entering commands in expert mode. Interactive commands can bypass SCI prompts if you enter them in expert mode. Finally, the section describes how to deactivate SCI.

NOTE

If DX10 is not ready for use, you should refer to Section 3.

You can enter SCI commands interactively or in batch streams. Commands that you enter from an interactive terminal are entered in foreground mode. SCI responds to the entry of a command by displaying or printing command field prompts associated with the command. If you enter commands from a hard-copy or teleprinter device, one field prompt at a time is displayed or printed. This is called TTY mode. (You can also use VDTs in TTY mode.) If you enter commands from a VDT, a series of prompts is displayed simultaneously. This is called VDT mode. VDT mode is the recommended operating mode for the 911 VDT, 913 VDT, 940 EVT, and Business System Terminal. Hard-copy devices, such as the Models 733, 743, and 820 terminals, always operate in TTY mode.

Commands entered in batch streams operate without the aid of an interactive terminal. These commands operate in background mode. For more information on batch mode, refer to Section 7.

You must enter all commands and field prompt responses in uppercase unless the lowercase-to-uppercase mapping option is specified by the .OPTION primitive in the M$00 procedure. The .OPTION primitive in the M$00 procedure optionally enables lowercase-to-uppercase mapping for each terminal. You must specify the .OPTION primitive for each terminal individually.

4.2 ACTIVATING SCI

To operate DX10, you must activate SCI at your terminal. This may require a log-on procedure that prompts you for your user ID and passcode. If the terminal does not require log-on, you should perform the following steps to activate SCI at your terminal:

1. Turn on the terminal if it is not already on.

2. Press the blank orange key.

3. Simultaneously press the SHIFT and exclamation point (!) keys.

4. To show that it is activated for VDTs, SCI responds with the top-level SCI menu, and the SCI prompt ([ ]) appears in the lower left corner of the screen. The news file is displayed first if one exists. To show that it is activated for TTYs, SCI responds only with the SCI prompt ([ ]) in the lower left corner of the screen of a VDT or on the printer of a hard-copy terminal.

NOTE

The default symbols displayed as the SCI prompt depend on the country type of the terminal and the associated controller. The prompt default symbols are as follows:

Country	Prompt
Germany/Austria	Ä Ü
Norway/Denmark	Æ Å
Sweden/Finland	Ä Å
All Others	[ ]

You can modify the SCI prompt by using the .OPTION SCI primitive. (Refer to Volume III.)

5. Begin operating the terminal by entering the available SCI commands. The privilege level assigned to the terminal determines which commands you can use. Privilege levels are assigned to terminals when the system is initialized by an Initialize System (IS) command.

To activate SCI at a terminal requiring log-on, perform the following steps:

1. Turn on the terminal if it is not already on.

2. Press the blank orange key.

3. Simultaneously press the SHIFT and exclamation point (!) keys.

4. The following message appears at all terminals. (The mode of a terminal is preset by a privileged user. Refer to MTS command description for an explanation of how to preset terminals.)

   SYSTEM COMMAND INTERPRETER-PLEASE LOG IN
   USER ID:

   VDTs set to VDT mode also display the following prompt:

   PASSCODE:

5. Type in your user ID and press the RETURN key to signal SCI that an entry has been made. On data terminals, the security of the user ID is preserved by not displaying the ID.

6. SCI responds by displaying or printing the following message if it is not already displayed:

   PASSCODE:

7. Type in your assigned passcode and press the RETURN key to signal SCI that an entry has been made. To preserve security, the passcode does not appear on the screen.

8. If the log-on is successful and the terminal is in TTY mode, SCI responds with the SCI prompt ([ ]). If the terminal is in VDT mode, SCI also displays the top-level SCI menu. For terminals in VDT mode, SCI displays the news file first if one exists. You should then press the CMD key to display the top-level SCI menu.

NOTE

The default symbols displayed as the SCI prompt depend on the country type of the terminal and the associated controller. The prompt default symbols are as follows:

Country	Prompt
Germany/Austria	Ä Ü
Norway/Denmark	Æ Å
Sweden/Finland	Ä Å
All Others	[ ]

You can modify the SCI prompt by using the .OPTION SCI primitive. (Refer to Volume III.)

9. Begin operating the terminal by entering the SCI commands that are available as determined by the privilege level associated with your user ID. Privileged users assign the privilege level to user IDs by using the Assign User ID (AUI) command. When you enter a command for which you are not authorized, SCI displays or prints the following message:

   ****ERROR 9012****PRIVILEGED COMMAND

4.3 WHEN SCI DOES NOT RESPOND TO THE ACTIVATION SEQUENCE

The following checklist describes what to do if SCI does not respond to the blank orange/exclamation point key sequence:

- Make sure the terminal is turned on.
- Make sure the terminal is plugged in.
- Make sure the brightness is turned up high enough.
- If touching the blank orange key causes a system crash, the system generation is not correct. Refer to Volume V.
- Perform a List Terminal Status (LTS) command on another terminal and make sure the terminal status says the terminal is on.
- Refer to Volume VI if none of these are causing the trouble.

4.4 KEYBOARD PURPOSE AND EQUIVALENCY CHARTS

Table 4-1 describes the purpose of the 911 VDT keys that DX10 supports. Any other keys are dependent on which program is using the terminal: SCI or an application program. Tables 4-6 and 4-7 explain the functions of the keys that SCI supports. Application programs can be written to use these keys for their own function, as described in Volume III.

Table 4-2 shows the equivalent keys for each of the keyboards for the various terminals that DX10 supports. When using this manual, you should refer to Table 4-2 if your terminal is not a 911 VDT. This manual uses only the 911 VDT keys when describing keys.

Figure 4-1 through Figure 4-4 show the keyboards for the terminals listed in Table 4-2. For keyboards of other terminals, refer to the appropriate installation and operation manuals.

Table 4-1. Function of Model 911 VDT Keys That DX10 Supports

Key(s)	Function
Alphabetic, numeric, and special character	The alphabetic, numeric, and special character keys enter information such as commands, parameters, and responses to requests for information.
Cursor movement	The right arrow (→) and left arrow (←) keys move the cursor left or right one character position on the display screen, respectively. The up arrow (↑) and down arrow (↓) keys move the cursor up or down one line, respectively. (The up arrow (↑) and down arrow (↓) keys can be programmed by a task to perform a different function.)

Table 4-1. Function of Model 991 VDT Keys That DX10 Supports (Continued)

Key(s)	Function
Blank orange	The blank orange key is the hold key; it temporarily stops output at a terminal. Pressing any alphabetic or numeric key continues output at the terminal. (This key is also called the reset key.)
Blank orange then RETURN	Aborts output at a terminal.
Blank orange then exclamation point (!)	SCI log-on sequence.
Blank orange then CONTROL/X	Kills the task performing input or output at a station and allows SCI to regain control of the station.
CONTROL	The CONTROL key allows character keys to perform different functions. Holding the CONTROL key down and pressing a character key creates a control character (producing a different and unique code for each control character selected). For example, the code that the character P generates is a different code than a CONTROL/P code. Both codes are unique.
DEL CHAR	The DEL CHAR key deletes the character over which the cursor is positioned. Any character or character string to the right of the cursor in the current field automatically moves one character position to the left, and a blank is inserted in the rightmost position of the field.
ERASE FIELD	The ERASE FIELD key erases the contents of a field.
INS CHAR	The INS CHAR key inserts characters in a field. As you insert each character, the cursor and any characters to the right of the cursor move right one character position.
REPEAT	The REPEAT key causes repetition of another key. Holding the REPEAT key down and then pressing another key repeats the key until you release the REPEAT key.
RETURN	The RETURN key terminates entry of data into a field and submits the data entered to the computer for processing.
UPPER CASE LOCK	The UPPER CASE LOCK key locks the keyboard in uppercase mode until you press the key a second time. This key affects only the alphabetic keys and does not affect the control keys.

Table 4-1. Function of Model 991 VDT Keys That DX10 Supports (Continued)

Key(s)	Function
SKIP	The SKIP key erases the contents of a field from the current position of the cursor to the end of the field, terminates entry of data into a field, and submits the data entered to the computer for processing.

Table 4-2. Keyboard Equivalents

DSR Output Code[1]	911 VDT Key[2]	913 VDT Key	940 EVT Key	Business System Terminal Key[2]	820 KSR/Teleprinter Devices Key[2]
7F	ERASE FIELD	CLEAR	ERASE EOF	ERASE FIELD	DEL
80	CONTROL/1	F0	F9	SHIFT/F1	CTRL/3
81	F1	F1	F1	F1	CTRL/A
82	F2	F2	F2	F2	CTRL/B
83	F3	F3	F3	F3	CTRL/C
84	F4	F4	F4	F4	CTRL/D
85	F5	F5	F5	F5	CTRL/E
86	F6	F6	F6	F6	CTRL/F
87	Right FIELD		LINE FEED	Right FIELD	CTRL/G
88	Left arrow	Left arrow	Left arrow	Left arrow	CTRL/H or BACKSPACE

Notes:

[1] The DSR output code is the result of the keyboard codes mapped by the DSR handling the device to the values in the table. Refer to Volume III.

[2] In this manual, the notation CONTROL/n, CTRL/n, or SHIFT/n indicates that you simultaneously press the CONTROL, CTRL, or SHIFT key and another key (n).

Table 4-2. Keyboard Equivalents (Continued)

DSR Output Code[1]	911 VDT Key[2]	913 VDT Key	940 EVT Key	Business System Terminal Key[2]	KSR/Teleprinter Devices Key[2]
89	TAB	TAB	TAB right	TAB	CTRL/I
8A	Down arrow	Down arrow	Down arrow	Down arrow	CTRL/J or LINE FEED
8B	SKIP	SET	SKIP right	SKIP	CTRL/K
8C	HOME	HOME	HOME	HOME	CTRL/L
8D	RETURN	NEW LINE	RETURN	RETURN	CTRL/M or RETURN
8E	ERASE INPUT	DELETE LINE	ERASE INPUT	ERASE INPUT	CTRL/N
8F	Blank gray	INSERT LINE	INS LINE	Blank gray	CTRL/O
90	DEL CHAR	DELETE CHAR	DEL CHAR	DEL CHAR	CTRL/P
91	INS CHAR	INSERT CHAR	INS CHAR	INS CHAR	CTRL/Q
92	Right arrow	Right arrow	Right arrow	Right arrow	CTRL/R
93	ENTER	SEND	SEND	ENTER	CTRL/S[3]

Notes:

[1] The DSR output code is the result of the keyboard codes mapped by the DSR handling the device to the values in the table. Refer to Volume III.

[2] In this manual, the notation CONTROL/n, CTRL/n, or SHIFT/n indicates that you simultaneously press the CONTROL, CTRL, or SHIFT key and another key (n).

[3] For teleprinter devices, CTRL/S instead of ESC performs the hold function. Therefore, the key sequence for initiating SCI at a teleprinter device is CTRL/S followed by an exclamation point (!). (However, if DSR820, DSRKSR, or ASRDSR are included in the link map, ESC is required instead of CTRL/S.)

Table 4-2. Keyboard Equivalents (Continued)

DSR Output Code[1]	911 VDT Key[2]	913 VDT Key	940 EVT Key	Business System Terminal Key[2]	KSR/Teleprinter Devices Key[2]
94	Left FIELD	BACK TAB	SKIP left	Left FIELD	CTRL/T
95	Up arrow	Up arrow	Up arrow	Up arrow	CTRL/U
96	F7	F7	F7	F7	CTRL/V
97	F8		F8	F8	CTRL/W
98	CMD	HELP	NEXT FORM	CMD	CTRL/X
99	PRINT	PRINT	PRINT	PRINT	CTRL/Y
9A	CONTROL/2		F10	SHIFT/F2	CTRL/Z
9B	Blank orange	RESET	PREV FORM	Blank orange	CTRL/S[3]
9C	CONTROL/4		F11	SHIFT/F3	CTRL/\
9D	CONTROL/5		F12	SHIFT/F4	CTRL/[[4]
9E	CONTROL/6	ROLL UP	F13	SHIFT/F5	CTRL/^
9F	CONTROL/7	ROLL DOWN	F14	SHIFT/F6	CTRL/_

Notes:

[1] The DSR output code is the result of the keyboard codes mapped by the DSR handling the device to the values in the table. Refer to Volume III.

[2] In this manual, the notation CONTROL/n, CTRL/n, or SHIFT/n indicates that you simultaneously press the CONTROL, CTRL, or SHIFT key and another key (n).

[3] For teleprinter devices, CTRL/S instead of ESC performs the hold function. Therefore, the key sequence for initiating SCI at a teleprinter device is CTRL/S followed by an exclamation point (!). (However, if DSR820, DSRKSR, or ASRDSR are included in the link map, ESC is required instead of CTRL/S.)

[4] The 733/743 ASR/KSR devices use CTRL/].

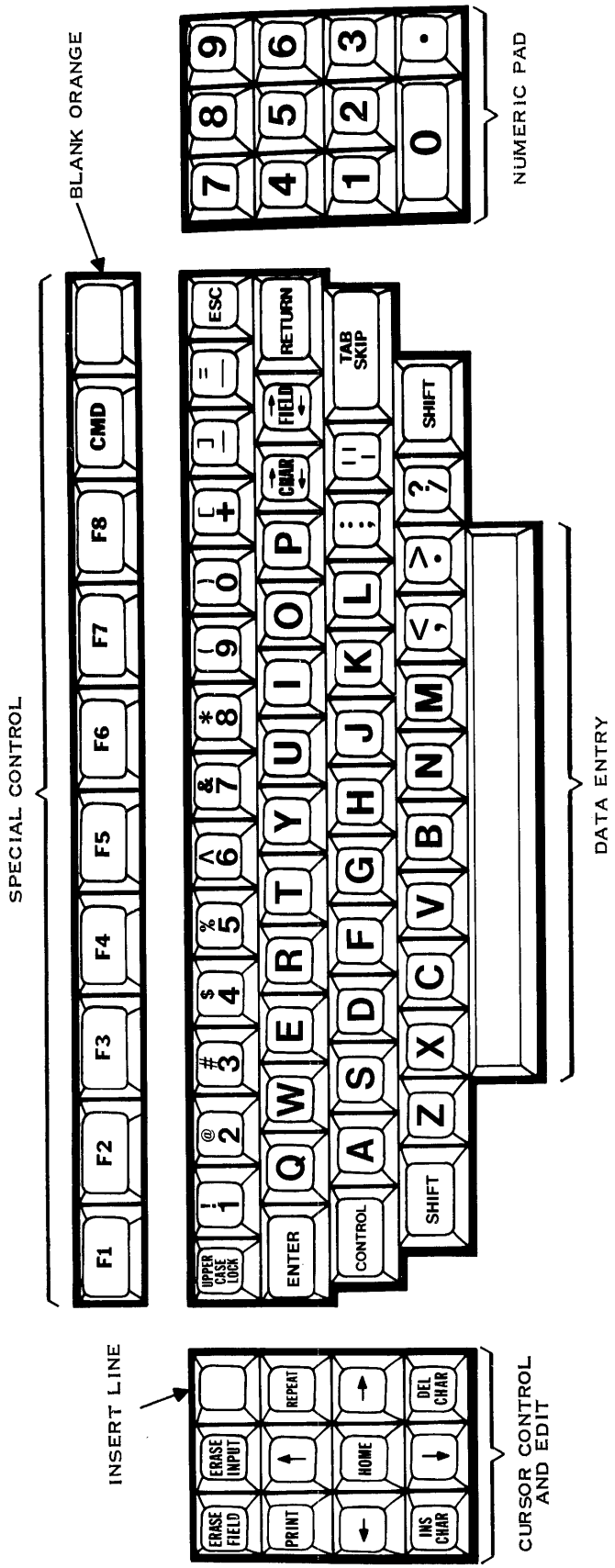
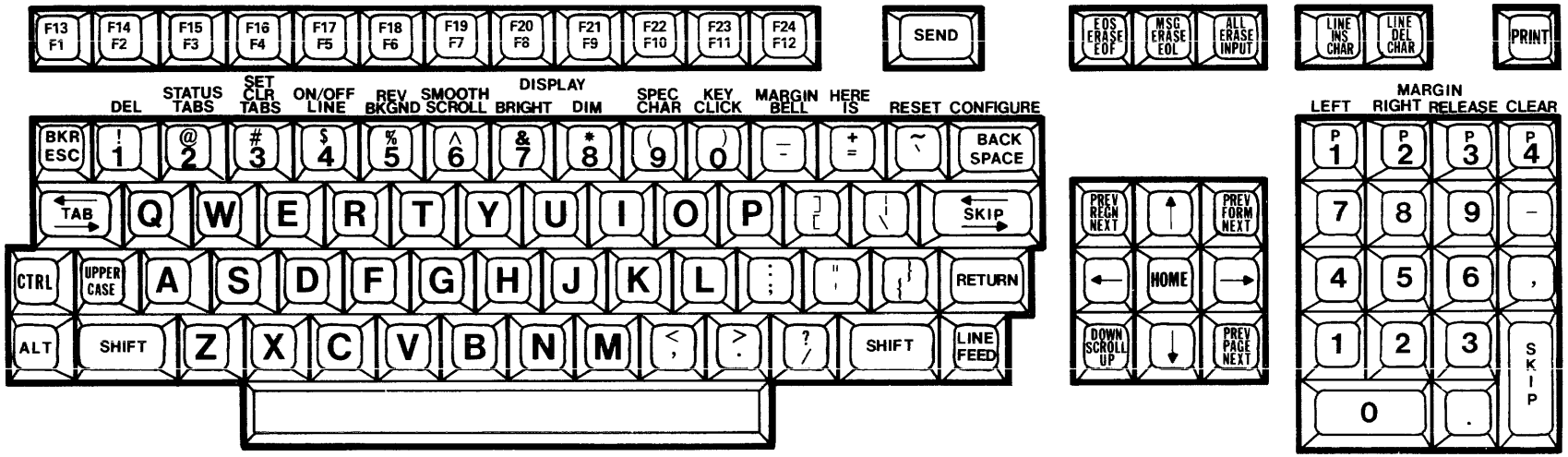


Figure 4-1. Model 911 VDT Standard Keyboard Layout



2283045

Figure 4-2. Model 940 EVT Standard Keyboard Layout

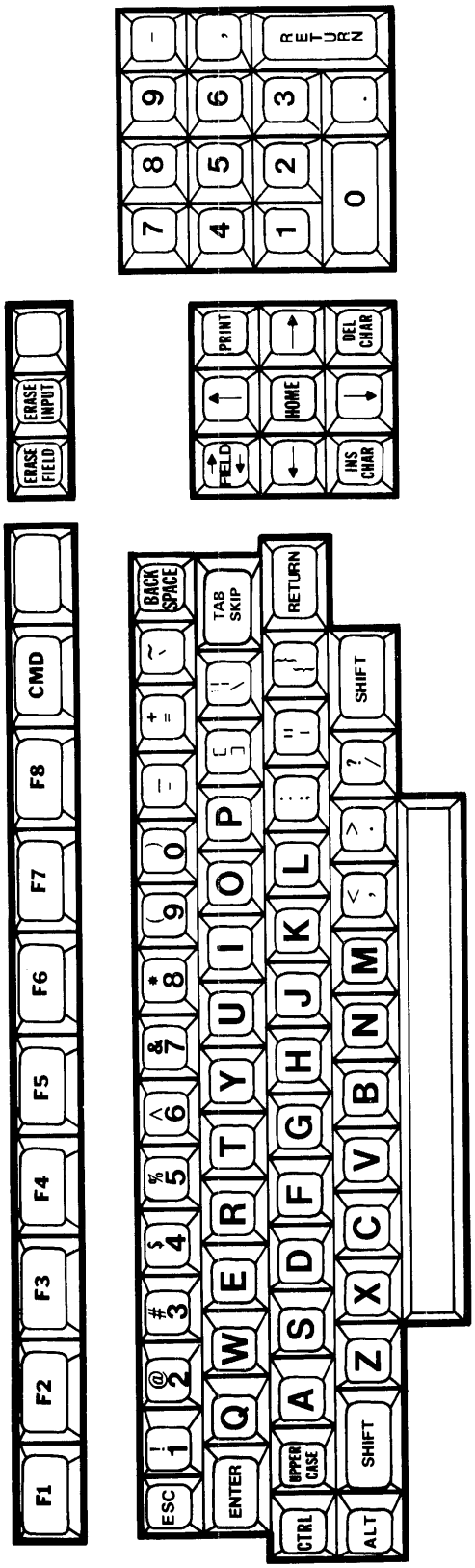


Figure 4-3. Business System Terminal Keyboard Layout

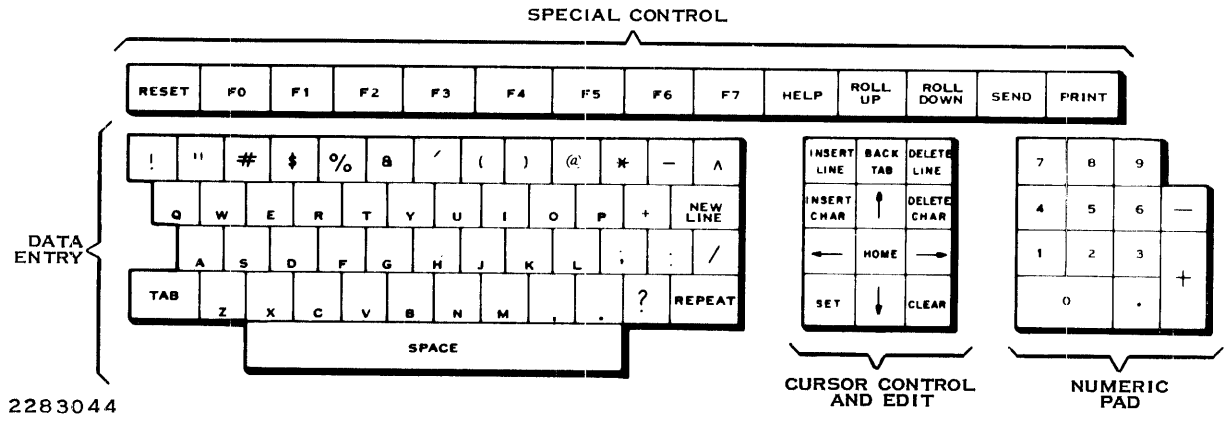


Figure 4-4. Model 913 VDT Standard Keyboard Layout

4.5 ENTERING COMMANDS

When you activate SCI at a VDT in VDT mode, the top-level menu shown in Figure 4-5 appears. When you activate SCI at a TTY terminal or at a VDT in TTY mode, enter /LC to display the top-level menu one line at a time.

```
SELECT ONE OF THE FOLLOWING COMMAND GROUPS

        /DEV    -- DEVICE OPERATIONS
        /FILE   -- FILE OPERATIONS
        /PDEV   -- PROGRAM DEVELOPMENT
        /SMAIN  -- DX10 MAINTENANCE
        /SOP    -- DX10 OPERATION
```

Figure 4-5. Top-Level Menu of SCI

After display of the menu, you can enter any SCI command, or you can enter one of the menu selectors to display a list of the available commands or additional menus in a particular category. For example, to display a list of commands dealing with device operations, perform the following steps:

1. Type the following characters at the keyboard: /DEV
2. Press the RETURN key.

Figure 4-6 shows a sample hierarchy of menus available for device operation commands, obtained when you enter the /DEV option.

This manual presents each command in the following format:

- The command format, which includes the characters of the command, the full name of the command, and any command field prompts
- User responses to the prompts of the command
- Example of how to use the command

The following paragraphs describe the meaning of notations used in command formats, and the procedure for entry of commands at VDTs in VDT mode.

4.5.1 Command Format and Notation

This manual uses a standard format and notation to describe SCI commands. Figure 4-7 points out the parts of that format as it is used to describe the Show File (SF) command. The following paragraphs describe the meaning and use of each of those parts of the command format.

4.5.1.1 Characters of the Command. To enter a command in VDT mode, you should type the characters of the command and then press RETURN to signal DX10 that the entry is complete. For example, if you want to execute the Show File (SF) command, you should type the characters SF and then press RETURN.

4.5.1.2 Prompts That SCI Returns. When you enter a command in VDT mode, SCI displays the full name of the command and all of the prompts associated with the command. Prompts provide you with information and request that you enter parameters to complete command execution.

When you enter a command in TTY mode, SCI prints or displays the full name of the command entered and the first prompt associated with the command. Prompts provide you with information and request that you enter parameters to complete command execution. When you have answered the first prompt, the second prompt appears if it exists.

Prompts that request a source of input or a destination of output use standard terminology that states an entry should be the name of a file (PATHNAME), the name of a device (DEVICE NAME), or either the name of a device or the name of a file (ACCESS NAME). For example, the following prompt requests that you enter a pathname to identify an output file:

```
OUTPUT PATHNAME:
```

4.5.1.3 Type of Response Expected. After you enter a command in VDT mode and SCI displays all prompts associated with the command, then SCI places the cursor after the first prompt and waits for you to enter a response. The following steps describe how to enter a response to a prompt:

1. Type the desired response, or accept the default or displayed initial value by typing nothing. The response must be of the type specified by the abbreviation given in the command format described in Table 4-3.
2. Press the RETURN key to signal SCI that the entry is complete.

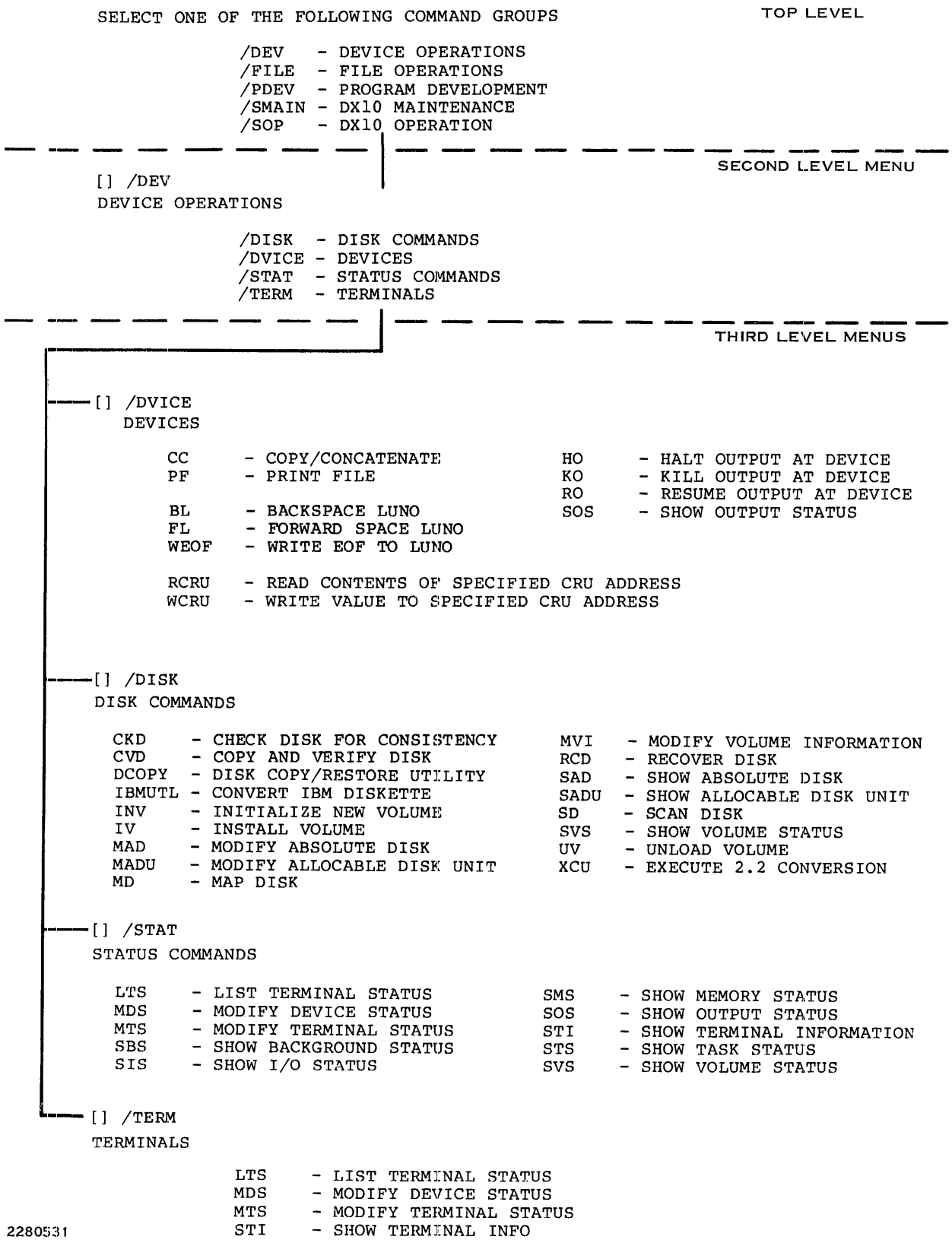


Figure 4-6. Example of the Hierarchy of Menus Available for Device Operation Commands

```
SF
SHOW FILE
FILE PATHNAME: < acnm> @
```

where:

SF	Characters of the command
SHOW FILE	Full name of the command returned by SCI
FILE PATHNAME:	Prompt returned by SCI
< >	Response notation
acnm	Type of response expected
@	Response notation

Figure 4-7. Command Format This Manual Uses

When you enter a response to the first prompt in VDT mode, the cursor moves to the field following the next prompt and waits for you to enter a response. After you enter a response to the last prompt, the command executes.

When you enter a response to the first prompt in TTY mode, SCI prints or displays the next prompt and positions the data entry position of a TTY terminal, or a cursor of a VDT, immediately after that message. After you enter a response to the last prompt, the command executes.

Table 4-3 explains the abbreviations used in command formats to define the types of responses that you can make to prompts. Table 4-4 defines the legal device names.

SCI sometimes helps you respond to prompts by displaying an initial value after a message or by having a default value available for a response. The following paragraphs describe initial values and default values.

Initial Values. An initial value is a value that SCI automatically displays as a response to some prompts. You can accept the initial value by pressing the RETURN key, or you can erase the initial value by pressing the ERASE FIELD key. You can reject the initial value by entering a different value.

The initial values for some prompts are fixed, and the same value always appears as the initial value for that prompt. In other cases, SCI saves values entered with one command for later display as initial values for a later entry of the same command or for the entry of a related command. SCI also saves some variable initial values from one terminal session to another.

Default Values. SCI often provides default values to speed up the entry of responses to prompts. This is especially true for optional user responses. A default value is a value that SCI automatically supplies in response to a prompt if there is no initial value or user-supplied value. To accept the default value for a prompt (where a default value exists), you should press the RETURN key without entering any other data. Such an entry is called a null response.

4.5.1.4 Response Notation. Notation symbols enclose each response abbreviation (except @) in the command format to define the type of response expected for a prompt. Table 4-5 describes the meaning of the response notations used.

More than one set of notation symbols can be used to define a response. For example, the following set of notations indicates that a response is optional and that it can be more than one character string separated by commas: [(string)].

Table 4-3. Expected Responses to Prompts

Abbreviation	Expected Responses
acnm	The device name of a device as described in Table 4-4 (for example, ST01) or the pathname of a file (for example, VOL1.MYFILE) as described in the discussion of pathnames in Section 7.
int	A hexadecimal or decimal integer or integer expression (for example, 360/12). You must precede hexadecimal values with a right angle bracket (>) or a leading zero.
name	An alphanumeric name such as a device name (for example, LP01).
string	Any sequence of printable characters (for example, "A = B"). You must enclose character strings containing any of the following characters in quotation marks ("): equals sign (=), at sign (@), ampersand (&), asterisk (*), exclamation point (!), comma (,), apostrophe ('), lowercase characters, and blanks. You can include quotation marks in the string by specifying the quotation mark twice for each quotation mark desired.
yes/no	You must enter Yes or No. Alternatively, you can enter any character string (three characters or less in length) that begins with Y for Yes, and any character string that begins with N for No.

Table 4-4. Legal Device Names

Device Name	Device
CMxx	A communication device specified by a device name in the range CM01 through CM99.
CRxx	A card reader specified by a device name in the range CR01 through CR99.
CSxx	A cassette unit of a Model 733 ASR terminal specified by a device name in the range CS01 through CS99.

Table 4-4. Legal Device Names (Continued)

Device Name	Device
DKxx	A diskette unit specified by a device name in the range DK01 through DK99. You only use DKxx when you are using single-sided, single-density diskette drives (FD800 units).
DSxx	A disk drive specified by a device name in the range DS01 through DS99; includes double-sided, double-density (DSDD) diskettes.
LPxx	A line printer specified by a device name in the range LP01 through LP99, or LP$1 through LP$9.
ME	Your terminal. You can find out the station ID of your terminal by entering the Show Terminal Information (STI) command as described in STI command description.
MTxx	A magnetic tape unit or cartridge tape unit specified by a device name in the range MT01 through MT99.
STxx	A terminal specified by a device name in the range ST01 through ST99. You can find out the station ID of your terminal by entering the STI command as described in the STI command description.

Table 4-5. Response Notations Used in Command Formats

Notation	Meaning
< >	The response is required.
[ ]	The response is optional.
{ }	The response must be one of the enclosed items. If the enclosed items are character strings in capital letters, you must enter one of the character strings.
( )	More than one response of the type specified can be entered in a list with each response separated by commas.
@	Synonyms are allowed as responses.

4.5.2 VDT Mode Command Entry Control

Table 4-6 shows the keys to which SCI gives special control functions when it is accepting your responses to prompts in VDT mode.

Table 4-6. VDT Mode Command Entry Control Keys

911 VDT Key	913 VDT Key	940 EVT Key	Business System Terminal Key	Function
HOME	HOME	HOME	HOME	Positions the cursor after the first prompt displayed on the screen. You can then reenter responses to all of the prompts. The HOME key does not allow you to return to a previous prompting screen of the same command.
Blank orange	RESET	PREV FORM	Blank orange	The blank orange key is the hold key; it temporarily stops output at a terminal. Pressing any alphabetic or numeric key continues output at the terminal. (This key is also called the reset key.)
Blank orange then RETURN	RESET then NEW LINE	PREV FORM then RETURN	Blank orange then RETURN	Aborts output at a terminal.
RETURN, ENTER, or TAB	NEW LINE or TAB	RETURN or TAB	RETURN or TAB	Accepts the currently displayed initial value as the response to the prompt. It can also terminate the current entry and proceed to the next prompt, or it can start processing by the command if the command has no more prompts.
FIELD ←	BACK TAB	← SKIP	← SKIP	Depending on the current position of the cursor, repositions the cursor to the start of the current field or to the previous prompt. (It does not do this if the cursor is already positioned at the start of the response field for the first prompt.)
ERASE FIELD	CLEAR	ERASE EOF	ERASE EOF	Deletes an entire response to a prompt including any initial value that the system supplies. You can then enter a new response to the same prompt.
CMD	HELP	NEXT FORM	CMD	Aborts a command currently accepting responses and allows you to enter another command. Also aborts those executing commands (such as LD or MD) that can be aborted to return control to you.

Table 4-6. VDT Mode Command Entry Control Keys (Continued)

911 VDT Key	913 VDT Key	940 EVT Key	Business System Terminal Key	Function
F4	F4	F4	F4	Duplicates the corresponding field in the previous line. This key does not cause field termination. You can accept the duplicated field by pressing RETURN, or you can modify the duplicated field before accepting it.

4.5.3 TTY Mode Command Entry Control

Table 4-7 shows the keys to which SCI gives special terminal control functions when it is accepting your responses to prompts in TTY mode.

Table 4-7. TTY Mode Command Entry Control Keys

911 VDT Key*	913 VDT Key	940 EVT Key	Business System Terminal Key*	Teleprinter Devices Key*	Function
RETURN	NEW LINE	RETURN	RETURN	RETURN	Accepts the currently displayed (or printed) initial value or default as the response to the prompt. It can also terminate the current entry and proceed to the next prompt, or it can start processing by the command if the command has no more prompts.
				CTRL/L	Reprints the first prompt for a command. You can then reenter values for all prompts of the command.
Left FIELD	BACK TAB	$\overleftarrow{\text{SKIP}}$	$\overleftarrow{\text{SKIP}}$	CTRL/T	Redisplays or reprints the previous prompt. The initial value associated with the prompt is redisplayed, and you can enter a new response.

Table 4-7. TTY Mode Command Entry Control Keys (Continued)

911 VDT Key*	913 VDT Key	940 EVT Key	Business System Terminal Key*	Teleprinter Devices Key*	Function
CONTROL/1	F0	F9	SHIFT/F1	CTRL/3	Redisplays the prompt without an initial value. This is the erase field function in TTY mode. You can then enter a new response.
CMD	HELP	NEXT FORM	CMD	CTRL/X	Aborts a command currently accepting responses and allows you to enter another command. Also aborts those executing commands (such as LD or MD) that can be aborted to return control to you.
F4	F4	F4	F4	CTRL/D	Duplicates the corresponding field in the previous line. This key does not cause field termination. You must manually terminate the field. Note that if a teleprinter device is attached via a switched line and is configured for disconnect on EOT, CTRL/D causes a disconnect.

Note:

* In this manual, the notation CONTROL/n or CTRL/n indicates that you simultaneously press the CONTROL (or CTRL) key and another key (n).

4.6 EXPERT MODE

Entering a command in expert mode is comparable to issuing a command in batch mode except that you enter commands at the interactive terminal. The advantages of expert mode over normal command entry are that it is faster, saves keystrokes, and saves paper for TTY devices. The disadvantage is that you must memorize the prompt and prompt responses.

To utilize expert mode effectively, you should be familiar with the command characteristics. Specifically, you should know all prompts, initial values, and required prompts for the command. With this in mind, you can enter the command and prompt assignments by using the continuation characters, comma (,) or equal sign (=), as the last character on each line.

NOTE

The assignments apply only to prompts that occur in the primary command. If the command procedure calls another command procedure, the primary command does not automatically pass the assignments to the secondary command.

On the screen, SCI formats the prompts and suppresses the display of the full name that usually appears on the top line. If you supply all values that require prompts and all values are syntactically correct, SCI processes the command. The following is an example:

```
[ ] SF FILE = .S$NEWS
```

This command shows the file .S$NEWS on the interactive terminal without prompting for the file pathname. You should note that you can abbreviate the field prompt FILE PATHNAME to FILE, as in batch streams. If you enter the SF command again, the initial value for FILE PATHNAME is .S$NEWS. Knowing this, you can enter the SF command followed by a period to again show the file .S$NEWS, as follows:

```
[ ] SF.
```

If a period appears after a command name in response to the SCI command prompt, the assumption is that you are using expert mode. The assignments FILE = .S$NEWS in the first example and period (.) in the second example cause SCI to execute in expert mode.

Another feature of SCI that is especially useful in expert mode is field prompt assertion. For instance, when you are making a field prompt assignment whose value is YES, listing the field prompt without the assignment is semantically equivalent. The following examples are equivalent:

```
[ ] QE ABORT = YES

[ ] QE ABORT

[ ] QE A
```

In this case, the command directs the Text Editor to abort. The semantic meaning is that you assert the abort condition. SCI treats this as if you syntactically typed Y after the ABORT = field prompt. You must take special care when using this feature since some field prompts have a negating qualifier at the end. For example, if you assert the field prompt DELETE, the full name of the field prompt may actually be DELETE PROTECT. You then delete protect a file rather than delete enabling it.

Expert mode has one additional quality that makes it particularly useful for a TTY device. If, when using expert mode, you do not supply a required value and an initial value is not supplied, SCI prompts you for that value. The following is an example:

```
[ ] PF FILE = .S$NEWS
```

If you executed a PF command previously, the LISTING DEVICE field prompt has an initial value. If this is the first PF command executed for this session, SCI prompts with the following:

```
LISTING DEVICE:
```

After you respond to this prompt, the command executes. However, in VDT mode, LISTING DEVICE is not the last prompt. In this mode, SCI does not issue the field prompts that have initial values or are optional. Nevertheless, in TTY mode, the value of this feature is the number of lines of saved paper. Note that this mechanism assumes that you know the unseen assignments or initial values.

4.7 DEACTIVATING SCI

To deactivate SCI at a terminal, you must enter the Quit (Q) command and press RETURN. Termination of SCI allows tasks to access the terminal as a physical device as well as free system resources used by the SCI tasks.

If the Text Editor or Debugger is active at the terminal, then the Q command automatically prompts for the Quit Edit (QE) or Quit Debug Mode (QD) command responses. Following the entry of responses to either the QE or QD commands, the following message appears:

```
QUIT PROCESSING INCOMPLETE: RE-ENTER 'Q' COMMAND
```

To terminate SCI at your terminal, press the RETURN key and then reenter the Q command.

If a background task is active at the terminal when you enter the Q command, the following message appears:

```
****ERROR 901E****CAN'T QUIT WITH BG TASK PENDING
```

You must allow the background task to complete, or you can terminate it by using the Kill Background Task (KBT) command. Then, you must reenter the Q command following the background task completion in order to terminate SCI at the terminal.

5

Maintaining Your System

5.1 GENERAL

This section describes the commands that you can use to initialize the system, create a new system disk, and make sure that your DX10 operating system is set up to operate satisfactorily. These commands include the Initialize Date and Time (IDT), the Initialize System Log (ISL), and the Create System Files (CSF) commands. The section also provides detailed information on planning and using proper backup procedures.

5.2 RESETTING THE DATE AND TIME

You use the Initialize Date and Time (IDT) command when you execute the Initialize System (IS) command, and you can also use it to reset the system date and time. After you initialize the DX10 system as described in Section 3, you may need to reset the system date and time if you entered the wrong responses to the IS command. The IDT command allows you to specify the year, month, day, hour, and minute. Upon receipt of the command, DX10 starts counting the seconds at zero. The system uses the 24-hour clock.

5.3 RESETTING THE SYSTEM LOG

You use the Initialize System Log (ISL) command when you execute the IS command, and you can also use it to reset the system log. After you initialize the DX10 system as described in Section 3, you may need to reset the system log if you entered the wrong responses to the IS command. You may also want to reinitialize the system log to change logging devices or to reinitialize the log after a logging error. To reinitialize the system log, you must first release global LUNOs > 12 and > 13.

5.4 CREATING SYSTEM FILES

You use the Create System Files (CSF) command to create the system roll file, system crash file, and the S$PRINT library. You use this command when you build a new system disk by using the Copy Directory (CD) command or a similar process such as backup and restore. The roll file has to be on the system disk to perform an initial program load (IPL). When creating the crash and roll files, CSF may return a 0027 SVC error. The error results from an attempt to delete files that do not exist before creating new ones. In this case, the new files are created correctly and the error can be ignored.

5.5 CHANGING DISK VOLUMES IN THE COMPUTER

The DS10, CD1400/32, and CD1400/96 disk drives contain a removable disk as well as a fixed one. You may want to change the volume mounted in the removable disk position during system operation. Note that the volume designated as the system disk must not be removed during system operation, even if it is mounted in the removable disk position of the drive. The system disk contains information that the operating system continually accesses during normal system operation. Absence or replacement of the system disk could cause system failure and possibly destroy any information on the disk.

If you try to change the system disk, the system crashes and you must perform an IPL. If you change a disk other than the system disk without using the Unload Volume (UV)/Install Volume (IV) sequence, DX10 allows further reads from the disk, but any attempt to write to the disk causes an > EB error; this error may or may not appear on the screen, depending on which utility accesses the disk. You can correct the > EB error by executing a UV command on the installed disk and an IV command on the new disk.

While the removable volume of a DS10, CD1400/32, or CD1400/96 disk drive is being changed, the fixed disk is inoperable. Therefore, if the fixed volume is the system disk, all system activity must be halted before the volume exchange begins. The following procedure is recommended:

1. If the volume you want to remove is installed, confirm that it is not being used; then execute the Unload Volume (UV) command. For additional information on the UV command, refer to the UV command description.

2. Perform the Show Volume Status (SVS) command to verify that the volume is not installed. The NAME INSTALLED: field of the display must be blank. For additional information on the SVS command, refer to the SVS command description.

3. Inform all active users of the system that the system disk is to be powered down briefly. All user and system activity must halt before you power down the disk. Wait until all users confirm that all foreground and background activities are complete and that no more activity is to be initiated until you finish the disk exchange.

4. After all activity has ceased, follow the procedure for dismounting the volume in the drive, and mounting the one to replace it. When the disk drive is operational again, inform users that they may resume activity on the system.

If the disk drive containing the volume to be removed does not contain the system disk volume, the system can continue to operate. Activity must be halted only on the volume to be removed and on the fixed volume of the same drive. The following procedure is recommended.

1. If the volume to be removed is installed, confirm that it is not being used, then perform the Unload Volume (UV) command. For additional information on the UV command, refer to the UV command description.

2. Perform the Show Volume Status (SVS) command to verify that the volume is not installed. The NAME INSTALLED: field of the display must be blank. For additional information on the SVS command, refer to the SVS command description.

3. Protect the fixed volume from being accessed while the drive is not operational. The procedure required to ensure that the system is not accessed at this time may vary from one system to another. The safest method is to unload both disks (use the UV command) and modify the device state (use the MDS command) to take the disk drives offline. After you change the disks, use the MDS command to put the disk drives online.

   You must contact all active users who might access information on the disk to confirm that they have ceased their current disk activity and will not resume it until you complete the exchange.

4. When all activity to the two disk volumes has ceased, you can dismount and replace the removable volume. When the disk is once again operational, notify the users that they may resume activity to the disk.

Another procedure that you can use to ensure that the disk is not accessed during the volume exchange operation involves halting the computer. This procedure may provide a quicker method of exchanging the disk volumes, but it has some undesirable side effects. The procedure requires that you place the computer in the HALT mode while the volumes are exchanged. Since the computer clock does not operate while the HALT mode is active, the clock will not be accurate when you return the computer to the RUN mode. You can correct the clock by using the Initialize Date and Time (IDT) command, but activities that are sensitive to clock information may be adversely affected. Another undesirable side effect is that the other users of the system may assume that the system has failed. Unless you forewarn them that the computer is to be halted, they may conclude that a system failure has occurred.

NOTE

In performing the following procedure, use caution when activating the switches on the front panel. Activation of a switch other than the ones indicated in the procedure may cause undesirable results such as system failure.

The procedure for halting the system is as follows:

1. If the volume to be removed is installed, confirm that it is not being used; then, perform the Unload Volume (UV) command. For additional information on the UV command, refer to the UV command description.

2. Perform the Show Volume Status (SVS) command to verify that the volume in the drive is not installed. The NAME INSTALLED: field of the display must be blank. For additional information on the SVS command, refer to the SVS command description.

3. Notify users that you are halting the computer until the disk exchange is complete.

4. Press the HALT switch on the front panel of the computer.

5. Remove the disk volume from the drive, and mount the desired volume.

6. Press the RUN switch on the front panel of the computer.

7. Install the new disk by using the IV command.

8. If desired, use the Initialize Date and Time (IDT) command to set the time to the correct value. For additional information on the IDT command, refer to the IDT command description.

9. Inform all users that you have returned the computer to the RUN mode.

5.6 BACKUP AND RECOVERY TECHNIQUES

This paragraph discusses the reasons for maintaining data backup and guidelines for designing a backup program. You can use some of these procedures to back up the system disk. Related information on each backup/copy utility is included to aid you in selecting the appropriate utility for your purposes. Step-by-step procedures explain how to use the devices on various configurations of the 990 computer system.

5.6.1 Why You Should Back Up Your Data

There is one basic reason to back up your data: to provide a basis for recovery in case part or all of the data suffers loss. Any of the following could make your data inaccessible:

1. Accidental deletion (erasure)

2. Power failure

3. Fire that damages your equipment and/or media

4. Head crash on a disk drive

5. Other hardware failure such as a broken disk controller

6. System software failure

7. Application software failure

DX10 provides several SCI commands and utilities that facilitate backup and recovery. This paragraph discusses each command and presents principles you can use to organize an effective backup and recovery program. This will enable you to recover your data with a minimum amount of time and difficulty in case of loss. Section 8 includes details on how to execute each command.

5.6.2 General Guidelines

The following list gives some general guidelines you can use to check any backup system. If your backup system does not meet the guideline, you run a greater risk of data loss because of the reason given. The guidelines are offered to help you understand how to protect your data.

1. Always keep a log of what is done to the system and when, including backups, hardware maintainance, and changes to application and system software. Such a log can help you trace problems and identify methods of preventing recurrence.

2. Never back up your current data over your most recent backup. If something fails, you may lose your backup and your current data, leaving you with an old backup or nothing. Use a rotation system involving a minimum of three copies so you never back up onto your latest backup disk.

3. Back up frequently enough to minimize data loss in case of failure. It is recommended that you back up the data at least once a week when data is not very volatile. Depending on how volatile the data is, you may need to back up the data at least once a day, and more often if it is critical enough. However, you have to be the final judge of what is sufficient.

4. Always keep an old system disk (one that does not contain current data) available to use for troubleshooting the system. If your equipment fails in such a way that it erases data or otherwise produces errors on your system disk, you do not want to put your good backup into the disk drive.

5. Always verify a copy operation. Copying without verify saves time in the short run, but it can later turn out to cost much. It is possible for hardware to fail in such a way that data is not copied correctly but no errors are reported by the copy utility. If you keep the original disk until you know the copy is good, you can proceed without verifying the copy.

6. Consider transaction logging. If done properly, it can reduce the frequency of backing up whole disks with no compromise of recovery capability. It can also make data recovery much easier and is not very difficult to implement. One of the TI standard packages (DBMS) contains a logging facility that you can use. Volume III discusses transaction logging principles.

5.6.3 Utilities for Backup and Recovery

DX10 provides a number of backup and recovery utilities to aid you in implementing an effective program. There are three categories of backup commands as follows:

- Volume backup:
  - Disk Copy/Restore (DCOPY)
  - Copy and Verify Disk (CVD)
- Directory backup:
  - Copy Directory (CD)
  - Backup Directory (BD)
  - Restore Directory (RD)
  - Verify Backup (VB)
  - Verify Copy (VC)
- File backup:
  - Copy/Concatenate (CC)
  - Copy Sequential Media (CSM)
  - Copy KIF to Sequential File (CKS)
  - Copy Sequential File to KIF (CSK)
  - Copy KIF Randomly (CKR)

This paragraph summarizes these utilities and their principal characteristics. For details on their use, refer to the command descriptions.

Your choice of backup utility depends on the hardware configuration you have and your method of identifying transactions. If you create your backup procedure and use a "physical" form of update data, such as currency blocks for KIFs, you must be extremely careful to have the currency match the transaction being logged, and you must use a form of backup that preserves the internal structure of the file. DBMS uses this form of backup. The log entries are created internally so the data always is correct for the transaction, but the backup must not depend on a utility that rebuilds the DBMS internal structures. It is easier, especially with KIF, to use a "logical" form of logging transactions, which would be keeping key values. Then the backup can be any form desired, including CKR.

5.6.3.1 Volume Backup Commands. The CVD and DCOPY commands copy and optionally verify disks used in the DX10 operating system. The CVD utility copies from disk to disk. The DCOPY command copies from disk to magnetic tape, magnetic tape to disk, or disk to disk. Their principle advantages are that they do not require the system disk to be physically installed and they are fast. The disadvantage is that the output disk must be the same size and type as the input disk.

DCOPY. This utility copies disks of like type and size, either directly disk to disk or through magnetic tape. It preserves all file structures (including inconsistencies, if there are any) and preserves all attributes of the master disk, including the name of the system to use for IPL (if it is a system disk). It does not require the system disk to be mounted to perform a copy operation, and can therefore be used to back up a data-only disk in a two-disk system.

DCOPY performs the copy track-by-track with no disk compression. When a copy is made, the destination disk must be the same type as the original source disk even when tape is the intermediate media. When you specify a magnetic tape as the output device in the DCOPY operating procedure, the first and last records are identification records of ASCII characters. Each identification record contains a volume name, the date of creation, a reel number, and data describing the type of disk from which the tape was created. The reel number contained in the identification records at the beginning and end of the tape provide the means for copying disks to or from multiple tapes.

DCOPY does not transmit disk errors. Indeed, if it encounters a disk error it will render the copy disk unusable and terminate the copy process. It is inflexible with disk allocation and cannot handle a condition where the input disk has data on a track and the output disk has a bad track at the same track number location.

CAUTION

The recommended method is that you use error-free disk packs only when using the DCOPY utility; you should not use flag-free packs. However, you can use DCOPY on non-error free disks by initializing each disk of a backup set with the union of the bad track sets from all the disks.

CVD. This utility works much like DCOPY except that it is flexible with disk allocation. It does not require the system disk to be mounted to perform a copy operation, so you can use it like DCOPY. It runs somewhat slower, but should be preferred to DCOPY because of its allocation flexibility. It rebuilds directory structures and does disk compression but preserves internal file structures. (The disk compression includes truncating unused space from the end of all files except KIFs and merging all secondary allocations of the file together into as few allocations as possible.) It does not perform compression within a file, which means it does not recover disk space released by deleting records from a KIF or deleting tasks, procedures, or overlays from a program file. Like DCOPY, it does not transmit disk errors. If it encounters a disk error, it renders the copy unusable and terminates the copy process. It does not support tape as intermediate media. CVD can work with disks that have bad tracks.

5.6.3.2 Directory Utility Commands. The CD, VC, BD, RD, and VB commands allow you to copy, back up, restore, and verify directories and/or specific files under directories. These utilities are capable of backing up single files or directories, in whole or in part, to another disk or to magnetic tape. They preserve file structures internal to files, but rebuild directory structures.

Like CVD, these utilities perform disk compression, but only a limited file compression. Under the control of options, they can recover the unused space at the end of a KIF. They cannot recover disk space released by deleting records. The disk space remains allocated to the file. They can only recover the unused space at the end of a program file. Under the control of the FAST and NOFAST options, BD may or may not recover space released by deleting tasks, procedures, or overlays to the end of the copy. If BD does recover space within the file not used for storing tasks, procedures, or overlays, this space is only recovered to the end of the file. A single copy never recovers space within the file not used for storing tasks, procedures, or overlays. Two copies are necessary with CD or a BD/RD sequence with the NOFAST option specified.

Disk compression is limited to the available space on the copy disk. Full compression only takes place if large enough blocks exist on the copy to accommodate whole files.

These utilities require the system disk to remain mounted, and so cannot back up a data-only pack in a two-disk system. They can back up selected files, and they can also back up files selected by the date of their last modification.

The CD and RD utilities work in such a way that the number of records in the copy as shown by a LD or MD is updated all at once, not gradually as records are put in the file. Therefore, if LD or MD shows the correct number of records in the copy, it is no assurance that the copy is complete. You must perform a verify operation to be sure.

5.6.3.3 File Backup Commands. The CC, CSM, CKS, and CSK commands allow you to make copies of files. Except for CSM, these utilities do not have a verify capability. You should not depend on these utilities for critical backup because of the lack of verify capability.

The primary use of CKR is to recover data from KIFs that have been damaged in some way (for example, by malfunctioning hardware) or that were under partial logging when a system failure occurred (for example, system table overflow or power failure). Like CC and CKS, CKR is not recommended for backup.

5.6.3.4 How These Utilities Preserve Physical Record Length. This paragraph describes what each of the backup commands do to physical and logical records.

DCOPY and CVD perform no processing that involves physical records or logical records and, therefore, perform no checking of file structure integrity. The files on the copy disk have exactly the same characteristics as they do on the master disk.

The CD, BD, RD, VB, and VC commands preserve physical records and their contents without processing the logical records in a file, unless you modify them with SPRL and RPRL options (see the command descriptions). These commands rebuild the file structures involving disk allocation and directories. Even with those options, they do only a limited amount of formatting. Integrity checks with those options are minimal. Without options, there is no integrity check on the logical records. The BD command creates its output file with a logical record length of 80 and a default physical record length. This does not affect its ability to back up files of any physical or logical record length.

The CC, CSM, CKS, CSK, and CKR commands copy logical records from an input file to an output file or device. The output file can reside on a different medium. If the output file does not already exist, the output file is created by the utility, except for the CSK and CSM commands. CSK requires the output KIF to be already created; you create the output KIF and this means you can preserve whatever you want. For the other commands, the output file is created with the same logical and physical record length as the input file. Logical records written to a device are reduced to the record length of the device if their logical record length is greater than the record length of the device. Output files that you precreate retain the logical and physical record length as you created them.

The CC, CSM, CKS, and CSK commands use standard I/O SVCs to transfer logical records to the output file or device. This leaves all integrity checking of logical records to DX10. File management builds the copy files, and the integrity checking is complete. CKR performs its own integrity checking on the logical records in the input KIF. Records written to the output file by CKR use standard I/O SVCs.

5.6.4 Some Suggested Procedures

This paragraph presents some suggestions for implementing a backup program. It presents the reasons for each step; you need to consider these reasons and any of your own ideas and extend the procedures to meet your needs. In general, you should plan to be able to recover from total loss of the disk pack in the drive at the time of a failure. This will enable you to recover data lost because of accidental deletion as well as power failure, severe head crash, and controller failure. You may want to plan for recovering from even more catastrophic loss as noted in the following paragraph.

5.6.4.1 Offsite Archive and Computing Capacity. Your data is valuable. If there were to be a fire or explosion or some similar catastrophic event that destroyed your computer installation and the media, your business could experience difficulty depending on how critical the data center is to its operation. You may want to consider archiving backup packs offsite. Based on the cost of recovery, the cost of maintaining extra backups, and the risk of such a loss, you may decide that you need total offsite backup. In many cases, maintaining offsite backup capable of restoring your data base completely is not worth the cost. You may only want to keep your oldest backup offsite, along with old archived transaction logs capable of bringing your data back to the status it had at the beginning of the day. Other methods of handling larger risks (such as insurance) may be appropriate for you.

If you lose your computer, you may be able to order another one. However, until it arrives, you may need computing capacity to perform the minimum amount of work needed to keep your business going. Sometimes you can make arrangements with another firm that has a computer similar to yours so that you have access to such capacity.

5.6.4.2 A Rotation Backup System. A minimum rotation backup system requires two extra disk packs for each online pack that you want to back up. Put labels on each pack to identify its position in the rotation cycle and to identify which online pack it backs up. If you use tape for backup, you need two sets of tapes for each online pack, each set capable of holding the contents of the disk to be backed up. Develop a well-documented system of keeping a written log of which packs were backed up, which pack they were copied to, who did the copy, when the copy was made (date and time), and any other information you may need to identify any problems with the backup cycle. Some common mistakes are to back up to the wrong pack in the cycle, or to back up to a pack meant for a different online pack.

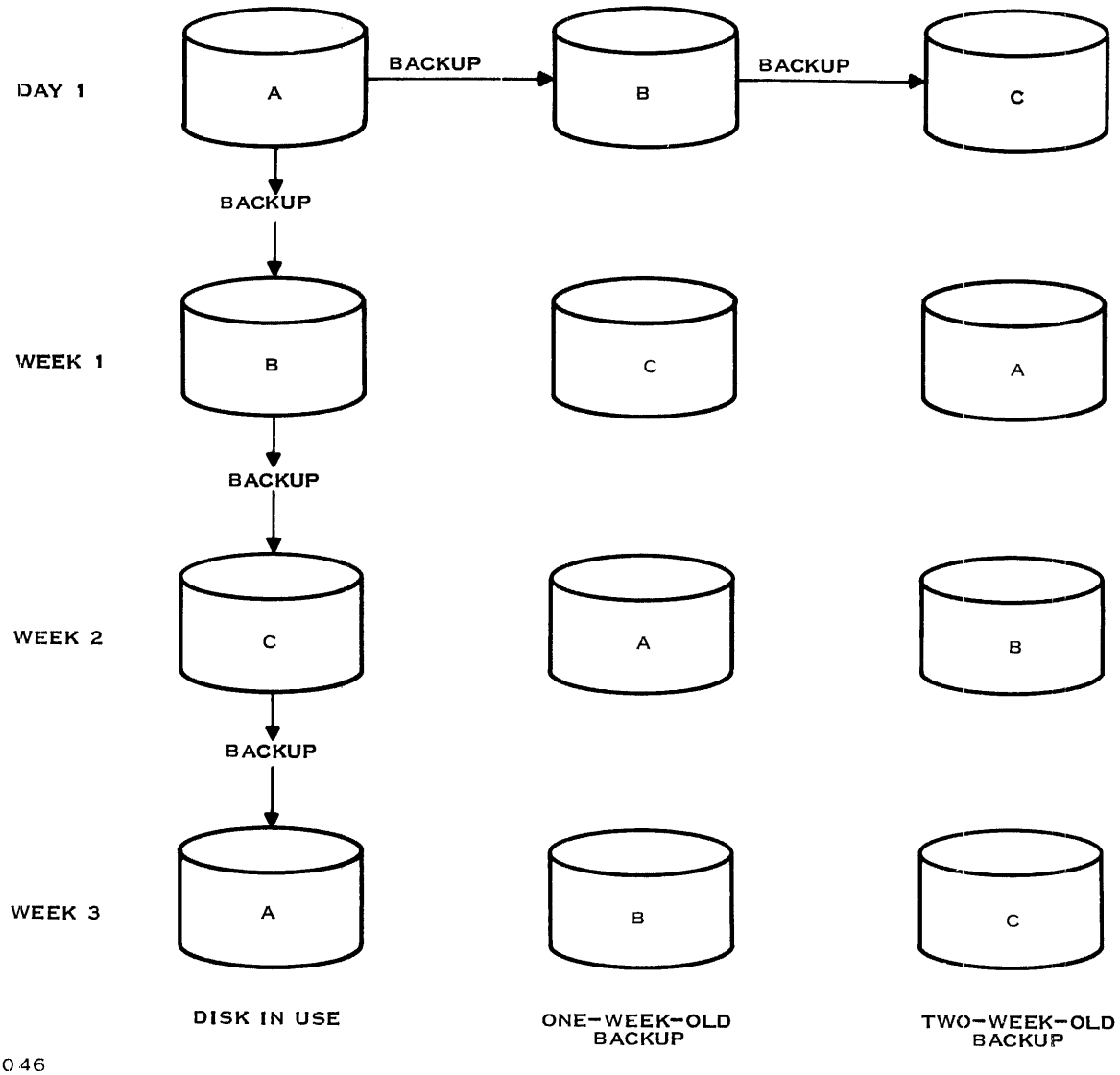


Figure 5-1. Suggested Disk Pack Rotation in a Weekly Backup Procedure

If you are using magnetic tape, label each reel in sequence. DCOPY and BD maintain headers and trailers to identify the reels and can detect out of sequence reels on restore, but your procedure should not depend on that alone to ensure correct backups.

In addition, check the integrity of the backup media periodically. Magnetic tape wears out and needs to be replaced. Disks can develop soft spots that hold data long enough for verification but lose the data over time. Plan to check the media periodically to ensure smooth functioning of your backup procedures and to ensure availability of the data if you need it.

Develop and document contingency plans to use in case any step of the backup procedure does not work correctly. Such a plan might include copying the disk with CD rather than CVD. The backup operator can flag any file in error, and you can recover from the previous backup prior to beginning the next day's business. Your plans should also include keeping extra tapes (and possibly disks too) for archiving extra logs and making extra backups. Any point of the procedure is subject to failure, and you should analyze what you need for recovery at any point.

Develop and document plans for recovery in case of failure during system operation between backups. Such a plan may include extra tapes to archive partial transaction logs, and procedures for when to use them.

5.6.5 Backup/Recovery Plan for Computer Systems with Fixed Disk Storage

In those computer systems equipped with a single disk drive with fixed disk storage such as the DS10, the disk drive accommodates two magnetic disks. One of the disks is a fixed (nonremovable) platter and the other is a replaceable disk cartridge. When the fixed disk contains the computer's operating system and other system files (as is the usual case), your data files are on the replaceable disk. In most instances of computer system operation, the disk cartridge containing the data files currently in use is mounted on the disk drive. Additional data files (including backup files) are kept on other disk cartridges that are available for mounting on the disk drive at your option.

Using the CVD command is normally the best way to back up either the fixed disk or the removable disk. DCOPY is faster, but it is inflexible when bad tracks exist on the copy disk. If you decide to use it, substitute DCOPY for CVD in the following procedure.

5.6.5.1 Single-Drive Fixed Disk Backup. The fixed disk can be copied to a backup disk in only a few minutes using the CVD command to copy from disk units DS01 (fixed) to DS02 (removable) as follows:

1. Prior to removing the removable disk from the disk drive spindle, release any LUNOs assigned to the removable disk by terminating applications using it and/or using the appropriate release LUNO command. For example, use the Release Global LUNO (RGL) command to remove any assigned global LUNOs. Unload the removable disk by entering the Unload Volume (UV) command. After you unload the volume, the operating system displays a message indicating that the disk unit has been unloaded.

2. Procedures for installation and removal of the disk cartridge are given in the appropriate disk drive manual. Stop the disk drive by pressing the START/STOP button on the drive. When the START/STOP indicator lamp extinguishes, remove the installed removable disk and replace it with the disk cartridge upon which the backup of the fixed disk is to be made.

3. Press the START/STOP switch to start the drive spindle motor. When the READY indicator on the disk drive lights, enter the CVD command. The CVD utility announces the beginning of execution followed by requests for interactive responses. The operating procedure for the CVD command is described in the CVD command description.

4. Have CVD perform the copy from DS01 to DS02.

5. Remove the backup disk from DS02. Identify the disk properly and store as the backup copy of the system files.

6. Replace in disk unit DS02, the disk cartridge that was removed in step 2.

7. Perform an initial program load (IPL).

The entire process of using the CVD command to copy the fixed disk to the removable disk takes about eight minutes for a DS10. Note that the data on the backup removable disk is now exactly the same, file for file, as that on the fixed disk. Consolidation of the various files occurs, and the volume information is the same as the fixed disk.

5.6.5.2 Single-Drive Removable Disk Backup. You can use the same CVD command to perform a backup of the data on the removable disk. This procedure is more involved than the backup of the fixed disk because it requires you to make four disk-to-disk copies. Two extra disk cartridges are required. One disk cartridge is used as a backup to hold the system data while the fixed disk is being used in the CVD process; the other is used for the retained backup copy of the removable disk. The disk cartridge used to back up the fixed disk is available for other use once its data is copied back to the fixed disk at the end of the process unless you elect to use it as the backup of the fixed disk. In this case, part of this procedure serves as the fixed disk backup.

CAUTION

In the procedure that follows, do not leave or terminate the CVD program until after data on the system backup disk is restored to the fixed disk. Once CVD begins, the entire program is in computer memory and can be used repeatedly until you have finished making copies. If this restoring of data to the fixed disk is not completed properly, you must perform an IPL using an installed backup disk in DS02. Then you can use CVD to restore the system backup disk data to the fixed disk.

In the following procedure, the fixed disk on DS01 is called FIXED, the removable disk cartridge containing data that is to be backed up is called CART, the disk cartridge used for backup of the fixed disk is called FIXEDBCK, and the disk cartridge used to backup the removable disk is called CARTBACK. To make a backup copy of data on the removable disk (CART), proceed as follows:

1. Remove CART and install FIXEDBCK on DS02. Using the CVD command, perform steps 1 through 4 of the backup procedures for the fixed disk to copy FIXED to FIXEDBCK.

2. Remove FIXEDBCK and install CART on DS02.

3. Use CVD to copy CART (DS02) to FIXED (DS01). CVD asks if you want to modify the system disk. Respond with Y.

4. Remove CART and install CARTBACK on DS02.

5. Use CVD to copy FIXED (DS01) to CARTBACK (DS02).

6. Remove CARTBACK and install FIXEDBCK on DS02. Properly identify and store CARTBACK as the backup copy of CART.

7. Use CVD to restore FIXEDBCK to FIXED.

8. Terminate the CVD process and perform an IPL from the fixed disk.

5.6.5.3 Single Disk Drive Data Recovery. In the computer system with a single disk drive, damage that causes loss of stored data can occur to either the fixed disk or to the removable disk. If the damage occurs through some malfunction of the computer system's hardware or software, take corrective action as described in Volume VI to restore the computer system to normal operation before attempting any data recovery procedures. Do not try to use backup disk packs to recover until the hardware has been repaired and verified operational with diagnostic procedures. Data recovery procedures can begin once the system is restored to normal operation, or can begin immediately if the data loss on the disk occurred through improper use of the computer system that is not likely to recur.

If damage causing data loss occurs to the removable disk, the procedure is simply to remove the damaged disk and replace it with the latest version of the backup copy of that disk. Reenter any data entries made to the damaged disk since the previous backup cycle occurred. Make a backup of the newly restored disk. You may elect to do this before or after reentering the data entries.

Data recovery is more involved if the damage occurs to the fixed disk causing data loss of operating system files. The general procedure for data recovery to the fixed disk is as follows:

1. Remove the undamaged removable disk from disk unit DS02.

2. Retrieve a valid backup copy of the system disk (FIXEDBCK) from storage and install it at disk unit DS02.

3. Perform an IPL from disk unit DS02 (FIXEDBCK). This causes the system to name the removable disk DS01 (FIXEDBCK) and the fixed disk DS02 (FIXED).

4. Use the CVD utility to copy DS01 to DS02. This copy process restores the backup copy of the operating system to the fixed disk.

5. Terminate CVD and perform an IPL from disk unit DS01. This causes the system to revert to the names normally used.

6. Reinstall the disk removed in step 1 and resume normal operation.

5.6.6 Backup/Recovery Plan for Computer Systems with Dual Disk Drives

In the discussion that follows, assume that the system operating files are on disk unit DS01, that your data files are on disk unit DS02, and that all disk cartridges are removable. To back up the system disk (DS01), proceed as follows:

1. Remove the data files disk pack at DS02 and install the disk pack to be used for backup.
2. Use CVD to copy DS01 to DS02.
3. Remove the backup copy of the system disk at DS02 and properly identify and store it. Reinstall the data files disk pack at DS02.
4. Terminate the CVD process.

To back up your data files at DS02, proceed as follows:

1. Enter the CVD command.
2. Remove the system disk at DS01 and install the disk pack to be used for backup.
3. Use CVD to copy DS02 to DS01. CVD asks if you want to modify the system disk. Verify that the correct backup disk is installed, then respond with Y.
4. Remove the backup copy of your data files at DS01 and properly identify and store it. Reinstall the system disk at DS01.
5. Terminate the CVD process.

As was true for the computer system with a single disk drive, damage that causes loss of stored data may occur to either of the two disk drives. If the damage occurs through some malfunction of the computer system's hardware or software, take corrective action as described in Volume VI to restore the computer system to normal operation before attempting any data recovery procedures. Begin data recovery procedures once the system is restored to normal operation or begin immediately if the data loss on the disk occurred through improper use of the computer system that is not likely to recur.

If damage causing data loss occurs to the system disk, the procedure is to simply remove the disk and replace it with the system backup disk. If the damage causing data loss occurs to the removable disk containing your data files, remove the disk and replace it with the latest version of the backup copy of that disk. Reenter any data entries made to the damaged disk since the previous backup cycle occurred. Make a backup of the newly restored disk. You may decide to do this before or after reentering the data entries.

5.6.7 Backup/Recovery Plan for Diskette-Based Computer Systems

The backup/recovery procedure for diskette-based computer systems is essentially the same as that for the computer systems with dual disk drives. The DS990 diskette-based computer systems have dual flexible-disk drives. The DX10 Micro Operating System provides the same directory utility commands to copy, back up, restore, and verify directories as does the DX10 operating system. The CVD utility is also available for use in the DX5 operating system.

As in the procedure for dual disk drive systems, backup of the diskette containing system files requires simply that you remove the diskette in the other drive and replace it with a diskette upon which you want to make the backup. Then use CVD to copy from the system diskette to the inserted diskette. The backup of the data files diskette requires that you bid CVD with the system diskette in place. Then replace the system diskette with the diskette upon which you want to make the data files backup. CVD copies from the data files diskette to the inserted diskette. You must reinstall the system diskette before quitting CVD.

Procedures for data-loss recovery for the diskette-based computer systems are the same as for computer systems with dual disk drives. Assuming that any system malfunction has been corrected, the procedure is to simply remove the damaged diskette and replace it with the latest version of the backup copy of that diskette. If you replace the data files diskette, reenter any data entries made to the diskette since the previous backup cycle occurred.

5.6.8 Backup/Recovery Plan for Disk Drive/Tape Drive Computer Systems

In the single disk drive/magnetic tape drive system, you can copy the system and data files resident on the computer system disk to magnetic tape using the DCOPY utility. The magnetic tape can serve as a backup copy. If you prefer to keep the backup on disk media, rewind the tape, enter the command DCOPY, remove the disk (allowed under DCOPY), install a disk cartridge for backup, and copy the tape to the disk. Since this disk is now an exact copy of the removed disk, you can retain it in the system for operation, and you can mark the disk just removed as a backup copy and properly store it.

As pointed out in Volume V, if you have a disk drive/tape drive computer system, you should build and retain a backup copy of the system disk on disk media rather than tape media. This saves you from having to do a disk build from tape if the system disk is destroyed.

The procedure for recovering lost data for the disk drive/tape drive computer system, assuming that any system malfunctioning has been corrected, is simply to remove the damaged disk or tape and replace it with the latest version of the backup copy of the disk or tape. If you replace data files, reenter any data entries made since the previous backup cycle occurred. Make a backup of the newly restored disk. You may decide to do this before or after reentering the data entries.

6

SCI, the User Interface to DX10

6.1 GENERAL

This section describes what the System Command Interpreter (SCI) is and how you use it to direct the DX10 operating system to perform the tasks you need to accomplish. The section also presents a list of all the SCI commands available in the DX10 software package.

6.2 WHAT SCI IS

SCI is a powerful software package that comes with the DX10 operating system. It is the means by which you, the user, communicate with the DX10 operating system.

