297-1001-820

DMS-100 Family

Nonmenu Commands
Historical Reference Manual
ABBT Through DRAM, Volume 1 of 4

Through BCS36   Standard 04.01   June 1999

**NORTEL
NETWORKS**
How the world shares ideas.

DMS-100 Family

Nonmenu Commands
Historical Reference Manual-ABBT Through DRAM
Volume 1 of 4

Publication history

June 1999

BCS36 Standard 04.01 Reissued to place book in historical reference.

Contents

About this document

This historical reference manual describes all Nonmenu commands applicable through the BCS36 software load only. These commands are used at a maintenance and administration position (MAP) in a Nortel Networks DMS-100.

When to use this document

Nortel Networks software releases are referred to as batch change supplements (BCS) and are identified by a number, for example, BCS29. This document is written for DMS-100 Family offices that have BCS36 and up.

More than one version of this document may exist. The version and issue are indicated throughout the document, for example, 01.01. The first two digits increase by one each time the document content is changed to support new BCS-related developments. For example, the first release of a document is 01.01, and the next release of the document in a subsequent BCS is 02.01. The second two digits increase by one each time a document is revised and rereleased for the same BCS.

To determine which version of this document applies to the BCS in your office, check the release information in *DMS-100 Family Guide to Northern Telecom Publications*, 297-1001-001.

How to identify the software in your office

The *Office Feature Record* (D190) identifies the current BCS level and the NT feature packages in your switch. You can list a specific feature package or patch on the MAP (maintenance and administration position) terminal by typing

>PATCHER;INFORM LIST identifier
and pressing the Enter key.

where
identifier   is the number of the feature package or patch ID

You can identify your current BCS level and print a list of all the feature packages and patches in your switch by performing the following steps. First, direct the terminal response to the desired printer by typing

>SEND printer_id
and pressing the Enter key.

where
printer_id is the number of the printer where you want to print the data

Then, print the desired information by typing

>PATCHER;INFORM LIST;LEAVE
and pressing the Enter key.

Finally, redirect the display back to the terminal by typing

>SEND PREVIOUS
and pressing the Enter key.

How commands reference documentation is organized

This reference manual is one of two commands reference manuals for all commands used at a MAP in a Nortel Networks DMS-100 switch. The two commands reference manuals are the following*:*

Number	Title
297-1001-820	*DMS-100 Nonmenu Commands Historical Reference Manual* describes all nonmenu commands used at a MAP in a Nortel Networks DMS-100 switch.
297-1001-821	*DMS-100 Menu Commands Historical Reference Manual* describes all menu commands used at a MAP in a Nortel Networks DMS-100 switch.

What are menu and nonmenu commands

For the commands reference documents the commands used at a MAP terminal have been divided into two categories, menu and nonmenu:

- Menu commands are associated with a MAP display containing a numbered list or menu of commands and parameters when the level or sublevel from which the commands are entered has be accessed. Commands that can be executed from an accessed menu, but are not displayed, are called hidden commands. The level from which a menu command is entered is referred to as its menu or menu level.

Note 1: Menus may not always appear when a menu level or sublevel has been accessed, such as when displays have been suppressed with the command mapci nodisp.

mapci nodisp.↵

Note 2: Hidden commands may be seen when the menu level has been accessed by entering the listst command and printing the top directory.

listst↵

print *dir*.↵

- Nonmenu commands are not associated with a MAP display, even when the level or sublevel from which they may be entered has been accessed. The level from which a nonmenu command is entered is referred to as its directory or directory level.

Note: Nonmenu commands can be seen when the directory level has been accessed by entering the print command with the name of the directory.

print *dir*.↵

How this manual is organized

The organization of this manual is designed to provide rapid access to comprehensive commands information, in an easy-to-use and easy-to-understand format. The manual has a modular structure designed around chapters, which group commands according to the directory from which they are accessed. Special tables are provided to allow instant location of any command.

How volumes are organized

The reference manual is divided into into 4 volumes. Each volume contains a publication history section, an about this document section, and the first chapter containing the reference tables. The front cover and title page of each volume indicates the range of command levels within that volume. Since directories are in alphabetical order, the volume containing the directory one wishes to reference is easily determined. Within volumes, page numbers begin with same letter of the alphabet as the directory.

How the command reference tables chapter is organized

The first chapter, "Commands reference tables," includes two tables which :

- directory description table-contains a list of all directories in alphabetical order and provides a brief description of each

- directory cross-reference table-lists all of the documented commands in alphabetical order and cross references them to the directory to which they pertain and the page where they are documented

How the directory chapters are organized

Each chapter following the "Commands reference tables" documents one directory and all its commands. The names of the chapters are the same as the names of the directories which they document. The chapters are organized in alphabetical order.

Chapter organization

Each directory chapter consists of an overview section, which introduces the directory level, followed by a separate section for each command.

How the overview section is organized

The overview section of each chapter contains the following, in the order listed:

- a brief description of the directory
- instructions for accessing the directory level
- a directory commands table listing all the commands available from the directory cross-referenced to the page where they are described
- a common responses table, included only when all or most of the commands at a level have many of the same responses
- other tables of common information, included only when all or most of the commands at a level share the same information, such as alarms or status displays

How command sections are organized

Each command section consists of the following elements in the order listed:

- a brief description of the use and function of the command
- a commands expansion table
- a qualifications section describing any special characteristics, exceptions, restrictions, limitations, cautions, or warnings
- an examples table
- a responses table

Commands convention

The following is the description of the commands convention used in this manual.

How commands are represented

The command convention is used for two distinct representations of commands. One representation includes all parameters, variables, and syntactic relationships and is called a command expansion. The other representation is of commands as they are actually entered and is called a command example.

How the convention is used in command expansions

A special command table is used for a command expansion. It consists of two sections. The first section is the command expansion itself in which the following characteristics are represented.

- all parameters
- all variables
- hierarchy (the order in which elements must be entered)
- syntax (specific requirements of command strings)
- truncated and abbreviated forms when allowed
- defaults

The second section is a description of all the parameters and variables.

Command elements are represented exactly as they are to be entered in actual commands, except when italic font is used indicating the element is not entered as represented, such as for variable names and certain defaults.

> *Note:* Italics always indicates an element that is not entered as part of a command in the form in which it is shown. It is either a variable that must be replaced with a value, a range or another element; or, it is a default condition which is not entered as part of a command.

How command words are presented

The actual command word is represented in lowercase, boldface, except where uppercase is required by case sensitivity. The command appears to the left of all other elements in the command expansion (parameters and variables).

```
bsy        ┌ link      ps_link ┐   noforce
b          │ pm                │   force    ┌ wait   ┐
           └ unit      unit_no ┘            └ nowait ┘
```

If a truncated or abbreviated form of a command is allowed, it will appear directly beneath the long form of the command.

```
bsy        ┌ link      ps_link ┐   noforce
b          │ pm                │   force    ┌ wait   ┐
           └ unit      unit_no ┘            └ nowait ┘
```

> *Note:* The b command is not a true truncated form of the bsy command and is used merely for illustration.

How parameters are presented

Parameters are lowercase, regular type (not boldface), except where uppercase is required by command case sensitivity.

```
bsy        ⎡ link        ps_link  ⎤   noforce         ⎡ wait   ⎤
b          ⎢ pm                   ⎥   force            ⎢ nowait ⎥
           ⎣ unit        unit_no  ⎦                   ⎣        ⎦
```
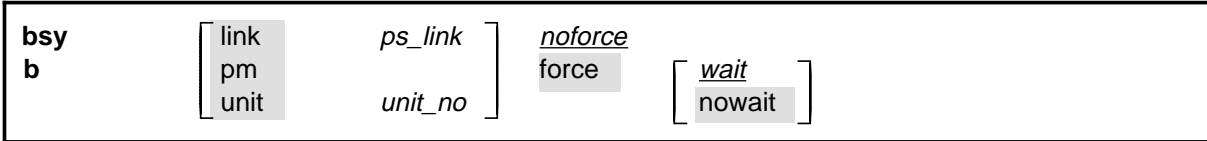
How variables are presented

Variable names are in italics. Italics indicates that the variable is not entered as shown, but must be replaced with some other element, such as a value, range, number, or item from a list.

The numbers, values, ranges, and lists that represent the substitutions or actual entries for variable names are not represented in the expansion of the command. These are described in detail for each variable in the description section below the expansion.

```
bsy        ⎡ link        ps_link  ⎤   noforce         ⎡ wait   ⎤
b          ⎢ pm                   ⎥   force            ⎢ nowait ⎥
           ⎣ unit        unit_no  ⎦                   ⎣        ⎦
```
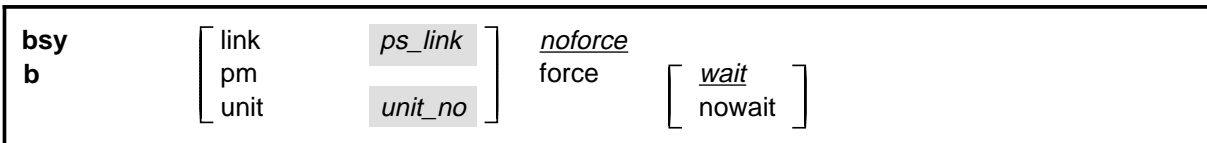
How hierarchy is presented

The order in which elements must be entered is represented by their order of appearance from left to right.

```
              1 ───▶ 2 ───▶ 3 ───▶ 4 ───▶ 5 ───▶ 6
bsy        ⎡ link        ps_link  ⎤   noforce         ⎡ wait   ⎤
b          ⎢ pm                   ⎥   force            ⎢ nowait ⎥
           ⎣ unit        unit_no  ⎦                   ⎣        ⎦
```
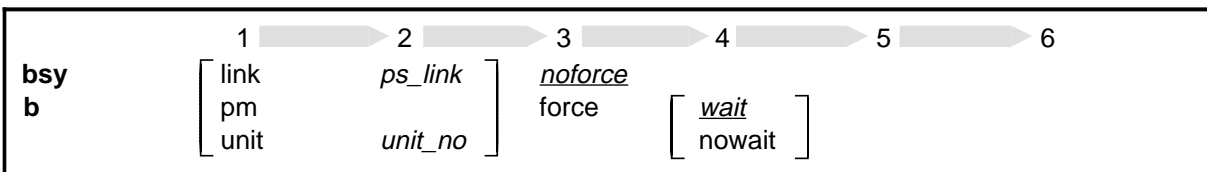
When several elements appear in the same horizontal position (that is, in a vertical list), one of them must be selected for that position, except when there is a default.

```
bsy            ⎡ link        ps_link  ⎤   noforce         ⎡ wait   ⎤
b              ⎢ pm                   ⎥   force            ⎢ nowait ⎥
    select one ⎣ unit        unit_no  ⎦                   ⎣        ⎦
```
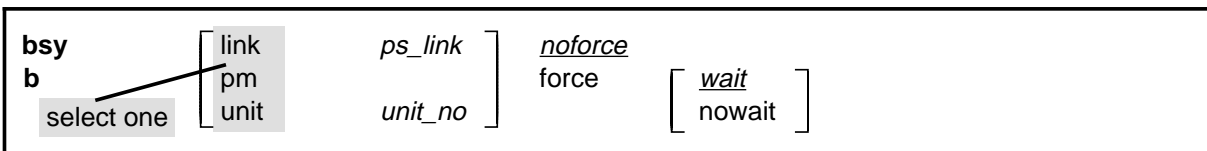
How long command expansions are presented

Some commands that have many parameters and variables with very long hierarchies require the expansion row to be continued. When this occurs, the horizontal lines of parameters and variables are numbered so that they

can be easily followed from one row to the next. Only numbered lines that are required to make syntax clear are in subsequent expansion rows (like row 2 in the third expansion continuation of the example).

command	parameter	variable	parameter	variable	parameter	variable	(1)
		parameter	parameter	variable	parameter	variable	parameter (2)
command (continued)	(1)	parameter	variable	parameter	variable		(1)
	(2)	variable	parameter	variable	parameter		(2)
command (continued)	(2)	parameter	variable	parameter		(end)	

How defaults are indicated

A default parameter is underlined. In a vertical list, if an element is entered, but not required, the system must act as if some element were entered. The action the system takes when an element is not entered is called a default action and is usually an action indicated by one of the elements that can be selected. Occasionally, the default action is something other than a selectable action. These nonselectable defaults are represented by the word, "default," or another word which indicates the action, and is in italics, to indicate that it cannot be entered. The default is fully described in the parameters and variables description section.

bsy	link	*ps_link*	*noforce*		
b	pm		force	*wait*	
	unit	*unit_no*		nowait	

How relationships between groups of elements are indicated

As a general rule of relationship, whenever an element is directly followed horizontally by another element; if the first element is selected, the second element is required.

bsy	link	*ps_link*	*noforce*		
b	pm		force	*wait*	
	unit	*unit_no*		nowait	

Within a command expansion, elements or groups of elements (parameters or variables) sometimes relate to elements that precede or follow them, but not all the elements that precede or follow them. To distinguish which elements relate to which, brackets surround those elements that, as a group, pertain to other elements. Only those elements that horizontally directly precede or follow the brackets are related to the elements within the

brackets. When elements are not in brackets, only individual elements that directly precede or follow others are related.

bsy b	$\begin{bmatrix} \text{link} & ps\_link \\ \text{pm} \\ \text{unit} & unit\_no \end{bmatrix}$	$\begin{matrix} \underline{noforce} \\ \text{force} \end{matrix}$ $\begin{bmatrix} \underline{wait} \\ \text{nowait} \end{bmatrix}$

How parameters and variables are described

The parameters and variables description contains a list of every parameter and variable that apply to the command, in alphabetical order. Each of these command elements is fully described, including replacement values and ranges for variables.

Following is an example of a command expansion table including the parameters and variables description.

bsy command parameters and variables

Command	Parameters and variables
bsy b	$\begin{bmatrix} \text{link} & ps\_link \\ \text{pm} \\ \text{unit} & unit\_no \end{bmatrix}$ $\begin{matrix} \underline{noforce} \\ \text{force} \end{matrix}$ $\begin{bmatrix} \underline{wait} \\ \text{nowait} \end{bmatrix}$

Parameters and variables	Description
force	This parameter overrides all other commands and states in effect on the specified units. If the whole peripheral module (PM) is to be taken out-of-service, confirmation (yes or no) is required.
link	This parameter busies one of the P-side links specified by *the ps_link* variable*.*
noforce	This default parameter indicates the condition when force parameter is not entered. Busy will not be forced.
nowait	This parameter enables the MAP to be used for other command entries before the bsy force command action is confirmed. The nowait parameter is used only with the force parameter.
pm	This parameter causes both units of the PM to be made busy.
ps_link	This variable specifies which of the P-side links is to be busied. The range is 0-3.
unit	This parameter causes the PM unit specified by the *unit_no* variable to be made busy*.*
-continued-	

bsy command parameters and variables (continued)	
Parameters and variables	**Description**
unit_no	This variable specifies which unit of the PM is to be busied. The range is 0-1.
wait	This default parameter indicates the default condition when no parameter is entered. The user must wait until the bsy force command action is confirmed before additional commands can be entered at the MAP.
End	

How the convention is used in command examples

Command examples use the same convention as a command expansion, except that all command elements are boldface. Commands can be entered exactly as they appear in examples except when an example does not use an actual variable entry, but a variable name shown in italics.

The following may be entered as shown.

bsy link 2↵

The variable *ps_link* must be replaced by an actual value before it can be entered.

bsy link *ps_link***↵

How other command conventions relate to the reference convention

The command convention used in this reference document is different from conventions used in some older Nortel Networks documents and from command information at a MAP terminal. This difference is intentional. The convention in this document is used to simplify explanations of command syntax and to eliminate possible confusion. For example, when the command information provided in a MAP help screen is unclear, reference to that command represented in a different convention, such as in this reference manual, should eliminate the ambiguity, whereas the same or a similar convention would merely repeat the confusion.

How to compare conventions

To illustrate the benefits of the convention used in this book, a comparison of the convention used in this document with the most common convention used in MAP help screens is provided in Table 1.