SCI has several capabilities. You can use it to communicate interactively (that is, you and DX10 have an exchange similar to a conversation), or you can use it as a programming language to perform a batch of operations that you would not want to perform by typing in each command at a terminal. You can also use the SCI programming language to create new SCI commands that you can execute interactively or in a batch of operations.

Generally, your first contact with SCI is interactive. That is, you type in a command, then SCI interprets that command and directs DX10 to perform the desired function. A command is a combination of characters that SCI recognizes as a command. You can think of a command as the name of a procedure (a set of instructions that accomplish the desired function). For example, when you enter the characters SF in response to the SCI prompt ([ ]) and press the RETURN key, the SCI software reads the letters SF and knows to execute a procedure by that name. The SF program performs all the internal operations necessary to retrieve the contents of a file anywhere in the system and display it on the terminal. SCI recognizes the characters SF as the Show File command and performs that function.

The SCI commands are called interactive because SCI asks you for all the information it needs to complete the function. When you type in the SF command, the SF SCI procedure starts executing. The procedure prompts you at the terminal for the name of the file you want to see. You enter the pathname and press RETURN. SCI uses this information to locate the file, and then displays it on the screen.

The noninteractive mode is referred to as batch mode. In batch mode, you place all of the SCI commands you want to execute in a file called a batch stream. You only issue one SCI command to execute the batch stream, and then all the rest of the SCI commands required to perform a function are given to SCI from the batch stream.

For example, you can use batch streams to create several directories, each containing several specific files. Instead of entering all of the commands from the terminal and waiting for them to execute, you can execute the batch stream. This is advantageous if you have to create this set of directories and files many times. You can also use synonyms in a batch stream. This allows you to create the directories and files with different names each time.

As mentioned previously, another capability of SCI is that you can use the SCI programming language to extend SCI's repertoire of commands. This is very important in developing applications. The two major uses for the SCI programming language are as follows:

- You can create your own SCI procedures that recognize new commands and perform functions not supplied with DX10.
- You can create a new command procedure that activates several other SCI commands during its execution.

For more information on batch mode, refer to Section 7 of this manual. For details of the SCI programming language, refer to Volume III. Generally, this manual focuses the discussion of SCI commands on interactive execution.

6.3 USING SCI

You can activate SCI at any terminal connected to DX10 (provided DX10 is running at the time) by pressing the blank orange key and then the exclamation point (!) key. (These keys apply to the 911 VDT. If you are using a different type of terminal, refer to Table 4-2 for equivalent keys.) Depending on whether the particular terminal has access protection, you may have to enter a valid ID and password before you can actually execute SCI commands. When the SCI prompt ([ ]) appears on the screen, you can begin to issue commands to DX10 through SCI.

Section 7 discusses using SCI for basic functions by presenting the commands in a logical scenario; that is, the section discusses the commands in the logical order that a new user might need to execute them to begin using the DX10 system and developing applications.

This manual describes the commands in a standard format that includes the following information:

- The command format
- User responses to the command prompting messages (both required and optional)
- Example of how to use the command

6.4 INDEX TO SCI COMMANDS

Table 6-1 lists every SCI command in the DX10 software package. The headings in Table 6-1 have the following meanings:

Code
The SCI command name.

Meaning
The meaning of the SCI command name.

Installed Type
The type of processor required by the command: T is task, O is overlay, and P is procedure.

Installed ID
The ID of the processor in the system program file, .S$PROGA. Two commands, XLE and XMA, get the processors from the system file .SSDS.

User Privilege Level Required
The user ID privilege required, or the privilege level of a terminal not requiring log-on, before the command can be executed.

Other Commands Required
The other commands that the command procedure calls. If you are going to improve memory space on the system disk by deleting unused commands, you should not delete commands that another command procedure you are going to keep requires. For example, you should not delete the LS command if you are going to keep the BATCH command since the BATCH command procedure calls the LS command.

Table 6-1. SCI Command Description Index

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
AA	Add Alias	T O P	7B, B 60-6B 1, C	0	
*AB	Assign Breakpoint	T O P	6C 8-1A, 1F 2, D	2	
AEM	Add DX10 Error Message	T	43	0	

Note:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
AF	Append File	T P	34 C, E	0	
AGL	Assign Global LUNO	T O P	7B, B 60-6B 1, C	0	
AL	Assign LUNO	T O P	7B, B 60-6B 1, C	0	
ALGS	Assemble and Link Generated System	T	20	6	XMA, XLE, XPS
ANS	Answer Incoming Call	T	5A	0	
AS	Assign Synonym	T	20	0	
*ASB	Assign Simulated Breakpoint	T O P	6C 8-1A, 1F 2, D	2	
AT	Activate Task	T P	35 C, E	0	
AUI	Assign User ID	T P	83 2, D	4	
**BATCH	Begin Batch Execution	T	20	0	Q$SYN, SDT, LS
BD	Backup Directory	T P	26, 13 1, C	0	MSG
BL	Backspace LUNO	T P	7B C	0	
CALL	Call Terminal	T	5A	0	
CC	Copy/Concatenate	T P	34 C, E	0	

Notes:

* Foreground Only
** Batch Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
CD	Copy Directory	T P	28, 13, 1F 1, C	0	MSG
CF	Create File			0	CFSEQ, CFREL, CFDIR, CFPRO, CFIMG
CFDIR	Create Directory File	T O P	7B, B 60-6B 1, C	0	
CFIMG	Create Image File	T O P	7B, B 60-6B 1, C	2	
CFKEY	Create Key Indexed File	T O P	7B, B 60-6B 1, C	0	KEY, ENDKEY
CFPRO	Create Program File	T O P	7B, B 60-6B 1, C	0	
CFREL	Create Relative Record File	T O P	7B, B 60-6B 1, C	0	
CFSEQ	Create Sequential File	T O P	7B, B 60-6B 1, C	0	
CKD	Check Disk for Consistency	T P	1A C, E	6	MSG
CKR	Copy KIF File Randomly	T P	48 C, E	0	
CKS	Copy KIF to Sequential File	T P	2C C, E	0	MSG
*CL	Copy Lines	T P O	63 2, D 3, 4	0	

Note:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
CM	Create Message	T P	83 2, D	0	
CPI	Copy Program Image	T P	39, 13, 1F 1, C, E	0	
CRV	Check and Reset Volume	T P	87 1, F	2	
CSF	Create System Files			6	DF, CFIMG, CFDIR
CSK	Copy Sequential File to KIF	T P	2C C, E	0	
CSM	Copy Sequential Media	T P	8 C	0	MSG
*CVD	Copy and Verify Disk	T	11	2	
DA	Delete Alias from Pathname	T O P	7B, B 60-6B 1, C	2	
*DB	Delete Breakpoint	T O P	6C 8-1A, 1F 2, D	2	
*DCOPY	Disk Copy/Restore Utility	T	7	2	
DD	Delete Directory	T P	29 C, E	2	
DF	Delete File	T O P	7B, B 60-6B 1, C	2	
DISC	Disconnect Terminal	T	5B	0	MSG

Note:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
*DL	Delete Lines	T O P	63 3, 4 2, D	0	
DO	Delete Overlay	T P	7B, 10, 13 C	2	
DP	Delete Procedure	T P	7B, 10, 13 C	2	
*DPB	Delete and Proceed from Breakpoint	T O P	6C 8-1A, 1F 2, D	2	
*DS	Delete String	T O P	63 3, 4 2, D	0	
*DSB	Delete Simulated Breakpoint	T O P	6C 8-1A, 1F 2, D	2	
DT	Delete Task	T P	7B, 10, 13 C	2	
DUI	Delete User ID	T P	83 2, D	4	
**DXTX	Convert DX10 File to TX990 Diskette File	T P	3E 5	0	
**EBATCH	End Batch Execution	T P O	20 2, 3 2	0	MSG, LS, SDT
EC	Batch Stream Error Counter	T P	83 2, D	0	
**ENDKEY	End CFKEY Specification	T	7B	0	MSG
*FB	Find Byte	T	6C	2	

Notes:

* Foreground Only
** Batch Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
FL	Forward Space LUNO	T	7B	0	
*FS	Find String	T	63	0	
*FW	Find Word	T	6C	2	
HO	Halt Output at Device	T P	84 2, D	0	
HT	Halt Task	T	6C	2	
IBMUTL	IBM Conversion Utility	T	41	2	
IDS	Initialize Disk Surface	T P	56, 4C 1, F	2	MSG
IDT	Initialize Date and Time	T P	19 C, E	4	IDT$1, RL
*IF	Insert File	T	63	0	
IGS	Install Generated System	T	44	6	
INV	Initialize New Volume	T P	56, 4C 1, F	2	MSG
IO	Install Overlay	T	7B, 13, 1F	2	
IP	Install Procedure	T	7B, 13, 1F	2	
IPF	International Print File	T	60	0	DF, PF
IRT	Install Real-Time Task	T	7B, 13, 1F	2	IT$$, IT$, MSG
*IS	Initialize System			4	IDT, ISL, MSG
ISL	Initialize System Log	T	1E, 04, 4A	4	ISL$

Notes:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
ISO	Install System Overlay	T P	30 C	2	
IT	Install Task	T	7B	2	IT$$, IT$, MSG
IV	Install Volume	T P	16 1	0	
*KBT	Kill Background Task	T	83	2	
**KEY	CFKEY Key Specification			0	MSG
KO	Kill Output at Device	T P	84 2, D	0	
KT	Kill Task	T	7B	2	
*LB	List Breakpoints	T	6C	2	
LC	List Commands			0	
LD	List Directory	T P	32 C, E	0	
LDC	List Device Configuration	T P	4E 1, F	2	
LHPC	List Hard-Copy Terminal Port Characteristics	T	5D	0	
LLR	List Logical Record	T P	2B C, E	2	
LM	List Memory	T	6C	2	
LS	List Synonyms	T P	36 C	0	
*LSB	List Simulated Breakpoints	T	6C	2	

Notes:

* Foreground Only
** Batch Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
LSM	List System Memory	T	6C	2	
LTS	List Terminal Status	T P	83 2, D	0	
LUI	List User IDs	T P	83 2, D	4	
MAD	Modify Absolute Disk	T P	18 C, E	6	
MADU	Modify Allocatable Disk Unit	T P	18 C, E	6	
MCC	Modify Country Code	T P	50 1, F	6	
MD	Map Disk	T P	22 C, E	0	MSG
MDS	Modify Device State	T P	4F 1, F	6	
MFN	Modify File Pathname	T	7B	2	
MFP	Modify File Protection	T	7B	2	RL
MHPC	Modify Hard-Copy Terminal Port Characteristics	T	5C	0	
MHR	Modify Horizontal Roll	T	63	0	
*MIR	Modify Internal Registers	T	6C	2	
MKF	Map Key Indexed File	T P	38 C, E	0	
MKL	Modify Key Indexed File Logging	T P	42 C, E	2	

Note:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
*ML	Move Lines	T	63	0	
MLP	Modify LUNO Protection	T P	3 1	2	
*MM	Modify Memory	T	6C	2	
MOE	Modify Overlay Entry	T	85, 10, 13	2	
MPE	Modify Procedure Entry	T	85, 10, 13	2	
MPF	Map Program File	T P	38 C, E	0	
MPI	Modify Program Image	T P	1C C, E	2	
*MR	Modify Roll	T	63	0	
MRF	Modify Relative to File	T P	21 C, E	2	
*MRM	Modify Right Margin	T	63	0	
*MS	Modify Synonym	T P	82 C	0	
MSG	Display Message	T P	83 2, D	0	
*MSM	Modify System Memory	T	6C	6	
*MT	Modify Tabs	T P O	63 2, D 4-6	0	
MTE	Modify Task Entry	T	85, 10, 13	2	
MTS	Modify Terminal Status	T P	83 2, D	4	

Note:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
MUI	Modify User ID	T P	83 2, D	4	
MVI	Modify Volume Information	T P	31 C	2	
*MWR	Modify Workspace Registers	T	6C	2	
*PB	Proceed from Breakpoint	T	6C	2	
PF	Print File	T P	84 2, D	0	
PGS	Patch Generated System	T	20	6	
Q	Quit SCI			0	QE, MSG, QD
Q$SYN	Clear Secret Synonyms			0	
*QD	Quit Debug Mode	T	6C	2	
*QE	Quit Text Editor	T	63	0	
RAL	Release All LUNOs	T P	2E C, E	2	
RCD	Recover Disk	T P	53 C	6	MSG
RCRU	Read Contents of Specified CRU Register	T P	4D C	6	
RD	Restore Directory	T P	27 C	2	MSG
RE	Recover Edit	T	63	0	
RGL	Release Global LUNO	T	7B	0	

Note:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
RL	Release LUNO	T	7B	0	
RO	Resume Output at Device	T	84	0	
*RS	Replace String	T	63	0	
*RST	Resume Simulated Task	T	6C	2	
*RT	Resume Task	T	6C	2	
RWL	Rewind LUNO	T	7B	0	
SAD	Show Absolute Disk	T P	18 C, E	2	
SADU	Show Allocatable Disk Unit	T P	18 C, E	2	
*SBS	Show Background Status	T	83	0	
SCC	Show Country Code	T P	50 1, F	0	
SD	Scan Disk	T	C	6	MSG
SDT	Show Date and Time	T	83	0	
SEM	Show DX10 Error Message	T	46	0	
SF	Show File	T	20	0	
*SIR	Show Internal Registers	T	6C	2	
SIS	Show I/O Status	T P	17 1, F	0	
*SL	Show Line	T	63	0	

Note:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
*SMM	Show Memory Map	T P	1 1	0	
SMS	Show Memory Status	T P	A0 2, D	0	
SOS	Show Output Status	T P	84 2, D	0	
*SP	Show Panel	T	6C	2	
SPI	Show Program Image	T P	1C C, E	2	
SRF	Show Relative to File	T P	21 C, E	2	
SSTM	Show System Table Map	T P	F 1	0	
*ST	Simulate Task	T	6C	2	
STI	Show Terminal Information			0	
STS	Show Task Status	T P	1D 1, F	0	
*SV	Show Value	T	6C	0	
SVL	Save Lines	T	63	0	
SVS	Show Volume Status	T P	37 1, F	0	MSG
*SWR	Show Workspace Registers	T	6C	2	
TGS	Test Generated System	T P	44 C	6	
TXCM	Compress Diskette File	T P	3C 5	2	

Note:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
*TXCP	Change Diskette File Protect Code	T P	3C 5	2	TXCP$, MSG
TXDF	Delete Diskette File	T P	3C 5	2	
*TXDX	Diskette File to DX10 File	T P	3D 5	0	
*TXFD	Format Diskette	T P	3B 5	2	
*TXMD	Map Diskette	T P	3A 5	0	
*TXSF	Set System File	T P	06 5	2	
TXXLE	Execute Link Editor to Produce TX Program File	T	85	0	(SSDS program file)
UV	Unload Volume	T	7B	0	
VB	Verify Backup	T P	40 C	0	MSG
VC	Verify Copy	T P	3F C	0	MSG
*WAIT	Wait for Background Task to Complete	T	83	0	
WCRU	Write Value to Specified CRU Address	T P	4D C	6	
WEOF	Write EOF to LUNO	T	7B	0	
XANAL	Analyze DX10 Crash File	T	6B	2	
XB	Execute Batch SCI	T	20	0	

Note:

* Foreground Only

Table 6-1. SCI Command Description Index (Continued)

Code	Meaning	Installed Type	Installed ID	User Privilege Level Required	Other Commands Required
*XCU	Execute 2.2 to 3.0 DX10 Conversion	T	2F	2	AL, RL
*XD	Execute Debugger	T	6C	2	MSG, XHT
*XE	Execute Text Editor	T O P	63 3, 4 2, D	0	MSG
*XES	Execute Text Editor with Scaling	T O P	63 3, 4 2, D	0	MSG
*XGEN	Execute System Generation Utility	T O	33 5-7	6	MSG
XHT	Execute and Halt Task	T	7B	2	
XLE	Execute Link Editor	T	86	2	MSG (SSDS program file)
XMA	Execute Macro Assembler	T	88	2	XMA2 (SSDS program file)
XPS	Executing the Patch Synonym Assignment Program	T P	E	6 C	
XT	Execute Task	T	7B	0	
XTS	Execute Task and Suspend SCI	T	7B	2	

Notes:

* Foreground Only

7

Introduction to Using SCI Commands

7.1 INTRODUCTION

This section provides an introduction to using SCI commands. The commands are described according to their functional groups.

7.2 INSTALLING, UNLOADING, AND SHOWING THE STATUS OF VOLUMES

The following commands are available to reset a media change condition on a disk drive, initialize disk volumes, install disk volumes, change disk volume information, read a disk and check for errors, show the status of a disk volume, and unload disk volumes.

Command	Description
CRV	Check and Reset Volume
IDS	Initialize Disk Surface
INV	Initialize New Volume
IV	Install Volume
MVI	Modify Volume Information
SD	Scan Disk
SVS	Show Volume Status
UV	Unload Volume

You must use the INV command for all new disks. However, for disks that have not had the diagnostic surface analysis run on them, you must use the IDS command before the INV command. The IDS command incorporates the INV prompts. The IDS command initializes the disk surface. The INV command establishes the volume catalog. The IDS command is required on new disks only. You can use the INV command as many times as desired.

7.3 CREATING AND DELETING FILES AND DIRECTORIES

DX10 supports the six file types that Table 7-1 describes. The following commands are available for creating and deleting these types of files:

Command	Description
CF	Create File
CFDIR	Create Directory File
CFIMG	Create Image File
CFKEY	Create Key Indexed File
CFPRO	Create Program File
CFREL	Create Relative Record File
CFSEQ	Create Sequential File
DD	Delete Directory
DF	Delete File

Volume III describes files, file types, and file usage in detail. In this manual, you can find information about physical and logical record sizes and how DX10 handles files.

Table 7-1. File Types DX10 Supports

File Type	Description
Relative record	Relative record files have fixed, even length records that are randomly accessed by record number. Physical records (units of disk access) can contain several logical records.
Sequential	Sequential files have variable length records that are only accessed sequentially, and may have odd, even, or zero length records.
Key indexed	Key indexed files (KIFs) have variable length records that are either randomly or sequentially accessed by key value for each of up to 14 separate keys. Odd or zero length records are not allowed.
Program	Program files are relative record files that contain executable programs including tasks, procedures, and overlays.
Directory	Directory files are relative record files whose contents are names of and pointers to other files.
Image	Image files are relative record files whose record size is related to physical divisions on disk. These are special purpose files that DX10 uses primarily for memory images and roll-in/roll-out.

DX10 arranges each disk volume into a hierarchical structure. The highest level of access in this hierarchy is the name of the volume itself. Within the volume are directories, subdirectories, and files. A directory is a specialized file that contains the names of and pointers to other files. It does not contain other data. A subdirectory is a directory to which another directory points. You can store a file directly under the disk volume or under a directory.

To access a file, you must specify not only the file name, but also the names of the volume and any intermediate directories. You must string the names of the volume, any intermediate directories, and the file together; you must separate each name with a period (.) and without any intervening blanks. This is called a pathname. If the disk volume is the system disk, you can omit the volume name from the pathname and begin the pathname with a period. For more information on file and directory structure, refer to Volume I.

7.4 USING PATHNAMES

You assign pathnames to files when you create them. Figure 7-1 shows an example of a pathname for the file MYFILE residing on a disk named VOL1. Pathnames can contain the following identifiers to specify each file stored on a disk:

1. The volume name of the disk on which a file resides. (You assign the volume name when you initialize the disk by using the Initialize New Volume (INV) or the Initialize Disk Surface (IDS) command.)

2. The name of directories and subdirectories within which a file is cataloged on a disk volume.

3. The name of the file.

The limits on the size of a pathname are as follows:

- The total length of a pathname must be 48 characters or less (after all synonym substitutions) including an 8-character volume name.

- If the file is on a secondary disk (disk other than the system disk), you must include the volume name in the total length.

- All periods (.) used in the pathname are counted as part of the length.

You can construct pathnames and their aliases using either uppercase or lowercase English language alphanumeric characters. (You can assign aliases only to individual file components, either directories or the file. Aliases are discussed later in this section. You can use lowercase characters only if the lowercase option has been selected with the .OPTION SCI primitive. Refer to Volume III.)

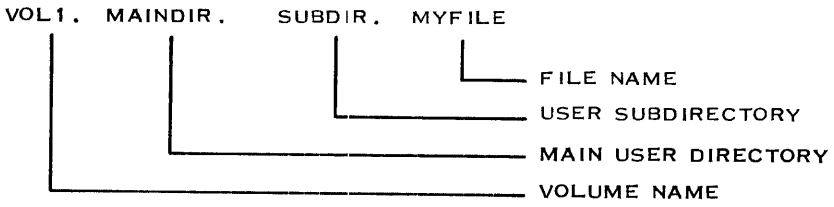


Figure 7-1. Format of a Complete Pathname

CAUTION

You can create files that you cannot access or move because the pathname can become too long. Note that the volume name may be omitted from the pathname of a file residing on the system disk. Suppose you create a file with a pathname 48 characters long beginning with a period such as the following:

.A345678.B345678.C345678.D345678.E345678.F345678

Assume the system disk name is Z and the disk is installed in DS02 (by using the Install Volume (IV) command). You cannot access the file because you must specify the following pathname that contains 49 characters:

Z.A345678.B345678.C345678.D345678.E345678.F345678

Since volume names can be eight characters in length, limiting pathnames in your applications to 40 characters guarantees that you are not denied access to your files because of pathname character length.

You must create directories before you can specify file names. The following paragraphs describe how to establish volume names, directories, subdirectories, and files.

7.4.1 Establishing Volume Names

The first element of every pathname is a volume name. Volume names are an alphabetic or numeric character string of as many as eight characters that identify the disk on which a file is found. The first character of a volume name must be an alphabetic character. The volume name by itself is the legal pathname of the volume directory.

You assign a volume name when you initialize a disk by using the INV or the IDS command. Once you initialize a disk with an IDS or INV command, you must include the volume name in the pathname to access files on that disk, unless the disk is the system disk.

7.4.2 Establishing Directories

You create the directory and subdirectory parts of each pathname when you establish a directory or subdirectory by entering the Create Directory File (CFDIR) command. The CFDIR command description explains the procedure for establishing directories and subdirectories in detail. You create directories and subdirectories when you want to group files into categories. You can create a subdirectory under a directory only after you first create the directory. For example, you can use the CFDIR command to establish a subdirectory with the following pathname:

VOL1.DIRA.DIRB

You can establish the pathname for the subdirectory only after you create a directory (using CFDIR) with the following pathname:

VOL1.DIRA

7.4.3 Establishing Files

After initializing a disk volume, creating directories, and creating subdirectories, you can create files that are accessible either directly under the volume or under a directory or subdirectories. You can create files by entering the following commands:

Command	Description
CF	Create File (requires that you select a file type)
CFIMG	Create Image File
CFKEY	Create Key Indexed File
CFPRO	Create Program File
CFREL	Create Relative Record File
CFSEQ	Create Sequential File

The descriptions for these commands describe the procedures for creating files in detail. For example, you can assign the following pathname to a file created under a single-level directory:

VOL1.JOESDIR.JOESFILE

7.5 FILE IDENTIFICATION AND PROTECTION

The following commands allow you to assign synonyms (abbreviations) to pathnames, modify synonyms, assign aliases to file names, delete aliases, change file names, and change protection for a file:

Command	Description
AA	Add Alias to Pathname
AS	Assign Synonym
DA	Delete Alias
LS	List Synonyms
MFN	Modify File Name
MFP	Modify File Protection
MS	Modify Synonym

An alias provides an alternate method of referring to a file. This can be very useful in applications where a file name or pathname is called out in the code of a program. You should use an alias pathname in the procedure or program itself, and then use the AA command to assign that alias to the file or directory of files as required. This is also helpful for testing purposes since you can assign the alias coded in the program to a set of files.

NOTE

Aliases are controlled by DX10 while synonyms are controlled by SCI. Aliases resemble pathnames, composed of directories, subdirectories, and files. Synonyms are single words containing no embedded periods, as explained later in this section.

Synonyms are usually shorter than their synonym value, although they can be longer. A common use of synonyms is to represent pathnames. A synonym can represent an entire pathname, or only the first part of a pathname. For example, if the character A is a synonym whose value is VOL1.DIR1.SUBDIR1 and MYFILE is a synonym whose value is VOL1.DIR1.SUBDIR1.FILE1 then the following character strings are equivalent responses to command prompting messages:

```
MYFILE
A.FILE1
VOL1.DIR1.SUBDIR1.FILE1
```

You cannot assign a value to a synonym that contains an embedded synonym, but SCI does accept embedded synonyms as responses to its prompts. If you attempt to assign a value to a synonym that contains an embedded synonym, the AS command accepts the value, but the embedded synonym does not get resolved properly when used elsewhere.

Although device types and file names must be entered into the system in uppercase English, lowercase or local language synonyms can be assigned to them, as follows:

Synonym	Synonym Value
DRüKKER	LP01
BüCH1	VOL1.DIR2.FILE1

NOTE

If the lowercase-to-uppercase option is selected at the terminal, no lowercase synonyms can be created.

When you create a synonym, DX10 automatically associates the synonym with the user ID under which you logged on (if log-on is required) or with the terminal that you are using (if log-on is not required). To use a synonym, you must log on under the user ID of the user who created the synonym, or you must use the same terminal where the synonym was created.

NOTE

Synonyms reserved for system use begin with the $ character. You should not define synonyms beginning with $. Refer to Volume III for more details.

7.6 VIEWING, LISTING, AND PRINTING FILES AND DIRECTORIES

The following SCI commands allow you to view or list the contents of directories and files, and to print and control output at a particular listing device:

Command	Description
HO	Halt Output
KO	Kill Output
LD	List Directory
LLR	List Logical Record
MD	Map Disk
MKF	Map Key Indexed File
MPF	Map Program File
PF	Print File
RO	Resume Output
SF	Show File
SOS	Show Output Status
SPI	Show Program Image
SRF	Show Relative to File

DX10 maintains a first-in-first-out (FIFO) queue for each output device in the system. The PF command and programs use this queue as follows:

- The PF command prints the contents of files at a printing device. Several PF command requests at once from several terminals in a system are put in a waiting line on that queue.

- Programs and SCI commands can use LP$x to queue output. Sending output to LP$x places it on the queue for printer LP0x, where x is a number from 1 through 9. The LP$x output is copied to a file with the name .S$PRINT.LPx##, where x is the printer number and ## is a unique number from 1 through 99. These files are automatically deleted after printing. LP$x output sent to a nonactive device is copied to a file in the S$PRINT directory but is not printed or deleted. These files must be manually printed by using the PF command and deleted by using the Delete File (DF) command.

You can view or change the status of output requests on the queue with the HO, RO, KO, and SOS commands.

7.7 COPYING FILES, DIRECTORIES, AND VOLUMES

You can use the following commands to copy the contents of files, directories, and volumes and to verify the copies made. These commands are presented in three groups: file, directory, and volume. The commands for files are as follows:

Command	Description
AF	Append File
CC	Copy/Concatenate
CKR	Copy KIF Randomly
CKS	Copy KIF to Sequential File
CPI	Copy Program Image
CSK	Copy Sequential File to KIF
CSM	Copy Sequential Media
MKL	Modify KIF Logging

The directory utility commands are as follows:

Command	Description
BD	Backup Directory
CD	Copy Directory
RD	Restore Directory
VB	Verify Backup or Restore
VC	Verify Copy

NOTE

The system treats volume names as directory names.

The volume commands are as follows:

Command	Description
CVD	Copy and Verify Disk
DCOPY	Disk Copy/Restore

You can use control files for directory utility commands. Control files are files containing directives and options that specify how the directory utilities are to copy a directory. Control files are discussed later in this section.

7.7.1 Directory Utility Commands

The CD, VC, BD, RD, and VB directory utility commands are provided in DX10 to copy, backup, restore, and verify directories and/or specific files under directories. These commands may be used to perform the indicated functions between disks of the same type or disks of different type.

NOTE

The CD, BD, RD, VC, and VB commands cannot process files with a physical record length greater than 10,000 bytes. Also, only direct unblocked I/O operations are supported.

You always specify files to copy or back up by giving a directory pathname. You can modify the choice of files and/or subdirectories to copy or back up by using control files. If you only want to copy or back up a single file, you can specify the total pathname for that file for the input. The directory utility converts this internally to a directory pathname with one INCLUDE statement. Therefore, the rules for INCLUDE statements discussed in the paragraph on control files apply. For output, you still specify the sequential name or output directory name.

Figure 7-2 shows a directory, .A, cataloged at the volume level of VOL1.

A directory copy operation (CD or BD followed by RD) can copy file set .A to file set .B (file set .B need not exist before the copy operation takes place).

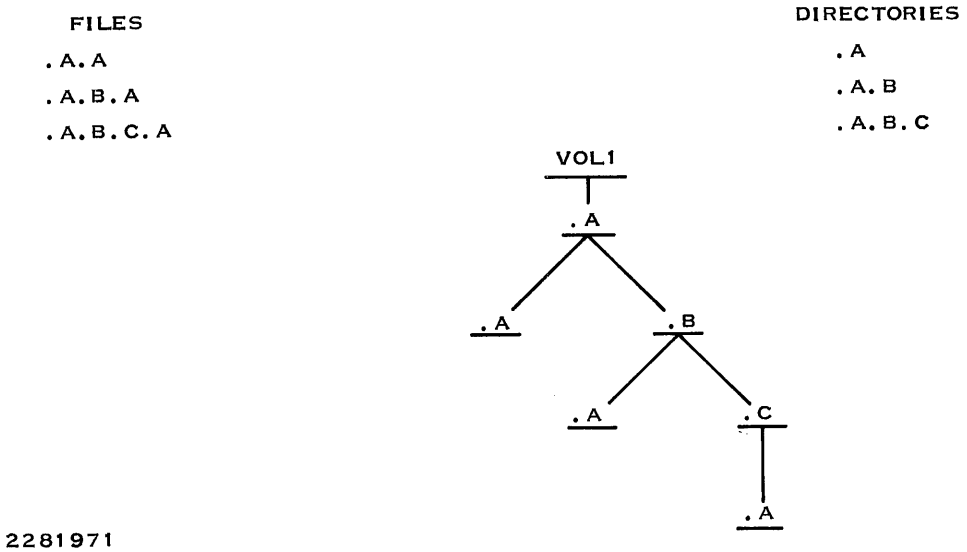


Figure 7-2. Sample File Set of a Directory

The directory copy operation .A to .B generates the file set for directory .B as shown in Figure 7-3.

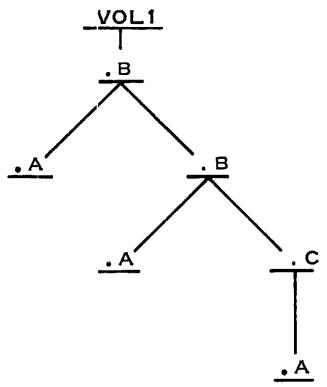


Figure 7-3. Sample File Set of a Directory After the CD Command

You can verify the results of the copy operation by using one of the verify commands that compares the file set of one directory with the file set of another directory and identifies the matches that it finds.

The BD, RD, and VB commands can process directories whose backup requires the capacity of more than one tape or disk volume. The first record of each disk/tape volume contains a header record for identification purposes. The header record consists of >A2 bytes in the following sequence:

- The ASCII characters '**HDR*'
- The time and date of the Backup Directory (in four words)
- The disk/tape volume number
- The block size in bytes (0 = no blocking)
- The sector size in bytes of the source
- Three reserved bytes
- A 41-byte ASCII time and date
- The literal value VOLUME = followed by a 9-byte volume number
- The literal value SECTOR SIZE = followed by a 4-byte value
- The literal value SOURCE = followed by a 57-byte source input name
- FAST, NOFAST flag

When the end of the disk/tape is encountered, the following message appears on your terminal:

"MOUNT VOLUME X. TYPE $ TO QUIT, Y TO CONTINUE"

where:

X is the disk or tape volume number.

You can terminate the command by pressing the dollar sign ($) key on your terminal, with control returning to SCI before processing is complete. If you do not want to terminate the command, mount the specified disk/tape. When the disk/tape is in the ready position, type Y for processing to continue. If the disk/tape is not in the ready position when you enter a Y to continue or if an error occurs (such as mounting the wrong disk/tape), the error message appears on your terminal and the mount disk/tape prompt reappears. If no errors are encountered upon entering the Y response, processing continues.

NOTE

The CD command copies one directory hierarchy to another directory, the BD command copies a directory hierarchy to a sequential medium, and the RD command copies a sequential medium to a directory hierarchy.

7.7.2 Control Files

The following discussion of control files applies to directory utility commands only. Control files are sequential files that contain special commands called directives and their parameters. You normally create control files through the Text Editor, althouth you can enter them directly from a terminal. Figure 7-4 shows a sample control file.

As Figure 7-4 shows, you can position directives anywhere on a line in a control file and you can abbreviate them. You must begin comments with exclamation points. Table 7-2 describes the directives that you can use in control files.

```
OPTION REPL
MOVE VOL1.MYFILES
EX LIST,CONTROL
O ADD
MOVE VOL1.HISFILES
I       A
IN      B
INC     C
INCL    X
I       PROG
END
```

Figure 7-4. Sample Control File

The CD, VC, RD, VB, and BD commands do not operate on the directory .S$PRINT and the files named .S$ROLLA, .S$DIAG, and .S$CRASH unless you specify them with INCLUDE directives. You can supply options by using OPTION directives in control files, or you can specify them in response to the OPTIONS prompt.

MOVE directives that you specify for RD and VB should correspond to MOVE directives for the original BD. If you did not specify MOVE directives for BD, you cannot use them for the resulting RD and VB commands. If you did specify MOVE directives for BD, you must use corresponding MOVEs for RD and VB to ensure that all files backed up will be restored or verified.

MOVE directives that you specify for VC should correspond to MOVE directives for the original CD. If you did not specify MOVE directives for CD, you cannot specify them for the resulting VC command. If you did specify MOVE directives for CD, you must use corresponding MOVEs for VC to ensure that all files copied will be verified.

You can use synonyms for pathnames or portions of pathnames only in the MOVE directive. You cannot use aliases as source pathnames in MOVE, EXCLUDE, or INCLUDE directives.

Table 7-2. Control File Directives

Directive	Purpose

MOVE < dirpath> ,< seqpath>
or
MOVE < seqpath> ,< dirpath>
or
MOVE < mastpath> ,< copypath>
	For a CD command, the MOVE directive specifies a pathname (or synonym representing a pathname) of a source file or directory (mastpath) that you want to copy to a destination directory (copypath). For example, a MOVE directive appears in a control file as follows:
	
	MOVE VOL1.DIR1.FILE1,VOL3.DIRC
	
	For a BD command, the MOVE directive specifies the pathname of a source file or directory (dirpath) that is copied to a destination sequential file, disk, or magnetic tape (seqpath); this is the copy access name. For example:
	
	MOVE VOL1.DIR1,MT01
	
	For VB and RD commands, the MOVE directive specifies the source sequential file, disk, or magnetic tape device (seqpath) and the destination pathname (dirpath). For example:
	
	MOVE MT01, VOL1.DIR1
	
	For a VC command, the MOVE directive specifies a pathname of a master file or directory (mastpath) that you want to verify against a copy file or directory (copypath).
	
	You can place the destination pathname on the next line of the control file.

Table 7-2. Control File Directives (Continued)

Directive	Purpose
MOVE < ,destpath>	This MOVE directive defines a new destination directory but the source specified in the previous MOVE directive or the source specified in the command prompts is used.
MOVE < sourcepath>	This MOVE directive defines a new source but the destination specified in the previous MOVE directive or the destination specified in the command prompts is used. **NOTE** Files or directories specified in response to the command prompts for source and destination are copied before files and directories specified in a MOVE directive are copied.
EXCLUDE < (filename)>	A file name is one to eight alphanumeric characters containing no periods. Each EXCLUDE directive names one or more data files or subdirectories (separated by commas) within the directory being copied that are not to be copied, backed up, restored, or verified. Legal file names are either directories or files that appear in the list produced by a List Directory (LD) command. An alias is not a legal file name. If an EXCLUDE directive specifies more than one file name, they are separated by commas. For example, the following EXCLUDE directive specifies that all files except FILE1, FILE2, and FILE3 are to be copied or verified: EXCLUDE FILE1,FILE2,FILE3 You can exclude a maximum of 50 directory members by the use of one or more EXCLUDE directives. EXCLUDE directives that apply to a MOVE directive must immediately follow the MOVE directive in the control file. An EXCLUDE directive cannot be assigned to a MOVE directive to which an INCLUDE directive is also assigned.

Table 7-2. Control File Directives (Continued)

Directive	Purpose
INCLUDE < (filename)>	A file name is from one to eight alphanumeric characters containing no periods. The INCLUDE directive specifies that no files or subdirectories, within the directory specified, are to be copied, backed up, restored, or verified except those specified by INCLUDE directives. Legal file names are either directories or files that appear in the list produced by an LD command. An alias is not a legal file name. An INCLUDE directive cannot follow a MOVE directive if an EXCLUDE directive already follows that MOVE directive. File names specified by an INCLUDE directive are separated by commas. For example, the following INCLUDE directive specifies that only FILE1, FILE2, and FILE14 are to be copied or verified: INCLUDE FILE1,FILE2,FILE14 You can include a maximum of 50 directory members by using one or more INCLUDE directives. INCLUDE directives that apply to a MOVE directive must immediately follow the MOVE directive in the control file. If you need to move more than 50 directory members, you can specify more than one MOVE directive with the same source and destination pathnames.
OPTION option	You can specify OPTION directives anywhere in a control file except between EXCLUDE directives. You can use the OPTION directive to specify any option that is legal for the command. The options that can be specified by means of the OPTION directive include the following:

Option*	Purpose
ADD	Files are to be copied or restored unless a file with the same name and at the same level already exists in the destination directory. The ADD option is ignored by the Verify Copy (VC) command.
REPLACE	Files are to be copied or restored even if a file with the same name and at the same level already exists on the destination directory. The existing file is deleted and replaced with the file being copied. The REPLACE option is ignored by the Verify Copy (VC) command.

Note:

* Options may be abbreviated provided the abbreviation is unambiguous. Since A could represent either ADD or ALIASES, it is ambiguous and should be replaced with AD or AL.

Table 7-2. Control File Directives (Continued)

Directive	Purpose
Option*	**Purpose**
ALIASES	All aliases associated with a file being copied, backed up, restored, or verified are to be copied, backed up, restored, or verified unless an alias already exists with the same name and at the same level in the destination directory.
NOALIASES	No aliases are to be copied, backed up, restored, or verified.
BLOCK	Entry of BLOCK specifies that records will be grouped in blocks of a certain size. When used as a control directive option, the blocking factor used is the initial value of 9600 bytes. If you desire another blocking factor, you must specify BLOCK in response to the OPTIONS prompt.
NOBLOCK	Entry of NOBLOCK specifies that each record is written separately without blocking.
REWIND	Entry of REWIND causes rewinding of tape volumes before they are used.
NOREWIND	Entry of NOREWIND indicates tape volumes are not rewound before use.
UNLOAD	Entry of UNLOAD indicates tape volumes are unloaded after use.
NOUNLOAD	Entry of NOUNLOAD indicates tape volumes are not unloaded after use.
CMP	Entry of CMP indicates unused physical blocks at the end of KIFs are truncated. Both CMP and NOCMP collect secondary allocations together. This option only applies to the CD and RD commands.
NOCMP	Entry of NOCMP indicates unused physical blocks at the end of KIFs are not truncated. This option only applies to the CD and RD commands.
FAST	Entry of FAST indicates the BD command copies the source directory as one file. This option also affects the RD command. For a detailed description, refer to the BD and RD command descriptions.

Note:

* Options may be abbreviated provided the abbreviation is unambiguous. Since A could represent either ADD or ALIASES, it is ambiguous and should be replaced with AD or AL.

Table 7-2. Control File Directives (Continued)

Directive	Purpose

Option*	Purpose
NOFAST	Entry of NOFAST indicates the BD command copies each task, procedure, or overlay image separately. This option also affects the RD command. For a detailed description, refer to the BD and RD command descriptions.
SYSFILE	Entry of SYSFILE indicates that system files are included in the operation that the command performs.
NOSYSFILE	Entry of NOSYSFILE indicates that system files are not included in the operation that the command performs.

Notes:

If no options are specified with an OPTION directive, it continues to use the previous options and a warning message is printed on the listing file.

* Options may be abbreviated provided the abbreviation is unambiguous. Since A could represent either ADD or ALIASES, it is ambiguous and should be replaced with AD or AL.

7.7.3 Volume Commands

The following volume commands copy and optionally verify entire disk volumes used in the DX10 operating system:

Command	Description
CVD	Copy and Verify Disk
DCOPY	Disk Copy/Restore

The CVD command copies from disk to disk. The DCOPY command copies from disk to magnetic tape, magnetic tape to disk, or disk to disk. The principle advantages of these commands are that they do not require the system disk to be physically installed and they are fast. The disadvantage is that the output disk must be the same size and type as the input disk.

7.8 CONTROLLING TERMINAL ACCESS

You can control access to DX10 by controlling the status of terminals in the system and by controlling user identification. The commands associated with controlling terminal status and user identification are as follows:

Command	Description
AUI	Assign User ID
DUI	Delete User ID
LTS	List Terminal Status
LUI	List User IDs
MTS	Modify Terminal Status
MUI	Modify User ID
STI	Show Terminal Information

DX10 installations that require log-on at terminals must provide users with user IDs and passcodes. Users identify themselves to DX10 by entering their ID and passcode in response to prompting messages during the log-on sequence. If DX10 recognizes the ID and the passcode as valid, it lets the user operate the terminal at the privilege level preassigned to the user ID. The paragraphs on the AUI, MUI, and LUI commands describe how installations can assign, modify, and list user IDs, passcodes, and privilege levels.

7.9 LOGICAL UNIT ASSIGNMENT AND CONTROL

Application programs under DX10 need not name specific devices or files to which output goes or from which input comes. Instead, they can specify logical unit numbers (LUNOs) to represent input and output devices. A LUNO is assigned to a device or file by a LUNO assignment command from you or from an executing program. The types of LUNO assignments that can be made are as follows:

LUNO Assignment Type	Scope of Assignment
Global	When a file or device must be available for input/output operations from more than one station, it is assigned to a global LUNO by the Assign Global LUNO (AGL) command. A file or device assigned to a global LUNO is available for input/output operations initiated by any station.
Station-local	When a file or device needs to be available for input/output operations from only one station, it is assigned to a station-local LUNO by using the Assign LUNO (AL) command.
Task-local	Executing programs can make LUNO assignments for their own input/output operations. Those task-local LUNO assignments do not require command entries by the user.

LUNO assignments are made according to the needs and efficiencies of each program. Volume III describes situations in which you would prefer one type of assignment over another.

The following commands are available to assign global and station-local LUNO assignments:

- Assign Global LUNO (AGL)
- Assign LUNO (AL)

The following commands are available to release global and station-local LUNO assignments:

- Release Global LUNO (RGL)
- Release Station-Local LUNO (RL)
- Release All Station-Local LUNOs (RAL)

Task-local LUNOs may be released by the task, or if they are not, they are automatically released by DX10 when a task terminates. Station-local LUNOs are released when SCI is terminated at the station.

The following commands are available for use in conjunction with LUNOs:

- Backspace LUNO (BL)
- Forward-Space LUNO (FL)
- Modify LUNO Protection (MLP)
- Rewind LUNO (RWL)
- Show I/O Status (SIS)
- Write EOF to LUNO (WEOF)

7.10 BATCH MODE OF SCI

All interactive tasks running under DX10 operate in foreground mode. In foreground mode, SCI expects input from terminals and directs output to terminals during execution. A background mode of operation is also available under DX10. In background mode, SCI accepts commands from an input file or from an input device, but not from a terminal. Entry of any of the following commands starts a process in background mode, after you enter initial parameters:

- Execute Batch (XB)
- Execute Link Editor (XLE)
- Execute Macro Assembler (XMA)

In addition to these commands, any other procedure coded by the user or modified to use the SCI primitive .QBID also starts a process in background. Refer to Volume III for information on SCI primitives.

You can enter any of these commands to initiate activity in background. Other command(s) entered may execute simultaneously in foreground. Background operations do not interfere with foreground operations. When the background operation terminates, it sends an ending message to the terminal that initiated the operation. The ending message is not displayed or printed at the terminal until SCI goes through a major loop, which is when SCI goes back to primary input from the terminal. (That is, the foreground task completes, you press the CMD key in text edit mode, you press the CMD key in show file mode, or you press the CMD key to stop entering answers to prompts.)

The only background mode command that this manual describes is the Execute Batch (XB) command. You enter an XB command to activate SCI in the background mode. All DX10 commands supplied in a batch stream must be in a batch format wherein all keyword and parameter information is provided with the command. No additional information can be supplied to the commands from terminals. In addition, the commands included must be suitable for execution in background mode. The following are not suitable for inclusion in a batch command stream:

- All debug and Text Editor commands
- Activate Task (AT) command
- Kill Background Task (KBT) command
- Modify Synonyms (MS) command
- Show Background Status (SBS) command
- Execute Debugger (XD) command
- Execute Text Editor (XE) command
- Execute and Halt Task (XHT) command

In addition to describing the XB command, this manual describes the SBS, KBT, BATCH, EBATCH, and WAIT commands that you can enter to view and control the status of a background task.

7.11 SYSTEM INITIALIZATION

After loading DX10 and activating SCI at a terminal, you can enter the Initialize System (IS) command to ready DX10 for operation. The IS command initializes the system with information that you enter and information supplied to the system by the IS command itself. The IS command incorporates the Initialize Date and Time (IDT) and Initialize System Log (ISL) commands.

You can also enter the IDT command to reset the system date and time. You can enter the ISL command to reset the system log.

7.12 LISTING DEVICE AND SYSTEM STATUS

The following commands allow you to list device and system status. These commands list the devices specified during system generation, read the contents of a CRU register, view the current date and time, and show the status of global, station-local, and task-local LUNOs. These commands also display the allocation of main memory, obtain information on system memory usage and SCI activity, show the status of queued output, display the allocation of the system table area, display the status of a specific task or of all tasks currently active in the system, and write a value to a CRU address. These commands are as follows:

Command	Description
LDC	List Device Configuration
RCRU	Read Contents of Specified CRU Register
SDT	Show Date and Time
SIS	Show I/O Status
SMM	Show Memory Map
SMS	Show Memory Status
SOS	Show Output Status
SSTM	Show System Table Map
STS	Show Task Status
WCRU	Write Value to Specified CRU Address

7.13 ADDING AND SHOWING EXPANDED ERROR MESSAGES

The following two commands enable expanded error message explanations to be displayed at your terminal:

Command	Description
AEM	Add DX10 Error Message
SEM	Show DX10 Error Message

The SEM and AEM procedures are optional. They can be installed after a custom system has been generated and tested. See the *Model 990 Computer DX10 Operating System Object Installation* document for installation instructions.

The expanded message files are key indexed files (KIFs). Therefore, KIF logic must be selected during the custom system generation before the SEM and AEM procedures can be installed or used.

7.14 SENDING MESSAGES TO TERMINALS

The following commands allow you to send a message to another terminal and display a message at a terminal:

Command	Description
CM	Create Message
MSG	Display Message

7.15 TEXT EDITOR COMMANDS

The following list shows the Text Editor commands. Refer to Volume IV for a detailed discussion of the Text Editor commands.

Command	Description
CL	Copy Lines
DL	Delete Lines
DS	Delete String
FS	Find String
IF	Insert File
ML	Move Lines
MR	Modify Roll
MRM	Modify Right Margin
MT	Modify Tabs
QE	Quit Edit
RE	Recover Edit
RS	Replace String
SL	Show Line
SVL	Save Lines
XE	Execute Text Editor
XES	Execute Text Editor with Scaling

7.16 LINKING AND INSTALLING PROGRAMS

The following commands are provided to link and install programs:

Command	Description
AL	Assign LUNO
DO	Delete Overlay
DP	Delete Procedure
DT	Delete Task
IO	Install Overlay
IP	Install Procedure
IRT	Install Real-Time Task
IT	Install Task
XLE	Execute Link Editor

7.17 PROGRAM EXECUTION AND CONTROL

Programs running under DX10 are called tasks. Programs become tasks when they are activated by one of the following interactive SCI commands: Execute Task (XT), Execute and Halt Task (XHT), and Execute Task and Suspend SCI (XTS). The XT command activates tasks that do not interact with terminals in the system. The XTS command activates tasks that interact with the terminal. The XHT command is used when you are debugging the task. When programs are activated, they are assigned a run-time ID by DX10, unless they are not replicatable. Refer to Volume III for an extensive discussion of how DX10 handles programs.

Once a task is executing, you can terminate it by using the Kill Task (KT) command. The Kill Background Task (KBT) command kills a task executing in background. The Show Task Status (STS) command shows the status of a specified task or the status of all tasks currently running in the system.

The following commands provide program execution and control:

Command	Description
KBT	Kill Background Task
KT	Kill Task
STS	Show Task Status
XT	Execute Task
XHT	Execute and Halt Task
XTS	Execute Task and Suspend SCI

7.18 MODIFYING PROGRAM ATTRIBUTES

SCI provides commands to change the information supplied when an overlay, task, or procedure was installed. The modifications can also be made to modules installed by the Link Editor. These commands allow you to modify the run-time environment of a program without reinstalling the module. These commands are as follows:

Command	Description
MOE	Modify Overlay Entry
MPE	Modify Procedure Entry
MTE	Modify Task Entry

Refer to Volume III for more information on these commands.

7.19 SYSTEM MAINTENANCE COMMANDS

The following commands are provided for system maintenance:

Command	Description
ALGS	Assemble and Link Generated System
CKD	Check Disk for Consistency
CSF	Create System Files
IGS	Install Generated System
ISO	Install System Overlay
MDS	Modify Device State
PGS	Patch Generated System
Q$SYN	Clear Secret Synonyms
RCD	Recover Disk
SD	Scan Disk
TGS	Test Generated System
XANAL	Execute Crash Analyzer
XCU	Execute Conversion Utility
XGEN	Execute System Generation Utility
XPS	Execute Patch Synonym Processor

7.20 TX/DX FILE CONVERSION UTILITY

The TX/DX file conversion utility commands provide you with the capability to: convert DX10 files to TX990 diskette files, convert TX990 diskette files to DX10 files, initialize a diskette, list the diskette directory, specify the system file, compress a diskette file, delete a diskette file, or change the level of diskette file protection.

The TX/DX file conversion utility does not support conversion of TIPE-990 files or IBM diskettes. For conversion of DX10 files to TIPE-990 files, consult your TIPE-990 documentation. For conversion of IBM files, refer to the IBMUTL command description.

The TX/DX file conversion commands are as follows:

Command	Description
DXTX	Convert DX10 File to TX990 Diskette File
IBMUTL	Convert IBM-Formatted Diskette Datasets
TXCM	Compress Diskette File
TXCP	Change Diskette File Protection
TXDF	Delete Diskette File
TXDX	Convert TX990 Diskette File to DX10 File
TXFD	Format Diskette
TXMD	Map Diskette
TXSF	Specify the System File
TXXLE	Execute Link Editor to Produce TX Program File

Should an error occur during command execution, DX10 generates a coded error message of the following form or a specific textual error message:

XX — 'error message'

where:

XX is the internal error code.

An example of a specific error message is as follows:

30 — READ ERROR

The error code and message text indicate that the error is an attempt to read a nonexistent record from a diskette file.

7.21 REMOTE TERMINAL UTILITIES

The Remote Terminal utilities facilitate communications between the Texas Instruments 990 DX10 system and hard-copy terminal products. The hard-copy terminal products include the Models 743, 745, 781, 783, 785, 787, 820, and the bubble memory terminals 763 and 765. You can do the following using these utilities:

- Call a remote hard-copy terminal
- Monitor incoming calls from remote hard-copy terminals
- Disconnect a line with a remote hard-copy terminal
- List communications port characteristics
- Modify defined port characteristics

All of the utilities are executed through SCI commands. The following list describes these commands:

Command	Description
CALL	Call Terminal
ANS	Answer Incoming Call
DISC	Terminal Disconnection
LHPC	List Hard-Copy Terminal Port Characteristics
MHPC	Modify Hard-Copy Terminal Port Characteristics

The utilities operate with communications ports that must be defined at system generation time as teleprinter device (TPD) ports.

7.21.1 Line Control

The Remote Terminal utilities are used with leased, switched, privately owned, or local lines. Either half-duplex or full-duplex can be used. Appendix C lists the hardware environments used with the teleprinter device ports.

7.21.2 Naming Conventions

You initiate these utilities by using SCI commands. You can use the utilities at any terminal supporting SCI. The controlling terminal is called the *command terminal*; the controlled port and the controlled terminal are called the *target port* and *target terminal*. The terminal access name identifies the target port. The form for the terminal access name is STXX, which is the station identifier assigned to a communications port during system generation. The station identifier identifies not only the communications port but also the terminal communicating through that port. When connected to the port, the terminal is an extension of the port. Ports used to communicate with the hardcopy terminals must be defined as type TPD at the time of system generation.

The terminal is accessed through the station identifier. Since a station identifier does not identify a particular terminal (if the port is a switched network port), each terminal can also have an identifier called an answerback. If used, this identifier is a set of characters that the terminal transmits upon request.

Answerbacks are assigned during the offline configuration phase for the terminal. Some terminals may have their answerbacks coded in a ROM memory when they are manufactured, while other terminals have modifiable answerback memories which are initialized by the terminal installer or operator. Consult the operators guide for the particular terminal being used for information on how the answerback value is assigned.

7.21.3 Custom Program Applications

It is possible to prepare variations on the functions of the remote terminal utilities by preparing custom SCI procedures. In particular, custom procedures can take advantage of the synonyms set by some of the tasks and tune the parameters used by some tasks.

7.21.3.1 Synonyms. The synonym ABM is updated by the CALL and ANS utilities. The value of ABM can be tested by a custom-written program for a special application. Any action taken by the custom program, after testing the synonym, is directed by the particular application.

One synonym application involves monitoring dial-in connections using a modified ANS utility. The answerback is requested and saved as ABM, which can be compared through a custom task or SCI command to a list of valid users. If the answerback value is not on the list, the terminal is disconnected by using the DISC utility.

The system completion code synonym, $$CC, is updated by all of the utilities and can be tested in a custom program. Different synonym values imply different actions based on the application. The error code values are in Volume VI.

7.21.3.2 Tuning Parameters. The CALL and ANS utilities have parameters that are passed to a task. The parameters of the task are coded into the procedure as opposed to being supplied by the operator. These parameters are called *tuning parameters*.

CALL. You can alter the following utility characteristics by modifying the SCI procedure that defines the CALL command. The set of parameters is comprehensive, allowing for international constraints and various manufacturers' automatic call units. The list applies to automatic dialing unless otherwise noted. The tuning parameters are listed in the command procedure beginning with the fourth parameter in the PARMS clause of the .BID line. Each parameter is represented in the clause as a number. The parameters appear in the following order:

Maximum delays in system time intervals to wait for the following events:

1. Next digit
2. Primary dial tone
3. Secondary dial tone
4. Answerback (automatic dial or manual dial)
5. Data Set Ready to go true after Data Set Status is set true
6. Present Next Digit to go false after Digit Present is set true

Maximum delays in seconds to wait for the following events:

7. Data Link Occupied to go false at the dialing start
8. Connection (automatic dial)
9. Connection (manual — zero means infinite)

Time Delay in system time intervals between the following:

10. The Assertion of Call Request and the Data Terminal Ready

Time delays in seconds between the following sets of events:

11. A failed attempt and a retry
12. Data Link Occupied going false and assertion of Call Request
13. Receipt of primary dial tone and dialing the first digit
14. Receipt of secondary dial tone and dialing the next digit

Maximum number of tries to accomplish the following:

15. Establish a valid connection
16. Read an answerback (automatic dial or manual dial)

Flags: A zero value is interpreted as no, and a nonzero value is interpreted as yes. The flags indicate whether to use the following items:

17. End of number after last digit
18. Pulse dialing if using TI internal ACU
19. Even parity testing on answerback. Zero is interpreted as no parity test and nonzero as use even parity test
20. Assert request to send when dialing on half-duplex circuits
21. Save answerback in ABM even if verification is not requested

NOTE

If you set the maximum time for manual dial completion to zero (infinite delay) and the manual dial cannot be completed, you can only exit the task by killing it from another terminal.

ANS. The following areas can be changed by modifying the procedure. The tuning parameters are listed in the command procedure in the following order:

1. Maximum delay in system time intervals for answerback.
2. Maximum delay in seconds waiting for connect. Zero is interpreted as an infinite delay.
3. Number of tries to read an answerback.
4. Use even parity testing on answerback. Zero is interpreted as no parity test, and nonzero is interpreted as use even parity test.
5. Save answerback in ABM even if verification is not requested. Zero means do not save unless verification is requested.

7.22 DEBUG COMMANDS

The following lists describes the debug commands. For more information on these commands, refer to Volume III.

Command	Description
AB	Assign Breakpoints
ASB	Assign Simulated Breakpoints
AT	Activate Task
DB	Delete Breakpoints
DPB	Delete and Proceed from Breakpoint
DSB	Delete Simulated Breakpoint
FB	Find Byte
FW	Find Word
HT	Halt Task
LB	List Breakpoints
LLR	List Logical Record
LM	List Memory
LSB	List Simulated Breakpoints
LSM	List System Memory
MAD	Modify Absolute Disk
MADU	Modify Allocatable Disk Unit
MIR	Modify Internal Registers
MM	Modify Memory
MPI	Modify Program Image
MRF	Modify Relative to File

Command	Description
MSM	Modify System Memory
MWR	Modify Workspace Register
PB	Proceed from Breakpoint
QD	Quit Debug
RST	Resume Simulated Task
RT	Resume Task
SAD	Show Absolute Disk
SADU	Show Allocatable Disk Unit
SIR	Show Internal Registers
SP	Show Panel
SPI	Show Program Image
SRF	Show Relative to File
ST	Simulate Task
SV	Show Value
SWR	Show Workspace Registers
XD	Execute Debugger

8

DX10 SCI Commands

8.1 INTRODUCTION

This section describes the DX10 SCI commands in alphabetical order. The SCI commands allow you to interface with DX10 interactively or in batch mode. You can enter these commands at all DX10 interactive devices, including hard-copy data terminals and video display terminals (VDTs).

This manual describes the commands in a standard format that includes the following information:

- The command format
- User responses to the command prompting messages (both required and optional)
- Example of how to use the command

Table 6-1 lists all SCI commands available in the DX10 software package. Section 7 presents a description of the functional groups of the SCI commands. The commands appear in a logical order that you might need to execute them to begin using the DX10 system and developing applications. Tables 4-6 and 4-7 show the keys to which SCI gives special control functions when it is accepting your responses to prompts in VDT and TTY mode, respectively.

AA

8.2 AA (ADD ALIAS TO PATHNAME)

The AA command assigns aliases (alternate names) for files or directories. In response to the prompts of the AA command, you enter the pathname of a file or directory and another pathname that includes, as its last element, the alias by which you can also identify the file or directory. Both pathnames must be identical except for the file name or the directory name that is the last element of both.

8.2.1 AA Command Format

```
AA
ADD ALIAS TO PATHNAME
PATHNAME: <acnm> @
ALIAS PATHNAME: <acnm> @
```

8.2.2 AA Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter a pathname that identifies an existing file or directory.
ALIAS PATHNAME:	R	Enter the same pathname as above except for the last element which is the alias (alternate) name by which you can identify the file or directory.

8.2.3 AA Command Example

The following example shows how you can use the AA command to assign the alias FILEB to a file named FILE1:

```
[ ] AA
ADD ALIAS TO PATHNAME
 PATHNAME: VOL1.PROGDIR.DIRA.FILE1
 ALIAS PATHNAME: VOL1.PROGDIR.DIRA.FILEB
```

8.2.4 Commands Related to AA

DA (Delete Alias from Pathname)

8.3 AB (ASSIGN BREAKPOINTS)

The AB command replaces the specified contents of the address(es) of a task with a breakpoint (an XOP 15,15 instruction), which stops execution of the task at that location. Refer to Volume III for a detailed description.

8.4 AEM (ADD DX10 ERROR MESSAGE)

The AEM command enables you to add, change, or delete an expanded error message explanation. You cannot use this command with hard-copy terminals (teleprinter devices); therefore, you should only invoke the command from a 24-line VDT.

8.4.1 Adding an Error Code

DX10 allows you to enter an error code, the error message text, and an explanation of that message. To add an error message explanation, you should enter AEM and then press the RETURN key in response to the SCI prompt symbol ([ ]). The message ADD DX10 ERROR MESSAGE appears and then the following appears:

```
                    ADDING ERROR CODES TO THE SYSTEM

PLEASE ENTER THE 4 CHARACTER ERROR CODE: _____

TO STOP ADDING ERROR CODES ENTER AN 'XXXX' FOR THE ERROR CODE
```

Enter the four-character error code for which you want an explanation. (You must enter four characters.)

When you enter the four-character error code, the following appears. (2019 is an example of user input.)

```
ADDING ERROR CODE 2019
PLEASE ENTER THE MESSAGE

_____________________________________

ENTER A '?' TO BYPASS THIS ERROR CODE
```

Enter the error message text (for example, ILLEGAL STATION NUMBER) and terminate each line of text by pressing the RETURN key. You can enter up to 10 lines of message text.

If you decide not to add the specified error code to the file, enter a question mark (?) in the first character position of a message text line. Any text that you entered is cleared, and the cursor is positioned at the beginning of the error code field.

After you completely enter the message text, you should press the RETURN key when the cursor is in the first character position of a blank line. (If the cursor is in the first character position of a line with text, the text on that line is discarded.) The following questions appear one at a time. Answer each of these questions with an uppercase Y for yes and N for no. You can enter a question mark (?) to abort entry of this error code.

```
IS THIS A USER'S ERROR (Y/N)? _____

IS THIS A SYSTEM ERROR (Y/N)? _____

IS THIS A HARDWARE ERROR (Y/N)? _____
```

If you exceed the end of a line while entering text for the error message or for the corrective action, the cursor automatically moves to the first position of the next line. Any additional typed characters are entered beginning at that position. This may cause a word to be split between lines. Entering the Show DX10 Error Message (SEM) command to display the resulting message shows the word split between lines exactly as it appeared when you entered it. To avoid splitting words, terminate each line by pressing the RETURN key at a convenient stopping point before the end of the line is reached. Also, when entering text, the cursor can backspace only on the line on which the cursor currently appears. The cursor cannot back up to the previous line. You can make corrections to the previous line only by deleting the error code and adding it again.

After you answer the last question, the following prompt appears, so that you can enter an explanation of the message. You can use numerous lines of 40 characters each.

```
PLEASE ENTER THE ERROR MESSAGE

_____________________________________
```

Enter the text of the message explanation and terminate each line by pressing the RETURN key.

After you enter the entire explanation text, press the RETURN key when the cursor is in the first character position of a blank line. The following prompt then appears, so that you can specify what corrective action should be taken. You can use numerous lines of 40 characters each. If you decide not to add the specified error code to the file, enter a question mark (?) in the first character position of a message text line. Any text that you entered is cleared, and the cursor is positioned at the beginning of the error code field.

```
PLEASE ENTER THE ACTION

_____________________________________
```

Enter a description of the corrective action required and terminate each line by pressing the RETURN key.

After you enter the entire action text, press the RETURN key when the cursor is in the first character position of a blank line.

The initial AEM prompts then appear so that you can enter another four-character error code. If you have no more error codes to enter, enter XXXX (four uppercase Xs); the system then displays the main SCI menu.

AEM

8.4.2 Deleting an Error Code

To delete an error code, enter the AEM command and the error code as if adding the error code to the system. Since the error code already exists, the following is displayed:

```
ERROR: THIS ERROR NUMBER ALREADY FOUND ON FILE WITH MESSAGE:

IF THIS IS NOT THE ERROR MESSAGE YOU WERE GOING TO ADD,
YOU MUST ENTER THIS ERROR CODE AS A DUPLICATE.

WOULD YOU LIKE TO ADD THIS AS A DUPLICATE (Y/N)? _____

ENTER A 'D' TO DELETE ENTRY (displayed at bottom of screen)
```

In response to WOULD YOU LIKE TO ADD THIS AS A DUPLICATE?, you should enter D. The error code is then deleted and the initial AEM menu is displayed.

8.4.3 Adding a Duplicate Error Code

To add a duplicate error code, follow the instructions given in the preceding paragraph and answer Y to the question WOULD YOU LIKE TO ADD THIS AS A DUPLICATE?.

8.4.4 Changing an Error Message, Explanation, or Action

To change an error message, explanation, or action, you must delete the error code and add it again with the changed information as described in preceding paragraphs.

8.4.5 Commands Related to AEM

SEM (Show DX10 Error Message)

8.5 AF (APPEND FILE)

The AF command can append one file to another file or append input from a sequential input device to a file. The AF command has the following characteristics:

- Sequential disk files, relative record disk files, or input from sequential devices can be appended to a sequential disk file or a relative record disk file.
- Output cannot be made to magnetic tape.
- Input appended to a file is added immediately after the last record of the file, writing over (destroying) the last end-of-file marks in the file.
- A new end-of-file (EOF) mark is added after the appended information.
- Directories, program files, image files, and key indexed files cannot be appended using the AF command.

8.5.1 AF Command Format

```
AF
APPEND FILE
INPUT ACCESS NAME(S): (< acnm>) @
OUTPUT PATHNAME: < acnm>  @
MAXIMUM RECORD LENGTH: [int]
```

8.5.2 AF Command User Responses

System Prompts	Response Required or Optional	User Responses
INPUT ACCESS NAME(S):	R	Enter one or more pathnames or device names (separated by commas) that identify the source of information to be appended to the output file. The number of files that can be appended is limited by the number of pathnames, device names, or synonyms that can fit in the space provided. You can enter 50 characters in this space interactively. If you use expert or batch mode, you can enter up to 235 characters, including commas and periods. However, if you use synonyms in response to this prompt, the expanded length can be up to 48 bytes.

AF

System Prompts	Response Required or Optional	User Responses
OUTPUT PATHNAME:	R	Enter a pathname that identifies the file to which DX10 appends the specified file or files, or enter the information input from a sequential device. If the specified file does not exist, the AF command automatically creates it.
MAXIMUM RECORD LENGTH:	O	Enter the maximum number of bytes to transfer per I/O operation. This value is the longest record of the input file that the Copy/Concatenate (CC) command does not truncate. Records in the input file, up to this value, are always copied with the exact same length appearing in the output file. Longer records are truncated to this value. *The default value is 1024 bytes.*

8.5.3 AF Command Example

The following example shows the contents of four files, the appending of three of those files onto the first file, and the listing of the new contents of the first file:

```
[] SF

SHOW FILE
 FILE PATHNAME:    .TEST

RECORD NUMBER ONE
RECORD NUMBER TWO
RECORD NUMBER THREE

[] SF

SHOW FILE
 FILE PATHNAME: VOL1.FOUR

RECORD NUMBER FOUR

[] SF

SHOW FILE
 FILE PATHNAME: VOL1.FIVE

RECORD NUMBER FIVE

[] SF

SHOW FILE
 FILE PATHNAME: VOL1.SIX

RECORD NUMBER SIX
```

```
[] AF

APPEND FILE
 INPUT ACCESS NAME(S):    VOL1.FOUR,VOL1.FIVE,VOL1.SIX

 OUTPUT PATHNAME:    .TEST

 MAXIMUM RECORD LENGTH:    80

APPEND COMPLETED

[] SF

SHOW FILE
 FILE PATHNAME: .TEST

RECORD NUMBER ONE
RECORD NUMBER TWO
RECORD NUMBER THREE
RECORD NUMBER FOUR
RECORD NUMBER FIVE
RECORD NUMBER SIX
```

8.5.4 Commands Related to AF

BD	(Backup Directory)
CC	(Copy/Concatenate)
CD	(Copy Directory)
RD	(Restore Directory)

AGL

8.6 AGL (ASSIGN GLOBAL LUNO)

The AGL command assigns a device or file to a logical unit number (LUNO) that is accessible by more than one station. If a device or file is currently assigned to the requested global LUNO, an error is returned. The system selects an unused global LUNO if you do not enter a LUNO in response to the LUNO prompt. In either case, the following message is displayed at the station:

ASSIGNED LUNO = XX

where:

XX is the LUNO that was assigned.

8.6.1 AGL Command Format

AGL
ASSIGN GLOBAL LUNO
LUNO: [int]
ACCESS NAME: < acnm> @
PROGRAM FILE?: < yes/no>

8.6.2 AGL Command User Responses

System Prompts	Response Required or Optional	User Responses
LUNO:	O	Enter a LUNO in the range from 0 through > FF. See Appendix B for a list of system reserved global LUNO assignments. DO NOT use a system reserved LUNO or the system may crash or become nonfunctional. If you enter nothing, the system generates an unused global LUNO assignment by default.
ACCESS NAME:	R	Enter a valid device name of a physical device or the pathname of a file to be assigned to the LUNO.
PROGRAM FILE?:	R	Enter Y (yes) if the file is a program file. Enter N (no) for anything else. NO is the initial value.

System Prompts	Response Required or Optional	User Responses
DISPLAY	O	The DISPLAY prompt is a hidden prompt that you can use in expert mode from other commands or in batch mode. Enter Y (yes) to have AGL write the message ASSIGNED LUNO = XX to the TLF and then display the message on the screen or in the batch listing when the command is complete. Enter N (no) if you do not want the message displayed. *The default value is Y (yes).*

8.6.3 AGL Command Example

The following example shows how you can use the AGL command to assign a line printer (LP04) to a global LUNO (>30):

```
[ ] AGL
ASSIGN GLOBAL LUNO
  LUNO: >30
  ACCESS NAME: LP04
  PROGRAM FILE?: NO
ASSIGNED LUNO = >30
```

8.6.4 Commands Related to AGL

RGL (Release Global LUNO)

8.7 AL (ASSIGN LUNO)

The AL command assigns a device or file to a logical unit number (LUNO) accessible to I/O operations of your own station. If a device or file is currently assigned to the station-local LUNO, an error is returned. The system selects an unused station-local LUNO if you do not enter a LUNO in response to the LUNO prompt. In either case, the following message is displayed at the station:

ASSIGNED LUNO = >XX

where:

>XX is the LUNO that was assigned.

NOTE

Do not assign LUNOs reserved for SCI and languages. See Appendix B for a list of reserved system LUNOs.

8.7.1 AL Command Format

```
AL
ASSIGN LUNO
LUNO: [int]
ACCESS NAME: <acnm> @
PROGRAM FILE?: <yes/no>
```

8.7.2 AL Command User Responses

System Prompts	Response Required or Optional	User Responses
LUNO:	O	Enter a LUNO in the range 0 to >FF. A user-assigned station-local LUNO may mask a global LUNO that SCI expects to be assigned. This can cause SCI to perform in unpredictable ways. See Appendix B for a list of system reserved global LUNO assignments. Do not use a system reserved LUNO or the system may crash or become nonfunctional. *If you enter nothing, the system generates an unused station-local LUNO.*
ACCESS NAME:	R	Enter a valid device name of a physical device or the pathname of a file to be assigned to the LUNO.

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE:	R	Enter Y (yes) if the file is a program file. Enter N (no) for anything else. NO is the initial value.
DISPLAY	O	The DISPLAY prompt is a hidden prompt that you can use in expert mode from other commands or in batch mode. Enter Y (yes) to have AL write the message ASSIGNED LUNO = >XX to the TLF and then display the message on the screen or in the batch listing when the command is complete. Enter N (no) if you do not want the message displayed. *The default value is Y (yes).*

8.7.3 AL Command Example

The following example shows how you can enter the AL command to assign a line printer (LP04) to a station-local LUNO (>30):

```
[ ] AL
ASSIGN LUNO
  LUNO: >30
  ACCESS NAME: LP04
  PROGRAM FILE?: NO
ASSIGNED LUNO = >30
```

8.7.4 Commands Related to AL

RAL (Release All LUNOs)
RL (Release Station-Local LUNO)

ALGS

8.8 ALGS (ASSEMBLE AND LINK GENERATED SYSTEM)

After you define your system configuration using GEN990, which is the system generation utility, the assembly language source code must be assembled into machine language. The assembled object modules must then be linked with one another to form a single object module. You can accomplish the entire process by entering the ALGS command.

NOTE

Prior to executing the ALGS command, verify that a .S$IMAGES program file exists on the target disk. If this program file does not exist, you must create it using the Create Program File (CFPRO) command. You must copy the procedure DUMMY from the existing .S$IMAGES program file to the one you created on the target disk.

ALGS activates SCI in batch mode. The batch SCI executes a file of commands created by GEN990 and writes a listing to the BATCH LISTING access name. The batch stream executes the Macro Assembler, the Link Editor, and a synonym assignment program that creates a patch file for the new system. You can apply the patch file to the system by using the PGS command.

A successful ALGS completion creates the following files:

- A D$DATA object module output from the Macro Assembler in .S$SYSGEN.< system name> .D$OBJECT
- A task image that is the linked system, in file .S$IMAGES on the target disk. The name of the task image is the system name specified during system generation.
- A patch file ready to be applied to the system image in .S$SYSGEN.< system name> .PATCHFIL
- A link map of the linked system in .S$SYSGEN.< system name> .LINKMAP
- A file containing any error messages output by the patch generator in .S$SYSGEN.< system name> .ERROR
- An optional batch SCI listing

If ALGS is unsuccessful (that is, terminates with an error), consult the error discussion in Volume V.