Table 1xxx
Command conventions comparison

Element	Commands reference manual	MAP screen
Commands	lowercase or case sensitive specific: **bsy**	uppercase: `BSY`
Truncated commands or abbreviations.	shown directly below long form: **bsy** **b**	Abbreviated form all uppercase, rest of command lowercase: `Bsy`
Parameters	lowercase or case sensitive specific: link	uppercase: `LINK`
Variables	italic, lowercase: *ps_link*	in angled brackets: `<ps_link>` ***note:*** angle brackets also indicate the the variable is mandatory.

Hierarchy	horizontal order, left to right:
	l pdtc *pm_numbers circuit*
		top to bottom:
		`{L <PDTC> {PDTC}`
		`    <PM_NUMBERS> {0 TO 255}`
		`    [<CIRCUIT> {0 to 16}]`

Defaults	underlined: <u>*wait*</u> nowait	no specific method established, but "optional" elements (meaning they do not have to be entered, implying defaults), are represented by square brackets: `[<CIRCUIT> {0 to 16}]`
Selectable elements	a vertical list: link pm unit	curly braces, separated by vertical bars: `{link	pm	unit}` or vertical list, separated by commas: `{link,` ` pm,` ` unit}`
Variable replacement values	defined under parameters and variables description	curly braces: `{0 to 16}`

What precautionary messages mean

Danger, warning, and caution messages in this document indicate potential risks. These messages and their meanings are listed in the following chart.

Message	Significance
DANGER	Possibility of personal injury
WARNING	Possibility of equipment damage
CAUTION	Possibility of service interruption or degradation

Examples of the precautionary messages follow.

DANGER
Risk of electrocution
The inverter contains high voltage lines. Do not open the front panel of the inverter unless fuses F1, F2, and F3 have been removed first. Until these fuses are removed, the high voltage lines inside the inverter are active, and you risk being electrocuted.

WARNING
Damage to backplane connector pins
Use light thumb pressure to align the card with the connectors. Next, use the levers to seat the card into the connectors. Failure to align the card first may result in bending of the backplane connector pins.

CAUTION
Loss of service
Subscriber service will be lost if you accidentally remove a card from the active unit of the peripheral module (PM). Before continuing, confirm that you are removing the card from the inactive unit of the PM.

Commands reference tables

To assist the user in locating a description, two commands reference tables are provided in this chapter, the directory description table and the directory cross reference table.

Directory descriptions

The directory description table provides a brief description of every directory documented in this manual.

Directory description table	
Directory	**Description**
ABBT	The ABBT directory accesses commands that are used to set up and run an automatic board-to-board test (ABBT).
ACDMR	The ACDMR directory works with the Meridian SL-100 Integrated Services Network to provide equal distribution of incoming calls to a predesignated group of telephone sets.
ACDPOOLS	The ACDPOOLS directory displays pool configurations and current status of Automatic Call Distribution (ACD) pools. These ACD commands partition ACD groups into data streams. This allows the down stream processor (DSP) to access data and receive call event messages for only the ACD groups within the selected data stream.
ACDRTDIS	The ACDRTDIS directory produces a simple management report for ACD groups. Statistics for the specified ACD groups are gathered and displayed at selected time intervals.
ACDSHOW	The ACDSHOW directory displays information about the current configuration of Automatic Call Distribution (ACD) groups and subgroups.
AFTCI	The AFTCI directory controls and monitors the automatic file transfer (AFT) system.
-continued-	

Directory description table (continued)	
Directory	**Description**
AMADUMP	The AMADUMP directory displays or prints the contents of Automatic Message Accounting (AMA) files produced in local or centralized AMA offices using the following formats: (1) block-by-block hexadecimal dump of the contents of a file for a specified range of blocks, (2) record-by-record dump of AMA call entries, data entries, or header entries within an AMA file (with or without screening specified), and (3) statistical profile charts of call entries by call record type and call duration
AMREPCI	The AMREPCI directory queries and changes the central processing unit (CPU) occupancy threshold. In addition, the AMREPCI directory amreped command produces the maintenance manager's morning report (A.M. report).
AUTOPATCH	The AUTOPATCH directory controls automatic application of patches.
AUTOTABAUDIT	The AUTOTABAUDIT directory checks table data integrity without external guidance. The AUTOTABAUDIT directory is accessed from the TABAUDIT directory, not the CI level.
BCSMON	The BCSMON directory dumps batch change supplement monitoring data.
BCSUPDATE	The BCSUPDATE directory accesses batch change supplement process driver commands.
C7MON	The C7MON (Common Channel Signaling No. 7 monitor) directory traces CCS7 messages passing through a Message Switch Buffer No. 7 (MSB7) or Link Interface Unit No. 7 (LIU7). When you enter search criteria, a template is created and stored in a match table. The system searches the message table to locate messages that match the template. If a match is found, a message dump is directed to either the MAP, logs, or to a specified disk file.
C7TU	The C7TU directory accesses commands that monitor CCS7 messages or links on both MSB7 and LIU7. The C7TU directory commands can be used on the Service Switching Point (SSP), Signal Transfer Point (STP), and Service Control Point (SCP) of the Digital Multiplex System (DMS) product line.
C7TUDTC	The C7TUDTC (CCS7 test utility digital trunk controller) directory accesses the digital trunk controller (DTC) test environment.
-continued-	

Directory description table (continued)	
Directory	**Description**
C7TULINK	The C7TULINK directory accesses commands for monitoring CCS7 messages. Links can be monitored as well. There are two versions of the C7TULINK environment. The basic C7TULINK environment (C7TULINK_PMT7) allows you to access commands that monitor messages only; building, sending, or intercepting messages is not allowed unless you provided a valid password when accessing the C7TU MAP level. The password-protected C7TULINK environment (C7TULINK_ILPT7) allows you to access the same basic commands as well as commands used for building, sending, or intercepting messages.
C7TURFC	The C7TURFC (CCS7 test utility traffic simulation test environment) directory accesses the traffic command environment.
CLOG	The CLOG directory accesses the switch-based Incoming Callers List which provides the subscriber with information pertaining to a limit of thirty-one of their incoming calls.
CPSTATUS	The CPSTATUS directory accesses the CPSTATUS tool to measure all CPU occupancies including call processing occupancy, to measure additional CPU time available for call processing work, and to indicate overload and switch performance with respect to the switch's engineering.
CUTOVER	The CUTOVER directory controls the cut-over mode for DTC, carriers, and CICs that have been swung over from the old switch to the DMS.
DASIM	The DASIM directory sets up parameters to control the simulator and monitor the messages between traffic operator position systems call processing and the simulator.
DBUT	The DBUT directory backs up and restores databases.
DCTTOOL	The DCTTOOL directory access the data call tester (DCT) tool commands.
DISKADM	The DISKADM directory initializes, configures, and administers the image files of several processors of the enhanced core switch called the system load module (SLM).
DISKUT	The DISKUT directory performs regular operations on the system load module (SLM), the volumes and files on the SLM disk, and the associated tape cartridge. In addition, the DISKUT directory stores image files on processors such as the message switch (MS) or the computing module (CM).
-continued-	

Directory description table (continued)	
Directory	**Description**
DRAM	The DRAM directory informs the system of the pre-recorded phrases in programmable read-only memory (PROM) and records phrases in random access memory (RAM) and erasable read-only memory (EEPROM).
DSINWT	The DSINWT directory controls the direct signaling inward wide-area telephone service (INWATS) increment.
DSKALLOC	The DSKALLOC directory allocates the storage space on the disk before a disk drive unit (DDU) is put in service.
DSKUT	The DSKUT directory displays or modifies information on files and volumes on input/output controller (IOC) disks.
DSMCCS	The DSMCCS directory displays management controls.
DSMTP	The DSMTP directory performs tests on the routing of direct signaling (DS) messages.
EDIT	The EDIT directory modifies store files.
EICERT	The EICERT directory enters the enhanced network integrity certification environment.
EICTS	The EICTS directory supports the enhanced network (ENET) version of the integrity check traffic simulator (ICTS).
ENETFAB	The ENETFAB directory (enhanced network fabric environment) manually controls ENETFAB testing for the SuperNode.
ENRETRO	The ENRETRO directory supports installation of an ENET in an existing DMS SuperNode office.
ESATOOLS	The ESATOOLS directory provides Emergency Stand-Alone (ESA) trunking information. ESA information includes data regarding the presence or lack of trunking capability during ESA, trunk data for a specific remote cluster controller (RCC) during ESA translations, and routing data used for a particular call during ESA.
FM	The FM directory accesses force management system (FM) commands for query management system (QMS) operators.
FOOTPRT	The FOOTPRT directory queries the information captured when a restart occurs. The fpbuf command can display all the events in the event buffer and the snapshot associated with each restart. The FOOTPRT directory commands can also reset the footprint event buffer on the active central control (CC) or central processing unit (CPU) or set the buffer to overwrite old events with new ones if it becomes full.
-continued-	

Directory description table (continued)	
Directory	**Description**
ICTS	The ICTS directory identifies available user-specified links to set up integrity check traffic simulator (ICTS) connections.
LDRCI	The LDRCI directory accesses the logical dump/restore increment.
LMCUT	The LMCUT directory (Line Maintenance Cutover facility) is used by the ABBT commissioning feature to transfer or cutover in-service lines from an existing switch to a DMS switch. This feature also provides message recording of all command executions in a progress file.
LNKUTIL	The LNKUTIL directory accesses commands that allow basic maintenance and manipulation of the datalinks used to transfer ACD statistics to a downstream processor.
LOADMGMT	The LOADMGMT directory tailors the ACD data configuration to prevent a loss of calls or alleviate the work load of a specific ACD group. The LOADMGMT directory enables senior ACD personnel to adjust the data configuration quickly.
LOGUTIL	The LOGUTIL directory manipulates the way logs are produced.
MAKERES	The MAKERES directory converts plain ordinary telephone systems (POTS) lines to Residential Enhanced Services (RES) lines over a specified range of line equipment numbers (LENs). The LENs to be converted are stored in Table LENLINES. Upon successful conversion, the LENs are moved to Table IBNLINES.
MASSTC	The MASSTC directory modifies rating information without affecting call processing or consuming large quantities of real time. A duplicate set of rating tables are created, the desired changes are made to the duplicate tables, and the table are tested. When the changes are complete, MASSTC directory commands are used to exchange the original set of tables with the duplicate set. The tables that originally were active and in use are taken offline and made inactive. Simultaneously, the tables that were changed and tested offline are made active.
MTXTRACK	The MTXTRACK directory activates tracking for several mobile telephone sets at a time. The MTXTRACK directory provides commands to flag events, tag mobiles, save the results in a file, display the data on the MAP, measure a mobile's RSSI while in call for hand-off boundary verification, and display the latest available data regarding the location of a mobile at the home switch.
-continued-	

Directory description table (continued)

Directory	Description
NETFAB	The NETFAB directory (network fabric environment) manually controls NETFAB testing network for the NT-40.
NMP	The NMP directory uses the strategic Focused Trunk Maintenance feature for DMS-250 TRK logs.
OCCTS	The OCCTS directory accesses the Equal Access Traffic Separation Measurement System (TSMS) operational measurement (OM) data.
PATCHER	The PATCHER directory performs manual and source level patching. (The directory reached with the patcher command is PTCHDIR.) The patch file contains the administrative section, load files, and the actual code that is applied to the DMS software. The file can be a change or a feature.
PROG	The PROG directory contains the command program listing for the command interpreter (CI) level of the map. The PROG directory is a read-only (R/O) directory which resides permanently on your Symbol Table (ST). It contains the command program listing for the CI system. All new command programs added to the DMS switch appear in this directory.
PT	The PT directory coordinates centralized MAP capability (CMAP) PassThru sessions. This directory provides commands to establish and quit either a CMAP PassThru session or a window between PassThru sessions.
PTCH	(See PATCHER directory description.)
QCALL	The QCALL directory details the refinement and call queue assignment of one particular call having a unique set of characteristics.
QVIEW	The QVIEW directory details the refinement and call queue assignment of a whole set of calls with all of their possible characteristics.
RASL	The robust application and session layer (RASL) directory manipulates network connections. The RASL parameters are set up in Table RASLAPPL and the office parameter RASL_PROTOCOL must be set in order for these commands to be available. The RASL directory provides commands that terminate a network connection, re-enable a network connection, disable a network connection for datafill changes, and summarize operational network connections.

-continued-

Directory description table (continued)	
Directory	**Description**
REG	The REG directory reads and resets the registers associated with lines and facilities including message rate (1MR), INWATS (INW), INW virtual facility groups (VFG), overflow hunt group (OFS), and two-way wide area telephone service (2WW).
SCPCDB	The SCPCDB directory creates a master database (the update processing instance database) during the installation of an SCP service.
SCPDBREQ	The SCPDBREQ directory is used by system designers to establish a working environment to update and retrieve a local master database. The commands in this directory are available in the lab environment only.
SCPEDDCI	The SCPEDDCI directory performs an external database dump for an SCP device. Records are retrieved from the update processor (UP) online local master database and written to the output device that you specify.
SCPEHPET	The SCPEHPET directory is used by system designers to enter valid and invalid updates for testing the Service Control Point II (SCPII) 800 Plus Enhanced (800+E) database. The commands in this directory are available in the lab environment only.
SERVORD	The SERVORD directory accesses Service Order system (SERVORD) commands. Some commands may not appear in all software loads due to absent feature packages or office parameter settings. The SERVORD commands are categorized the function for which they are used: adding, changing, removing, echoing, establishing lines and services, and suspending and restoring. In addition, six miscellaneous commands are provided. *Note*: The system identifies the SERVORD system as the SO directory. All references in the documentation to the SO directory pertain to the SERVORD system.
SHADOWUT	The SHADOWUT directory is used to administer shadowsets on the file processor (FP). Shadowing is the ability to group a set of physical disks into one logical disk that maintains multiple copies of the data.
SIGMON	The SIGMON directory performs signalling monitoring for up to four multifrequency compelled (MFC) trunks.
SIGRTU	The SIGRTU directory performs signalling route utilization (SIGRTU) functions.
-continued-	

Directory description table (continued)

Directory	Description
SLU	The SLU directory performs tasks related to the subscriber line usage (SLU) input tables.
SMDILNK	The SMDILNK directory queries the status of the Simplified Message Desk Interface (SMDI) application I/O and related datalinks.
SMDRLNK	The Station Message Detail Recording (SMDR) link directory queries routing information for SMDR call records, routes SMDR call records to a datalink pool, and deletes routing information for SMDR call records to a specified datalink pool.
SNIPINGCI	The SNIPINGCI directory sends a Supernode internet control message protocol (ICMP) echo packet to an internet protocol (IP) address. The destination host address, number of echo packets, size of packets, delay time between multiple packets, and data display control can be controlled using this directory. If the data display control is active, a report on the sequence number and round-trip time displays as each echo packet is received. When a series of pings completes, the packet loss percentage and the minimum, average, and maximum data displays.
SPMS	The SPMS directory displays results generated by the Switch Performance Monitoring System (SPMS). The SPMS directory commands are used to select the branches of the indexing hierarchy for which index results are to be reported, the extent to which each branch is to be reported, the number of characters per output line, and the ASCII as opposed to EBCDIC formfeed characters. (The SPMS operates automatically when SPMS Customer Option Feature Package NTX738AA is present in the switch.)
SRAMCI	The SRAMCI directory reconfigures the program contents of high-speed static RAM (SRAM) without requiring a system restart. The purpose of this function is to provide capacity gain.
SSAC	The SSAC directory generates station-specific authorization codes (SSACs) and to initiate automatic datafill of the appropriate tables for a specified range of directory numbers (DNs) within a designated customer group. In addition, the view command displays SSAC assignments.
SWACTCI	The SWACTCI directory performs warm switch activity (SWACT) functions.

-continued-

Directory description table (continued)	
Directory	**Description**
SYS	The SYS directory accesses all the CI system commands related to system operation and common to all DMS switch types. The system directory is a R/O directory which resides permanently in the ST. The contents of this directory can be viewed using the print sysdir command string.
TAB	The TAB directory performs table editor (TE) functions for any tuple in a table.
TABAUDIT	The TABAUDIT directory checks table data integrity without external guidance. Reports are produced for generic table checks, syntax checks, and table-specific data checks.
TFAN	The TFAN directory evaluates and processes traffic separation data.
VIP	The VIP directory enables and disables VIP service for local exchange codes (LECs) or queries the current status of VIP service.
XBERT	The XBERT directory detects bit errors in the transmission of high speed data in the external peripheral module (XPM) and line concentrating module/Integrated Services Line Module (LCM/ISLM) circuit packs. The XPM bit error rate test (XBERT) diagnostic supports six separate tests which test different hardware components in the peripheral speech and data paths. Several XPM peripheral side (P-side) ports or LCM bus interface cards (BIC) can be tested sequentially. XBERT is designed to be a fault detection and isolation tool. The XBERT command can be used by only one user at a time.
XPMLFP	The XPMLFP directory accesses the XPM loadfile utility. This level is used to start, stop, list, and obtain information about the status of loadfile patchs.
End	

Directory cross-reference

The directory cross reference table provides a complete alphabetical list of every command and indicates its associated directory and the number of the page in this manual where the description of that command is located.

Command/directory cross reference table		
Command	**Directory**	**Page**
8chol	SCPEHPET	S-69
8cnpa	SCPEHPET	S-71
8num	SCPEHPET	S-73
8nxx	SCPEHPET	S-75
8ocr	SCPEHPET	S-77
8odr	SCPEHPET	S-79
8pots	SCPEHPET	S-81
8serv	SCPEHPET	S-83
8servdel	SCPEHPET	S-85
8servsort	SCPEHPET	S-87
8shol	SCPEHPET	S-89
8ssp	SCPEHPET	S-91
8stat	SCPEHPET	S-93
8time	SCPEHPET	S-95
8toddow	SCPEHPET	S-97
abbt	PROG	P-97
abnn	SERVORD	S-135
abort	TAB	T-5
abort	XPMLFP	X-37
abortswact	SWACTCI	S-529
accsver	PROG	P-99
acddns	ACDSHOW	A-127
acdgrps	ACDPOOL	A-79
acdmr	PROG	P-103
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
acdpools	PROG	P-105
acdrtdis	PROG	P-107
acdshow	PROG	P-109
activate	MASSTC	M-29
ada	SERVORD	S-139
add	DSKALLOC	D-333
add	LOADMGMT	L-141
add	SERVORD	S-145
add	SRAMCI	S-491
add	TAB	T-7
addclass	LOGUTIL	L-199
addmember	SHADOWUT	S-309
ado	SERVORD	S-149
addrep	LOGUTIL	L-201
admingroup	ACDSHOW	A-131
aftci	PROG	P-111
agtpos	ACDSHOW	A-137
alloc	TQMIST	T-153
almstat	NMP	N-23
alter	C7TULINK	C-89
amadump	PROG	P-113
amadumpb	PROG	P-117
amrepci	PROG	P-119
amreped	AMREPCI	A-309
ann	DASIM	D-3
annsdebug	DRAM	D-273
apply	PATCHER	P-5
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
assess	BCSMON	B-3
assign	DRAM	D-275
assign	TAB	T-13
assigndump	DRAM	D-279
attach	SYS	S-571
audiogroup	ACDSHOW	A-145
auto	QCALL	Q-3
auto	TABAUDIT	T-91
autodump	PROG	P-121
autopatch	PROG	P-129
back	LOGUTIL	L-205
backup	DISKUT	D-203
backup	LOGUTIL	L-207
backupdb	DBUT	D-79
backuplog	DBUT	D-93
bcsmon	PROG	P-131
bcsupdate	PROG	P-133
bicrelay	PROG	P-135
bottom	TAB	T-15
broadcast	FM	F-3
buff	FOOTPRT	F-19
buffer	FM	F-5
build	C7TULINK	C-95
bulk	SERVORD	S-153
bundle	PATCHER	P-11
c7mon	PROG	P-141
c7tu	PROG	P-143

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
c7tudtc	C7TU	C-37
c7tulink	C7TU	C-39
c7tuprt	C7TU	C-41
c7turec	C7TU	C-45
c7turfc	C7TU	C-49
calldump	PROG	P-145
cancel	AUTOPATCH	A-325
cancel	C7TUTRFC	C-159
cancel	DBUT	D-105
car	QCALL	Q-5
ccannopt	DASIM	D-7
ccbiltype	DASIM	D-9
ccpoolid	DASIM	D-11
cdn	SERVORD	S-159
cdcsetup	PROG	P-149
change	EDIT	E-3
change	LOADMGMT	L-145
change	TAB	T-17
chdn	SERVORD	S-163
check	PATCHER	P-13
checkcm	MAKERES	M-3
checkrel	PROG	P-151
checktab	PROG	P-155
chf	SERVORD	S-167
chg	SERVORD	S-171
chl	SERVORD	S-181
cicp	SERVORD	S-187
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
ciprompt	SYS	S-575
ckln	SERVORD	S-191
clas	QCALL	Q-9
class	LOGUTIL	L-209
cld	QCALL	Q-13
clear	AUTOTABAUDIT	A-353
clear	DASIM	D-13
clear	LOGUTIL	L-213
clear	MTXTRACK	M-63
clear	TABAUDIT	T-93
clearboot	DSKUT	D-361
clearbootfl	DISKUT	D-211
clearst	SYS	S-579
clearvol	DISKUT	D-217
cln	SERVORD	S-195
clog	PROG	P-163
clr	TQMIST	T-155
clrbuf	NMP	N-25
clrinvreg	REG	R-19
clrroute	ACDSHOW	A-147
cltg	SERVORD	S-199
cnamdcag	PROG	P-165
co	QCALL	Q-17
command	SYS	S-581
compress	PROG	P-167
connect	DRAM	D-281
context	LOGUTIL	L-215

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
continue	ABBT	A-15
convert	MAKERES	M-5
copy	MAKERES	M-9
copy	PROG	P-171
copyaft	AFTCI	A-235
copyfile	SYS	S-585
count	TAB	T-21
counts	ACDSHOW	A-149
cpstat	PROG	P-175
cpstatus	PROG	P-177
create	MTXTRACK	M-65
createvol	DISKADM	D-167
ct4q	QCALL	Q-21
ctype	PROG	P-179
cutmode	LMCUT	L-13
cutoff	LMCUT	L-17
cutover	LMCUT	L-23
cutover	PROG	P-181
cutreport	LMCUT	L-29
dasim	PROG	P-183
data	DASIM	D-15
datadump	BCSUPDATE	B-55
date	SYS	S-589
dblocks	BCSMON	B-7
dbnn	SERVORD	S-203
dbstatus	DBUT	D-109
dbut	PROG	P-185
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
dcttool	PROG	P-187
dea	SERVORD	S-207
debug	DRAM	D-285
define	ABBT	A-17
defineset	SHADOWUT	S-311
del	SIGRTU	S-367
del	SERVORD	S-211
delaft	AFTCI	A-241
delay	AUTOPATCH	A-327
delcf	SERVORD	S-215
delclass	LOGUTIL	L-219
deldevice	LOGUTIL	L-221
delete	C7MON	C-3
delete	DCTTOOL	D-133
delete	DSKALLOC	D-335
delete	EDIT	E-7
delete	LOADMGMT	L-175
delete	TAB	T-25
deletefl	DISKUT	D-221
deletevol	DISKADM	D-175
delmember	SHADOWUT	S-313
delnode	SCPEHPET	S-99
delopt	MAKERES	M-15
delorigin	SCPEHPET	S-101
delrep	LOGUTIL	L-223
delset	SHADOWUT	S-315
demount	SYS	S-591

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
deo	SERVORD	S-219
deq	CLOG	C-187
describe	SPMS	S-467
detach	SYS	S-593
devcon	LNKUTIL	L-111
devdisc	LNKUTIL	L-115
device	BCSUPDATE	B-59
devstart	LNKUTIL	L-119
devstop	LNKUTIL	L-123
dgtables	PROG	P-189
diradd	DSKALLOC	D-337
dirdel	DSKALLOC	D-339
directory	SYS	S-595
dirpcopy	PROG	P-193
dirppfmt	PROG	P-197
disable	CUTOVER	C-221
disconnect	DRAM	D-289
disctrl	DSMCCS	D-389
disctrl	DSMTP	D-401
diskadm	PROG	P-201
diskut	PROG	P-205
dispall	NMP	N-27
dispbuf	NMP	N-31
display	C7MON	C-5
display	C7TULINK	C-103
display	DCTTOOL	D-141
display	DRAM	D-291
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
display	DSKALLOC	D-341
display	FOOTPRT	F-21
display	MTXTRACK	M-67
display	PATCHER	P-19
display	SIGMON	S-341
display	SPMS	S-469
display	SWACTCI	S-531
display	TAB	T-29
display	XBERT	X-5
displaydisk	DISKADM	D-179
displayset	SHADOWUT	S-317
displayvols	DISKADM	D-183
dlcheck	PATCHER	P-25
dmopro	PROG	P-207
dncutoff	LMCUT	L-39
dncutover	LMCUT	L-47
dnlpcdmo	PROG	P-211
dnnobtst	LMCUT	L-55
dnpicdmo	PROG	P-215
dnpiclist	PROG	P-219
down	EDIT	E-11
down	TAB	T-31
dpc	C7TU	C-51
dramrec	PROG	P-229
ds30test	ENRETRO	E-155
ds512test	ENRETRO	E-159
dsinwt	PROG	P-233

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
dskalloc	DSKALLOC	D-343
dskalloc	PROG	P-235
dskut	PROG	P-239
dsmccs	PROG	P-241
dsmtp	PROG	P-243
dsp	SERVORD	S-223
dump	AMADUMP	A-283
dump	C7TULINK	C-105
dump	DASIM	D-19
dump	FOOTPRT	F-25
dump	PROG	P-245
dump	SIGRTU	S-369
dump	TQMIST	T-157
dumpall	BCSMON	B-9
dumplogs	LOGUTIL	L-227
duplicate	DISKUT	D-225
duplicate	MASSTC	M-33
eadasfmt	PROG	P-249
eadaskey	PROG	P-255
echo	SERVORD	S-231
eddcancel	SCPEDDI	S-43
edddelete	SCPEDDI	S-45
edddump	SCPEDDI	S-49
eddresume	SCPEDDI	S-53
eddstatus	SCPEDDI	S-57
edit	EDIT	E-15
edit	PROG	P-259
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
eicert	EICTS	E-79
eicts	PROG	P-263
ejecttape	DISKUT	D-229
emulate	CUTOVER	C-223
enable	MASSTC	M-37
end	EDIT	E-19
endpof	TAB	T-33
enretro	PROG	P-265
enretroswct	ENRETRO	E-163
enretrover	ENRETRO	E-167
eqpcounts	BCSMON	B-11
erase	DRAM	D-293
erase	FM	F-7
erase	SYS	S-597
erasefl	DSKUT	D-363
erasesf	SYS	S-599
esatools	PROG	P-267
esatraver	ESATOOLS	E-199
esatrunk	ESATOOLS	E-203
esgoff	PROG	P-269
esp	PROG	P-271
est	SERVORD	S-235
event	MTXTRACK	M-69
event	TQMIST	T-161
eventlist	MTXTRACK	M-73
exception	SPMS	S-473
exclude	AUTOTABAUDIT	A-355

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
exclude	TABAUDIT	T-95
execute	AUTOTABAUDIT	A-357
execute	TABAUDIT	T-97
expand	PROG	P-275
explain	QCALL	Q-25
failcnt	NMP	N-35
failmessage	SYS	S-601
fiaudgrp	ACDSHOW	A-151
file	EDIT	E-21
file	MTXTRACK	M-75
filter	AMADUMP	A-291
find	DRAM	D-295
find	EDIT	E-23
find	LDRCI	L-3
first	LOGUTIL	L-231
first	TAB	T-35
flash	CUTOVER	C-225
fm	PROG	P-281
foaudgrp	ACDSHOW	A-155
footprt	PROG	P-283
forceout	SYS	S-603
forceswact	SWACTCI	S-533
format	LOGUTIL	L-233
format	TAB	T-37
formatdisk	DISKADM	D-185
forward	LOGUTIL	L-235
fpbuf	FOOTPRT	F-29
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
fromtable	QVIEW	Q-69
gen	SSAC	S-513
getmate	FOOTPRT	F-35
getpat	PROG	P-285
gfntest	PROG	P-289
groupinfo	ACDSHOW	A-159
groupname	ACDSHOW	A-169
grpnumon	PROG	P-291
grpsetup	PROG	P-293
gwxref	PROG	P-299
heading	TAB	T-41
help	ABBT	A-35
help	ACDMR	A-55
help	ACDPOOL	A-83
help	ACDRTDIS	A-103
help	ACDSHOW	A-173
help	AFTCI	A-247
help	AMADUMP	A-301
help	AMREPCI	A-313
help	AUTOPATCH	A-329
help	AUTOTABAUDIT	A-361
help	BCSMON	B-15
help	BCSUPDATE	B-61
help	C7TU	C-55
help	C7TUDTC	C-67
help	C7TULINK	C-109
help	C7TUTRFC	C-161

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
help	CLOG	C-191
help	CUTOVER	C-227
help	DASIM	D-21
help	DBUT	D-113
help	DCTTOOL	D-149
help	DISKADM	D-191
help	DISKUT	D-231
help	DRAM	D-297
help	DSINWT	D-319
help	DSKALLOC	D-347
help	DSKUT	D-367
help	DSMCCS	D-391
help	DSMTP	D-403
help	EICERT	E-55
help	EICTS	E-83
help	ENETFAB	E-135
help	ENRETRO	E-169
help	ESATOOLS	E-205
help	FM	F-9
help	FOOTPRT	F-41
help	ICTS	I-3
help	LDRCI	L-5
help	LMCUT	L-63
help	LNKUTIL	L-125
help	LOADMGMT	L-179
help	LOGUTIL	L-239
help	MAKERES	M-19
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
help	MASSTC	M-39
help	NETFAB	N-3
help	NMP	N-37
help	OCCTS	O-3
help	PROG	P-303
help	PT	P-891
help	PATCHER	P-29
help	QCALL	Q-27
help	QVIEW	Q-73
help	RASL	R-3
help	REG	R-21
help	SCPCBD	S-3
help	SCPDBREQ	S-15
help	SCPEDDI	S-59
help	SCPEHPET	S-103
help	SHADOWUT	S-321
help	SIGMON	S-345
help	SIGRTU	S-371
help	SLU_CIDIR	S-383
help	SMDILNK	S-423
help	SMDRLNK	S-435
help	SNPINGCI	S-449
help	SERVORD	S-241
help	SPMS	S-475
help	SRAMCI	S-493
help	SSAC	S-517
help	SWACTCI	S-535