8.8.1 ALGS Command Format

```
ALGS
ASSEMBLE AND LINK GENERATED SYSTEM
DATA DISK: [name]@
TARGET DISK: [name]@
SYSTEM NAME: < name> @
D$DATA LISTING: [acnm]@
BATCH LISTING: [acnm]@
```

8.8.2 ALGS Command User Responses

System Prompts	Response Required or Optional	User Responses
DATA DISK:	O	Enter the volume name or device name of the disk that contains the GEN990 output files. These files are located in a directory called .S$SYSGEN.<name>, where <name> is the system name parameter. The response to this prompt is the same disk that you specify for the Execute System Generation Utility (XGEN) command to use as the data disk. *The default value is DS01.*
TARGET DISK:	O	Enter the device name or the volume name of the disk to receive the generated system. The target disk must contain the program file called .S$IMAGES, with procedure DUMMY already installed. *The default value is DS01.*
SYSTEM NAME:	R	Enter the one- to five-character name given to the new system in response to the GEN990 OUTPUT prompt.
D$DATA LISTING:	O	Enter the access name of a sequential file or output device to which the Macro Assembler listing is to be written. *The default value is the following:* *<target disk>.S$SYSGEN.<system name>.D$LIST*
BATCH LISTING:	O	Enter the access name of a sequential file or device to which the batch SCI listing is to be written. *The default value is the following:* *<target disk>.S$SYSGEN.<system name>.ALGSLIST*

ALGS

8.8.3 ALGS Command Example

The following example shows how you can use the ALGS command to execute a batch stream to assemble, link, and install a system image. The ALGS command also creates a patch stream to be executed by the Patch Generated System (PGS) command:

```
[ ] ALGS
ASSEMBLE AND LINK GENERATED SYSTEM
  DATA DISK: SYSDSK
  TARGET DISK: SYSDSK
  SYSTEM NAME: MAR04
  D$DATA LISTING: .S$SYSGEN.MAR04.D$LIST
  BATCH LISTING: .S$SYSGEN.MAR04.BATCHLST
***ALGS - NORMAL TERMINATION *** :
```

8.8.4 Commands Related to ALGS

IGS	(Install Generated System)
PGS	(Patch Generated System)
TGS	(Test Generated System)

8.9 ANS (ANSWER INCOMING CALL)

The ANS command monitors incoming calls that are initiated from a particular terminal. When the call is received, a completion message appears on the screen of the terminal issuing the command.

NOTE

You do not need to execute ANS in order for a terminal to dial to a port on the 990 system. The device service routine (DSR) always connects with the terminal if enabled (see the utility MHPC). ANS allows monitoring a dial-in port and notification when a connection has occurred. This is primarily useful in batch streams where a function is to be performed following a dial-in connection.

8.9.1 ANS Command Format

ANS
ANSWER INCOMING CALL
TERMINAL ACCESS NAME: < acnm>
ANSWERBACK: [string]

8.9.2 ANS Command User Responses

System Prompts	Response Required or Optional	User Responses
TERMINAL ACCESS NAME:	R	Enter the name that identifies the remote terminal connection to be monitored.
ANSWERBACK:	O	Enter the answerback value, which is an identifying set of characters. This optional entry and the terminal answerback value are compared. If not equal, the connection is terminated. The answerback comparison determines connection with a specific terminal. Any characters in the set >00 through >1F and >7F are deleted from the received answerback before the comparison takes place. If you do not enter an answerback, the comparison is not made.

After a successful answerback, the synonym ABM is assigned the answerback value. If no answerback is received, the synonym is deleted. The DX10 completion synonym $$CC is set according to the completion status. (Refer to the CALL command description.)

ANS

NOTE

The hard-copy terminal port must be logically turned on (see the utility MHPC) and not presently in use.

8.9.3 ANS Command Example

The following example shows how you can use the ANS command to monitor incoming calls initiated from ST06:

```
[ ] ANS
ANSWER INCOMING CALL
  TERMINAL ACCESS NAME: ST06
  ANSWERBACK: BROADWAY BRANCH
```

8.9.4 Commands Related to ANS

CALL	(Call Terminal)
DISC	(Terminal Disconnection)
LHPC	(List Hard-Copy Terminal Port Characteristics)
MHPC	(Modify Hard-Copy Terminal Port Characteristics)

8.10 AS (ASSIGN SYNONYM)

A synonym is a string of one or more characters that substitutes for another string of characters. The string of characters for which the synonym substitutes is called the synonym value. You can assign a synonym by using the AS command. Then you can use the synonym instead of the original string as a response to a prompt. For example, you can legally assign the following synonyms under DX10:

Synonym	Synonym Value
PRINTER	LP01
PAYFILE	VOL1.DIR2.FILE13

The AS command assigns synonyms (abbreviations) to file pathnames. In response to the field prompts of the AS command, you should enter the abbreviation you want to assign and then enter the full pathname of the file. When you complete the pathname and press RETURN, the SCI menu appears.

8.10.1 AS Command Format

AS
ASSIGN SYNONYM VALUE
SYNONYM: < name>
VALUE: [string]

8.10.2 AS Command User Responses

System Prompts	Response Required or Optional	User Responses
SYNONYM:	R	Enter one or more alphabetic or numeric characters, beginning with an alphabetic character. This value is the synonym to be assigned.
VALUE:	O	Enter the character string to which you want to assign the synonym. If you do not enter a value for the synonym, the current value assigned to the synonym is erased and the synonym is deleted.

AS

8.10.3 AS Command Example

The following example shows how you can use the AS command to assign the synonym VJ to the pathname VOL1.PROGDIR.DIRB.FILE1:

```
[ ] AS
ASSIGN SYNONYM VALUE
  SYNONYM: VJ
  VALUE: VOL1.PROGDIR.DIRB.FILE1
```

8.10.4 Commands Related to AS

LS	(List Synonyms)
MS	(Modify Synonym)

8.11 ASB (ASSIGN SIMULATED BREAKPOINT)

The ASB command sets up a breakpoint on a range of values for memory as follows:

- Memory alteration (A)
- CRU access (C)
- PC value (P)
- Memory references (R)
- Status value (S)

Refer to Volume III for a detailed description.

AT

8.12 AT (ACTIVATE TASK)

The AT command activates an unconditionally suspended task. Refer to Volume III for a detailed description.

8.13 AUI (ASSIGN USER ID)

Authorized users can assign user IDs, passcodes, user privilege codes, and user descriptions to other users at any time by entering the AUI command.

8.13.1 AUI Command Format

AUI
ASSIGN USER ID
USER DESCRIPTION: < string>
NEW USER ID: < name>
NEW PASSCODE: [string]
USER PRIVILEGE CODE (0..7): < int>

8.13.2 AUI Command User Responses

System Prompts	Response Required or Optional	User Responses
USER DESCRIPTION:	R	Enter a string of 1 to 16 characters to identify the terminal user for informational purposes only.
NEW USER ID:	R	Enter a six-character user ID in the following format: aaannn where: aaa three alphabetic characters nnn any three numeric characters unique to this particular user. To conserve file space, this should be the smallest number not currently in use since nnn is the record number of a record in a relative record file (.S$TCALIB) that contains the synonyms of the user ID for the next log-on. Each record is 864 bytes in length.

AUI

System Prompts	Response Required or Optional	User Responses
NEW PASSCODE:	O	Enter a string of one to six alphabetic or numeric characters that comprise a passcode to be associated with the user ID assigned in response to the NEW USER ID.
USER PRIVILEGE CODE:	R	Enter a number, in the range of 0 through 7, that specifies the privilege level at which the user can operate DX10. 0 gives the least privileges and 7 gives access to the most privileged commands.

8.13.3 AUI Command Example

The following example shows how an authorized user can use the AUI command to assign a user description, a user ID, a passcode, and a user privilege code to a user.

```
[ ] AUI
ASSIGN USER ID
  USER DESCRIPTION: TECHPUBS
  NEW USER ID: TEC046
  NEW PASSCODE:
  USER PRIVILEGE CODE <0..7>: 0
```

8.13.4 Commands Related to AUI

DUI (Delete User ID)
LUI (List User IDs)
MUI (Modify User ID)

8.14 BATCH (BEGIN BATCH EXECUTION)

The BATCH command removes unnecessary SCI generated synonyms from your synonym list by executing the Q$SYN SCI procedure. The BATCH command should be the first command in any batch stream, and the EBATCH command should be the last command in any batch stream. You can use the Create Message (CM) command in batch mode to send status messages to the initiating terminal.

NOTE

The synonyms cleared by the Q$SYN procedure are not available to the batch stream. Therefore, if you have modified Q$SYN, you need to be sure the synonyms cleared do not affect the batch stream.

The BATCH command does nothing in foreground mode. The prompts are provided so you can have a header in the batch listing to show such items as user ID and station ID. The synonyms available to the batch stream are the ones the system usually sets.

8.14.1 BATCH Command Format

```
BATCH (BEGIN BATCH EXECUTION),
USER ID: [string]
STATION ID: [string]
BATCH INPUT ACCESS NAME: [string]
BATCH LISTING ACCESS NAME: [string]
LS (LIST SYNONYMS)?: [yes/no] (NO)
```

8.14.2 BATCH Command User Responses

System Prompts	Response Required or Optional	User Responses
USER ID:	O	Enter a six-character user ID.
STATION ID:	O	Enter a four-character station ID.
BATCH INPUT ACCESS NAME:	O	Enter the input access name from the Execute Batch (XB) command.
BATCH LISTING ACCESS NAME:	O	Enter the listing access name from the XB command.
LS (LIST SYNONYMS)?:	O	If you specify YES in response to this prompt, the List Synonyms (LS) command lists all synonyms on the batch listing file. If you specify NO, it does not list the synonyms. The default value is NO.

BATCH

8.14.3 BATCH Command Example

The following example shows a batch stream that performs a weekly backup of data and also verifies the backup. The BATCH command is the first command in the batch stream, and the EBATCH is the last.

```
BATCH   LS = YES
*
****** ASSIGN LUNO TO TAPE
*
AL      AN = @CT,
        PF = NO
EC
*
****** REWIND TAPE
*
RWL     LUNO = @$$LU
EC
*
****** UNBLOCKED BACKUP
*
BD      DIR = @WD,
        SEQ = @SEQ,
EC
*
****** VERIFY
*
RWL     L = @$$LU
VB      SEQ = @SEQ,
        DIR = @WD,
EC
*
EBATCH
```

8.15 BD (BACKUP DIRECTORY)

The BD command allows you to copy a set of files under a directory to a sequential file or to a magnetic tape device (does not include ASR 733 cassette tape, CSxx) or to an image file (can span disk volumes). You can restore the files on the backup by entering the Restore Directory (RD) command. You can verify a backup or a restoration of a directory by entering the Verify Backup (VB) command.

Backup to an image file is done when a disk device is specified as the sequential access name. This type of backup is best for backups of large disks to smaller disks.

NOTE

If you specify the FAST option when backing up files to a disk device as a sequential access name, you must restore the backup to a disk with the same sector size as the disk from which it was backed up.

When a control file is used to control the backup operation, a corresponding control file is required for VB and RD so that the backup is properly restored or verified.

The BD command does not copy the directory named .S$PRINT and the files named .S$ROLLA, .S$DIAG, and .S$CRASH unless you specify them with INCLUDE directives. You supply directives by using a control file as described in Section 7. You specify the pathname identifying the control file in response to a prompt of the BD command.

Control files can include the MOVE, EXCLUDE, INCLUDE, and OPTION directives. You can supply options either with an OPTION directive in a control file, or in response to the OPTION prompt of the BD command. Options that you can specify in either case include ALIASES, NOALIASES, BLOCK, NOBLOCK, REWIND, NOREWIND, UNLOAD, NOUNLOAD, DATE, NODATE, FAST, NOFAST, SYSFILE, and NOSYSFILE. These options are described in the description of the user responses.

The BD, RD, and VB utilities can process backups that span more than one magnetic tape or disk. The first record of each disk/tape volume contains a header record for identification purposes. Refer to the paragraph on directory utility commands in Section 7 for the format of the headers.

The BD command can copy a directory to several disk or tape volumes. When multivolume disk output is required, respond to the SEQUENTIAL ACCESS NAME prompt by entering a disk device name (DS02, for example). After you answer the remaining prompts, the system displays the following:

```
PATHNAME OF BACKUP FILE
PATHNAME: < acnm>  @
```

The access name consists of a volume name and the name of a file. The volume name that you specify must be the volume name given to the disk by the INV command. When the volume has been used previously for backup, a backup file has been created on the volume; you must use the name of that file. When the volume has been initialized to be used in the backup, enter any valid file name for the backup file. You must use the same volume name on all volumes used in the backup operation. The BD command creates all of the file names the same. If the files already exist, they must be the same on all volumes.

BD

When you enter a list of disk or tape drives for the output, the drives are used in the order in which you entered them. As the space on each listed drive is filled, the utility uses the next drive in the list. When the list is exhausted and there is more information to back up, the following message appears:

MOUNT VOLUME X; TYPE $ TO QUIT, Y TO CONTINUE

Then, you change disks or tapes, check to see if the device is ready, and enter Y to continue. To terminate the command, you should enter $. (You should not enter $ except to abort the backup process.) If there is no more information to back up, this message does not appear.

NOTE

When you back up a fixed system disk to a removable disk, be sure that no one tries to access the system disk while it is spun down.

A disk or tape written by the BD command cannot be successfully used in loading the system. The Restore Directory (RD) command must be used to restructure the system disk before loading the system from it. If you use BD to maintain a backup of your system files, you should retain a copy of the system disk from which the system can be loaded. Then, you can use RD to recover if your system disk is lost.

The BD utility handles the install and unload functions. You must not install (IV command) disk volumes used for backup. When you use multiple disks for backup, all disks must have the same volume name. You should initialize backup disks before executing the BD command. You load a disk volume used as the sequential access name in the BD, RD, or VB commands at the start of the operation and unload it when the utility fills the disk or the backup operation completes.

If you specify the BLOCK option, BD prompts you to enter the blocking factor in bytes. The output records are grouped in blocks of the specified size (9600 is the initial value). Each block is written as a physical record. This improves the tape utilization by reducing the number of interrecord gaps. You should not specify the BLOCK option for disk files since file management may not be able to handle blocks of the specified size. The default option is NOBLOCK.

The FAST and NOFAST options are for program files only. If you select the FAST option, the BD command performs faster than when you select the default option, NOFAST. With the FAST option specified, the BD command copies the program file as a whole file. With the NOFAST option specified, the BD command copies each task, procedure, and overlay image separately. The limitation on the FAST option is that the destination disk for the subsequent RD process must have the same sector size as the disk from which the directory was originally backed up. Therefore, you cannot use the FAST option if you want to restore to a disk with a different sector size. The FAST and NOFAST options only apply when DSxx is specified in response to the SEQUENTIAL ACCESS NAME prompt.

If you select the NOSYSFILE option, system files are not included in the backup operation. If you select the default option SYSFILE, system files are included in the backup operation. These files are .SCI990, .S$SYSLIB, .S$ERROR, .S$CNTRY, .S$SDS$, .S$OVLYA, .S$PROC, .S$IMAGES, .S$TCALIB, .S$FGTCA, .S$BGTCA, .S$SLG1, .S$SLG2, .S$NEWS, .S$LOADER, and .PATCH. The SYSFILE and NOSYSFILE options only apply when using the system disk.

The BD command can list or copy a summary of its results to a device or to a file that you specify.

If a hardware error is encountered on the source disk while backing up a file, a message informing you about this error condition is written to the listing file. Any files flagged as having been backed up with errors cannot be restored. Files not backed up due to disk errors are listed in a bad file summary on the listing.

8.15.1 BD Command Format

```
BD
BACKUP DIRECTORY
DIRECTORY PATHNAME: [acnm] @
SEQUENTIAL ACCESS NAME: [(acnm)] @
CONTROL ACCESS NAME: [acnm] @
LISTING ACCESS NAME: [acnm] @
OPTIONS: [(string)]
EXECUTION MODE (F,B): < F/B>
```

8.15.2 BD Command User Responses

System Prompts	Response Required or Optional	User Responses
DIRECTORY PATHNAME:	O	Enter a pathname identifying the directory that is the topmost directory of a set of files to be backed up. Optionally, the pathname can identify a single file when only one file is being backed up. *If you do not enter a directory pathname, you must specify a control access name.*

BD

System Prompts	Response Required or Optional	User Responses
SEQUENTIAL ACCESS NAMES:	O	Enter one or more unique tape device names, one or more unique disk device names, or a sequential file pathname, to which BD should back up the directory. *If you do not enter a sequential access name, you must specify a control access name.* When you respond with DSxx, this option displays the following prompt: PATHNAME OF BACKUP FILE PATHNAME: < acnm> @ The pathname consists of a volume name and a file name. The volume name that you specify must be the volume name given to the disk by the INV command. When the volume has been used previously for backup, a backup file has been created on the volume; the name of that file must be used. When the volume has been initialized to be used in the backup, enter any valid file name for the backup file. You must use the same volume name on all volumes used in the backup operation. The BD command creates all of the file names the same. If the files already exist, they must be the same on all volumes.
CONTROL ACCESS NAME:	O	Enter the device name or the file pathname from which DX10 reads directives to control the backup operation. (The directives that you can include in a control stream are described in the paragraph on control files in Section 7.) *If you do not enter a control name, you must specify both a directory pathname and sequential access name.*
LISTING ACCESS NAME:	O	Enter the device name or the file pathname to which DX10 should list a summary of the results of the backup operation. *The default value is the terminal local file (TLF).*

System Prompts	Response Required or Optional	User Responses
OPTIONS:	O	You can enter the following options to control the backup operation. The default options are ALIASES, NOBLOCK, NODATE, NOREWIND, NOUNLOAD, SYSFILE, and NOFAST.

Option	Purpose
ALIASES	Entering ALIASES specifies that all aliases associated with the files are also to be backed up.
NOALIASES	Entering NOALIASES specifies that no aliases are to be backed up.
BLOCK	Entering BLOCK specifies that records will be grouped in blocks of a certain size. The blocking factor in bytes is requested: BLOCKING FACTOR: [int] Each block is written as a physical record. The initial value is 9600 bytes. Files to be moved with the 3780 communications protocol must be blocked at 288 bytes or less.
NOBLOCK	Entering NOBLOCK specifies that each record will be written separately without blocking.
DATE	Entering DATE limits the backup to those files created or updated on or after the specified date. The date and time (year, month, day, hour, minute) of the oldest file are requested: YEAR: <int> MONTH: <int> DAY: <int> HOUR: [int] MINUTE: [int]

BD

System Prompts	Response Required or Optional	User Responses
	Option	Purpose
	NODATE	Entering NODATE indicates no date constraints on the backup.
	REWIND	Entering REWIND causes rewinding of tape volumes before the backup operation is performed.
	NOREWIND	Entering NOREWIND indicates tape volumes are not rewound before the backup operation is performed.
	UNLOAD	Entering UNLOAD indicates tape volumes are unloaded after use.
	NOUNLOAD	Entering NOUNLOAD indicates tape volumes are not unloaded after use.
	SYSFILE	Entering SYSFILE indicates that you want system files included in the backup operation. The SYSFILE and NOSYSFILE options only apply when using the system disk.
	NOSYSFILE	Entering NOSYSFILE indicates that you do not want system files included in the backup operation.
	FAST	Entering FAST indicates that the BD command copies the program file as a whole file. Consequently, when you specify the FAST option, the destination disk for the subsequent RD process must have the same sector size as the disk from which the directory was originally backed up. The FAST and NOFAST options only apply when DSxx is used as the sequential access name.

System Prompts	Response Required or Optional	User Responses
	Option	Purpose
	NOFAST	Entering NOFAST indicates that the BD command copies each task, procedure, and overlay image separately.
EXECUTION MODE (F,B):	R	Enter F (foreground) to execute the task in foreground mode. Enter B (background) to execute the task in background mode. *The initial value is FOREGROUND.*

NOTE

Although a control file, a directory pathname, and a sequential access name are all optional responses to prompting messages, you must specify a control file if either of the other responses is not given. You must provide both a directory pathname and a sequential access name if you do not specify a control file.

8.15.3 BD Command Examples

The following examples show how you can enter the BD command to copy the contents of the set of files under a directory onto a backup file or to tape. The first example shows a backup operation that encountered data errors. A message follows any file that had a data error. A list of the bad files appears at the end of the listing. Any file in the bad file summary is not backed up.

BD

```
   BACKUP DIRECTORY        12:30:01 FRIDAY, APRIL 30, 1982.

ORIGINAL SOURCE:        HWBAD
ORIGINAL DESTINATION:   PAE29.HW.SEQ
ORIGINAL OPTIONS:       ALIASES,NODATE,NOBLOCK,NOREWIND,NOUNLOAD,SYSFILE,NOFAST
CONTROL FILE:
LIST FILE:              PAE29.HW.BDBAD


 **       DIRECTORY  HWBAD
 **   BDTEXT
**** BDTEXT    BACKED UP WITH ERRORS - CANNOT BE RESTORED
 **   S$LOADER
**** S$LOADER  BACKED UP WITH ERRORS - CANNOT BE RESTORED
 **   LST
 **   CONFIG
 **   OBJ
 **   LKED
 **   PROG      - PROGRAM FILE

 **   TASK        ID
 **   LPTEST     >01
 **   BD         >02
 **   RD         >03
 **   VB         >04
 **   CD         >05
 **   VC         >06
****   PROGRAM FILE BACKED UP WITH ERRORS
****   PRECEDING TASK AND ALL SUCCEEDING TASKS ON THIS PROGRAM FILE NOT BACKED UP
 **   BDDATA
**** BDDATA    BACKED UP WITH ERRORS - CANNOT BE RESTORED
 **   BDPGM
 **   VCNORM
 **   RDPGM
 **   RELREC
 **   CD
**** CD        BACKED UP WITH ERRORS - CANNOT BE RESTORED
 **   BDFILE
 **   KEY
 **   KIF

 **       DIRECTORY  HWBAD.PROCS
 **   CD
 **   BD
 **   RD
 **   VB
 **       END OF DIRECTORY  HWBAD.PROCS

 **       END OF DIRECTORY  HWBAD
****    BAD FILE SUMMARY   ****
 **   HWBAD.BDTEXT
 **   HWBAD.S$LOADER
 **   HWBAD.PROG
 **   HWBAD.BDDATA
 **   HWBAD.CD

ELAPSED TIME = 3 MINUTES   18 SECONDS
SIZE OF INPUT = 1573 ADU'S
5 WARNINGS ISSUED

****** BACKUP DIRECTORY COMPLETED
```

The second example shows a backup operation to DSxx.

```
  BACKUP DIRECTORY          11:17:55 FRIDAY, APRIL 30, 1982.

ORIGINAL SOURCE:          PAE29.HW.DIR
ORIGINAL DESTINATION:     DS04
ORIGINAL OPTIONS:         ALIASES,NODATE,BLOCK,NOREWIND,NOUNLOAD,SYSFILE,NOFAST
CONTROL FILE:             ST01
LIST FILE:                PAE29.HW.BDDISK

EXCL SUBDIR
  **        DIRECTORY  PAE29.HW.DIR
  **  FILE2
  **  FILE1
  **        END OF DIRECTORY  PAE29.HW.DIR

END

ELAPSED TIME = 0 MINUTES    14 SECONDS
SIZE OF INPUT = 0 ADU'S

****** BACKUP DIRECTORY COMPLETED
```

The third example shows a backup operation that uses MOVE, INCLUDE, and EXCLUDE directives. The control file that contains these directives appears first.

```
MOV PAE29.HW.DIR
INCL FILE1
MOV PAE29.HW.DIR.SUBDIR
EX FILE3
END
```

```
  BACKUP DIRECTORY          11:12:55 FRIDAY, APRIL 30, 1982.

ORIGINAL SOURCE:          PAE29.HW.DIR
ORIGINAL DESTINATION:     PAE29.HW.SEQ
ORIGINAL OPTIONS:         ALIASES,NODATE,NOBLOCK,NOREWIND,NOUNLOAD,SYSFILE,NOFAST
CONTROL FILE:             ST01
LIST FILE:                PAE29.HW.BDOPT

MOV PAE29.HW.DIR
INCL FILE1
  **        DIRECTORY  PAE29.HW.DIR
  **  FILE1
  **        END OF DIRECTORY  PAE29.HW.DIR
```

BD

```
MOV PAE29.HW.DIR.SUBDIR
EX FILE3
  **        DIRECTORY  PAE29.HW.DIR.SUBDIR
  **  PFILE    - PROGRAM FILE

  **  TASK       ID
  **  FUTIL     >0B

  **  OVERLAY    ID
  **  FUTOV0    >60
  **  FUTOV1    >61
  **  FUTOV2    >62
  **  FUTOV3    >63
  **  FUTOV4    >64
  **  FUTOV5    >65
  **  FUTOV6    >66
  **  FUTOV7    >67
  **  FUTOV8    >68
  **  FUTOV9    >69
  **  FUTOVA    >6A
  **  FUTOVB    >6B
  **        END OF PROGRAM FILE

  **        END OF DIRECTORY  PAE29.HW.DIR.SUBDIR

  **  FILE3     INCLUDE/EXCLUDE FILE NOT FOUND
END

ELAPSED TIME = 0 MINUTES   42 SECONDS
SIZE OF INPUT = 53 ADU'S
1 WARNINGS ISSUED


****** BACKUP DIRECTORY COMPLETED
```

8.15.4 Commands Related to BD

RD	(Restore Directory)
VB	(Verify Backup)

8.16 BL (BACKSPACE LUNO)

The BL command moves the current data-access position backwards one or more records.

8.16.1 BL Command Format

```
BL
BACKSPACE LUNO
LUNO: < int>
RECORD COUNT: < int>
```

8.16.2 BL Command User Responses

System Prompts	Response Required or Optional	User Responses
LUNO:	R	Enter the LUNO assigned to the sequential device or file to be backspaced.
RECORD COUNT:	R	Enter the number of records (from 1 to 65535) that the sequential file or device is to be backspaced. If DX10 encounters an EOF before backspacing the number of records specified, it positions the sequential device or file so that the first record to be read from the device or file is the EOF.

8.16.3 BL Command Example

The following example shows how you can enter a BL command to backspace four records on a magnetic tape unit assigned to LUNO 32:

```
[ ] BL
BACKSPACE LUNO
  LUNO: 32
  RECORD COUNT: 4
```

8.16.4 Commands Related to BL

FL	(Forward Space LUNO)
RWL	(Rewind LUNO)

CALL

8.17 CALL (CALL TERMINAL)

The CALL command connects the 990 system to a remote terminal by using either automatic or manual dialing procedures. If a connection cannot be made, the utility tries again. The number of retries that the CALL command attempts is defined by one of the modifiable parameters in the command procedure. (Refer to the paragraph on remote terminal utilities in Section 7.) If an automatic call unit (ACU) is to be used, you must specify a telephone number in the CALL command. If you are going to dial the number, you should not specify a telephone number.

The CALL command is a command to the computer, not the terminal. Therefore, you can only issue the command at an SCI authorized terminal already connected to the computer. There are certain terminal types that have a dial or call command supported by the terminal that causes the terminal to call a computer.

8.17.1 CALL Command Format

```
CALL TERMINAL
TERMINAL ACCESS NAME: < acnm>
TELEPHONE NUMBER: [string]
ANSWERBACK: [string]
```

8.17.2 CALL Command User Responses

System Prompts	Response Required or Optional	User Responses
TERMINAL ACCESS NAME:	R	Enter the name that identifies the port to be used.
TELEPHONE NUMBER:	O	Enter the telephone number that identifies the terminal dataset to be called. An ACU must be associated with the terminal where you specify a telephone number. The following characters are accepted:

Character	Description
0–9	Defines number
=	Wait for dial tone
*	Touchtone telephone special character
#	Touchtone telephone special character
-	Improves readability only; does not affect dialing process

System Prompts	Response Required or Optional	User Responses
		You can use the wait for dial tone feature only if the ACU supports this function. If you omit the telephone number, manual dialing is assumed.
ANSWERBACK:	O	Enter the answerback value, which is an identifying set of characters that can be associated with a terminal. The terminal associated with the telephone number is identified by entering the answerback value for that terminal. This optional entry (answerback entered) and the terminal answerback value (answerback received) are compared. If the values are not the same, the connection is terminated. The answerback comparison verifies connection with a specific terminal. If the answerback value of the terminal contains lowercase ASCII characters, the comparison value must be entered using lowercase characters. Any characters in the set >00 through >1F and >7F are deleted from the received answerback before the comparison takes place. A comparison is not made if an answerback value is not entered.

After a successful answerback, the synonym ABM is assigned the answerback value. No answerback received causes the synonym to be deleted.

The following examples show how the DX10 completion code synonym ($$CC) is used in conjunction with the displayed message to identify the answerback/connect status.

- $$CC = 4101. CONNECTION UNSUCCESSFUL — NO ANSWERBACK. An answerback value was entered, but none was received. The connection is not made. After the maximum allowable retries have been performed, the message appears on the screen.

- $$CC = 4102. CONNECTION UNSUCCESSFUL — ANSWERBACK = <value received>. The answerback value received did not match the answerback entered. The connection is not made. After the maximum allowable retries have been performed, the message appears on the screen.

CALL

- $$CC = 0000. CONNECTION SUCCESSFUL. The answerback value received matched the answerback entered. The connection is made, and the completion message appears on the screen.

- $$CC = 0000. CONNECTION SUCCESSFUL — NO ANSWERBACK. No answerback value was entered, and no answerback value was received. The connection is made, and the completion message appears on the screen.

- $$CC = 0000. CONNECTION SUCCESSFUL — ANSWERBACK = <value received>. No answerback value was entered but an answerback value was received. The connection is made, and the completion message appears on the screen.

The hard-copy terminal port must be logically turned on (see the utility MHPC) and not presently in use. If automatic dialing is to be performed, the terminal must have an associated ACU.

8.17.3 CALL Command Example

The following example shows how you can use the CALL command to connect the 990 system to a remote terminal:

```
[ ] CALL
CALL TERMINAL
  TERMINAL ACCESS NAME: ST05
  TELEPHONE NUMBER: 2506352
  ANSWERBACK: GREETINGS
```

8.17.4 Commands Related to CALL

ANS	(Answer Incoming Call)
DISC	(Terminal Disconnection)
LHPC	(List Hard-Copy Terminal Port Characteristics)
MHPC	(Modify Hard-Copy Terminal Port Characteristics)

8.18 CC (COPY/CONCATENATE)

The CC command copies one or more files of information from physical devices or from disk storage into one file and makes them into a single file accessible under one file name. The target file cannot be one of the input files. The only limitation on the number of files that you can concatenate with one CC command is the space allowed for entry of pathnames or device names that identify the files or the source of the files to be concatenated. (This space is 50 characters in interactive mode or 235 characters in expert or batch mode.) You can save space by entering synonyms for pathnames or device names, thereby allowing concatenation of more files.

If the output file does not exist, the CC command creates it automatically according to the type of the first input file. When the input is a device, the output file is created as an expandable blank suppressed sequential file, with a logical record length of 80 bytes. If you do not want these parameters, you must precreate the output file with the desired parameters.

If the input is a file, the CC command creates the output file with exactly the same characteristics except that the current total number of records in the input file is specified as the primary allocation of the output file and the secondary allocation is defaulted. The write protect, delete protect, and forced write characteristics are not copied.

You can use also the CC command, like the Print File (PF) command, to copy one or more files to a line printer or to another output device. When an output device needs carriage control, the CC command adds single space control to the output unless the file being copied contains carriage control characters. Then it uses those existing characters.

NOTE

The CC command cannot copy directories, program files, or image files. Special commands exist to copy these files, primarily the Copy Directory (CD) command. The CC command can copy KIFs but there are two restrictions. The output file cannot already exist and be a KIF. If the output does not already exist, the input file cannot be a KIF.

8.18.1 CC Command Format

```
CC
COPY/CONCATENATE
INPUT ACCESS NAME(S): (< acnm>) @
OUTPUT ACCESS NAME: < acnm>  @
REPLACE?: < yes/no>
MAXIMUM RECORD LENGTH: [int]
```

CC

8.18.2 CC Command User Response

System Prompts	Response Required or Optional	User Responses
INPUT ACCESS NAMES(S):	R	Enter one or more pathnames or device names (separated by commas) that specify a file or files that you want to concatenate together into one output file. The CC command concatenates the files in the order that you enter input pathnames or device names. The number of files that can be concatenated is limited by the number of pathnames, device names, or synonyms that can fit in the space provided for a response. You can enter 50 characters in this space interactively. If you use expert or batch mode, you can enter up to 235 characters, including commas and periods. However, if you use synonyms in response to this prompt, the expanded length can be up to 48 bytes.
OUTPUT ACCESS NAME:	R	Enter the device name or the file pathname to which DX10 should write the concatenated files. If the pathname does not specify an existing file, the CC command automatically creates it. *The device name or pathname you specify must be different from any of the input access names.*
REPLACE?:	R	Enter Y (yes) if you want the file being copied to replace the current contents of the output file specified if one exists. Enter N (no) to abort command if the output file already exists. Default value is N.
MAXIMUM RECORD LENGTH:	O	Enter the maximum number of bytes to transfer per I/O operation. This value is the longest record of the input file that the CC command does not truncate. Records in the input file, up to this value, are always copied with the exact same length appearing in the output file. Longer records are truncated to this value. *The default value is 512 bytes.*

8.18.3 CC Command Example

The following example shows how you can enter the CC command to copy information from a cartridge tape (MT01) to a terminal (ST01), and how you can enter two SF commands to list the contents of two files (VOL1.TWO and VOL1.THREE). The example then shows how you can enter another CC command to concatenate the three files together into one file (.TEST). Another SF command lists the contents of the concatenated files.

```
[] CC

COPY/CONCATENATE
 INPUT ACCESS NAME(S):    MT01

 OUTPUT ACCESS NAME:    ST01

 REPLACE?: NO

 MAXIMUM RECORD LENGTH:    80

MT01   14:36:35 MONDAY, MAY 09, 1977.              PAGE 0001


 RECORD NUMBER ONE


 COPY COMPLETED

 [] SF

 SHOW FILE
  FILE PATHNAME: VOL1.TWO

 RECORD NUMBER TWO

 [] SF

 SHOW FILE
  FILE PATHNAME: VOL1.THREE

 RECORD NUMBER THREE

 [] CC

 COPY/CONCATENATE
  INPUT ACCESS NAME(S):    MT01,VOL1.TWO,VOL1.THREE

  OUTPUT ACCESS NAME:    .TEST

  REPLACE?: NO

  MAXIMUM RECORD LENGTH:    80

 COPY COMPLETED

 [] SF

 SHOW FILE
  FILE PATHNAME:    .TEST

 RECORD NUMBER ONE
 RECORD NUMBER TWO
 RECORD NUMBER THREE
```

CC

NOTE

MT01 was manually rewound after the first CC command.

8.18.4 Commands Related to CC

AF	(Append File)
CD	(Copy Directory)
PF	(Print File)

8.19 CD (COPY DIRECTORY)

The CD command allows you to copy a set of files under one directory to another directory. The copy operation does not change the contents of the source directory. The CD command automatically creates files and directories that do not exist in the destination directory.

The CD command does not copy the directory named .S$PRINT and files named .S$ROLLA, .S$DIAG, and .S$CRASH unless you specify them in INCLUDE directives. You use a control file to supply directives to the CD command as described in Section 7. Control files can include the following directives: MOVE, EXCLUDE, INCLUDE, and OPTION. You can specify options either with an OPTION directive in a control file, or in response to a prompt of the CD command. ADD, ALIASES, CMP, NOALIASES, NOCMP, NOSYSFILE, REPLACE, DATE, NODATE, SPRL, NOSPRL, RPRL, NORPRL and SYSFILE are the options that you can specify in either case. The CD command can list or copy a summary of its results to a user-specified device or file. This summary includes the names of the directories and files copied.

NOTE

Although a control file, an input pathname, and an output pathname are all optional responses to prompts, you must specify a control file if either of the other responses is not given. You must provide both an input and an output pathname if you do not specify a control file.

CD copies files using a method called fast copy. To do this, it performs direct disk I/O while reading from the source file and writing to the destination file. Each read and write moves a block of contiguous data up to a maximum of >9FA0 bytes.

If you are copying between disks that have different physical record lengths, you should use the SPRL and RPRL options to use the destination disk more efficiently.

When copying a program file, the CD command writes the following message to the listing file only if that program file has no holes in it (no deletions have been performed to the program file, only installations) and if the destination program file does not exist:

```
XXXX — PROGRAM FILE
PROGRAM FILE COPIED USING FAST COPY
```

where XXXX is the name of the program file being copied. If the above conditions do not exist, CD does not use the fast copy method while copying that program file.

The CD command usually writes the pathname of the program file being copied, plus the names of the tasks, procedures, and overlays in the listing file. However, in the case of fast copy, CD does not list the names of the tasks, procedures, and overlays of the program file.

CD

NOTE

Using the INCLUDE VCATALOG command in the control file for the CD command causes the CD operation to fail. You must use the volume name instead of VCATALOG. You can obtain a volume name by entering the Show Volume Status (SVS) command and specifying the appropriate drive.

8.19.1 CD Command Format

```
CD
COPY DIRECTORY
INPUT PATHNAME: [acnm] @
OUTPUT PATHNAME: [acnm] @
CONTROL ACCESS NAME: [acnm] @
LISTING ACCESS NAME: [acnm] @
OPTIONS: [(string)] @
EXECUTION MODE (F,B): < F/B>
```

8.19.2 CD Command User Responses

System Prompts	Response Required or Optional	User Responses
INPUT PATHNAME:	O	Enter a pathname identifying the directory that is the topmost directory of the set of files you want to copy. Optionally, the pathname can identify a single file when only one file is being copied. If the pathname does identify a single file, then that pathname must not be an alias. *If you do not enter an input pathname, you must specify a control access name.*
OUTPUT PATHNAME:	O	Enter a pathname identifying the directory to which DX10 copies the file or set of files identified by the input pathname. If the output directory specified does not exist, the CD command creates the directory equal in size to the lowest-level directory being copied. *If you do not enter an output pathname, you must specify a control access name.*

System Prompts	Response Required or Optional	User Responses

CAUTION

You can copy an entire directory, or you can modify which files and/or subdirectories to copy by using a control file. The output pathname must always specify only a directory in either case. If you only want to copy a single file, you can specify the entire pathname for that file for the input. The utility converts this internally to a directory pathname with one INCLUDE statement. The rules for INCLUDE statements discussed in the paragraph on control files in Section 7 apply.

For example, to copy file .C.A to file .B.A, specify .C.A as the input pathname and .B as the output pathname. Do not use .B.A as the output pathname. This implies the directory .B.A. The resulting file pathname would be .B.A.A. To copy the entire directory .C to the directory .B, specify .C as the input pathname and .B as the output pathname.

CONTROL ACCESS NAME:	O	Enter the device name or the file pathname from which DX10 reads directives to control the copy operation. The directives that you can include in a control stream are described in the paragraph on control files in Section 7. *If you do not enter a control access name, you must specify both an input pathname and an output pathname.*
LISTING ACCESS NAME:	O	Enter the device name or the file pathname to which DX10 should list a summary of the results of the copy operation. *The default value is the terminal local file (TLF).*
OPTIONS:	O	You can enter one or more of the following options separated by commas to specify whether files and aliases being copied are to replace files and aliases of the same name on the destination directory. The physical record length of sequential or relative record files can be modified in the copy. The default options are ADD, ALIASES, NOSPRL, NORPRL, NODATE, CMP, and SYSFILE.

CD

System Prompts	Response Required or Optional	User Responses
	Option	Purpose
	ADD	Entering ADD specifies that files or aliases are copied only if a file or alias with the same name does not exist in the output set of files.
	REPLACE	Entering REPLACE specifies that files are copied even if a file with the same name exists in the output set of files unless the files are write or delete protected. The existing file or files are deleted before the source file or files are copied. Aliases are not replaced.
	ALIASES	Entering ALIASES specifies that all aliases associated with the files are also to be copied.
	NOALIASES	Entering NOALIASES specifies that no aliases are to be copied.
	DATE	Entering DATE limits the copying to those files created or updated on or after the specified date. The date and time (year, month, day, hour, minute) of the oldest file are requested: YEAR: <int> MONTH: <int> DAY: <int> HOUR: [int] MINUTE: [int]
	NODATE	Entering NODATE indicates no date constraints on the copying.

System Prompts	Response Required or Optional	User Responses
	Option	Purpose
	SPRL	Entering SPRL indicates that sequential files are to be converted to a new physical record size. DX10 displays the following prompt: PHYSICAL RECORD LENGTH: < int> where: int is a decimal or hexadecimal integer that specifies the physical record length to be used when creating the output file. If the physical record length of some system files are changed, the system will not operate. Do not copy system files with the option specified.
	NOSPRL	Entering NOSPRL indicates that sequential files are not to be converted to a new physical record size.
	RPRL	Entering RPRL indicates that relative record files are to be converted to a new physical record size. DX10 displays the following prompt: PHYSICAL RECORD LENGTH: < int> where: int is a decimal or hexadecimal integer that specifies the physical record length to be used when creating the output file. If the physical record length of some system files are changed, the system will not operate. Do not copy system files with this option specified.
	NORPRL	Entering NORPRL indicates that relative record files are not to be converted to a new physical record size.

CD

System Prompts	Response Required or Optional	User Responses
	Option	Purpose
	CMP	Entering CMP indicates that unused physical blocks at the end of KIFs are truncated and released back to the disk. Both CMP and NOCMP collect all of the secondary allocations together.
	NOCMP	Entering NOCMP indicates that unused physical blocks at the end of KIFs are not truncated. When copying a KIF, you may want to retain the original space allocation since compression resets the primary allocation space to the current size of the file, thereby increasing access time to a KIF.
	SYSFILE	Entering SYSFILE indicates that system files are included in the copy of the directory. These files are .SCI990, .S$SYSLIB, .S$ERROR, .S$CNTRY, .S$SDS$, .S$OVLYA, .S$PROC, .S$IMAGES, .S$TCALIB, .S$FGTCA, .S$BGTCA, .S$SLG1, .S$SLG2, .S$NEWS, .S$LOADER, and .PATCH. The SYSFILE and NOSYSFILE options only apply when using the system disk.
	NOSYSFILE	Entering NOSYSFILE indicates that system files are not included in the copy of the directory.
EXECUTION MODE (F,B)	R	Enter F (foreground) to execute the task in foreground mode. Enter B (background) to execute the task in background mode. *The initial value is FOREGROUND.*

8.19.3 CD Command Example

The following example shows how you can enter the CD command to copy a set of files under one directory (PAE29.HW.DIR) to another directory (CART.DATADIR).

```
  COPY DIRECTORY          11:21:27 FRIDAY, APRIL 30, 1982.

ORIGINAL SOURCE:        PAE29.HW.DIR
ORIGINAL DESTINATION:   CART.DATADIR
ORIGINAL OPTIONS:       ADD, ALIASES, NODATE, SYSFILE, CMP
CONTROL FILE:           ST01
LIST FILE:              PAE29.HW.CDLIST

EX FILE1

  **       DIRECTORY  CART.DATADIR.SUBDIR
  **  PFILE    - PROGRAM FILE
  **  PROGRAM FILE COPIED USING FAST COPY
  **       END OF DIRECTORY  CART.DATADIR.SUBDIR

  **  FILE2
END

ELAPSED TIME = 0 MINUTES   22 SECONDS
SIZE OF INPUT = 53 ADU'S

****** COPY DIRECTORY COMPLETED
```

8.19.4 Commands Related to CD

BD	(Backup Directory)
RD	(Restore Directory)
VC	(Verify Copy)

CF

8.20 CF (CREATE FILE)

The CF command can create any of the DX10 file types available.

8.20.1 CF Command Format

```
CF
CREATE FILE - SEQ, REL, KEY, DIR, PRO, IMG
FILE TYPE: <{SEQ,REL,KEY,DIR,PRO,IMG}>
```

8.20.2 CF Command User Responses

System Prompts	Response Required or Optional	User Responses
FILE TYPE:	R	Enter one of the following abbreviations:

Abbreviation	File Type
DIR	Directory
IMG	Image
KEY	Key indexed
PRO	Program
REL	Relative record
SEQ	Sequential

Subsequent system prompts and user responses depend on the type of file specified. Entering the CF command specifying a file type has the same effect as entering the specialized Create File command for the type of file specified.

8.20.3 CF Command Example

The following example shows how you can create a relative record file using the CF command. Note the similarities between the prompts that appear for this CF command, and those that appear when you use the CFREL command to create the file.

```
[ ] CF
CREATE FILE - SEQ, REL, KEY, DIR, PROG, IMG
  FILE TYPE: REL
CREATE RELATIVE RECORD FILE
  PATHNAME: VOL.1.FILE2
  LOGICAL RECORD LENGTH: > 80
  PHYSICAL RECORD LENGTH: > 230
  INITIAL ALLOCATION: 500
  SECONDARY ALLOCATION: 1000
  EXPANDABLE?: YES
  FORCED WRITE?: YES
```

8.20.4 Commands Related to CF

CFDIR (Create Directory File)
CFIMG (Create Image File)
CFKEY (Create Key Indexed File)
CFPRO (Create Program File)
CFREL (Create Relative Record File)
CFSEQ (Create Sequential File)

CFDIR

8.21 CFDIR (CREATE DIRECTORY FILE)

The CFDIR command creates special files, called directories, that contain pointers to other files and to subdirectories cataloged under the directory. You must create each directory and subdirectory in a hierarchy by using a separate CFDIR command. For example, a file may be identified by the pathname VOL1.DIR.SUBDIR.FILE1. The directory and the subdirectory in that hierarchy were created by separately entered CFDIR commands that created the following directories: VOL1.DIR and VOL1.DIR.SUBDIR.

You create files under directories or subdirectories by using separate file creation commands.

8.21.1 CFDIR Command Format

CFDIR
CREATE DIRECTORY FILE
PATHNAME: < acnm> @
MAX ENTRIES: < int>
DEFAULT PHYSICAL RECORD SIZE: [int]

8.21.2 CFDIR Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname by which you want to identify the directory. All directories in the hierarchy under which the directory is cataloged must be included in the pathname. Pathnames cannot exceed 48 characters.
MAX ENTRIES:	R	Enter an integer specifying the maximum number of file names to be kept in this directory. The number entered is rounded up to the nearest prime number. You should note that the List Directory (LD) command is limited to listing directories of 3,000 entries or less, and the Map Disk (MD) command is limited to mapping directories of 3,000 entries or less.

System Prompts	Response Required or Optional	User Responses
DEFAULT PHYSICAL RECORD SIZE:	O	Enter an even integer that specifies the physical record size to be used in all subsequent creation operations under this directory. If not specified, the value of the default physical record size in which this directory is cataloged is used. If none is specified there, the value specified for the disk during system generation is used.

8.21.3 CFDIR Command Example

If directories with pathnames VOL1.SOURCE and VOL1.SOURCE.MYFILE exist and VOL1 is the system disk, you can create a directory with pathname VOL1.SOURCE.MYFILE.TEST by using the CFDIR command as follows:

```
[ ] CFDIR
CREATE DIRECTORY FILE
  PATHNAME: .SOURCE.MYFILE.TEST
  MAX ENTRIES: 100
  DEFAULT PHYSICAL RECORD SIZE:
```

CFDIR creates a directory with the indicated pathname that allows 101 (next prime number) entries.

NOTE

If you do not specify a volume in the pathname, DX10 assumes the system disk.

8.21.4 Commands Related to CFDIR

CF	(Create File)
CFIMG	(Create Image File)
CFKEY	(Create Key Indexed File)
CFPRO	(Create Program File)
CFREL	(Create Relative Record File)
CFSEQ	(Create Sequential File)

CFIMG

8.22 CFIMG (CREATE IMAGE FILE)

The CFIMG command creates an image file. DX10 supports image files only for roll-in/roll-out, system loader and crash files, and other system files. No general use is suited for this file type. The use of image files is not recommended because record divisions for image files vary as the file is transferred from one type of disk to another type, and data may not be recovered using ordinary techniques.

8.22.1 CFIMG Command Format

```
CFIMG
CREATE IMAGE FILE
PATHNAME: < acnm> @
INITIAL ALLOCATION: < int>
SECONDARY ALLOCATION: [int]
EXPANDABLE?: < yes/no>
```

8.22.2 CFIMG Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname by which you want to identify the image file.
INITIAL ALLOCATION:	R	Enter an even integer that specifies the size of the file in logical records when the file is created. The logical record size is the sector size of the disk. An estimate of the number of records needed is as follows: R = ((length of object + sector size of disk)/(sector size of disk)) where all values are in bytes. The initial value is 100 records.
SECONDARY ALLOCATION:	O	Enter the amount of disk space, expressed in logical records, by which the system expands the file if the file grows beyond its initial allocation. The default value is one ADU or the number of ADUs required to hold one physical record, whichever is greater. Refer to Volume III for information on how DX10 allocates file space.
EXPANDABLE?:	R	Enter Y (yes) if the file may grow beyond its initial size. Enter N (no) if it may not.

8.22.3 CFIMG Command Example

The following example shows how you can enter a CFIMG command to create an image file of 1000 sectors in length.

```
[ ] CFIMG
CREATE IMAGE FILE
  PATHNAME: VOL1.FILE3
  INITIAL ALLOCATION: 1000
  SECONDARY ALLOCATION:
  EXPANDABLE?: NO
```

8.22.4 Commands Related to CFIMG

CF	(Create File)
CFDIR	(Create Directory File)
CFKEY	(Create Key Indexed File)
CFPRO	(Create Program File)
CFREL	(Create Relative Record File)
CFSEQ	(Create Sequential File)

CFKEY

8.23 CFKEY (CREATE KEY INDEXED FILE)

The CFKEY command creates a key indexed file (KIF). The correct specification and sizing of a KIF relates closely to the designated use of the file. For further information, refer to Volume III.

8.23.1 CFKEY Command Format

```
CFKEY
CREATE KEY INDEXED FILE
PATHNAME: < acnm> @
LOGICAL RECORD LENGTH: < int>
PHYSICAL RECORD LENGTH: [int]
INITIAL ALLOCATION: [int]
SECONDARY ALLOCATION: [int]
MAXIMUM SIZE: < int>
```

After you answer these prompts, the following prompts appear:

```
START POSITION: < int>
KEY LENGTH: < int>
DUPLICATES?: < yes/no>
MODIFIABLE?: < yes/no>
ANY MORE KEYS?: < yes/no>
```

} You can repeat these entries for multiple key files.

8.23.2 CFKEY Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname of the KIF to be created.
LOGICAL RECORD LENGTH:	R	Enter the length, in bytes, of the records to be entered into the file.

System Prompts	Response Required or Optional	User Responses
PHYSICAL RECORD LENGTH:	O	Enter the size of a block of data transferred to and from the disk. A physical record usually contains several logical records. The physical record length must be at least 22 bytes larger than the logical record length. You specify the default value when you create the directory that contains the file (default physical record size parameter). If you do not specify a default value when you create the directory, the default value specified during system generation is used (default record size parameter). The block size should be an even multiple, or factor, of the ADU size of the disk on which the file is being created. Refer to Volume III for more details on physical records.
INITIAL ALLOCATION:	O	Enter the amount of disk space, in logical records, that should be reserved for the file when it is created. The default value is system defined and is the suggested value to use.

NOTE

The value entered for the INITIAL ALLOCATION prompt is compared against the results of a calculation involving the logical record length, physical record length, and maximum size. (Refer to Volume III for information on initial allocation for KIFs.) The larger of the two is used as the initial allocation. If the default is taken for this prompt, the results of the calculation are used. If the initial allocation is too small, the file may become fragmented if it is on a disk containing many other active files.

SECONDARY ALLOCATION:	O	Enter the amount of disk space, expressed in logical records, by which the system expands the file if the file grows beyond its initial allocation. The default value is one ADU or the number of ADUs required to hold one physical record, whichever is greater. Refer to Volume III for information on how DX10 allocates file space.

CFKEY

System Prompts	Response Required or Optional	User Responses
MAXIMUM SIZE:	R	Enter the estimated adjusted number of logical records to be placed in the file. That is, the number of logical records times the percent of nonblanks in each logical record. For example, if there are to be 10,000 logical records in the file and about 50% of each record is made up of blanks, the adjusted number of logical records would be 5000.
START POSITION:	R	Enter the column number of the first column of the key. Columns of a record begin with the number 1.
KEY LENGTH:	R	Enter the number of columns in the key. The range of legal values is from 1 through 100.
DUPLICATES?:	R	The value of a key in a record may be unique throughout the file, or there may be several records containing the same key value. These same values are called duplicates. If the key need not be unique, the response to this prompt should be YES. If the key value must be unique, such as a key for a social security number, the response to this prompt should be NO. This feature can aid in preventing erroneous data from being entered into the file and in guaranteeing unique key values for every record in the file. You cannot specify duplicate values for the primary key.
MODIFIABLE?:	R	You can change the value of a key if you define it to be modifiable. All keys, except the primary key (the first key defined), may be modifiable. Keys such as an employee's salary and marital status are examples of modifiable keys. Entering YES makes the key modifiable, and entering NO makes the key nonmodifiable.
ANY MORE KEYS?:	R	Enter YES if there are more keys to be defined; otherwise, enter NO. After you enter NO, the KIF is created.

8.23.3 CFKEY Command Example

The following example shows how to interactively create a KIF with three keys.

```
[ ] CFKEY
CREATE KEY INDEXED FILE
  PATHNAME: VOL1.FILE1
  LOGICAL RECORD LENGTH: 80
  PHYSICAL RECORD LENGTH: 560
  INITIAL ALLOCATION: 1200
  SECONDARY ALLOCATION: 500
  MAXIMUM SIZE: 1000
KEY DESCRIPTION FOR KEY NUMBER 1
  START POSITION: 1
  KEY LENGTH: 2
  DUPLICATES?: N
  MODIFIABLE?: N
  ANY MORE KEYS?: YES
KEY DESCRIPTION FOR KEY NUMBER 2
  START POSITION: 1
  KEY LENGTH: 4
  DUPLICATES?: N
  MODIFIABLE?: N YES
  ANY MORE KEYS?: YES
KEY DESCRIPTION FOR KEY NUMBER 3
  START POSITION: 3
  KEY LENGTH: 8
  DUPLICATES?: N
  MODIFIABLE?: Y NO
  ANY MORE KEYS?: YES NO
```

8.23.4 CFKEY in Batch Mode

To create a KIF in batch mode, the batch stream must include a CFKEY command followed by one or as many as 14 KEY commands and an ENDKEY command.

Example:

Command		Comment
CFKEY	parameters	Create KIF
KEY	parameters	Specify from 1 to 14 keys
.	.	
.	.	
.	.	
KEY	parameters	
ENDKEY		Terminate batch mode CFKEY.

CFKEY

The batch parameters for CFKEY include PATHNAME, LOGICAL RECORD LENGTH, PHYSICAL RECORD LENGTH, INITIAL ALLOCATION, SECONDARY ALLOCATION, and MAXIMUM SIZE. The batch parameters for each KEY command are START POSITION, KEY LENGTH, DUPLICATES?, and MODIFIABLE?. The ENDKEY command does not have any parameters.

You cannot vary the indicated order of these commands without producing errors. The inputs for parameters must correspond to the definitions given earlier for CFKEY parameters in interactive mode.

8.23.5 Commands Related to CFKEY

CF	(Create File)
CFDIR	(Create Directory File)
CFIMG	(Create Image File)
CFPRO	(Create Program File)
CFREL	(Create Relative Record File)
CFSEQ	(Create Sequential File)
CKS	(Copy KIF to Sequential File)
CSK	(Copy Sequential File to KIF)
MKF	(Map Key Indexed File)
MKL	(Modify KIF Logging)

8.24 CFPRO (CREATE PROGRAM FILE)

The CFPRO command creates a special type of relative record file used for storing DX10 program images. Program files are usually created expandable.

8.24.1 CFPRO Command Format

```
CFPRO
CREATE PROGRAM FILE
PATHNAME: < acnm> @
MAX NUMBER OF TASKS: < int>
MAX NUMBER OF PROCEDURES: < int>
MAX NUMBER OF OVERLAYS: < int>
INITIAL ALLOCATION: < int>
SECONDARY ALLOCATION: [int]
EXPANDABLE?: < yes/no>
```

8.24.2 CFPRO Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname by which you want to identify the program file.
MAX NUMBER OF TASKS:	R	Enter the maximum number of tasks allowed in this program file. The initial value is 255.
MAX NUMBER OF PROCEDURES:	R	Enter the maximum number of procedures allowed in this program file. The initial value is 255.
MAX NUMBER OF OVERLAYS:	R	Enter the maximum number of overlays allowed in this program file. The initial value is 255.
INITIAL ALLOCATION:	R	Enter an integer that specifies the initial file size in logical records. The displayed initial value of 85 is the minimum size for overhead information for files created to contain the maximum number of tasks, procedures, and overlays.
SECONDARY ALLOCATION:	O	Enter an integer that specifies the amount of disk space (in logical records) to be allocated when the file must grow. The system provides a default value if you do not specify a size.

CFPRO

System Prompts	Response Required or Optional	User Responses
EXPANDABLE?:	R	Enter Y (yes) if the file may grow. Enter N (no) if it may not grow. The initial value is YES.

8.24.3 CFPRO Example

The following example shows how you can enter a CFPRO command to create a program file:

```
[ ] CFPRO
CREATE PROGRAM FILE
  PATHNAME: VOL1.DIR2.FILE2
  MAX NUMBER OF TASKS: 255
  MAX NUMBER OF PROCEDURES: 255
  MAX NUMBER OF OVERLAYS: 255
  INITIAL ALLOCATION: 1000
  SECONDARY ALLOCATION: 500
  EXPANDABLE?: YES
```

8.24.4 Commands Related to CFPRO

CF	(Create File)
CFDIR	(Create Directory File)
CFIMG	(Create Image File)
CFKEY	(Create Key Indexed File)
CFREL	(Create Relative Record File)
CFSEQ	(Create Sequential File)

8.25 CFREL (CREATE RELATIVE RECORD FILE)

The CFREL command creates a relative record file.

8.25.1 CFREL Command Format

```
CFREL
CREATE RELATIVE RECORD FILE
PATHNAME: < acnm> @
LOGICAL RECORD LENGTH: [int]
PHYSICAL RECORD LENGTH: [int]
INITIAL ALLOCATION: [int]
SECONDARY ALLOCATION: [int]
EXPANDABLE?: < yes/no>
FORCED WRITE?: < yes/no>
```

8.25.2 CFREL Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname by which you want to identify the relative record file.
LOGICAL RECORD LENGTH:	O	Enter the length in bytes of a record for this file. The default value is 80.
PHYSICAL RECORD LENGTH:	O	Enter the disk transfer unit size. You specify the default value when you create the directory that contains the file (default physical record size parameter). If you do not specify a default value when you create the directory, the default value specified during system generation is used (default record size parameter). Refer to Volume III for more information on physical records.
INITIAL ALLOCATION:	O	Enter the initial size, in logical records, of this file. The system provides a default value if you do not specify a size.
SECONDARY ALLOCATION:	O	If the file grows beyond the initial size, this number, in logical records, specifies the growth segment size. The system provides a default value if you do not specify a size.
EXPANDABLE?:	R	Enter Y (yes) if the file can grow beyond its initial size. Enter N (no) if it cannot.

CFREL

System Prompts	Response Required or Optional	User Responses
FORCED WRITE?:	R	Disk writes to relative record files are ordinarily deferred until memory is needed. Enter Y (yes) to specify that physical records be written immediately if any logical records contained in them are modified. Enter N (no) to specify that DX10 is to do deferred writes. For more information on forced write, refer to Volume III.

8.25.3 CFREL Command Example

The following example shows how you can enter a CFREL command to create a relative record file.

```
[ ] CFREL
CREATE RELATIVE RECORD FILE
 PATHNAME: VOL1.FILE2
 LOGICAL RECORD LENGTH: 80
 PHYSICAL RECORD LENGTH: > 230
 INITIAL ALLOCATION: 500
 SECONDARY ALLOCATION: 1000
 EXPANDABLE?: YES
 FORCED WRITE?: YES
```

8.25.4 Commands Related to CFREL

CF	(Create File)
CFDIR	(Create Directory File)
CFIMG	(Create Image File)
CFKEY	(Create Key Indexed File)
CFPRO	(Create Program File)
CFSEQ	(Create Sequential File)

8.26 CFSEQ (CREATE SEQUENTIAL FILE)

The CFSEQ command creates a sequential file.

8.26.1 CFSEQ Command Format

```
CFSEQ
CREATE SEQUENTIAL FILE
PATHNAME: < acnm> @
LOGICAL RECORD LENGTH: [int]
PHYSICAL RECORD LENGTH: [int]
INITIAL ALLOCATION: [int]
SECONDARY ALLOCATION: [int]
EXPANDABLE?: < yes/no>
BLANK SUPPRESS?: < yes/no>
FORCED WRITE?: < yes/no>
```

8.26.2 CFSEQ Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname by which you want to identify the sequential file.
LOGICAL RECORD LENGTH:	O	Enter an even integer that specifies the average record size in bytes for this file. If records are to be blank suppressed, do not include blank strings in the size estimate. The default value is 80.
PHYSICAL RECORD LENGTH:	O	Enter an even integer that specifies the disk transfer size, in bytes. You specify the default value when you create the directory that contains the file (default physical record size parameter). If you do not specify a default value when you create the directory, the default value specified during system generation is used (default record size parameter). Refer to Volume III for more information on physical records.
INITIAL ALLOCATION:	O	Enter an integer that specifies the initial file size in logical records. The system provides a default value if you do not specify a size.

CFSEQ

System Prompts	Response Required or Optional	User Responses
SECONDARY ALLOCATION:	O	Enter an integer that specifies the increment of additional disk space in logical records, to be allocated when the existing space is full. The system provides a default value if you do not specify a size.
EXPANDABLE?:	R	Enter Y (yes) if the file may grow beyond its initial size. Enter N (no) if it may not.
BLANK SUPPRESS?:	R	Enter Y (yes) if the file is to be stored with strings of blanks encoded in compact form (blank suppressed). Enter N (no) if blank strings are not to be encoded. If a file is blank suppressed, rewrite operations are almost impossible and should not be used. (After the encoding operation, the rewritten record must exactly match the previous record in length.)
FORCED WRITE?:	R	Disk writes to sequential files are normally deferred until memory is needed. Enter Y (yes) to specify that DX10 is to immediately write physical records to disk if any logical records contained in them are modified. Enter N (no) to specify that DX10 is to do deferred disk writes. For more information on forced write, refer to Volume III.

8.26.3 CFSEQ Command Example

The following example shows how you can enter a CFSEQ command to create a sequential file.

```
[ ] CFSEQ
CREATE SEQUENTIAL FILE
  PATHNAME: VOL1.FILE1
  LOGICAL RECORD LENGTH:
  PHYSICAL RECORD LENGTH:
  INITIAL ALLOCATION: 100
  SECONDARY ALLOCATION:
  EXPANDABLE?: YES
  BLANK SUPPRESS?: NO
  FORCED WRITE?: NO
```

8.26.4 Commands Related to CFSEQ

CF	(Create File)
CFDIR	(Create Directory File)
CFIMG	(Create Image File)
CFKEY	(Create Key Indexed File)
CFPRO	(Create Program File)
CFREL	(Create Relative Record File)

CKD

8.27 CKD (CHECK DISK FOR CONSISTENCY)

The CKD command checks the disk structure consistency. This command displays the following: all directories on the specified disk, with an analysis of the entry counts for each directory; the number of ADUs dually allocated; and a dump of the bit maps for the disk.

8.27.1 CKD Command Format

```
CKD
CHECK DISK FOR CONSISTENCY
DISK UNIT: < acnm> @
LISTING ACCESS NAME: [acnm]@
```

8.27.2 CKD Command User Responses

System Prompts	Response Required or Optional	User Responses
DISK UNIT:	R	Enter the device name of the disk unit that contains the disk you want to check (such as DS01 or DS02).
LISTING ACCESS NAME:	O	Enter the pathname to which output is to be listed. The specified pathname must not be a file on the disk being checked.
AUTO CORRECTION	O	Call your dealer or customer representative before using this prompt because it is very dangerous. A YES response is used only when the disk is known to have a few USED BUT NOT ALLOCATED or ALLOCATED BUT NOT USED ADUs. It only makes the bit maps match the file structure and corrects BAD DIRECTORY messages. It *does not* correct dual allocation and *does not* correct disk errors.

8.27.3 CKD Command Operation

Follow these steps to execute the CKD command:

1. Mount the volume to be checked, leaving it write protected. If the volume is the system disk, it is preferable not to perform another initial program load (IPL). A system disk being analyzed need not be write protected but the listing access name should be ME or a printer to keep from allocating disk space, which causes invalid indications of inconsistencies to appear. Write protection is a caution against disk allocation or deallocation on the disk during checking.

2. Install the volume using the Install Volume (IV) command.

3. Ensure that no disk allocation activity is occurring. This means that no compiler, assembler, or application activity can run that places files on the disk being checked. If the disk is the system disk, this precludes logging on or off and using any utility that produces output to a file (including the terminal local file), such as the MD, LD, or STS commands. However, you can use these commands if their output goes to a file not on the disk being checked or to a hard-copy device. Violating these cautions causes errors to appear.

4. Execute the CKD utility. Reply to the DISK UNIT prompt with the device name of the disk unit where the disk you want to check is mounted. The CKD command does not execute if you enter the volume name. For the LISTING ACCESS NAME: prompt, specify a file where the output is to be listed. Do not specify a file on the disk being checked, since this causes errors to appear.

8.27.4 Analyzing CKD Command Output

CKD creates the first section of its output during the first pass made on the disk. This section contains a list of all directories on the disk, with an analysis of the entry counts printed in five columns of numbers to the left of the directory pathname.

The columns are defined as follows:

Column	Definition
SIZE	The total number of entries the directory can hold.
NFL	The number of files currently in the directory as recorded in the directory overhead record. This value should match the value in the FILES column.
NAR	The number of available records as recorded in the directory overhead record. This value should match the value in the AVAIL column.
FILES	The actual number of files currently in the directory determined by count. If this does not match the NFL column, the message *****BAD DIRECTORY is printed above the line describing the bad directory.
AVAIL	The actual number of available entries in the directory determined by count. If this does not match the NAR column, the bad directory message is printed as for FILES.

NOTE

All numbers printed by this utility are hexadecimal.

CKD

The contents of these five fields do not directly affect any dual allocation problems that might exist. Errors indicate system crashes or power failures during critical sequences that update these counts on disk. However, bad directories can exist on a disk that is being used without causing problems.

CKD produces the remaining sections of its output during the second pass on the disk. The first of these is a dump of all file descriptor records (FDRs) that claim storage that is either not allocated in the bit maps, dually allocated, or both. The full pathnames of the files are printed, along with an index unique to the file on that disk. The index is used later for reference.

The next part of the output for the second pass is a single line that specifies how many ADUs are dually allocated. If the number specified is zero, this concludes the information printed in the output listing file.

If the number of dually allocated ADUs is greater than zero, the output file contains a list of all ranges of dually allocated storage. Under each range is a list in the format FFFF(AAAA-BBBB). FFFF is the file index and AAAA-BBBB is the subrange of ADUs claimed by that particular file. Since a range of dually allocated storage can be larger than any particular file's range, all files listed under a given range are not necessarily dually allocated over each other. Only those files whose individual ranges do overlap are dually allocated.

Finally, CKD produces a dump of the bit maps for the disk. This dump is used to determine what problems are occurring with the bit maps. Interspersed in the dump may be lines of output stating the ranges of ADUs that are either used and not allocated or allocated but not used. Allocated but not used produces no problems; however, used but not allocated means the disk structures are damaged and you need to correct them. You should use the Copy Directory (CD) command if possible; otherwise, contact your software supplier.

8.27.5 CKD Command Examples

The following examples show the output that the CKD command produces. In the first example, the disk being checked had no multiply allocated ADUs; in the second example, the disk being checked did have multiply allocated ADUs.

Example 1:

```
[] CKD
CHECK DISK FOR CONSISTENCY
 DISK UNIT: DS02
 LISTING ACCESS NAME: LP$1
PASS # 1
SIZE  NFL   NAR   FILES AVAIL DIRECTORY
000B  0002  0009  0002  0009  SS9.DX.SYSTEM.TABLES
000B  0001  000A  0001  000A  SS9.DX.SYSTEM
000B  0001  000A  0001  000A  SS9.DX.BATCH.ASM
000B  0003  0008  0003  0008  SS9.DX.BATCH
000B  0001  000A  0001  000A  SS9.DX.DX10.SOURCE
000B  0001  000A  0001  000A  SS9.DX.DX10
000B  0004  0007  0004  0007  SS9.DX.SYSTSK.SOURCE
000B  0001  000A  0001  000A  SS9.DX.SYSTSK
000B  0004  0007  0004  0007  SS9.DX
000B  0002  0009  0002  0009  SS9
0000 ADUS MULTIPLY ALLOCATED
```

Example 2:

```
[] CKD
CHECK DISK FOR CONSISTENCY
 DISK UNIT: DS02
 LISTING ACCESS NAME: LP$1
PASS # 1
SIZE  NFL   NAR   FILES AVAIL DIRECTORY
000B  0002  0009  0002  0009  SS9.DX.SYSTEM.TABLES
000B  0001  000A  0001  000A  SS9.DX.SYSTEM
000B  0001  000A  0001  000A  SS9.DX.BATCH.ASM
000B  0003  0008  0003  0008  SS9.DX.BATCH
000B  0008  0003  0007  0004  SS9.DX.DX10.SOURCE
000B  0001  000A  0001  000A  SS9.DX.DX10
000B  0004  0007  0004  0007  SS9.DX.SYSTSK.SOURCE
000B  0001  000A  0001  000A  SS9.DX.SYSTSK
000B  0006  0005  0006  0005  SS9.DX
000B  0002  0009  0002  0009  SS9
PASS # 2
 DUAL ALLOCATION FILE 000B SS9.DX.GEN1
0000 0002 4745 4E31 2020 2020 0000 0000
0C20 0050 0050 0005 0105 0001 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 07BB 01C2 CBA1 07BB 01C2
CBA1 0101 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 000B 0007 0004 0000 534F
 DUAL ALLOCATION FILE 0013 SS9.DX.DX10.SOURCE.GETCHR
0001 0008 4745 5443 4852 2020 0000 0000
1A00 0360 0050 0003 0212 0003 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 07BB 01C2 CB43 07BB 01C2
CB42 0300 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 000B 0007 0004 0000 534F
 DUAL ALLOCATION FILE 0018 SS9.DX.SYSTSK.SOURCE.SYSLG
0001 0003 5359 534C 4720 2020 0000 0000
1A00 0360 0050 0033 0101 0003 0000 0000
0000 0147 0000 000E 01DA 0000 0000 0000
0000 0000 0000 07BB 01B6 E25B 07BB 01B6
E255 0300 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 000B 0007 0004 0000 534F
 DUAL ALLOCATION FILE 0019 SS9.DX.SYSTSK.SOURCE.GETLOG
0001 0007 4745 544C 4F47 2020 0000 0000
1A00 0360 0050 0075 0212 0003 0000 0000
0000 0303 0000 0023 0000 0000 0000 0000
0000 0000 0000 07BB 01B6 E360 07BB 01B6
E353 0300 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 000B 0007 0004 0000 534F
0010 ADUS MULTIPLY ALLOCATED
ADU 0214 DUALLY ALLOCATED
0019  0013
```

CKD

```
ADU 0213 DUALLY ALLOCATED
0019  0013
ADU 0212 DUALLY ALLOCATED
0019  0013
ADU 0109 DUALLY ALLOCATED
0018  000B
ADU 0108 DUALLY ALLOCATED
0018  000B
ADU 0107 DUALLY ALLOCATED
0018  000B
ADU 0106 DUALLY ALLOCATED
0018  000B
ADU 0105 DUALLY ALLOCATED
0018  000B
ADU 0215  USED BUT NOT ALLOCATED IN BIT MAPS
ADU 0216  USED BUT NOT ALLOCATED IN BIT MAPS
ADU 0217  USED BUT NOT ALLOCATED IN BIT MAPS
ADU 0218  USED BUT NOT ALLOCATED IN BIT MAPS
ADU 0381  ALLOCATED BUT NOT USED
ADU 0382  ALLOCATED BUT NOT USED
ADU 0383  ALLOCATED BUT NOT USED
ADU 0384  ALLOCATED BUT NOT USED
ADU 0385  ALLOCATED BUT NOT USED
ADU 0386  ALLOCATED BUT NOT USED
ADU 0387  ALLOCATED BUT NOT USED
ADU 0388  ALLOCATED BUT NOT USED
ADU 0389  ALLOCATED BUT NOT USED
```

8.27.6 Commands Related to CKD

RCD	(Recover Disk)
SD	(Scan Disk)

8.28 CKR (COPY KEY INDEXED FILE RANDOMLY)

The CKR command copies the contents of a key indexed file (KIF) to an output file or device. Files are not copied in a specific order. (That is, the records are not ordered according to any key.) You can use this command for the following two purposes:

- To completely recover a KIF after a system failure while in PARTIAL logging mode.
- To recover as much data as possible from a KIF that has been partially destroyed or is unusable due to disk I/O errors.

CKR reads the physical records of the KIF and checks the validity of the records in the data block. The logical records extracted from the file are written to an output file. The output file can be a sequential file or a previously created KIF. If a data block has an inconsistent structure, an informative message is written to the listing file. This message indicates the physical record number of the bad block and the offset into the block of the error.