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
help	TABAUDIT	T-101
help	TFAN	T-123
help	TQMIST	T-163
help	VIP	V-3
help	XBERT	X-7
highcpocc	BCSMON	B-17
highlogs	BCSMON	B-19
highparms	BCSMON	B-21
hlrquery	PROG	P-305
hx	SYS	S-607
ibnpiclist	PROG	P-313
icert	EICERT	E-57
iclear	EICTS	E-85
iclear	ICTS	I-5
iconfig	EICTS	E-87
iconfig	ICTS	I-9
icts	PROG	P-321
if	SYS	S-611
iinstruct	EICERT	E-65
include	AUTOTABAUDIT	A-365
include	TABAUDIT	T-105
info	AUTOTABAUDIT	A-367
info	TABAUDIT	T-107
info	TQMIST	T-165
inform	PATCHER	P-31
inform	TAB	T-43
inhibit	AUTOPATCH	A-331
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
init	ACDMR	A-57
initiate	XBERT	X-11
initupd	SCPEHPET	S-105
input	EDIT	E-25
inserttape	DISKUT	D-233
insinw	DSINWT	D-321
insmcc	DSMCCS	D-393
insmtp	DSMTP	D-405
insnode	SCPEHPET	S-107
intdn	DASIM	D-23
intercept	C7TUDTC	C-69
intercept	C7TULINK	C-113
ioption	EICTS	E-97
ioption	ICTS	I-19
iquery	EICTS	E-107
iquery	ICTS	I-29
irefresh	EICTS	E-115
irefresh	ICTS	I-39
isetup	EICTS	E-119
isetup	ICTS	I-43
italk	SERVORD	S-245
iterminate	EICERT	E-69
itrnsl	EICTS	E-125
itrnsl	ICTS	I-49
jffreeze	PROG	P-323
ktreport	PROG	P-327
lang	DASIM	D-25
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
lang	QCALL	Q-31
last	LOGUTIL	L-241
last	TAB	T-45
lastct4q	QCALL	Q-33
ldmate	PROG	P-339
ldrci	PROG	P-345
leave	DASIM	D-27
leave	ICTS	I-53
leave	MASSTC	M-43
leave	SYS	S-615
lindex	SYS	S-619
line	EDIT	E-29
linestr	EDIT	E-33
list	PROG	P-347
list	SYS	S-621
list	TAB	T-47
listab	PROG	P-349
listbootfl	DISKUT	D-237
listdevs	LOGUTIL	L-243
listfl	DISKUT	D-241
listing	DASIM	D-29
listlogs	LOGUTIL	L-245
listnodes	LOGUTIL	L-247
listreps	LOGUTIL	L-249
listroute	LOGUTIL	L-253
listst	SYS	S-627
listtime	LOGUTIL	L-257
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
listvips	VIP	V-5
listvol	DSKUT	D-369
listvols	DISKUT	D-245
lmcut	PROG	P-351
lnkstat	LNKUTIL	L-127
lnkutil	PROG	P-353
load	PROG	P-355
loadmgmt	ACDSHOW	A-177
locate	MTXTRACK	M-77
locate	TAB	T-53
logbuffer	BCSMON	B-23
logcheck	BCSUPDATE	B-63
logcount	BCSMON	B-27
logdtl	DASIM	D-35
logformat	PROG	P-359
login	SYS	S-629
loginid	ACDSHOW	A-179
logout	SYS	S-633
logtrace	LOGUTIL	L-259
logutil	PROG	P-367
loop	C7TUDTC	C-71
lpiclist	PROG	P-369
makeres	PROG	P-377
mapci	PROG	P-379
masstc	PROG	P-383
match	PATCHER	P-45
matchall	PATCHER	P-49
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
matelink	PROG	P-385
mdbcreate	SCPCBD	S-5
memattr	PROG	P-395
memory	BCSMON	B-29
modcheck	SWACTCI	S-537
mode	ACDSHOW	A-185
mode	LOGUTIL	L-261
modify	C7TUTRFC	C-163
mon	SIGRTU	S-373
monitor	C7MON	C-13
monitor	C7TUDTC	C-73
monitor	C7TULINK	C-129
mount	PROG	P-397
mount	SYS	S-637
movebcs	PROG	P-399
mrstat	ACDMR	A-59
msg	SYS	S-641
msgcode	C7TU	C-57
mtcchk	PROG	P-403
mtxalm	PROG	P-405
mtxtrack	PROG	P-409
ncsci	PROG	P-411
netfab	ICTS	I-55
new	SERVORD	S-247
newacd	SERVORD	S-251
newdn	SERVORD	S-257
newpatch	BCSMON	B-31
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
next	TAB	T-55
nmp	PROG	P-415
nmreloc	ENRETRO	E-171
nmtest	ENRETRO	E-173
nobtst	LMCUT	L-65
nodeset	PATCHER	P-51
norestartswact	SWACTCI	S-545
nsaudgrp	ACDSHOW	A-187
nsroute	ACDSHOW	A-189
occquerycarr	OCCTS	O-5
occqueryclli	OCCTS	O-7
occqueryint	OCCTS	O-11
occqueryreg	OCCTS	O-15
occqueryts	OCCTS	O-17
occts	PROG	P-417
occtsrepreg	OCCTS	O-19
occtsreptsno	OCCTS	O-23
omdump	PROG	P-419
ommaster	PROG	P-423
oms	BCSMON	B-33
omshow	PROG	P-429
open	LOGUTIL	L-263
opensecret	LOGUTIL	L-265
opr	BCSMON	B-35
oprtco	LMCUT	L-73
oprthold	LMCUT	L-81
order	QCALL	Q-35
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
order	QVIEW	Q-77
origclg	QCALL	Q-37
origtrnk	QCALL	Q-41
out	SERVORD	S-263
outdn	SERVORD	S-267
override	BCSUPDATE	B-65
override	TAB	T-57
ovflroute	ACDSHOW	A-191
owner	SYS	S-643
package	PROG	P-437
parmcalc	PROG	P-441
password	ACDSHOW	A-193
password	FM	F-11
patchedit	PROG	P-445
patcher	PROG	P-449
patchlist	XPMLFP	X-39
perm	MASSTC	M-45
permit	SYS	S-645
pfxt	QCALL	Q-43
phmerge	PROG	P-451
phmerge	SYS	S-653
piclist	PROG	P-453
ping	SNPINGCI	S-453
pingdef	SNPINGCI	S-459
playback	DRAM	D-299
plp	SERVORD	S-271
pmaudit	BCSUPDATE	B-67
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
pmconfig	BCSMON	B-39
pmloader	PROG	P-461
pmloads	BCSMON	B-43
pmmoveinv	ENRETRO	E-177
pmtrnsl	ENRETRO	E-181
pof	TAB	T-59
poolid	DASIM	D-37
pools	ACDPOOL	A-85
poolstart	LNKUTIL	L-129
poolstop	LNKUTIL	L-133
pops	PROG	P-467
portinfo	XBERT	X-21
position	DRAM	D-301
position	TAB	T-61
posrsn	DASIM	D-39
postswact	BCSUPDATE	B-69
precheck	BCSUPDATE	B-71
preswact	BCSUPDATE	B-75
prev	TAB	T-63
previous	XBERT	X-23
print	SYS	S-657
printmap	PROG	P-471
printtrack	MTXTRACK	M-79
privclas	PROG	P-473
profile	SYS	S-659
prompt	LOADMGMT	L-183
promptme	QCALL	Q-45

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
pt	PROG	P-477
pt	PT	P-893
pte	TAB	T-65
ptquit	PT	P-895
pttime	PT	P-899
putpof	TAB	T-67
pvnacg	PROG	P-479
q	ACDSHOW	A-197
q	C7MON	C-21
q	DASIM	D-41
q	MTXTRACK	M-91
q	PATCHER	P-55
q	SCPEDDI	S-61
qbb	PROG	P-481
qbclid	PROG	P-485
qbert	PROG	P-489
qbnv	PROG	P-497
qcall	PROG	P-511
qcm	PROG	P-513
qcopyaft	PROG	P-519
qcounts	PROG	P-521
qcpugno	PROG	P-527
qcust	PROG	P-529
qc7mon	C7MON	C-23
qdch	PROG	P-535
qdn	PROG	P-549
qdna	PROG	P-553
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
qdnsu	PROG	P-557
qdnwrk	PROG	P-561
qgrp	PROG	P-569
qha	PROG	P-581
qhasu	PROG	P-587
qhold	LMCUT	L-87
qhu	PROG	P-593
qit	PROG	P-599
qlen	PROG	P-607
qlenwrk	PROG	P-615
qload	PROG	P-621
qloop	PROG	P-627
qlt	PROG	P-629
qmadn	PROG	P-633
qncos	PROG	P-637
qphf	PROG	P-641
qphi	PROG	P-653
qprio	PROG	P-657
qscmp	PROG	P-661
qsconn	PROG	P-665
qscugno	PROG	P-669
qsl	PROG	P-671
qsrdb	PROG	P-679
qsrdbxfr	PROG	P-683
qtopspos	PROG	P-685
query	AUTOPATCH	A-335
query	CUTOVER	C-229

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
query	FOOTPRT	F-43
query	PROG	P-689
query ports	XBERT	X-25
queryaft	AFTCI	A-251
queryclli	TFAN	T-125
querycputhresh	AMREPCI	A-315
queryint	TFAN	T-129
querypld	PROG	P-711
queryrcc	ESATOOLS	E-207
queryrdt	PROG	P-713
queryreg	TFAN	T-133
queryts	TFAN	T-135
queryxfer	PROG	P-715
queue	CLOG	C-195
quit	C7TUTRFC	C-165
quit	ABBT	A-37
quit	ACDMR	A-63
quit	ACDPOOL	A-91
quit	ACDRTDIS	A-105
quit	ACDSHOW	A-199
quit	AFTCI	A-257
quit	AMADUMP	A-303
quit	AMREPCI	A-317
quit	AUTOPATCH	A-337
quit	AUTOTABAUDIT	A-371
quit	BCSMON	B-45
quit	BCSUPDATE	B-79
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
quit	C7MON	C-25
quit	C7TU	C-61
quit	C7TUDTC	C-75
quit	C7TULINK	C-141
quit	C7TUTRFC	C-167
quit	CLOG	C-203
quit	CPSTATUS	C-215
quit	CUTOVER	C-231
quit	DBUT	D-115
quit	DCTTOOL	D-151
quit	DISKADM	D-193
quit	DISKUT	D-249
quit	DRAM	D-305
quit	DSINWT	D-323
quit	DSKALLOC	D-349
quit	DSKUT	D-371
quit	DSMCCS	D-395
quit	DSMTP	D-407
quit	EDIT	E-35
quit	EICERT	E-71
quit	EICTS	E-129
quit	ENETFAB	E-139
quit	ENRETRO	E-183
quit	ESATOOLS	E-209
quit	FM	F-13
quit	FOOTPRT	F-45
quit	LDRCI	L-7
quit	LMCUT	L-93

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
quit	LNKUTIL	L-135
quit	LOADMGMT	L-185
quit	LOGUTIL	L-267
quit	MAKERES	M-23
quit	MASSTC	M-47
quit	MTXTRACK	M-93
quit	NETFAB	N-5
quit	NMP	N-39
quit	OCCTS	O-27
quit	PATCHER	P-57
quit	PT	P-901
quit	QCALL	Q-49
quit	QVIEW	Q-79
quit	RASL	R-5
quit	REG	R-23
quit	SCPCBD	S-9
quit	SCPDBREQ	S-17
quit	SCPEDDI	S-63
quit	SCPEHPET	S-109
quit	SHADOWUT	S-323
quit	SIGMON	S-347
quit	SIGRTU	S-377
quit	SLU_CIDIR	S-385
quit	SMDILNK	S-427
quit	SMDRLNK	S-437
quit	SNPINGCI	S-461
quit	SERVORD	S-275
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
quit	SPMS	S-477
quit	SRAMCI	S-495
quit	SSAC	S-519
quit	SWACTCI	S-547
quit	TAB	T-69
quit	TABAUDIT	T-111
quit	TFAN	T-139
quit	TQMIST	T-167
quit	VIP	V-7
quit	XBERT	X-27
quit	XPMLFP	X-41
quote	SYS	S-661
qvep	PROG	P-717
qview	PROG	P-721
qwucr	PROG	P-723
range	TAB	T-73
rasl	PROG	P-727
raslclose	RASL	R-9
raslstart	RASL	R-11
raslstop	RASL	R-13
rculen	PROG	P-729
read	REG	R-27
read	SYS	S-663
readpx	REG	R-31
readreset	REG	R-33
readresetpx	REG	R-37
readresetvfg	REG	R-41

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
readvfg	REG	R-43
reassign	LOADMGMT	L-189
reclaim	PATCHER	P-61
record	DRAM	D-309
reg	PROG	P-731
reinit	DSKALLOC	D-353
reinitvol	DISKADM	D-197
relocate	SRAMCI	S-499
remlogin	PROG	P-733
remlogout	PROG	P-739
remove	C7TUDTC	C-79
remove	C7TULINK	C-143
remove	PATCHER	P-65
remove	SRAMCI	S-501
renamefl	DISKUT	D-253
renamefl	DSKUT	D-375
renumber	LOGUTIL	L-271
repack	SRAMCI	S-503
repeat	SYS	S-665
replace	TAB	T-75
report	AUTOTABAUDIT	A-375
report	C7TUTRFC	C-171
report	FOOTPRT	F-49
report	TABAUDIT	T-115
reqdn	DASIM	D-43
reroute	LOGUTIL	L-273
res	SERVORD	S-279
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
reset	BCSMON	B-49
reset	BCSUPDATE	B-83
reset	C7TUTRFC	C-173
reset	CLOG	C-207
reset	FOOTPRT	F-53
reset	LOGUTIL	L-275
reset	SIGMON	S-351
reset	XBERT	X-31
resetovr	AFTCI	A-261
resetpft	AFTCI	A-265
resetroute	LOGUTIL	L-277
resgrp	SERVORD	S-283
rest	QCALL	Q-53
restab	PROG	P-741
restart	SYS	S-667
restartbase	SYS	S-669
restartinfo	BCSMON	B-51
restartswact	SWACTCI	S-551
restore	C7TUDTC	C-81
restore	C7TULINK	C-145
restore	DISKUT	D-259
restore	VIP	V-11
restoredb	DBUT	D-119
restoreexecs	SWACTCI	S-557
restrict	VIP	V-15
resume	ENETFAB	E-143
resume	LOGUTIL	L-279
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
resume	NETFAB	N-9
resumedev	LOGUTIL	L-281
resumepm	SWACTCI	S-559
retrieve	SCPEHPET	S-113
retroinit	ENRETRO	E-187
return	TAB	T-79
revive	PROG	P-743
rextest	PROG	P-751
rfmap	MTXTRACK	M-97
rfmtdisp	PROG	P-755
rfpdata	DASIM	D-45
rindex	SYS	S-671
rlsco	LMCUT	L-97
rlshold	LMCUT	L-103
rst	DASIM	D-49
rst	TQMIST	T-171
rtdstat	ACDRTDIS	A-109
runstep	BCSUPDATE	B-85
save	EDIT	E-39
save	MASSTC	M-51
savemap	PROG	P-757
scencci	DASIM	D-51
scenibm	DASIM	D-59
schedule	AUTOPATCH	A-341
scpcdb	PROG	P-759
scpclose	SCPDBREQ	S-21
scpdbreq	PROG	P-761
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
scpeddci	PROG	P-763
scpehpet	PROG	P-765
scpget	SCPDBREQ	S-23
scpopen	SCPDBREQ	S-25
scpput	SCPDBREQ	S-27
scpread	SCPDBREQ	S-29
scpreqid	SCPDBREQ	S-31
scpresp	SCPDBREQ	S-33
scpset	SCPDBREQ	S-35
scpsmrreq	SCPDBREQ	S-37
scpsmureq	SCPDBREQ	S-39
scrap	MASSTC	M-55
sdna	SERVORD	S-287
seiquery	PROG	P-767
sel	TQMIST	T-173
select	C7TULINK	C-147
select	SIGMON	S-353
send	ACDMR	A-67
send	ACDRTDIS	A-113
send	C7TULINK	C-151
send	SYS	S-673
sendsmdr	SMDRLNK	S-441
servnum	DASIM	D-65
servord	PROG	P-771
set	PATCHER	P-71
set	SPMS	S-481
setaft	AFTCI	A-269
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
setbanner	PROG	P-773
setboot	DSKUT	D-377
setbootfl	DISKUT	D-267
setdate	SYS	S-677
setencp	ENRETRO	E-189
setlink	DASIM	D-69
setnode	DBUT	D-129
setnode	SHADOWUT	S-327
setovr	AFTCI	A-273
setrcc	ESATOOLS	E-213
setrep	SPMS	S-485
settime	SYS	S-679
setup	C7TUTRFC	C-175
shadowut	PROG	P-777
shadowut	SHADOWUT	S-329
sherlock	PROG	P-779
show	ABBT	A-41
show	QCALL	Q-57
show	QVIEW	Q-83
show	SYS	S-681
show	TQMIST	T-177
showboot	DSKUT	D-379
showfl	DSKUT	D-383
shownode	SCPEHPET	S-115
showrasl	RASL	R-15
showrec	SCPEHPET	S-117
showret	SCPEHPET	S-119
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
showvol	DSKUT	D-385
sigmon	PROG	P-791
sigrtu	PROG	P-793
sim	DASIM	D-71
sitload	DRAM	D-313
sleep	SYS	S-683
slu	PROG	P-795
sluadd	SLU_CIDIR	S-389
slu_deinstall	SLU_CIDIR	S-393
sludel	SLU_CIDIR	S-395
sludump	SLU_CIDIR	S-399
slufindi	SLU_CIDIR	S-401
slufindo	SLU_CIDIR	S-405
slu_install	SLU_CIDIR	S-409
slu_lminstall	SLU_CIDIR	S-413
sluset	SLU_CIDIR	S-417
slu_table_status	SLU_CIDIR	S-419
smdidisp	PROG	P-797
smdistat	SMDILNK	S-431
smdilnk	PROG	P-801
smdrlnk	PROG	P-803
smdrstat	SMDRLNK	S-443
snpingci	PROG	P-805
sortnode	SCPEHPET	S-121
sortorigin	SCPEHPET	S-123
spms	PROG	P-807
sramci	PROG	P-809
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
srdbreq	PROG	P-811
srdbupd	PROG	P-819
ssac	PROG	P-823
start	ABBT	A-47
start	AUTOPATCH	A-345
start	C7MON	C-29
start	C7TUTRFC	C-177
start	ENETFAB	E-145
start	LOGUTIL	L-285
start	MTXTRACK	M-101
start	NETFAB	N-11
start	QCALL	Q-59
start	QVIEW	Q-85
start	SIGMON	S-357
start	XPMLFP	X-45
startaft	AFTCI	A-277
startdev	LOGUTIL	L-287
startmember	SHADOWUT	S-331
startshadow	SHADOWUT	S-333
status	AUTOTABAUDIT	A-379
status	ACDPOOL	A-95
status	ACDSHOW	A-203
status	BCSUPDATE	B-87
status	C7TUDTC	C-83
status	C7TULINK	C-155
status	C7TUTRFC	C-179
status	CLOG	C-209
-continued-		

Command/directory cross reference table (continued)

Command	Directory	Page
status	ENETFAB	E-147
status	ENRETRO	E-193
status	MASSTC	M-57
status	MTXTRACK	M-103
status	NETFAB	N-13
status	PATCHER	P-75
status	SIGMON	S-361
status	SRAMCI	S-507
status	SWACTCI	S-561
status	TABAUDIT	T-119
status	VIP	V-17
status	XPMLFP	X-47
statuscheck	SWACTCI	S-563
stop	ABBT	A-51
stop	ACDMR	A-73
stop	C7MON	C-33
stop	C7TUTRFC	C-181
stop	ENETFAB	E-149
stop	LOGUTIL	L-291
stop	MTXTRACK	M-105
stop	NETFAB	N-17
stop	SIGMON	S-363
stop	XBERT	X-33
stopaft	AFTCI	A-279
stopdev	LOGUTIL	L-293
stopdump	PROG	P-825
stopecho	SERVORD	S-293

-continued-

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
stopmember	SHADOWUT	S-335
stopshadow	SHADOWUT	S-337
stopsmdr	SMDRLNK	S-445
store	PROG	P-827
subpools	ACDPOOL	A-97
subtable	TAB	T-81
sum	PROG	P-845
summary	QVIEW	Q-89
supervisor	ACDSHOW	A-207
suppress	LOGUTIL	L-297
sus	SERVORD	S-295
susgrp	SERVORD	S-299
suspend	ENETFAB	E-151
suspend	NETFAB	N-19
swactci	BCSUPDATE	B-91
swap	SERVORD	S-303
swnode	PROG	P-849
tabaudit	PROG	P-853
tabentry	ACDSHOW	A-215
table	PROG	P-855
tape	SYS	S-685
tapeconfirm	SYS	S-693
tcmmon	PROG	P-857
terminate	AUTOTABAUDIT	A-383
testbook	DCTTOOL	D-155
testoff	CUTOVER	C-235
teston	CUTOVER	C-237
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
tfan	PROG	P-865
threshold	ACDSHOW	A-219
threshold	LOGUTIL	L-299
throute	ACDSHOW	A-223
time	QCALL	Q-61
time	SYS	S-695
timeframe	AUTOTABAUDIT	A-385
timereset	LOGUTIL	L-301
top	EDIT	E-41
top	TAB	T-83
topspw	PROG	P-867
totable	QVIEW	Q-91
tqmist	PROG	P-869
trace	DASIM	D-73
trace	TQMIST	T-179
traceco	QVIEW	Q-95
tracect4q	QVIEW	Q-99
track	MTXTRACK	M-107
translate	DSINWT	D-327
trnsl	FOOTPRT	F-55
tsndmp	PROG	P-871
tsrepreg	TFAN	T-143
tsreptsno	TFAN	T-147
tsttrnsl	DSMTP	D-411
type	EDIT	E-43
type	LOGUTIL	L-303
unlock	FOOTPRT	F-63
-continued-		

Command/directory cross reference table (continued)		
Command	**Directory**	**Page**
unpermit	SYS	S-697
unsel	TQMIST	T-181
unset	PATCHER	P-81
up	EDIT	E-47
up	TAB	T-85
update	DSKALLOC	D-355
use	QCALL	Q-65
use	QVIEW	Q-103
validaudio	ACDSHOW	A-225
validroutes	ACDSHOW	A-229
vendor	DASIM	D-75
verbose	C7TUTRFC	C-183
verify	EDIT	E-51
verify	TAB	T-87
view	SSAC	S-523
vip	PROG	P-875
wideband	PROG	P-877
xbert	PROG	P-881
xplist	PATCHER	P-85
xpmlfp	PROG	P-887
End		

ABBT level commands

Use the ABBT level of the MAP to access commands that are used to set up and run an automatic board-to-board test (ABBT).

Before using the ABBT directory, you must provide information about the old office, the number of ABBT test units, output files, devices, and directory numbers (DNs) to be tested. When all aspects of the test are defined and the test is started, test results are accumulated in a file and sent to an output device that you specify.

Accessing the ABBT level

To access the ABBT level, enter the following command from the CI level:

abbt ↵

ABBT commands

The commands available at the ABBT MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

ABBT commands	
Command	**Page**
continue	A-15
define	A-17
help	A-35
quit	A-37
show	A-41
start	A-47
stop	A-51

ABBT result codes

The ABBT test data are accumulated in a file and printed at a location you specify. Each horizontal row of data identify the ABBT test parameters in the first seven fields and the test results in the last two fields. The test results are represented by a result code followed by a short-form explanation of the status. The following table provides examples ABBT result codes.

ABBT test results error codes

MAP output	Meaning and action

```
----------------------------------------------------------------
SETSET NO                      NEW OFFICE LEN RESULT
NO OLD DN      NEW DN   FRAME UNIT LD   LC     CODE  EXPLANATION
----------------------------------------------------------------
0  7353020     7259020  1     1    10   2      0     'OK'
0  7253029     7259029  2     1    10   2      0     'OK'
0  7253040     7259040  0     0    10   3      1     'UNASSIGNED'
0  7253061     7259061  0     1    10   3      0     'OK'
0  7253062     7259062  2     0    10   5      2     'TEST ACCESS
                                                       FAIL'
0  7253074     7259074  2     1    10   5      0     'OK'
0  7253110     7259110  0     1    10   6      5     'OUTPULSING
                                                       FAIL'
0  7253191     7259191  1     0    10   5      0     'OK'
0  7353201     7359201  1     0    10   3      6     'SEIZE FAIL'
0  7253276     7259276  1     1    10   3      7     'TRUNK
                                                       OVERFLOW'
0  7253291     7259291  2     0    10   3      11    'IDLE FAULT
.  .           .        .     .    .    .      .     .
.  .           .        .     .    .    .      .     .
.  .           .        .     .    .    .      .     .
               *** TEST COMPLETED FOR ABBSET 0 ***
               *** TEST COMPLETED FOR ALL ABBSETS ***
```

Meaning: This illustration portrays the format of ABBT results data. The headers do not appear in the printed report. They are included in this example for explanatory purposes.

Action: None

```
0  7253730   7259730  2  1  10  8  0   'OK'
```

Meaning: The result code 0 (OK) in the example report line indicates that the ABBT was accomplished without error.

Action: None

-continued-

ABBT test results error codes (continued)

MAP output	Meaning and action
0 7253730 7259730 2 1 10 8 1 UNASSIGNED	
	Meaning: The result code 1 (UNASSIGNED) in the example report line indicates that there is no line equipment corresponding to the specified DN. Normal calls would get some kind of intercept treatment. In these cases, no attempt is made to query the old office.
	Action: None
0 7253730 7259730 2 1 10 8 2 TEST ACCESS FAIL BAD HORIZONTAL or 0 7253730 7259730 2 1 10 8 2 TEST ACCESS FAIL BAD VERTICAL or 0 7253730 7259730 2 1 10 8 2 TEST ACCESS FAIL SOFTWARE ERROR or 0 7253730 7259730 2 1 10 8 2 TEST ACCESS FAIL HORIZONTAL BUSY	
	Meaning: The result code 2 (TEST ACCESS FAIL) in the example report line indicates that incorrect information was specified for the MTA unit. The "test access fail" message could be followed by an indication of a bad horizontal, a bad vertical, a software error, a hardware failure, a busy horizontal, or a busy vertical.
	Action: For a bad horizontal, a bad vertical, or a software error, perform the following: 1. Enter the show abbtset *test_unit_no* command string and verify that enough columns were specified to access all the verticals used by this ABBT test unit. Also verify the number of the row and horizontal assigned to the ABBT test unit. 2. Using the TABLE directory Table Editor (TE) commands, check Table MTATRK to ensure that all entries have a selector specified as type B and that all entries are associated with the correct ABBT test unit number. If this data is incorrect, enter a define abbtset *test_unit_no* command string and redefine all parameters for this test unit. For a busy horizontal, use the TE commands to check Table MTATRK and ensure that no other trunk is associated with the horizontal for the ABBT test unit.
	-continued-

ABBT test results error codes (continued)

MAP output	Meaning and action
0 7253730	7259730 2 1 10 8 5 OUTPULSING FAIL
	Meaning: The result code 5 (OUTPULSING FAIL) in the example report line indicates an error that could be caused by incorrect connections to the outgoing (OG) trunk or information specified for the trunk is incorrect. Most likely, a stop-dial signal from the old office during outpulsing was encountered. (For example, a busy test connector in a step-by-step office produces this message.)
	Action: Enter the show general command string and show abbtset *test_unit_no* command string and examine the displayed information to ensure that the type of start signal for the notest trunk is correct and that the disconnect time is correct. An incorrect disconnect time often is indicated by a return code 7 (TRUNK OVERFLOW).
	Also, check the following:
	1. Verify that the MTADRIVER in the MTA unit is operating.
	2. Access the TTP level of the MAP and ensure that the trunk is in the RTS state.
	3. Verify that the REV key on the ABBT test unit is set correctly. An incorrect setting for the REV key often is indicated by a result code 6 (SEIZE FAIL).
0 7253730	7259730 2 1 10 8 6 SEIZE FAIL
	Meaning: The result code 6 (SEIZE FAIL) in the example report line indicates a problem either in the seizure protocol or in the connection between the outpulsing trunk (through the ABBT circuits) and the test trunk or equivalent in the old office, a combination of both.
	Action: Reverse the setting of the REV button on the ABBT test set and check the wiring from the OG trunk card to the old office.
	-continued-

ABBT test results error codes (continued)

MAP output	Meaning and action

`0  7253730  7259730  2  1  10  8  7  TRUNK OVERFLOW`

Meaning: The result code 7 (TRUNK OVERFLOW) in the example report line indicates that a connection could not be made to the old office. (That is, the notest trunk could not access a line in old office.) Trunk overflow faults can occur because the MTA unit for the old office is busy, or the old DN is of the incorrect type, such as the case when the old DN is subject to intercept treatment.

Action: Verify that the old DN is the correct type and try the test again. If the overflow persists, relays in the ABBT test unit may be operating too quickly. Using the define outptype command string, enter the relay parameter for the test sequence value and run ABBT again. If the difficulty disappears, specify a larger *disctime* variable value replacement when you enter the define offpars command string. Return the type of test sequence to its previous value and run ABBT on the line once more. Repeat this process until the difficulty is corrected.

`0  7253730  7259730  2  1  10  8  8  OFLO AT DMS MTA`

Meaning: The result code 8 (OFLO AT DMS MTA) in the example report line indicates that a connection could not be made through the MTA of the DMS to access the new line because the crosspoints of the MTA associated with that line were busy. That is, the vertical in the MTA of the accessed DMS is busy. This cannot occur if the testing order is conducted in LEN order.

Action: If this error condition persists, conduct testing in LEN order.

`0  7253730  7259730  2  1  10  8  11 IDLE FAULT`

Meaning: The result code 11 (IDLE FAULT) in the example report line indicates that the ABBT test set detected a known fault condition that does not occur normally and, therefore, has not been refined to either result code 51 or 55.

Action: If this error condition persists for any given line, it should be reported so that it can be classified.

-continued-

ABBT test results error codes (continued)	
MAP output	**Meaning and action**
0 7253730 7259730 2 1 10 8 12 BUSY FAULT	
	Meaning: The result code 12 (BUSY FAULT) in the example report line is similar to result code 11 (IDLE FAULT), except that the line was busy at the time of the error. Result code 12 indicates that the ABBT test set detected a known fault condition that does not occur normally and, therefore, has not been refined to either result code 51 or 55.
	Action: If this error condition persists for any given line, it should be reported so that it can be classified.
0 7253730 7259730 2 1 10 8 13 T/T REVERSAL	
	Meaning: The result code 13 (T/T REVERSAL) in the example report line can indicate that the tip and ring leads of the old and new lines are reversed. Aside from reversals, this result code can indicate mismatched party types in the case of a step-by-step office. For example, if party is ring party according to new office data but wired as tip party in the old office, this result code displays.
	Action: Check the cross-connections between the horizontal of the MTA unit for this ABBT test unit and the vertical for the new line. Try the test on a line associated with a different column of the MTA Unit, or try moving the horizontal to different set of cross-points. If the old office is a step-by-step office, the problem could be a mismatch between line class codes of the lines as assigned in the old and new offices.
0 7253730 7259730 2 1 10 8 14 LINE CLASS FAULT	
	Meaning: The result code 14 (LINE CLASS FAULT) in the example report line occurs when the data indicates that the line is a Private Branch Exchange (PBX) line but the ABBT results disagree.
	Action: Not currently available
-continued-	

ABBT test results error codes (continued)

MAP output	Meaning and action
0 7253730	7259730 2 1 10 8 15 PARTY FAULT

Meaning: The result code 15 (PARTY FAULT) in the example report line indicates that the new and old office disagree on whether or not the line is tip or ring party. If this problem occurs frequently, suspect an incorrect voltage setting of the variable battery.

Note: Result code 15 is not used for old step-by-step offices.   For step-by-step offices, refer to result code 13 (T/T REVERSAL).

Action: Choose a sample number of lines that are known to be connected properly and run ABBT repeatedly on those lines, adjusting the voltage of the variable battery each time until consistent results are obtained.

0 7253730	7259730 2 1 10 8 16 ABNORMAL SCAN

Meaning: The result code 16 (ABNORMAL SCAN) in the example report line indicates that the ABBT test set detected a condition which is supposed to be impossible. This can happen occasionally for a variety of reasons. For example, if a line is ringing at time of test, simplex voltage is incorrect for busy lines. If this problem occurs frequently, suspect an incorrect voltage setting of the variable battery.

Action: If this error condition persists, stop the run and test the ABBT test unit using the TEST button (lamps OT, OR, NT, NR, and STOP should light). Choose a sample number of lines that are known to be connected properly and run ABBT repeatedly on those lines, adjusting the voltage of the variable battery each time until consistent results are obtained.

0 7253730	7259730 2 1 10 8 17 BUSY START

Meaning: The result code 17 (BUSY START) in the example report line indicates that the line passed the basic test, but that the start or party test could not be performed because the line was busy. These results only occur if the optional loop versus ground start test has been requested or when the all parameter is specified for the define testtypes command string and the line in the old office is busy.

Action: If this error condition occurs while the line is not busy, ensure that the new line is in the cut-off condition and that the tip and ring voltage in the old office is set to the correct value.

-continued-

ABBT test results error codes (continued)

MAP output	Meaning and action
0 7253730	7259730 2 1 10 8 20 START FAULT

Meaning: The result code 20 (START FAULT) in the example report line indicates that the line passed the basic test but failed the optional loop versus ground start test. For example, the line is defined as loop-start according to new office data but equipped as ground-start in the old office. An origination on a ground-start line during this test produces a "test failed" result.

Action: If this problem occurs frequently, suspect an incorrect voltage setting of the variable battery. Choose a sample number of lines that are known to be properly connected and run ABBT repeatedly on those lines, adjusting the voltage of the variable battery each time until consistent results are obtained.