8.28.1 CKR Command Format

```
CKR
COPY KEY INDEXED FILE RANDOMLY
KEY INDEXED FILE PATHNAME: < acnm> @
OUTPUT FILE PATHNAME: < acnm> @
LISTING ACCESS NAME: [acnm]@
```

8.28.2 CKR Command User Responses

System Prompts	Response Required or Optional	User Responses
KEY INDEXED FILE PATHNAME:	R	Enter a pathname that identifies the KIF you want to copy.
OUTPUT FILE PATHNAME:	R	Enter the pathname that identifies an existing KIF or sequential file to which the logical records are written. The file is created as sequential if it does not already exist.
LISTING ACCESS NAME:	O	Enter the pathname of the file or device to which a description of the bad areas of the KIF is written. Enter a null response to specify the terminal local file (TLF).

CKR

8.28.3 CKR Command Example

The following example shows how you can execute the CKR command for a KIF containing five physical records that could cause errors or system crashes if accessed by CKS or a user program.

```
[] CKR
COPY KEY INDEXED FILE RANDOMLY
 KEY INDEXED FILE PATHNAME: VOL1.KIF
 OUTPUT FILE PATHNAME: VOL1.SEQ
 LISTING ACCESS NAME: VOL1.LIST

1809        LOGICAL RECORDS COPIED

[] SF
SHOW FILE
 FILE PATHNAME: VOL1.LIST
RECORD  OFFSET        MESSAGE
 00062   00D2   LOGICAL RECORD SIZE TOO LARGE
 00063   00F0   KEY ID TOO LARGE
 00064   0000   INCORRECT RECORD NUMBER
 00080   00F2   UNSUPPRESSION ERROR
 0022B   00EE   LOGICAL RECORD SIZE IS ODD

1809        LOGICAL RECORDS COPIED
```

8.28.4 CKR Listing File Messages

Table 8-1 contains the listing file messages for the CKR command. These error messages indicate that a block has not been processed or has only been partially processed. Some records might have been inaccessible and therefore are lost. You can use the Show Relative to File (SRF) command to view the data in the file. Then, you can copy the data out of the bad file and verify if the data is good.

Table 8-1. CKR Listing File Messages

Message	Meaning
INCORRECT RECORD NUMBER	The first two words of the physical record are not the same as the physical record number.
LOGICAL RECORD SIZE IS ODD	The blank-suppressed logical record size must be an even number.
LOGICAL RECORD SIZE TOO LARGE	The logical record size is considered too large if it is more than 50 percent larger than the defined logical record size.
KEY ID TOO LARGE	The key ID cannot be larger than the maximum key ID of the physical record.
UNSUPPRESSION ERROR	While the record was being unsuppressed, it became too large.
00XX READ ERROR	The specified error code (XX) was returned when the physical record was read. Refer to the section on SVC error codes in Volume VI for an explanation of SVC > 00 error XX.
XXXXXX LOGICAL RECORDS COPIED	The specified number (XXXXXX) of logical records was recovered from the KIF. The number is a decimal number.
00XX ERROR WRITING TO OUTPUT FILE	The specified error code (XX) was returned while writing to the output file. Refer to the section on SVC error codes in Volume VI for an explanation of SVC > 00 error XX.
00XX KEY POINTS TO NONEXISTENT DATA RECORD	The leaf node key (indicated by XX offset) points to a nonexistent data block. Refer to the section on SVC error codes in Volume VI for an explanation of SVC > 00 error XX.

8.28.5 Commands Related to CKR

CKS (Copy KIF to Sequential File)
CSK (Copy Sequential File to KIF)

CKS

8.29 CKS (COPY KIF TO SEQUENTIAL FILE)

The CKS command copies the contents of a key indexed file (KIF) to a sequential file or device in an order that you specify. You specify the order by naming a key by which the file is read while being placed onto the sequential media. For example, if you specify the KEY entry as the number 2 and that key contains zip codes, the KIF is read sequentially by zip code as it is copied to the sequential media. If the device is STXX or LPXX, only the first 80 characters of the record are output to the device.

NOTE

The CKS command is not intended to be used to back up KIFs. To back up a KIF, use the Copy Directory (CD) or the Backup Directory (BD) command, specifying the pathname of the file you want to back up as the input pathname parameter.

8.29.1 CKS Command Format

```
CKS
COPY KIF TO SEQUENTIAL FILE
INPUT PATHNAME: < acnm> @
KEY: < integer>
BEGINNING KEY VALUE: [string]
ENDING KEY VALUE: [string]
ASCII OR BINARY (A/B): [A/B]
OUTPUT ACCESS NAME: < acnm> @
REPLACE?: < yes/no>
```

8.29.2 CKS Command User Responses

System Prompts	Response Required or Optional	User Responses
INPUT PATHNAME:	R	Enter a pathname that identifies the KIF you want to copy.
KEY:	R	Enter an integer number in the range of 1 through 14 that specifies the key number of the key by which the KIF is to be read as it is being copied to the sequential file.

System Prompts	Response Required or Optional	User Responses
BEGINNING KEY VALUE:	O	Enter a value less than or equal to the value of a key to begin copying logical records from the KIF. The number of characters you enter need not be the same as the size of the specified key. If you specify fewer characters, the rest of the key value is filled with binary zeros. If you enter no response, the copy begins with the lowest key value.
ENDING KEY VALUE:	O	Enter a value greater than the value of a key to stop copying logical records from the KIF. The number of characters you enter need not be the same as the size of the specified key. If you specify fewer characters, the rest of the key value is filled with binary zeros. If you do not enter a response, the copy stops when all logical records have been copied.
ASCII OR BINARY(A/B):	R	Enter A (ASCII) to specify that the given key values are processed as ASCII characters; enter B (binary) to specify that the given key values are processed as binary values coded in hexadecimal. Binary values can contain only the characters 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. If the desired key contains both ASCII and binary characters, you must enter a B and specify any ASCII characters as binary. For example, if the first byte of a key is the binary number >2C and the second byte is the letter C, you must specify the key value as 2C43. The initial value is A.
OUTPUT ACCESS NAME:	R	Enter the device name or the file pathname to which CKS should write the copy of the KIF. CKS creates a sequential file if the file you name does not already exist. If the device is STXX or LPXX, only the first 80 characters of the record are written to the device.

CKS

System Prompts	Response Required or Optional	User Responses
REPLACE?:	R	Enter Y (yes) to specify that the file being copied is to replace the current contents of the output file if it already exists. Enter N (no) to specify that the command is to be aborted if the output file already exists.
DISPLAY	O	DISPLAY is a nonprompted keyword in the CKS command. If a batch stream or procedure calls CKS with N specified for the DISPLAY prompt, the message indicating the number of logical records copied by the CKS command is not displayed. *The default value is YES.*

8.29.3 CKS Command Example

The following example shows how you can use the CKS command to copy a KIF into a sequential file. Because the CKS command shown orders the information being copied by the second key, the information now on the sequential file is in a different order than it is in the example for the CSK command.

```
[] CKS
COPY KIF TO SEQUENTIAL FILE
 INPUT PATHNAME: VOL1.KIF
 KEY: 2
 BEGINNING KEY VALUE:
 ENDING KEY VALUE:
 ASCII OR BINARY:
 OUTPUT ACCESS NAME: VOL1.KIFKEY2
 REPLACE?: NO
3 RECORDS OUTPUT TO FILE
[]

[] SF
SHOW FILE
 FILE PATHNAME: VOL1.KIFKEY2
05243   0027   15   REGULAR WIDGET
05314   0092   04   MICRO WIDGET
05289   0153   07   SUPER WIDGET
[]
```

8.29.4 Commands Related to CKS

CSK (Copy Sequential File to KIF)

8.30 CKSR (COPY KIF TO SEQUENTIAL FILE RANDOMLY)

See the CKR command.

CL

8.31 CL (COPY LINES)

After you activate the Text Editor, the CL command specifies which lines are to be copied from the current file to another position. Refer to Volume IV for a detailed description.

8.32 CM (CREATE MESSAGE)

You can send a one-line message to any terminal by entering the CM command. A receiving terminal can only receive a message created by the CM command if the following are true:

- A user is logged on at the terminal (if log-on is required).
- The terminal is not involved in an operation that prevents receipt of the message.
- The receiving user presses the CMD or RETURN key.

If no user is logged on at the terminal, or if the current state of the terminal prevents receipt of the message, DX10 places the message in a queue in the system table area until it can be transmitted. You can use the CM command in batch mode to send status messages to the terminal initiating the batch stream.

8.32.1 CM Command Format

```
CM
CREATE MESSAGE
RECEIVING TERMINAL NAME: < name> @
MSG: [string]
```

8.32.2 CM Command User Responses

System Prompts	Response Required or Optional	User Responses
RECEIVING TERMINAL NAME:	R	Enter the device name of the terminal to which the system should send the message.
MSG:	O	Enter a one-line message from 1 to 70 characters long.

8.32.3 CM Command Example

The following example shows how you can use the CM command to send a message to terminal ST02.

```
[ ] CM
CREATE MESSAGE
  RECEIVING TERMINAL NAME: ST02
  MSG: SYSTEM WILL BE DOWN 0900 to 1100 6/14/78
```

CM

You can send a message to all terminals by creating a special command (FLASH) that issues a CM command for each terminal generated in the system. Be careful when sending messages to all terminals since system table area is required to hold the message until it can be sent to the receiving terminal. The following example shows how you can use the CM command to send the previous message to three terminals. You can include extra terminals by adding more CM commands in the FLASH command.

```
FLASH (SYSTEM NEWS FLASH)
CM RECEIVING = ST02,
     MSG = "SYSTEM WILL BE DOWN 0900 to 1100 6/14/78"
CM RECEIVING = ST10,
     MSG = "SYSTEM WILL BE DOWN 0900 to 1100 6/14/78"
CM RECEIVING = ST04,
     MSG = "SYSTEM WILL BE DOWN 0900 to 1100 6/14/78"
```

8.32.4 Commands Related to CM

MSG (Display Message)

8.33 CPI (COPY PROGRAM IMAGE)

The CPI command allows you to copy the image of a task, procedure, or overlay from one program file to another. You can also copy the image to a different location within the same program file. If a task image is being copied, you can also copy all procedures and/or overlays associated with the task unless the image is being copied to the same program file. You can define the name and/or ID of the image in the output program file to be different from those in the input program file. When the image is being copied to a different location within the same program file, you must specify a unique name and ID. The names and IDs of the procedures and overlays being copied concurrently with their associated task must be the same. If you want to change the names and/or IDs, you must copy the procedures and overlays separately. If you change overlay IDs, you must change any code referencing those IDs. The following list describes the cases in which the code references IDs:

- System images (created in S$IMAGES by the system generation process)
- Any program linked with the LOAD directive (automatic overlay loading)
- Any program linked in IMAGE format that has REF statements referencing the phase name of an overlay

If you copy overlays of a task at the same time as the task, the Map Program File (MPF) of the output will show the overlays linked in reverse order to what an MPF shows of the input. However, this should cause no problems.

8.33.1 CPI Command Format

```
CPI
COPY PROGRAM IMAGE
INPUT PROGRAM FILE: < acnm> @
OUTPUT PROGRAM FILE: [acnm]@
INPUT IMAGE NAME OF ID: < string>
OUTPUT IMAGE NAME AND/OR ID: [string]
IMAGE TYPE (T/P/O): < {T/P/O}>
OPTIONS: < ADD/REPLACE>
```

If you enter T for IMAGE TYPE and specify an output program file pathname, the following two prompts appear:

```
COPY PROGRAM IMAGE TASK EXTENSION
COPY ASSOCIATED OVERLAYS?: < Y/N>
COPY ASSOCIATED PROCEDURES?: < Y/N>
```

If you enter O for IMAGE TYPE, the following prompt appears:

```
COPY PROGRAM IMAGE OVERLAY EXTENSION
ASSOCIATED TASK NAME OR ID: < string>
```

CPI

8.33.2 CPI Command User Responses

System Prompts	Response Required or Optional	User Responses
INPUT PROGRAM FILE:	R	Enter a pathname that identifies the program file containing the image you want to copy.
OUTPUT PROGRAM FILE:	O	Enter a pathname that identifies the program file into which the image is to be copied. If the image is to be copied to the same program file, you must leave this prompt blank.
INPUT IMAGE NAME OR ID:	R	Enter the name or the ID of the image to be copied.
OUTPUT IMAGE NAME AND/OR ID:	O	Enter the name and/or ID under which the image is to be installed in the output program file. If left blank, the input name and ID will be used. If you specify only one, the other will be the same as on the input. If you specify both, you must separate them by a comma. If you leave the OUTPUT PROGRAM FILE prompt blank, you must specify both a name and an ID unique to the input program file.
IMAGE TYPE (T/P/O):	R	Enter T if the image is a task; enter P if it is a procedure; enter O if it is an overlay.
OPTIONS:	R	Enter REPLACE if the image already exists in the output and it should be replaced; otherwise enter ADD.
COPY ASSOCIATED OVERLAYS?:	R	Enter Y if all overlays, linked to the task being copied, should also be copied; otherwise, enter N.
COPY ASSOCIATED PROCEDURES?:	R	Enter Y if all procedures, linked to the task being copied, should also be copied; otherwise, enter N.
ASSOCIATED TASK NAME OR ID:	R	Enter the name or ID of the task in the output program file to which the overlay should be linked.

8.33.3 CPI Command Examples

The first CPI example copies a task and all its associated overlays and procedures from one program file to another and installs the task, by ID, on the output program file. The second example copies a specified overlay from one program file to another, installs the overlay by ID and name on the output program file, and links the overlay to a specified task. The third example copies a specified procedure from one program file to another and installs the procedure on the output program file using the ID specified in the INPUT IMAGE NAME OR ID prompt.

```
[ ] CPI
COPY PROGRAM IMAGE
  INPUT PROGRAM FILE: .S$PROGA
  OUTPUT PROGRAM FILE: .EXAMPLE
  INPUT IMAGE NAME OR ID: >33
  OUTPUT IMAGE NAME AND/OR ID: >01
  IMAGE TYPE (T/P/O): T
  OPTIONS: ADD
COPY PROGRAM IMAGE TASK EXTENSION
  COPY ASSOCIATED OVERLAYS?: Y
  COPY ASSOCIATED PROCEDURES?: Y

[ ] CPI
COPY PROGRAM IMAGE
  INPUT PROGRAM FILE: .S$PROGA
  OUTPUT PROGRAM FILE: .EXAMPLE
  INPUT IMAGE NAME OR ID: >3
  OUTPUT IMAGE NAME AND/OR ID: >7,NEW
  IMAGE TYPE (T/P/O): O
  OPTIONS: ADD
COPY PROGRAM IMAGE OVERLAY EXTENSION
  ASSOCIATED TASK NAME OR ID: >1

[ ] CPI
COPY PROGRAM IMAGE
  INPUT PROGRAM FILE: .S$PROGA
  OUTPUT PROGRAM FILE: .EXAMPLE
  INPUT IMAGE NAME OR ID: >2
  OUTPUT IMAGE NAME AND/OR ID:
  IMAGE TYPE (T/P/O): P
  OPTIONS: ADD
```

8.33.4 Commands Related to CPI

BD	(Backup Directory)
CD	(Copy Directory)
RD	(Restore Directory)

8.34 CRV (CHECK AND RESET VOLUME)

The Check and Reset Volume (CRV) command resets a media change condition on a disk drive. The system registers a media change condition whenever it detects that media has been changed on a disk drive (or that the disk drive has been cycled through the not ready to ready state) without an UV/IV command sequence.

NOTE

The system does not detect a media change condition on DS10, DS25, DS31, DS50, DS200 disks.

When a media change condition occurs, a read operation is allowed but a write operation results in an >EB error. The CRV command can reset this media change condition; any other command causes an >EB error.

To reset a media change condition, the CRV command compares the volume information of the disk inserted in a specified drive with the volume information of the disk that was in the drive during the last IV command. If the volume information matches, the CRV command resets the MEDIA CHANGE bit of the disk status word, thus resetting the media change condition. If the disk has been changed and the volume information does not match, the CRV command performs a UV command on the installed disk. The UV command, in turn, resets the MEDIA CHANGE bit. In this case, you must still perform an IV command on the new disk.

When you execute the CRV command in expert mode, a hidden prompt, UNCONDITIONAL RESET, is available. This prompt forces a reset of the media change status whether or not the inserted disk is the previously installed disk or another disk. This feature is designed for use after a power failure on a volume whose volume information was changed by an MVI command before the power failure. The UNCONDITIONAL RESET prompt tells DX10 that although the volume information has changed, the inserted disk is the same disk that received the IV command. If the inserted disk is not the previously installed disk, the use of this feature might corrupt the data on the inserted disk.

CAUTION

DX10 detects a media change to help you avoid media change conditions that could destroy data. If you use the UNCONDITIONAL RESET prompt improperly, you could destroy data.

8.34.1 CRV Command Format

```
CRV
CHECK AND RESET VOLUME
UNIT NAME: < name> @
```

8.34.2 CRV Command User Responses

System Prompt	Response Required or Optional	User Response
UNIT NAME:	R	Enter the name of the disk drive containing the volume you want to check; enter this data in the following form: DSxx.

8.34.3 CRV Command Example

The following example shows how you can use the CRV command to check the volume installed in disk drive DS04 for a media change condition.

```
[ ] CRV
CHECK AND RESET VOLUME
  UNIT NAME: DS04
RESET DISK SUCCESSFUL
```

The message RESET DISK SUCCESSFUL indicates that the CRV command checks the disk and finds no error. If the CRV command finds an error, it issues the following message:

```
RESET DISK UNSUCCESSFUL VOLUME UNLOADED
```

8.34.4 Commands Related to CRV

IV	(Install Volume)
UV	(Unload Volume)

CSF

8.35 CSF (CREATE SYSTEM FILES)

The CSF command deletes and/or creates a system crash file, roll file, and print directory for use by LP$x. CSF first attempts to delete the crash file and then the roll file; then it creates new files. CSF calls Delete File (DF) twice (once for each file to be deleted), Create File Image (CFIMG) twice (once for each file created), and Create File Directory (CFDIR) once to create the print directory. When creating the crash and roll files, CSF may return a 0027 SVC error. The error results from an attempt to delete files that do not exist before creating new ones. In this case, the new files are created correctly and the error can be ignored. If you issue the CSF command on a running system, you must perform an initial program load (IPL) or loss of data from the system disk may occur.

8.35.1 CSF Command Format

```
CSF
CREATE SYSTEM FILES
VOLUME NAME: < name> @
MEMORY SIZE IN BYTES: < int>
DISK SECTOR SIZE IN BYTES: < int>
ROLL FILE INITIAL ALLOCATION: < int>
```

8.35.2 CSF Command User Responses

System Prompts	Response Required or Optional	User Responses
VOLUME NAME:	R	Enter the one- to eight-character alphanumeric string entered in the disk build procedure (refer to Volume V).
MEMORY SIZE IN BYTES:	R	Enter the memory size in bytes, as an integer number (that is, 128K and 131072 indicate the same memory size).
DISK SECTOR SIZE IN BYTES:	R	Enter an integer value that is the number of bytes per sector. *The default value is 256.*
ROLL FILE INITIAL ALLOCATION:	R	Enter the initial size in ADUs of the roll file. *The initial value is 240.*

8.35.3 CSF Command Example

The following example shows how you can use the CSF command to create the system roll and crash files:

```
[ ] CSF
CREATE SYSTEM FILES
  VOLUME NAME: REL30
  MEMORY SIZE IN BYTES: 128K
  DISK SECTOR SIZE IN BYTES: 256
  ROLL FILE INITIAL ALLOCATION: 240
```

8.35.4 Commands Related to CSF

CF (Create File)

CSK

8.36 CSK (COPY SEQUENTIAL FILE TO KIF)

The CSK command copies the contents of a sequential file to a previously defined key indexed file (KIF). You can execute the CSK command any number of times and build the KIF a step at a time. For faster loading of the KIF, you should execute the Modify KIF Logging (MKL) command prior to executing the CSK command or the initial loading of records into the KIF.

8.36.1 CSK Command Format

```
CSK
COPY SEQUENTIAL FILE TO KIF
INPUT ACCESS NAME: <acnm> @
OUTPUT PATHNAME: <acnm> @
```

8.36.2 CSK Command User Responses

System Prompts	Response Required or Optional	User Responses
INPUT ACCESS NAME:	R	Enter the device name or the file pathname of a sequential file from which CSK is to copy sequential information to the KIF specified.
OUTPUT PATHNAME:	R	Enter a pathname identifying the KIF to which the sequential information is to be copied. *You must create the KIF before entering the CSK command.*

8.36.3 CSK Command Example

The following example shows the creation of a KIF, the display of the contents of a sequential file, and the copying of the sequential file into the KIF by means of a CSK command.

```
[] CFKEY
CREATE KEY INDEXED FILE
 PATHNAME:    VOL1.KIF
 LOGICAL RECORD LENGTH:    80
 PHYSICAL RECORD LENGTH:    >240
 INITIAL ALLOCATION:    100
 SECONDARY ALLOCATION:    200
 MAXIMUM SIZE:    125
KEY DESCRIPTION FOR KEY NUMBER 1
 START POSITION:  1
 KEY LENGTH:    5
 DUPLICATES?: NO
 MODIFIABLE?: NO
 ANY MORE KEYS?:  YES
KEY DESCRIPTION FOR KEY NUMBER 2
 START POSITION:  1    8
 KEY LENGTH:    4
 DUPLICATES?: NO    Y
 MODIFIABLE?: NO
 ANY MORE KEYS?:  YES
KEY DESCRIPTION FOR KEY NUMBER 3
 START POSITION:  8    14
 KEY LENGTH:    2
 DUPLICATES?: NO  Y
 MODIFIABLE?: NO  Y
 ANY MORE KEYS?: YES  NO

[] SF
SHOW FILE
 FILE PATHNAME: VOL1.KIFSRC
05314  0092  04  MICRO WIDGET
05289  0153  07  SUPER WIDGET
05243  0027  15  REGULAR WIDGET

[] CSK
COPY SEQUENTIAL FILE TO KIF
 INPUT ACCESS NAME:    VOL1.KIFSRC
 OUTPUT PATHNAME:    VOL1.KIF

[] CKS
COPY KIF TO SEQUENTIAL FILE
 INPUT PATHNAME:     VOL1.KIF
 KEY:     1
 BEGINNING KEY VALUE:
 ENDING KEY VALUE:
 ASCII OR BINARY:
 OUTPUT ACCESS NAME:     VOL1.KIFKEY1
 REPLACE?     NO
3 RECORDS OUTPUT TO FILE

[] SF
SHOW FILE

 FILE PATHNAME: VOL1.KIF
05243  0027  15  REGULAR WIDGET
05289  0153  07  SUPER WIDGET
05314  0092  04  MICRO WIDGET
```

8.36.4 Commands Related to CSK

CKS (Copy KIF to Sequential File)

CSM

8.37 CSM (COPY SEQUENTIAL MEDIA)

The CSM command copies the contents of a sequential media to another sequential media. Allowable media, for either source or destination, are cassette, magnetic tape, and sequential disk file. You can also use the CSM command to verify that a copy operation is successful.

8.37.1 CSM Command Format

```
CSM
COPY/VERIFY SEQUENTIAL MEDIA UTILITY
MASTER ACCESS NAME: < acnm> @
COPY ACCESS NAME: < acnm> @
MAXIMUM RECORD LENGTH: [int]
COPY + VERIFY? (NO: = VERIFY): < yes/no>
TERMINATE ON (EOM,2EOF,COUNT): < string>
LISTING ACCESS NAME: < acnm>
```

8.37.2 CSM User Responses

System Prompts	Response Required or Optional	User Responses
MASTER ACCESS NAME:	R	Enter the access name (device or file pathname) or synonym of the sequential media from which CSM is to copy sequential information. This name must specify a sequential device (magnetic tape, cartridge tape, or cassette) or a sequential disk file.
COPY ACCESS NAME:	R	Enter the access name (device name or file pathname) or synonym for the sequential device or sequential file to which CSM is to copy sequential information. If you specify a file name, the file must already exist. CSM does not automatically create it.
MAXIMUM RECORD LENGTH:	O	Enter an integer representing the size in bytes of the largest record to be copied. *The initial value is 8512 bytes.* Due to memory limitations, the largest record that you can copy using this utility is 12,288 bytes.
COPY + VERIFY? (NO: = VERIFY):	R	Enter YES to specify both copy and verify copy operations; enter NO to specify a verify only operation.

System Prompts	Response Required or Optional	User Responses
TERMINATE ON (EOM, 2EOF, COUNT)	R	Enter the character string EOM specifying copy to end of media, the character string 2EOF specifying copy to a double end of file, or an integer representing the number of files (COUNT) CSM is to copy.
LISTING ACCESS NAME:	R	Enter the name of the listing device or file to which the report on the copy or verify operation is written.

8.37.3 CSM Command Example

The following example shows how you can enter a CSM command to copy 20 files from sequential file .TEST1 to cartridge tape unit MT02.

```
[ ] CSM
COPY/VERIFY SEQUENTIAL MEDIA UTILITY
  MASTER ACCESS NAME: .TEST1
  COPY ACCESS NAME: MT02
  MAXIMUM RECORD LENGTH: 8512
  COPY + VERIFY? (NO: = VERIFY): Y
  TERMINATE ON (EOM,2EOF,COUNT): 20
  LISTING ACCESS NAME: LP$1
```

8.37.4 Commands Related to CSM

CC (Copy/Concatenate)

CVD

8.38 CVD (COPY AND VERIFY DISK)

The CVD utility copies and optionally verifies disks used in the DX10 operating system. The copy is made from disk to disk. CVD performs the copy file by file with disk compression on all files except key indexed files (KIFs). Program files are partially compressed by releasing the unused space at the end of the file, while space made available elsewhere by deleting tasks is not released. You must execute the Copy Directory (CD) command to release this additional space in program files. CVD requires that the destination disk be the same type (for example, disks of type DS10 to DS10 or DS80 to DS80) as the source disk.

NOTE

CVD does not support a "verify only" function. Therefore, when you want verification, you must verify the copy during the copy process or you can use the more time-consuming Verify Copy (VC) command.

8.38.1 CVD Operating Procedure

The proper procedure for using CVD when DS01 is involved (either as master or copy disk) is as follows:

1. Perform an initial program load (IPL).
2. Set the date and time by using the IDT command. (Do not use the IS command.)
3. Execute CVD.

NOTE

You can initiate the CVD command with lowercase letters, but you must enter the alphabetic responses to the command prompts in uppercase letters. SCI does not perform lowercase to uppercase mapping for the prompt responses to this command, as explained in the .OPTION primitive in Volume III.

4. When CVD completes, perform an IPL and normal start-up procedures.

Performing the IPL without using the IS command ensures that memory-resident or time-delay tasks do not attempt to modify DS01 during the course of the copy.

NOTE

These conditions are not met if you have placed functions of IS in your M$00 procedure and these functions start the system log, start any tasks, or install volumes. You must remove the M$00 procedure from use before using CVD.

To execute CVD, perform the following steps:

1. Enter CVD as follows:

   [ ] CVD

2. CVD begins execution with the following message and request:

   COPY AND VERIFY DISK—81.305

   ANSWER (Y/N) QUESTIONS WITH Y FOR YES OR
   ANY OTHER CHARACTER EXCEPT $ FOR NO.

   RESPOND ANYTIME WITH $ TO RESTART

   LISTING DEVICE NAME —

3. Enter the device name for a secondary print device, such as LP01 for line printer, when the LISTING DEVICE NAME prompt appears. Press the RETURN key (cr) or enter the device name DUMY if you do not want a secondary list. (See Table 4-2 for equivalent keys on other terminals.) Once you select a listing device, you *cannot* change it.

   Indicating a device in response to the LISTING DEVICE NAME prompt provides a duplicate listing of the prompts and responses occurring at the terminal. The purpose is to direct output to a hard-copy device such as a line printer.

   When you answer prompts that request a Yes or No response, you should remember that CVD uses the *last* character entered for the response. For example, CVD interprets an answer of *YES* as having the value No, and interprets an answer of *NOY* as having the value Yes. Any character other than Y is interpreted as N.

   Entering the $ character in response to a prompt causes CVD to terminate any current operation and display the QUIT?(Y/N) prompt (as shown in step 8). This feature is enabled only when CVD has prompted for input.

4. The CVD questions that you can change begin with the following:

   COPY WITHOUT VERIFY? (Y/N) default = N

   You can choose to make a copy without verifying during the copy by responding Y.

   It is not recommended that a copy be performed without verification. You should only do this if you are going to keep the original source disk until you are sure the copy is good. However, CVD does support the capability with approximately a one third savings in the time required to perform the copy.

5. The copy utility prompts for the following input:

   MASTER DEVICE-
   VOLUME-

   Enter the device name and the volume name of the disk to be copied by the CVD utility in response to the prompts. The device and volume names entered are followed by pressing the RETURN key (cr).

   If DS01 is used as one of the devices in the copy, CVD requires that all other terminals be logged off (Q) and prohibits logging on while DS01 is being used. When DS01 is used in the copy process, log-on is disabled until CVD is terminated. If all terminals are not disabled, CVD displays the following message:

   'ALL OTHER ACTIVE TERMINALS AND TASKS MUST BE TERMINATED'
   'TYPE CR WHEN READY'

   This message is repeated until all terminals are logged off or until you enter the '$' character and press the RETURN key.

6. CVD then displays the following prompts:

   COPY DEVICE-
   VOLUME-

   Enter the destination device and volume name, each followed by the RETURN key (cr). If the destination device is the system disk, the CVD utility requests verification that the system disk is to be modified.

7. The following message appears:

   MOUNT DESIRED VOLUMES—TYPE CR WHEN READY

   Mount the desired volume(s) and press the RETURN key (cr) when ready to copy.

   When this prompt is issued, you can remove the system disk if desired, since it is no longer needed by CVD. This frees one disk unit, which can be used in the copy procedure. Be sure that no other tasks that may need the disk are operating.

   The copy operation is initiated when you press the RETURN key (cr). Should volume names differ from user-supplied names, one of the following messages appears:

   MASTER VOLUME NAME ERROR - SPECIFIED = <vnspecif>/ACTUAL = <vnactual>

   COPY VOLUME NAME ERROR - SPECIFIED = <vnspecif>/ACTUAL = <vnactual>

   where:

   vnspecif is the volume name that you specified.

   vnactual is the actual volume name.

You can elect to continue with the specified name, mount a different volume, or respecify the device and volume names. During a copy, the specified name is written to the output device as the volume name.

8. The copy operation is complete and a good backup was produced when the following messages are displayed:

```
COPY AND VERIFY COMPLETE
10 SEP 1981—13:00:00
QUIT? (Y/N)
```

When you answer N to the QUIT? (Y/N) prompt, CVD returns to the COPY WITHOUT VERIFY? (Y/N) prompt, allowing another copy/verify function. When you answer Y to the message QUIT? (Y/N), CVD terminates.

If DS01 has been used in the copy, CVD issues the following message:

```
PROCESS USED DS01—IPL REQUIRED
```

This causes the system to halt with >177 in the front panel lights of the programmer panel and turns on the fault light for both the operator and programmer panels. To continue, you must initiate an IPL.

8.38.2 CVD Error Messages

Disk copy operations are performed with safeguards for user responses and system operation. Improper use or failure of the hardware or software is a violation of those safeguards and is signaled by the following message:

```
I/O ERROR CODE zz
```

where:

zz is a two-digit hexadecimal number representing the error returned from an SVC as described in Volume VI.

If any error is not recoverable, CVD will abort the copy process and render the copy disk unusable without a subsequent initialize new volume (INV) command. This guarantees the integrity of the data on the copy disk.

CVD

8.38.3 Example CVD Operations

The following example describes a copy and verify disk operation. User responses are underlined. Pressing the RETURN (cr) key enters user responses and causes the program to continue. A comment column is included as an interpretation of responses.

	Comment
COPY AND VERIFY DISK—81.305	Announcement Banner
ANSWER (Y/N) QUESTIONS WITH Y FOR YES OR ANY OTHER CHARACTER EXCEPT $ FOR NO RESPOND ANYTIME WITH $ TO RESTART	
LISTING DEVICE NAME—LP01	Line printer
COPY WITHOUT VERIFY? (Y/N)YESN	No
MASTER DEVICE-DS01	Source disk
VOLUME-ANYTHING	
COPY DEVICE-DS03	Destination disk
VOLUME-SCRATCH	
MOUNT DESIRED VOLUMES, TYPE CR WHEN READYcr	Ready
10 SEP 1981—13:21:22	Date and time of copy
MASTER VOLUME NAME ERROR. SPECIFIED = ANYTHING. ACTUAL = SYSSAVER.	
USE NAMES AS TYPED? (Y/N) Y	Yes
COPY AND VERIFY COMPLETE	
10 SEP 1981—13:25:01	Date and time of copy
QUIT? (Y/N)Y	Yes

8.38.4 Commands Related to CVD

BD	(Backup Directory)
CD	(Copy Directory)
DCOPY	(Disk Copy/Restore)
RD	(Restore Directory)
VB	(Verify Backup)
VC	(Verify Copy)

8.39 DA (DELETE ALIAS)

The DA command deletes an alias assignment. A single DA command can delete only one alias per command.

8.39.1 DA Command Format

```
DA
DELETE ALIAS FROM PATHNAME
PATHNAME: < acnm>  @
```

8.39.2 DA Command User Response

System Prompt	Response Required or Optional	User Response
PATHNAME:	R	Enter a pathname that identifies the alias you want to delete. The alias is the last element of the pathname.

8.39.3 DA Command Example

The following example shows how you can use the DA command to delete the assignment of the alias FILEB from a file named FILE1 (see AA command example):

```
[ ] DA
DELETE ALIAS FROM PATHNAME
  PATHNAME: VOL1.PROGDIR.DIRA.FILEB
```

8.39.4 Commands Related to DA

AA (Add Alias)

DB

8.40 DB (DELETE BREAKPOINTS)

The DB command deletes a breakpoint(s) from a specified address(es) in a task and restores the original value at that address(es) that was replaced with an XOP 15,15 instruction. Refer to Volume III for a detailed description.

8.41 DCOPY (DISK COPY/RESTORE)

The DCOPY utility copies and optionally verifies disks used in the DX10 operating system. The copy is made from disk to magnetic tape, magnetic tape to disk, or disk to disk. DCOPY performs the copy track by track with no disk compression. DCOPY requires that the destination disk be the same type (for example, disks of type DS10 to DS10 or DS80 to DS80) as the source disk even when you use tape as an intermediate media.

When you use a magnetic tape unit as an output device, the first and last records are identification records of ASCII characters. Each identification record contains a volume name, the date of creation, a reel number, and data describing the type of disk used when the tape was created. The reel number contained in the identification records at the beginning and end of the tape provides the means for copying disks to or from multiple tapes.

NOTE

When making a copy of the system disk, you must verify the copy before you terminate DCOPY because one or more system files on the system disk are updated when DCOPY is terminated.

8.41.1 DCOPY Operating Procedure

The proper procedure for using DCOPY when DS01 is involved (either as master or copy disk) is as follows:

1. Perform an initial program load (IPL).
2. Set the date and time with the IDT command. (Do not use the IS command.)
3. Execute DCOPY.

NOTE

You can initiate the DCOPY command with lowercase letters, but you must enter the alphabetic responses to the command prompts in uppercase letters. SCI does not perform lowercase to uppercase mapping for the prompt responses to this command, as explained in the .OPTION primitive in Volume III.

4. When DCOPY completes, perform an IPL and normal start-up procedures.

DCOPY

Performing the IPL without the IS command ensures that memory-resident or time-delay tasks do not attempt to modify DS01 during the course of the copy.

NOTE

These conditions are not met if you have placed functions of IS in your M$00 procedure and these functions start the system log, start any tasks, or install volumes. You must remove the M$00 procedure before using DCOPY.

To execute DCOPY, perform the following steps:

1. Enter DCOPY as follows:

   [ ] DCOPY

2. DCOPY begins execution with the following announcement and prompt:

   DISK COPY/RESTORE—81.305

   ANSWER (Y/N) QUESTIONS WITH Y FOR YES OR
   ANY OTHER CHARACTER EXCEPT $ FOR NO.

   RESPOND ANYTIME WITH $ TO RESTART

   LISTING DEVICE NAME —

3. Enter the device name for a secondary print device, such as LP01 for line printer, when the LISTING DEVICE NAME prompt is displayed. Press the RETURN key (cr) or enter the device name DUMY if you do not want a secondary list. Once you select a device, you *cannot* change it.

   Indicating a device in response to the LISTING DEVICE NAME prompt provides a duplicate listing of the prompts and responses occurring at the terminal. The purpose is to direct output to a hard-copy device such as a line printer.

   When you answer prompts that request a Yes or No response, it is important to remember that DCOPY uses the *last* character entered for the response. For example, DCOPY interprets an answer of *YES* as having the value No, and interprets an answer of *NOY* as having the value Yes.

   Entering the $ character in response to a prompt causes DCOPY to terminate any current operation and display the QUIT?(Y/N) prompt (as shown in step 9). This feature is enabled only when DCOPY has prompted for input.

4. The DCOPY questions that you can change begin with the following:

   VERIFY ONLY? (Y/N)

   If verification of a previous copy is required, respond Y. To perform a copy, respond N.

5. You can choose to make a copy without verifying during the copy by responding Y to the following prompt:

   COPY WITHOUT VERIFY (Y/N) default = N

   It is not recommended that a copy be performed without verification. However, DCOPY does support the capability with approximately a one third savings in the time required to perform the copy.

6. DCOPY then displays the following prompts:

   MASTER DEVICE-
   VOLUME-

   Enter the device name and the volume name of the disk to be copied by the DCOPY utility in response to the prompts. The device and volume names entered are followed by pressing the RETURN key (cr). (See Table 4-2 for equivalent keys on other terminals.)

   If DS01 is used as one of the devices in the copy, DCOPY requires that all other terminals be logged off (Q) and prohibits logging on while DS01 is being used. When DS01 is used in the copy process, log-on is disabled until DCOPY is terminated. If all terminals are not disabled, DCOPY displays the following message:

   'ALL OTHER ACTIVE TERMINALS AND TASKS MUST BE TERMINATED'
   'TYPE CR WHEN READY'

   This message is repeated until all terminals are logged off or until you enter the $ character and press the RETURN key.

7. DCOPY then displays the following prompts:

   COPY DEVICE-
   VOLUME-

   Enter the destination device and volume names, each followed by the RETURN key (cr). If the destination device is the system disk, the DCOPY utility requests verification that the system disk is to be modified.

DCOPY

8. The following message appears:

   MOUNT DESIRED VOLUMES—TYPE CR WHEN READY

   Mount the desired volume(s) and press the RETURN key (cr) when ready to copy or verify.

   When this prompt is issued, you can remove the system disk if desired, since it is no longer needed by DCOPY. This frees one disk unit, which can be used in the copy procedure. Be sure that no other tasks that may need the disk are operating.

   The copy operation is initiated when you press the RETURN key (cr). Should volume names differ from user-supplied names, one of the following messages appears:

   MASTER VOLUME NAME ERROR - SPECIFIED = <vnspecif>/ACTUAL = <vnactual>

   COPY VOLUME NAME ERROR - SPECIFIED = <vnspecif>/ACTUAL = <vnactual>

   where:

   vnspecif is the volume name that you specified.

   vnactual is the actual volume name.

   You can elect to continue with the specified name, mount a different volume, or respecify the device and volume names. During a copy, the specified name is written to the output device as the volume name.

9. The copy operation is complete and a good backup was produced when the following messages are displayed:

   COPY AND VERIFY COMPLETE
   10 SEP 1981—13:00:00
   QUIT? (Y/N)

   When you enter N in response to the QUIT? (Y/N) prompt, DCOPY returns to the VERIFY ONLY (Y/N) prompt, allowing another copy/verify function. When you enter Y in response to the message QUIT? (Y/N), DCOPY terminates.

If DS01 has been used in the copy, DCOPY issues the following message:

PROCESS USED DS01—IPL REQUIRED

This causes the system to halt with >177 in the front panel lights of the programmer panel and turns on the fault light for both the operator and programmer panels. To continue, you must initiate an IPL.

8.41.2 DCOPY Error Messages

Disk copy operations are performed with safeguards for user responses and system operation. Improper use or failure of the hardware or software is a violation of those safeguards and is signaled by the following message:

I/O ERROR CODE zz

where:

zz is a two-digit hexadecimal number representing the error returned from an SVC as described in Volume VI.

If any error is not recoverable, DCOPY aborts the copy process and renders the copy disk unusable without a subsequent Initialize New Volume (INV) command. This guarantees the integrity of the data on the copy disk.

8.41.3 Example DCOPY Operations

The following examples describe the disk copy operation. User responses are underlined. Pressing RETURN (cr) enters user responses and causes the program to continue. A comment column is included as an interpretation of responses.

Example 1

DCOPY transfers data from disk to disk.

	Comment
DISK COPY/RESTORE—81.305	Announcement Banner
ANSWER (Y/N) QUESTIONS WITH Y FOR YES OR	
ANY OTHER CHARACTER EXCEPT $ FOR NO	
RESPOND ANYTIME WITH $ TO RESTART	
LISTING DEVICE NAME—LP01	Line printer
VERIFY ONLY? (Y/N)N	No
COPY WITHOUT VERIFY? (Y/N)YESN	No
MASTER DEVICE-DS01	Source disk
VOLUME-ANYTHING	
COPY DEVICE-DS03	Destination disk
VOLUME-SCRATCH	
MOUNT DESIRED VOLUMES, TYPE CR WHEN READYcr	Ready
10 SEP 1981—13:21:22	Date and time of copy
MASTER VOLUME NAME ERROR. SPECIFIED = ANYTHING.	
ACTUAL = SYSSAVER.	
USE NAMES AS TYPED? (Y/N)Y	Yes
COPY AND VERIFY COMPLETE	
10 SEP 1981—13:25:01	Date and time of copy
QUIT? (Y/N)Y	Yes

DCOPY

Example 2

DCOPY transfers data from disk to tape (multireel).

	Comment
DISK COPY/RESTORE—81.305	Announcement Banner
ANSWER (Y/N) QUESTIONS WITH Y FOR YES OR ANY OTHER CHARACTER EXCEPT $ FOR NO RESPOND ANYTIME WITH $ TO RESTART	
LISTING DEVICE NAME—cr	No device
VERIFY ONLY? (Y/N) N	No
COPY WITHOUT VERIFY? (Y/N)NABCDY	Yes
MASTER DEVICE-DS01	System disk
VOLUME-S	Volume S
COPY DEVICE-MT01	Magnetic tape Unit 0
VOLUME-SYSBACK1	Volume SYSBACK1
MOUNT DESIRED VOLUMES. TYPE CR WHEN READY	Ready
10 SEP 1981—13:21:22	Date and time of copy
MASTER VOLUME NAME ERROR. SPECIFIED = S ACTUAL = SYSSAVER.	
USE NAMES AS TYPED? (Y/N)Y	Yes
MOUNT TAPE #02. TYPE CR WHEN READY	Ready
COPY AND VERIFY COMPLETE	
10 SEP 1981—13:26:53	Date and time of copy
QUIT? (Y/N)Y	Yes

8.41.4 Commands Related to DCOPY

BD	(Backup Directory)
CD	(Copy Directory)
CVD	(Copy and Verify Disk)
RD	(Restore Directory)
VB	(Verify Backup)
VC	(Verify Copy)

8.42 DD (DELETE DIRECTORY)

The DD command deletes a specified directory plus all subdirectories and files in which the directory is a pathname component. However, the DD command does not delete any directory, subdirectory, or file that is delete protected. If a directory, subdirectory, or file is not deleted because it is delete protected, the DD command specifies the pathname in the listing file and indicates that it is delete protected. The DD command then continues execution. You can stop the command by pressing the CMD key.

NOTE

If a directory is delete protected, the files or directories under it are not automatically delete protected. The catalog is delete protected, but each data file must have been delete protected individually. If the files are delete protected, changing the protection of the directory entry by using the Modify File Protection (MFP) command does not change the protection of the data files cataloged in the directory. You must use the MFP command to modify the protection of each data file individually. You can create a batch stream to modify the protection of all files in a given directory. You cannot modify the protection of files to which a LUNO is currently assigned (that is, a file being currently used).

8.42.1 DD Command Format

```
DD
DELETE DIRECTORY
PATHNAME: < acnm> @
LISTING ACCESS NAME: [acnm]@
ARE YOU SURE?: < yes/no>
```

8.42.2 DD Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname identifying the directory you want to delete.
LISTING ACCESS NAME:	O	Enter the device name or the file pathname to which DD should write the results of the delete operation. *The default value is the terminal local file (TLF).*
ARE YOU SURE?:	R	Enter Y (yes) to specify that you are sure you want to delete the directory. Enter N (no) to cancel the command.

8.42.3 DD Command Example

The following example shows how you can enter the Map Disk (MD) command to map a directory and then the DD command to delete the directory:

```
 [] MD
MAP DISK
 PATHNAME:     .EXAMPLE
 LISTING ACCESS NAME:    ME
 SHORT FORM?: YES
 TOP LEVEL ONLY?: NO
 DIRECTORY NODES ONLY?: NO
 EXECUTION MODE (F,B): FOREGROUND
     DISK MAP OF .EXAMPLE
       TODAY IS 16:56:30 TUESDAY, JAN 17, 1978.

                                          NUMBER CURRENT TOTAL
                                     FILE   OF     EOM   ALLOC
 LV NAME                             TYPE RECORDS  ADU    ADU        LAST UPDATE
 -------                             ---- ------- ------ -----    --------------

     EXAMPLE                          D    12      4       4         1/17/78  16:54:35
 1  EXAMPLE:                                           FILES=2  AVAILABLE=9
      DIR1                            D    24      8       8         1/17/78  16:56: 0
      FILE1                           S    0       0       1         1/17/78  16:54:35
 2  DIR1:                                              FILES=2  AVAILABLE=20
       FILE1                          R    0       0       1         1/17/78  16:50:23
       FILE2                          K            59      101       1/17/78  16:56: 2
**.VCATALOG.EXAMPLE.DIR1                               TOTAL  SIZE = 110 ADUS
**.VCATALOG.EXAMPLE                                    TOTAL  SIZE = 115 ADUS

 []

      DD
      DELETE DIRECTORY
       PATHNAME:    .EXAMPLE
       LISTING ACCESS NAME:   ME
       ARE YOU SURE?:   YES
        DELETED   .EXAMPLE.DIR1.FILE1
        DELETED   .EXAMPLE.DIR1.FILE2
        DELETED   .EXAMPLE.DIR1
        DELETED   .EXAMPLE.FILE1
        DELETED   .EXAMPLE
```

8.42.4 Commands Related to DD

CFDIR	(Create Directory File)
LD	(List Directory)

8.43 DF (DELETE FILE)

The DF command deletes a file. If the file is a directory, it must be empty or the command is aborted. You can delete nonempty directories only by using the Delete Directory (DD) command. If the file to be deleted is delete protected, the DF command fails and reports an error.

8.43.1 DF Command Format

```
DF
DELETE FILE
PATHNAME: (< acnm>) @
```

8.43.2 DF Command User Responses

System Prompt	Response Required or Optional	User Response
PATHNAME:	R	Enter the access name(s) of the file(s) you want to delete.

8.43.3 DF Command Example

The following example shows how you can use the DF command to delete two files:

```
[ ] DF
DELETE FILE
  PATHNAME: .JIM,.TEMP
```

8.43.4 Commands Related to DF

CF (Create File)

DISC

8.44 DISC (TERMINAL DISCONNECTION)

The DISC command terminates a connection. If no task is active with respect to the remote terminal, the connection is terminated. A completion message appears on the screen.

8.44.1 DISC Command Format

```
DISC
TERMINAL DISCONNECTION
TERMINAL ACCESS NAME: < acnm>
```

8.44.2 DISC Command User Responses

System Prompt	Response Required or Optional	User Response
TERMINAL ACCESS NAME:	R	Enter the name that identifies the terminal to be disconnected.

After you enter the requested name, a completion message appears on the screen. The message shows the status of the disconnection. If the response to TERMINAL ACCESS NAME is ME, a Quit (Q) command is performed and no message appears.

8.44.3 DISC Command Example

The following example shows how you can use the DISC command to terminate a connection initiated from a remote terminal:

```
[ ] DISC
TERMINAL DISCONNECTION
  TERMINAL ACCESS NAME: ME
```

8.44.4 Commands Related to DISC

ANS	(Answer Incoming Call)
CALL	(Call Terminal)
LHPC	(List Hard-Copy Terminal Port Characteristics)
MHPC	(Modify Hard-Copy Terminal Port Characteristics)

8.45 DL (DELETE LINES)

After you activate the Text Editor, the DL command specifies those lines that are to be deleted from the current file. Refer to Volume IV for a detailed description.

DO

8.46 DO (DELETE OVERLAY)

The DO command removes a previously installed overlay from a program file. You can specify the overlay by name or by installed ID.

You cannot delete a delete-protected overlay unless you remove the delete protection by using the Modify Overlay Entry (MOE) command.

8.46.1 DO Command Format

```
DO
DELETE OVERLAY
PROGRAM FILE OR LUNO: < acnm> @ or < int> @
OVERLAY NAME OR ID: < string> @ or < int> @
```

8.46.2 DO Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter the file name of, or the LUNO assigned to, the program file on which the overlay has been installed. If you specify a LUNO in response to this prompt, it must be assigned to the program file prior to execution of the DO command.
OVERLAY NAME OR ID:	R	Enter the name or ID of the overlay that you want to delete from the specified program file.

8.46.3 DO Command Example

The following example shows how you can use the DO command to delete the overlay named OLAY1 from the program file specified by LUNO > 80.

```
[ ] DO
DELETE OVERLAY
  PROGRAM FILE OR LUNO: > 80
  OVERLAY NAME OR ID: OLAY1
```

8.46.4 Commands Related to DO

DP	(Delete Procedure)
DT	(Delete Task)
IO	(Install Overlay)
MOE	(Modify Overlay Entry)

8.47 DP (DELETE PROCEDURE)

The DP command removes a previously installed procedure from a program file. You can specify the procedure by name or by installed ID.

You cannot delete a delete-protected procedure until you remove the delete protection by using the Modify Procedure Entry (MPE) command. If you delete the attached procedure of a task, the task will not execute. You must reinstall the procedure to execute the task.

8.47.1 DP Command Format

```
DP
DELETE PROCEDURE
PROGRAM FILE OR LUNO: < acnm> @ or < int> @
PROCEDURE NAME OR ID: < string> @ or < int> @
```

8.47.2 DP Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter the file name of, or the LUNO assigned to, the program file in which the procedure is installed. If you specify a LUNO in response to this prompt, it must be assigned to the program file prior to executing the DP command.
PROCEDURE NAME OR ID:	R	Enter the name or ID of the procedure you want to delete from the specified program file.

8.47.3 DP Command Example

The following example shows how you can use the DP command to delete a procedure named MYPROC from the program file specified by LUNO > 7B.

```
[ ] DP
DELETE PROCEDURE
  PROGRAM FILE OR LUNO: > 7B
  PROCEDURE NAME OR ID: MYPROC
```

8.47.4 Commands Related to DP

DO	(Delete Overlay)
DT	(Delete Task)
IP	(Install Procedure)
MPE	(Modify Procedure Entry)

DPB

8.48 DPB (DELETE AND PROCEED FROM BREAKPOINT)

The DPB command proceeds from a breakpoint at which a task is currently stopped and deletes that breakpoint. Refer to Volume III for a detailed description.

8.49 DS (DELETE STRING)

After you activate the Text Editor, the DS command causes the system to search each line in the current file, beginning with the current line, for the specified string of characters and to delete the string when it is found. Refer to Volume IV for a detailed description.

DSB

8.50 DSB (DELETE SIMULATED BREAKPOINTS)

The DSB command deletes a list of simulated breakpoints that were assigned using the Assign Simulated Breakpoint (ASB) command. Refer to Volume III for a detailed description.

8.51 DT (DELETE TASK)

The DT command removes a previously installed task from a program file. You can specify the task either by name or by installed ID. If associated overlays exist, they are also deleted if they are not delete protected.

You cannot delete a delete-protected task until you remove the delete protection by using the Modify Task Entry (MTE) command.

8.51.1 DT Command Format

```
DT
DELETE TASK
PROGRAM FILE OR LUNO: < acnm> @ or < int> @
TASK NAME OR ID: < string> @ or < int> @
```

8.51.2 DT Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter the file name of, or the LUNO assigned to, the program file on which the task is installed. If you specify a LUNO in response to this prompt, it must be assigned to the program file prior to execution of the DT command.
TASK NAME OR ID:	R	Enter the name or ID of the task you want to delete from the specified program file.

8.51.3 DT Command Example

The following example shows how you can use the DT command to delete a task with an ID of >83 from a program file specified by LUNO >4A.

```
[ ] DT
DELETE TASK
  PROGRAM FILE OR LUNO: >4A
  TASK NAME OR ID: >83
```

8.51.4 Commands Related to DT

DO	(Delete Overlay)
DP	(Delete Procedure)
IT	(Install Task)
MTE	(Modify Task Entry)

DUI

8.52 DUI (DELETE USER ID)

Authorized users can enter a DUI command to delete a user's identification information to prevent that user from accessing DX10. If a user is currently logged in under the user ID being deleted, the user ID is not deleted and the following error message is returned:

**** ERROR 9021 **** USER ID IN USE

You can delete your own user ID providing you are not performing an activity in batch mode and your privilege code is equal to the privilege level assigned to the DUI procedure.

8.52.1 DUI Command Format

DUI
DELETE USER ID
USER ID: < name>

8.52.2 DUI Command User Responses

System Prompt	Response Required or Optional	User Response
USER ID:	R	Enter the user ID you want to remove from the system.

8.52.3 DUI Command Example

The following example shows how you can enter the List User IDs (LUI) command to show the existing user ID, passcode, and privilege code assigned to a user. Then the example shows how you can enter the DUI command to delete the user's ID, and another LUI command to show that the user ID has been deleted.

```
[] LUI
LIST USER IDS
 OUTPUT ACCESS NAME:
 USER ID      DESCRIPTION       PRIVILEGE CODE
    SYS001      SYSTEM                 3
    HERB03      HERB                   7
    EXM005      EXAMPLE USER           4
    KDK007      KIRBY                  7
    JCK008      CLARK                  7
    JCK010      CLARK                  7
    KWD011      KENNETH                7
    DAR012      DARLENE                6
    SUE020      SUE                    7
    MLH022      MARILYN                7
01:55:30 SATURDAY, JUL 29, 1978.
```

```
[] DUI
DELETE USER ID
 USER ID:    EXM005
[] LUI
LIST USER IDS
 OUTPUT ACCESS NAME:
 USER ID     DESCRIPTION      PRIVILEGE CODE
    SYS001     SYSTEM                3
    HERB03     HERB                  7
    KDK007     KIRBY                 7
    JCK008     CLARK                 7
    JCK010     CLARK                 7
    KWD011     KENETH                7
    DAR012     DARLENE               6
    SUE020     SUE                   7
    MLH022     MARILYN               7
01:56:13 SATURDAY, JUL 29, 1978.
```

8.52.4 Commands Related to DUI

AUI	(Assign User ID)
LUI	(List User IDs)
MUI	(Modify User ID)

DXTX

8.53 DXTX (CONVERT DX10 FILE TO TX990 DISKETTE FILE)

The DXTX command invokes the conversion (DX10 files to TX990 diskette files) processor. The DXTX processor converts DX10 sequential or relative record files to TX990 sequential or relative record noncontiguous files respectively. TX990 does not support contiguous files.

8.53.1 DXTX Command Format

```
DXTX
DX10 FILE TO TX990 FILE
DX10 ACCESS NAME: < acnm> @
TX990 ACCESS NAME: < acnm>
REPLACE?: < yes/no>
```

8.53.2 DXTX Command User Responses

System Prompts	Response Required or Optional	User Responses
DX10 ACCESS NAME:	R	Enter a valid DX10 access name.
TX990 ACCESS NAME:	R	Enter a valid TX990 access name. The access name must be unique.
REPLACE?:	R	Enter YES (replace flag) or NO. If the file exists, an error is returned for a N (no) reply. The initial value is NO.

If this TX990 diskette file does not exist, a noncontiguous file of the input file type is created.

8.53.3 DXTX Command Example

```
[ ] DXTX
DX10 FILE TO TX990 FILE
  DX10 ACCESS NAME: DK01.FILEA
  TX990 ACCESS NAME: DK02:FILEA
  REPLACE?: YES
```

8.53.4 Commands Related to DXTX

IBMUTL (Convert IBM-Formatted Diskette Data-Sets)
TXCM (Compress Diskette File)
TXCP (Change Diskette File Protection)
TXDF (Delete Diskette File)
TXDX (Convert TX990 Diskette File to DX10 File)
TXFD (Format Diskette)
TXMD (Map Diskette)
TXSF (Set System File)
TXXLE (Execute Link Editor to Produce TX Program File)

8.54 EBATCH (END BATCH EXECUTION)

The EBATCH command indicates there are no more commands in the batch stream to process. EBATCH functions as a no-operation in foreground mode and flags partially completed CFKEY sequences in batch mode. The BATCH command should be the first command in a batch stream, and the EBATCH command should be the last command in a batch stream.

8.54.1 EBATCH Command Format

EBATCH (END BATCH EXECUTION),
LS (LIST SYNONYMS)?: [yes/no] (NO)

8.54.2 EBATCH Command User Responses

System Prompts	Response Required or Optional	User Responses
LS (LIST SYNONYMS)?:	O	If you specify YES in response to this prompt, the List Synonyms (LS) command lists all synonyms on the batch listing file. If you specify NO, it does not list the synonyms. The default value is NO.

8.54.3 EBATCH Command Example

The following example shows a batch stream that performs a weekly backup of data and also verifies the backup. The BATCH command is the first command in the batch stream, and the EBATCH is the last.

```
BATCH   LS = YES
*
****** ASSIGN LUNO TO TAPE
*
AL      AN = @CT,
        PF = NO
EC
*
****** REWIND TAPE
*
RWL     LUNO = @$$LU
EC
*
****** UNBLOCKED BACKUP
*
BD      DIR = @WD,
        SEQ = @SEQ,
EC
*
****** VERIFY
*
RWL     L = @$$LU
VB      SEQ = @SEQ,
        DIR = @WD,
EC
*
EBATCH
```

EBATCH

8.54.4 Commands Related to EBATCH

BATCH (Begin Batch Execution)

8.55 EC (BATCH STREAM ERROR COUNTER)

The EC command collects the batch stream error counter errors encountered by commands in batch streams. Most commands in DX10 set the synonym $$CC if an error occurs. The EC command tests the value of the synonym $$CC which is set when the command processor completes. When the value of $$CC is not zero, an error accumulation synonym, EC, is incremented by one. You can subsequently display the value of EC by using the Create Message (CM) command.

If an error was encountered in the previous command, the EC command prints a banner line that helps you identify where the error occurred. Otherwise, it does not print anything. This banner line is especially useful in identifying an error in a batch stream.

8.55.1 EC Command Format

EC

8.55.2 EC Command User Responses

No user responses are required.

8.55.3 EC Command Example

The following example shows a banner line that the EC command prints.

```
**<>*<>*<>*<>*<>*<> ERROR -- $E$C = 1 <>*<>*<>*<>*<>*<>*<>*
```

FB

8.56 FB (FIND BYTE)

The FB command searches for the specified value(s) in a memory area of a task. The search begins on a byte boundary. Refer to Volume III for a detailed description.

8.57 FL (FORWARD SPACE LUNO)

The FL command moves the current data-access position forward one or more records.

8.57.1 FL Command Format

```
FL
FORWARD SPACE LUNO
LUNO: <int>
RECORD COUNT: <int>
```

8.57.2 FL Command User Responses

System Prompts	Response Required or Optional	User Responses
LUNO:	R	Enter the LUNO assigned to the sequential device or file to be forward-spaced.
RECORD COUNT:	R	Enter the number of records (from 1 to 65535) that the sequential file or device is to be forward-spaced. If DX10 encounters an EOF before forward spacing the number of records specified, it positions the sequential device or file at the beginning of the record following the EOF.

8.57.3 FL Command Example

The following example shows how you can enter an FL command to space forward three records on a magnetic tape unit assigned to LUNO 32:

```
[ ] FL
FORWARD SPACE LUNO
  LUNO: 32
  RECORD COUNT: 3
```

8.57.4 Commands Related to FL

BL (Backspace LUNO)
RWL (Rewind LUNO)

FS

8.58 FS (FIND STRING)

After you activate the Text Editor, the FS command specifies a character string the system is to find, beginning with the current line. When the specified occurrence of the string is found, the system displays the line that contains the string. Refer to Volume IV for a detailed description.

8.59 FW (FIND WORD)

The FW command searches for the specified value(s) in a memory area of a task. The search begins on a word boundary. Refer to Volume III for a detailed description.

HO

8.60 HO (HALT OUTPUT AT DEVICE)

You can enter the HO command to temporarily halt queued output at a given device (for example, to change forms at a line printer). The HO command allows you to optionally specify when the output is to be halted. You can enter I if you want halt immediate and E if you want halt at the end-of-file (EOF).

You can release halted output by entering the Resume Output (RO) command. Entering the RO command resumes output from the point at which output had been halted.

8.60.1 HO Command Format

```
HO
HALT OUTPUT AT DEVICE
DEVICE NAME: < acnm> @
IMMEDIATELY OR AT EOF? (I/E): [string]
```

8.60.2 HO Command User Responses

System Prompts	Response Required or Optional	User Responses
DEVICE NAME:	R	Enter the device name of the device at which you want to halt queued output.
IMMEDIATELY OR AT EOF (I/E):	O	Enter I if you want output halted immediately. Enter E if you want to halt output at the EOF. *The default value is an immediate halt.* Any character typed other than I is taken as an E response; halt at EOF.

8.60.3 HO Command Example

The following example shows how you can enter an HO command to immediately halt output at a line printer:

```
[ ] HO
HALT OUTPUT AT DEVICE
  DEVICE NAME: LP01
  IMMEDIATELY OR AT EOF? (I/E): I
```

8.60.4 Commands Related to HO

KO	(Kill Output at Device)
RO	(Resume Output at Device)
SOS	(Show Output Status)

8.61 HT (HALT TASK)

The HT command unconditionally suspends a task at the end of the current time slice. Refer to Volume III for a detailed description.

8.62 IBMUTL (DX10 IBM CONVERSION UTILITY PROGRAM)

The DX10 IBM Conversion Utility (IBMUTL) Program transfers standard IBM-formatted diskette data-sets to DX10 files and DX10 files to standard IBM-formatted diskette data-sets. IBMUTL also formats diskettes to standard IBM specification for single-density diskettes (as designated in "The IBM Diskette for Standard Data Interchange," GA21-9182-2, which you can order from IBM).

IBMUTL works only with single-sided, single-density diskettes. You can insert these diskettes in either a CRU floppy diskette drive (FD800) or a double-sided, double-density (DSDD) diskette drive (FD1000).

IBMUTL allows you to read or write data-sets on an IBM-formatted diskette used by systems and devices that are based on IBM sequentially-sectored diskettes that use the EBCDIC character set. The IBM-formatted diskette may already contain data-sets created by another process or may have been newly formatted by this utility or other means. All preexisting data-sets are preserved.

The basic exchange data-sets supported by DX10 have requirements that assure that diskettes can be exchanged between systems. These requirements are as follows:

- The exchange type indicator (data-set label position 44) must be blank.
- The data-set is organized sequentially.
- The records are a maximum of 128 bytes long.
- The records are of fixed length, unblocked, and unspanned.
- The data-set identifier (data-set label position 6-22) is not longer than eight positions.
- The diskette is initialized with physically sequential records (volume label positions 77-78 equal blanks or 01).
- The basic exchange data-sets use only tracks 1-73.
- An alternative physical record relocation is not used in a basic exchange data-set.

8.62.1 Formatting an IBM Diskette

Entering the F (format) response to the SERVICE? prompt causes a diskette to be formatted to IBM format. If more than two bad tracks are found or if track zero is bad, the diskette is unusable, and you should use another diskette. Track zero contains the data-set headers (sectors 8-26) and other information about the diskette (sectors 1-7). The data-set headers are written to include name, record length, beginning of extent (BOE), end of extent (EOE), and end of data (EOD) fields only. All other sectors are left in the initialized state (blank).

The diskette cannot be formatted correctly on double-sided, double-density (DSDD) diskette drives if the diskette has any bad tracks. The CRU floppy diskette drive is capable of handling the bad tracks so that IBM equipment recognizes the bad tracks.

8.62.2 Transferring DX10 Files to IBM Data-Sets

You must use a standard DX10 pathname to specify the DX10 files that you want to convert to IBM format. The new data-set will begin with the first available label following the last used label in the IBM diskette directory. This utility skips empty labels between used labels. The name of the new data-set may be the same as an already existing data-set, but the existing data-set is not replaced. This utility cannot create multivolume data-sets on IBM diskettes.

8.62.3 Transferring IBM Data-Sets to DX10 Files

You must specify the desired data-set label and the DX10 pathname. The data-set labels from the IBM diskette are displayed when you enter the transfer command. If the DX10 file does not already exist, it is created as an expandable sequential file. If two data-sets have the same name, this utility can only access the first data-set. You can transfer multivolume data-sets from IBM diskettes to DX10 files.

8.62.4 Operator Interaction

8.62.4.1 Special Characters. IBMUTL recognizes two special characters: the asterisk (*) and the ampersand (&). The following list explains how to use these characters:

- When entered in response to a prompt, an asterisk (*) followed by a carriage return terminates IBMUTL.
- When entered in response to a prompt, an ampersand (&) followed by a carriage return restarts IBMUTL by requesting the IBM diskette drive name.

8.62.4.2 Execution. The SCI command IBMUTL allows you to use the DX10 conversion utility. Throughout this discussion, all utility messages and prompts are shown in uppercase letters, while user responses are indicated by lowercase letters. Each message or prompt is listed with any necessary explanatory comments. These messages and prompts are listed approximately in the order the utility can produce them. Three examples are given at the end of this IBMUTL command description.

IBMUTL 79.319 IBM DISKETTE CONVERSION UTILITY

This is the first message to appear after entering the utility.

COUNTRY NAME (USA):

The following list describes the character set used:

UK	England
UK1	USA
JA	Japan — Nippon
FR	France Data Processing/Belgium
FRWP	French Word Processing
BE	Belgium/France
DE	Denmark/Norway
NOR	Norway/Denmark
GE	Germany/Austria
AU	Austria/Germany
SP	Spanish Speaking Countries
SWIS	Switzerland
SW	Sweden/Finland
FI	Finland/Sweden

IBMUTL

DISKETTE DRIVE NAME: DK01	Enter a legal DX10 device name for the diskette.
SERVICE? F = FORMAT, T = TRANSFER:	Enter F to initiate the formatting function. Enter T to initiate the transfer function. You should select the format function only when first writing on the diskette.
ARE YOU SURE? Y/N:	Enter Y to continue, N to abort.
FORMAT IN PROGRESS . . . FORMAT COMPLETE.	These two messages appear only after the FORMAT process has been selected. The utility will then request the diskette drive name again.
ARE THESE RPGII FILES? Y/N:	If RPG uses the files being converted, answer Y. Otherwise, answer N. Special conversion characters are required for RPG files.
FILE1 FILE2 FILE3	If you entered T in response to the SERVICE? F = FORMAT, T = TRANSFER prompt, all data-set labels are printed.
DISKETTE NOT IBM FORMAT	The specified diskette is not an IBM-formatted diskette. Proper format is verified when listing data-set labels. The utility then requests the diskette drive name again.
FUNCTION? F = FILE TO DATASET, D = DATASET TO FILE:	File to data-set means DX10 file to IBM diskette file. Data-set to file means IBM diskette to DX10 file. Enter F to initiate the file to data-set function. Enter D to initiate the data-set to file function.
CHARACTERS PER RECORD? 2 MIN. - 128 MAX.	Enter an integer between 2 and 128. The initial value is 80.
TI FILE PATHNAME:	Enter a DX10 file pathname. The pathname can be a maximum of 26 characters long.
TI FILE PHYSICAL RECORD LENGTH:	If the data-set-to-file function is being performed, specify a physical record length for the TI file being created. The initial value for the physical record length is 288.
IBM DATASET NAME:	Enter a diskette data-set label. The maximum length is eight characters.
FILE TRANSFER IN PROGRESS . . .	This message indicates that a transfer of data is in progress.

MOUNT NEXT VOLUME, REPLY "C" WHEN READY: c	When reading multivolume diskettes, this message appears at the end of any volume expecting a continuation of another volume. Reply after changing this diskette. You cannot create multivolume diskettes by using this utility. Many of the preceding prompts are requested for succeeding volumes.
TRANSFER COMPLETE	This message indicates successful completion of the transfer function. The utility then prompts for the number of characters per record for the next data-set.
UTILITY SERVICE TERMINATED	If no more data-sets are to be transferred, you can reply with the special character * for any prompt. This message then appears.
ARE THERE ANY PACKED DECIMAL DATA FIELDS? Y/N:	This prompt appears if you selected the data-set to file function and you specified a value other than the default (80) in response to the CHARACTERS PER RECORD prompt. Answer Y for yes or N for no.
SPECIFY STARTING POSITION IN RECORD:	These two prompts appear only as a result of a Y response to the packed decimal prompt.
SPECIFY ENDING POSITION IN RECORD:	Enter the leftmost and rightmost character position of the packed decimal field. The utility then repeats these prompts until you enter a null value. You can specify a maximum of 20 fields. Numeric values of the range from 0 through 128 are permitted. You do not need to justify numbers. Leading and trailing blanks are acceptable. The utility does not convert the field selected here. You should also specify binary fields.

8.62.5 Error Reporting and Recovery

IBMUTL reports errors encountered during execution to you in accordance with Table 8-2. Whenever recovery from such errors is possible, the program returns to a logical restart point and continues its function.

Table 8-2. IBMUTL Error Messages

Message	Meaning	Recovery
DISKETTE NOT IBM FORMAT	Specified diskette is not an IBM format diskette.	Insert a properly formatted diskette in specified drive, or return to DISKETTE DRIVE NAME prompt (enter &) and enter the correct drive name.
DISKETTE NOT SINGLE DENSITY	Specified diskette is not single density.	Insert a single-density diskette in the drive. IBMUTL supports only single-sided, single-density diskettes.
PATHNAME UNDEFINED	Illegal pathname has been entered.	Validate the pathname and reenter it.
DISKETTE DIRECTORY FULL	An attempt to exceed the maximum number of data-sets (19) allowable per IBM formatted diskette.	Program control returns to DISKETTE DRIVE NAME prompt. Install new IBM format diskette and retry transfer function.
IBM DISKETTE IS FULL	Data capacity of IBM format diskette has been exceeded. The data-set being written to contains as much data as could be transferred. Also occurs when the file-to-data-set function is selected and the diskette is already full.	Program control returns to DISKETTE DRIVE NAME prompt. You cannot produce multivolume diskettes by using this utility.
TOO MUCH DATA	The IBM diskette is not a basic exchange diskette. The data-set to be transferred extends past track 73.	None. The IBM utility only handles basic exchange data-sets. (Refer to the introduction to this IBMUTL command description.)
MORE THAN 2 BAD TRACKS, FORMAT ABORTED	Bad diskette.	Program control returns to DISKETTE DRIVE NAME prompt. Install new diskette and retry format function.
FILE SERVICE ERROR nn	Error encountered while accessing DX10 user file. Refer to Volume VI.	Program control returns to DISKETTE DRIVE NAME prompt. Respond according to the individual error code.
DISKETTE ACCESS ERROR nn	Error encountered while accessing IBM data-set. Refer to Volume VI.	Program control returns to DISKETTE DRIVE NAME prompt. Respond according to individual error code.

Table 8-2. IBMUTL Error Messages (Continued)

Message	Meaning	Recovery
I/O ERROR nn	I/O error encountered during program execution. Refer to Volume VI.	Program control returns to DISKETTE DRIVE NAME prompt. Respond according to individual error code.
DATA-SET NAME NOT FOUND	Specified data-set name not found on diskette.	Program control prompts for data-set name again. Respond with correct data-set name.
FATAL I/O ERROR	An unrecoverable read/write error was encountered. IBMUTL is terminated.	Try to execute IBMUTL again.
WRONG DEVICE TYPE	A device other than a disk device (DSxx or DKxx) was specified in response to the DISKETTE DRIVE NAME prompt.	Execute IBMUTL again and identify a disk device in response to the DISKETTE DRIVE NAME prompt.

IBMUTL

8.62.6 Examples

The following examples show various ways you can use the IBMUTL utility. User responses are underlined. The first example shows how you can use the IBMUTL utility to format a fresh diskette to IBM format:

```
IBMUTL — 79.319 IBM DISKETTE CONVERSION UTILITY
COUNTRY NAME (USA):
SELECTED COUNTRY: USA
DISKETTE DRIVE NAME: DK01
SERVICE? F = FORMAT, T = TRANSFER: F
FORMAT IN PROGRESS . . .
FORMAT COMPLETE
DISKETTE DRIVE NAME: *
UTILITY SERVICE TERMINATED
```

The second example shows how you can use the IBMUTL utility to transfer a DX10 file to a diskette data-set:

```
IBMUTL — 79.319 IBM DISKETTE CONVERSION UTILITY
COUNTRY NAME (USA):
SELECTED COUNTRY: USA
DISKETTE DRIVE NAME: DK01
SERVICE? F = FORMAT, T = TRANSFER: T
ARE THESE RPGII FILES? Y/N: Y
FUNCTION? F = FILE TO DATASET, D = DATASET TO FILE: F
CHARACTERS PER RECORD? 2 MIN. - 128 MAX: 128
TI FILE PATHNAME: vol.directory.filename
IBM DATASET NAME: data-set label
FILE TRANSFER IN PROGRESS . . .
TRANSFER COMPLETE
SERVICE? F = FORMAT, T = TRANSFER: *
UTILITY SERVICE TERMINATED
```

The third example shows how you can use the IBMUTL utility to transfer a diskette data-set to a DX10 file:

```
IBMUTL — 79.319 IBM DISKETTE CONVERSION UTILITY
COUNTRY NAME (USA):
SELECTED COUNTRY: USA
DISKETTE DRIVE NAME: DK01
SERVICE? F = FORMAT, T = TRANSFER: T
ARE THESE RPGII FILES? Y/N: N

FUNCTION? F = FILE TO DATASET, D = DATASET TO FILE: D
CHARACTERS PER RECORD? 2 MIN. - 128 MAX: 128
ARE THERE ANY PACKED DECIMAL FIELDS? Y/N: Y
SPECIFY STARTING POSITION IN RECORD: 81
SPECIFY ENDING POSITION IN RECORD: 86
SPECIFY STARTING POSITION IN RECORD:
TI FILE PATHNAME: vol.directory.filename
IBM DATASET NAME: data-set label
FILE TRANSFER IN PROGRESS . . .
TRANSFER COMPLETE
SERVICE? F = FORMAT, T = TRANSFER: *
UTILITY SERVICE TERMINATED
```

8.62.7 Commands Related to IBMUTL

DXTX	(Convert DX10 File to TX990 Diskette File)
TXCM	(Compress Diskette File)
TXCP	(Change Diskette File Protection)
TXDF	(Delete Diskette File)
TXDX	(Convert TX990 Diskette File to DX10 File)
TXFD	(Format Diskette)
TXMD	(Map Diskette)
TXSF	(Set System File)
TXXLE	(Execute Link Editor to Produce TX Program File)

IDS

8.63 IDS (INITIALIZE DISK SURFACE)

The IDS command analyzes the disk surface, checking for flaws in the medium and marking any bad tracks. This command does not check the disk drive or controller. IDS may not find all bad tracks; therefore, you must enter any bad tracks that the manufacturer lists on the medium. When the surface is initialized, any bad track information that is on the disk is included in the bad track list placed on the disk as a result of a previous surface analysis.

Each disk must have the surface initialized before the disk can be used. You only need to use the IDS command once. Once the surface is initialized, the INV command is used to format the volume. Alternatively, the IDS command performs both surface and volume initialization.

When you perform the IDS command, the disk cannot be installed and the write protect must be off.

8.63.1 IDS Command Format

```
IDS
INITIALIZE DISK SURFACE
UNIT NAME: < name> @
INITIALIZE NEW VOLUME?: < yes/no>
BAD TRACK ACCESS NAME: [acnm]
LISTING ACCESS NAME: [acnm]
EXECUTION MODE (F,B): < F/B>
```

If you enter YES in response to INITIALIZE NEW VOLUME?, the following prompts appear:

```
VOLUME NAME: < name>  @
NUMBER OF VCATALOG ENTRIES: [int]
DEFAULT PHYSICAL RECORD SIZE: [int]
HARDWARE INTERLEAVING FACTOR: [int]
USED AS SYSTEM DISK?: < yes/no>
```

8.63.2 IDS Command User Responses

System Prompts	Response Required or Optional	User Responses
UNIT NAME:	R	Enter the device name of the disk unit in which the volume to be initialized is mounted.
INITIALIZE NEW VOLUME?:	R	Enter YES if you want the IDS command to automatically perform the INV function after initializing the surface. (Refer to the INV command for further information on volume initialization.) Enter NO if you only want to test the disk for bad tracks and format the disk. An INV command is required to make the volume usable to the system.

System Prompts	Response Required or Optional	User Responses
BAD TRACK ACCESS NAME:	O	Enter the pathname of the file or device that is to supply bad track information. The default is a null response (press RETURN key), which indicates there is no bad track information. If you enter ME in response to the prompt, SCI prompts for bad track information as follows: ENTER BAD TRACKS IN THE FORMAT: HEAD, CYLINDER; OR HEAD, CYLINDER; HEAD, CYLINDER; and so on. TO END LIST, ENTER AN EMPTY LINE. Enter bad track information in the requested format. All numbers entered are assumed to be decimal. For a T25, the system uses half of the cylinder that a T50 uses. The bad tracks labeled on the side of the disk pack are labeled as if the pack were formatted for a T50. When you enter those bad tracks for a T25, ignore all odd numbered cylinders. Divide each even numbered cylinder by two and enter the result as the cylinder number of the bad track.
LISTING ACCESS NAME:	O	Enter the pathname of the file or device to which IDS lists the results of the disk surface initialization. A null response specifies the terminal local file (TLF).
EXECUTION MODE (F,B):	R	Enter F (foreground) to execute the task in foreground mode. Enter B (background) to execute the task in background mode. *The initial value is FOREGROUND.*

IDS

Refer to the INV command for information concerning responses to the following prompts:

```
VOLUME NAME: < name>  @
NUMBER OF VCATALOG ENTRIES: [int]
DEFAULT PHYSICAL RECORD SIZE: [int]
HARDWARE INTERLEAVING FACTOR: [int]
USED AS SYSTEM DISK?: < yes/no>
```

8.63.3 IDS Command Example

In the following example, the IDS command initializes the surface of the disk mounted in device DS02. The write protect must be off and the disk must not be installed.

```
[ ] IDS
INITIALIZE DISK SURFACE
  UNIT NAME: DS02
  INITIALIZE NEW VOLUME?: NO
  BAD TRACK ACCESS NAME: .BADTRACK
  LISTING ACCESS NAME:
  EXECUTION MODE (F,B): FOREGROUND
```

8.63.4 Commands Related to IDS

INV	(Initialize New Volume)

8.64 IDT (INITIALIZE DATE AND TIME)

You can reinitialize the system date and time at any time by entering the IDT command. You specify the year, month, day, hour, and minute with the IDT command. Upon receipt of the command, DX10 starts counting the seconds at zero.

8.64.1 IDT Command Format

```
IDT
INITIALIZE DATE AND TIME
YEAR: < int>
MONTH: < string>
DAY: < int>
HOUR: < int>
MINUTE: < int>
```

8.64.2 IDT User Responses

System Prompts	Response Required or Optional	User Responses
YEAR:	R	Enter two to four numeric characters representing the current year. For example, 82 and 1982 both mean 1982.
MONTH:	R	Enter one or two numeric characters representing the current month or the first three alphabetic characters of the month.
DAY:	R	Enter one or two numeric characters representing the current day.
HOUR:	R	Enter one or two numeric characters representing the current hour (according to the 24-hour clock).
MINUTE:	R	Enter one or two numeric characters representing the current minute.

IDT

8.64.3 IDT Command Example

This example shows how you can enter an IDT command to initialize the system time and date for 1:34 PM on June 5, 1977:

```
[ ] IDT
INITIALIZE DATE AND TIME
  YEAR: 1977
  MONTH: 6
  DAY: 5
  HOUR: 13
  MINUTE: 34
```

8.64.4 Commands Related to IDT

IS	(Initialize System)
ISL	(Initialize System Log)

8.65 IF (INSERT FILE)

After you activate the Text Editor, the IF command inserts an entire file into the edited file after a specified record. Refer to Volume IV for a detailed description.

IGS

8.66 IGS (INSTALL GENERATED SYSTEM)

After a new system has been run and thoroughly tested by using the Test Generated System (TGS) command, you can use the IGS command to install it as the primary DX10 system. The current primary system is installed as the secondary system.

8.66.1 IGS Command Format

```
IGS
INSTALL GENERATED SYSTEM
TARGET DISK: [name]@
SYSTEM NAME: [name]@
```

8.66.2 IGS Command User Responses

System Prompts	Response Required or Optional	User Responses
TARGET DISK:	O	Enter the device name of the disk that contains the program file, .S$IMAGES, in which the new system is installed. *The default is DS01 or the target disk previously assigned by a PGS, IGS, TGS, or ALGS command.*
SYSTEM NAME:	O	Enter the name of the new system, specified as the GEN990 OUTPUT parameter. *The default is the system name previously assigned by a PGS, IGS, TGS or ALGS command.* If you do not specify a system name, a display of the current IPL status is given with no alternation made.

8.66.3 IGS Command Example

The following example shows how you can use the IGS command to install a system named ITGS. PAE5 is the disk name.

```
[] IGS
INSTALL GENERATED SYSTEM
 TARGET DISK: DS01
 SYSTEM NAME: ITGS

                                IGS/TGS
                        PAE5    SYSTEM IPL STATUS

    PRIMARY SYSTEM = ITGS                    SECONDARY SYSTEM = S$SUP
                          IPL ON PRIMARY SYSTEM
```

8.66.4 Commands Related to IGS

ALGS	(Assemble and Link Generated System)
PGS	(Patch Generated System)
TGS	(Test Generated System)

INV

8.67 INV (INITIALIZE NEW VOLUME)

The INV command prepares a disk volume for use under the operating system. Its primary functions are to assign a volume name, establish information used for allocating space to files, create the volume directory of the disk (.VCATALOG), establish a diagnostic file (volume.S$DIAG), and install the volume. Disk volume names must not be the same as disk device names (for example, DS01 or DS02).

Before you execute the INV command, you must execute the Initialize Disk Surface (IDS) command to check the disk for bad tracks. You only need to use the IDS command once. Once the IDS command has initialized the surface of a disk, you can use the INV command as often as desired. All bad track information is saved when the INV command is performed. If you then execute the Show Volume Status (SVS) command, the total number of available ADUs is different from the total ADUs. The bad track information and other system overhead take up space on track 0 and 1 that cannot be seen in the output of the Map Disk (MD) command. However, the total ADUs include those on track 0 and 1. These ADUs are not available since the system occupies them.

The FORCE CLEARING OF DISK? option allows you to ensure that any sensitive information on the disk is destroyed. Normally, INV deletes all files from the file structure, but the file contents still exist on the disk. Entering YES for this prompt significantly increases execution time.

NOTE

You can initiate the INV command with lowercase letters, but you must enter the alphabetic responses to the INV command prompts in uppercase letters. SCI does not perform lowercase to uppercase mapping for the prompt responses to this command, as explained in the .OPTION primitive in Volume III.

8.67.1 INV Command Format

```
INV
INITIALIZE NEW VOLUME
UNIT NAME: < name> @
VOLUME NAME: < name> @
NUMBER OF VCATALOG ENTRIES: [int]
BAD TRACK ACCESS NAME: [acnm]@
DEFAULT PHYSICAL RECORD SIZE: [int]
HARDWARE INTERLEAVING FACTOR: [int]
FORCE CLEARING OF DISK?: < yes/no>
USED AS SYSTEM DISK?: < yes/no>
LISTING ACCESS NAME: [acnm]@
EXECUTION MODE (F,B): < F/B>
```

8.67.2 INV Command User Responses

System Prompts	Response Required or Optional	User Responses
UNIT NAME:	R	Enter the device name of the disk unit on which the volume to be initialized is mounted.
VOLUME NAME:	R	Enter a one- to eight-character alphanumeric string (the first character of the string must be alphabetic) that is to identify the disk cartridge to DX10. Several disk cartridges may have the same volume name; however, only one at a time may be installed on the system.
NUMBER OF VCATALOG ENTRIES:	O	Enter a number representing the maximum number of entries (directories, subdirectories, and files) in the master volume directory (VCATALOG). The system rounds this number up to the next prime number. *The default value depends on the type of disk, as follows:*

Disk	Number of VCATALOG Entries
DS10	211
DS31	101
FD300	29
FD1000	53
DS25	347
DS50	347
DS200	347
CMD 16 M bytes	211
CMD 48 M bytes	347
CMD 80 M bytes	347
WD500 5 M bytes	211
WD800 18 M bytes	347
WD800 43 M bytes	347
DS80 80 M bytes	347
DS300	347

INV

System Prompts	Response Required or Optional	User Responses
BAD TRACK ACCESS NAME:	O	Enter the pathname of the file or the device name or synonym that is to supply bad track information. *The default is a null response, which indicates there is no bad track information.* If you enter ME, then SCI prompts for bad track information as follows: ENTER BAD TRACKS IN THE FORMAT: HEAD, CYLINDER; OR HEAD, CYLINDER; HEAD CYLINDER; etc. TO END LIST, ENTER AN EMPTY LINE. Enter bad track information in the requested format. All numbers entered are assumed to be decimal. For a T25, the system uses half of the cylinder that a T50 uses. The bad tracks labeled on the side of the disk pack are labeled as if the pack were formatted for a T50. When you enter those bad tracks for a T25, ignore all odd numbered cylinders. Divide each even numbered cylinder by two and enter the result as the cylinder number of the bad track.
DEFAULT PHYSICAL RECORD SIZE:	O	Enter the physical record size to be used in all subsequent file creations on the disk. The default value is defined during system generation and is usually 768. Refer to Volume III for more information on physical record size.

System Prompts	Response Required or Optional	User Responses
HARDWARE INTERLEAVING FACTOR:	O	Enter the interleaving factor used for double-density diskettes. The default value depends on the type of disk, as follows:

Disk	Interleaving Factor
DS10	1
DS31	1
FD300	3
FD1000	3
DS25	1
DS50	1
DS200	1
CMD 16 M bytes	1
CMD 48 M bytes	1
CMD 80 M bytes	1
WD500 5 M bytes	10
WD800 18 M bytes	1
WD800 43 M bytes	1
DS80 80 M bytes	1
DS300	1

FORCE CLEARING OF DISK?:	R	Enter YES if all previous data on the disk is to be erased. If you enter NO, the data is physically present but not accessible.
USED AS SYSTEM DISK?:	R	Enter YES if the disk is to be used as a system disk and thus needs a loader installed on it. If you specify YES, you must also specify YES for the FORCED CLEARING OF DISK prompt. Enter NO for disks that are not going to be system disks and need no loader. The default is YES.
LISTING ACCESS NAME:	O	Enter the access name of a file or device to which the returned information is to be written. *The default is the terminal local file (TLF).*

INV

System Prompts	Response Required or Optional	User Responses
LOADER ACCESS NAME	O	You can only specify this keyword value in batch mode or expert mode. You should use a pathname here only if you are initializing a disk with a nonstandard loader for your own purposes. If you specify this value, it is the access name of the file to be installed as the loader. If you do not enter a response, the standard system track 1 loader is installed on the disk.
EXECUTION MODE (F,B):	R	Enter F (foreground) to execute the task in foreground mode. Enter B (background) to execute the task in background mode. *The initial value is FOREGROUND.*

8.67.3 INV Command Example

The following example shows the initialization of a disk volume named VOL1 mounted on disk unit DS02. VOL1 can contain a maximum of 101 files (the next prime number larger than the specified size). The disk will be used as a system disk. The write protect must be off and the disk must not be installed.

```
[ ] INV
INITIALIZE NEW VOLUME
  UNIT NAME: DS02
  VOLUME NAME: VOL1
  NUMBER OF VCATALOG ENTRIES: 342 100
  BAD TRACK ACCESS NAME: .BADTRACK
  DEFAULT PHYSICAL RECORD SIZE: 768
  HARDWARE INTERLEAVING FACTOR: 1
  FORCED CLEARING OF DISK?: NO
  USED AS A SYSTEM DISK?: YES
  LISTING ACCESS NAME:
  EXECUTION MODE (F,B): FOREGROUND
```

8.67.4 Commands Related to INV

IDS	(Initialize Disk Surface)
IV	(Install Volume)
UV	(Unload Volume)

8.68 IO (INSTALL OVERLAY)

The IO command places an overlay associated with a task on the program file with the task. You must install the task before the overlay; you can specify the task by name or by installed ID.

NOTE

You should not install a task on the S$PROGA program file. If you use the S$PROGA program file, your IDs and names will be destroyed upon each new release of DX10. It is recommended that you install tasks in your own program file. This recommendation also applies to installing real-time tasks, procedures, and overlays.

8.68.1 IO Command Format

```
IO
INSTALL OVERLAY
PROGRAM FILE OR LUNO: < acnm> @ or < int> @
OVERLAY NAME: [string]@
OVERLAY ID: [int]@
OBJECT PATHNAME OR LUNO: < acnm> @ or < int> @
RELOCATABLE?: < yes/no>
DELETE PROTECT?: < yes/no>
ASSOCIATED TASK NAME OR ID: < string> @ or < int> @
```

8.68.2 IO Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter the file name of, or the LUNO assigned to, the program file on which you want to install the overlay. If you specify a LUNO in response to this prompt, it must be assigned to the program file prior to the execution of the IO command. If you specify zero, the program file selected at IPL is assumed (usually .S$PROGA).
OVERLAY NAME:	O	Enter a user-defined character string, a maximum of eight characters, that identifies the overlay and is unique to the program file. If you enter a null response, the system uses the IDT name of the object module as the name of the overlay.

IO

System Prompts	Response Required or Optional	User Responses
OVERLAY ID:	O	Enter an integer value in the range of 1 through 255 that is associated with the overlay name and is unique to the program file. If you specify zero or a null response, the system assigns an ID to the overlay.
OBJECT PATHNAME OR LUNO:	R	Enter the name of, or the LUNO assigned to, the device or file where the object module for the overlay resides. You must use a LUNO if the tasks, procedures, and overlays are in one file.
RELOCATABLE?:	R	If you enter YES, the overlay is allowed to be loaded at an address other than its natural load address.
DELETE PROTECT?:	R	If you enter YES, the overlay cannot be deleted from the program file unless the Modify Overlay Entry (MOE) command is used to unprotect the overlay prior to the execution of the Delete Overlay (DO) command. If you enter NO, the overlay can be deleted by executing the DO command.
ASSOCIATED TASK NAME OR ID:	R	Enter the name or ID of a previously installed task on the same program file as the overlay. The overlay is automatically deleted when the task is deleted unless the overlay is delete protected.

8.68.3 IO Command Example

The following example shows how you can enter the IO command to install a relocatable overlay named OLAY1 on the .USERPROG program file. LUNO > 0A has been assigned to the file where the object module resides, and ID > 83 identifies the associated task.

```
[ ] IO
INSTALL OVERLAY
  PROGRAM FILE OR LUNO: .USERPROG
  OVERLAY NAME: OLAY1
  OVERLAY ID: 0
  OBJECT PATHNAME OR LUNO: >0A
  RELOCATABLE?: YES
  DELETE PROTECT?: NO
  ASSOCIATED TASK NAME OR ID: >83
```

8.68.4 Commands Related to IO

DO	(Delete Overlay)
IP	(Install Procedure)
IRT	(Install Real-Time Task)
IT	(Install Task)
MOE	(Modify Overlay Entry)

IP

8.69 IP (INSTALL PROCEDURE)

The IP command places a procedure on a program file and assigns a procedure ID for use by subsequent IT commands.

NOTE

You should not install a task on the S$PROGA program file. If you use the S$PROGA program file, your IDs and names will be destroyed upon each new release of DX10. It is recommended that you install tasks in your own program file. This recommendation also applies to installing real-time tasks, procedures, and overlays.

8.69.1 IP Command Format

```
IP
INSTALL PROCEDURE
PROGRAM FILE OR LUNO: < acnm> @ or < int> @
PROCEDURE NAME: [string]@
PROCEDURE ID: [int]@
OBJECT PATHNAME OR LUNO: < acnm> @ or < int> @
MEMORY RESIDENT?: < yes/no>
DELETE PROTECT?: < yes/no>
EXECUTE PROTECT?: < yes/no>
WRITE PROTECT?: < yes/no>
WRITABLE CONTROL STORAGE?: < yes/no>
```

8.69.2 IP Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter the file name of, or the LUNO assigned to, the program file on which you want to install the procedure. If you specify a LUNO in response to this prompt, it must be assigned to the program file prior to execution of the IP command. If you specify zero, the program file selected at IPL is assumed (usually .S$PROGA).

System Prompts	Response Required or Optional	User Responses
PROCEDURE NAME:	O	Enter a user-defined character string, up to eight characters, that identifies the procedure and is unique to the program file. If you do not specify the procedure name, the system assigns the IDT name of the object module as the procedure name.
PROCEDURE ID:	O	Enter a hexadecimal integer to be assigned as the ID of the procedure and is unique to the program file. If you specify zero or a null response, the system assigns the ID.
OBJECT PATHNAME OR LUNO:	R	Enter the name of, or the LUNO assigned to, a device or file where the object module for the procedure resides. You must use a LUNO if the tasks, procedures, and overlays are in one file.
MEMORY RESIDENT?:	R	If you enter YES and the procedure is installed on the program file selected during initial program load (IPL) (usually .S$PROGA), the procedure is loaded into memory during IPL and stays in memory after execution. You should enter YES for this prompt only if you are very familiar with the system.
DELETE PROTECT?:	R	If you enter YES, the procedure cannot be deleted from the program file unless the Modify Procedure Entry (MPE) command is used to unprotect the procedure prior to the execution of the Delete Procedure (DP) command. If you enter NO, the procedure can be deleted by executing the DP command.
EXECUTE PROTECT?:	R	If you enter YES, the procedure cannot be executed. The protection is enforced only on a 990/12 computer.
WRITE PROTECT?:	R	If you enter YES, the procedure cannot be modified when in memory. The procedure is enforced only on a 990/12 computer.

IP

System Prompts	Response Required or Optional	User Responses
WRITABLE CONTROL STORAGE?:	R	If you enter YES, the procedure uses the writable control storage area. Writable control storage is available only on a 990/12 computer with supplied software.

8.69.3 IP Command Example

The following example shows how you can enter the IP command to install a procedure named MYPROC on your program file. LUNO > AA has been assigned to the program file, and LUNO > BB has been assigned to the file where the object module resides. The system assigns a procedure ID.

```
[ ] IP
INSTALL PROCEDURE
  PROGRAM FILE OR LUNO: > AA
  PROCEDURE NAME: MYPROC
  PROCEDURE ID: 0
  OBJECT PATHNAME OR LUNO: > BB
  MEMORY RESIDENT?: NO
  DELETE PROTECT?: NO
  EXECUTE PROTECT?: NO
  WRITE PROTECT?: NO
  WRITABLE CONTROL STORAGE?: NO
```

8.69.4 Commands Related to IP

DP	(Delete Procedure)
IO	(Install Overlay)
IRT	(Install Real-Time Task)
IT	(Install Task)
MPE	(Modify Procedure Entry)

8.70 IPF (INTERNATIONAL PRINT FILE)

The IPF command formats output files to match the requirements of the printwheel or printer type to be used and places the file in the output queue for the selected printer. This requires the printer to be internationalized to support the country's character set.

Default conversion tables can be selected to convert DX10 files for output to an LQ45 printer with either LQ45 FRANCE or LQ45 DEUTSCHLAND printwheels. No conversion is required for the other countries supported by DX10, provided the standard 911 international keyboard for that country is used with the appropriate national printwheel. You can create conversion files for other printwheels, printers, languages, or overstrike requirements.

You can create output formatting files to match raw data output to the code requirements of any type of printer or to specify overstrike characters. You must create these output formatting files using the following data format:

Record	File Column 1	Description
1	XX	Country code (two characters)
2	YY	Buffer coefficient × 10 (two characters)
3	Z1, Z2, . . . , Zn	List of characters affected by backspace. None if left blank.
4	VALUE 0	Code to be substituted for ASCII > 00
5	VALUE 1	Code to be substituted for ASCII > 01
.		
.		
.		
131		

8.70.1 IPF Command Format

```
IPF
INTERNATIONAL PRINT FILE
INPUT FILE PATHNAME: < acnm> @
OUTPUT FILE PATHNAME: < acnm> @
COUNTRY: < acnm> @
ANSI FORMAT?: [yes/no]
LISTING DEVICE: < acnm> @
DELETE AFTER PRINTING?: [yes/no]
DELETE OUTPUT AFTER PRINTING?: [yes/no]
NUMBER OF LINES/PAGE: [int]
```

IPF

8.70.2 IPF Command User Responses

System Prompts	Response Required or Optional	User Responses
INPUT FILE PATHNAME:	R	Enter a pathname that identifies the file to be printed. For TIPE-990 users, the TIPP command is used to format your file and then place the file in an output file. The pathname of this output file is specified as the INPUT FILE PATHNAME to the IPF command.
OUTPUT FILE PATHNAME:	R	Enter a device name of a device or the pathname of a file to which the information should be printed.
COUNTRY:	R	Enter the pathname of a formatting table used to process the output file into the required printing format. If you enter F, the formatting table for the LQ45 FRANCE printwheel is used. This causes the circumflex (^) and the umlaut (¨) to be typed above any character they precede in the output file. If you enter D, the formatting table for the LQ45 DEUTSCHLAND printwheel is used. This matches the German character set used in DX10 to that of the LQ45. (See Volume III for character sets.)
ANSI FORMAT?:	O	Enter Y (yes) to indicate that the file being printed contains ANSI carriage control characters. Enter N (no) to specify that the file does not contain ANSI carriage control characters. The ANSI control character (the first portion of each record) has the following meanings: blank — single space before printing 0 — double space before printing 1 — top of page before printing + — suppress line space (overprint)

System Prompts	Response Required or Optional	User Responses
		The initial value is YES. TIPE-990 users should accept the initial value.
LISTING DEVICE:	O	Enter the device name of the device to which the contents of the file(s) are printed.
DELETE AFTER PRINTING?:	O	Enter Y (yes) to indicate that the original output file is to be deleted after it is formatted. Enter N (no) to specify that the original output file should not be deleted. The initial value is NO.
DELETE OUTPUT AFTER PRINTING?:	O	Enter Y (yes) to indicate the file being printed is to be deleted after it is printed. Enter N (no) to specify that the file should not be deleted. The initial value is YES.
NUMBER OF LINES/PAGE?:	O	Enter the number of lines to be printed on one page. The default value is 62.

8.70.3 IPF Command Example

In the following example, the contents of the file VOL1.DIR2.FRENCH1 are printed on the LQ45 letter quality printer with 55 lines per page.

```
[ ] IPF
INTERNATIONAL PRINT FILE
  FILE PATHNAME: VOL1.DIR2.FRENCH1
  OUTPUT FILE PATHNAME: .TIPE03
  COUNTRY: F
  ANSI FORMAT?: YES
  LISTING DEVICE: ST04
  DELETE AFTER PRINTING?: NO
  DELETE OUTPUT AFTER PRINTING?: YES
  NUMBER OF LINES/PAGE: 55

  Test d'un texte français
  utilisant les mots clés
  suivants:

  Noël, élève, tête, français,
  gîte, hôtel, à, où, était,
  pâte, mais, flûte.

  La tabutation utilisée
  est de 20 à 50.
```

IPF

A Noël les élèves en tête
de classe de français
eurent un gîte fourni
dans un hôtel à Nice, où
le menu principal était à
base de pâtes et de maïs
avec heureusement une
flûte de Champagne.

Ce texte doit continuer
sur la page suivante.

8.70.4 IPF Error Messages

The following error messages are unique to the IPF command.

Error Code	Text	Meaning
01	IPF ERROR	Buffer coefficient is not numeric.
02	IPF ERROR	Buffer coefficient is less than 10.
03	COUNTRY FILE ERROR	Conversion table does not contain either 128 or 256 entries.
04	INPUT FILE ERROR	Input file is not sequential.
05	IPF ERROR	Unknown character found in input file.

8.70.5 Commands Related to IPF

PF (Print File)

8.71 IRT (INSTALL REAL-TIME TASK)

The Install Real-Time Task (IRT) command places an executable real-time task on a program file. If the task has attached procedures, you must install the procedures before executing the IRT command. For an explanation of the task attributes (priority, privileged, system, memory-resident, and replicative), consult Volume III.

NOTE

You should not install a task on the S$PROGA program file. If you use the S$PROGA program file, your IDs and names will be destroyed upon each new release of DX10. It is recommended that you install tasks in your own program file. This recommendation also applies to installing real-time tasks, procedures, and overlays.

8.71.1 IRT Command Format

```
IRT
INSTALL REAL-TIME TASK
PROGRAM FILE OR LUNO: < acnm> @ or < int> @
TASK NAME: [string]@
TASK ID: [int]@
OBJECT PATHNAME OR LUNO: < acnm> @ or < int> @
PRIORITY: < int> @
DEFAULT TASK FLAGS?: < yes/no>
ATTACHED PROCEDURES?: < yes/no>
PRIVILEGED?: < yes/no>
SYSTEM TASK?: < yes/no>
MEMORY RESIDENT?: < yes/no>
REPLICATABLE?: < yes/no>
DELETE PROTECTED?: < yes/no>
EXECUTE PROTECTED?: < yes/no>
OVERFLOW CHECKING?: < yes/no>
WRITABLE CONTROL STORAGE?: < yes/no>
1ST PROCEDURE ID: < int> @
P1 FROM TASKS PROGRAM FILE?: < yes/no>
2ND PROCEDURE ID: < int> @
P2 FROM TASKS PROGRAM FILE?: < yes/no>
```

These questions are asked only if the answer to DEFAULT TASK FLAGS?: is NO.

These questions are asked only if the answer to ATTACHED PROCEDURES is YES.

IRT

8.71.2 IRT Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter the file name of, or the LUNO assigned to, the program file on which you want to install the task. If you specify a LUNO in response to this prompt, it must be assigned to the program file prior to execution of the IRT command. If you specify zero, the program file selected at IPL is assumed (usually .S$PROGA).
TASK NAME:	O	Enter a user-defined character string, up to eight ASCII characters, that identifies the task and is unique to the program file. If you specify zero or a null response, the system assigns the IDT name of the object module as the name of the task.
TASK ID:	O	Enter a hexadecimal value to be associated with the installed task and is unique to the program file. If you specify zero or a null response, the system assigns an available ID.
OBJECT PATHNAME OR LUNO:	R	Enter the pathname of, or the LUNO assigned to, the device or file where the object module for the task resides. You must use a LUNO if the tasks, procedures, and overlays are in one file.
PRIORITY:	R	Enter the integer value that represents the execution priority level of the task. Priorities range from 1 through 127, with 1 being the highest priority.
DEFAULT TASK FLAGS?:	R	If you enter NO in response to this prompt, you have the option to set the task flags. If you enter YES, the initial values are used for the flags.

System Prompts	Response Required or Optional	User Responses
PRIVILEGED?:	R	If you enter YES, the task is allowed to execute privileged hardware instructions. Privileged hardware instructions should be executed cautiously and only by someone who is very familiar with the system.
SYSTEM TASK?:	R	If you enter YES, the task is allowed to execute in system memory space. Tasks should be executed in system memory space with caution and only by someone who is very familiar with the system.
MEMORY RESIDENT?:	R	If you enter YES and the task is installed on the program file selected during initial program load (IPL) (usually .S$PROGA), the task is loaded into memory during IPL and remains in memory after execution. You should enter YES for this prompt only if you are very familiar with the system.
REPLICATABLE?:	R	If you enter YES, multiple copies of the task can be in memory simultaneously.
DELETE PROTECTED?:	R	If you enter YES, the task cannot be deleted from the program file unless the Modify Task Entry (MTE) command is used to unprotect the task prior to the execution of the Delete Task (DT) command. If you enter NO, the task can be deleted by executing the DT command.
EXECUTE PROTECTED?:	R	If you enter YES, execution of the task is prohibited. The protection is enforced only on a 990/12 computer.
OVERFLOW CHECKING?:	R	If you enter YES, the occurrence of arithmetic overflow causes control of the task to pass to the end action routine of the task. Overflow checking is available only on a 990/12 computer.
WRITABLE CONTROL STORAGE?:	R	If you enter YES, the task uses the writable control storage area. Writable control storage is available only on a 990/12 computer with supplied software.

IRT

System Prompts	Response Required or Optional	User Responses
1ST PROCEDURE ID:	R	Enter the integer value representing the ID of a procedure attached to the task. If you specify zero, there are no procedures.
P1 FROM TASKS PROGRAM FILE?:	R	If you enter YES, the attached procedure whose ID was specified for the 1ST PROCEDURE ID prompt resides on the same program file as the task. If you enter NO, that procedure must reside on the program file selected at IPL (usually .S$PROGA).
2ND PROCEDURE ID:	R	Enter the integer value representing the ID of a procedure attached to the task. If you enter zero, there is no second procedure.
P2 FROM TASKS PROGRAM FILE?:	R	If you enter YES, the attached procedure whose ID was specified for the 2ND PROCEDURE ID prompt resides on the same program file as the task. If you enter NO, that procedure must reside on the program file selected at IPL (usually .S$PROGA).

8.71.3 IRT Command Example

The following example shows how you can enter the IRT command to install a task onto your program file, with one attached procedure on your program file. LUNO > 4A is assigned to the file where the object module resides. The default for task flags is taken.

```
[ ] IRT
INSTALL REAL-TIME TASK
  PROGRAM FILE OR LUNO: VOL1.MYFILE
  TASK NAME: MYFILE
  TASK ID: 0
  OBJECT PATHNAME OR LUNO: >4A
  PRIORITY: 3
  DEFAULT TASK FLAGS?: YES
  ATTACHED PROCEDURES?: YES
ATTACH TASK PROCEDURES
  1ST PROCEDURE ID: >42
  P1 FROM TASKS PROGRAM FILE?: NO
  2ND PROCEDURE ID: 0
  P2 FROM TASKS PROGRAM FILE?: NO
```

8.71.4 Commands Related to IRT

DT	(Delete Task)
IO	(Install Overlay)
IP	(Install Procedure)
IT	(Install Task)
MTE	(Modify Task Entry)

IS

8.72 IS (INITIALIZE SYSTEM)

After loading DX10 and activating SCI at a terminal, you can enter the IS command to ready DX10 for operation. The IS command initializes the system with information input by you and with information supplied to the system by the IS command itself.

The IS command assigns additional global LUNOs needed by the system to operate. If properly set up, the IS command also defines the characteristics of the terminals in the system. To set up the IS command so that it automatically assigns characteristics to terminals, you must add Modify Terminal Status (MTS) commands to the text of the IS command itself by following the procedure given in Volume V.

Besides the actions automatically taken by the IS command, the command also interacts with you to initialize the following:

- The system date and time
- The system log

8.72.1 IS Command Format

```
IS
INITIALIZE SYSTEM
INITIALIZE DATE AND TIME
YEAR: < int>
MONTH: < string>
DAY: < int>
HOUR: < int>
MINUTE: < int>
INITIALIZE SYSTEM LOG
ATTENTION DEVICE: < acnm> @
LOGGING DEVICE: < acnm> @
SYSTEM LOG PROCESSING TASK ID: < int>
ANALYSIS OUTPUT PRINTER: [acnm]
USER LOG PROCESSOR TASK ID: < int>
```

8.72.2 IS Command User Responses

System Prompts	Response Required or Optional	User Responses
YEAR:	R	Enter two to four numeric characters representing the current year. For example, 82 and 1982 both mean 1982.
MONTH:	R	Enter one or two numeric characters representing the current month or the first three alphabetic characters of the month.

System Prompts	Response Required or Optional	User Responses
DAY:	R	Enter one or two numeric characters representing the current day.
HOUR:	R	Enter one or two numeric characters representing the current hour (according to the 24-hour clock).
MINUTE:	R	Enter one or two numeric characters representing the current minute.
ATTENTION DEVICE:	R	Enter a device name to specify a device as the receiver of attention messages produced by the system during the logging process. Enter DUMY if you do not desire an attention device.
LOGGING DEVICE:	R	Enter a device name to specify a device to which the system log is written. Enter DUMY if you do not desire logging to a device.
SYSTEM LOG PROCESSOR TASK ID:	R	Enter the installed ID of a task on the system program file. The task is bid whenever a log file is filled. This parameter is to be used by add-on packages to DX10. The initial value is >5E, which is the task ID of the system log analysis task for online diagnostics. You can use this value if online diagnostics are installed or if no analysis task is used.
ANALYSIS OUTPUT PRINTER:	O	Enter the device name of a line printer used by the system log analyzer. Valid responses are LP01 through LP09. If left blank, the system log analyzer report file is used. LP$x is not a valid response.
USER LOG PROCESSOR TASK ID:	R	Respond to this prompt with the task ID of an installed task on the system program file. The task will be bid whenever a log file is filled. This parameter is to be used with a user-supplied task. An ID of 0 indicates that no task is to be bid.

IS

NOTE

When the logging or attention device is also used for text editing, extraneous data can be written in an edited file. Do not specify a terminal that is to be used for text editing as the logging or attention device.

8.72.3 IS Command Example

The following example shows how you can enter the IS command to initialize a system that employs logging files.

```
WARNING: SYSTEM IS NOT INITIALIZED
[ ] IS
INITIALIZE SYSTEM
INITIALIZE DATE AND TIME
  YEAR: 1977
  MONTH: 8
  DAY: 10
  HOUR: 17
  MINUTE: 35
INITIALIZE SYSTEM LOG
  ATTENTION DEVICE: ST02
  LOGGING DEVICE: ST02
  SYSTEM LOG PROCESSOR TASK ID: 05E
  ANALYSIS OUTPUT PRINTER: LP01
  USER LOG PROCESSOR TASK ID: 0
WARMSTART PROCEDURE COMPLETE:
```

8.72.4 Commands Related to IS

IDT	(Initialize Date and Time)
ISL	(Initialize System Log)

8.73 ISL (INITIALIZE SYSTEM LOG)

The DX10 operating system can automatically record the following information about system operation in a system log:

- Messages generated by user programs
- Device statistics
- Device hardware errors
- Task abnormal termination errors
- Memory errors (16K RAM only)

System log information can be recorded on system log files and at a physical device such as a line printer or data terminal. You initialize the system log with the Initialize System (IS) command after DX10 is first loaded into memory. During operation of DX10, you may want to reinitialize the log to change logging devices or to reinitialize the log after a logging error. To reinitialize the system log, enter the Initialize System Log (ISL) command.

When you execute the ISL command, you specify a system log processor task and a user log processor task. Both system and user log processors must be installed on the system program file. These tasks are bid whenever a log file is filled. When the system bids either the system or user log processors, the number of the log file that has just been filled (ASCII 1 or ASCII 2) is passed to the task. This number is accessible via a Get Parameters SVC (> 17), which is in the upper byte of the second parameter word.

When you initialize or reinitialize the system log, the system begins recording the log information on whichever file was last updated. Log messages will be logged in that file until it is full. At that time the following messages appear at the device specified as the attention device by the ISL command:

```
***** LOG FILE .S$SLG1 FULL
***** SYSTEM LOG FILE FULL: pathname
```

The first message may indicate that log file .S$SLG2 is full rather than .S$SLG1. The system log full message tells you that you should copy the contents of the log file (identified by pathname) to another file or device because it will be overwritten if the other system log file fills up. Optionally, you can specify a task to be bid whenever a log file is full.

You can keep the log on log files and on a logging device. The advantage of this approach is that an error at the logging device does not disable logging to the system log files. An error in the log file, however, stops the logging of messages to both the log files and the logging device.

ISL

8.73.1 ISL Command Format

```
ISL
INITIALIZE SYSTEM LOG
ATTENTION DEVICE: < acnm>  @
LOGGING DEVICE: < acnm>  @
SYSTEM LOG PROCESSOR TASK ID: < int>
ANALYSIS OUTPUT PRINTER: [acnm]
USER LOG PROCESSOR TASK ID: < int>
```

8.73.2 ISL Command User Responses

System Prompts	Response Required or Optional	User Responses
ATTENTION DEVICE:	R	Enter a device name to specify a device as the receiver of attention messages produced by the system during the logging process. Enter DUMY if you do not desire an attention device.
LOGGING DEVICE:	R	Enter a device name to specify a device to which the system log is written. Enter DUMY if you do not desire logging to a device.
SYSTEM LOG PROCESSOR TASK ID:	R	Enter the installed ID of a task on the system program file. The task is bid whenever a log file is filled. This parameter is to be used by add-on packages to DX10. The initial value is >5E, which is the task ID of the system log analysis task for online diagnostics. You can use this value if online diagnostics are installed or if no analysis task is used.
ANALYSIS OUTPUT PRINTER:	O	Enter the device name of a line printer used by the system log analyzer. Valid responses are LP01 through LP09. If left blank, the system log analyzer report file is used. LP$x is not a valid response.
USER LOG PROCESSOR TASK ID:	R	Respond to this prompt with the task ID of an installed task on the system program file. The task will be bid whenever a log file is filled. This parameter is to be used with a user-supplied task. An ID of 0 indicates that no task is to be bid.

NOTE

When the logging or attention device is also used for text editing, extraneous data can be written in an edited file. Do not specify a terminal that is to be used for text editing as the logging or attention device.

8.73.3 ISL Command Example

The following example shows how you can enter an ISL command to start a system log at both a data terminal and on the system log files.

```
[ ] ISL
INITIALIZE SYSTEM LOG
  ATTENTION DEVICE: ME
  LOGGING DEVICE: ST01
  SYSTEM LOG PROCESSOR TASK ID: 05E
  ANALYSIS OUTPUT PRINTER: LP01
  USER LOG PROCESSOR TASK ID: 0
```

The system signals successful start-up of the system log by displaying (or printing) the following message at the logging device:

```
***** LOG STARTED ****************************************
```

Error messages may precede this message if the system encountered errors before the ISL command was successfully processed. Refer to Volume VI for a description of ISL command error messages.

8.73.4 Commands Related to ISL

IDT	(Initialize Date and Time)
IS	(Initialize System)

8.74 ISO (INSTALL SYSTEM OVERLAY)

The ISO command allows you to add a system object module as an overlay on the system overlay file (.S$OVLYA) of an active system. You can install a system overlay on the system overlay file without going through the disk build process. You can find additional discussion describing specific errors and user action in Volume VI.

NOTE

You use the ISO command when maintaining or updating DX10.

CAUTION

Use of this command without specific instructions from the software supplier may damage the integrity of the DX10 system.

8.74.1 ISO Command Format

```
ISO
INSTALL SYSTEM OVERLAY
OVERLAY FILE PATHNAME: < acnm>
OBJECT PATHNAME: < acnm>
OVERLAY NUMBER: < integer>
```

8.74.2 ISO Command User Responses

System Prompts	Response Required or Optional	User Responses
OVERLAY FILE PATHNAME:	R	Enter the pathname that identifies the system overlay to be installed.
OBJECT PATHNAME:	R	Enter the pathname that identifies the object to be added as an overlay.
OVERLAY NUMBER:	R	Enter the integer number of the overlay to be added to the system overlay file.

8.74.3 ISO Command Example

The following example shows how you can use the ISO command to add an overlay to the system overlay file (.S$OVLYA) of an active system:

```
[ ] ISO
INSTALL SYSTEM OVERLAY
  OVERLAY FILE PATHNAME: .S$OVLYA
  OBJECT PATHNAME: .MYOVLY
  OVERLAY NUMBER: > 1B
```

8.75 IT (INSTALL TASK)

The IT command places an executable task on a program file. If the task has attached procedures, you must install the procedures before executing the IT command. For an explanation of the task attributes (priority, privileged, system, memory-resident, and replicative), refer to Volume III.

NOTE

You should not install a task on the S$PROGA program file. If you use the S$PROGA program file, your IDs and names will be destroyed upon each new release of DX10. It is recommended that you install tasks in your own program file. This recommendation also applies to installing real-time tasks, procedures, and overlays.

8.75.1 IT Command Format

```
IT
INSTALL TASK
PROGRAM FILE OR LUNO: < acnm> @ or < int> @
TASK NAME: [string]@
TASK ID: [int]@
OBJECT PATHNAME OR LUNO: < acnm> @ or < int> @
PRIORITY: < int> @
DEFAULT TASK FLAGS?: < yes/no>
ATTACHED PROCEDURES?: < yes/no>
PRIVILEGED?: < yes/no>
SYSTEM TASK?: < yes/no>
MEMORY RESIDENT?: < yes/no>
REPLICATABLE?: < yes/no>
DELETE PROTECTED?: < yes/no>
EXECUTE PROTECTED?: < yes/no>
OVERFLOW CHECKING?: < yes/no>
WRITABLE CONTROL STORAGE?: < yes/no>
1ST PROCEDURE ID: < int> @
P1 FROM TASKS PROGRAM FILE?: < yes/no>
2ND PROCEDURE ID: < int> @
P2 FROM TASKS PROGRAM FILE?: < yes/no>
```

These prompts appear only if you answer NO to DEFAULT TASK FLAGS?.

These prompts appear only if you answer YES to ATTACHED PROCEDURES?.

IT

8.75.2 IT Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter the file name of, or the LUNO assigned to, the program file on which you want to install the task. If you specify a LUNO in response to this prompt, it must be assigned to the program file prior to the execution of the IT command. If you specify zero, the program file selected at IPL is assumed (usually .S$PROGA).
TASK NAME:	O	Enter a user-defined character string, up to eight ASCII characters long, that identifies the task and is unique to the program file. If you specify zero or a null response, the system assigns the IDT name of the object module as the task name.
TASK ID:	O	Enter a hexadecimal value to be associated with the installed task and is unique to the program file. If you specify zero or a null response, the system assigns an available ID.
OBJECT PATHNAME OR LUNO:	R	Enter the pathname of, or the LUNO assigned to, a device or file in which the object module of the task resides. You must use a LUNO if the tasks, procedures, and overlays are in one file.
PRIORITY:	R	Enter the integer value that represents the execution priority level of the task. Priorities can range from zero through four. Priorities zero through three are fixed, with priority zero as the highest level and three the lowest. Priority four is dynamically managed by the operating system. Four is the default priority level.
DEFAULT TASK FLAGS?:	R	If you enter NO in response to this prompt, you have the option to set the task flags. If you enter YES, the initial values are used for the flags.

System Prompts	Response Required or Optional	User Responses
ATTACHED PROCEDURES?:	R	If you enter YES in response to this prompt, you are prompted for the ID(s) of procedures attached to this task and asked if the procedures reside on the same program file as the task.
PRIVILEGED?:	R	If you enter YES, the task is allowed to execute privileged hardware instructions.
SYSTEM TASK?:	R	If you enter YES, the task is allowed to execute in system memory space.
MEMORY RESIDENT?:	R	If you enter YES and the task is installed on the program file selected during initial program load (IPL) (usually .S$PROGA), the task is loaded into memory during IPL and remains in memory after execution. You should enter YES for this prompt only if you are very familiar with the system.
REPLICATABLE?:	R	If you enter YES, multiple copies of the task can be in memory simultaneously.
DELETE PROTECTED?:	R	If you enter YES, the task cannot be deleted from the program file unless the Modify Task Entry (MTE) command is used to unprotect the task prior to the execution of the Delete Task (DT) command. If you enter NO, the task can be deleted by the DT command.
EXECUTE PROTECTED?:	R	If you enter YES, execution of the task is prohibited. The protection is enforced only on a 990/12 computer.
OVERFLOW CHECKING?:	R	If you enter YES, the occurrence of arithmetic overflow causes control of the task to pass to the end action routine of the task. Overflow checking is enforced only on a 990/12 computer.
WRITABLE CONTROL STORAGE?:	R	If you enter YES, the task uses the writable control storage area. Writable control storage is available only on a 990/12 computer with supplied software.

IT

System Prompts	Response Required or Optional	User Responses
1ST PROCEDURE ID:	R	Enter the integer value representing the ID of a procedure attached to the task. If you enter zero, there are no procedures.
P1 FROM TASKS PROGRAM FILE?:	R	If you enter YES, the attached procedure whose ID was specified for the 1ST PROCEDURE ID prompt resides on the same program file as the task. If you enter NO, that procedure must reside on the program file selected at IPL (usually .S$PROGA).
2ND PROCEDURE ID:	R	Enter the integer value representing the ID of a procedure attached to the task. If you enter zero, there is no second procedure.
P2 FROM TASKS PROGRAM FILE?:	R	If you enter YES, the attached procedure whose ID was specified for the 2ND PROCEDURE ID prompt resides on the same program file as the task. If you enter NO, that procedure must reside on the program file selected at IPL (usually .S$PROGA).

8.75.3 IT Command Example

The following example shows how you can enter the IT command to install a task, with one attached procedure on the .USERPROG program file. The default to the task flags prompt is taken.

```
[ ] IT
INSTALL TASK
  PROGRAM FILE OR LUNO: .USERPROG
  TASK NAME: MYFILE
  TASK ID: 0
  OBJECT PATHNAME OR LUNO: >4A
  PRIORITY: 3
  DEFAULT TASK FLAGS?: YES
  ATTACHED PROCEDURES?: YES
ATTACH TASK PROCEDURES
  1ST PROCEDURE ID: >42
  P1 FROM TASKS PROGRAM FILE?: NO
  2ND PROCEDURE ID: 0
  P2 FROM TASKS PROGRAM FILE?: NO
```

8.75.4 Commands Related to IT

DT	(Delete Task)
IO	(Install Overlay)
IP	(Install Procedure)
IRT	(Install Real-Time Task)
MTE	(Modify Task Entry)

IV

8.76 IV (INSTALL VOLUME)

To install a disk volume for operation, you must mount an initialized disk volume in a disk unit, ready the unit, and enter the IV command. DX10 cannot install a volume when the volume name specified with the IV command does not match the volume name assigned to the volume being installed.

The DS10, CD1400/32, and CD1400/96 disk drives are physically organized as a removable platter and a nonremovable platter turned by a single disk spindle. When the removable platter is being loaded or unloaded, no accesses may be made to the nonremovable platter. The fixed platter cannot be used if there is no removable platter in the drive.

Further, no volume being installed may have the same volume name as one already installed (including the system disk).

8.76.1 IV Command Format

```
IV
INSTALL VOLUME
UNIT NAME: < name> @
VOLUME NAME: < name> @
```

8.76.2 IV Command User Responses

System Prompts	Response Required or Optional	User Responses
UNIT NAME:	R	Enter the device name of the disk unit on which the volume is being installed. The volume must be mounted on this device.
VOLUME NAME:	R	Enter the one- to eight-character alphanumeric string naming the volume being installed.

8.76.3 IV Command Example

The following example shows the installation of a disk volume named VOL1 on disk unit DS02:

```
[ ] IV
INSTALL VOLUME
  UNIT NAME: DS02
  VOLUME NAME: VOL1
```

8.76.4 Commands Related to IV

CRV	(Check and Reset Volume)
INV	(Initialize New Volume)
UV	(Unload Volume)

8.77 KBT (KILL BACKGROUND TASK)

The KBT command kills all background activity currently executing at the initiating terminal. If there is no background activity at that terminal, the KBT command returns the following message:

NO BACKGROUND ACTIVITY AT THIS TERMINAL

8.77.1 KBT Command Format

KBT
KILL BACKGROUND TASK

8.77.2 KBT Command User Responses

No user responses are required.

8.77.3 KBT Command Example

The following example shows entry of the KBT command to terminate a background task and the message that the system produces:

```
[ ] KBT

KILL BACKGROUND TASK

EXECUTION TERMINATED
[ ]
```

The background task is still active until another message appears indicating the state of completion.

8.77.4 Commands Related to KBT

KT	(Kill Task)
SBS	(Show Background Status)
WAIT	(Wait for Background Task to Complete)

KO

8.78 KO (KILL OUTPUT AT DEVICE)

The KO command terminates either the first file in the output queue (that is, the currently printing output) or all queued output including the currently printing output to a given device. If you kill only the currently printing file, the next item in the queue starts printing. If you kill all queued output, you must then execute an RO command to reactivate the output device. You can execute an SOS command to see which file is active (currently printing) and which files are waiting. If no file is active, there is no currently printing file.

8.78.1 KO Command Format

```
KO
DEVICE NAME: < acnm>  @
KILL CURRENT FILE ONLY?: [yes/no]
```

8.78.2 KO Command User Responses

System Prompts	Response Required or Optional	User Responses
DEVICE NAME:	R	Enter the device name of the device at which you want to terminate queued output.
KILL CURRENT FILE ONLY?:	O	Enter Y (yes) to terminate the output of the currently printing file. Enter N (no) to terminate the output of all files in the output queue including the currently printing file. *The default is to terminate the currently printing file.* If you specify N, you must then execute an RO command to reactivate the output devices.

8.78.3 KO Command Example

The following example shows how to enter the KO command to terminate output at a line printer:

```
[ ] KO
KILL OUTPUT AT DEVICE
  DEVICE NAME: LP01
  KILL CURRENT FILE ONLY?: YES
```

8.78.4 Commands Related to KO

HO	(Halt Output at Device)
RO	(Resume Output at Device)
SOS	(Show Output Status)

8.79 KT (KILL TASK)

The KT command terminates an active task. You can issue the KT command from any terminal connected to DX10.

Tasks that perform an end action are allowed to perform the end action before terminating. DX10 displays or prints at your terminal the state of the task at the time you executed the KT command. Tasks assume different states as they execute or await execution. Each state is listed in the Show Task Status (STS) command description.

Tasks in the following states will proceed to end action or terminate immediately after a kill request is made on them:

00	Task awaiting time slice allotment
01	Task awaiting time slice allotment
04	Task is terminated
05	Task in time delay
06	Task unconditionally suspended
08	Task awaiting character input
09	Task awaiting I/O completion
0A	Task awaiting assignment of a device for I/O
0D	Task awaiting file management services
0F	Task awaiting initial bid
12	Task awaiting tape management services
17	Task awaiting coroutine activation
19	Task awaiting completion of any I/O
24	Task terminated awaiting service queue input

The KT command does not terminate tasks in any other state until their state converts to one of those listed.

8.79.1 KT Command Format

```
KT
KILL TASK
RUN ID: <int>
STATION NUMBER: <int>
```

KT

8.79.2 KT Command User Responses

System Prompts	Response Required or Optional	User Responses
RUN ID:	R	Enter the run-time ID assigned to the task by DX10. You can determine this ID by using the STS command.
STATION NUMBER:	R	Enter the station number of the station with which the task is associated. Enter >FF if the task is not associated with any station. Enter zero to specify your own station.

8.79.3 KT Command Example

The following example shows entry of a KT command to terminate a task with a run-time ID of >5C that is associated with station number 4 (the station with device name ST04). DX10 responds by terminating the task and printing a message indicating that the task was awaiting the completion of Input/Output (task state >09) at the time it was terminated.

```
[ ] KT
KILL TASK
  RUN ID: >5C
  STATION NUMBER: 4
TASK STATE = >09
```

8.79.4 Commands Related to KT

KBT (Kill Background Task)
STS (Show Task Status)

8.80 LB (LIST BREAKPOINTS)

The LB command displays the breakpoints for a specified task. Refer to Volume III for a detailed description.

LC

8.81 LC (LIST COMMANDS)

The LC command displays the main SCI menu.

8.81.1 LC Command Format

LC

8.81.2 LC Command User Responses

No user responses are required.

8.81.3 LC Command Example

[ ] LC

8.82 LD (LIST DIRECTORY)

The LD command lists the names of all files and subdirectories in a directory. The names of files within subdirectories are not listed. The LD command cannot process directories with 3,000 or more entries. If a directory contains 3,000 or more entries, the LD command returns the following error message:

CANNOT GET ENOUGH MEMORY.

8.82.1 LD Command Format

LD
LIST DIRECTORY
PATHNAME: < acnm> @
LISTING ACCESS NAME: [acnm]@

8.82.2 LD Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname that identifies the directory whose file names and subdirectory names you want to list.
LISTING ACCESS NAME:	O	Enter the device name of the device or the pathname of the file to which DX10 should write the list. *The default value is the terminal local file (TLF).*

8.82.3 LD Command Example

Figure 8-1 is an example of the information produced by an LD command. The item numbers in Table 8-3 refer to the numbered items in Figure 8-1.

LD

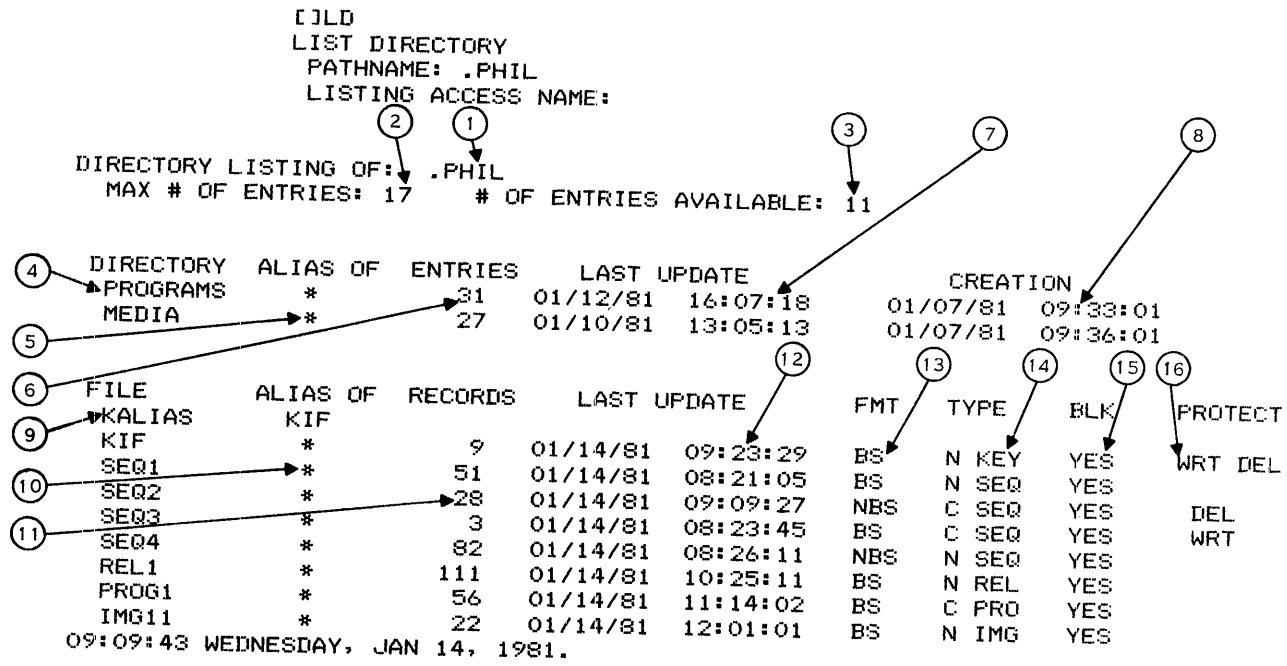


Figure 8-1. Output the LD Command Produces

Table 8-3. Output of LD Command

	Field	Description
1.	DIRECTORY LISTING OF:	The name of the directory being listed. A period (.) preceding a directory name indicates that the directory is on the system disk. The system assumes the volume name is the name of the system disk.
2.	MAX # OF ENTRIES:	The maximum number of subdirectories or files that can be placed in this directory.
3.	# OF ENTRIES AVAILABLE:	The maximum number of subdirectories or files currently available for placement in this directory.
4.	DIRECTORY	The name of each subdirectory in the directory being listed.

Table 8-3. Output of LD Command (Continued)

	Field	Description
5.	ALIAS OF	An alternate name that you can use instead of the directory name to access the directory.
6.	ENTRIES	The maximum number of subdirectories and files that can exist in the directory. This number is always a prime number.
7.	LAST UPDATE	The last date and time a file in this directory was updated. (That is, the last time the file was replaced by a Quit Edit (QE), Copy/Concatenate (CC), or similar command.)
8.	CREATION	The date and time the directory was created.
9.	FILE	The name of each file in the directory being listed.
10.	ALIAS OF	An alternate name that can be used instead of the file name to access the file.
11.	RECORDS	The number of logical records in the specified file. This is an exact count. For relative record files, this is the largest record plus one since the first record of relative record files is numbered zero. For sequential files, any EOF markers are included in the count.
12.	LAST UPDATE	The last date and time the specified file was updated.
13.	FMT	The format of the data stored on the file. BS indicates blank-suppressed format; NBS indicates no blank-suppressed format.
14.	TYPE	The first character is either N or C. N indicates noncontiguous (that is, the file is expandable); C indicates contiguous (the file is not expandable). The next three characters define the file type of the files as follows:

File Type	Meaning
SEQ	Sequential file
KEY	Key indexed file
REL	Relative record file
IMG	Image file (relative record file, special usage)
PRO	Program file

LD

Table 8-3. Output of LD Command (Continued)

	Field	Description
15.	BLK	Indicates whether or not the file is blocked. YES indicates blocked; NO indicates not blocked.
16.	PROTECT	Indicates the file protection, as follows:

Protection Code	Meaning
WRT	Write protected (which implies delete protected)
DEL	Delete protected
WRT DEL	Write and delete protected

8.82.4 Commands Related to LD

CD (Copy Directory)
CFDIR (Create Directory File)
DD (Delete Directory)

8.83 LDC (LIST DEVICE CONFIGURATION)

The LDC command lists the devices specified during system generation and the state of those devices.

8.83.1 LDC Command Format

```
LDC
LIST DEVICE CONFIGURATION
LISTING ACCESS NAME: [acnm]@
```

8.83.2 LDC Command User Response

System Prompt	Response Required or Optional	User Response
LISTING ACCESS NAME:	O	Enter the access name of a file or device to which the information is to be returned. *The default value is the terminal local file (TLF).*

8.83.3 LDC Command Example

In the following example, the LDC command returns the LDC information to the TLF. This is the software configuration of the executing system. The three column headings shortened due to spacing requirements have the following meanings: INT = interrupt, CHAS = chassis, POS = position. The STATE of the device is either offline (OFF), online (ON), or diagnostic (DIAG).

The column labeled CODE8 designates if the device is capable of handling 8-bit ASCII character codes (Y = YES, N = NO).

```
[] LDC
LIST DEVICE CONFIGURATION
 LISTING ACCESS NAME:     ME
DEVICE      TYPE       ADDRESS     INT    CHAS    POS    STATE    CODE8
 CR01     CARD READE     0040       4                     ON        Y
 CS01     CASSETTE       0000       6                     OFF       N
 CS02     CASSETTE       0000       6                     OFF       N
 DS01     DISK           F800      13                     ON        N
 DS02     DISK           F800      13                     ON        N
 DS03     DISK           F820       7      1      13      ON        N
 DS04     DISK           F820       7      1      13      ON        N
 LP01     PRINTER        0460       7      1      14      ON        Y
 LP02     PRINTER        0420       7      1      15      OFF       N
 MT01     MAG TAPE       F880       9                     OFF       N
 ST01     VDT            0100      10                     ON        N
 ST02     VDT            0120       8                     ON        N
 ST03     VDT            00C0      11                     ON        N
 ST04     KEYBOARD       0000       6                     ON        N
 ST05     KEYBOARD       0020      15                     ON        Y
 ST06     KEYBOARD       0060      14                     ON        N
```

8.83.4 Commands Related to LDC

MDS (Modify Device State)

LHPC

8.84 LHPC (LIST HARD-COPY TERMINAL PORT CHARACTERISTICS)

The LHPC command lists the communications port characteristics.

8.84.1 LHPC Command Format

```
LHPC
LIST HARDCOPY TERMINAL PORT CHARACTERISTICS
OUTPUT ACCESS NAME: [acnm]
```

8.84.2 LHPC Command User Responses

System Prompt	Response Required or Optional	User Response
OUTPUT ACCESS NAME:	O	This parameter indicates where the information is to be listed. If you do not enter a response, the information is listed at the terminal where you made the request. Only parameters that can be modified using the MHPC command are listed.

8.84.3 LHPC Command Example

The following example shows how you can use the LHPC command to list the communications port characteristics to LP01:

```
[ ] LHPC
LIST HARDCOPY TERMINAL PORT CHARACTERISTICS
 OUTPUT ACCESS NAME: LP01

    HARDCOPY TERMINAL PORT CHARACTERISTICS

TERM   IN     READ TIMEOUT      WRITE   LINE   TERM   DS    DE   USE  LTA
NAME   SERV   ASCII   DIRECT    TIMEOUT SPEED  TYPE   CODE  CODE LTA  CHAR
ST17   Y        0       0         0     300    765    2     1
ST18   Y        0       0         0     300    765    2     1
```

The following list describes the meanings of the column headings. Refer to the MHPC command for a detailed description of how to modify these characteristics.

Column Heading	Description
TERM NAME	Terminal access name
IN SERV	In service
READ TIMEOUT — ASCII	Read ASCII timeout
READ TIMEOUT — DIRECT	Read direct timeout
WRITE TIMEOUT	Write timeout
LINE SPEED	Line speed
TERM TYPE	Terminal type
DS CODE	Disconnect sequence code
DE CODE	Duplex/echo code
USE LTA	Use line turnaround
LTA CHAR	Line turnaround character

8.84.4 Commands Related to LHPC

ANS	(Answer Incoming Call)
CALL	(Call Terminal)
DISC	(Terminal Disconnection)
MHPC	(Modify Hard-Copy Terminal Port Characteristics)

8.85 LLR (LIST LOGICAL RECORD)

The LLR command lists the contents of a record or records in a file. The contents of the record or records specified are listed in both hexadecimal representation and ASCII representation.

8.85.1 LLR Command Format

```
LLR
LIST LOGICAL RECORD
PATHNAME: < acnm>  @
STARTING RECORD: < int>
NUMBER OF RECORDS: [int]
LISTING ACCESS NAME: [acnm] @
```

8.85.2 LLR Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter a pathname that identifies the file in which the records to be listed reside.
STARTING RECORD:	R	Enter a decimal or hexadecimal integer that identifies the first record whose contents are to be listed. The first record of any file is zero. If you want to start listing somewhere else in the file, you should put the number of records to skip over. *The initial value is 0 (zero).*
NUMBER OF RECORDS:	O	Enter a decimal or hexadecimal integer that identifies how many records are to be listed. *The default is all records.*
LISTING ACCESS NAME:	O	Enter the device name of a device or the pathname of a file to which the LLR command should list the information that it produces. *The default value is the terminal local file (TLF).*

8.85.3 LLR Command Example

The following example shows how you can enter an LLR command to list the contents of the first three records of a file at your terminal. The word SAME in the example indicates that the last word printed on the preceding line is the same as all following words until the end of record is reached or until a different word is encountered. The last word of each record is always printed.

```
[] LLR
LIST LOGICAL RECORD
 PATHNAME: .LLREG
 STARTING RECORD:  0              HEXADECIMAL       ASCII REPRESENTATION
 NUMBERS OF RECORDS:  3           REPRESENTATION    OF CONTENTS
 LISTING ACCESS NAME:             OF CONTENTS
FILE ACCESS NAME:  .LLREG
RECORD:  000000
0000  5245 434F 5244 204E 554D 4245 5220 4F4E     RE CO RD  N UM BE R  ON
0010  4520 2020 2020 2020 2020 2020 2020 2020     E
  SAME
004E  2020
RECORD:  000001
0000  5245 434F 5244 204E 554D 4245 5220 5457     RE CO RD  N UM BE R  TW
0010  4F20 2020 2020 2020 2020 2020 2020 2020     O
  SAME
004E  2020
RECORD:  000002
0000  5245 434F 5244 204E 554D 4245 5220 5448     RE CO RD  N UM BE R  TH
0010  5245 4520 2020 2020 2020 2020 2020 2020     RE E
  SAME
004E  2020
```

8.85.4 Commands Related to LLR

SRF (Show Relative to File)

LM

8.86 LM (LIST MEMORY)

The LM command lists the specified memory area of a task to a specified output device or file. Refer to Volume III for a detailed description.

8.87 LS (LIST SYNONYMS)

The LS command displays, prints, or writes to a file, a list of all synonyms and synonym values that are assigned under your user ID (if log-on is required) or assigned at your terminal (if log-on is not required).

NOTE

If the synonym table for a user ID is full, the LS command does not work. You can use the Modify Synonym (MS) command to delete or display synonyms.

8.87.1 LS Command Format

```
LS
LIST SYNONYMS
LISTING ACCESS NAME: [acnm] @
```

8.87.2 LS Command User Responses

System Prompt	Response Required or Optional	User Response
LISTING ACCESS NAME:	O	Enter the name of a device or the pathname of a file to which the system displays or lists the synonyms and synonym values assigned under a user ID or terminal. *The default value is the terminal local file (TLF).*

8.87.3 LS Command Example

The following example shows how you can enter the LS command to display a list of synonyms and synonym values at a terminal with the device name ST09:

```
[ ] LS
LIST SYNONYMS
  LISTING ACCESS NAME: ST09
```

8.87.4 Commands Related to LS

AS	(Assign Synonym)
MS	(Modify Synonym)

LSB

8.88 LSB (LIST SIMULATED BREAKPOINTS)

The LSB command displays all current simulated breakpoints. Refer to Volume III for a detailed description.

8.89 LSM (LIST SYSTEM MEMORY)

The LSM command lists the memory occupied by the operating system; similar to the List Memory (LM) command, except that you specify an overlay name or ID instead of a run ID. The LSM command is intended for use only by someone very familiar with the operating system source. Refer to Volume III for a detailed description.

LTS

8.90 LTS (LIST TERMINAL STATUS)

The LTS command displays or lists the attributes assigned to any one terminal or the attributes of all terminals in the system and shows the user ID of users currently using the terminal. Figure 8-2 shows an example of a list of the attributes of all terminals in a system.

Table 8-4 describes the status information given under each heading of the output of the LTS command.

```
TERMINAL   USER    LI REQ'D   MODE   DEFAULT
  ST01     SYSTEM     NO       TTY     VDT
  ST02                YES      VDT     VDT
  ST03     WHT016     YES      VDT     VDT
  ST04     CLN002     YES      VDT     VDT
  ST05                YES      VDT     VDT
  ST06     GPS021     YES      VDT     VDT
  ST07     JKB044     YES      TTY     VDT
  ST08     KAC009     YES      VDT     VDT
  ST09                YES      VDT     VDT
  ST10                YES      VDT     VDT
  ST11                NO       TTY     VDT
  ST12                NO       TTY     VDT
  ST13                NO       TTY     VDT
  ST14                NO       TTY     VDT
  ST15                NO       TTY     VDT
```

Figure 8-2. Output the LTS Command Produces

Table 8-4. Status Information the LTS Command Produces

Heading	Description
TERMINAL	The device name of the terminal or terminals for which statuses are listed. A maximum of 39 terminals can be listed; only those terminals whose status is ON are listed (see the MTS command).
USER	The ID of the user currently logged on at the terminal.
LI REQ'D	Y (yes) indicates that users must log on at the terminal to use SCI. N (no) indicates that SCI can be used at the terminal without the user having to log on.
MODE	TTY indicates that the terminal is currently set for use in TTY mode. VDT indicates that the terminal is currently set for use in VDT mode.
DEFAULT	TTY indicates that the terminal is always used in TTY mode unless its status is changed by an MTS command. VDT indicates that the terminal is always used in VDT mode unless its status is changed by an MTS command.

8.90.1 LTS Command Format

LTS
LIST TERMINAL STATUS
TERMINAL NAME: [name]
OUTPUT ACCESS NAME: [acnm]

8.90.2 LTS Command User Responses

System Prompts	Response Required or Optional	User Responses
TERMINAL NAME:	O	Enter the device name of the terminal whose status you want to display. *The default value is to display the status of all terminals.*
OUTPUT ACCESS NAME:	O	Enter the name of the device or file pathname to which DX10 is to display or list the results of the LTS command. *The default value is the terminal local file (TLF).*

8.90.3 LTS Command Example

The following example shows how you can enter the LTS command to obtain the status of all terminals in a system:

```
[ ] LTS
LIST TERMINAL STATUS
  TERMINAL NAME:
  OUTPUT ACCESS NAME: LP01
```

8.90.4 Commands Related to LTS

MTS (Modify Terminal Status)

LUI

8.91 LUI (LIST USER IDs)

The LUI command displays a list of users currently authorized to use the DX10 systems. The information displayed includes each user's ID, the description of each user, and the privilege code associated with that ID.

8.91.1 LUI Command Format

```
LUI
LIST USER IDS
OUTPUT ACCESS NAME: [acnm]
```

8.91.2 LUI Command User Responses

System Prompt	Response Required or Optional	User Response
OUTPUT ACCESS NAME:	O	Enter the device name or the file pathname to which DX10 should write the map of the file specified. *The default is the terminal local file (TLF).*

8.91.3 LUI Command Example

The following example shows how you can enter the LUI command to print a list of user IDs at your terminal:

```
[] LUI
LIST USER IDS
 OUTPUT ACCESS NAME:

    USER ID     DESCRIPTION       PRIVILEGE CODE
       SYS000     SYSTEM                 7
       TCR001     TOM                    7
       CLN002     LARRY                  7
```

8.91.4 Commands Related to LUI

AUI	(Assign User ID)
DUI	(Delete User ID)
MUI	(Modify User ID)

8.92 MAD (MODIFY ABSOLUTE DISK)

The MAD command places specified data on a disk at a specified absolute track, sector, and word address. Only privileged users can enter this command. Refer to Volume III for a detailed description.

MADU

8.93 MADU (MODIFY ALLOCATABLE DISK UNIT)

The MADU command modifies a specified ADU. If verification data does not match the data already on the disk, the modification is not performed. Refer to Volume III for a detailed description.

8.94 MCC (MODIFY COUNTRY CODE)

The MCC command modifies the country code selected during system generation. Table 8-5 lists the legal country codes that can be specified.

WARNING

Incorrect use of this command can cause key indexed files (KIFs) to become nonusable. You should modify the country code only if you desire the collating sequence of the new country code.

You must enter a legal country code in response to the COUNTRY prompt. You must enter the MCC command after the system is loaded, but before beginning production on the system.

8.94.1 MCC Command Format

```
MODIFY COUNTRY CODE
COUNTRY NAME: < string>
```

8.94.2 MCC Command User Responses

System Prompt	Response Required or Optional	User Response
COUNTRY NAME:	R	Enter the characters that specify the country code for the system. The default is the current value in the response field of the prompt. You can specify any of the abbreviations or country names listed in Table 8-5. That is, you can specify JAPAN or only the character J.