MAP output	Meaning and action
0 7253730	7259730 2 1 10 8 21 A/B BITS FAILURE

Meaning: The result code 21 (A/B BITS FAILURE) in the example report line indicates that the SMS never detected the A/B bit pattern for the channel test signaling pattern on the channel it was scanning. This fault is possible for an RCS in any mode of operation and indicates one of the following conditions:

1. The line equipment in the old switch and the new switch, associated with the DN being tested, does not terminate to the same subscriber loop.

2. The DN being tested is not assigned in the old switch.

3.   Test access to the line on the old switch is busy.

Action: Check the DN of the subscriber line as used in the old office, the test access availability to the subscriber line, and the line equipment designations for both the old and new switches.

-continued-

ABBT test results error codes (continued)

MAP output	Meaning and action

`0  7253730  7259730  2  1  10  8  22  DDL FAILURE`

Meaning: The result code 22 (DDL FAILURE) in the example report line indicates that the SMS never detected the trunk assign message on the DDL for the line being tested. This fault applies to an RCS operating in Mode II and indicates one of the following conditions:

1. The line equipment in the old switch and the new switch, associated with the DN being tested, does not terminate to the same subscriber loop.

2. The DN being tested is not assigned in the old switch.

3. Test access to the line on the old switch is busy.

Action: Check the DN of the subscriber line as used in the old office, the test access availability to the subscriber line, and the line equipment designations for both the old and new switches.

`0  7253730  7259730  2  1  10  8  23  ABBT IN PROGRESS FOR RCS`

Meaning: The result code 23 (ABBT IN PROGRESS FOR RCS) in the example report line indicates that the SMS notified ABBT software that an ABBT already is in progress for the RCS. This should not occur unless Table MTALME is datafilled improperly. Before an ABBT is requested on a RCS, the RCS is checked to see whether it has an ongoing ABBT.

Action: Check the datafill in Table MTALME to ensure that all MTA are datafilled correctly for RCS modules.

`0  7253730  7259730  2  1  10  8  24  UNEQUIPPED PORT ON RCS`

Meaning: The result code 24 (UNEQUIPPED PORT ON RCS) in the example report line indicates that the SMS has been instructed to scan on a channel associated with an unequipped port. This fault arises when the CC and SMS data for the ports to the RCS are inconsistent.

Action: An SMS audit will clear this fault.

`0  7253730  7259730  2  1  10  8  25  SPECIAL MODE II CASE OF RCS`

Meaning: The result code 25 (SPECIAL MODE II CASE OF RCS) in the example report line indicates that a line terminating on a single-circuit plug-in (channel unit) in one of the four slots in the far-right position of an RCS operating in Mode II cannot be tested using ABBT software.

Action: Not currently available

-continued-

ABBT test results error codes (continued)	
MAP output	**Meaning and action**
`0   7253730   7259730   2   1   10   8   26   INVALID SHELF MODE`	
	Meaning: The result code 26 (INVALID SHELF MODE) in the example report line indicates that the line to be tested is located on an RCS shelf which is in an invalid mode. Inconsistencies between the CC and SMS data on shelf modes cause this fault.
	Action: Correct the datafill in the RCSINV Table.
`0   7253730   7259730   2   1   10   8   27   OFLO AT DMS MTA FOR RCS`	
	Meaning: The result code 27 (OFLO AT DMS MTA FOR RCS) in the example report line indicates busy crosspoints for the minibar switch associated in Table MTALME with the RCS to be tested. This fault could arise if the vertical of the minibar switch was not released after an ABBT or if the ABBT terminated abnormally (that is, the vertical was connected and remained connected after the abnormal termination).
	Action: In the multiple ABBT environment, the ABBT software tests the line later. If not in the multiple ABBT environment, a reload restart clears the fault.
`0   7253730   7259730   2   1   10   8   28   RCS NOT MANBUSY`	
	Meaning: The result code 28 (RCS NOT MANBUSY) in the example report line indicates that the line to be tested belongs to an RCS that is not manually busy to the new, DMS-100 switch.
	Action: Post the RCS at the MAP and manually busy it.
`0   7253730   7259730   2   1   10   8   29   INVALID NUMBER OF CIRCUITS`	
	Meaning: The result code 29 (INVALID NUMBER OF CIRCUITS) in the example report line indicates that the channel unit to which the line to be tested connects had an invalid number of circuits. This indicates a data inconsistency between the CC and SMS about the number of circuits in the channel unit.
	Action: Correct the datafill in Table LNINV.
`0   7253730   7259730   2   1   10   8   30   TIMEOUT ON SMS TEST REPLY`	
	Meaning: The result code 30 (TIMEOUT ON SMS TEST REPLY) in the example report line indicates that the SMS did not return the ABBT test results to the CC within a specified time period. This fault could arise if the SMS went into a system busy state during the test.
	Action: Check the status of the SMS and its links at the MAP.
-continued-	

ABBT test results error codes (continued)

MAP output	Meaning and action

```
0   7253730   7259730   2  1  10  8       INVALID NODE NUMBER

or

0   7253730   7259730   2  1  10  8   31  RCS NODE STATUS NOT OBTAINED

or

0   7253730   7259730   2  1  10  8       VERTICAL RELEASE FAILURE
```

Meaning: These miscellaneous (MISC) errors produce a software error (SWERR). A SWERR log indicating that a fault occurred while testing the line is produced. Only one of the three errors uses the result code 31. The other two display a blank result code field.

The "INVALID NODE NUMBER" error has a blank result code field and indicates that the node of the line being tested is invalid.

The "RCS NODE STATUS NOT OBTAINED" error has a result code of 31 and indicates that the status of the RCS node could not be determined.

The "VERTICAL RELEASE FAILURE" error has a blank result code field and indicates that the vertical associated with the minibar switch and RCS failed to release.

Action: The ABBT continues, but when you attempt another test on a line that connects to this RCS, the test result file receives and stores the message, "OFLO AT DMS MTA FOR RCS."

-continued-

ABBT test results error codes (continued)

MAP output	Meaning and action

```
0   7253730   7259730   2   1   10   8   32   BAD VERTICAL

or

0   7253730   7259730   2   1   10   8   32    NO MESSAGE

or

0   7253730   7259730   2   1   10   8   32   SMS NODE NUMBER NOT OBTAINED

or

0   7253730   7259730   2   1   10   8   32   GET SOLICITOR NUMBER FAILED

or

0   7253730   7259730   2   1   10   8   32   ABBT MESSAGE PROBLEM

or

0   7253730   7259730   2   1   10   8   32   INVALID RETURN CODE FROM SMS/ABBT
```

Meaning: The result code 32 indicates a SWERR condition. A SWERR log indicating that a fault occurred while testing the line is produced for each type of result code 32 error. The error type for this result code is MISC except when a fault exists in the messaging from the CC to the SMS. In this case, the error type is Bad Record (BADRC).

The result code 32 (BAD VERTICAL) indicates the vertical of the minibar switch (used by DMS software to determine if an ABBT is ongoing on the RCS to which a second line, for which an ABBT is requested, connects) is invalid.

The result code 32 (NO MESSAGE) indicates that a problem exists in messaging from the CC to SMS. The result code 32 (SMS NODE NUMBER NOT OBTAINED) indicates that the SMS node number could not be determined. The result code 32 (GET SOLICITOR NUMBER FAILED) indicates that a problem exists in messaging from the CC to the SMS. The result code 32 (ABBT MESSAGE PROBLEM) indicates that a problem exists in messaging from the CC to the SMS. The result code 32 (INVALID RETURN CODE FROM SMS/ABBT) indicates that the test result the SMS returned to the CC was an unknown value.

Action: Not currently available

-continued-

ABBT test results error codes (continued)

MAP output	Meaning and action
0 7253730	`7259730  2  1  10  8  33  LINE NOT IDLE`
	Meaning: The result code 33 (LINE NOT IDLE) in the example report line indicates that the line to be tested is busy. The +116 volt dc potential sent to the RCS during the ABBT severely degrades the talking path of a busy line, so only idle lines are tested.
	Action: The ABBT for this line will be delayed and run later.
0 7253730	`7259730  2  1  10  8  38  FILE READ FAULT`
	Meaning: The result code 38 (FILE READ FAULT) in the example report line only occurs when the DNs to be tested are read from an input file and an error occurs while reading the file. The probability of this is very low.
	Action: The ABBT aborts.
0 7253730	`7259730  2  1  10  8  51  T OPEN, R OPEN REV`
	Meaning: The result code 51 (T OPEN, R OPEN REV) in the example report line indicates that the tip or ring is open.
	Action: Not currently available
0 7253730	`7259730  2  1  10  8  52  R OPEN, T OPEN REV`
	Meaning: The result code 52 (R OPEN, T OPEN REV) in the example report line indicates that the tip or ring is open.
	Action: Not currently available
	-continued-

ABBT test results error codes (continued)

MAP output	Meaning and action
0 7253730	7259730 2 1 10 8 53 T/R OPEN

Meaning: The result code 53 (T/R OPEN) in the example report line indicates (aside from open T and R leads) a faulty or missing line card in the new office or one of many possible faults in testing continuity.

Action: 1. If the cabling between the old and new offices is faulty, check connections to the old office.

2. The old office could have released the connection from the notest trunk to the line under test. Using the define outptype command string, use the relay parameter for the test sequence and run ABBT again, checking to make sure that no disconnect signal appears on the sleeve lead of the notest trunk. If a disconnection occurs, the sleeve lead current on the notest trunk may be low, a fault may exist in the ABBT circuit, or a relay in the MTA may be stuck.

3. The connection in the MTA unit between the line to the new office and the horizontal for this ABBT test unit could be faulty. Check the appropriate crosspoints in the MTA unit.

| 0   7253730 | 7259730   2   1   10   8   54   T OR R GROUND |

Meaning: The result code 54 (T OR R GROUND) in the example report line indicates that some ground conditions on busy lines are detected as reversals, or not at all. In an isolated case, this result code indicates also that a short is on an idle tip party line in a step-by-step office.

Action: Not currently available

| 0   7253730 | 7259730   2   1   10   8   55   T/R SHORT |

Meaning: The result code 55 (T/R SHORT) in the example report line indicates a short. In an isolated case, this result code indicates also that a short is a tip ground on a busy tip party line in a step-by-step office.

Action: Not currently available

End

continue

Function

Use the continue command to continue an ABBT from the point at which the test was halted using the ABBT directory stop command.

continue command parameters and variables	
Command	**Parameters and variables**
continue	all *unit*

Parameters and variables	**Description**
all	This parameter continues all ABBT testing associated with all ABBT test units.
unit	This variable specifies the number of the ABBT test unit. The valid entry range is 0-8.

Qualification

Setting up the range of DNs can take up to ten minutes.

Examples

The following table provides examples of the continue command.

Examples of the continue command		
Example	**Task, response, and explanation**	
continue all ↵		
	Task:	Continue ABBT testing for all ABBTs associated with all test units.
	Response:	System continues testing
	Explanation:	The system continues testing.
	-continued-	

continue (end)

Examples of the continue command (continued)	
Example	**Task, response, and explanation**
continue 3 ↵ *where* 3 specifies the number of the ABBT test unit	
	Task: Continue ABBT testing of a specified ABBT test unit. **Response:** `Specified ABBT Test Unit is being tested` **Explanation:** The system continues testing.
End	

Response

The following table provides an explanation of the response to the continue command.

Response for the continue command	
MAP output	**Meaning and action**
`continue as it is?`	
	Meaning: This response appears only if testing by line equipment number (LEN) has been requested. The ABBT software questions whether any changes have been made to the test set-up since the last test. Such changes could include new test definition criteria or changes caused by a system restart. **Action:** Enter either Yes or No. If Yes is entered, the ABBT software resumes testing immediately. If No is entered, the ABBT software sets up the range of DNs again, then resumes testing. *Note:* Setting up the range of DNs can take up to ten minutes, so enter Yes when possible.

define

Function

Use the define command to set up or modify test definition parameters for an ABBT. The define command requires a unique entry for each of the test definition parameters including the old office description data (offpars parameter), the test unit definition for one NT5X73AB test unit at a time (abbtset parameter), the range of DNs to be tested (dninput parameter), the name of the file to receive test results and results output device (outpfile parameter), the limits of test results data desired (outptype parameter), and the types of test performed for each DN (testype parameter).

The ABBT software provides prompts for each series of entries. However, as you become very familiar with ABBT directory operation, the definition criteria can be entered as single command strings instead of using the lengthy prompting sequence.

The ABBT software provides error detection. When an entry error is detected, the ABBT software begins prompting for data after the last correct entry in the sequence. Prompting continues until all necessary test definition data is entered.

define command parameters and variables	
Command	**Parameters and variables**
define	offpars *nbr_sets* *predial_delay* *disc_relay* *disc_time* *step_by_step* (1) (2) (3) (4)
define (continued)	(1) (2) ⎡ *outp_n* (3) ⎣ *outp_y* *pulse_type* *digit_nbr* *starts* *idgtime* ⎤ ↵ abbtset (1) (2) (3) (4)
define (continued)	(1) *set_nbr* ⎡ no (2) ⎣ yes ⎡ *relay_delay* ⎤ ⎦ *sd_mem* *sd_pm* *sd_nbr* *sd_ckt* (1) (2) (3) ⎣ *scan_delay* ⎦ (3) (4) (4)
define (continued)	(1) *sc_mem* *sc_pm* *sc_nbr* *sc_ckt* *out_mem* *out_pm* *out_nbr* (1) (2) (3) (4)
-continued-	

define (continued)

define command parameters and variables (continued)		
Command	**Parameters and variables**	
define (continued)	(1) *out_ckt* *vert* *horiz* *horizgrp* *nbr_verts* ↵ dninput *access_by1_tu* (1) (2) (2) (3) (3) (4) (4)	
define (continued)	(1) file *old_difdn* *input_file* (1) (2) *test_order* ↵ (2) (3) manual *unit_nbr* *new_startdn* *new_enddn* *old_startdn* (3) (4) (4)	
define (continued)	(1) outpfile *device_name* *file_name* ↵ outptype *results_output* ↵ (1) (2) (2) (3) (3) (4) (4)	
define (continued)	(1) testtype *testunit_nbr* *type_of_test* ↵ (2) (3) (4) (end)	
Parameters and variables	**Description**	
abbtset	This parameter defines information about one ABBT test unit. Since the abbtset test definition parameter defines information for only one NT5X73 ABBT test unit at a time, a unique define abbtset command string must be entered for each NT5X73AB test unit used. (The offpars parameter and its entries must be entered before the abbtset parameter and its entries.)	
access_by1_tu	This variable specifies whether or not the range of DNs only is accessible by a particular ABBT test unit. The DNs are considered to be accessible by a particular ABBT test unit if the test access device to the old office provides access to a range of DNs that cannot be accessed through an access device connected to another ABBT test unit. If the same range of DNs can be accessed through more than one test access device, the ABBT software assigns 30 of the DNs to each ABBT test unit and runs the tests. When testing is complete, the ABBT software assigns the next 30 DNs to each ABBT test unit and runs the tests again until all DNs are tested. The valid entry values are yes and no.	
device_name	This variable specifies the name of an output device to which accumulated ABBT results are sent. The valid entry value is an eight-character alphanumeric string.	
-continued-		

define (continued)

define command parameters and variables (continued)	
Parameters and variables	**Description**
digit _nbr	This variable specifies the number of digits that must be outpulsed by the Digital Multiplex System (DMS) to connect to a line in the old office. The valid entry range is 4-7. (This entry is required for step-by-step offices only.)
disc_relay	This variable specifies the relay in the NT5X73AB ABBT test unit which, when operated, disconnects the ABBT test unit from the old office. The valid entry values are the characters a and k. ***Note:*** For all old office types except C1EAX and NX1D, the relay a must be specified.
disc_time	This variable specifies the time in 100-millisecond (ms) increments required for the old office to release properly once a disconnect signal has been received from the DMS. For example, entering a value of 2 produces a disconnect time of 200 ms. The valid entry range is 1-300. (The typical disconnect time for most offices is 300-500 ms.)
dninput	This parameter defines the range of DNs to be tested.
file	This parameter indicates whether or not the list of DNs to be tested is contained in a file or is to be entered manually. If you use the file parameter, more than one range of DNs can be specified. This parameter must be followed by value replacements for the variables *old_dn, input_file*, and *test_order.*
file_name	This variable specifies the name of the file in which ABBT results are to be accumulated before being sent to the specified output device. The valid entry value is an eight-character alphanumeric string.
horiz	This variable specifies the row of the metallic test access (MTA) unit containing a number of MTADRIVERs on the horizontal associated with this ABBT test unit. The MTA of the new DMS is not connected during ABBT with remote concentrator SLC-96 (RCS) subscriber lines, even if the MTA is present. An arbitrary value within the valid entry range for the row parameter must be entered. The valid entry range is 0-127.
horizgrp	This variable specifies the number of the horizontal in the MTA unit connected to this ABBT test unit. The MTA of the new DMS is not connected during ABBT with RCS subscriber lines, even if the MTA is present. An arbitrary value within the valid entry range for the horizontal parameter must be entered. The valid entry range is 0-159.
-continued-	