8.94.3 MCC Command Example

In the following example, the MCC command modifies the country code for a system from its current value to that of Finland (FI).

```
[ ] MCC
MODIFY COUNTRY CODE
  COUNTRY: FINLAND
```

MCC

Table 8-5. Country Codes

Abbreviation	Country
AU	Austria
B	Belgium
D	Denmark
FI	Finland
FR	France
FRWP	French word processing
G	Germany
J	Japan
N	Norway
S	Sweden
SP	Spanish
SWIS	Switzerland
UK	United Kingdom
US	United States

8.94.4 Commands Related to MCC

SCC (Show Country Code)

8.95 MD (MAP DISK)

The MD command lists information about directories and files held on a disk volume. MD does not provide information about the data contained in a file but it does provide information about the attributes of a file. MD processes directories that contain up to 3,000 entries. The amount of information listed depends on the option or options that you specify with the command. Figure 8-3 shows the information that is listed when the *long form* option is specified.

Table 8-6 describes the information that the MD command produces when you select the *long form* option. The item numbers in Table 8-6 refer to the numbered items in Figure 8-3. You should note that the directory name VCATALOG that appears in disk maps refers to a directory maintained by DX10. VCATALOG is the top-level or zero level directory in each volume. You can abort the MD command by pressing the CMD key. (Refer to Table 4-2 for equivalent keys on other terminals.)

8.95.1 MD Command Format

```
MD
MAP DISK
PATHNAME: < acnm>  @
LISTING ACCESS NAME: [acnm] @
SHORT FORM?: < yes/no>
TOP LEVEL ONLY?: < yes/no>
DIRECTORY NODES ONLY?: < yes/no>
EXECUTION MODE (F,B): < F,B>
```

8.95.1.1 MD Command Short Form Option. By specifying the *short form* option, you can limit the information that the MD command produces to items 1 through 10 as shown in Figure 8-4. Table 8-6 describes those items.

8.95.1.2 MD Command Top Level Only Option. The top level only option produces information about all files and subdirectories in the directory specified. The contents of subdirectories are not listed. Figure 8-5 shows a sample output.

8.95.1.3 MD Command Directory Nodes Only Option. By specifying the directory nodes only option, you can limit the output of the MD command to the following information about the directories on a disk volume:

- The number of subdirectories and files that have been created under each directory
- The number of spaces for files and subdirectories that are still available for creation under the directory
- The total size occupied by the directory and its subdirectories and files

Figure 8-6 is an example of the information the MD command produces when you specify the directory nodes only option.

MD

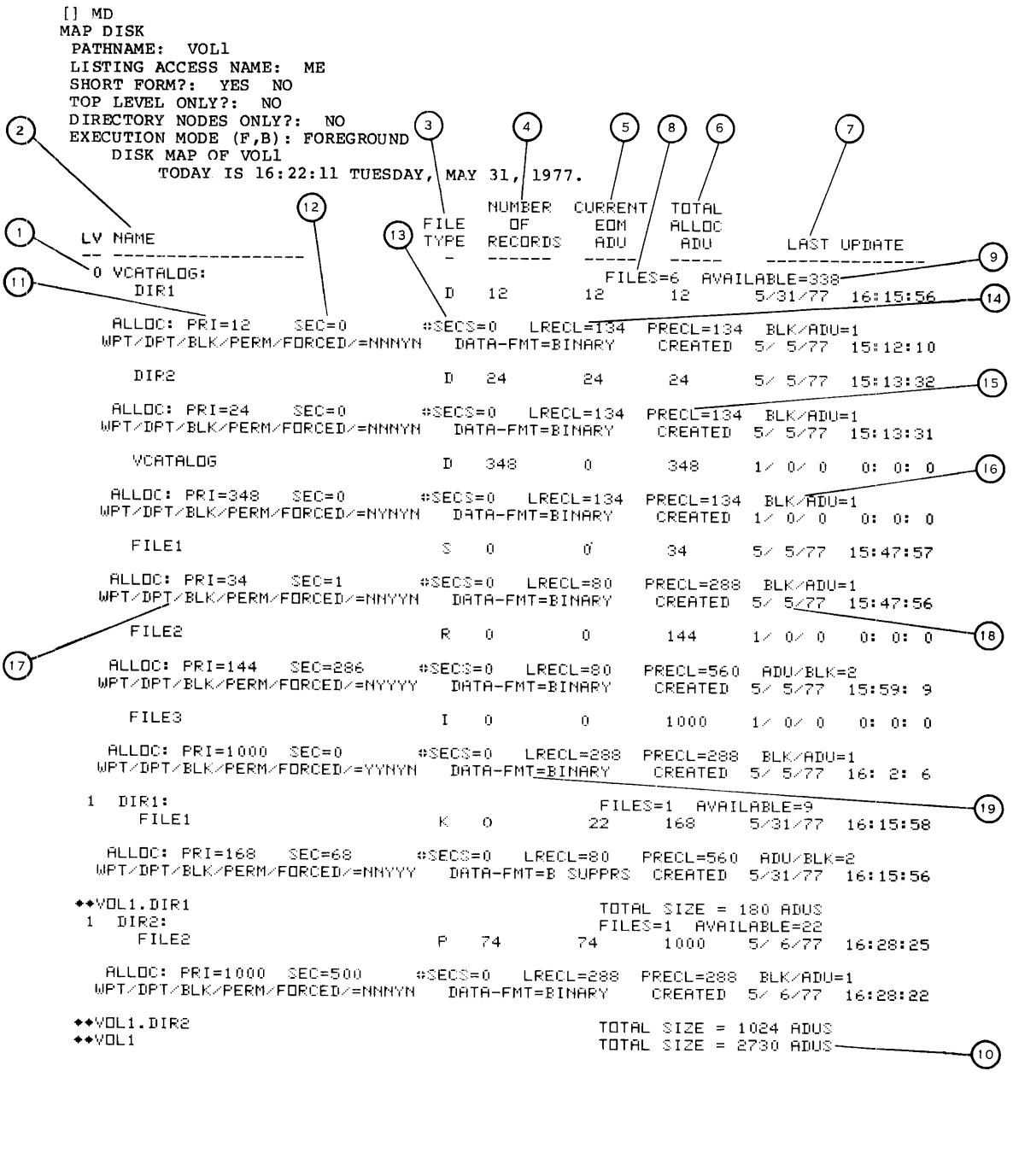


Figure 8-3. Long Form Output of the MD Command

Table 8-6. Long Form Output of the MD Command

	Field	Description
1.	LV	A level number (LV) identifies each directory and subdirectory on the disk volume by the level at which each was created. (For example, the VCATALOG directory is level 0; system files, user files, and user directories can be level 1; user files or user directories under the user directory of level 1 are level 2; and so on.)
2.	NAME	The name of each directory or file contained in the disk volume or directory being mapped. If you are mapping a file, this contains that file name.
3.	FILE TYPE	The file types of directories and subdirectories are identified as follows:

File Type	Meaning
D	Directory (relative record file, special usage).
I	Image file (relative record file, special usage).
K	Key indexed file.
P	Program file.
R	Relative record file.
S	Sequential file.

4.	NUMBER OF RECORDS	The current number of logical records in the file. For relative record files, that number is the number of the largest numbered record written on the file.
5.	CURRENT EOM ADU	The number of allocatable disk units (ADUs) that are needed to contain the current amount of data.
6.	TOTAL ALLOC ADU	The total number of ADUs allocated to the directory or file.
7.	LAST UPDATE	Date when the directory or file was last updated (month/day/year, hour:minute:second).
8.	FILES =	The number of subdirectories and files that have been created under a directory.

Table 8-6. Long Form Output of the MD Command (Continued)

	Field	Description
9.	AVAILABLE =	The number of spaces for subdirectories and files that are still available for creation under a directory.
10.	TOTAL SIZE =	The total space (in ADUs) currently allocated for all files under the directory named at the left.
11.	PRI =	The primary allocation size of a file or directory in ADUs. (The primary allocation size for a file is the amount of disk space that was reserved for a file when it was created.)
12.	SEC =	The secondary allocation size of a file or directory in ADUs. Zero is specified if the file is not expandable.
13.	#SECS =	The number of secondary allocations that have been made in the file.
14.	LRECL	The logical record length (in bytes) assigned to the file when it was created.
15.	PRECL	The physical record length (in bytes) assigned to the file when it was created.
16.	ADU/BLK (or BLK/ADU)	The relationship of physical records (blocks) to ADU size. ADU/BLK is the number of ADUs in one block or BLK/ADU is the number of blocks in one ADU. You can use this information to derive wasted space and to compute block numbers. This information is primarily used in troubleshooting procedures.
17.	WPT/DPT/BLK/ PERM/FORCED	Shows whether any of the following file conditions are set: Write-protected Delete-protected Blocked Permanent Forced write A Y appears (in a string of Ys and Ns) if the flag for the condition is set. An N appears in the string if the flag for the condition is not set.
18.	CREATED	The date that the file was created (month/day/year, hour:minute:second).

Table 8-6. Long Form Output of the MD Command (Continued)

Field	Description
19. DATA-FMT	The format of the data stored on the file. B SUPPR indicates that the data is stored on the file in blank-suppressed format. BINARY indicates that data on the file is stored in nonblank suppressed (straight data) format.

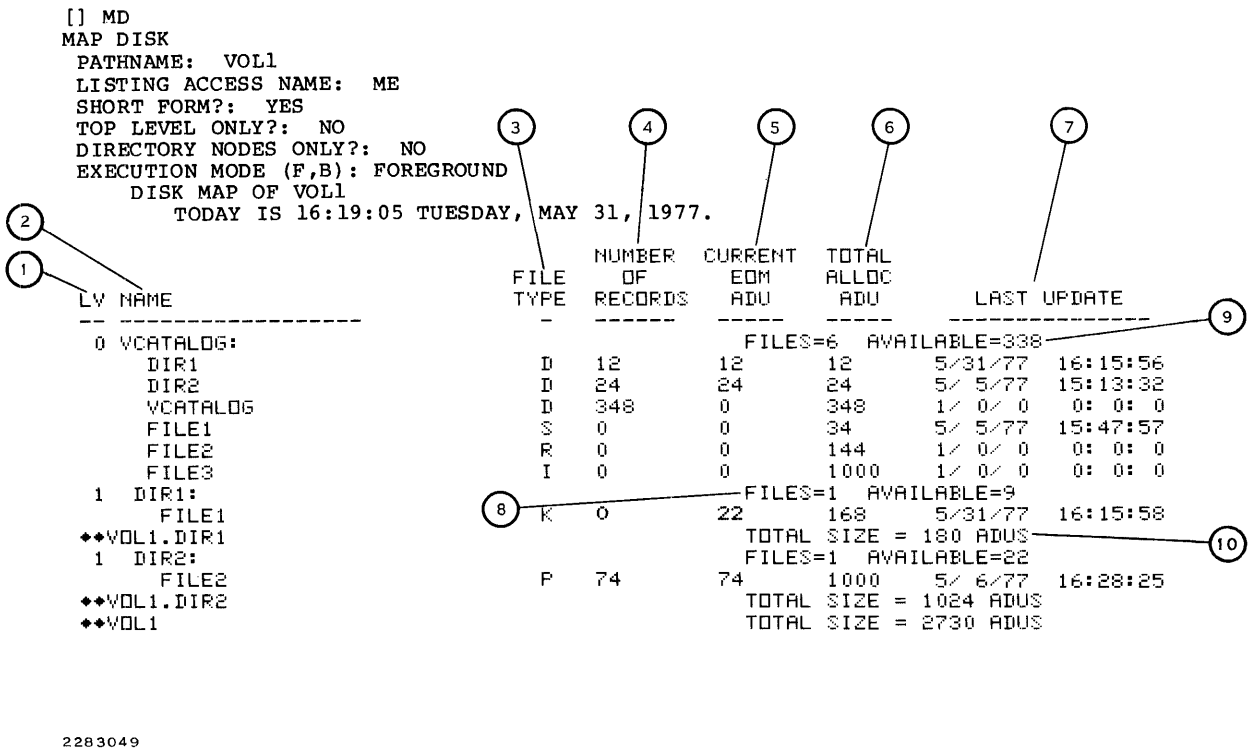


Figure 8-4. Short Form Output of the MD Command

MD

```
[ ] MD
MAP DISK
 PATHNAME: VOL1
 LISTING ACCESS NAME: ME
 SHORT FORM?: YES
 TOP LEVEL ONLY?: NO  YES
 DIRECTORY NODES ONLY?: NO
 EXECUTION MODE (F,B): FOREGROUND
     DISK MAP OF VOL1
       TODAY IS 20:18:38 WEDNESDAY, JUN 29, 1977.

                                    NUMBER CURRENT TOTAL
                                FILE   OF     EOM    ALLOC
LV NAME                         TYPE RECORDS  ADU     ADU        LAST UPDATE
-- --------------------           -   ------  -----  -----    ---------------
 0 VCATALOG:                                      FILES=6 AVAILABLE=338
     DIR1                       D    12       12     12       5/31/77  16:15:56
     DIR2                       D    24       24     24       5/ 5/77  15:13:32
     VCATALOG                   D    348      0      348      1/ 0/ 0   0: 0: 0
     FILE1                      S    0        0      34       5/ 5/77  15:47:57
     FILE2                      R    0        0      144      1/ 0/ 0   0: 0: 0
     FILE3                      I    0        0      1000     1/ 0/ 0   0: 0: 0
**VOL1                                      TOTAL SIZE = 2730 ADUS
```

2283050

Figure 8-5. Top Level Only Output of the MD Command

```
[ ] MD
MAP DISK
 PATHNAME: VOL1
 LISTING ACCESS NAME: ME
 SHORT FORM?: YES  NO
 TOP LEVEL ONLY?: NO  NO
 DIRECTORY NODES ONLY?: NO  YES
 EXECUTION MODE (F,B): FOREGROUND
     DISK MAP OF VOL1
       TODAY IS 16:49:31  TUESDAY, MAY 31, 1977.

 0 VCATALOG:                                        FILES=6  AVAILABLE=338
 1  DIR1:                                           FILES=1  AVAILABLE=9
**VOL1.DIR1                                        TOTAL SIZE = 180 ADUS
 1  DIR2:                                           FILES=1  AVAILABLE=22
**VOL1.DIR2                                        TOTAL SIZE = 1024 ADUS
**VOL1                                             TOTAL SIZE = 2730 ADUS
```

2283051

Figure 8-6. Directory Nodes Only Output of the MD Command

8.95.2 MD Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname that identifies the disk volume, top-level directory, or file about which you want to list information.
LISTING ACCESS NAME:	O	Enter the device name of the device or the pathname of a file to which you want to list the information that the MD command produces. *The default value is the terminal local file (TLF).*
SHORT FORM:	R	Enter Y (yes) to indicate that only the *short form* information is to be produced, or enter N (no) to specify that the *long form* is to be produced.
TOP LEVEL ONLY?:	R	Enter Y (yes) to indicate that only information about the directory specified in response to the PATHNAME prompt is to be listed. Enter N (no) to indicate that the map is to include information about all files and subdirectory files under the directory.
DIRECTORY NODES ONLY?:	R	Enter Y (yes) to indicate that only information about directories (none about files) is to be listed as follows: 1. The number of subdirectories and files that have been created under each directory. 2. The number of spaces for files and subdirectories that are still available for creation under the directory. 3. Total size of each directory. Enter N (no) to indicate that information about all files under the directory is to be produced.

MD

System Prompts	Response Required or Optional	User Responses
EXECUTION MODE (F,B):	R	Enter F (foreground) to execute the task in foreground mode. Enter B (background) to execute the task in background mode. *The initial value is FOREGROUND.*

8.95.3 MD Command Example

Figure 8-3 through Figure 8-6 show examples of the MD command and the results that you can produce by entering the different options available.

8.95.4 Commands Related to MD

LD (List Directory)

8.96 MDS (MODIFY DEVICE STATE)

The MDS command enables you to modify the state of specific devices without regenerating the system. This facility enables you to place devices offline or online to the system. The ability to place devices offline enables you to anticipate future system requirements by generating system configurations for an expanded system and then setting these phantom devices offline until they are physically present. The ability to place devices offline while they undergo routine maintenance also protects the system from erroneous access attempts that might be made to the device while it is disconnected.

Placing the device in the online state gives the system access to the device.

8.96.1 MDS Command Format

```
MDS
MODIFY DEVICE STATE
DEVICE NAME: < acnm> @
DEVICE STATE: < ON/OFF/DIAG>
```

8.96.2 MDS Command User Responses

System Prompts	Response Required or Optional	User Responses
DEVICE NAME:	R	Enter the four-character name of the device (for example, ST01).
DEVICE STATE:	R	Enter OFF either if the device specified is not in the system-generated configuration or if the device must be placed offline. Enter ON if the device specified must be placed online. Enter DIAG for use with online diagnostics.

In addition to the displayed prompts and responses, you can enter the following prompts and responses by using the expert mode with the MDS command:

Prompt Name	Function	User Responses
CODE8	Turns on 8-bit character manipulation.	YES = on NO = off
RAW	Turns on disk hardware read after write for referenced unit. Affects all disk units except DS31.	YES = on NO = off
BRW	Turns on software read after write check of the bit maps for the unit referenced. Affects all disk units.	YES = on NO = off

MDS

For information on using expert mode, refer to Section 4.

8.96.3 MDS Command Example

The following example shows how you can enter the additional prompts of the MDS command in expert mode:

[ ] MDS DEVICE N = DS01, DEVICE S = ON, CODE 8 = Y, RAW = Y, BRW = Y

8.96.4 Commands Related to MDS

LDC (List Device Configuration)

8.97 MFN (MODIFY FILE NAME)

The MFN command changes the name of a file. In response to prompting messages of the MFN command, you must enter the pathname that currently identifies a file and the new pathname that is to identify the file. The directory structure of the new pathname must already exist.

NOTE

You cannot change the names of files to which a LUNO is currently assigned.

8.97.1 MFN Command Format

MFN
MODIFY FILE NAME
OLD PATHNAME: < acnm> @
NEW PATHNAME: < acnm> @
REPLACE?: < yes/no>

8.97.2 MFN Command User Responses

System Prompts	Response Required or Optional	User Responses
OLD PATHNAME:	R	Enter the pathname that currently identifies the file. Any aliases of the old pathname are deleted.
NEW PATHNAME:	R	Enter the new pathname that is to identify the file.
REPLACE?:	R	Enter Y (yes) to specify that the new pathname is to become the file pathname previously identified by the old pathname. If the new pathname identifies an existing file, that file is deleted when you execute the MFN command. However, if the new pathname previously existed, any previous aliases of the new pathname are retained. Enter N (no) to specify that the new pathname is to become the file pathname previously identified by the old pathname unless the new pathname identifies an existing file. If you enter N and the new pathname does identify a file, the MFN command is rejected and you receive an error message.

MFN

8.97.3 MFN Command Example

The following example shows how you can enter an MFN command to change the name of a file from VOL1.DIR2.FILE1 to VOL1.DIR2.FILEA. Specifying Y for REPLACE? destroys the current contents of the file VOL1.DIR2.FILEA.

```
[ ] MFN
MODIFY FILE NAME
  OLD PATHNAME: VOL1.DIR2.FILE1
  NEW PATHNAME: VOL1.DIR2.FILEA
  REPLACE?: Y
```

8.97.4 Commands Related to MFN

SF (Show File)

8.98 MFP (MODIFY FILE PROTECTION)

The MFP command can assign write-delete protection or delete protection to a file or remove existing protection from a file. All files are created unprotected and they remain unprotected until an MFP command assigns protection. You cannot delete files that are write-protected.

NOTE

You cannot modify the protection of files to which a LUNO is currently assigned (that is, a file being currently used). You can protect directory entries as well as data files by using this command. Changing the protection of a directory entry does not change the protection of the data files cataloged in the directory. You must modify the protection of each data file individually. You can create a batch stream to modify the protection of all files in a given directory.

8.98.1 MFP Command Format

```
MFP
MODIFY FILE PROTECTION
PATHNAME: < acnm>  @
WRITE-DELETE PROTECT?: < yes/no>
DELETE PROTECT?: < yes/no>
```

8.98.2 MFP Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname that identifies the file whose protection is being modified.
WRITE-DELETE PROTECT?:	R	Enter Y (yes) to add write-delete protection to a file. Enter N (no) to remove existing write-delete protection from a file.
DELETE PROTECT?:	R	Enter Y (yes) to add delete protection to a file. Enter N (no) to remove existing delete protection from a file.

MFP

8.98.3 MFP Command Example

The following example shows how you can enter the MFP command to assign write-delete protection to a file identified by the pathname VOL1.DIRA.FILE2:

```
[ ] MFP
MODIFY FILE PROTECTION
  PATHNAME: VOL1.DIRA.FILE2
  WRITE-DELETE PROTECT?: Y
  DELETE PROTECT?: N
```

Since you specified write-delete protection, both write protection and delete protection are in effect.

8.98.4 Commands Related to MFP

LD	(List Directory)
MD	(Map Directory)

8.99 MHPC (MODIFY HARD-COPY TERMINAL PORT CHARACTERISTICS)

Communications ports on the TI 990 are defined during system generation. A terminal accessed through a port is expected to conform to the characteristics defined. You can modify port attributes by using the MHPC command. Changes remain in effect until modified through the MHPC command or until an initial program load (IPL) is performed on the system. The following attributes are modifiable:

- Terminal in service
- Read ASCII time-out
- Write time-out
- Read direct time-out
- Speed
- Terminal type
- Disconnect sequence code
- Duplex/echo code
- Use line turnaround
- Line turnaround character

8.99.1 MHPC Command Format

```
MHPC
MODIFY HARDCOPY TERMINAL PORT CHARACTERISTICS
TERMINAL ACCESS NAME: < acnm>
IN SERVICE (YES/NO): [yes/no]
READ ASCII TIMEOUT: [int]
WRITE TIMEOUT: [int]
READ DIRECT TIMEOUT: [int]
SPEED: [int]
TERMINAL TYPE: [int]
DISCONNECT SEQUENCE CODE: [int]
DUPLEX/ECHO CODE: [int]
USE LINE TURNAROUND (YES/NO): [yes/no]
LINE TURNAROUND CHARACTER: [int]
```

MHPC

8.99.2 MHPC Command User Responses

System Prompts	Response Required or Optional	User Responses
TERMINAL ACCESS NAME:	R	Enter the name that identifies the communications port whose attributes are being modified.
IN SERVICE (YES/NO):	O	This is a software switch that determines whether the port is available. If not in service, the DSR does not accept incoming calls and the CALL utility does not allow dial-out connections.
READ ASCII TIMEOUT:	O	The Read ASCII operation uses this value to determine how long to wait to receive a character. Enter the value in hexadecimal or decimal seconds. A zero indicates an infinite wait.
WRITE TIMEOUT:	O	The Write ASCII and Write Direct operations use this value to determine how long to wait for completion of sending a character. Enter the value in hexadecimal or decimal seconds. A zero value indicates an infinite wait.
READ DIRECT TIMEOUT:	O	This value determines how long the Read Direct operation is to wait to receive a character. Enter the value in either hexadecimal or decimal seconds. A zero indicates an infinite wait.
SPEED	O	This entry determines the communication speed for the port: 110, 300, 600, 1200, 2400, 4800, 9600, or >FFFF. This modification applies only to ports using the communications interface module. (Changing the TTY/EIA interface requires changing jumpers on the hardware.) The value >FFFF applies only when using the 212A modem, which operates at either 300 or 1200. The actual speed used is determined by the speed of the modem at the other end of the communications link. Entering >FFFF specifies that the port is to use this feature of the modem.

System Prompts	Response Required or Optional	User Responses
TERMINAL TYPE:	O	This entry defines the terminal type allowed at the port: 743, 745, 763, 765, 781, 783, 785, 787, and 820.
DISCONNECT SEQUENCE CODE:	O	The options are as follows: **Option** — **Description** 0 — Do not recognize any disconnect sequences 1 — Disconnect upon receipt of EOT, and transmit EOT to force disconnection 2 — Disconnect upon receipt of DLE EOT and transmit DLE EOT to force disconnection. The value assigned during system generation is 2.
DUPLEX/ECHO CODE:	O	The codes are as follows: **Code** — **Meaning** 0 — Full-duplex/echo 1 — Full-duplex/no echo 2 — Half-duplex If echo is enabled in full-duplex, characters transmitted by the terminal are echoed unless echo is disabled by the application program. The value of 0 is supplied for full-duplex during system generation. In half-duplex, no echo is automatic.
USE LINE TURNAROUND (YES/NO):	O	This parameter applies to half-duplex mode. If you enter YES, specify the turnaround character to be used. The default value is NO.

MHPC

System Prompts	Response Required or Optional	User Responses
LINE TURNAROUND CHARACTER:	O	If the answer to USE LINE TURNAROUND prompt is YES, this line turnaround character is used. This character is a decimal or hexadecimal number less than > 20 (space). The default character is > 04 (EOT).

NOTE

If using half-duplex with line turnaround characters, EOT is the recommended choice for use with SCI-oriented applications. If using EOT as the line turnaround character, the DISCONNECT SEQUENCE CODE cannot be 1.

If you do not enter a response to a particular prompt (other than TERMINAL ACCESS NAME), the current existing value of that parameter applies.

NOTE

MHPC does not allow modifications if another task is using the port.

8.99.3 MHPC Command Example

The following example shows how you can use the MHPC command to modify the terminal in service, speed, terminal type, and disconnect sequence code for ST02:

```
[ ] MHPC
MODIFY HARDCOPY TERMINAL PORT CHARACTERISTICS
 TERMINAL ACCESS NAME: ST02
 IN SERVICE (YES/NO): Y
 READ ASCII TIMEOUT:
 WRITE TIMEOUT:
 READ DIRECT TIMEOUT:
 SPEED: 300
 TERMINAL TYPE : 763
 DISCONNECT SEQUENCE CODE: 1
 DUPLEX/ECHO CODE:
 USE LINE TURNAROUND (YES/NO):
 LINE TURNAROUND CHARACTER:
```

8.99.4 Commands Related to MHPC

ANS	(Answer Incoming Call)
CALL	(Call Terminal)
DISC	(Terminal Disconnection)
LHPC	(List Hard-Copy Terminal Port Characteristics)

8.100 MHR (MODIFY HORIZONTAL ROLL)

After you activate the Text Editor, the MHR command specifies the number of columns the file being edited will roll right or left when the horizontal roll function is activated. Refer to Volume IV for a detailed description.

8.101 MIR (MODIFY INTERNAL REGISTERS)

The MIR command modifies the internal registers (program counter (PC), workspace pointer (WP), and status register (ST)) of a task. Refer to Volume III for a detailed description.

MKF

8.102 MKF (MAP KEY INDEXED FILE)

The MKF command lists the characteristics of a key indexed file (KIF). Figure 8-7 shows the format of the information listed. Table 8-7 describes the information given under each heading of the KIF map.

8.102.1 MKF Command Format

MKF
MAP KEY INDEX FILE
PATHNAME: < acnm> @
LISTING ACCESS NAME: [acnm]@

8.102.2 MKF Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname that identifies the KIF you want to map.
LISTING ACCESS NAME:	O	Enter the device name of a device or the pathname of a file to which the MKF command should list the information it produces. *The default value is the terminal local file (TLF).*

```
[] MKF
MAP KEY INDEX FILE
 PATHNAME:    VOL1.DIR1.FILE1
 LISTING ACCESS NAME:    ME
     FILE MAP OF VOL1.DIR1.FILE1
       TODAY IS 16:17:34 TUESDAY, MAY 31, 1977.

KEYS:
          START                                  DUPLICATES
  KEY    COLUMN    LENGTH    MODIFIABLE          ALLOWED
    1         1         5         N                 N
    2         8         3         Y                 Y
    3        25        15         Y                 N
```

2283052

Figure 8-7 KIF Information the MKF Command Displays

Table 8-7. Information the MKF Command Produces

Heading	Description
KEY	A number from 1 through 14 that identifies which key of the file. Key number 1 is the primary key.
START COLUMN	The character position in a record at which a keyed field starts. The first character position in a record is 1.
LENGTH	The length of a key field.
MODIFIABLE?	N (no) indicates that the contents of a keyed field must be present and cannot change when the record containing the key is rewritten. Y (yes) indicates that the contents of a keyed field can change when the record containing the key is rewritten or need not be present when the record is written.
DUPLICATES ALLOWED	Y (yes) indicates that the value of the key can be duplicated in the same keyed field in other records in the file. N (no) indicates that the value of the key cannot be duplicated in the same field of any other records in the file.

8.102.3 MKF Command Example

Figure 8-7 shows an example of using the MKF command to obtain a map of a KIF.

8.102.4 Commands Related to MKF

CFKEY (Create Key Indexed File)

8.103 MKL (MODIFY KIF LOGGING)

The MKL command selects the logging type for KIFs. The logging type controls switching off the forced write attribute of the KIF. Any operation that modifies the contents of the file (insert, rewrite, and delete) performs significantly faster when the file is in deferred write mode. This faster performance results from a decreased number of disk I/O operations. The MKL command is valid for both empty files and files that contain data.

The following list describes the logging options available with the MKL command:

Logging Option	Description
FULL	FULL logging makes the file a forced write file for all KIF operations. A KIF is initially created as a FULL logging file (forced immediate writes).
PARTIAL	PARTIAL logging switches the file to deferred write for the insert operation. The delete and rewrite operations behave like full logging files (forced immediate writes).
NONE	The file is in deferred write mode for all file operations. The file returns to the full logging mode after the file is closed (forced immediate writes).

CAUTION

If you use the MKL command to change the logging on a KIF to NONE until the next close operation and the correct operation of the KIF is interrupted due to a system crash or any other condition causing the KIF I/O operation to terminate abnormally, the specified KIF is rendered unusable. You should delete the file, recreate it, and then use the MKL command again.

Table 8-8 illustrates whether disk accesses are required during the initial creation o a single KIF in which the tree structure requires only a root node and a leaf node. Keys are inserted in sequential order.

Table 8-8. Disk Accesses Required for Initial Loading of a KIF

Operation	FULL[1]	PARTIAL[1]	NONE[1]
Read Data Record	M	M	M
Prelog Data Record	Y	N	N
Write Data Record	Y	Y	M
Read Root Node	M	M	M
Read Leaf Node	M	M	M
Prelog Leaf Node	Y	N	N
Write Leaf Node	Y	N[2]	M
Modify FDR	Y	Y	M

Notes:

[1] The entries in this column are as follows:

M means the record need not be read from disk because it is in memory. The record is written to disk only when the file is closed or when other buffers or tasks need the memory occupied by the file. (In the case of the file descriptor record (FDR), the FDR is written to disk when the LUNO assigned to the file is released.)

N means the operation is not performed.

Y means the record must be read from or written to disk.

[2] This is Y for single-key KIFs.

NONE logging significantly increases the speed of all KIF operations because it turns off logging and forced write. However, problems arise when errors occur while the file is in deferred write mode. Because parts of the file may have been modified before the error is detected, the file is no longer totally correct and you should not use it. Therefore, before you set file logging to NONE, you should make a copy of it if the file already contains data. If an error occurs or the system crashes while the file is in deferred write mode, you can restore the file to the condition it was in before you made changes to it. You can then reapply the changes. Also, all programs that modify the file are required to terminate whenever any error or informative code is returned. Thus, you can accomplish the building of a KIF from a sequential file by using the MKL command with the NONE option specified, followed by the CSK command. If CSK does not complete normally, you can correct the reason for the failure, delete the incorrect KIF, create a new file, execute the MKL command on the new file, and execute another CSK command. You can substitute your loading program in place of the CSK command, but it must properly handle error conditions.

PARTIAL logging significantly increases the speed of KIF insert operations but you might need to rebuild the file. If a failure occurs after the first insert to a file and before the last close, you must rebuild the file using the CKR command. The CKR command copies the potentially inconsistent file to a sequential file or precreated KIF file. All data in the potentially inconsistent file is recovered except the last file operation, if the operation was taking place at the time of failure.

MKL

FULL logging files are always forced write files. All data is prelogged before any changes are made to the data. If a KIF operation (rewrite, delete, or insert) fails due to a system crash or disk I/O error, the modified data is replaced with the prelogged copy of the data at the start of the next KIF operation. The prelogging procedure provides a safe and fast recovery mechanism for KIF operations. The penalty for these desirable features is reduced access speed during inquiry or maintenance operations.

8.103.1 MKL Command Format

```
MKL
MODIFY KIF LOGGING
KEY INDEX FILE: <acnm> @
LOGGING(FULL,PARTIAL,NONE): <{F,P,N}>
```

NOTE

The initial value for the LOGGING(FULL,PARTIAL,NONE) prompt is different for background and foreground. In background mode, the initial value is NONE. In foreground mode, the initial value is FULL.

8.103.2 MKL Command User Responses

System Prompts	Response Required or Optional	User Responses
KEY INDEX FILE:	R	Enter the pathname of the KIF to prepare for initial loading.
LOGGING(FULL, PARTIAL,NONE):	R	Specify F (FULL) if you want the KIF forced write for all KIF operations. Specify P (PARTIAL) to switch the file to deferred write for the insert operation; the delete and rewrite operations then behave like full logging files. Specify N (NONE) if you want the file in deferred write mode for all file operations; the file returns to full logging mode when the file is closed.

8.103.3 MKL Command Example

The following example demonstrates the creation of a KIF and the preparation of the file for the initial record loading using the MKL command.

```
[] CFKEY
CREATE KEY INDEXED FILE
 PATHNAME:   VOL1.KIF
 LOGICAL RECORD LENGTH:     80
 PHYSICAL RECORD LENGTH:     >240
 INITIAL ALLOCATION:   100
 SECONDARY ALLOCATION:   200
 MAXIMUM SIZE:    125
KEY DESCRIPTION FOR KEY NUMBER 1
 START POSITION:   1
 KEY LENGTH:   5
 DUPLICATES?: NO
 MODIFIABLE?: NO
 ANY MORE KEYS?:   YES
KEY DESCRIPTION FOR KEY NUMBER 2
 START POSITION:   1   8
 KEY LENGTH:   4
 DUPLICATES?: NO   Y
 MODIFIABLE?: NO
 ANY MORE KEYS?:  YES  NO

[] MKL
MODIFY KIF LOGGING
 KEY INDEX FILE:    VOL1.KIF
 LOGGING(FULL,PARTIAL,NONE):   NONE
```

8.103.4 Commands Related to MKL

CSK (Copy Sequential File to KIF)

ML

8.104 ML (MOVE LINES)

After you activate the Text Editor, the ML command moves lines from one position in the current file to another position and deletes the specified lines from their original positions. Refer to Volume IV for a detailed description.

8.105 MLP (MODIFY LUNO PROTECTION)

You can protect or unprotect a global LUNO at system initialization time by using the MLP command. You should protect system global LUNOs since release of them may adversely affect system performance. You should execute the MLP command at system initialization time; however, you can invoke MLP at any time. Note that LUNO protection is not a security feature and its use is discretionary. Any user with a privilege level of two or higher can unprotect a LUNO and then release it. If you enter a LUNO other than a global LUNO, MLP ignores the response and returns to SCI.

CAUTION

Do not unprotect system LUNOs listed in Appendix B.

8.105.1 MLP Command Format

```
MLP
MODIFY LUNO PROTECTION
PROTECT OR UNPROTECT? (P/U): <{P,U}>
LUNO: <integer>
```

8.105.2 MLP Command User Responses

System Prompts	Response Required or Optional	User Responses
PROTECT OR UNPROTECT? (P/U):	R	Enter P to protect the LUNO or U to unprotect the LUNO.
LUNO:	R	Enter an integer in the range of >00 to >FF. If you do not specify a LUNO, MLP selects the global LUNO most recently accessed by the station initiating the command.

8.105.3 MLP Command Examples

The following example shows how you can enter the MLP command to protect global LUNO 10:

```
[ ] MLP
MODIFY LUNO PROTECTION
  PROTECT OR UNPROTECT? (P/U): P
  LUNO: 10
```

MLP

The following example shows how you can enter the MLP command to unprotect the most recently used global LUNO accessed at the station initiating the command. The most recently accessed global LUNO selected by MLP in the example is LUNO >7.

```
[ ] AGL
ASSIGN GLOBAL LUNO
  LUNO:
  ACCESS NAME: LP01
ASSIGNED LUNO = >7

[ ] MLP
MODIFY LUNO PROTECTION
  PROTECT OR UNPROTECT? (P/U): U
  LUNO: >7
```

8.105.4 Commands Related to MLP

AGL	(Assign Global LUNO)
RGL	(Release Global LUNO)
SIS	(Show I/O Status)

8.106 MM (MODIFY MEMORY)

The MM command modifies the memory image of a task, starting at the address specified. Refer to Volume III for a detailed description.

MOE

8.107 MOE (MODIFY OVERLAY ENTRY)

The MOE command allows you to alter the data supplied when an overlay was installed.

8.107.1 MOE Command Format

```
MOE
MODIFY OVERLAY ENTRY
PROGRAM FILE PATHNAME: < acnm> @
MODULE NAME OR ID: < string> @ or < int> @
```

After you respond to these prompts, the following appears:

```
ID: [int]@
NAME: [string]@
RELOCATABLE: < yes/no>
DELETE PROTECTED: < yes/no>
```

The values defined when the overlay was installed are displayed, with the cursor in the first position of the NAME field. You can change any of the entries or accept them by pressing the RETURN key.

8.107.2 MOE Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE PATHNAME:	R	Enter the pathname of the program file upon which the overlay is resident.
MODULE NAME OR ID:	R	Enter the overlay name or ID of the overlay installed on the specified program file.
ID:	O	Enter the ID of the overlay. If you entered the overlay name, the system automatically places the associated overlay ID in the response field of this prompt.
NAME:	O	Enter the name of the overlay. If you entered the overlay ID, the system automatically places the associated overlay name in the response field of this prompt.
RELOCATABLE:	R	If you enter YES, the overlay is allowed to be loaded at an address other than its natural load address.
DELETE PROTECTED:	R	If you enter YES, the overlay is protected against accidental deletion.

8.107.3 MOE Command Example

In the following example, the IO command installs a relocatable overlay named OLAY1 on the .USERPROG program file. LUNO >0A has been assigned to the file where the object module resides, and task >83 identifies the associated task segment. The MOE command causes the overlay to be delete protected.

```
[ ] IO
INSTALL OVERLAY
  PROGRAM FILE OR LUNO: .USERPROG
  OVERLAY NAME: OLAY1
  OVERLAY ID: 0
  OBJECT PATHNAME OR LUNO: >0A
  RELOCATABLE?: YES
  DELETE PROTECT?: NO
  ASSOCIATED TASK NAME OR ID: >83

[ ] MOE
MODIFY OVERLAY ENTRY
  PROGRAM FILE PATHNAME: .USERPROG
  MODULE NAME OR ID: OLAY1
  ID: >40
  NAME: OLAY1
  RELOCATABLE: YES
  DELETE PROTECTED?: YES
```

8.107.4 Commands Related to MOE

DO	(Delete Overlay)
IO	(Install Overlay)
MPE	(Modify Procedure Entry)
MTE	(Modify Task Entry)

MPE

8.108 MPE (MODIFY PROCEDURE ENTRY)

The MPE command allows you to modify the data supplied when a procedure was installed.

8.108.1 MPE Command Format

```
MPE
MODIFY PROCEDURE ENTRY
PROGRAM FILE PATHNAME: < acnm> @
MODULE NAME OR ID: < string> @ or < int> @
```

After you respond to these prompts, the following appears:

```
ID: [int]@
NAME: [string]@
MEMORY RESIDENT: < yes/no>
DELETE PROTECTED: < yes/no>
EXECUTE PROTECTED: < yes/no>
WRITE PROTECTED: < yes/no>
WRITABLE CONTROL STORAGE: < yes/no>
```

The values displayed are those that were defined when the procedure was installed. The cursor is in the first position of the NAME field. You can change displayed values or accept each one by pressing the RETURN key.

8.108.2 MPE Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE PATHNAME:	R	Enter the pathname of the program file within which the procedure resides.
MODULE NAME OR ID:	R	Enter the procedure name or the ID of the procedure installed on the specified program file.
ID:	O	Enter the ID of the procedure. If you entered the procedure name, the system automatically places the associated procedure ID in the response field of this prompt.
NAME:	O	Enter the name of the procedure. If you entered the procedure ID, the system automatically places the associated procedure name in the response field of this prompt.

System Prompts	Response Required or Optional	User Responses
MEMORY RESIDENT:	R	If you enter YES and the procedure is installed on the program file selected during initial program load (IPL) (usually .S$PROGA), the procedure is loaded into memory during IPL and remains in memory after execution. You should enter YES for this prompt only if you are very familiar with the system.
DELETE PROTECTED:	R	If you enter YES, the procedure is protected against accidental deletion.
EXECUTE PROTECTED:	R	If you enter YES, execution of the procedure is prohibited. The protection is enforced only on a 990/12 computer.
WRITE PROTECTED:	R	If you enter YES, procedure data cannot be modified in memory. The protection is enforced only on a 990/12 computer.
WRITABLE CONTROL STORAGE:	R	If you enter YES, the procedure uses the writable control storage area. Writable control storage is available only on a 990/12 computer with supplied software.

8.108.3 MPE Command Example

The following example shows how you can use the IP command to install a procedure named MYPROC on your own program file. LUNO > AA has been assigned to the program file, and LUNO > BB has been assigned to the file in which the object module resides. The MPE command changes the procedure name and specifies that the procedure is delete protected.

```
[ ] IP
INSTALL PROCEDURE
  PROGRAM FILE OR LUNO: > AA
  PROCEDURE NAME: MYPROC
  PROCEDURE ID: 0
  OBJECT PATHNAME OR LUNO: > BB
  MEMORY RESIDENT?: NO
  DELETE PROTECT?: NO
  EXECUTE PROTECT?: NO
  WRITE PROTECT?: NO
  WRITABLE CONTROL STORAGE?: NO
```

MPE

```
[ ] MPE
MODIFY PROCEDURE ENTRY
  PROGRAM FILE PATHNAME: MYPROG
  MODULE NAME OR ID: MYPROC
  ID: >40
  NAME: URPROC
  MEMORY RESIDENT?: NO
  DELETE PROTECTED?: YES
  EXECUTE PROTECTED?: NO
  WRITE PROTECTED?: NO
  WRITABLE CONTROL STORAGE?: NO
```

8.108.4 Commands Related to MPE

MOE	(Modify Overlay Entry)
MTE	(Modify Task Entry)

8.109 MPF (MAP PROGRAM FILE)

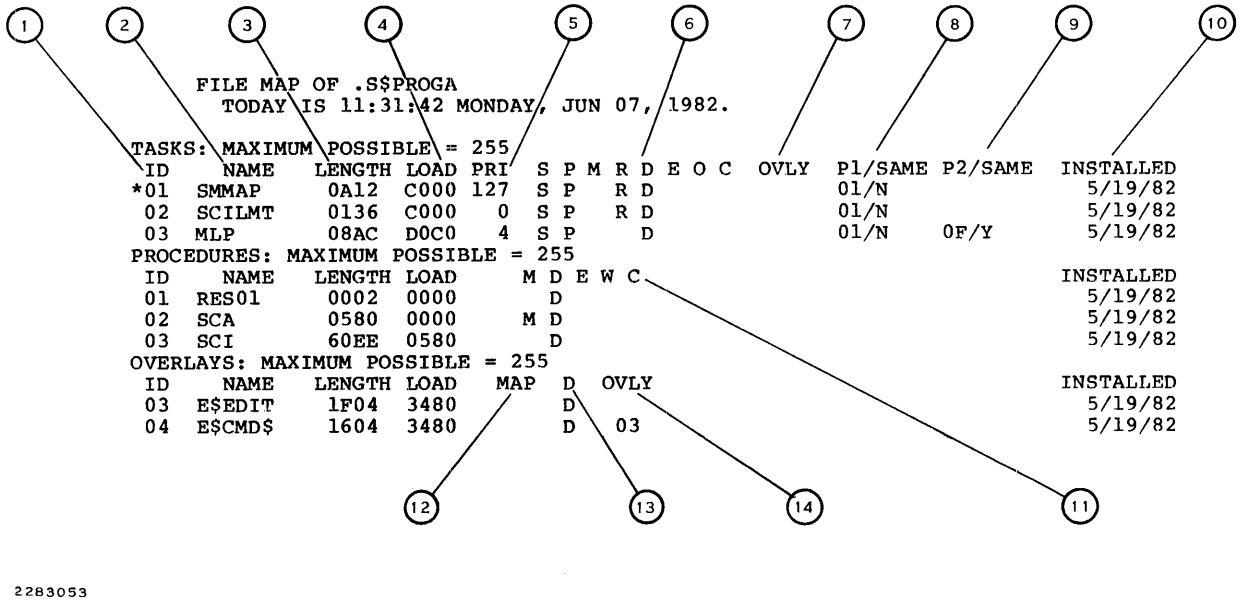
The MPF command displays information about the tasks, procedures, and overlays contained within a program file in ID order. Figure 8-8 shows the format of the information displayed. Table 8-9 describes the information given under each heading of the program file map.

8.109.1 MPF Command Format

MPF
MAP PROGRAM FILE
PATHNAME: < acnm> @
LISTING ACCESS NAME: [acnm]@

8.109.2 MPF Command User Responses

System Prompts	Response Required or Optional	User Responses
PATHNAME:	R	Enter the pathname that identifies the program file to be mapped.
LISTING ACCESS NAME:	O	Enter the device name of a device or the pathname of a file to which DX10 should write the map of the file specified. *The default value is the terminal local file (TLF).*



2283053

Figure 8-8. Information the MPF Command Displays

MPF

Table 8-9. Program File Information the MPF Command Produces

	Heading	Description
1.	ID	The identification number assigned to the task, procedure, or overlay when it was installed. An asterisk to the left of the ID number indicates that the task is a real-time task.
2.	NAME	The name of the task, procedure, or overlay.
3.	LENGTH	The amount of memory (in bytes) that the task, procedure, or overlay occupies when loaded.
4.	LOAD	The natural load address of the task, procedure, or overlay (the ORIGIN shown on a link edit map). Natural refers to the address assigned by the Link Editor.
5.	PRI	The priority of a task. An asterisk (*) to the left of the ID number indicates a real-time priority.
6.	SPMRDEOC	A corresponding letter in one of the columns indicates that the task has the characteristic as follows:

Column Heading	Task Characteristic
S	The task is a system task.
P	The task is privileged.
M	The task is memory resident.
R	The task is replicatable.
D	The task is delete protected.
E	The task is execute protected.
O	The task is overflow protected, and will terminate with task error code >F if arithmetic overflow occurs.
C	The task uses the predefined XOPs in writable control storage.

7.	OVLY (for tasks)	The ID number of the overlay last associated with a task.

Table 8-9. Program File Information the MPF Command Produces (Continued)

	Heading	Description
8.	P1/SAME	A number in the P1 column is the installed ID number of the first procedure associated with a task. A Y (yes) in the SAME column indicates that the procedure identified in the P1 column is from the same program file. An N (no) indicates that it is from the current system program file.
9.	P2/SAME	A number in the P2 column is the installed ID number of the second procedure associated with a task. A Y (yes) in the SAME column indicates that the procedure identified in the P2 column is from the same program file. An N (no) indicates that it is from the current system program file.
10.	INSTALLED	The date that the task, procedure, or overlay was installed. The format in which the date and time appear is as follows: month/day/year
11.	MDEWC	A corresponding letter in one of the columns indicates that the procedure has the characteristic as follows:

Column Heading	Procedure Characteristic
M	The procedure is memory resident.
D	The procedure is delete protected.
E	The procedure is execute protected.
W	The procedure is write protected.
C	The procedure uses the predefined XOPs in writable control storage.

12.	MAP	The overlay has a relocation bit map associated with it.
13.	D	The overlay is delete protected.
14.	OVLY (for overlays)	If another overlay is attached to a task, the ID number of that overlay appears in this column.

MPF

8.109.3 MPF Command Example

Figure 8-8 shows how you can use the MPF command to produce a map of a program file.

8.109.4 Commands Related to MPF

CFPRO (Create Program File)

8.110 MPI (MODIFY PROGRAM IMAGE)

The MPI command modifies a module (defined to be a task, procedure, or overlay) in a specified program file. Refer to Volume III for a detailed description.

MR

8.111 MR (MODIFY ROLL)

After you activate the Text Editor, the MR command changes the increment or decrement applied on a roll up or roll down function (the F1 and F2 function keys, respectively). Refer to Volume IV for a detailed description.

8.112 MRF (MODIFY RELATIVE TO FILE)

The MRF command changes data at an absolute word address within a file. Refer to Volume III for a detailed description.

MRM

8.113 MRM (MODIFY RIGHT MARGIN)

After you activate the Text Editor, the MRM command changes the edit line right margin. The system does not allow you to enter characters to the right of the right margin position. Refer to Volume IV for a detailed description.

8.114 MS (MODIFY SYNONYM)

The MS command allows you to access and change the synonyms and synonym values associated with your user ID or terminal. SCI responds to entry of an MS command in VDT mode by displaying the first available synonym (in alphabetical order) and its value as shown in Figure 8-9.

```
SYNONYM: A
VALUE: VOL1.DIRA.FILE1
```

Figure 8-9. Display of a Synonym and Its Value

When the first synonym and synonym value appear in response to entry of an MS command, you can enter a new synonym and new synonym value or you can modify the synonym value. You cannot modify synonyms without entering a new synonym value. If you modify the synonym name to contain lowercase characters, you cannot use the synonym as a response to an SCI prompt. Table 8-10 and Table 8-11 describe the keys that you can use to modify a synonym and its value. Table 8-10 describes the keys used at VDTs and teleprinter devices operating in TTY mode; Table 8-11 describes the keys used at VDTs in VDT mode.

NOTE

You can initiate the MS command with lowercase letters, but you must enter alphabetic responses to the command prompts in uppercase letters. SCI does not perform lowercase to uppercase mapping for the prompt responses to this command, as explained in the .OPTION primitive in Volume III.

Synonyms are displayed in alphabetical order. A blank (null) is displayed after the last synonym to allow you to add synonyms with the MS command. System synonyms, which begin with a dollar sign ($), are displayed after the blank synonym if you press the down arrow key when the blank synonym appears. Entering a system synonym in place of the blank and pressing RETURN also displays its value.

You can enter any synonym in the place of the blank, and the synonym and its value appear. You can then continue to list from that point. You can also put in the name of a synonym you want to examine even if you are not at the blank.

MS

The following list describes additional features of the MS command:

- You can obtain the alphabetically last synonym/value pair by clearing the synonym field and pressing the up arrow key.

NOTE

Clearing the synonym field does not affect the synonym or its value.

- You can obtain the alphabetically first system synonym/value pair by clearing the synonym field and pressing the down arrow key.
- If the synonym field is clear, you cannot tab through the field (by pressing the RETURN key). You must enter a response for this prompt (for example, entering of synonym characters or pressing the up or down arrow keys).
- If you clear the VALUE prompt field and press the RETURN key, the synonym and its value are deleted and the alphabetically next synonym/value pair is displayed.
- If you press the up arrow key when the alphabetically first synonym/value pair appears, the last system synonym/value pair appears. By pressing the up arrow key again, the next to last system synonym/value pair appears. When the first system synonym/value appears and you press the up arrow key, a blank synonym/value pair appears. By pressing the up arrow key, the MS command displays the alphabetically last synonym/value pair.
- If you press the down arrow key when the alphabetically last synonym/value pair appears, a blank synonym/value pair appears. By pressing the down arrow key again, the first system synonym/value pair appears. When the last system synonym/value pair appears and you press the down arrow key, the alphabetically first synonym/value pair appears.
- You can selectively delete any unwanted default synonyms, rather than executing the Clear Secret Synonyms (Q$SYN) command that deletes all system synonyms.

Table 8-10. Keys Used to Modify Synonyms and Synonym Values at Model 911 VDTs and Teleprinter Devices in TTY Mode

Key	Use
CMD	Terminates the MS command.
CONTROL/J (Down arrow)	You can press this key sequence at any time to abort the current synonym modification operation and to cause the next available synonym (in alphabetical order) to be listed.
CONTROL/U (Up arrow)	Causes the previous synonym (in alphabetical order) to be listed.
CONTROL/X	Exits from the MS command.
LINE FEED	Same use as the CONTROL/J key sequence.
RETURN	Signals DX10 that you are through modifying a synonym name or value. If you press RETURN without modifying a synonym name or synonym value, DX10 accepts the current value of the synonym name or synonym value and displays an appropriate response. In TTY mode, ending modification of a synonym name causes the system to display the synonym value. Ending modification of the synonym value causes DX10 to display the next synonym name or, if there are no more, a blank synonym name field.
CONTROL/1 (CTRL/3 on 820 and teleprinters) (ERASE FIELD) (SKIP)	Clears the value field.

MS

Table 8-11. Keys Used to Modify Synonyms and Synonym Values at Model 911 VDTs in VDT Mode

Key	Use
CMD	Exits from the MS command.
Down arrow	Causes the display of the next available synonym (in alphabetical order).
RETURN	Signals DX10 that you are through modifying a synonym name or value. If you press RETURN without modifying a synonym name or synonym value, DX10 accepts the current value of the synonym name or synonym value and displays an appropriate response. When the terminal is in VDT mode, ending modification to a synonym name (by pressing the RETURN KEY) causes DX10 to move the cursor to the synonym value field. If the synonym name is not defined, a blank field is displayed for the synonym value. When ending modification to the synonym value, DX10 displays the next synonym name and synonym value. If there are no more synonyms defined, blank synonym name and value fields are displayed.
TAB	Same use as the RETURN key.
Up arrow	Causes the display of the previous synonym (in alphabetical order).
SKIP	Deletes the synonym.
ERASE FIELD	Clears the displayed field.

8.114.1 MS Command Format

```
MS
MODIFY SYNONYM
SYNONYM: < name>
VALUE: [string]
```

8.114.2 MS Command User Responses

System Prompts	Response Required or Optional	User Responses
SYNONYM:	R	Enter one or more alphabetic or numeric characters, beginning with an alphabetic character. This value is the synonym to be assigned.
VALUE:	O	Enter the character string to which you want to assign the synonym. If you do not enter a value for the synonym, the current value assigned to the synonym is erased and the synonym is deleted.

8.114.3 MS Command Example

The following series of examples shows how you can use the MS command to modify synonyms and synonym values at a 911 VDT. To save space, the actual editing steps that you take to change a synonym or synonym value are not shown.

[ ] MS	You enter the MS command.
MODIFY SYNONYM SYNONYM: A VALUE: VOL1.DIRA.FILEA	SCI replies with the first available synonym and synonym value.

You modify the synonym and press the RETURN key.

SYNONYM: AA VALUE:	SCI displays the value of the new synonym and moves the cursor to the start of the synonym value to allow entry of a new value.

You enter the synonym value VOL1.DIRA.FILEAA and press the RETURN key.

SYNONYM: B VALUE: VOL1.DIRA.FILEB	The next synonym (in alphabetical order) and synonym value appear.

You press the RETURN key.

SYNONYM: B VALUE: VOL1.DIRA.FILEB	The cursor moves to the start of the value field to allow modification of the value.

You enter the synonym value VOL1.DIRB.FILEBB and press the RETURN key.

SYNONYM: C VALUE: VOL1.DIRA.FILEC	The next synonym (in alphabetical order) and synonym value are displayed.

MS

You clear the synonym field and press the down arrow key.

SYNONYM: $$12 VALUE: Y	The next synonym and synonym value appear.

You press the up arrow key twice.

SYNONYM: C VALUE: VOL1.DIRA.FILEC	The previous synonym (in alphabetical order) and its value are displayed.

You press the up arrow key.

SYNONYM: B VALUE: VOL1.DIRA.FILEBB	the previous synonym (in alphabetical order) and its value are displayed.

You press the down arrow key.

SYNONYM: C VALUE: VOL1.DIRA.FILEC	The next synonym (in alphabetical order) and its value are displayed.

You press the CMD key to exit from the MS command.

8.114.4 Commands Related to MS

AS	(Assign Synonym)
LS	(List Synonyms)

8.115 MSG (DISPLAY MESSAGE)

The MSG command displays a message at the terminal. If you execute the command in foreground or the command is called by an interactive procedure, the text is displayed at the bottom line of the VDT or the next line of a TTY mode device. When you press RETURN, SCI proceeds with the command procedure.

NOTE

The sending and receiving terminals are the same.

If the MSG command was coded with a response to the REPLY prompt, that response is a synonym name. The interactive user can enter the characters in response to the message and those characters become the synonym value of the synonym name given in the REPLY prompt.

If the command was executed in batch mode, the characters of the TEXT prompt are written to the batch listing file and the REPLY prompt is ignored.

8.115.1 MSG Command Format

```
MSG
TEXT: < string>
REPLY: [string]
```

8.115.2 MSG Command User Responses

System Prompts	Response Required or Optional	User Responses
TEXT:	R	Enter the message text in response to this prompt. The text cannot exceed 50 characters. The message appears exactly as you enter it on the bottom line of your VDT, the next line of a TTY mode device, or the next list line of a batch stream list if MSG is called from a batch stream. (You do not need to enclose the message text in quotes.) If the MSG command is called from expert mode, a procedure, or a batch stream, it is scanned for synonym substitution.

MSG

System Prompts	Response Required or Optional	User Responses
REPLY:	O	Enter a synonym name that will receive any text entered by an interactive user in response to the message text. The text entered by the interactive user in response to the message text will be the synonym value. The REPLY prompt is ignored in batch mode. The response to this prompt may not exceed 50 characters.

8.115.3 MSG Command Example

In the following example, assume you back up files from the system disk to a secondary disk every day, and the first component of the output file name is a synonym named DEV whose value is the device name where the secondary disk is installed. The availability of secondary disk drives varies from day to day; therefore, the same secondary disk drive usually is not used on consecutive days. You can send a message to the terminal prior to backing up the files and set the synonym DEV to correspond to the device name of the available secondary disk drive as follows:

```
[ ] MSG
  TEXT: TYPE NEW BACKUP DEVICENAME
  REPLY: DEV
TYPE NEW BACKUP DEVICENAME: DS02
```

In this example, the synonym DEV, defined in response to the REPLY prompt, is assigned the synonym value DS02 by SCI when an interactive user receives the message TYPE NEW BACKUP DEVICENAME: at his terminal.

In the next example, assume that within a command procedure a condition is detected that the user should be aware of. Within that procedure, the MSG command could be used to communicate with the user as follows:

```
.
.
.
MSG TEXT = "WARNING: NON FATAL ERROR. CONTINUE (Y/N)?",
   REPLY = ANSWER
.IF @ANSWER,LT,Y
.EXIT
.ENDIF
.
.
.
```

8.115.4 Commands Related to MSG

CM (Create Message)

8.116 MSM (MODIFY SYSTEM MEMORY)

The MSM command modifies the memory occupied by the operating system. It is similar to the Modify Memory (MM) command, except that you specify an overlay name or ID instead of a task run ID. Refer to Volume III for a detailed description.

MT

8.117 MT (MODIFY TABS)

After you activate the Text Editor, the MT command changes or clears the tab settings of the device. Refer to Volume IV for a detailed description.

8.118 MTE (MODIFY TASK ENTRY)

The MTE command allows you to alter the data supplied when a task was installed.

8.118.1 MTE Command Format

```
MTE
MODIFY TASK ENTRY
PROGRAM FILE PATHNAME: < acnm> @
MODULE NAME OR ID: < string> @ or < int> @
```

After you respond to these prompts, the following appears:

```
ID: [int]@
NAME: [string]@
REAL TIME: < yes/no>
PRIORITY: < int> @
MODIFY FLAGS?: < yes/no>
ATTACHED PROCEDURES?: < yes/no>
```

The values displayed for the ID, NAME, REAL TIME, and PRIORITY prompts are the values that were defined when the task was installed. The values displayed for MODIFY FLAGS? and ATTACHED PROCEDURES? are YES and NO, respectively. The cursor is set in the first position of the NAME field. You can change any of the entries or accept them by pressing the RETURN key.

After you enter the responses to the previous prompts, the task flags are displayed if MODIFY FLAGS? is YES:

```
SYSTEM: < yes/no>
PRIVILEGED: < yes/no>
MEMORY RESIDENT: < yes/no>
REPLICATABLE: < yes/no>
DELETE PROTECTED: < yes/no>
EXECUTE PROTECTED: < yes/no>
OVERFLOW: < yes/no>
WRITABLE CONTROL STORAGE: < yes/no>
```

The values defined when the task was installed are the values displayed. You can change any of the entries or accept them by pressing the RETURN key. After you enter the responses, the procedure prompts are displayed if you changed the ATTACHED PROCEDURES?: prompt to YES:

```
1ST PROCEDURE ID: < int> @
P1 FROM TASKS PROGRAM FILE: < yes/no>
2ND PROCEDURE ID: < int> @
P2 FROM TASKS PROGRAM FILE: < yes/no>
```

The values displayed are the values defined when the task was installed. You can change or accept them by pressing the RETURN key.

MTE

8.118.2 MTE Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE PATHNAME:	R	Enter the pathname of the program file on which the task you want to modify resides.
MODULE NAME OR ID:	R	Enter the task name or ID of the task installed on the specified program file.
ID:	O	Enter the ID of the task. If you entered the task name, the system automatically places the associated task ID in the response field of this prompt.
NAME:	O	Enter the name of the task. If you entered the task ID, the system automatically places the associated task name in the response field of this prompt.
REAL TIME:	R	If you enter YES, the task to be modified is installed as a real-time task.
PRIORITY:	R	If you enter YES in response to the REAL TIME prompt, the priority value specified must be in the range from 1 through 127. If you enter NO, the priority value specified must be in the range from 0 through 4.
MODIFY FLAGS?:	R	If you enter YES, you have the option of modifying the task flags.
ATTACHED PROCEDURES?:	R	If you enter YES, you have the option of modifying the attached procedures.
SYSTEM:	R	If you enter YES, the task is allowed to execute in system memory space. However, two conditions must be met. The ATTACHED PROCEDURES?: prompt in the first display must have been changed to YES (to specify that the task is linked with procedure 1, a dummy procedure on S$PROGA file), and the task's load address must be ≥C000. Otherwise, you cannot make the change.

System Prompts	Response Required or Optional	User Responses
PRIVILEGED:	R	If you enter YES, the task is allowed to execute privileged hardware instructions.
MEMORY RESIDENT:	R	If you enter YES and the task is installed on the program file selected during initial program load (IPL) (usually .S$PROGA), the task is loaded into memory during IPL and remains in memory after execution. You should enter YES for this prompt only if you are very familiar with the system.
REPLICATABLE:	R	If you enter YES, multiple copies of the task can be in memory simultaneously.
DELETE PROTECTED:	R	If you enter YES, the task cannot be deleted from the program file unless the Modify Task Entry (MTE) command is used to unprotect the task prior to the execution of the Delete Task (DT) command. If you enter NO, the task can be deleted by the DT command.
EXECUTE PROTECTED:	R	If you enter YES, execution of the task is prohibited. The protection is enforced only on a 990/12 computer.
OVERFLOW:	R	If you enter YES, the occurrence of arithmetic overflow causes control of the task to pass to the end action routine of the task. Overflow checking is enforced only on a 990/12 computer.
WRITABLE CONTROL STORAGE:	R	If you enter YES, the task uses the writable control storage area. Writable control storage is available only on a 990/12 computer with supplied software.
1ST PROCEDURE ID:	R	Enter the integer value representing the ID of a procedure attached to the task. If you enter zero, there are no procedures.

MTE

System Prompts	Response Required or Optional	User Responses
P1 FROM TASKS PROGRAM FILE?:	R	If you enter YES, the attached procedure whose ID was specified for the 1ST PROCEDURE ID prompt resides on the same program file as the task. If you enter NO, that procedure must reside on the .S$SHARED program file.
2ND PROCEDURE ID:	R	Enter the integer value representing the ID of a procedure attached to the task. If you enter zero, there is no second procedure.
P2 FROM TASKS PROGRAM FILE?:	R	If you enter YES, the attached procedure whose ID was specified for the 2ND PROCEDURE ID prompt resides on the same program file as the task. If you enter NO, that procedure must reside on the .S$SHARED program file.

8.118.3 MTE Command Example

In the following example, the IT command installs a task, having one attached procedure on the same program file. The MTE command modifies the procedure ID of the attached procedure. The default task flags are taken for both commands.

```
[ ] IT
INSTALL TASK
  PROGRAM FILE OR LUNO: .USERPROG
  TASK NAME: MYFILE
  TASK ID: 0
  OBJECT PATHNAME OR LUNO: >4A
  PRIORITY: 3
  DEFAULT TASK FLAGS?: YES
  ATTACHED PROCEDURES?: YES
ATTACH TASK PROCEDURES
  1ST PROCEDURE ID: >42
  P1 FROM TASKS PROGRAM FILE?: YES
  2ND PROCEDURE ID: 0
  P2 FROM TASKS PROGRAM FILE?: NO
```

```
[ ] MTE
MODIFY TASK ENTRY
  PROGRAM FILE PATHNAME: .USERPROG
  MODULE NAME OR ID: MYFILE
  ID: >40
  NAME: MYFILE
  REAL TIME: NO
  PRIORITY: 3
  MODIFY FLAGS?: NO
  ATTACHED PROCEDURES?: YES
  1ST PROCEDURE ID: >45
  P1 FROM TASKS PROGRAM FILE: YES
  2ND PROCEDURE ID: 0
  P2 FROM TASKS PROGRAM FILE: NO
```

8.118.4 Commands Related to MTE

DT	(Delete Task)
IT	(Install Task)
MOE	(Modify Overlay Entry)
MPE	(Modify Procedure Entry)

MTS

8.119 MTS (MODIFY TERMINAL STATUS)

You can employ MTS commands to control the following attributes of each terminal in your system:

- Terminals can be set ON or OFF. Terminals that are ON are in contact with SCI, and you can use them to enter commands to control DX10 operations. Terminals that are OFF are not in contact with SCI, cannot be used for command entry, and can only be used as physical input or output devices.
- You can set a terminal so that it requires log-on by the user before it can be operated under SCI.
- You can set VDTs to operate in either TTY mode or VDT mode. In TTY mode, a VDT displays only one line of output at a time and accepts only one line of input at a time. In VDT mode, a VDT displays a full screen of output at one time and accepts a full screen of input at one time. Teleprinter devices may only operate in TTY mode.

NOTE

If SCI is currently activated at a terminal that you are modifying, the new attributes do not take effect until after the Q command is entered and a user reactivates SCI at that terminal.

8.119.1 MTS Command Format

```
MTS
MODIFY TERMINAL STATUS
TERMINAL NAME: < name>
NEW STATUS (ON/OFF): [string]
NEW MODE (TTY/VDT): [string]
LOGIN REQUIRED?: [yes/no]
USER PRIVILEGE CODE: [integer]
DEFAULT MODE (TTY/VDT): [string]
```

8.119.2 MTS Command User Responses

System Prompts	Response Required or Optional	User Responses
TERMINAL NAME:	R	Enter the device name of the terminal whose status is to be modified.
NEW STATUS (ON/OFF):	O	Enter ON if users are allowed to enter commands to SCI from the terminal. Enter OFF to prevent users of the terminal from accessing SCI. *The default value is the current status of the terminal.*
NEW MODE (TTY/VDT):	O	Enter TTY to assign to TTY mode. Enter VDT to assign a VDT to VDT mode. *The default mode is the current mode of the terminal.*
LOG-IN REQUIRED?:	O	Enter Y to require log-on at the terminal before the terminal can be operated under SCI. Enter N (no) to allow users to access SCI at the terminal without logging on. *The default value is the current log-on requirements.*
USER PRIVILEGE CODE:	O	Enter an integer in the range 0 to 7 to specify the privilege level to be enforced at the terminal when log-on is not required. *The default value is the current value for the terminal.* This parameter is ignored when log-on is required.
DEFAULT MODE (TTY/VDT):	O	Enter either TTY or VDT to control what is accepted for the NEW MODE prompt. If you enter TTY, you can only enter TTY for NEW MODE. If you enter VDT, you can enter VDT or TTY for NEW MODE. *The default is VDT mode on the VDT.*

MTS

8.119.3 MTS Command Example

The following example shows how you can enter the MTS command to set the attributes of terminal ST07, and the List Terminal Status (LTS) command to confirm that the attributes were set accordingly:

```
[] MTS
MODIFY TERMINAL STATUS
 TERMINAL NAME:       ST07
 NEW STATUS (ON/OFF): ON
 NEW MODE (TTY/VDT): TTY
 LOGIN REQUIRED?: YES
 USER PRIVILEGE CODE:  2
 DEFAULT MODE (TTY/VDT):  VDT

[] LTS
LIST TERMINAL STATUS
 TERMINAL NAME:     ST07
 OUTPUT ACCESS NAME:    ME
 TERMINAL    USER     LI REQ'D   MODE    DEFAULT
   ST07     JKB044       YES       TTY       VDT
```

8.119.4 Commands Related to MTS

LTS (List Terminal Status)

8.120 MUI (MODIFY USER ID)

The MUI command allows you to modify the passcode and the user privilege code associated with your user ID. You cannot change the user ID itself. If a user is currently logged in at a different station under the user ID being modified, the user ID is not modified and the following error message is returned:

**** ERROR 9021 **** USER ID IN USE

You can modify your own user ID if you are not doing an activity in batch mode, and your privilege code is equal to the privilege level assigned to the MUI procedure.

8.120.1 MUI Command Format

MUI
MODIFY USER ID
USER ID: < name>
NEW PASSCODE: [string]
NEW USER PRIVILEGE CODE: [int]

8.120.2 MUI Command User Responses

System Prompts	Response Required or Optional	User Responses
USER ID:	R	Enter a six-character user ID.
NEW PASSCODE:	O	Enter a string of from one to six alphabetic or numeric characters that comprise the passcode to be associated with the new user ID. *The default is no change.*
NEW USER PRIVILEGE CODE:	O	Enter a number in the range of 0 through 7 that specifies the new privilege level at which the user can operate the DX10 operating system. *The default is no change.*

MUI

8.120.3 MUI Command Example

The following example shows how you can enter the List User ID (LUI) command to show the existing user ID, description, and privilege code assigned to a user. Then the example shows how you can enter the MUI command to change the user's ID, passcode, and privilege code. Another LUI command shows the user's changed status.

```
    [] LUI
    LIST USER IDS
     OUTPUT ACCESS NAME:  LP01
    []

USER ID     DESCRIPTION       PRIVILEGE CODE
   SYS000     SYSTEM                 7
   TCR001     TOM                    7
   CLN002     LARRY                  7
   TEC050     TECHPUBS               0<-

    [] MUI
    MODIFY USER ID
     USER ID: TEC050
     NEW PASSCODE:
     NEW USER PRIVILEGE CODE:  7

    [] LUI
    LIST USER IDS
     OUTPUT ACCESS NAME:  LP01

USER ID     DESCRIPTION       PRIVILEGE CODE
   SYS000     SYSTEM                 7
   TCR001     TOM                    7
   CLN002     LARRY                  7
   TEC050     TECHPUBS               7<-
```

8.120.4 Commands Related to MUI

AUI	(Assign User ID)
DUI	(Delete User ID)
LUI	(List User IDs)

8.121 MVI (MODIFY VOLUME INFORMATION)

You can change the DX10 disk volume information by using the MVI command. The MVI command has four command options: L (list), C (change), S (swap), and Q (quit). These options appear in the command action prompt COMMAND (L,C,S,Q)?. You should use the L option to list the current volume information. You should use the C option to change the information or the S option to swap information. To terminate MVI, you should use the Q option. The information under the control of MVI includes the following:

File	Designator
System image	S
Overlay	O
Program	P
Loader	L
WCS	W
Diagnostic	D
Volume name	V

You should study the designators closely since they are used in several of the MVI prompts.

When you change a volume name by using the MVI command, DX10 does not recognize the new volume name until you unload the disk (if installed) by using the UV command and then install it again by using the IV command. If you change the name of the system disk, DX10 does not recognize the change until you perform another initial program load (IPL). (That is, you still use the old volume name to access files until you use the IV command or perform an IPL.)

You can execute the MVI command interactively, or you can make a control file to use with the MVI command by placing control commands in a disk file.

8.121.1 MVI Command Format

```
MVI
MODIFY VOLUME INFORMATION
CONTROL ACCESS NAME: < acnm>
```

The following paragraphs describe how to execute the MVI command interactively.

The initial value for the CONTROL ACCESS NAME prompt is ME. If you accept this value, the MVI command prompts appear at your terminal in TTY mode.

After you respond to the CONTROL ACCESS NAME prompt, the MVI command prompts for the name of the disk drive with the following prompt:

```
DISK?: < name> @
```

The responses are in the format DSXX, where the names are in the range DS01 through DS99 or a synonym with a value of DSXX.

MVI

NOTE

MVI checks the syntax of all file names used; however, if the disk is not installed, file names cannot be verified. When the disk is installed, MVI checks the file volumeID.S$IMAGES to see if the system image selected is present for the next IPL. If the file or system image is not present, the MVI operation is performed but the following message is returned:

WARNINGSYSTEM IMAGE NOT VERIFIED

The MVI command then prompts for one of the command options as follows:

COMMAND (L,C,S,Q)?:

The possible command options are L (list), C (change), S (swap), or Q (quit). The following paragraphs describe each of these command options.