define (continued)

define command parameters and variables (continued)	
Parameters and variables	**Description**
idgtime	This variable specifies the maximum time allowed (in 10 ms increments) between individual digits outpulsed on the no-test trunk to the old office. For example, entering a value of 3 specifies 30 ms between digits. The valid entry range is 2-100.
input_file	This variable specifies the name of file containing a range or ranges of DNs to be tested. The valid entry value is an eight-character alphanumeric string. An input file used for specifying DN for ABBT must contain entries in the following format: *bbt_set_nbr new_start_dn old_start_dn new_end_dn*
manual	This parameter indicates whether or not the list of DNs to be tested is contained in a file or is to be entered manually. If manual entry is requested, only one range of DNs can be specified. This parameter must be followed by value replacements for the variables *bbt_set_number, new_startdn, new_enddn, old_startdn,* and *test_order.*
nbr_sets	This variable specifies the number of simultaneous ABBTs to be performed. The valid entry range is 1-8. **Note:** The number of sets specified cannot exceed the number of NT5X73AB test units and associated hardware available, such as the outgoing (OG) trunk circuits, signal distribution (SD) points, scan points, test trunks, and so on. If the office does not have feature package NTX057BA, only one NT5X73AB test unit can be specified.
nbr_verts	This variable specifies the number of columns in the MTA unit that contain MTADRIVERs on the horizontal associated with this ABBT test unit. Specify enough columns to provide the ABBT horizontal with access to all new lines being tested. The valid entry range is 1-640.
new_enddn	This variable specifies the end DN of a range of DNs to be tested in the new office. The valid entry value is a seven-digit string in the range from 0000000-9999999.
new_startdn	This variable specifies the starting DN of a range of DNs to be tested in the new office. The valid entry value is a seven-digit string in the range from 0000000-9999999.
no	This parameter indicates that the location of the ABBT test unit is not the host site. When the location of the ABBT test unit is not the host site, you also must enter value replacements for the variables *relay_delay* and *scan_delay*.
offpars	This parameter defines information about the old office.
-continued-	

define (continued)

define command parameters and variables (continued)	
Parameters and variables	**Description**
old_difdn	This variable specifies whether the DNs being tested in the old office are the same as those in the new office. The valid entry values are yes and no.
old_startdn	This variable specifies the starting DN in a range of DNs to be tested in the old office. The DN is incremented by one each time a DN in the new office is tested. This continues until the range specified by variables *new start_dn* and *new end_dn* is reached. The valid entry value is a seven-digit string in the range from 0000000-9999999.
out_ckt	This variable specifies the number of the OG trunk circuit on the NT2X57 SD card connected to the ABBT test unit. The valid entry range is 0-29.
out _mem	This variable specifies an unused external trunk number associated with the OG trunk circuit card connected to this ABBT test unit. (This is the trunk used for outpulsing digits to the old office.) An unused external trunk number can be determined by examining Table TRKMEM. If a trunk number already in use is entered, the system prompts for the correct information. The valid entry range is 1-10000.
outpfile	This parameter defines both the name of the file in which test results are accumulated and the device to which accumulated test results are sent.
outp _n	This variable specifies whether the old office uses optimized outpulsing. The valid entry value is no. (This entry is required for step-by-step offices only.)
out_nbr	This variable specifies the number of the trunk module (TM) containing the OG trunk circuit connected to the ABBT test unit. The valid entry range is 0-2047.
out_pm	This variable specifies the type of peripheral module (PM) with the OG trunk circuit connected to the ABBT test unit. The valid entry values are tm0, tm2, tm4, tm8, mtm, and rsm.
outptype	This parameter defines the type of test results output. If you do not specify an output type, the system prompts for the required information.
outp _y	This variable specifies whether the old office uses optimized outpulsing. The valid entry value is yes. (This entry is required for step-by-step offices only.)
-continued-	

define (continued)

define command parameters and variables (continued)	
Parameters and variables	**Description**
predial_delay	This variable specifies the time delay (in hundreds of ms) between the DMS sending an off-hook on the no-test trunk to the old office and outpulsing digits on that trunk. For example, entering a value of 2 produces a delay of 200 ms. The valid entry range is 0-300. For old offices which indicate successful seizure of the no-test trunk by tip and ring lead reversals, a predial delay of zero must be specified. Entering zero causes the DMS to outpulse digits after receiving a specified number of tip and ring reversals from the old office. These reversals indicate successful seizure of the no-test trunk by the old office. The number of reversals is specified by the *starts* variable If anything other than zero is specified, outpulsing begins after the specified delay. All reversals of tip and ring leads put on the no-test trunk by the old office are ignored.
pulse_type	This variable specifies the type of pulsing, dial pulse (DP), or multifrequency (MF) used by the no-test trunk to the old office. The valid entry values are dp and mf. (This entry is required for step-by-step offices only.)
relay_delay	This variable specifies the time in 100 ms increments for a signal output from an SD point of the host DMS to travel to the ABBT test unit located at a remote site and to operate a relay in the ABBT test unit. For example, entering a value of 2 specifies a delay of 200 ms. The valid entry range is 0-100. This variable must be entered if the ABBT test unit is not the host site.
results_output	This variable specifies the type of test results output. The valid entry values are all, fail, failunassigned, relay, scan, test, or unassigned. ▪ Entering all produces test results of all types. ▪ Entering fail produces only results indicating a test failure. ▪ Entering failunassigned produces results indicating test failure or unassigned lines. ▪ Entering relay produces unassigned lines and simplex scans. This value stops tests before operation of each relay in the ABBT Test Unit; use the continue command to resume testing. ▪ Entering scan produces the same results as entering relay except that testing stops before reading each scan point. ▪ Entering test produces the same results as entering relay except that testing stalls before each line is tested. ▪ Entering unassigned performs no tests but indicates all unassigned lines.
-continued-	

define (continued)

define command parameters and variables (continued)	
Parameters and variables	**Description**
scan_delay	This variable specifies the time in 100 ms increments for a signal generated by an ABBT test unit located at a remote site to travel and to be read by a scan point of the host DMS. For example, entering a value of 2 specifies a delay of 200 ms. The valid entry range is 0-10. This variable must be entered if the ABBT test unit is not the host site.
sc_ckt	This variable specifies the number of the scan circuit on the NT0X10 card connected to the ABBT test unit. The scan circuit specified must be dedicated to ABBT uniquely. The valid entry range is 0-29.
sc_ mem	This variable specifies an unused group number associated with the scan circuit of an NT0X10 card connected to the ABBT test unit. An unused scan group number can be determined by examining Table SCGRP. If a number for a group already in use is entered, the system prompts for the correct information. The valid entry range is 0-511.
sc_nbr	This variable specifies the number of the PM containing the scan circuit connected to the ABBT test unit. The valid entry range is 0-2047.
sc_pm	This variable specifies the type of PM containing the scan circuit connected to the ABBT test unit. The valid entry values are tm0, tm2, tm4, tm8, mtm, and rsm.
sd_ckt	This variable specifies the number of the primary circuit on the NT2X57 card connected to the ABBT test unit. The SD card contains an even-numbered primary circuit and an odd-numbered secondary circuit. Each circuit contains seven SD points for a total of 14 SD points per card. Since ten SD points must be connected to each ABBT test unit, two SD circuits must be used. Both circuits must be on the same NT2X57 card. To ensure that this is the case, an even-numbered, primary circuit must be specified for this variable. The SD circuit specified must be dedicated uniquely to ABBT The valid entry value is an even-numbered digit in the range 0-28.
sd_mem	This variable specifies an unused group number associated with the primary SD circuit of an NT2X57 card connected to the ABBT test unit. An unused SD group number can be determined by examining Table SDGRP. If a number for a group already in use is entered, the system prompts for the correct information. The valid entry range is 0-51.
sd_nbr	This variable specifies the number of the PM containing the SD circuit connected to the ABBT unit. The valid entry range is 0-2047.
sd_pm	This variable specifies the type of PM containing the primary SD circuit connected to the ABBT test unit. The valid entry values are tm0, tm2, tm4, tm8, mtm, and rsm.
-continued-	

define (continued)

define command parameters and variables (continued)	
Parameters and variables	**Description**
set_nbr	This variable specifies the number of the NT5X73AB test unit for which information is being defined. The valid entry range is 0-7.
starts	This variable specifies the type of start signal provided by the old office to indicate seizure of the no-test trunk. This entry is required for step-by-step offices only. The valid entry values are im, xd, wk, or dd. The definitions of these values are as follows: ▪ Enter im for no tip or ring reversals. ▪ Enter xd for one reversal of the tip and ring leads. ▪ Enter wk for two reversals of the tip and ring leads. ▪ Enter dd for two reversals of the tip and ring leads.
step_by_step	This variable indicates whether or not the old office is a step-by-step office. The valid entry values are yes and no. If yes is entered, you have finished entries for the offpars test definition parameter. If no is entered, you must enter value replacements for variables *out_n* or *out_y, pulse_type, digit_nbr, starts, idgtime,* and *output_file_name.*
test_order	This variable specifies the testing order. The valid entry values are bylen and bydn. Entering bylen specifies testing in LEN order and entering bydn specifies testing in DN order. ***Note:*** If DNs in the old office are different from those in the new office, or if the old office is a step-by-step office that uses optimized outpulsing, the bydn value must be entered.
testtype	This parameter defines the types of tests to be performed for each DN.
testunit_nbr	This variable specifies the number of the ABBT test unit performing the types of tests to be performed. The valid entry range is 0-7.
-continued-	

define (continued)

define command parameters and variables (continued)	
Parameters and variables	**Description**
type_of _test	This variable specifies the type of test to be performed. The valid entry values are all, basic, class, and start. Test definitions follow.
	▪ Entering all performs all tests in sequence.
	▪ Entering basic tests for continuity and absence of tip and ring lead reversals.
	▪ Entering class performs a basic test plus a class of service test if the DN passes the basic test.
	▪ Entering start performs a basic test and a start test, provided the DN passes the basic test.
	Note: Entering all performs a basic test initially. If the DN passes the basic test, it performs a class test. If the DN passes both tests, a start test is performed.
unit_nbr	This variable specifies the number of the ABBT test unit used to test a range of DNs which are specified manually. This value only is required if the range of DNs is specified manually and is accessible by only one ABBT test unit. The valid entry range is 0-7.
vert	This variable specifies the first column of the DMS MTA unit containing an MTADRIVER which controls the horizontal associated with this ABBT test unit. (The horizontal of the MTA Unit used for this ABBT test unit may appear across several columns of the MTA Unit.) The *nbr_verts* variable indicates the number of columns over which the horizontal appears (starting from the value specified for the *vert* variable). The MTA of the new DMS is not connected during ABBT with RCS subscriber lines, even if the MTA is present. An arbitrary value within the valid entry range for the col parameter must be entered. The valid entry range is 0-639.
yes	This parameter indicates that the location of the ABBT test unit is the host site.
End	

Qualification

After the test is defined, use the ABBT directory start command to initiate the ABBT test.

define (continued)

Examples

The following table provides examples of the define command.

Examples of the define command	
Example	**Task, response, and explanation**
define offpars 3 0 a 3 yes dp 7 xd 2 ↵ *where* 3 specifies the number of simultaneous ABBTs to run 0 specifies the predial delay a specifies the test unit relay that disconnects the ABBT test unit from the old office 3 specifies the a delay of 300 ms before outpulsing digits from the no-test trunk dp specifies that the signals on the no-test trunk are DP-type 7 specifies that seven-digit numbers are pulsed out to the old office xd specifies the type of start signal (one reversal of the tip and ring leads) 2 specifies the maximum amount of time between individual digits outpulsed on the no-test trunk to the old office	
	Task: Define the old office using no-prompt entry mode. **Response:** Currently not available **Explanation:** This command defines the old office.
define offpars ↵	

	Task: Define the old office using using prompt entry mode.
	
	Response: ```Enter: nbr of sets predial delay discrelay```
	                  ```disctime sxs?```
	                  ```>3 0 a 3 yes```
	                  ```Enter: pulse type digit nbr starts idgtime```
	                  ```>dp 7 xd 2```
	
	Explanation: There are no default values for the offpars parameter, so the system prompts you for entries since no data is specified. These entries specify that three NT5X73 ABBT test units are installed at the host DMS site. The time between sending an off-hook signal on the no-test trunk to the old office and outpulsing digits on that trunk is specified by a predial delay of zero. The old office is an NTL office of the No. 5 XBAR type (step-by-step). Signals on the no-test trunk are of the DP type and seven-digit numbers are pulsed to the old office. The no-test trunks from the DMS to the old office use XD start signals (one tip and ring lead reversal) and there is a delay of 300 ms before digits outpulse on the no-test trunk. The outpulse time delay is 20 ms.

-continued-	

define (continued)

Examples of the define command (continued)

Example	Task, response, and explanation

---

define abbtset 0 yes 3 mtm 3 22 5 mtm 2 28 0 tm8 0 20 0 0 11 1 ↵
where

0	specifies the set number
3	specifies the unused group in Table SDGRP to which the SD circuit is connected
mtm 3 22	specifies the location of the SD primary circuit
5	specifies the unused group in Table SCGRP to which the SCAN circuit is connected
mtm 2 28	specifies the location of the SCAN circuit
0	specifies the unused group in Table TRKMEM to which the OG trunk circuit is connected
tm8 0 20	specifies the location of the trunk circuit
0 0	specifies the column and row of the MTA unit containing the first MTADRIVER on the horizontal used by ABBT test unit 0
11	specifies the horizontal of the MTA unit used by the ABBT test unit
1	specifies the number of columns over which the horizontal appears starting at column 0

Task:	Define an ABBT test unit using no-prompt entry mode.
Response:	Currently not available
Explanation:	This command defines an ABBT test unit.