L Command Option. If you want to list those items under the control of MVI, you should enter L in response to the COMMAND (L,C,S,Q)? prompt and press RETURN. The following example shows the type of information the MVI command displays with the L command option:

	PRIMARY	SECONDARY	SELECT
SYSTEM IMAGE:	SYS03	SYS02	P
PROGRAM FILE:	S$PROGA		P
OVERLAY FILE:	S$OVLYA		P
LOADER FILE:	S$LOADER		P
WCS FILE:			S
DIAGNOSTIC:	————		N
VOLUME NAME:	REL35		

The PRIMARY column indicates the file name for the function at the left that will be loaded the next time an IPL is performed if the SELECT prompt value is P (primary). (For example, SYS03 will be loaded as the system image at the next IPL.) The SECONDARY column indicates the file name for the function at the left that will be loaded the next time an IPL is performed if the SELECT prompt value is S (secondary).

The following list defines the SELECT field codes that may be listed under the SELECT column:

Code	Meaning
P	Primary
S	Secondary
T	Test
N	No
Y	Yes

The SYSTEM IMAGE, PROGRAM FILE, OVERLAY FILE, LOADER FILE, and WCS FILE allow the select codes P, S, or T. The SELECT codes N and Y apply only to the DIAGNOSTIC function. The file names listed, and those that you enter in response to prompting messages, are actual file names and NOT pathnames. You must enter these file names in the form specified. The files are assumed to be at the volume directory level (that is, VOL1.S$PROGA). For example, you should enter the file VOL1.S$PROGA by the file name S$PROGA, without a period.

You can change any of the six names by entering a new value. For system volumes, you can select to load from the primary image the next time an IPL is performed by specifying P for SELECT or load from the secondary image next time an IPL is performed by specifying S. You can also load from the secondary image the next time an IPL is performed and then specify that the loader automatically revert (if the disk volume is not write protected) to the primary image on the following IPL by specifying T. The code T is used for testing a newly created image of DX10.

In order to test a new operating system after it has been generated, assembled, and link edited, you can execute the MVI command, specify as the secondary SYSTEM IMAGE, the name of the task image in S$IMAGES for the system you want to test, and specify T (test) as the SELECT for SYSTEM IMAGE. You can also accomplish this by using the TGS command. You should then perform another IPL to activate the new system. If the system is satisfactory, you can execute the MVI command, change the PRIMARY system image to the image name that was the test image or use the IGS command.

If you specify Y SELECT for the diagnostic file, then DIAGNOSTIC determines the file to be loaded rather than performing another IPL. As with the T SELECT, Y SELECT is a one-time function reverting to N after the IPL.

C Command Option. If you want to change any of the information that the MVI command displays, you should enter C in response to the COMMAND (L,C,S,Q)? prompt and press RETURN.

When you enter C, the following prompting message appears:

```
WHICH ITEM (S,O,P,L,D,V,W)?:
```

The WHICH ITEM (S,O,P,L,D,V,W)? prompt allows you to enter the character designating the information you want to change. Your response should be S, O, P, L, D, V, or W followed by RETURN. You should then specify the information in response to further prompting messages.

If you select the character S, O, P, L, or W, the system prompts for the following file information:

```
PRIMARY:
SECONDARY:
SELECT:
```

For each prompt, the current value is displayed following the colon. You can accept the displayed value by pressing RETURN. You can change the displayed value by typing in the desired new value and pressing RETURN. To delete a value, enter as many blanks as there are characters in the original value.

MVI

If you select the character D, the system prompts for the following information:

```
DIAGNOSTIC:
SELECT:
```

You can accept the current diagnostic and select values or change them. The same current value and change procedure described previously apply.

If you select V, the system prompts for the following information.

```
VOLUME NAME:
```

You can accept the current volume name or change it. The same current value and change procedure as described previously apply. Disk volume names must not be the same as disk device names (for example, DS01 or DS02).

NOTE

If you change the name of an installed volume, you must unload the volume under the old name and install it under the new name.

If a syntax error is found in any of the names you enter in response to the prompts, the following message appears and the MVI command terminates:

```
FILE NAME SYNTAX ERR
```

S Command Option. If you want to swap the primary and secondary names, you should enter S in response to the COMMAND (L,C,S,Q)? prompt. The same prompt that appears for the C command option then appears, as follows:

```
WHICH ITEM (S,O,P,L,W)?:
```

You can then swap the primary and secondary file names for the files that these characters designate by entering the character corresponding to the file.

Q Command Option. You can terminate MVI by responding to the COMMAND (L,C,S,Q)? prompt with Q and pressing RETURN.

Control File. If you want to make a control file to use with the MVI command, you should place the control commands in a disk file. You can name the file with any unique file name. The control commands are MVI prompt responses. You should enter them into the file (by using the Text Editor) exactly as you would enter them interactively. You should enter the pathname of this file in response to the CONTROL ACCESS NAME prompt. For example, the following file changes the volume name to Y on DS02. All lines must begin with the first character at the left margin.

```
DS02
C
V
VOL2
Q
```

8.121.2 MVI Command User Responses

System Prompts	Response Required or Optional	User Responses
CONTROL ACCESS NAME:	R	Enter the device name or the file pathname from which DX10 reads directives to control the modify volume information operation. The initial value ME is for an interactive session at the terminal where the MVI command was entered.
DISK:	R	Enter a name in the format DSxx, where the names are in the range DS01 through DS99, or enter a synonym with a value of DSxx.
COMMAND (L,C,S,Q):	R	Enter L, C, S, or Q where L means list, C means change, S means swap, and Q means quit. *The default is quit.*

8.121.3 MVI Command Example

The following MVI example shows how you can enter the MVI command to obtain a list of the information subject to change. The example also specifies a change in the diagnostic file and quits MVI. User action and system returns are documented to the right of the example.

How the Terminal Looks*	Explanation of Data on the Terminal
[ ] MVI	You enter the MVI command.
MODIFY VOLUME INFORMATION CONTROL ACCESS NAME: ME	SCI displays the command name and prompts you for the control access name. You can accept the default ME by pressing RETURN.

*User responses underlined.

MVI

How the Terminal Looks*	Explanation of Data on the Terminal
MVI DISK?: DS01	You type DS01 and press RETURN to change the volume information on DS01.
COMMAND (L,C,S,Q)?: L	You enter L and press RETURN to list the current content of volume information.

	PRIMARY	SECONDARY	SELECT
SYSTEM IMAGE:	RLSYS	SYS2	P
PROGRAM FILE:	S$PROGA		P
OVERLAY FILE:	S$OVLYA		P
LOADER FILE:	S$LOADER		P
WCS FILE:			P
DIAGNOSTIC:	JUNK	————	N
VOLUME NAME:	A19		

COMMAND (L,C,S,Q)?: C WHICH ITEM (S,O,P,L,D,V,W)?: D	You enter C and press RETURN to change the volume information. Then you enter D and press RETURN to change diagnostic information.
DIAGNOSTIC: JUNK JUNK2	The banner message for diagnostic appears. You then enter the change designation JUNK2.
SELECT: N Y	Then you enter Y for SELECT and press RETURN

	PRIMARY	SECONDARY	SELECT
SYSTEM IMAGE:	RLSYS	SYS2	P
PROGRAM FILE:	S$PROGA		P
OVERLAY FILE:	S$OVLYA		P
LOADER FILE:	S$LOADER		P
WCS FILE:			P
DIAGNOSTIC:	JUNK2	————	Y
VOLUME NAME:	A19		

COMMAND (L,C,S,Q)?: Q	You enter Q to terminate the MVI command.
MVI TERMINATED	MVI terminated.
[ ] Q	SCI terminated.

*User responses underlined.

8.121.4 Commands Related to MVI

IGS	(Install Generated System)
SVS	(Show Volume Status)
TGS	(Test Generated System)

MWR

8.122 MWR (MODIFY WORKSPACE REGISTERS)

The MWR command modifies specific workspace registers of a task. Refer to Volume III for a detailed description.

8.123 PB (PROCEED FROM BREAKPOINT)

The PB command assigns new breakpoints in the specified task. Refer to Volume III for a detailed description.

PF

8.124 PF (PRINT FILE)

The PF command places files in a waiting line (output queue) from which their contents are printed at a printing device. You can print sequential, sequential key indexed, and relative record files by using the PF command.

When printing a file in response to a PF command, DX10 automatically single spaces the lines of output unless the file contains carriage control characters. Files containing carriage control characters (except for ANSI format control characters) are automatically written in the format specified by the carriage control characters. Files containing ANSI format carriage control characters are single spaced when written unless you select ANSI format when entering the PF command.

NOTE

Characters with a value of less than > 20 are control characters. The PF command does not print these characters, but they do cause formatting problems in the printed output.

ANSI carriage control characters in the first position of each record have the following meanings:

blank	—	single space before printing
0	—	double space before printing
1	—	top of page before printing
+	—	suppress line space (overprint)

NOTE

The logical record length specified when the file was created is used as the maximum length of any record to be printed. The PF command truncates larger records. Any records larger than the largest record size handled by the print device are also truncated.

If the device is offline when you attempt the PF command, an error message is written to the system log.

8.124.1 PF Command Format

```
PF
PRINT FILE
FILE PATHNAME: < acnm>  @
ANSI FORMAT?: < yes/no>
LISTING DEVICE: < name>  @
DELETE AFTER PRINTING?: [yes/no]
NUMBER OF LINES/PAGE: [int]
```

8.124.2 PF Command User Responses

System Prompts	Response Required or Optional	User Responses
FILE PATHNAME:	R	Enter a pathname to identify the file whose contents are to be printed.
ANSI FORMAT?:	R	Enter Y (yes) to specify that the file being printed contains ANSI carriage control characters. Enter N (no) to specify that the file being printed does not contain ANSI control characters.
LISTING DEVICE:	R	Enter a device name to identify a device to which output should be printed. LP$x is not allowed as a listing device because it is not considered a device.
DELETE AFTER PRINTING?:	O	Enter Y (yes) to specify that the file being printed is to be deleted after it is printed. Enter N (no) to specify that it should not be deleted. *The default value is N.*
NUMBER OF LINES/PAGE:	O	Enter the number of lines to be printed on one page. *The default is 62.*

8.124.3 PF Command Example

The following example shows how you can enter a PF command to list the contents of a file at a line printer:

```
[] PF
PRINT FILE
 FILE PATHNAME(S):  .XYZ
 ANSI FORMAT?:  NO
 LISTING DEVICE:  LP02
 DELETE AFTER PRINTING?:  NO
 NUMBER OF LINES/PAGE:
```

After you respond to the last prompt, the queue status of the device specified is printed on your screen as follows:

```
USER    DEVICE  STATUS    FILE NAME
 SYSTEM  LP02  ACTIVE    VOL1.PRINT1
 SYSTEM  LP02  WAITING   .XYZ
```

8.124.4 Commands Related to PF

SF (Show File)

8.125 PGS (PATCH GENERATED SYSTEM)

After ALGS finishes executing, you should execute the PGS command to apply the patch file generated by ALGS to the new system. You cannot perform an initial program load (IPL) on a system unless you have executed the PGS command. The patches are in the form of an SCI batch command stream, and you apply them by issuing the PGS command.

The PGS command activates the batch command stream within the patch file created by ALGS. The SCI listing is written to the value you enter in response to the BATCH LISTING prompt. An example patch file is in Volume V. Volume V also contains information on PGS error recovery.

The output of PGS is a usable DX10 image and the batch SCI listing.

8.125.1 PGS Command Format

```
PGS
PATCH GENERATED SYSTEM
DATA DISK: [name]@
TARGET DISK: [name]@
SYSTEM NAME: < name> @
BATCH LISTING: [acnm]@
```

8.125.2 PGS Command User Responses

System Prompts	Response Required or Optional	User Responses
DATA DISK:	O	Enter the device name of the disk that contains all of the files output by XGEN and ALGS. *The default value is DS01.*
TARGET DISK:	O	Enter the device name of the disk that contains the system to be patched. This system is a task on the program file, .S$IMAGES, on the target disk. *The default value is DS01 or the value previously assigned as the target disk by a PGS, IGS, TGS, or ALGS command.*
SYSTEM NAME:	R	Enter a one- to five-character alpha-numeric string that is the name of the system that was generated. This is the name given in response to the XGEN OUTPUT prompt. *The default value is the system name previously assigned by a PGS, IGS, TGS, or ALGS command.*

System Prompts	Response Required or Optional	User Responses
BATCH LISTING:	O	Enter the access name of a sequential file or device to which the batch SCI listing is to be written. *The default value is the following:* *<target disk>.S$SYSGEN.<system name>.PGSLIST*

8.125.3 PGS Command Example

In the following example, a new system has been generated. The name specified for the system is SYS1. The PGS command patches the system image with the following prompt responses:

```
[ ] PGS
PATCH GENERATED SYSTEM
  DATA DISK:
  TARGET DISK:
  SYSTEM NAME: SYS1
  BATCH LISTING:
SYSTEM PATCH STREAM ERROR COUNT = 0:
BACKGROUND EXECUTION HAS COMPLETED
```

8.125.4 Commands Related to PGS

ALGS	(Assemble and Link Generated System)
IGS	(Install Generated System)
TGS	(Test Generated System)

Q

8.126 Q (QUIT)

To deactivate SCI at a terminal, you must enter the Quit (Q) command and press RETURN. Termination of SCI allows tasks to access the terminal as a physical device as well as free system resources used by the SCI tasks.

If the Text Editor or Debugger is active at the terminal, then the Q command automatically prompts for the Quit Edit (QE) or Quit Debug Mode (QD) command responses. Following the entry of responses to either the QE or QD commands, the following message appears:

```
QUIT PROCESSING INCOMPLETE: RE-ENTER 'Q' COMMAND
```

To terminate SCI at your terminal, press the RETURN key and then reenter the Q command.

If a background task is active at the terminal when you enter the Q command, the following message appears:

```
****ERROR 901E****CAN'T QUIT WITH BG TASK PENDING
```

You must allow the background task to complete, or you can terminate it by using the Kill Background Task (KBT) command. Then, you must reenter the Q command following the background task completion in order to terminate SCI at the terminal.

8.126.1 Q Command Format

```
Q
```

8.126.2 Q Command User Responses

No user responses are required.

8.126.3 Q Command Format

```
[ ] Q
```

8.127 QD (QUIT DEBUG)

The QD command takes a task out of debug (or controlled) mode. Refer to Volume III for a detailed description.

QE

8.128 QE (QUIT EDIT)

After you activate the Text Editor, the QE command terminates the Text Editor. Refer to Volume IV for a detailed description.

8.129 Q$SYN (CLEAR SECRET SYNONYMS)

When synonym table overflow errors occur, the synonym table has been filled. System-defined secret synonyms and user synonyms share the same table area. You can clear the system-defined secret synonyms by executing the Q$SYN command. You can execute the Show File (SF) command on the file .S$PROC.Q$SYN to show the synonyms cleared by the Q$SYN command.

8.129.1 Q$SYN Command Format

Q$SYN

8.129.2 Q$SYN User Responses

No user responses are required.

8.129.3 Q$SYN Command Example

The following example shows how you can use the Q$SYN command to clear synonyms for your terminal:

[ ] Q$SYN

RAL

8.130 RAL (RELEASE ALL LUNOs)

The RAL command releases all logical unit number (LUNO) assignments local to your station.

8.130.1 RAL Command Format

```
RAL
RELEASE ALL LOGICAL UNITS
ARE YOU SURE: < yes/no>
```

8.130.2 RAL Command User Responses

System Prompt	Response Required or Optional	User Response
ARE YOU SURE?:	R	Enter Y (yes) or N (no) to confirm or deny that you want a release of all station-local LUNOs.

8.130.3 RAL Command Example

The following example shows how you can enter the RAL command to release all devices from assignment to LUNOs local to your station:

```
[ ] RAL
RELEASE ALL LOGICAL UNITS
 ARE YOU SURE?: YES
```

8.130.4 Commands Related to RAL

AL (Assign LUNO)
RL (Release Station-Local LUNO)

8.131 RCD (RECOVER DISK)

The RCD utility restores the volume information to track 0, sector 0, of the disk from a copy that was stored on track 1 when the disk was initialized. You can only recover a disk built on a DX10 3.3 or later system, and byte >A2 on track 0, sector 0, of the disk must contain a 1.

If the volume information on track 0, sector 0 has been destroyed, you cannot install the disk and you cannot access its files. If you can install the disk using the Install Volume (IV) command, you should not use the RCD command.

If the track 1 copy of the volume information has been destroyed, you cannot recover the disk by using this utility.

You should execute the RCD command to a disk if a flash crash code of >1, >6, or >B occurs while performing an initial program load (IPL) from that disk. If an error code of >10 - >12, >15, >17, or >1B - >1C occurs while installing a disk with the IV command, you should load the disk as a secondary disk and perform a verify on track 0, sector 0 by using the Show Absolute Disk (SAD) command. You should verify that the data on track 0, sector 0 is the same as that on track 1, sector n-2 (where n is sectors per track). If errors occur in the verification or if the data on track 0, sector 0, is unreadable, you should recover the disk by using RCD.

You cannot perform the RCD command to the system disk (DS01). If you must recover the system disk, mount that disk in a disk drive other than DS01 and use a backup system disk.

RCD flags all ADUs on the disk as allocated. Therefore, you must copy the data files on the disk to another media after RCD is performed by using the Copy Directory (CD) or Backup Directory (BD) command. After all data files have been copied, you must reinitialize the recovered disk by using the Initialize Disk Surface (IDS) or the Initialize New Volume (INV) command in order to use that disk in the future. As a safeguard, you should not reinitialize the recovered disk immediately because you might need to recopy files from that disk.

After the RCD command has executed, you must install the disk using the Install Volume (IV) command to access any data files on that disk. After the RCD command has been performed and prior to copying the data files, you can execute the Check Disk for Consistency (CKD) command on the recovered disk. As a result, you will know beforehand if there are any bad directories or files that you cannot copy with the CD or BD commands. Because RCD flags all ADUs as allocated, the CKD dump of the disk's bit maps will contain lines showing the range of ADUs that are allocated but not used. Refer to the CKD command description for more information concerning the CKD command. You can use the expert mode of CKD to correct the bit map allocation and produce a usable disk instead of using CD.

8.131.1 RCD Command Format

```
RCD
RECOVER DISK
UNIT NAME: <acnm> @
```

RCD

8.131.2 RCD Command User Responses

System Prompt	Response Required or Optional	User Response
UNIT NAME:	R	Enter the unit name of the device other than DS01, containing the disk to be restored.

8.131.3 RCD Command Example

The following example shows how you can use the RCD command to recover a disk mounted in disk unit DS04:

```
[ ] RCD
RECOVER DISK
  UNIT NAME: DS04
```

8.132 RCRU (READ CONTENTS OF SPECIFIED CRU REGISTER)

The RCRU command reads the contents of the specified CRU register. This command will transfer 16 bits from the specified CRU offset. This output is in the form of four hexadecimal digits output to the specified access name.

8.132.1 RCRU Command Format

```
RCRU
READ CONTENTS OF SPECIFIED CRU REGISTER
CRU ADDRESS: < int>
OUTPUT ACCESS NAME: [acnm] @
```

8.132.2 RCRU Command User Responses

System Prompts	Response Required or Optional	User Responses
CRU ADDRESS:	R	Enter the CRU address offset from which to read 16 bits of information. (This value must be less than or equal to > 1FE0.)
OUTPUT ACCESS NAME:	O	Enter the device name or file name to which DX10 should write the CRU data produced by the read. *The default value is the terminal local file (TLF).*

8.132.3 RCRU Command Example

The following example shows how you can read 16 bits of information from a specific CRU address and display the data at your terminal:

```
[ ] RCRU
READ CONTENTS OF SPECIFIED CRU
  CRU ADDRESS: >0100
  OUTPUT ACCESS NAME:
  CRU REGISTER VALUE: >0DA0
```

8.132.4 Commands Related to RCRU

WCRU (Write Value to Specified CRU Address)

RD

8.133 RD (RESTORE DIRECTORY)

The RD command restores a set of files from the backup file written by the Backup Directory (BD) command. The backup file may have been written on one or more disk or magnetic tape volumes. As in the BD command, entering one or more disk device names as the input provides restoration of more than one disk without your intervention. The volumes, which all have the same name and contain a file by the same name, must be supplied to the RD command in the order in which they were written by the BD command.

The RD command does not restore the directory named S$PRINT and the files named .S$CRASH, .S$DIAG, and .S$ROLLA unless you specify them in INCLUDE statements.

If a hardware error was encountered on the source disk while backing up a file, that file was flagged as not restorable. A file so flagged is not restored, and a message informing you of this condition is written to the listing file. Any files not restored due to this not-restorable condition are listed in a bad file summary on the listing.

8.133.1 RD Command Format

```
RD
RESTORE DIRECTORY
SEQUENTIAL ACCESS NAME: [(acnm)] @
DIRECTORY PATHNAME: [acnm] @
CONTROL ACCESS NAME: [acnm] @
LISTING ACCESS NAME: [acnm] @
OPTIONS: [(string)] @
EXECUTION MODE (F,B): < F/B>
```

8.133.2 RD Command User Responses

System Prompts	Response Required or Optional	User Responses
SEQUENTIAL ACCESS NAME:	O	Enter the device name(s) or the pathname of the sequential file from which to restore information to the directory. Information must have been previously written to this device or file by the BD command. When you reply with DSxx, this option displays the following prompt: PATHNAME OF BACKUP FILE PATHNAME: < acnm> @ The pathname consists of a volume name and the name of a file. The pathname that you specify must be the pathname used in the backup operation (BD command).

System Prompts	Response Required or Optional	User Responses
DIRECTORY PATHNAME:	O	Enter a pathname identifying the directory or file to which information should be restored.
CONTROL ACCESS NAME:	O	Enter the device name or file pathname from which DX10 reads directives to control the restore operation. If you used a control file in the backup, you may have to use a control file in the restore. (For more information on using control files and the directives that you can include in a control stream, refer to the paragraph on control files in Section 7.) *If you do not enter a control name, you must specify both a directory pathname and a sequential access name.*
LISTING ACCESS NAME:	O	Enter the device name or the file pathname to which DX10 can write a summary of the results of a restore operation. *The default value is the terminal local file (TLF).*
OPTIONS:	O	You can enter the following options to control the restore operation. The ADD, ALIASES, NODATE, NOREWIND, NOUNLOAD, CMP, and SYSFILE are the default options.

Option	Purpose
ADD	Entering ADD specifies that files or aliases are copied only if a file or alias with the same name does not exist in the output set of files.
REPLACE	Entering REPLACE specifies that files are restored even if a file with the same name exists in the output set of files unless the files are write or delete protected. The existing file or files are deleted before the source file or files are copied. Aliases are not replaced.

RD

System Prompts	Response Required or Optional	User Responses
	Option	Purpose
	ALIASES	Entering ALIASES specifies that all aliases associated with the files are also to be restored.
	NOALIASES	Entering NOALIASES specifies that no aliases are to be restored.
	DATE	Entering DATE limits the restore to those files created or updated on or after the specified date. The date and time (year, month, day, hour, minute) of the oldest file are requested: YEAR: <int> MONTH: <int> DAY: <int> HOUR: [int] MINUTE: [int]
	NODATE	Entering NODATE indicates no date constraints on the restore.
	REWIND	Entering REWIND causes rewinding of tape volumes before the directory is restored.
	NOREWIND	Entering NOREWIND indicates tape volumes are not rewound before the directory is restored.
	UNLOAD	Entering UNLOAD indicates tape volumes are unloaded after use.
	NOUNLOAD	Entering NOUNLOAD indicates tape volumes are not unloaded after use.
	CMP	Entering CMP indicates that unused physical blocks at the end of KIFs are truncated and released back to the disk. Both CMP and NOCMP collect secondary allocations together.

System Prompts	Response Required or Optional	User Responses
	Option	Purpose
	NOCMP	Entering NOCMP indicates that unused physical blocks at the end of KIFs are not truncated. When restoring a KIF, you may want to retain the original space allocation since compression resets the primary allocation space to the current size of the file, thereby increasing access time to a KIF.
	SYSFILE	Entering SYSFILE indicates that you want system files included in the restore operation. These files are .SCI990, .S$SYSLIB, .S$ERROR, .S$CNTRY, .S$SDS$, .S$OVLYA, .S$PROC, .S$IMAGES, .S$TCALIB, .S$FGTCA, .S$BGTCA, .S$SLG1, .S$SLG2, .S$NEWS, .S$LOADER, and .PATCH. The SYSFILE and NOSYSFILE options only apply when using the system disk.
	NOSYSFILE	Entering NOSYSFILE indicates that you do not want system files included in the restore operation.
EXECUTION MODE (F,B):	R	Enter F (foreground) to execute the task in foreground mode. Enter B (background) to execute the task in background mode. *The initial value is FOREGROUND.*

RD

8.133.3 RD Command Examples

The following examples show how you can enter the RD command to restore a directory backed up by the BD command. The first example restores the directory backed up to DS04 by one of the BD command examples given in the BD command description.

```
  RESTORE DIRECTORY        11:19:30 FRIDAY, APRIL 30, 1982.

ORIGINAL SOURCE:         DS04
ORIGINAL DESTINATION:    PAE29.HW.NEWDIR
ORIGINAL OPTIONS:        ADD, ALIASES, NODATE, NOREWIND, NOUNLOAD, NOFAST, SYSFILE
CONTROL FILE:
LIST FILE:               PAE29.HW.RDDISK


  **       DIRECTORY  PAE29.HW.NEWDIR
  **  FILE2
  **  FILE1
  **       END OF DIRECTORY  PAE29.HW.NEWDIR


ELAPSED TIME = 0 MINUTES     5 SECONDS
SIZE OF INPUT = 0 ADU'S

****** RESTORE DIRECTORY COMPLETED
```

The second example shows a restore operation that uses MOVE directives. The control file for this operation appears first.

```
OPT REP
MOV ,PAE29.HW.DIR
MOV ,PAE29.HW.DIR.SUBDIR
END


  RESTORE DIRECTORY        11:15:49 FRIDAY, APRIL 30, 1982.

ORIGINAL SOURCE:         PAE29.HW.SEQ
ORIGINAL DESTINATION:
ORIGINAL OPTIONS:        ADD, ALIASES, NODATE, NOREWIND, NOUNLOAD, NOFAST, SYSFILE
CONTROL FILE:            ST01
LIST FILE:               PAE29.HW.RDOPT

OPT REP
MOV ,PAE29.HW.DIR

  **       DIRECTORY  PAE29.HW.DIR
  **  FILE1     WAS REPLACED
  **       END OF DIRECTORY  PAE29.HW.DIR
```

```
MOV ,PAE29.HW.DIR.SUBDIR

  **       DIRECTORY  PAE29.HW.DIR.SUBDIR
  **  PFILE    - PROGRAM FILE

  **  TASK       ID
  **  FUTIL     >0B  MODULE DELETE PROTECTED

  **  OVERLAY    ID
  **  FUTOV0    >60  MODULE DELETE PROTECTED
  **  FUTOV1    >61  MODULE DELETE PROTECTED
  **  FUTOV2    >62  MODULE DELETE PROTECTED
  **  FUTOV3    >63  MODULE DELETE PROTECTED
  **  FUTOV4    >64  MODULE DELETE PROTECTED
  **  FUTOV5    >65  MODULE DELETE PROTECTED
  **  FUTOV6    >66  MODULE DELETE PROTECTED
  **  FUTOV7    >67  MODULE DELETE PROTECTED
  **  FUTOV8    >68  MODULE DELETE PROTECTED
  **  FUTOV9    >69  MODULE DELETE PROTECTED
  **  FUTOVA    >6A  MODULE DELETE PROTECTED
  **  FUTOVB    >6B  MODULE DELETE PROTECTED
  **       END OF PROGRAM FILE

  **       END OF DIRECTORY  PAE29.HW.DIR.SUBDIR

END

ELAPSED TIME = 0 MINUTES   31 SECONDS
SIZE OF INPUT = 53 ADU'S
13 WARNINGS ISSUED

****** RESTORE DIRECTORY COMPLETED
```

8.133.4 Commands Related to RD

BD	(Backup Directory)
VB	(Verify Backup)

RE

8.134 RE (RECOVER EDIT)

The RE command recovers text edits after system crashes or power failures. Refer to Volume IV for a detailed description.

8.135 RGL (RELEASE GLOBAL LUNO)

The RGL command releases a device or file from a global logical unit number (LUNO) assignment. Global LUNOs actively performing I/O operations cannot be released. DO NOT release global LUNO assignments reserved for the system (see Appendix B).

8.135.1 RGL Command Format

```
RGL
RELEASE GLOBAL LUNO
LUNO: < int>
```

8.135.2 RGL Command User Responses

System Prompt	Response Required or Optional	User Response
LUNO:	R	Enter the LUNO from which to release the global assignment of the device or file.

8.135.3 RGL Command Example

The following example shows how you can enter an RGL command to release a device or file from a global assignment to a LUNO (> 85):

```
[ ] RGL
RELEASE GLOBAL LUNO
  LUNO: > 85
```

8.135.4 Commands Related to RGL

AGL (Assign Global LUNO)

RL

8.136 RL (RELEASE STATION-LOCAL LUNO)

The RL command releases a device from a logical unit number (LUNO) assignment that is local to your station. A LUNO actively performing I/O operations cannot be released.

8.136.1 RL Command Format

```
RL
RELEASE LUNO
LUNO: <int>
```

8.136.2 RL Command User Responses

System Prompt	Response Required or Optional	User Response
LUNO:	R	Enter the LUNO from which to release the station-local assignment of the device or file.

8.136.3 RL Command Example

The following example shows how you can enter an RL command to release a device from assignment to station-local LUNO > 85:

```
[ ] RL
RELEASE LUNO
  LUNO: > 85
```

8.136.4 Commands Related to RL

AL	(Assign LUNO)
RAL	(Release All LUNOs)

8.137 RO (RESUME OUTPUT AT DEVICE)

The RO command resumes queued output that is temporarily halted at a device by an HO command or reenables a device for later output after a KO command has terminated the output of all files in the output queue. After you enter the RO command, output continues from the point at which you halted it.

8.137.1 RO Command Format

```
RO
RESUME OUTPUT AT DEVICE
DEVICE NAME: <acnm> @
```

8.137.2 RO Command User Responses

System Prompt	Response Required or Optional	User Response
DEVICE NAME:	R	Enter the device name of the device at which queued output is to resume or be reenabled.

8.137.3 RO Command Example

The following example shows how you can enter the RO command to resume queued output that is temporarily halted at a line printer.

```
[ ] RO
RESUME OUTPUT AT DEVICE
  DEVICE NAME: LP01
```

8.137.4 Commands Related to RO

HO	(Halt Output at Device)
KO	(Kill Output at Device)
SOS	(Show Output Status)

RS

8.138 RS (REPLACE STRING)

After you activate the Text Editor, the RS command replaces a specified number of occurrences of a string of characters with a new character string, beginning with the current line. Refer to Volume IV for a detailed description.

8.139 RST (RESUME SIMULATED TASK)

The RST command resumes task simulation following a breakpoint, a simulated breakpoint, or simulation of a specified number of instructions (time-out). Refer to Volume III for a detailed description.

RT

8.140 RT (RESUME TASK)

The RT command activates a task at the point at which it was suspended. Refer to Volume III for a detailed description.

8.141 RWL (REWIND LUNO)

The RWL command rewinds a sequential file or a rewindable device. Rewinding a file or device moves the current data-access position of DX10 to the first record of that file or device. DX10 ignores entry of the RWL command if the device or file cannot be rewound.

8.141.1 RWL Command Format

```
RWL
REWIND LUNO
LUNO: < int>
```

8.141.2 RWL Command User Responses

System Prompts	Response Required or Optional	User Responses
LUNO:	R	The logical unit number (LUNO) of the file or device to be rewound.

8.141.3 RWL Command Example

The following example shows how you can enter the RWL command to rewind a file assigned to LUNO > 86:

```
[ ] RWL
REWIND LUNO
  LUNO: > 86
```

8.141.4 Commands Related to RWL

BL	(Backspace LUNO)
FL	(Forward Space LUNO)
WEOF	(Write EOF to LUNO)

SAD

8.142 SAD (SHOW ABSOLUTE DISK)

The SAD command prints the contents of a specified absolute address on a disk. Refer to Volume III for a detailed description.

8.143 SADU (SHOW ALLOCATABLE DISK UNIT)

The SADU command outputs the contents of the specified ADU to the specified device. Refer to Volume III for a detailed description.

SBS

8.144 SBS (SHOW BACKGROUND STATUS)

You enter the SBS command to view the status of a program that is currently executing in background mode and that was activated from your terminal. When the background is active, the SBS command produces the following status messages, which reflect task state codes:

ACTIVE
AWAITING ANY I/O COMPLETION
AWAITING ASSIGNMENT OF I/O DEVICE
AWAITING COMPLETION OF I/O
AWAITING COROUTINE ACTIVATION
AWAITING DISK FILE UTILITY SERVICES
AWAITING FILE MANAGEMENT SERVICES
AWAITING GETMEM REQUEST
AWAITING INITIAL LOAD
AWAITING TASK DRIVEN SVC PROCESSOR
AWAITING TERMINATION
ROLLING OUT
SUSPENDED
SUSPENDED FOR INITIATE I/O THRESHOLD
SUSPENDED FOR TIME DELAY

Any one of the following status messages indicate that the background is not active:

NO BACKGROUND ACTIVITY AT THIS TERMINAL
TERMINATED
WAITING FOR EXECUTION TO BEGIN

You can also use SBS to display the background terminal local file (TLF). SBS displays the TLF when the background task has terminated and the TLF has been written into by the background task. (When SCI is in batch mode, the background TLF is written to the listing file or device when each background command completes so SBS is not used in an SCI batch file.)

The background TLF is handled in such a way that output written to the TLF is not lost if you forget to do an SBS between two background task executions. However, this can cause confusion if the background TLF is not shown between background tasks. For example, execution of a background task that produced an error message will have the error in the background TLF. If you correct the error and the task runs again without an error, issuing the WAIT or SBS command automatically displays the background TLF, and the error from the first execution is still in the first records of the file.

8.144.1 SBS Command Format

SBS

8.144.2 SBS Command User Responses

System Prompt	Response Required or Optional	User Response
STATE SYNONYM	O	STATE SYNONYM is a nonprompted keyword that you can specify only within a command procedure or when expert mode is used. The value of the STATE SYNONYM prompt is the name of a synonym. If you specify a synonym name for this keyword, the task state of the background task is assigned to this name as a synonym value. (Refer to Volume III for information on task state codes.) If no background task is in process, the name is assigned the value >FF. You can use this within an interactive procedure to see if a background task is already active before executing a command that runs in background. An attempt to execute a background task when one is already active causes SCI to report an error and terminate the interactive procedure and all procedures that it has called to reach the procedure that attempted to execute the background task.

8.144.3 SBS Command Example

The following example shows how you can enter the Execute Batch (XB) command to execute a batch command stream and how you can enter successive SBS commands to view the status of the background program at different stages of execution:

```
[ ] SBS
SHOW BACKGROUND STATUS
NO BACKGROUND ACTIVITY AT THIS TERMINAL:

[ ] XB
EXECUTE BATCH
  INPUT ACCESS NAME: MY.BATCH
  LISTING ACCESS NAME: MY.LIST

[ ] SBS
SHOW BACKGROUND STATUS
AWAITING FILE MANAGEMENT SERVICES:
```

SBS

```
[ ] SBS
SHOW BACKGROUND STATUS
ACTIVE:

[ ] SBS
SHOW BACKGROUND STATUS
AWAITING COROUTINE ACTIVATION:
```

The following SBS command specifies A as the name of the synonym that is to receive as its value the status of the background task (for example, > 17, which means awaiting coroutine activation):

```
SBS STATE SYNONYM = A
```

8.144.4 Commands Related to SBS

KBT	(Kill Background Task)
WAIT	(Wait for Background Task to Complete)
XT	(Execute Task)

8.145 SCC (SHOW COUNTRY CODE)

The SCC command lists the country code selected when the operating system is generated. Refer to the MCC command for a list of valid country codes. The response to the LISTING ACCESS NAME prompt must be a legal device or file name.

8.145.1 SCC Command Format

```
SHOW COUNTRY CODE
LISTING ACCESS NAME: [acnm] @
```

8.145.2 SCC Command User Responses

System Prompt	Response Required or Optional	User Response
LISTING ACCESS NAME:	R	Enter the device or file name in which you want to list the current country code. If you enter a null response, the country code is listed to the terminal local file (TLF).

8.145.3 SCC Command Example

In the following example, the SCC command lists the country code US (United States) to the TLF.

```
[ ] SCC
SHOW COUNTRY CODE
  LISTING ACCESS NAME:
COUNTRY = USA
```

8.145.4 Commands Related to SCC

MCC (Modify Country Code)

SD

8.146 SD (SCAN DISK)

The SD command reads a disk, checking for errors that may indicate the presence of scratches, flaws, parity errors, and other errors. This command scans the disk, then produces a report of the locations at which an error status was returned. The report lists the error locations in terms of physical characteristics of the disk as well as in the allocation units that the software uses. If a file uses the error location, the report also gives the pathname of the file. The report may also be useful in determining if any files need to be reconstructed. Bad track information on the disk is not changed, and dual allocation problems are not analyzed. You can enter the information with the Install New Volume (INV) command to update the bad track information for a volume. You can invoke the SD command for a disk whether or not the volume is installed.

8.146.1 SD Command Format

```
SD
SCAN DISK
UNIT NAME: <name> @
OUTPUT FILE NAME: <acnm> @
```

8.146.2 SD Command User Responses

System Prompts	Response Required or Optional	User Responses
UNIT NAME:	R	Enter the device name of the disk unit in which the disk you want to scan is mounted.
OUTPUT FILE NAME:	R	Enter the access name of the file to which the bad sector information is written.
AUTOCORRECT	O	The AUTOCORRECT prompt is a hidden prompt that you can use in expert mode. If you specify YES for this prompt, SD reads and rewrites tracks in an attempt to clear the error condition. The default value is NO.

WARNING

This feature is potentially dangerous. Any file creations or deletions to the disk under autocorrect can introduce more errors rather than correcting all errors. Use this feature with extreme caution or further loss of data may result. Also, this feature does not guarantee that data is correct in sectors in which an error has been cleared. Any affected file should be processed to detect any structure or data errors, and the file should be rebuilt.

8.146.3 SD Command Example

The following example shows a disk scan performed on the volume mounted in disk unit DS02. The output is written to the file .REPORT. The disk is a diskette that has four bad sectors.

```
[] SD
SCAN DISK
 UNIT NAME: DS02
 OUTPUT FILE NAME: .REPORT


SCAN DISK     VOLUME NAME:FLOPPY      06/10/82     10:49:04

 BAD TRACK SUMMARY

  PHYSICAL DATA              ALLOCATION DATA

   SECTOR  HEAD    CYL       ADU   SECTOR  FILE NAME
        0     1     24      1907        0  DS02.FILE1
        0     1     26      2404        0  DS02.FILE1
        0     1     27      2453        0  DS02.PROGA
        0     1     29      2609        0  NO FILE ASSOCIATED
```

8.146.4 Commands Related to SD

CKD (Check Disk for Consistency)
RCD (Recover Disk)

SDT

8.147 SDT (SHOW DATE AND TIME)

The SDT command displays or prints the current date and time at a terminal. The time is given according to the 24-hour clock with the hour given in the range 00 to 23. For example, 2:13 p.m. is shown as 14:13 and 2:13 a.m. as 02:13.

8.147.1 SDT Command Format

SDT

8.147.2 SDT Command User Responses

No user responses are required.

8.147.3 SDT Command Example

The following example shows how you can enter the SDT command to obtain the current system date and time:

```
[ ] SDT
13:34:10 SATURDAY, FEB 05, 1977.:
```

NOTE

You can modify the SDT procedure to have the date and time returned as the value of a synonym rather than a displayed message. To do this, specify the synonym as the second argument in the PARMS keyword of the SDT procedure, as follows:

PARMS = (28,synonym)

8.147.4 Commands Related to SDT

IDT (Initialize Date and Time)

8.148 SEM (SHOW DX10 ERROR MESSAGE)

The SEM command displays an explanation of a system error code. You cannot use this command with hard-copy terminals (teleprinter devices); therefore, you should invoke the command only from a 24-line VDT.

The SEM command is provided as a convenient tool. Its message file may not contain all error messages that the system reports. You can use the Add DX10 Error Message (AEM) command to add messages and change message descriptions. However, the message file may be changed in subsequent releases of DX10; consequently, any changes you make might not be included when the system is upgraded to a new release.

To show the expanded explanation of an error message, enter SEM and then press the RETURN key in response to the SCI prompt symbol ([ ]). The message SHOW DX10 ERROR MESSAGE appears and then the following appears:

```
DECODE ERROR CODES

PLEASE ENTER THE 4 CHARACTER ERROR CODE: _____

    ERRORS IN THE DX10 SYSTEM ARE OF THE FORM:

        CCEE ERROR 00EE FOR SVC CC
        PPQQ PARAMETER PP, SUB PARAMETER QQ
        XXEE ERROR 00EE FOR I/O OPCODE XX
        EEYY ERROR 00EE FOR LUNO YY

    IF AN EXPLANATION FOR ERROR XXXX IS VAGUE
    OR NOT FOUND, TRY DISPLAYING ERROR 00EE

    FOR SYSTEM CRASH ERRORS ENTER: Xxxx
    FOR SYSTEM FLASH CRASH ERRORS ENTER: FXxx
    xx, xxx = THE TWO OR THREE DIGIT CRASH CODE
```

You should enter the four-character error code for which you desire an expanded explanation (for example, 0F01). (You should not use the RETURN key when entering the error code.) The appropriate explanation appears. For example, if you enter 0F01 as the error code, the following appears. (Due to the line length of this document, the explanation and action text shown here are slightly different from that shown on the actual display.)

```
ERROR CODE 0F01 TYPE OF ERROR: ERROR ON "BID TASK"
THE ERROR BELOW OCCURRED ON A BID TASK SVC.
THE ERROR MESSAGE IS:                    CAUSE OF ERROR: USER
ILLEGAL STATION NUMBER
```

SEM

Explanation	Action
The specified station number does not exist in the system.	Check specified station number. Ensure that the active system has been generated with the station number desired.

HIT RETURN WHEN YOU ARE DONE WITH THIS SCREEN

After viewing the display, press the RETURN or CMD key. The system then displays the main menu.

8.148.1 Commands Related to SEM

AEM (Add DX10 Error Message)

8.149 SF (SHOW FILE)

The SF command displays the contents of a file at your VDT or prints the contents at your data terminal. On VDTs in VDT mode, you see only the first part of the file. To display the rest of the file, you must press the appropriate function keys on the keyboard as described in Table 8-12. You cannot enter another command while the contents of a file are being displayed. You can use the SF command to display the following types of files: sequential files, key indexed files (KIFs), and relative record files.

Table 8-12. Show File Control Keys (VDT Mode Only)

Key	Description
F1	Displays the next portion of a file (the amount displayed depends on the type of terminal). When you reach the end-of-file, further pressing of the F1 key has no effect.
F2	Displays the previous portion of a file. If the beginning of the file is displayed, pressing the F2 key has no effect.
F3	Scrolls the display horizontally 10 columns to the left. Scrolling can continue through column 160 without line numbers displayed or 170 with line numbers displayed. The bottom line of the VDT screen displays a scaling line, unless the display is positioned in column 1. The last character that can be shown is in character position 240 of the input record.
F4	Scrolls the display horizontally 10 columns to the right. The bottom line of the VDT screen displays a scaling line, unless the display is positioned in column 1.
F5	Performs no function while displaying a file. The error message ERROR, COMMAND OR SCROLL FUNCTION REQUIRED: appears at the bottom of a VDT screen.
F6	Displays line numbers in the leftmost columns of the VDT screen. Pressing the F6 key again removes the line numbers from the screen.
F7	Performs no function while displaying a file. The error message ERROR, COMMAND OR SCROLL FUNCTION REQUIRED: appears at the bottom of a VDT screen.
F8	Performs no function while displaying a file. The error message ERROR, COMMAND OR SCROLL FUNCTION REQUIRED: appears at the bottom of a VDT screen.
——→ FIELD	Scrolls the display horizontally 1 column to the right. The bottom of the VDT screen displays a scaling line unless the display is positioned in column 1.

SF

Table 8-12. Show File Control Keys (VDT Mode Only) (Continued)

Key	Description
FIELD ←	Scrolls the display horizontally 1 column to the left. Scrolling can continue through column 160 without line numbers displayed and 170 with line numbers diplayed. The bottom line of the VDT screen displays a scaling line unless the display is positioned in column 1.
Up arrow	Displays the next line of a file at the bottom of the screen and moves all other displayed lines up one position.
Down arrow	Displays the previous line of a file at the top of the display screen and moves all other displayed lines down one position. If the first line of the file is displayed at the top of the screen, the down arrow key has no effect.
+ Integer then RETURN	Entering a plus sign (+) and integer (decimal or hexadecimal) and then pressing the RETURN key positions the display of the file contents ahead the number of lines specified by the integer.
Integer then RETURN	Entering an integer and pressing the key positions the display of the file contents to the line specified by the integer.
– Integer then RETURN	Entering a minus sign (–), an integer (decimal or hexadecimal), and then pressing the RETURN key positions the display of the file contents back the number of lines specified by the integer.
CMD (command)	Terminates the display of a file and allows you to enter another command.

8.149.1 SF Command Format

```
SF
SHOW FILE
FILE PATHNAME: < acnm> @
```

8.149.2 SF Command User Responses

System Prompt	Response Required or Optional	User Response
FILE PATHNAME:	R	Enter the pathname that identifies the file you want to display.

8.149.3 SF Command Example

The following example shows how you can enter an SF command to list the contents of a file at a data terminal:

```
[ ] SF
SHOW FILE
  FILE PATHNAME: VOL1.DIRA.FILE1
```

8.149.4 Commands Related to SF

PF (Print File)

SIR

8.150 SIR (SHOW INTERNAL REGISTERS)

The SIR command displays the contents of the internal registers (program counter (PC), workspace pointer (WP), and status register (ST)) of a task, along with the task state. Refer to Volume III for a detailed description.

8.151 SIS (SHOW I/O STATUS)

The SIS command allows you to view the I/O status of global, station-local, and task-local logical unit numbers (LUNOs). You can view the status of one LUNO or of all LUNOs of a type (global, station-local, or task-local). You can also view the status of all LUNOs usable by tasks at a given station including global LUNOs, station-local LUNOs, and task-local LUNOs for all tasks associated with that station. You can also view every LUNO assignment in the whole system. Table 8-13 describes the headings under which the status information can be displayed.

8.151.1 SIS Command Format

```
SIS
SHOW I/O STATUS
LUNO: [int]
TYPE (TASK, STATION, GLOBAL, A): <{string}>
TASK RUN ID: <int>   [message seen only if type specified is TASK]
```

8.151.2 SIS Command Responses

System Prompts	Response Required or Optional	User Responses
LUNO:	O	Enter the LUNO whose status is being requested. *If you do not enter a LUNO, all LUNOs of the type specified are displayed, unless you select type A.*
TYPE (TASK, STATION, GLOBAL, A):	R	Enter a character string (ALL, GLOBAL, STATION, or TASK) to identify the type of LUNO whose status is being requested. You can abbreviate the character strings by entering T, S, or G for TASK, STATION, or GLOBAL, respectively. A response of A displays the following prompt: WHOLE SYSTEM?: The default of NO for this prompt lists all task-local LUNOs for all tasks associated with your station, then lists all station-local LUNOs, followed by all global LUNOs. Entering YES lists every LUNO assignment in the system. First, all task-local LUNOs for tasks not associated with a station are listed. Then, for each station, the task-local LUNOs for all tasks associated with the station are listed, followed by all station-local LUNOs. Finally, all global LUNOs are listed.

SIS

System Prompts	Response Required or Optional	User Responses
		A response of T displays the following prompt: TASK RUN ID: See the following description.
TASK RUN ID:	R	Enter a task ID only if you are requesting the status of task-local LUNOs. The ID entered must be the ID assigned by the system when the task was placed in execution (run-time ID).

Table 8-13. Information the SIS Command Displays

Heading	Information Displayed
TYPE & LUN	The type of LUNO is listed under the ampersand (&): T for task local, S for station local, and G for global. Under LUN is the LUNO whose status is requested or all LUNOs of the type whose status is requested.
DEV. NAME	The device name assigned to the LUNO. (If a file is assigned to the LUNO, the device name is the name of the device on which the file resides.)
TSK ID	If a task-local LUNO is displayed, TSK ID is the run-time ID of the task that assigned it. If a station-local LUNO or global LUNO is assigned to a file or file-oriented device and the LUNO is open, TSK ID is the run-time ID of the opening task. Otherwise, TSK ID is blank.
ST NO	The station number under which all station-local LUNOs are assigned. If a task ID is displayed, ST NO displays the station with which that task is associated or ST NO is blank if the task is not associated with any station.
OPN ACC	If the LUNO is not open, an N (no) appears here. If the LUNO is open but is not assigned to a file, a Y (yes) appears here. If the LUNO is open to a disk file, one of the following appears which designates the access privileges that were used to open the file: • RO (Read only) • EW (Exclusive write) • SH (Shared) • EA (Exclusive all) Refer to Volume III for the meaning of access privileges.

Table 8-13. Information the SIS Command Displays (Continued)

Heading	Information Displayed
FILE TYPE	One of the following file types is listed for files assigned to a LUNO: • SEQ (Sequential record file) • REL (Relative record file) • PGM (Program file) • IMG (Image file) • DEV (Device, not a file) • KEY (Key indexed file) • DIR (Directory file)
FILE NAME	The pathname identifying a file to which a LUNO is assigned.

8.151.3 SIS Command Example

The following example shows how you can request the status of all devices and files assigned to LUNOs and available to tasks at your station:

```
[] SIS
SHOW I/O STATUS
 LUNO:
 TYPE (TASK,STATION,GLOBAL,A): A

SHOW WHOLE SYSTEM I/O STATUS
 WHOLE SYSTEM?: NO

TYPE  DEV.  TSK ST OPN FILE FILE
& LUN NAME   ID NO ACC TYPE NAME
T 04  DS01   6B  1 RO  SEQ. PAES.S$PROC.SIS
S 00  ST01        N  DEV.
G 09  DS01        N  DEV.
G 07  DS01        N  PGM. PAES.DORIS.PROGB
G 14  ST01        N  DEV.
G 15  ST01        N  DEV.
G 50  ST04        N  DEV.
G 03  DS01        N  REL. PAES.S$TCALIB
G 02  DS01        N  REL. PAES.S$BGTCA
G 01  DS01        N  REL. PAES.S$FGTCA
G 10  DS01        N  PGM. PAES.S$SDS$
G 02  DS02        N  DEV.
G 03  DS03        N  DEV.
G 00  ST01        N  DEV.
G 0A  DS01        N  REL. PAES.S$OVLYA
G 0D  DS01        N  PGM. PAES.S$PROGA
G 0B  DS01        N  PGM. PAES.S$PROGA
G 0F  DS01        N  PGM. PAES.S$PROGA
G 0E  DS01   00   N  IMG. PAES.S$ROLLA
```

SIS

NOTE

Station numbers appear only on the LUNO status line for a LUNO associated with a station or task associated with a station.

8.151.4 Commands Related to SIS

SOS (Show Output Status)

8.152 SL (SHOW LINE)

After you activate the Text Editor, the SL command positions the current file to any specified line (record). Refer to Volume IV for a detailed description.

SMM

8.153 SMM (SHOW MEMORY MAP)

The SMM command displays the allocation of the main memory. You can only use this command on a 24-line VDT. The SMM command monitors memory usage by system and user tasks.

8.153.1 SMM Command Format

SMM
SHOW MEMORY MAP

8.153.2 SMM Command User Responses

No user responses are required.

8.153.3 SMM Command Example

Figure 8-10 shows an example of the entry of the SMM command.

The memory size indicated in the upper left portion of the display is the total amount of memory installed on the system, expressed in K bytes (one K byte equals 1024 bytes). The static memory is the amount of memory used for the operating system root, plus any memory-resident procedures or memory-resident tasks. This unit is also shown in K bytes.

The second line of the display provides information about the memory that is currently available for allocation. The # OF HOLES (number of holes) is the number of separate pieces of memory that are available for allocation. The LARGEST BLOCK is the size of the largest contiguous piece of available memory. The sum of all available pieces of memory is shown in the right portion of the second line (AVAILABLE).

The third line of the display provides usage statistics for the system disk (on the left portion) and the CPU (on the right portion). These statistics are updated every four seconds and indicate the percentage of time that the system disk and the CPU were busy during the preceding four-second time interval. These values are 0 if the response was NO to the PANEL DISPLAY(BAR CHART)? prompt during system generation. A small portion of 990 CPU time is used when the SMM display is in execution.

The remaining portion of the display gives a pictorial representation of memory, with the lowest address of memory at the top and the highest address at the bottom. This portion of the display always has 16 lines; each line has 64 graphic character positions. The amount of memory represented by each position is indicated at the bottom of the display. Within the display, the operating system shows buffer and task segments. Each segment beginning with a highlighted P is a procedure segment. The two-digit numeric indicator following the P is the installed procedure ID. A segment that begins with a highlighted T is a task segment. The two-digit numeric indicator following the T is the installed task ID. The third and fourth digits of the task segments indicate the run-time ID of the task. A segment beginning with a highlighted B is a file management blocking buffer. Blank space on the display is allocated space being used by the header to the left of the blank. Available memory is indicated by the solid graphics symbol.

If you press the F1 key, the display changes. The used area is blank and the available area is shaded.

If you press the CMD key, the SCI prompt appears but the screen display remains. You can then press RETURN or any function key to obtain the main menu.

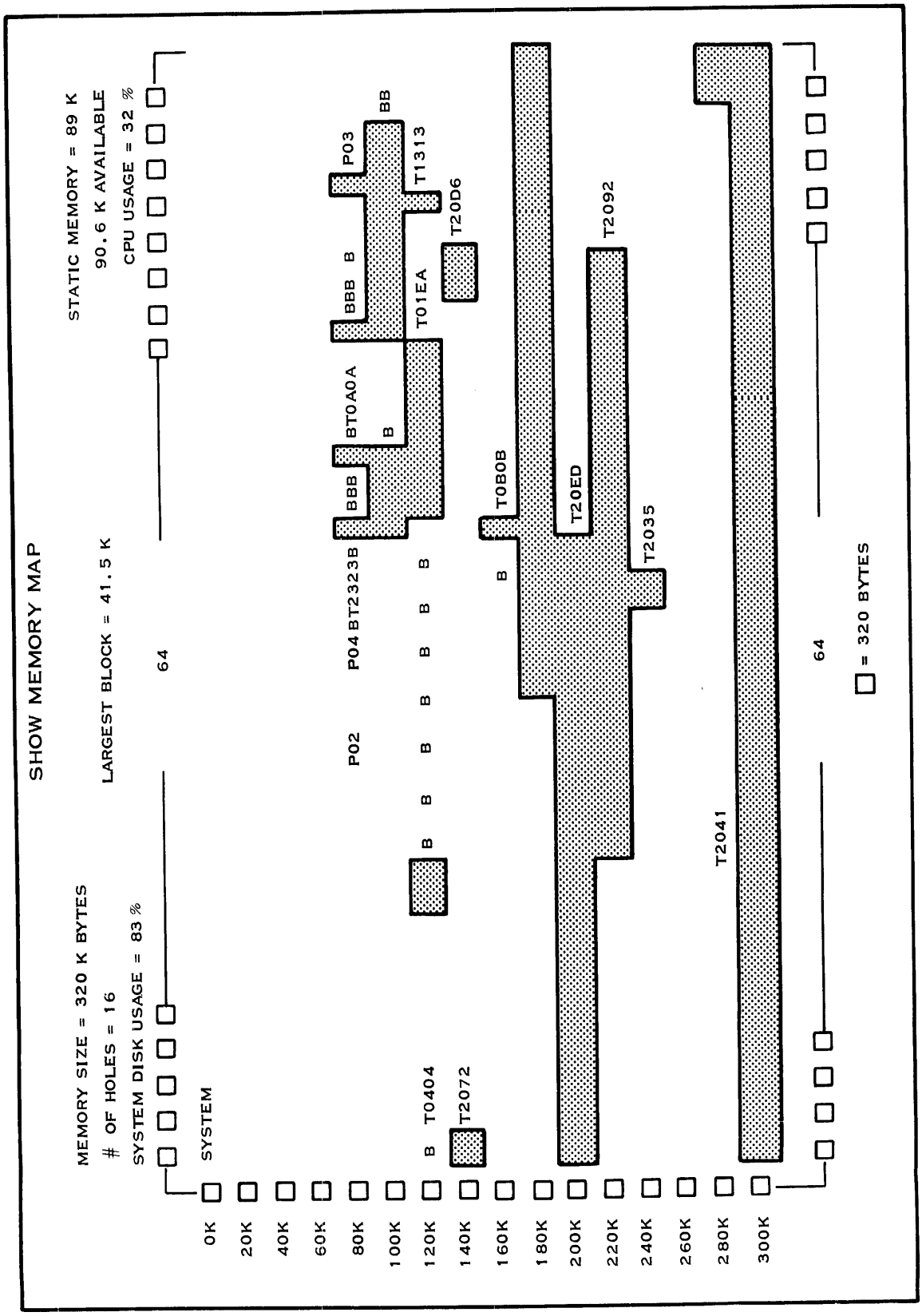


Figure 8-10. Sample Output the SMM Command Produces

SMM

8.153.4 Commands Related to SMM

SMS (Show Memory Status)
SSTM (Show System Table Map)

8.154 SMS (SHOW MEMORY STATUS)

The SMS command allows you to obtain information on system memory usage and SCI activity. The SMS command displays information about the following: memory size, system size, system table area sizes, and SCI usage. The information is returned in the following format:

```
                        SHOW MEMORY STATUS

                   TOTAL MEMORY SIZE = <number> K
     DX10 OS SIZE = <number> K          USER AREA = <number> K

                SYSTEM TABLE AREA = <int> WORDS
   CURRENT USAGE = <int> WORDS      LARGEST AREA USED = <int> WORDS

                            SCI INFORMATION
 FOREGROUND LIMIT = <int> TERMINALS    CURRENTLY ACTIVE = <int> TERMINALS
 BACKGROUND LIMIT = <int> TASKS        ACTIVE/WAITING = <int>/<int> TASKS
```

8.154.1 SMS Command Format

SMS
SHOW MEMORY STATUS
LISTING ACCESS NAME: [acnm]@

8.154.2 SMS Command User Responses

System Prompt	Response Required or Optional	User Response
LISTING ACCESS NAME:	O	Enter the access name of a file or device to which the returned information is to be written. *The default value is the terminal local file (TLF).*

8.154.3 SMS Command Example

The following example returns the SMS information to your TLF:

```
[] SMS
                        SHOW MEMORY STATUS

                   TOTAL MEMORY SIZE = 128 K
     DX10 OS SIZE = 0.9 K              USER AREA = 127.1 K

                SYSTEM TABLE AREA = 9687 WORDS
   CURRENT USAGE = 2385 WORDS            LARGEST AREA USED = 4533 WORD

                            SCI INFORMATION
  FOREGROUND LIMIT =  9 TERMINALS       CURRENTLY ACTIVE = 2 TERMINAL
 BACKGROUND ACTIVE LIMIT =  4 TASKS     ACTIVE/WAITING = 0/0 TASKS
```

SMS

8.154.4 Commands Related to SMS

SMM (Show Memory Status)
SSTM (Show System Table Map)

8.155 SOS (SHOW OUTPUT STATUS)

The SOS command shows one of the following types of status of the queued output:

- The status of one user's files (as identified by the user ID) in the output queue.
- The status of all users' files in the output queue.
- The status of queued files waiting for output at all devices. You cannot determine the status of a device used for output by using the SOS command (for example, whether or not the device is online.) You should use the List Device Configuration (LDC) command to determine device states. Device errors for conditions such as out of paper or physically not present are written to the system log.

8.155.1 SOS Command Format

```
SOS
SHOW OUTPUT STATUS
USER ID: [string] @
DEVICE NAME: [acnm] @
```

8.155.2 SOS Command User Responses

System Prompts	Response Required or Optional	User Responses
USER ID:	O	To view the status of one user's files in the output queue, enter the identifying user ID. Enter a null response to list the status of all users' files in the specified device output queue (or all output queues).
DEVICE NAME:	O	To view the status of files waiting for output at one particular device, enter the device name of that device. Enter a null response to list the status of queued files waiting for output at all devices that have queued output.

8.155.3 SOS Command Examples

The following examples show how you can enter the SOS command to get the status of one user's files in a particular output queue, to get the status of all user's files in all output queues, to get the status of all user's queued files waiting for output at a particular device, and to get the status of all user's queued files waiting for output at all devices. Table 8-14 describes the information that appears under each heading of the results.

SOS

NOTE

If no ACTIVE status appears, as in the first example, something is wrong with the device queues, you have entered the HO command followed by the KO command in which case you need to execute an RO command, or you have entered the pathname of a nonexistent file. If the RO command does not cause output to resume and the system log files (.S$SLG1 and .S$SLG2) do not contain a message indicating that you entered a nonexistent pathname, then something is wrong with the device or the device queues. Device errors are also written to the system log.

```
[] SOS
SHOW OUTPUT STATUS
 USER ID:   KOC018
 DEVICE NAME:   LP01
 USER   DEVICE  STATUS   FILE NAME
  KOC018  LP01  WAITING  .FILMGR.LIST.FM$TSK
  KOC018  LP01  WAITING  .FILMGR.SOURCE.FMCLOS
  KOC018  LP01  WAITING  .FILMGR.SOURCE.FMOPEN

[] SOS
SHOW OUTPUT STATUS
 USER ID:   JOD009
 DEVICE NAME:
 USER   DEVICE  STATUS   FILE NAME
  JOD009  LP01  ACTIVE   .DXMISC.SOURCE.STARTR
  JOD009  LP02  ACTIVE   .DXMISC.SOURCE.INTRPT

[] SOS
SHOW OUTPUT STATUS
 USER ID:
 DEVICE NAME:   LP01
 USER   DEVICE  STATUS   FILE NAME
  JOD009  LP01  ACTIVE   .DXMISC.SOURCE.STARTR
  JOD009  LP01  WAITING  .DXMISC.SOURCE.INTRPT
  KOC018  LP01  WAITING  .FILMGR.LIST.FM$TSK
  KOC018  LP01  WAITING  .FILMGR.SOURCE.FMCLOS
  KOC018  LP01  WAITING  .FILMGR.SOURCE.FMOPEN

[] SOS
SHOW OUTPUT STATUS
 USER ID:
 DEVICE NAME:
 USER   DEVICE  STATUS   FILE NAME
  JOD009  LP01  ACTIVE   .DXMISC.SOURCE.STARTR
  JOD009  LP02  WAITING  .DXMISC.SOURCE.INTRPT
  KOC018  LP01  WAITING  .FILMGR.LIST.FM$TSK
  KOC018  LP03  WAITING  .FILMGR.SOURCE.FMCLOS
  KOC018  LP01  WAITING  .FILMGR.SOURCE.FMOPEN
```

Table 8-14. Information the SOS Command Produces

Heading	Description
USER	The user ID of a user whose file is waiting in the output queue to be written to the output device.
DEVICE	The device name of a device at which a file is being written or at which a file is waiting to be written.
STATUS	The status of a file is ACTIVE if it is currently being written to a device and WAITING if it is waiting to be written to a device.
FILE NAME	The name of a file that is currently waiting to be written or that is currently being written to the device.

8.155.4 Commands Related to SOS

HO (Halt Output at Device)
KO (Kill Output at Device)
RO (Resume Output at Device)

SP

8.156 SP (SHOW PANEL)

The SP command displays the debug panel for a specified task. Refer to Volume III for a detailed description.

8.157 SPI (SHOW PROGRAM IMAGE)

The SPI command displays the disk-resident memory image of a module (defined as a task, procedure, or overlay) for the specified program file. You can direct the display to a device or a file. Refer to Volume III for a detailed description.

SRF

8.158 SRF (SHOW RELATIVE TO FILE)

The SRF command displays any word or group of words within a file. It assumes that you have a knowledge of the file structure and allows you to address any word within the file. Refer to Volume III for a detailed description.

8.159 SSTM (SHOW SYSTEM TABLE MAP)

The SSTM command displays the allocation of the system table area. You can enter the SSTM command to monitor the dynamic allocation of table space. You can only use this command on a 24-line VDT. Approximately every five seconds, the SSTM command processor scans the DX10 system table allocation list and updates the display.

8.159.1 SSTM Command Format

SSTM
SHOW SYSTEM TABLE MAP

8.159.2 SSTM Command User Responses

No user responses are required.

8.159.3 SSTM Command Example

The following example shows how you can enter the SSTM command to display the allocation of the system table area.

[ ] SSTM

Line one contains the table size that is the total amount of memory reserved for system table use. NUMBER OF BLOCKS is the number of separate pieces of memory available for allocation. AVERAGE BLOCK is the average size of each of the blocks.

Line two shows the smallest and largest blocks allocated. TOTAL AVAILABLE is the number of bytes of memory that are available to be allocated.

Line three shows HIGHEST USAGE, which is the highest amount of table space ever allocated since the last initial program load (IPL). CURRENT USAGE is the amount of space currently being used.

All amounts are in bytes.

The graph is a pictorial representation of the memory in the system table area. The numbers on the left are displacements into the table and not actual addresses. The amount of memory represented by each graphic block is indicated at the bottom of the graph. Available space is indicated by the solid graphics symbols.

You can obtain an itemized list of the allocation of table space by pressing the F1 key.

In the upper left is the time of day in hours:minutes:seconds. SIZE is the total number of bytes of system table area. CURRENT is the number of bytes you are currently using. HIGHEST is the highest number of bytes used since the last initial program load (IPL). # HOLES is the number of blocks allocated. % FULL is the percentage of table space used. INTERTASK usage is the amount of space allocated to the intertask area. ADD. I/O is the limit on the amount of space allocatable to I/O buffers. MOST USED is the highest amount used by I/O buffers.

The graph shows the percentage allocation to each of the data structures shown. The numbers to the left of the graph are the number of bytes allocated.

If you press the CMD key, the SCI prompt appears but the screen display remains. You can then press RETURN or any function key to obtain the main menu.

SSTM

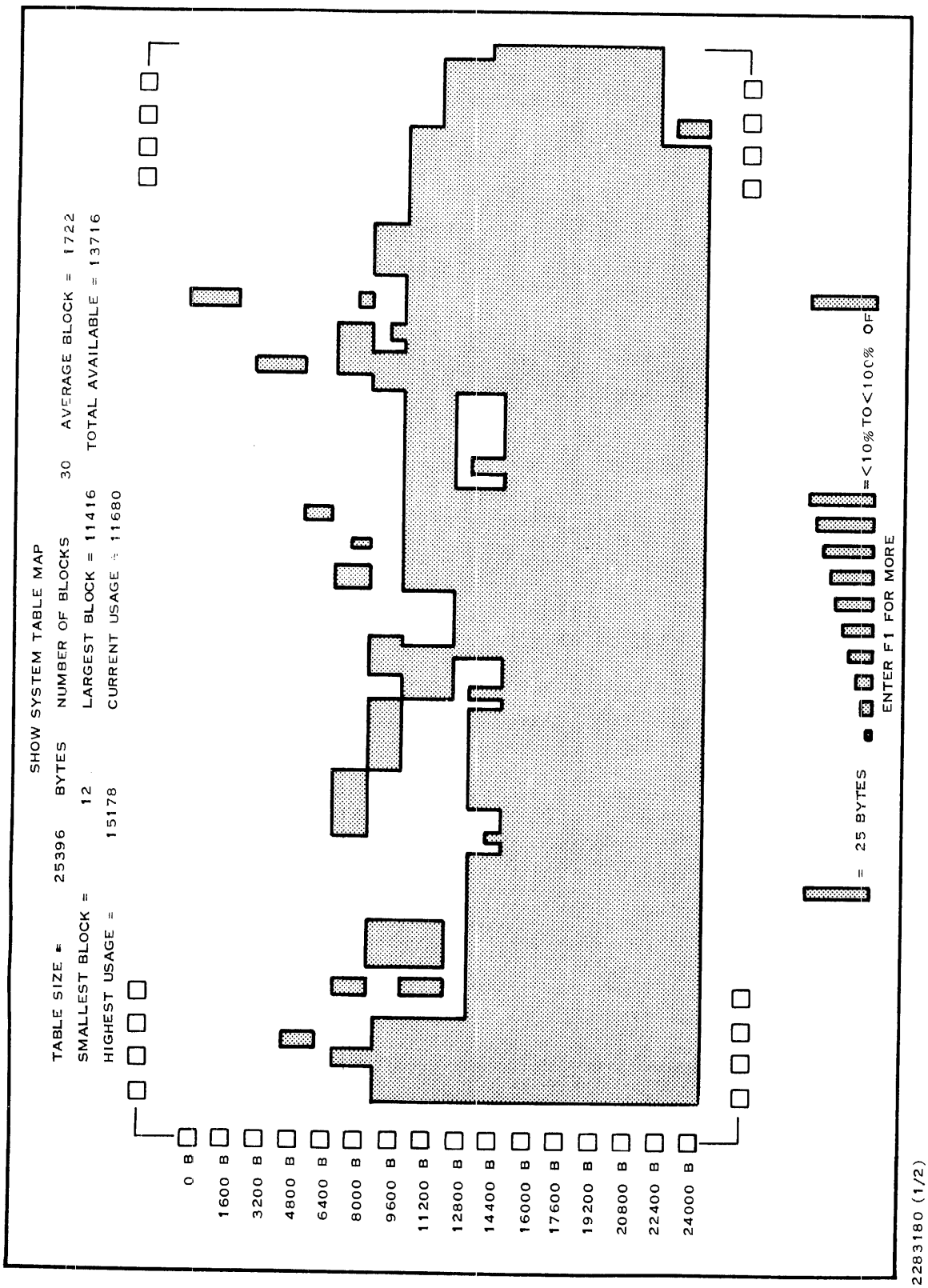


Figure 8-11. Sample Output the SSTM Command Produces (Sheet 1 of 2)

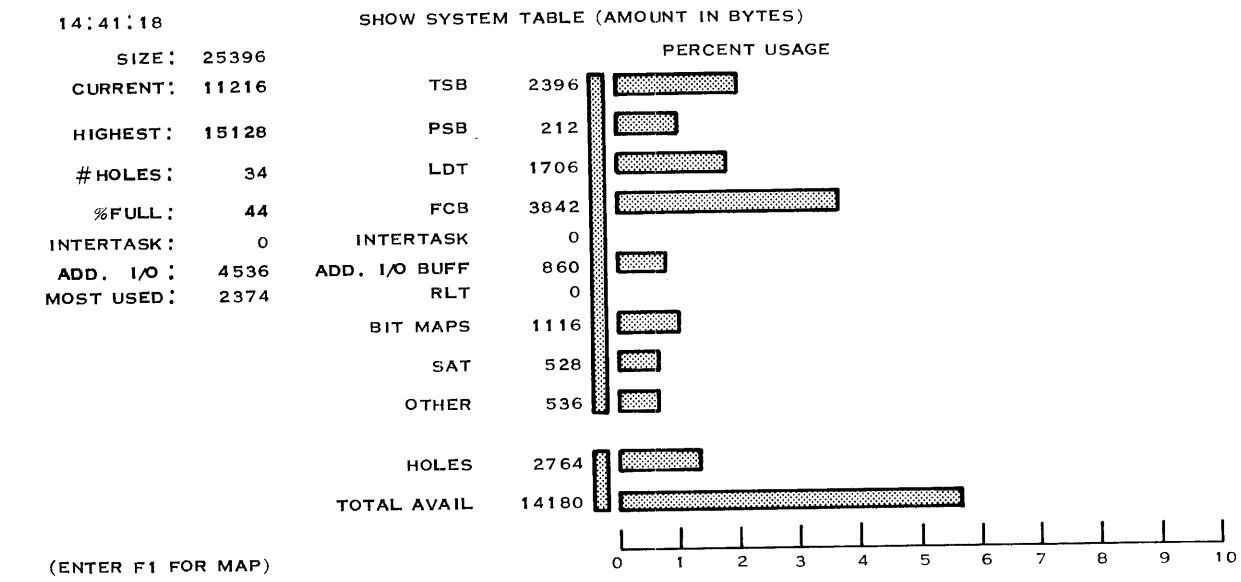


Figure 8-11. Sample Output the SSTM Command Produces (Sheet 2 of 2)

8.159.4 Commands Related to SSTM

SMM (Show Memory Map)
SMS (Show Memory Status)

ST

8.160 ST (SIMULATE TASK)

The ST command provides controlled and traced execution of the instructions in a task. Refer to Volume III for a detailed description.

8.161 STI (SHOW TERMINAL INFORMATION)

The STI commands allows you to examine the status of your own terminal. The STI command displays or prints the command description message and status information in the following form:

MOST RECENT TASK RUN ID: [string1]
MOST RECENT LUNO ASSIGNMENT: [string2]
TEXT EDIT IN PROGRESS: [string3]
MOST RECENT EDIT FILE: [string4]
TASK BEING DEBUGGED: [string5]

where:

string1 = the run-time ID of the most recently activated task

string2 = the station-local or global LUNO most recently assigned interactively at the identified terminal

string3 = Y (yes) if a text edit is currently in progress
N (no) if a text edit is not in progress

string4 = the file name of the most recently edited edit file

string5 = the task ID of a task currently being debugged

8.161.1 STI Command Format

STI
TERMINAL INFORMATION . . . TERMINAL: devicename . . . USER:user id

NOTE

The STI command displays or prints a command description message that contains the device name of the user's terminal and the user ID of the user (if log-on is required at the terminal).

8.161.2 STI Command User Responses

You can change the following:

- Most recent task run ID.
- Most recent LUNO assignment.
- Most recent edit file.

STI

8.161.3 STI Command Example

The following example shows entry of an STI command by a user logged on under user ID WHT016 at terminal ST01. The status information displayed shows that the ID of the task most recently run from the terminal is >86 and that neither a text edit operation nor a debug operation is in progress.

```
[] STI
TERMINAL INFORMATION ..... TERMINAL: ST01 ..... USER: WHT016
 MOST RECENT TASK RUN ID: >86
 MOST RECENT LUNO ASSIGNMENT:
 TEXT EDIT IN PROGRESS: N
 MOST RECENT EDIT FILE:
 TASK BEING DEBUGGED
```

8.161.4 Commands Related to STI

LTS	(List Terminal Status)
MTS	(Modify Terminal Status)

8.162 STS (SHOW TASK STATUS)

The STS command displays the status of a specific task or of all tasks currently active in the system. Table 8-15 describes the information that is given for each task under each heading of output produced by an STS command. During execution of the STS command, a task in memory may be rescheduled causing the task to appear twice in the STS output. This occurs infrequently and does not indicate a problem.

Table 8-15. Task Status Information the STS Command Produces

Heading	Description
BID ID	The installed ID of the task whose status is shown.
RUN ID	The run-time ID assigned to the task by DX10 when it activated the task.
STATION	The station number with which the executing task is associated.
STATE	A code that describes the state of the task as follows:

Code (Hex.)	Meaning
00	Active task
01	Active task
04	Terminated task
05	Task in time delay
06	Suspended task
07	Currently executing task
09	Task awaiting completion of I/O
0A	Task awaiting assignment of a device for I/O
0B	Task awaiting disk file utility services
0D	Task awaiting file management services
0E	Task awaiting overlay loader services
0F	Task awaiting initial load
11	Task awaiting disk management services
12	Task awaiting tape management services
13	Task awaiting system overlay loader services
14	Task awaiting task driven supervisor call processor
15	Task awaiting Get Memory services
17	Task suspended for coroutine activation
18	Task awaiting termination task services
19	Task awaiting completion of an I/O
1A	Task awaiting memory management services
1B	Task is eligible for roll-out when requested I/O completes
1C	Task activated while roll is in progress
1D	Suspended for initiate I/O threshold
1E	Suspended for locked directory
1F	Suspended for task management directory buffer
24	Task suspended for queue input
FF	Dummy task state

STS

Table 8-15. Task Status Information the STS Command Produces (Continued)

Heading	Description
PRIORITY	A code that identifies the priority of the task as follows:

Code	Meaning
00	The task is a critical system task (highest priority)
R1	Real-time priority one
.	
.	
.	
R127	Real-time priority one hundred twenty-seven
01	Second priority
02	Third priority
03	Lowest priority

FLAG1 — A task status block flag within which bits are set to indicate the characteristics of a task as follows:

Bit	Task Characteristic
0	System task
1	Privileged task
2	Memory resident task
3	The task must take an end action on error
4	Roll-out candidate
5	Rolled out
6	The task is scheduled to be aborted or terminated.
7	A call has been made from another task to reactivate the task but the task has not yet suspended. As soon as the task suspends, it is reactivated.
8	The task was bid by another task and the other task suspended. When this task finishes, the task is to reactivate the suspended task.
9	Serially reusable task
10	Task quieting in progress
11	This is the first activation of the task
12	Leave task alone
13	Task is under control of an alternate TSB
14	SCI flag for scanning TSB chain
15	Replicatable task

Table 8-15. Task Status Information the STS Command Produces (Continued)

Heading	Description
FLAG 2	A task status block flag within which bits are set to indicate the characteristics of a task as follows:

Bit	Task Characteristic
0	The task is to be suspended the next time it tries to execute
1	The task is being controlled
2	SVC traps to be taken when specified
3	SVC switch: when 0 SVC traps taken
4	Execution stopped by scheduler
5	Execution stopped by trapped SVC
6	Execution stopped by XOP 15, 15
7	Dynamic priority management

WP	The workspace pointer register address relative to the task address in memory
PC	The currently executing task instruction address relative to the task address in memory.

8.162.1 STS Command Format

STS
SHOW TASK STATUS
INSTALLED ID: [int]
OUTPUT ACCESS NAME: [acnm] @

8.162.2 STS Command User Responses

System Prompts	Response Required or Optional	User Responses
INSTALLED ID:	O	Enter the installed ID number of the task whose status you want to display or list. *If you enter a null response, the status of all activated user tasks is provided.*
OUTPUT ACCESS NAME:	O	Enter the device name or the file pathname to which DX10 should write the status information produced by the STS command. *The default value is the terminal local file (TLF).*

STS

8.162.3 STS Command Example

The following example shows how you can enter the STS command to show the status of a task with an installed ID of >20. The status of all tasks with an installed ID of >20 is displayed. You must know the station ID to choose the task for which you want status information.

```
[] STS
SHOW TASK STATUS
 INSTALLED ID: >20
 OUTPUT ACCESS NAME: ME
BID ID  RUN ID  STATION  STATE  PRIORITY  FLAG1  FLAG2   WP    PC
  20      1C       6      09       01     5001   0100   3B30  24C0
  20      0D       1      17       01     5001   0100   3B30  5518
  20      C2       4      09       01     5001   0100   3B30  24C0
  20      62       7      05       01     5001   0100   3BBC  487C
  20      48       9      09       01     5401   0100   3B30  24C0
  20      2B       8      09       01     5401   0100   3B30  24C0
```

To show the status for all tasks currently executing, press the RETURN key in response to the INSTALLED ID and the OUTPUT ACCESS NAME prompts.

8.162.4 Commands Related to STS

KT (Kill Task)

8.163 SV (SHOW VALUE)

The SV command displays the value of the specified expression. The hexadecimal, decimal, and ASCII representations of the value are given. Refer to Volume III for a detailed description.

SVL

8.164 SVL (SAVE LINES)

After you activate the Text Editor, the SVL command copies selected lines from the current edit file to an external file that you specify. Refer to Volume IV for a detailed description.

8.165 SVS (SHOW VOLUME STATUS)

The SVS command displays or lists the status of a disk volume. Table 8-16 describes the information that the SVS command displays or lists.

Table 8-16. Information the SVS Command Produces

Field	Description
VOLUME NAME:	The name of the volume that is physically in the disk drive.
ADUS:	The size in ADUs of the disk. This size includes the overhead information that the INV command creates. The space that the overhead information occupies is not available for files.
#BAD:	The number of bad ADUs on the disk volume.
BYTES/ADU:	The number of bytes per ADU in the volume.
AVAILABLE:	The total number of ADUs still available for directories, subdirectories, and files.
LARGEST AVAILABLE BLOCK:	The size, in ADUs, of the largest contiguous block available for a file.
CONTROLLER ERRORS:	The number of times data has had to be rewritten or reread on the volume since IPL for the system disk or since IV for a secondary disk because of an error detected by the disk controller. This value should normally be zero. However, a value of from 1 to 10 during a week should not be cause for alarm.
PRIMARY SYSTEM IMAGE:	If the volume is a system disk volume, the name of the primary system image in S$IMAGES is given.
SECONDARY SYSTEM IMAGE:	If the volume is a system disk volume, the name of the secondary image in S$IMAGES is given.
NAME INSTALLED:	Name of the disk volume that is currently installed. This field is blank if no volume is installed. If this name is not the same as the VOLUME NAME given previously, then this volume was removed from the drive without being unloaded (UV command). You must enter the UV command for this name before installing another. If this disk is being used by DCOPY, INV, CVD, or IDS, this field is "$$$$$$$$" and the disk cannot be unloaded.
DEVICE NAME:	Name of the device of which the SVS command is giving information.

SVS

8.165.1 SVS Command Format

SVS
SHOW VOLUME STATUS
VOLUME NAME: [name]@
DRIVE NAME: [name]@
OUTPUT ACCESS NAME: [acnm]@

8.165.2 SVS Command User Responses

System Prompts	Response Required or Optional	User Responses
VOLUME NAME:	O	Enter the name of the volume whose status you want to display. If you omit this response, you must specify the drive name. If you supply both drive name and volume name, the volume name is used.
DRIVE NAME:	O	Enter the device name of the disk unit on which the volume is installed. If you omit the drive name, you must supply the volume name. If you supply both names, the volume name is used.
OUTPUT ACCESS NAME:	O	Enter the device name or the file pathname to which DX10 writes the status information produced by the SVS command. *The default is the terminal local file (TLF).*

8.165.3 SVS Command Example

The following example shows how you can enter the SVS command to list the status at your terminal of a volume mounted in disk unit DS02:

```
[] SVS
SHOW VOLUME STATUS
 VOLUME NAME:
 DRIVE NAME:  DS02
 OUTPUT ACCESS NAME:  ME

VOLUME NAME: HAWK2    ADUS: 16320   # BAD: 0    BYTES/ADU: 288
AVAILABLE: 4519    LARGEST AVAILABLE BLOCK: 4519    CONTROLLER ERRORS: 0
PRIMARY SYSTEM IMAGE: CC4    SECONDARY SYSTEM IMAGE:
NAME INSTALLED: HAWK2        DEVICE NAME: DS02
```

8.165.4 Commands Related to SVS

INV (Initialize New Volume)
IV (Install Volume)
UV (Unload Volume)

8.166 SWR (SHOW WORKSPACE REGISTERS)

The SWR command displays the current workspace of a task. Refer to Volume III for a detailed description.

TGS

8.167 TGS (TEST GENERATED SYSTEM)

The TGS command enables you to test a new system before you install it as the primary DX10 system. After testing successfully completes, you can use the Install Generated System (IGS) command to install the system.

The TGS command does not install the new system as the primary system. It designates the system under test to be the secondary system. The next time an initial program load (IPL) is performed, this secondary system is loaded as the operating system. On subsequent IPLs, the primary system is loaded. This offers you an opportunity to ensure that the generated system will in fact load successfully and function as desired.

To install the new system as the primary system, you must execute the IGS command. In effect, the IGS command causes the system entered for the SYSTEM NAME prompt to become the primary system each time an IPL is performed. If you do not execute the IGS command on the newly generated system, the original primary system is loaded at the next IPL. Section 3 discusses the IPL.

8.167.1 TGS Command Format

```
TGS
TEST GENERATED SYSTEM
TARGET DISK: [name]@
SYSTEM NAME: [name]@
```

8.167.2 TGS Command User Responses

System Prompts	Response Required or Optional	User Responses
TARGET DISK:	O	Enter the device name of the disk that contains the program file, .S$IMAGES, in which the new system is installed. *The default value is DS01 or the target disk previously assigned by a PGS, IGS, TGS, or ALGS command.*
SYSTEM NAME:	O	Enter a one- to five-character alphanumeric string that is the name of the system that was generated. *The default value is the system name previously assigned by a PGS, IGS, TGS, or ALGS command.* If you do not specify a system name, a display of the current IPL status is given with no alternation made.

8.167.3 TGS Command Example

The following example shows how you can use the TGS command to test a system named ITGS. PAE5 is the disk name.

```
[] TGS
TEST, GENERATED SYSTEM
 TARGET DISK:    DS01
 SYSTEM NAME:    ITGS


                                    IGS/TGS
                           PAE5     SYSTEM IPL STATUS

PRIMARY SYSTEM = S$SUP                        SECONDARY SYSTEM = ITGS
                        TEST SECONDARY SYSTEM
```

8.167.4 Commands Related to TGS

ALGS	(Assemble and Link Generated System)
IGS	(Install Generated System)
PGS	(Patch Generated System)

TXCM

8.168 TXCM (COMPRESS DISKETTE FILE)

The TXCM command releases ADUs that are allocated to the specified file but not currently used.

8.168.1 TXCM Command Format

```
TXCM
COMPRESS DISKETTE FILE
TX990 ACCESS NAME: <name>
```

8.168.2 TXCM Command User Response

System Prompt	Response Required or Optional	User Response
TX990 ACCESS NAME:	R	Enter a valid TX990 access name of the file to be compressed.

8.168.3 TXCM Command Example

The following example shows how you can enter the TXCM command to release ADUs allocated to a file but not currently used:

```
[ ] TXCM
COMPRESS DISKETTE FILE
  TX990 ACCESS NAME: DK01:FILEA/SRC
```

8.168.4 Commands Related to TXCM

DXTX	(Convert DX10 File to TX990 Diskette File)
IBMUTL	(Convert IBM-Formatted Diskette Data-Sets)
TXCP	(Change Diskette File Protection)
TXDF	(Delete Diskette File)
TXDX	(Convert TX990 Diskette File to DX10 File)
TXFD	(Format Diskette)
TXMD	(Map Diskette)
TXSF	(Set System File)
TXXLE	(Execute Link Editor to Produce TX Program File)

8.169 TXCP (CHANGE DISKETTE FILE PROTECTION)

The TXCP command enables you to modify file protection (change code) of a diskette file. Diskette files are unprotected (U), delete protected (D), or write protected (W).

8.169.1 TXCP Command Format

```
TXCP
CHANGE DISKETTE FILE PROTECT CODE
TX990 ACCESS NAME: < acnm>
CODE (U,D,W): < string>
```

8.169.2 TXCP Command User Responses

System Prompts	Response Required or Optional	User Responses
TX990 ACCESS NAME:	R	Enter a valid TX990 pathname.
CODE (U,D,W):	R	Enter U (unprotected), D (delete protected), or W (write protected). The default is U (unprotected).

8.169.3 TXCP Command Example

The following example shows how you can use the TXCP command to modify the file protection of a diskette file:

```
[ ] TXCP
CHANGE DISKETTE FILE PROTECT CODE
 TX990 ACCESS NAME: DK01:FILEA/SRC
 CODE (U,D,W): U W
```

8.169.4 Commands Related to TXCP

DXTX	(Convert DX10 File to TX990 Diskette File)
IBMUTL	(Convert IBM-Formatted Diskette Data-Sets)
TXCM	(Compress Diskette File)
TXDF	(Delete Diskette File)
TXDX	(Convert TX990 Diskette File to DX10 File)
TXFD	(Format Diskette)
TXMD	(Map Diskette)
TXSF	(Set System File)
TXXLE	(Execute Link Editor to Produce TX Program File)

TXDF

8.170 TXDF (DELETE DISKETTE)

The TXDF command deletes the specified diskette file.

8.170.1 TXDF Command Format

```
TXDF
DELETE DISKETTE FILE
TX990 ACCESS NAME: <acnm>
```

8.170.2 TXDF Command User Response

System Prompt	Response Required or Optional	User Response
TX990 ACCESS NAME:	R	Enter the TX990 access name of the file you want to delete.

8.170.3 TXDF Command Example

The following example shows how you can enter the TXDF command to delete a diskette file:

```
[ ] TXDF
DELETE DISKETTE FILE
  TX990 ACCESS NAME: DK01:FILEA/SRC
```

8.170.4 Commands Related to TXDF

DXTX	(Convert DX10 File to TX990 Diskette File)
IBMUTL	(Convert IBM-Formatted Diskette Data-Sets)
TXCM	(Compress Diskette File)
TXCP	(Change Diskette File Protection)
TXDX	(Convert TX990 Diskette File to DX10 File)
TXFD	(Format Diskette)
TXMD	(Map Diskette)
TXSF	(Set System File)
TXXLE	(Execute Link Editor to Produce TX Program File)

8.171 TXDX (CONVERT TX990 DISKETTE FILE TO DX10 FILE)

The TXDX command converts a TX990 diskette sequential or relative record file to a DX10 sequential or relative record file, respectively.

8.171.1 TXDX Command Format

```
TXDX
TX990 FILE TO DX10 FILE
TX990 ACCESS NAME: < acnm>
DX10 ACCESS NAME: < acnm> @
REPLACE?: < yes/no>
```

8.171.2 TXDX Command User Responses

System Prompts	Response Required or Optional	User Responses
TX990 ACCESS NAME:	R	Enter a valid TX990 access name.
DX10 ACCESS NAME:	R	Enter a valid DX10 access name.
REPLACE?:	R	Enter YES (replace flag) or NO. If the file exists, an error is returned for a N (no) reply. The initial value is N.

If the DX10 file does not exist, a file of the input file type is created.

8.171.3 TXDX Command Example

The following example shows how you can use the TXDX command to convert a TX990 file to a DX10 file:

```
[ ] TXDX
TX990 FILE TO DX10 FILE
  TX990 ACCESS NAME: :FILEA/SRC
  DX10 ACCESS NAME: .FILEA
  REPLACE?: YES
```

8.171.4 Commands Related to TXDX

DXTX	(Convert DX10 File to TX990 Diskette File)
IBMUTL	(Convert IBM-Formatted Diskette Data-Sets)
TXCM	(Compress Diskette File)
TXCP	(Change Diskette File Protection)
TXDF	(Delete Diskette File)
TXFD	(Format Diskette)
TXMD	(Map Diskette)
TXSF	(Set System File)
TXXLE	(Execute Link Editor to Produce TX Program File)

TXFD

8.172 TXFD (FORMAT DISKETTE)

The TXFD command formats the diskette and initializes the file directory, allocation bit maps, and information block required by TX990. TXFD also copies TXBOOT to the diskette physical track zero and optionally assigns a volume name to the diskette.

8.172.1 TXFD Command Format

```
TXFD
FORMAT DISKETTE
DISKETTE: < name> @
VOLUME NAME: [name]@
IDENTIFICATION NAME: < string> @
```

8.172.2 TXFD Command User Responses

System Prompts	Response Required or Optional	User Responses
DISKETTE:	R	Enter the name of the diskette you want to format (for example, DK02).
VOLUME NAME:	O	Enter a one- to four-character diskette volume name.
IDENTIFICATION NAME:	R	Enter a 1- to 32-character field (ID) to be stored in the diskette information block to identify the diskette.

8.172.3 TXFD Command Example

The following example shows how you can use the TXFD command to format a diskette:

```
[ ] TXFD
FORMAT DISKETTE
  DISKETTE: DK01
  VOLUME NAME: TXDS
  IDENTIFICATION NAME: DISKETTE#1
CHECKING DK01
INITIALIZATION COMPLETE
```

8.172.4 Commands Related to TXFD

DXTX	(Convert DX10 File to TX990 Diskette File)
IBMUTL	(Convert IBM-Formatted Diskette Data-Sets)
TXCM	(Compress Diskette File)
TXCP	(Change Diskette File Protection)
TXDF	(Delete Diskette File)
TXDX	(Convert TX990 Diskette File to DX10 File)
TXMD	(Map Diskette)
TXSF	(Set System File)
TXXLE	(Execute Link Editor to Produce TX Program File)

8.173 TXMD (MAP DISKETTE)

The TXMD command requests the diskette name and listing access name. The TXMD processor displays the diskette directory contents, its file name, file type, level of protection, and the number of allocation units assigned to each file.

8.173.1 TXMD Command Format

TXMD
MAP DISKETTE
DISKETTE: < name> @
LISTING ACCESS NAME: [acnm]@

8.173.2 TXMD Command User Responses

System Prompts	Response Required or Optional	User Responses
DISKETTE	R	Enter the diskette name of the diskette you want to map (for example, DK02).
LISTING ACCESS NAME:	O	Enter the name of a map listing file or device (for example, .MAPLIST or LP01). *The default value is the terminal local file (TLF).*

8.173.3 TXMD Command Example

The following example shows how you can enter the TXMD command to map a diskette:

```
[] TXMD
MAP DISKETTE
 DISKETTE:   DK01
 LISTING ACCESS NAME:   ME
DISC I.D.: DISKETTE #1

DK01    ALLOC. UNITS(TOTAL): 333     FREE: 268      BAD: 0

   FILE       TYPE   PT   ALLOC. UNITS

FILEA           S     U          8
FILEB           S     U          8
FILEC           S     W         35
TEMP            S     U          8

** SYSTEM RESERVED AU'S = 6
```

TXMD

8.173.4 Commands Related to TXMD

DXTX	(Convert DX10 File to TX990 Diskette File)
IBMUTL	(Convert IBM-Formatted Diskette Data-Sets)
TXCM	(Compress Diskette File)
TXCP	(Change Diskette File Protection)
TXDF	(Delete Diskette File)
TXDX	(Convert TX990 Diskette File to DX10 File)
TXFD	(Format Diskette)
TXSF	(Set System File)
TXXLE	(Executive Link Editor to Produce TX Program File)

8.174 TXSF (SET SYSTEM FILE)

The TXSF command designates a file on the diskette as the system file (that is, the file that is loaded when the diskette loader in ROM is executed).

8.174.1 TXSF Command Format

```
TXSF
SET SYSTEM FILE
TX990 ACCESS NAME: < acnm>
```

8.174.2 TXSF Command User Response

System Prompt	Response Required or Optional	User Response
TX990 ACCESS NAME:	R	Enter the name of a file to be designated as the system file on the diskette.

8.174.3 TXSF Command Example

The following example shows how you can use the TXSF command to designate a file on a diskette as the system file:

```
[ ] TXSF
SET SYSTEM FILE
  TX990 ACCESS NAME: DK01:FILEA/OBJ
```

8.174.4 Commands Related to TXSF

DXTX	(Convert DX10 File to TX990 Diskette File)
IBMUTL	(Convert IBM-Formatted Diskette Data-Sets)
TXCM	(Compress Diskette File)
TXCP	(Change Diskette File Protection)
TXDF	(Delete Diskette File)
TXDX	(Convert TX990 Diskette File to DX10 File
TXFD	(Format Diskette)
TXMD	(Map Diskette)
TXXLE	(Execute Link Editor to Produce TX Program File)

TXXLE

8.175 TXXLE (EXECUTE LINK EDITOR TO PRODUCE TX PROGRAM FILE)

The TXXLE command invokes the special DX10 link editor that produces TX program files. This link editor allows you to create TX program files on DX10 systems and then copy them to diskettes using the DXTX file conversion utility. If you want automatic overlays for the TX task being linked, then you must include the following command:

```
LIBRARY .S$SYSLIB.TX
```

If you are using the load overlay subroutine described in the *DX10 Link Editor Reference Manual*, then you must include the following command:

```
INCLUDE .S$$SYSLIB.TX.L$$OLD
```

8.175.1 TXXLE Command Format

```
TXXLE
EXECUTE LINK EDITOR PRODUCING TX PROG FILE
CONTROL ACCESS NAME: < acnm> @
LINKED OUTPUT ACCESS NAME: < acnm> @
LISTING ACCESS NAME: [acnm]@
PRINT WIDTH: < int>
```

8.175.2 TXXLE Command User Responses

System Prompts	Response Required or Optional	User Responses
CONTROL ACCESS NAME:	R	Enter the pathname of the DX10 link edit control file to link the task to be downloaded to TX.
LINKED OUTPUT ACCESS NAME:	R	Enter the pathname of the relative record file that is to be the TX program file. The link editor will autocreate this file.
LISTING ACCESS NAME:	R	Enter the pathname of the file that is to contain the link map.
PRINT WIDTH:	R	Enter the number of characters per line of the link map. The initial value is 80.

8.175.3 TXXLE Command Example

The following example shows how you can use the TXXLE command to produce TX program files:

```
[ ] TXXLE
EXECUTE LINK EDITOR PRODUCING TX PROG FILE
  CONTROL ACCESS NAME: .FILE.CONTROL
  LINKED OUTPUT ACCESS NAME: .FILE.TXPROG
  LISTING ACCESS NAME: .FILE.LNKMAP
  PRINT WIDTH: 80
```

8.175.4 Commands Related to TXXLE

DXTX	(Convert DX10 File to TX990 Diskette File)
IBMUTL	(Convert IBM-Formatted Diskette Data-Sets)
TXCM	(Compress Diskette File)
TXCP	(Change Diskette File Protection)
TXDF	(Delete Diskette File)
TXDX	(Convert TX990 Diskette File to DX10 File)
TXFD	(Format Diskette)
TXMD	(Map Diskette)
TXSF	(Set System File)

UV

8.176 UV (UNLOAD VOLUME)

You must unload installed disk volumes by entering the UV command before you remove them from a disk unit. Removing a loaded volume without first entering a UV command to unload it prevents other users from using the disk unit. After you unload the volume, DX10 displays a message indicating which disk unit has been unloaded.

NOTE

The DS10, CD1400/32, and CD1400/96 disk drives are physically organized as a removable platter and a nonremovable platter turned by a single disk spindle. When you load or unload the removable platter, no one can access the nonremovable platter.

8.176.1 UV Command Format

```
UV
UNLOAD VOLUME
VOLUME NAME: < name> @
```

8.176.2 UV Command User Responses

System Prompt	Response Required or Optional	User Response
VOLUME NAME:	R	Enter the name of the installed volume that is being unloaded.

8.176.3 UV Command Example

The following example shows how you can enter the UV command to unload a volume named VOL1 installed in DS02.

```
[ ] UV
UNLOAD VOLUME
  VOLUME NAME: VOL1
UNLOAD DS02
```

8.176.4 Commands Related to UV

CRV	(Check and Reset Volume)
INV	(Initialize New Volume)
IV	(Install Volume)

8.177 VB (VERIFY BACKUP)

The VB command allows you to verify the output of the Backup Directory (BD) or Restore Directory (RD) command. The VB command compares a set of files on a sequential file, magnetic tape, or multiple disks/tapes to a set of files under a given directory on a disk file to see which files in each set match. The VB command detects matches by comparing the file type, file use, file name, and file contents of files at a device or copied to a file that you specified. When the backup you want to verify is a multivolume disk file, enter one or more disk device names as the sequential file input. The disk volumes all have the same volume name and contain a file having the same name. They must be verified in the same order in which they were written by the BD command.

The VB command does not verify the directory named S$PRINT and the files named .S$CRASH, .S$DIAG, and .S$ROLLA unless you specify them in INCLUDE statements.

NOTE

The VB command does not proceed to the next directory after verifying a backup if the directories to be verified are on magnetic tape. Optionally, you can use MOVE directives in a control file to make VB proceed. However, if you perform separate BD commands, you must perform separate VB commands to verify each backup.

If a hardware error was encountered on the source disk while backing up a file, that file cannot be verified. A message informing you of this condition is written to the listing file. Any files not verified because of disk errors encountered during the backup process are listed in a bad file summary on the listing.

8.177.1 VB Command Format

```
VB
VERIFY BACKUP
SEQUENTIAL ACCESS NAME: [(acnm)]@
DIRECTORY PATHNAME: [acnm]@
CONTROL ACCESS NAME: [acnm]@
LISTING ACCESS NAME: [acnm]@
OPTIONS: [(string)]@
EXECUTION MODE (F,B): < F/B>
```

VB

8.177.2 VB Command User Responses

System Prompts	Response Required or Optional	User Responses
SEQUENTIAL ACCESS NAME:	O	Enter the device name(s) or the pathname of a sequential file that contains the backup copy. The VB command compares the set of files in those results against the set of files specified by the DIRECTORY PATHNAME prompt. When you specify DSxx as the sequential access name, the option responds with the following prompt: PATHNAME OF BACKUP FILE: PATHNAME: < acnm> @ The pathname consists of a volume name and a file name. The pathname that you specify must be the pathname used in the backup operation (BD command).
DIRECTORY PATHNAME:	O	Enter a pathname identifying the top-level directory containing a set of files to be compared against the set of backup files identified by the response to the SEQUENTIAL ACCESS NAME prompt.
CONTROL ACCESS NAME:	O	Enter the device name or the file pathname of a file from which the VB command reads directives that limit the files in the master directory to be compared against the copy directory. Directives in the control file take precedence over prompt responses. (For more information on the directives that you can include in control files, refer to Section 7.) If you do not enter a control access name, you must specify both an input pathname and an output pathname.
LISTING ACCESS NAME:	O	Enter the device name or the pathname to which DX10 should list a summary of the results of the verify operation. *The default value is the terminal local file (TLF).*

System Prompts	Response Required or Optional	User Responses
OPTIONS:	O	The following options can be entered to control the verify backup operation. The default options are ALIASES, NODATE, NOREWIND, NOUNLOAD, and SYSFILE.

Option	Purpose
ALIASES	Entering ALIASES specifies that all aliases associated with the files are also to be verified.
NOALIASES	Entering NOALIASES specifies that no aliases are to be verified.
DATE	Entering DATE limits the verification to those files created or updated on or after the specified date. The date and time (year, month, day, hour, minute) of the oldest file are requested: YEAR: <int> MONTH: <int> DAY: <int> HOUR: [int] MINUTE: [int]
NODATE	Entering NODATE indicates no date constraints on the verification.
REWIND	Entering REWIND causes rewinding of tape volumes before they are used.
NOREWIND	Entering NOREWIND indicates tape volumes are not rewound before use.
UNLOAD	Entering UNLOAD indicates tape volumes are unloaded after use.
NOUNLOAD	Entering NOUNLOAD indicates tape volumes are not unloaded after use.

VB

System Prompts	Response Required or Optional	User Responses
	Option	Purpose
	SYSFILE	Entering SYSFILE indicates that system files are included in the verification of the backup operation. These files are .SCI990, .S$SYSLIB, .S$ERROR, .S$CNTRY, .S$SDS$, .S$OVLYA, .S$PROC, .S$IMAGES, .S$TCALIB, .S$FGTCA, .S$BGTCA, .S$SLG1, .S$SLG2, .S$NEWS, .S$LOADER, and .PATCH. The SYSFILE and NOSYSFILE options only apply when using the system disk.
	NOSYSFILE	Entering NOSYSFILE indicates that system files are not included in the verification of the backup operation.
EXECUTION MODE (F,B)	R	Enter F (foreground) to execute the task in foreground mode. Enter B (background) to execute the task in background mode. *The initial value is FOREGROUND.*

8.177.3 VB Command Examples

The following examples show how you can enter the VB command to verify a backup operation. The first example verifies the output to DS04 of one of the BD command examples given in the BD command description.

```
  VERIFY BACKUP            11:18:36 FRIDAY, APRIL 30, 1982.

ORIGINAL SOURCE:        DS04
ORIGINAL DESTINATION:   PAE29.HW.DIR
ORIGINAL OPTIONS:       ALIASES, NODATE, NOREWIND, NOUNLOAD, SYSFILE, NOFAST
CONTROL FILE:
LIST FILE:              PAE29.HW.VBDISK


  **        DIRECTORY  PAE29.HW.DIR
  **   FILE2
  **   FILE1
  **        END OF DIRECTORY  PAE29.HW.DIR


ELAPSED TIME = 0 MINUTES     4 SECONDS
SIZE OF INPUT = 0 ADU'S

****** VERIFY BACKUP COMPLETED
```

The second example shows how you can use the VB command with MOVE directives. The control file for this operation appears first.

```
MOV ,PAE29.HW.DIR
MOV ,PAE29.HW.DIR.SUBDIR
END


  VERIFY BACKUP            11:14:37 FRIDAY, APRIL 30, 1982.

ORIGINAL SOURCE:         PAE29.HW.SEQ
ORIGINAL DESTINATION:
ORIGINAL OPTIONS:        ALIASES, NODATE, NOREWIND, NOUNLOAD, SYSFILE, NOFAST
CONTROL FILE:            ST01
LIST FILE:               PAE29.HW.VBOPT

MOV ,PAE29.HW.DIR

  **        DIRECTORY  PAE29.HW.DIR
  **  FILE1
  **        END OF DIRECTORY  PAE29.HW.DIR

MOV ,PAE29.HW.DIR.SUBDIR

  **        DIRECTORY  PAE29.HW.DIR.SUBDIR
  **  PFILE     - PROGRAM FILE

  **  TASK        ID
  **  FUTIL      >0B

  **  OVERLAY     ID
  **  FUTOV0     >60
  **  FUTOV1     >61
  **  FUTOV2     >62
  **  FUTOV3     >63
  **  FUTOV4     >64
  **  FUTOV5     >65
  **  FUTOV6     >66
  **  FUTOV7     >67
  **  FUTOV8     >68
  **  FUTOV9     >69
  **  FUTOVA     >6A
  **  FUTOVB     >6B
  **        END OF PROGRAM FILE

  **        END OF DIRECTORY  PAE29.HW.DIR.SUBDIR

END

ELAPSED TIME = 0 MINUTES   46 SECONDS
SIZE OF INPUT = 53 ADU'S

****** VERIFY BACKUP COMPLETED
```

8.177.4 Commands Related to VB

BD (Backup Directory)
RD (Restore Directory)

VC

8.178 VC (VERIFY COPY)

The VC command compares a file set under a master directory against a file set under a copy directory to determine which files in each set match. The VC command detects matches by comparing the file type, file use, file name, and file contents of files at corresponding levels of each set. The results of the verify operation are listed at a device or copied to a file that you specify.

The VC command does not verify the directory named .S$PRINT or the files named .S$DIAG, .S$ROLLA, and .S$CRASH unless you specify them in INCLUDE statements.

If you specify a control file with the VC command, it limits the files in the master directory that are compared against files in the copy directory in the same manner as it would limit a copy operation of the Copy Directory (CD) command. Aliases are compared unless the control file specifies a NOALIASES directive.

NOTE

Although a control file, a master pathname, and a copy pathname are all optional responses to a prompt, you must specify a control file if you do not provide either of the other responses. You must specify both a master pathname and copy pathname if you do not specify a control file.

8.178.1 VC Command Format

```
VC
VERIFY COPY
MASTER PATHNAME: [acnm]@
COPY PATHNAME: [acnm]@
CONTROL ACCESS NAME: [acnm]@
LISTING ACCESS NAME: [acnm]@
OPTIONS: (string)
EXECUTION MODE (F,B): < F/B>
```

8.178.2 VC Command User Responses

System Prompts	Response Required or Optional	User Responses
MASTER PATHNAME:	O	Enter a pathname identifying the directory that is the node of a master set of files against which the set of files specified by the copy pathname are compared. *If you do not enter a master pathname, you must specify a control access name.*

System Prompts	Response Required or Optional	User Responses
COPY PATHNAME:	O	Enter a pathname identifying a directory that is the node of a set of files to be compared against the set of files specified by the master pathname. *If you do not enter a copy pathname, you must specify a control access name.*
CONTROL ACCESS NAME:	O	Enter the device name or the file pathname of a file from which the VC command reads directives that limit the files in the master directory that are compared against the copy directory. Directives in the control file take precedence over prompt responses. (For more information on the directives that you can include in a control file, refer to Section 7.) If you do not specify a control access name, you must specify both an input pathname and an output pathname.
LISTING ACCESS NAME:	O	Enter the device name or the file pathname to which DX10 is to write a summary of the results of the verify operation. *The default value is the terminal local file (TLF).*
OPTIONS:	O	You can enter the following options to control the verify copy operation. The default options are ALIASES, NODATE, NOSPRL, NORPRL, and SYSFILE.

Option	Purpose
ADD	You can specify the ADD option, but it is ignored by the verify operation.
REPLACE	You can specify the REPLACE option, but it is ignored by the verify operation.
ALIASES	Entering ALIASES specifies that all aliases associated with the files are also to be verified.
NOALIASES	Entering NOALIASES specifies that no aliases are to be verified.

VC

System Prompts	Response Required or Optional	User Responses
	Option	**Purpose**
	DATE	Entering DATE limits the verification to those files created or updated on or after the specified date. The date and time (year, month, day, hour, minute) of the oldest file are requested. YEAR: <int> MONTH: <int> DAY: <int> HOUR: [int] MINUTE: [int]
	NODATE	Entering NODATE indicates no date constraints on the verification.
	SPRL	Entering SPRL indicates that the sequential files were converted to a new physical record size. If SPRL was specified in the CD command, you must also specify it in the VC command.
	NOSPRL	Entering NOSPRL indicates that the sequential files were not converted to a new physical record size.
	RPRL	Entering RPRL indicates that the relative record files were converted to a new physical record size. If RPRL was specified in the CD command, you must also specify it in the VC command.
	NORPRL	Entering NORPRL indicates that the relative record files were not converted to a new physical record size.

System Prompts	Response Required or Optional	User Responses
	Option	Purpose
	SYSFILE	Entering SYSFILE indicates that system files are included in the verification of the copy of the directory. These files are .SCI990, .S$SYSLIB, .S$ERROR, .S$CNTRY, .S$SDS$, .S$OVLYA, .S$PROC, .S$IMAGES, .S$TCALIB, .S$FGTCA, .S$BGTCA, .S$SLG1, .S$SLG2, .S$NEWS, .S$LOADER, and .PATCH. The SYSFILE and NOSYSFILE options only apply when using the system disk.
	NOSYSFILE	Entering NOSYSFILE indicates that system files are not included in the verification of the copy of the directory.
EXECUTION MODE (F, B):	R	Enter F (foreground) to execute the task in foreground mode. Enter B (background) to execute the task in background mode. *The initial value is FOREGROUND.*

8.178.3 VC Command Example

The following example shows how you can enter the VC command to verify the copy operation that was described in the CD command description:

```
  VERIFY COPY                11:22:12 FRIDAY, APRIL 30, 1982.

ORIGINAL SOURCE:          PAE29.HW.DIR
ORIGINAL DESTINATION:     CART.DATADIR
ORIGINAL OPTIONS:         ALIASES, NODATE, SYSFILE
CONTROL FILE:
LIST FILE:                PAE29.HW.VCLIST


  **        DIRECTORY   PAE29.HW.DIR

  **        DIRECTORY   PAE29.HW.DIR.SUBDIR
  **   PFILE     - PROGRAM FILE

  **   TASK         ID
  **   FUTIL       >0B

  **   PROCEDURE   ID
```

VC

```
**  OVERLAY     ID
**  FUTOV0     >60
**  FUTOV1     >61
**  FUTOV2     >62
**  FUTOV3     >63
**  FUTOV4     >64
**  FUTOV5     >65
**  FUTOV6     >66
**  FUTOV7     >67
**  FUTOV8     >68
**  FUTOV9     >69
**  FUTOVA     >6A
**  FUTOVB     >6B
**        END OF PROGRAM FILE
```

8.178.4 Commands Related to VC

CD (Copy Directory)

8.179 WAIT (WAIT FOR BACKGROUND TASK TO COMPLETE)

DX10 does not display background command termination messages except after termination of foreground commands or when you enter the SBS command. When you do not desire foreground activity, you can enter the WAIT command to cause background reporting without constantly entering SBS. WAIT displays the following message until the background task terminates:

—WAITING FOR BACKGROUND TASK TO COMPLETE —

To abort the WAIT command and continue foreground processing, press the CMD key. The WAIT command displays the background TLF, if it exists, when the background task completes.

8.179.1 WAIT Command Format

WAIT

8.179.2 WAIT Command User Responses

System Prompt	Response Required or Optional	User Response
MESSAGE?:	O	MESSAGE? is a hidden prompt that you can use in expert mode for interactive procedures. YES is the default for this prompt. If a procedure executing in foreground executes more than one background task, you should use NO to avoid the required response to the BACKGROUND COMPLETE message. That is, the foreground command procedure can execute a second background task after waiting for a previously executed background task to complete without interruption.

8.179.3 WAIT Command Example

The following example shows how you can use the WAIT command to inform you when a background task is complete:

[ ] WAIT

8.179.4 Commands Related to WAIT

KBT (Kill Background Task)
SBS (Show Background Status)

WCRU

8.180 WCRU (WRITE VALUE TO SPECIFIED CRU ADDRESS)

The WCRU command allows you to write a specified number of bits to the specified CRU address.

8.180.1 WCRU Command Format

```
WCRU
WRITE VALUE TO SPECIFIED CRU ADDRESS
CRU ADDRESS: < int>
NUMBER OF BITS: < int>
VALUE TO BE WRITTEN: < int>
```

8.180.2 WCRU Command User Responses

System Prompts	Response Required or Optional	User Responses
CRU ADDRESS:	R	Enter the CRU address to which to write. (This value must be less than or equal to >1FE0 and not in the range >1F00 - >1FDF.)
NUMBER OF BITS:	R	Enter the number of bits to be written. If the value is zero, 16 bits are written. Otherwise, this must be a positive integer that is less than or equal to fifteen.
VALUE TO BE WRITTEN:	R	Enter an integer value to be written to the specified CRU address.

8.180.3 WCRU Command Example

The following example shows how you can write 16 bits of information to the specified CRU address >0100.

```
[ ] WCRU
WRITE VALUE TO SPECIFIED CRU ADDRESS
  CRU ADDRESS: >0100
  NUMBER OF BITS: 0
  VALUE TO BE WRITTEN: >4141
```

8.180.4 Commands Related to WCRU

RCRU (Read Contents of Specified CRU Address)

8.181 WEOF (WRITE EOF TO LUNO)

DX10 accesses information on sequential files and sequential devices on a record-by-record basis. You can write an end-of-file (EOF) marker at the current data-access position by entering the WEOF command.

8.181.1 WEOF Command Format

```
WEOF
WRITE EOF TO LUNO
LUNO: < int>
```

8.181.2 WEOF Command User Responses

System Prompt	Response Required or Optional	User Response
LUNO:	R	Enter the logical unit number (LUNO) of the sequential device or file to which you want to write the EOF.

8.181.3 WEOF Command Example

The following example shows how you can enter the WEOF command to write an EOF to a magnetic tape unit assigned to LUNO 133:

```
[ ] WEOF
WRITE EOF TO LUNO
  LUNO: 133
```

8.181.4 Commands Related to WEOF

BL	(Backspace LUNO)
FL	(Forward Space LUNO)
RWL	(Rewind LUNO)

8.182 XANAL (EXECUTE CRASH ANALYZER)

The ANALZ utility is available to analyze system crash dumps. The SCI command XANAL invokes the utility, which then produces a formatted version of the crash file. You access the formatted crash information by using the XANAL auxiliary commands. These commands, which are listed in Table 8-17, enable an experienced DX10 programmer to detect and correct system logic errors.

If ANALZ is running in batch mode, with a file or sequential input device (other than a terminal) as the control access name, each input value or command must start in column one of a separate record (card). By using files for ANALZ commands, you can keep a standard ANALZ command stream on file or cards. This way you can quickly and easily execute the ANALZ utility after every system crash.

NOTE

You can initiate the XANAL command with lowercase letters, but you must enter alphabetic responses to command prompts in uppercase letters. SCI does not perform lowercase to uppercase mapping for prompt responses to this command, as explained in the .OPTION primitive in Volume III.

Crash analysis is very difficult, and you need a great deal of knowledge about the system to determine the underlying causes of a system crash. Refer to Volume VI for information on how to read the mass of tables and structures the ANALZ utility creates.

8.182.1 XANAL Command Format

```
[ ] XANAL
ANALYZE CRASH FILE
CONTROL ACCESS NAME: < acnm> @
LISTING ACCESS NAME: [acnm]@
ANALYZE RUNNING SYSTEM?: < yes/no>
DISK DEVICE NAME: < name> @
CRASH-FILE NAME: < name> @
```

8.182.2 XANAL Command User Responses

System Prompts	Response Required or Optional	User Responses
CONTROL ACCESS NAME:	R	Enter a device name or the file pathname from which you enter auxiliary commands to the XANAL command.
LISTING ACCESS NAME:	O	Enter the device name or the file pathname to which DX10 should write the specified contents of the system crash file. The default is the terminal local file (TLF).
ANALYZE RUNNING SYSTEM:	R	Enter YES to specify that DX10 is to list an analysis of the running system rather than the contents of the crash file. Enter NO to specify that DX10 is to list the specified contents of the system crash file.
DISK DEVICE NAME:	R	Enter the device name of the disk unit on which the system crash file resides.
CRASH-FILE NAME:	R	Enter a name that identifies the file containing the crash dump to be analyzed.

After you respond to the prompts, the following prompt appears in TTY mode (one line at a time) from the bottom of your screen:

```
CRASH DUMP ANALYZER - DX10 3.5
COMMAND?
```

You are now ready to enter the command most appropriate to the error you are dealing with. Table 8-17 gives a brief description of the auxiliary XANAL commands. You can find a detailed description of each auxiliary command and the results obtained by its use in Volume VI.

XANAL

Table 8-17. Auxiliary Commands Used with the XANAL Command

Command	Purpose
AL	Perform the following auxiliary commands in the indicated order: GI, TS, SS, MM, AQ, PQ, TR, TA, and a DM with 0, 0, and > FFFF as parameters.
AM	Perform the same commands as AL, omitting TA.
AQ	List a representation of the four active queues.
DI	List disk information (simple map disk function).
DM	Dump a specific area of memory.
FC	List the file control blocks (FCBs) for all currently assigned files.
GI	List general information about the system crash.
LD	List the logical device tables (LDTs) for all assigned LUNOs in the system.
MM	List a map of memory.
PB	List the partial bit maps (PBMs) for all installed disk volumes.
PD	List the physical device tables (PDTs) for all devices in the system.
PQ	List a representation of the other system queues.
PS	List the procedure status blocks (PSBs) for all procedures in the system.
QU	Terminate execution of the ANALZ command.
SA	Show system table area.
SS	Write memory images of the system structures. (Perform TB, PS, PD, FC, PB, and LD commands in that order.)
TA	List memory images of every task area in memory.
TB	Lists the task status blocks (TSBs) for all tasks in the system.
TR	List the workspace register contents of all tasks in memory.
TS	List the task state of all tasks in memory.

8.182.3 XANAL Command Examples

The following examples show various uses of the XANAL command. The first example shows how you can enter the XANAL command and the AQ auxiliary command to list general information about a system crash and active queries.

Example 1

```
[ ] XANAL
ANALYZE CRASH FILE
  CONTROL ACCESS NAME: ME
  LISTING ACCESS NAME: ME
  ANALYZE RUNNING SYSTEM?: NO
  DISK DEVICE NAME: DS01
  CRASH-FILE NAME: S$CRASH
```

Accepting the defaults for all of these prompts results in the following:

```
CRASH DUMP ANALYZER - DX10 3.5
COMMAND?
```

Enter AQ to display the following:

```
ACTIVE QUEUE
NEWEST    OLDEST    TSB     FLAG    STATE    COUNTERS    TSBS
0000      0000      0000    4000    0000     644A
0000      0000      0000    4000    0100     640E
0000      0000      0000    4000    0200     645F
2EAA      2B8E      0000    4000    0302     6439
                                                         2B8E
                                                         2EAA

COMMAND?
```

To return to the main menu, enter QU in response to the COMMAND? prompt.

Example 2

The following example shows how you can use the XANAL command to produce an ANALZ dump to be analyzed:

```
[ ] XANAL
ANALYZE CRASH FILE
  CONTROL ACCESS NAME: ME
  LISTING ACCESS NAME: LP01
  ANALYZE RUNNING SYSTEM?: NO
  DISK DEVICE NAME: DS01
  CRASH-FILE NAME: S$CRASH
```

XANAL

Accepting the default for all prompts gives you the following:

```
CRASH DUMP ANALYZER - DX10 3.5
COMMAND?
```

Respond with AL to create a dump and system link map that should be sufficient to allow you to identify the problem. To abort the dump, press the CMD key and respond QU to the prompt COMMAND?. If you cannot identify the problem, the resulting dump and the system link map should be sufficient to allow your dealer or customer representative to identify the problem.

If you send the dump to TI, copy the .S$CRASH file with a CD or BD command and copy the link map with the CC command. Put these copies on double-sided, double-density (DSDD) diskette, hard disk, or magnetic tape media. This allows access to all memory structures for a complete analysis.

8.183 XB (EXECUTE BATCH)

The XB command activates SCI to run in background mode. Once activated, SCI runs without interacting with the terminal that initiated it until finished, at which time it sends an ending message to the terminal. You can create a batch command stream by using the Text Editor.

8.183.1 XB Command Format

```
XB
EXECUTE BATCH
INPUT ACCESS NAME: < acnm> @
LISTING ACCESS NAME: < acnm> @
```

8.183.2 XB Command User Responses

System Prompts	Response Required or Optional	User Responses
INPUT ACCESS NAME:	R	Enter the device name or the file pathname from which DX10 should read the batch command stream.
LISTING ACCESS NAME:	R	Enter the device name or the file pathname to which DX10 should write the results of the batch command stream execution. This device or file must not be used by any command in the batch command stream.

8.183.3 XB Command Example

The following example shows how you can enter the XB command to execute a batch command stream from a file and to have the output of the batch command stream sent to a file. The example also shows the printing of an ending message upon termination of the batch.

```
[ ] XB
EXECUTE BATCH
  INPUT ACCESS NAME: MY.BATCH
  LISTING ACCESS NAME: MY.LIST

[ ] WAIT
BACKGROUND EXECUTION HAS COMPLETED
```

8.183.4 Commands Related to XB

BATCH (Begin Batch Execution)
EBATCH (End Batch Execution)

XCU

8.184 XCU (EXECUTE 2.2 TO 3.0 DISK CONVERSION)

The XCU command activates a conversion utility that transfers data files from a DX10 2.X disk to a release 3 or later version disk in a multiple disk configuration. Refer to Volume V for a detailed description of DX10 2.X disk conversion.

When converting 2.X files, you have the option of placing the 2.X files under a release 3 or later version directory or simply under the volume catalog of the disk receiving the files. By being cataloged under the volume catalog, the 2.X files retain the same pathnames as on the 2.X disk, a useful characteristic when using programs developed under DX10 2.X. However, placing the 2.X files under a newly created release 3 or later version directory avoids the possibility of pathname conflicts.

If you want to catalog the 2.X files under a new directory, the directory must exist before executing the conversion utility. In addition, the pathname of the directory may not exceed 27 characters in length, including periods.

Whenever the conversion utility converts a 2.X file, the pathname of the release 3 or later version file is the old 2.X pathname appended to whatever you responded to the OUTPUT DISK NAME? prompt. The following table shows how the conversion utility names converted files.

User Response to OUTPUT DISK NAME?	New Pathname of 2.X File UFD1.LIB1 (MEM1)
< default>	.UFD1.LIB1.MEM1
.MYCAT	.MYCAT.UFD1.LIB1.MEM1
DS03.MYCAT.DX2XFILE	DS03.MYCAT.DX2XFILE.UFD1.LIB1.MEM1
DS02	DS02.UFD1.LIB1.MEM1

NOTE

For compatibility purposes, DX10 release 3 or later versions support the 2.X convention of enclosing a library member name in parentheses, rather than delimiting it with a period. For example, the file in Example 2 can be referred to as the following:

.MYCAT.UFD1.LIB(FILEA)

8.184.1 XCU Command Format

XCU
EXECUTE 2.2 TO 3.0 DISK CONVERSION
DISK TO BE CONVERTED: < acnm>
LISTING DEVICE: < acnm>

After you answer these prompts, the following prompt appears:

OUTPUT DISK NAME?: < acnm>

The following prompt appears after you specify the output disk name:

CONVERT FILE < pathname> (Y/N/A/S)? < Y/N/A/S>

8.184.2 XCU Command User Responses

System Prompts	Response Required or Optional	User Responses
DISK TO BE CONVERTED:	R	Enter the device name of the disk drive that contains the 2.X disk.
LISTING DEVICE:	R	Enter the device name of a device or the pathname of a file to which the utility should send any error messages and the log of converted files.
OUTPUT DISK NAME?:	R	Enter the access name of a disk or a directory under which the 2.X files are to be cataloged. The default value is the system disk, with all 2.X files and libraries cataloged under the volume catalog (VCATALOG) with their 2.X pathnames. You can specify any disk in the system, and any existing release 3 or later version directory hierarchy under which the 2.X files should be cataloged, provided that the directory hierarchy has been previously defined.

XCU

System Prompts	Response Required or Optional	User Responses
CONVERT FILE < pathname> (Y/N/A/S)?	R	This prompt provides you with the option of converting all 2.X files or only specific files. Respond with one of the following:

Response	Action
Y (yes)	Converts this file (< pathname>). The file is converted (if possible), and the utility prompts for the next file.
N (no)	Does not convert this file. The utility prompts for the next file.
A (all)	Converts all files on the directory specified by < pathname>. The utility terminates.
S (stop)	Terminates the conversion utility.

8.184.3 XCU Command Example

The following example shows how you can use the XCU command to convert all files on DS02:

```
[ ] XCU
EXECUTE 2.2 TO 3.0 DISK CONVERSION
  DISK TO BE CONVERTED: DS02
  LISTING DEVICE: LP01
  OUTPUT DISK NAME?: .MYCAT
  CONVERT FILE DS02.ADD (Y/N/A/S)? A
```

8.185 XD (EXECUTE DEBUGGER)

The XD command places a specified task into controlled mode. Refer to Volume III for a detailed description.

XE

8.186 XE (EXECUTE TEXT EDITOR)

The XE command activates the Text Editor. Refer to Volume IV for a detailed description.

8.187 XES (EXECUTE TEXT EDITOR WITH SCALING)

The XES command is identical to the XE command, except that when the terminal is in VDT mode, scaling information is displayed on the bottom line of the screen. Refer to Volume IV for a detailed description.

XGEN

8.188 XGEN (EXECUTE SYSTEM GENERATION UTILITY)

The XGEN command initiates GEN990, the system generation utility, at your terminal. The XGEN command prompts you to aid GEN990 in the definition of the computer system by supplying the operating parameters. Then, GEN990 builds the data files that are used by the remainder of the system generation process. You should refer to Volume V for an extensive discussion of system generation.

Volume V discusses the following topics:

- Files created and used by GEN990
- GEN990 modes of operation (command and inquiry)
- GEN990 commands

8.188.1 XGEN Command Format

```
XGEN
EXECUTE SYSTEM GENERATION UTILITY
DATA DISK/VOLUME: < acnm> @
INPUT CONFIGURATION: [name]@
OUTPUT CONFIGURATION: < name> @
```

8.188.2 XGEN Command User Responses

System Prompts	Response Required or Optional	User Responses
DATA DISK/VOLUME:	R	Enter the device name or volume name of the disk drive that stores the system generation directory .S$SYSGEN.
INPUT CONFIGURATION:	O	If this is the first generation of a particular configuration, press the RETURN key without entering any response. If you are regenerating an existing configuration, enter that configuration's unique name and press the RETURN key.

System Prompts	Response Required or Optional	User Responses
OUTPUT CONFIGURATION:	R	If this is the first generation of a particular configuration, enter a unique name. This name, identifying the configuration, must begin with an alphabetic character and contain no more than five characters. If you are regenerating an existing configuration, you can enter that configuration's name and the new configuration replaces the old one. You can preserve the existing configuration and create a new one by entering a new name. **CAUTION** **Do not specify the system from which you currently performed an initial program load (IPL) as the output configuration.**

When you respond to these prompts, GEN990 displays the following heading:

= = = = = = = = = GEN990-SYSTEM GENERATION 3.5.0 = = = = = = = = =

GEN990 continues by prompting you for the parameter values in TTY mode.

8.188.3 XGEN Command Example

For an example of using the XGEN command, refer to Volume V.

8.188.4 Commands Related to XGEN

ALGS	(Assemble and Link Generated System)
IGS	(Install Generated System)
PGS	(Patch Generated System)
TGS	(Test Generated System)
XPS	(Execute Patch Synonym Processor)

XHT

8.189 XHT (EXECUTE AND HALT TASK)

The XHT command places a task in memory in a suspended state so that you can debug it. Typically, you place the task to be debugged in memory using XHT, establish the debug environment (including breakpoints), and then activate the task using the Activate Task (AT) or Resume Task (RT) command.

8.189.1 XHT Command Format

```
XHT
EXECUTE AND HALT TASK
PROGRAM FILE OR LUNO: < string> @
TASK NAME OR ID: < string> @
PARM1: < int>
PARM2: < int>
STATION ID: < string>
```

8.189.2 XHT Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter a pathname or LUNO that identifies the program file on which the program is installed. Enter a value of zero to specify the system program file.
TASK NAME OR ID:	R	Enter either the name or the ID number under which the program is installed.
PARM1: PARM2:	R	Enter decimal or hexadecimal numbers in the range of 0 through 65535 representing a value to be passed to the program. The initial value is 0.
STATION ID:	R	Enter the number (> 02, not ST02) of the station with which the executing task is to be associated. Enter 0 (zero) or ME to specify your own station. Enter > FF to specify that the task is not associated with any station on the system. The initial value is ME.

8.189.3 XHT Command Example

The following example shows how you can enter the XHT command to activate a task with an installed ID of > A5 from your program file and to associate that executing task with station > 02. As the example shows, DX10 lists the run-time ID that it automatically assigns when it activates the program.

```
[ ] XHT
EXECUTE AND HALT TASK
  PROGRAM FILE OR LUNO: .USERPROG
  TASK NAME OR ID: > A5
  PARM1: 0
  PARM2: 0
  STATION ID: > 02
RUNTIME TASK ID  =  > 3A
```

8.189.4 Commands Related to XHT

RT	(Resume Task)
XT	(Execute Task)
XTS	(Execute Task and Suspend SCI)

XLE

8.190 XLE (EXECUTE LINK EDITOR)

The XLE command executes the Link Editor. Refer to the *DX10 Link Editor Reference Manual* for a detailed description.

8.191 XMA (EXECUTE MACRO ASSEMBLER)

The XMA command assembles the source code of an assembly language program. You can assemble modules of the program separately and later link them together, or you can include all of the code in one module. Refer to the *990/99000 Assembly Language Reference Manual* for a detailed description.

XPS

8.192 XPS (EXECUTE PATCH SYNONYM PROCESSOR)

Occasionally, when patching the operating system using patches from a patch release, you may want to execute the patch synonym assignment program without assembling and linking the system again. You can accomplish this by using the XPS command.

8.192.1 XPS Command Format

```
XPS
EXECUTE PATCH SYNONYM PROCESSOR
LINK: < acnm> @
INPUT: < acnm> @
OUTPUT: < acnm> @
ERROR: < acnm> @
```

8.192.2 XPS Command User Responses

System Prompts	Response Required or Optional	User Responses
LINK:	R	Enter the pathname of a file that contains the link map for the program or system to be patched. It must be an image format link. For a system, use the pathname .S$SYSGEN.< system name> .LINKMAP.
INPUT:	R	Enter the pathname of the patch file that contains the patches that must be relocated before being run. An example is .PATCH.MEMRES.
OUTPUT:	R	Enter the pathname of the file that contains the relocated patches. It can be run as a batch stream. An example is .S$SYSGEN.<system name>.PATCHFIL.
ERROR:	R	Enter the pathname of the error file. If any abnormal conditions are detected, the conditions are noted in this file. An example is .S$SYSGEN.< system name> .ERROR.

NOTE

If you receive the message DIFFERENT SYNONYMS FOR SAME OVERLAYS, it is an informative warning, not a fatal error. Usually, this message does not indicate difficulty.

8.192.3 XPS Command Example

The following example shows how you can use the XPS command to execute the patch synonym assignment program without assembling and linking the system again:

```
[ ] XPS
EXECUTE PATCH SYNONYM PROCESSOR
  LINK: .S$SYSGEN.MAR04.LINKMAP
  INPUT: .PATCH.MEMRES
  OUTPUT: .S$SYSGEN.MAR04.PATCHFIL
  ERROR: .S$SYSGEN.MAR04.ERROR
```

8.192.4 Commands Related to XPS

PGS	(Patch Generated System)
XGEN	(Execute System Generation Utility)

XT

8.193 XT (EXECUTE TASK)

The XT command activates a program that must not interact with a terminal where SCI is active (for example, your terminal) since XT does not suspend SCI. Two 16-bit words of information can be passed to the program being activated in response to the PARM1 and PARM2 prompts. DX10 automatically assigns a run-time ID to each program that it activates. It displays or prints the run-time ID at your terminal upon successful activation of the program. If you use this command to activate a task, the task has to contend with SCI for event characters and other I/O at the station where it performs I/O unless SCI is deactivated at that station.

8.193.1 XT Command Format

```
XT
EXECUTE TASK
PROGRAM FILE OR LUNO: < string> @
TASK NAME OR ID: < string> @
PARM1: < int>
PARM2: < int>
STATION ID: < string>
```

8.193.2 XT Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter a pathname or LUNO that identifies the program file on which the program is installed. Enter a value of zero to specify the system program file.
TASK NAME OR ID:	R	Enter either the name or the ID number under which the program is installed.
PARM1: PARM2:	R	Enter decimal or hexadecimal numbers in the range of 0 through 65535 representing a value to be passed to the program. The initial value is 0.
STATION ID:	R	Enter the number (>02, not ST02) of the station with which the executing task is to be associated. Enter 0 (zero) or ME to specify your own station. Enter >FF to specify that the task is not associated with any station on the system. The initial value is ME.

8.193.3 XT Command Example

The following example shows how you can enter the XT command to activate a task with an installed ID of >A5 from your program file and to associate that executing task with station >02. As the example shows, DX10 lists the run-time ID that it automatically assigns when it activates the program.

```
[ ] XT
EXECUTE TASK
  PROGRAM FILE OR LUNO: .USERPROG
  TASK NAME OR ID: >A5
  PARM1: 0
  PARM2: 0
  STATION ID: >02
RUNTIME TASK ID = >3A
```

8.193.4 Commands Related to XT

XTS (Execute Task and Suspend SCI)

XTS

8.194 XTS (EXECUTE TASK AND SUSPEND SCI)

The XTS command activates an interactive program at your station. The XTS command automatically suspends SCI at the initiating terminal to prevent it from interfering with the execution of the interactive program. If you associate the task with a terminal other than the initiating terminal, XTS does not suspend SCI at the associated terminal but it does suspend SCI at the initiating terminal. The associated terminal should not have SCI active if the task requires terminal I/O to that terminal.

NOTE

The problem with I/O contention is not necessarily with the associated station but with SCI or another interactive program at the terminal to which the task wants to do I/O.

8.194.1 XTS Command Format

XTS
EXECUTE TASK AND SUSPEND SCI
PROGRAM FILE OR LUNO: < string> @
TASK NAME OR ID: < string>
PARM1: < int>
PARM2: < int>
STATION ID: < string>

8.194.2 XTS Command User Responses

System Prompts	Response Required or Optional	User Responses
PROGRAM FILE OR LUNO:	R	Enter a pathname or LUNO that identifies the program file on which the program is installed. Enter a value of zero to specify the system program file.
TASK NAME OR ID:	R	Enter either the name or the ID number under which the program is installed.
PARM1: PARM2:	R	Enter decimal or hexadecimal numbers in the range 0 through 65535 representing a value to be passed to the program.
STATION ID:	R	Enter the number (> 02, not ST02) of the station with which the executing task is to be associated. Enter 0 (zero) or ME to specify your own station. Enter > FF to specify no associated station.

8.194.3 XTS Command Example

The following example shows how you can enter the XTS command to activate an interactive task with an installed ID of > A5 and to have that executing task associated with your own station. As the example shows, DX10 lists the run-time ID that it automatically assigns when SCI is reactivated at the terminal.

```
[ ] XTS
EXECUTE TASK AND SUSPEND SCI
  PROGRAM FILE OR LUNO: .USERPROG
  TASK NAME OR ID: > A5
  PARM1: 0
  PARM2: 0
  STATION ID: ME
RUNTIME TASK ID = > 38
```

8.194.4 Commands Related to XTS

XT (Execute Task)

Appendix A

Task State Codes

A.1 GENERAL

Several supervisor calls (SVCs) return a task state code in byte 1 of the SVC block. The STS as well as several debug commands list the state codes of tasks. Table A-1 defines these codes.

Table A-1. Task State Codes

Code (Hexadecimal)	Definition
00	Active task
01	Active task
04	Terminated task
05	Task in time delay
06	Suspended task
07	Currently executing task
09	Task awaiting completion of I/O
0A	Task awaiting assignment of a device for I/O
0B	Task awaiting disk file utility services
0D	Task awaiting file management services
0E	Task awaiting overlay loader services
0F	Task awaiting initial load
11	Task awaiting disk management services
12	Task awaiting tape management services
13	Task awaiting system overlay loader services
14	Task awaiting task driven supervisor call processor
15	Task awaiting Get Memory services
17	Task suspended for activation of a called task
18	Task awaiting termination task services
19	Task awaiting completion of any I/O
1A	Task awaiting memory management services
1B	Task is eligible for roll-out when requested I/O completes
1C	Task activated while roll is in progress
1D	Suspended for initiate I/O threshold
1E	Suspended for locked directory
1F	Suspended for task management directory buffer
24	Task suspended for queue input
FF	Dummy task state

Appendix B

Global LUNOs

Table B-1 provides a list of the global LUNO assignments.

Table B-1. Global LUNO Assignments

LUNO	Assignment*
00	System Console (ST01)
01	Foreground TCA File S$FGTCA
02	Background TCA File S$BGTCA
03	TCA Library File S$TCALIB
04	***Reserved
05	***Reserved
06	***Reserved
07	News File
08	***Reserved
09	System Disk
0A	S$OVLYA — Overlay File (System LUNO)
0B	S$PROGA — Program File (System LUNO)
0C	***Reserved
0D	S$PROGA — Program File (System LUNO)
0E	S$ROLLA — Roll File (System LUNO)
0F	S$PROGA — Program File System (System LUNO)
10	SSDS — Program File (System LUNO)
11	***Reserved/WT***
12	System Log File S$SLG1
13	System Log File S$SLG2
14	System Log Device
15	System Log Attention Device
16	.S$WP — Word Processing Program File
17	.S$ODIAG.PGM -- Online Diagnostic Program File
18-20	***Reserved
21	Pascal Program File
22	.S$COMMPF — Communications Program File
40	.IDSCOMM.PROGRAM — DX10 HDLC
90	.INDSCOMM.USERPROG — DX10 HDLC
D0-D1	***Reserved
D2	Disk 2
D3	Disk 3
D4-FF	Reserved for System Assignment

Note:

* ***Reserved means reserved for system use.

Appendix C

Hardware Requirements for Remote Terminal Utilities

The Remote Terminal utilities use the TPD (teleprinter device) device service routine (DSR), which operates with the following communications environments:

	TPD DSR Environments		
I/F Card	**Modem or Hardware**	**Terminal**	**Optional ACU**
TTY	Hardwire	743/5	NO
TTY	Hardwire	820 KSR	NO
COMM/TTY	103J/212A	743/5	External 801C
COMM/TTY	103J/212A	763/5	External 801C
COMM/TTY	103J/212A	781/3/5/7	External 801C
COMM/TTY	103J/212A	820 KSR	External 801C
COMM	202S	763/5	External 801C
COMM	202S	781/3	External 801C
COMM	Internal 202	763/5	Internal
COMM	Internal 202	781/3	Internal

Alphabetical Index

Introduction

HOW TO USE INDEX

The index, table of contents, list of illustrations, and list of tables are used in conjunction to obtain the location of the desired subject. Once the subject or topic has been located in the index, use the appropriate paragraph number, figure number, or table number to obtain the corresponding page number from the table of contents, list of illustrations, or list of tables.

INDEX ENTRIES

The following index lists key words and concepts from the subject material of the manual together with the area(s) in the manual that supply major coverage of the listed concept. The numbers along the right side of the listing reference the following manual areas:

- Sections — Reference to Sections of the manual appear as "Sections x" with the symbol x representing any numeric quantity.

- Appendixes — Reference to Appendixes of the manual appear as "Appendix y" with the symbol y representing any capital letter.

- Paragraphs — Reference to paragraphs of the manual appear as a series of alphanumeric or numeric characters punctuated with decimal points. Only the first character of the string may be a letter; all subsequent characters are numbers. The first character refers to the section or appendix of the manual in which the paragraph may be found.

- Tables — References to tables in the manual are represented by the capital letter T followed immediately by another alphanumeric character (representing the section or appendix of the manual containing the table). The second character is followed by a dash (-) and a number.

  Tx-yy

- Figures — References to figures in the manual are represented by the capital letter F followed immediately by another alphanumeric character (representing the section or appendix of the manual containing the figure). The second character is followed by a dash (-) and a number.

  Fx-yy

- Other entries in the Index — References to other entries in the index preceded by the word "See" followed by the referenced entry.

CUT ALONG LINE

USER'S RESPONSE SHEET

Manual Title: DX10 Operating System Operations Guide (Volume II) (946250-9702)

Manual Date: 1 September 1982 **Date of This Letter:** _____

User's Name: _____ **Telephone:** _____

Company: _____ **Office/Department:** _____

Street Address: _____

City/State/Zip Code: _____

Please list any discrepancy found in this manual by page, paragraph, figure, or table number in the following space. If there are any other suggestions that you wish to make, feel free to include them. Thank you.

Location in Manual	Comment/Suggestion
_____	_____
	_____
	_____
	_____
_____	_____
	_____
	_____
	_____
_____	_____
	_____
	_____
	_____

NO POSTAGE NECESSARY IF MAILED IN U.S.A.
FOLD ON TWO LINES (LOCATED ON REVERSE SIDE), TAPE AND MAIL

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 7284 DALLAS, TX

POSTAGE WILL BE PAID BY ADDRESSEE

TEXAS INSTRUMENTS INCORPORATED

DIGITAL SYSTEMS GROUP

ATTN: TECHNICAL PUBLICATIONS
P.O. Box 2909 M/S 2146
Austin, Texas 78769

FOLD

Texas Instruments U.S. District Sales and Service Offices

(A complete listing of U.S. offices is available from the district office nearest your location)

California
831 S. Douglas Street
El Segundo, California 90245
(213) 973-2571

100 California Street
Suite 480
San Francisco, California 94111
(415) 781-9470

776 Palomar Avenue
P.O. Box 9064
Sunnyvale, California 94086
(408) 732-1840*

3186 Airway
Suite J
Costa Mesa, California 92626
(714) 540-7311

Colorado
9725 East Hampden Avenue
Suite 301
Denver, Colorado 80231
(303) 751-1780

Florida
1850 Lee Road
Suite 115
Winter Park, Florida 32789
(305) 644-3535

Georgia
3300 Northeast Expressway
Building 9
Atlanta, Georgia 30341
(404) 458-7791

*Service telephone number

Illinois
515 West Algonquin Road
Arlington Heights, Illinois 60005
(312) 640-2900
(800) 942-0609*

Massachusetts
504 Totten Pond Road
Waltham, Massachusetts 02154
(617) 890-7400

Michigan
24293 Telegraph Road
Southfield, Michigan 48034
(313) 353-0830
(800) 572-8740*

Minnesota
7625 Parklawn Avenue
Minneapolis, Minnesota 55435
(612) 830-1600

Missouri
2368 Schuetz
St. Louis, Missouri 63141
(314) 569-0801*

New Jersey
1245 Westfield Avenue
Clark, New Jersey 07066
(201) 574-9800

Ohio
4124 Linden Avenue
Dayton, Ohio 45432
(513) 258-3877

Pennsylvania
420 Rouser Road
Coraopolis, Pennsylvania 15108
(412) 771-8550

Texas
8001 Stemmons Expressway
P.O. Box 226080
M/S 3108
Dallas, Texas 75266
(214) 689-4460

13510 North Central Expressway
P.O. Box 225214
M/S 393
Dallas, Texas 75265
(214) 238-3881

9000 Southwest Freeway, Suite 400
Houston, Texas 77074
(713) 776-6577

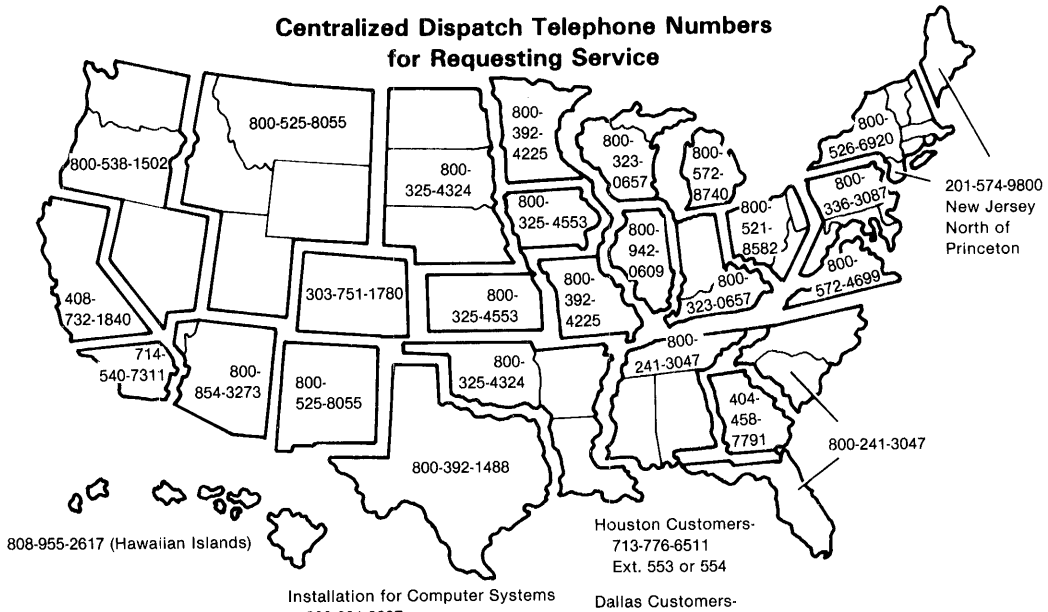
8585 Commerce Drive, Suite 518
Houston, Texas 77036
(713) 776-6531
(713) 776-6553*

Virginia
1745 Jefferson Davis Highway
Crystal Square 4, Suite 600
Arlington, Virginia 22202
(703) 553-2200

Wisconsin
205 Bishops Way
Suite 214
Brookfield, Wisconsin 53005
(414) 784-1323

TI-CARE*

Centralized Dispatch Telephone Numbers for Requesting Service



*Service mark of Texas Instruments

The **TI Customer Support Line** is available to answer our customers' complex technical questions. The extensive experience of a selected group of TI senior engineers and systems analysts is made available directly to our customers. The **TI Customer Support Line** telephone number is **(512) 250-7407.**

TEXAS INSTRUMENTS
INCORPORATED
DIGITAL SYSTEMS GROUP
POST OFFICE BOX 2909 AUSTIN, TEXAS