-continued-

---

define (continued)

Examples of the define command (continued)	
Example	**Task, response, and explanation**

define abbtset ↵	
Task:	Define an ABBT test unit using prompt entry mode.
| **Response:** | ```
Enter:  set_number  at host?
>O Y
sd mem pm nbr ckt sc mem  pm  nbr  ckt out mem
pm  nbr  ckt vert horiz horizgrp nbr verts
>3 MTM 3 22 5 MTM 2 8 0 TM8 0 20 0 0 11 1
``` |
| **Explanation:** | There is no default entry for the abbtset test definition parameter, so the system prompts you for entries. The responses to the prompts in this example specify that the ABBT test unit is located at the host DMS site. The primary SD circuit connected to ABBT Unit 0 is to be assigned to unused group 3 in Table SDGRP. The SD primary circuit is located in MTM 3 22. The SCAN circuit connected to ABBT unit 0 is to be assigned to unused group 5 in Table SCGRP. The SCAN circuit is located in MTM 2 28. The outgoing trunk circuit connected to ABBT test unit 0 is to be assigned as unused trunk member 0 in Table TRKMEM. The trunk circuit is located in TM8 0 20. The column and row of the MTA Unit containing the first MTADRIVER on the horizontal used by ABBT test unit 0 are 0 and 0 respectively. Starting at column 0, the horizontal appears over one column. The horizontal of the MTA Unit used by ABBT Test Unit 0 is horizontal 11. |
| **-continued-** | |

Examples of the define command (continued)

Example   **Task, response, and explanation**

define dninput no manual 6210000 6210099 7770000 bydn ↵
where

no	specifies that the range of DNs are not accessible by a particular ABBT test unit
6210000	specifies the new starting DN number in the range
6210099	specifies the new ending DN number in the range
7770000	specifies the old DN starting number
bydn	specifies the testing order

Task: Define the range of DN to be tested using no-prompt entry mode.

Response: Currently not available

Explanation: This command manually defines the range of DNs to be tested.

define   dninput ↵

Task: Specify the range of DNs to be tested using using prompt entry mode.

Response:
```
Enter:  dn_with_set? input type
>no manual
Enter: new start_dn new end_dn old start_dn
testing order
>6210000  6210099  7770000  bydn
```

Explanation: There is no default entry for the dninput test definition parameter, so the system prompts you for entries. The response to the prompt in this example specifies that the DN range to be tested in the new office is from 621-0000 to 621-0099. The range of DNs are not associated with one particular ABBT test unit. (Testing is conducted in DN order.)

-continued-

define (continued)

Examples of the define command (continued)	
Example	**Task, response, and explanation**

define outpfile prt1 bbt1 ↵
where

prt1	specifies the output device name
bbt1	specifies the output file name

	Task:	Specify the output system file and output device using no-prompt entry mode.
	Response:	Currently not available
	Explanation:	This command specifies the system file in which results are accumulated and the output device to which the accumulated results are sent.

define   outpfile ↵

	Task:	Specify the output system file and output device using prompt entry mode.
	Response:	`Enter:  output device name   output file name` `>prt1 bbt1`
	Explanation:	There is no default entry for the outpfile test definition parameter, so the system prompts you for an entry if no test results file or output device is specified. The response to the prompt in this example specifies the system file in which results are accumulated (bbt1) and the output device to which the accumulated results are sent (prt1).
-continued-		

Examples of the define command (continued)	
Example	**Task, response, and explanation**

define outptype all ↵		
where		
all	specifies the type of test results output	
	Task:	Specify the way the test results are produced using no-prompt entry mode.
	Response:	Currently not available
	Explanation:	This command defines the test results output as all. Specifying all produces test results of all types.

define outptype ↵		
	Task:	Specify the way the test results are produced using prompt entry mode.
	Response:	`Enter:  type of output` `>all`
	Explanation:	There is no default entry for the outptype test definition parameter, so the system prompts you for an entry. The response to the prompt in this example specifies the test results output as all. Specifying all produces test results of all types.

-continued-

define (continued)

Examples of the define command (continued)	
Example	**Task, response, and explanation**
define testtype basic ↵ *where* basic specifies the type of test to be performed	
	Task: Specify the test type using no-prompt entry mode.
	Response: Currently not available
	Explanation: This command specifies the test type as basic. The basic selection tests for continuity and absence of tip and ring lead reversals.
define testtype ↵	
	Task: Specify the test type using prompt entry mode.
	Response: `Enter:  test type` `>basic`
	Explanation: There is no default entry for the testtype test definition parameter, so the system prompts you for an entry. This command specifies the test type as basic. The basic selection tests for continuity and absence of tip and ring lead reversals.
End	

define (end)

Responses

The following table provides explanations of the responses for the define command.

Responses for the define command	
MAP output	**Meaning and action**
`could not add <tuple_name> to table <tuple_name>`	
	Meaning: Entering a define command string changes the contents of some system data tables. The ABBT software was unable to do this in the specified tables due to a system fault or because the define offpars command string and the define abbtset command string were entered in the wrong order.
	Action: Enter the define offpars command string before the define abbtset command string. Otherwise, contact your maintenance support group.
`Tuple SC_MEM OUT_MEM is already  in SCGRP.  Do you want me to overwrite the existing tuple?`	
	Meaning: The group number you entered already is assigned in Table SCGRP.
	Action: Verify with qualified personnel that overwriting data is allowed. Enter Y to overwrite the existing tuple. Or, enter N to leave the existing tuple in place and abort the prompt sequence.
`Tuple SD_MEM is already  in SDGRP.  Do you want me to update the existing tuple?`	
	Meaning: The group number you entered already is assigned in Table SDGRP.
	Action: Verify with qualified personnel that overwriting data is allowed. Enter Y to overwrite the existing tuple. Or, enter N to leave the existing tuple in place and abort the prompt sequence.
`Tuple BBTOUT OUT_MEM is already  in TRKMEM.  Do you want me to update the existing tuple?`	
	Meaning: The external trunk number you entered already is assigned in Table TRKMEM for some other purpose.
	Action: Verify with qualified personnel that overwriting data is allowed. Enter Y to overwrite the existing tuple. Or, enter N to leave the existing tuple in place and abort the prompt sequence.

help

Function

Use the help command to receive online documentation for the ABBT directory.

help command parameters and variables	
Command	**Parameters and variables**
help	*all* *command_nam*

Parameters and variables	Description
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
command_nam	This variable specifies a valid ABBT directory command name. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help abbt ↵ *where*	
abbt	specifies the help query
	Task: Access online documentation.
	Response: `This is to enter ABBT.  Subcommands are : DEFINE, START, STOP, ERRFILE, SHOW, CONTINUE, CLEAR, QUIT.`
	Explanation: This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

quit

Function

Use the quit command to exit the ABBT directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	$\begin{bmatrix} \textit{1 level} \\ \text{all} \\ \textit{name} \\ \textit{n\_levels} \end{bmatrix}$
Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	CI:
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
CI:	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
QUIT -- Increment not found	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
QUIT -- Unable to quit requested number of levels	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

show

Function

Use the show command to display general information concerning the old office as well as testing data and associated equipment information for a specifies ABBT test unit.

show command parameters and variables		
Command	**Parameters and variables**	
show	abbtset *unit* general	
Parameters and variables	**Description**	
abbtset	This parameter displays information associated with a particular abbtset unit.	
general	This parameter displays general information such as data concerning the old office, the ranges of DNs tested, and the number of tests that are running.	
unit	This variable specifies the number of the ABBT test unit for which the associated information displays. The valid entry range is 0-8.	

Qualification

If a define dninput command string is entered in order to redefine the ranges of DNs tested, the show command does not display the updated information until the ABBT software rearranges the DNs. This occurs only after the ABBT directory start command is entered.

Examples

The following table provides examples of the show command.

show (continued)

Examples of the show command

Example	Task, response, and explanation

show   abbtset   1 ↵
where

1	defines the number of the ABBT test unit

Task:	Display information for a specified ABBT test unit.
| **Response:** | ```
ABBT 1 :
-----------
Test Type is: ALL - all tests if basic test
passed.
EQUIPMENT ASSOCIATED :
  SDCARD :   SCSDGRPNO 5, on MTM 3 22.
  SDCARD :   SCSDGRPNO 5, on MTM 2 8.
  BBTOUT 0 : TM8 0 20.
  MTA HORIZONTAL: Vertical 235.
        : Horizontal 10 horizontal group 9.
     Nbr of verticals : 16.
TESTING INFO :
  Testing order is by DN.
  Dn range presently associated with set :
     NEW OFFICE START DN : 722 4222
     NEW OFFICE END DN   : 722 4222
     OLD OFFICE START DN : 621 1234
  LAST DN DONE: 722 4222
  STATUS: NOT currently active.
``` |
| **Explanation:** | This response displays information defining ABBT test unit 1. |

-continued-

---

show (continued)

Examples of the show command (continued)

Example	Task, response, and explanation

show   general ↵

	Task:	Display general information concerning the old office.
	Response:	GENERAL DATA :

```
            OFFPARS :
                You have 2 bbt sets,
                PRE_DIAL DELAY : 0 secs,
                Disconnect relay : A
                Disconnect time : 50 tenms,
                The old office IS NOT SXS,
                Old office PULSING type: DP,
                Number of digits to outpulse to the old
office: 5
                Interdigital time : 10
                Start signal of 'BBTOUT' : XD

            DNINPUT :
                The DN range is linked to set 0
                and we currently are testing
                    START        : 722 4222
                    TO           : 722 4222
                    WITH OLD NBT : 621 1234

            0 boards/processes are currently actively
running.
```

	Explanation:	This response provides general information about the old office.

End

Responses

The following table provides explanations of the responses to the show command.

show (continued)

Responses for the show command

MAP output   Meaning and action

```
ABBT 1 :
-----------
Test Type is: ALL - all tests if basic test passed.
EQUIPMENT ASSOCIATED :
  SDCARD :   SCSDGRPNO 5, on MTM 3 22.
  SDCARD :   SCSDGRPNO 5, on MTM 2 8.
  BBTOUT 0 : TM8 0 20.
  MTA HORIZONTAL: Vertical 235.
             : Horizontal 10 horizontal group 9.
     Nbr of verticals : 16.
TESTING INFO :
  Testing order is by DN.
  Dn range presently associated with set :
     NEW OFFICE START DN : 722 4222
     NEW OFFICE END DN   : 722 4222
     OLD OFFICE START DN : 621 1234
  LAST DN DONE: 722 4222
  STATUS: NOT currently active.
```

Meaning: The system displays information defining an ABBT test unit.

Action: None

-continued-

Responses for the show command (continued)

MAP output   Meaning and action

```
GENERAL DATA :

   OFFPARS :
      You have 2 bbt sets,
      PRE_DIAL DELAY : 0 secs,
      Disconnect relay : A
      Disconnect time : 50 tenms,
      The old office IS NOT SXS,
      Old office PULSING type: DP,
      Number of digits to outpulse to the old office: 5
      Interdigital time : 10
      Start signal of 'BBTOUT' : XD

   DNINPUT :
      The DN range is linked to set 0
      and we currently are testing
         START         : 722 4222
         TO            : 722 4222
         WITH OLD NBT : 621 1234

0 boards/processes are currently actively running.
```

Meaning: The system displays information defining the old office.

Action: None

End

start

Function

Use the start command to run the ABBT associated with a specified ABBT test unit.

start command parameters and variables	
Command	**Parameters and variables**
start	all *unit*

Parameters and variables	Description
all	This parameter starts all tests associated with all ABBT test units.
unit	This variable specifies the number of the ABBT test unit for which the associated ABBT is to be run. The valid entry range is 0-8.

Qualifications

The start command is qualified by the following exceptions, restrictions, and limitations:

- The start command does not execute unless you complete the following actions:
  - Enter a define abbtset command string for each ABBT test unit that is connected.
  - Enter the define command once with each of the parameters (offpars, dninput, outpfile, outptype, and testtype).
- If the test is underway, you must use the stop command to end the test before it can be restarted.
- The outgoing (OG) trunk to the old office must be in one of the following states in order to begin ABBT:
  - SB (system busy)
  - IDL (idle)
  - INI (initialize)

  Note: To ensure that the trunk is in a valid state, access the trunk test position (TTP) level of the MAP, post, busy (BSY), and return the trunk to service (RTS).

- If you enter the start command without parameters, the system prompts for the required information.

start (continued)

- Since the COT is not slave-timed from the SMS during ABBT for RCS subscriber lines (the SMS and COT clocks are not in synchrony), slips will occur on the DS-1 links. When a DS-1 frame slips, DDL framing is lost; the SMS resynchronizes DDL framing and sends an unsolicited message to the DMS switch. The extent of the timing mismatch will determine how frequently slips occur. When slips occur, a software error (SWERR) log is printed. These occurrences do not affect SLC-96 or DMS-100 operation. If a line fails the ABBT because of a frame slip, it will be retested.

Example

The following table provides an example of the start command.

Example of the start command		
Example	**Task, response, and explanation**	
start 1 ↵ *where*		
1	specifies the ABBT test unit number	
	Task:	Start the ABBT test associated with the specified test unit.
	Response:	`Process/Board 1 started.`
	Explanation:	The system indicates that the ABBT test for test unit 1 started. The accumulated test results are sent to the specified output device.

Response

The following table provides an explanation of the response to the start command.

start (end)

Response for the start command	
MAP output	**Meaning and action**
Software is setting up the dn ranges. It might take up to 10 minutes, and will tell you when the test starts.	
	Meaning: This response appears only if testing a line equipment number (LEN) order is specified. Up to ten minutes can be required to set up the DNs in LEN order before testing begins. Before the test begins, a list of unassigned lines display. When the test completes, accumulated test results are sent to the specified output device.
	Action: None

stop

Function

Use the stop command to stop the ABBT associated with a specified ABBT test unit. If the stop command is entered without parameters, the system prompts for the required information.

stop command parameters and variables	
Command	**Parameters and variables**
stop	all *unit*

Parameters and variables	Description
all	This parameter stops all tests associated with all ABBT test units.
unit	This variable specifies the number of the ABBT test unit for which the associated ABBT is to be stopped. The valid entry range is 0-8.

Qualifications

None

Example

The following table provides an example of the stop command.

Example of the stop command		
Example	**Task, response, and explanation**	
stop all ↵		
	Task:	Stop ABBT testing.
	Response:	Not currently available
	Explanation:	ABBT testing has been stopped.

stop (end)

Response

The following table provides an explanation of the response to the stop command.

Response for the stop command	
MAP output	**Meaning and action**
`INPUT A NBR BETWEEN 0 AND 7, OR 'ALL':  <PROCESS NUMBER OF 'ALL'> STRING` `Enter: <PROCESS NUMBER OF 'ALL'>`	
	Meaning: The test unit number you specified is invalid.
	Action: Reissue this command string with a valid test unit number.

ACDMR level commands

Use the ACDMR level of the MAP with the Meridian SL-100 Integrated Services Network to provide equal distribution of incoming calls to a predesignated group of telephone sets.

The send command of the ACDMR directory routes Automatic Call Distribution (ACD) data to a datalink pool. The stop command deletes routing information for ACD groups. The init command manually downloads configuration data to the down stream processor (DSP). The mrstat command routes information for one ACD group or all ACD groups.

To use these commands properly, you should be familiar with Tables ACDGRP and SLLNKDEV. In addition, you should be familiar with the LNKUTIL (Link Utility) directory commands devcon, devstart, poolstart, devstop, and devdisc.

Accessing the ACDMR level

To access the ACDMR level, enter the following command from the CI level:

acdmr ↵

ACDMR commands

The commands available at the ACDMR MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

ACDMR commands	
Command	**Page**
help	A-55
init	A-57
mrstat	A-59
quit	A-63
-continued-	

ACDMR commands (continued)	
Command	**Page**
send	A-67
stop	A-73
End	

help

Function

Use the help command to receive online documentation for the ACDMR directory.

help command parameters and variables	
Command	**Parameters and variables**
help	*all* *command_nam*

Parameters and variables	Description
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
command_nam	This variable specifies a valid ACDMR directory command name. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command		
Example	**Task, response, and explanation**	
help ↵		
	Task:	Access online documentation.

	Response:	`Enter the ACD Management Reports (ACDMR)`
		`increment.  The available commands are::`
		`Quit, Send, Stop, Init, MRStat`

	Explanation:	This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

init

Function

Use the init command to start downloading the ACDMR configuration data to a DSP.

init command parameters and variables	
Command	**Parameters and variables**
init	*poolname*
Parameters and variables	**Description**
poolname	This variable specifies the name of the pool identified by the devcon command in the LNKUTIL directory.

Qualifications

The init command is qualified by the following exceptions, restrictions, and limitations:

- The init command does not prompt for the ACD group name. The pool name must be datafilled with its ACD groups using the ACDMR directory send command.
- The SLNK104 log indicates the beginning of the initialization process and the SLNK105 log indicates the end of the initialization process.

Example

The following table provides an example of the init command.

Example of the init command	
Example	**Task, response, and explanation**
init poolname1 ↵ *where* poolname1 specifies the name of the pool	
	Task: Initiate downloading the ACDMR to a DSP.
	Response: INITIALIZATION HAS BEEN STARTED.
	Explanation: This command initiates the download. The download is verified by the SLNK104 log.

init (end)

Responses

The following table provides explanations of the responses to the init command.

Responses for the init command	
MAP output	**Meaning and action**
INITIALIZATION PROCESS HAS FAILED TO START.	
	Meaning: The name of the pool was entered and the command aborted.
	Action: Verify the name of the pool using the ACDMR directory mrstat command. Verify that ACD groups are datafilled in Table ACDGRP.
NO ACD GROUPS ARE ROUTED TO POOL POOLNAME1.	
	Meaning: No ACD groups were assigned to the specified pool.
	Action: Verify the name of the pool using the ACDMR directory mrstat command. Verify that ACD groups are datafilled in Table ACDGRP.
NO DATALINKS ARE ASSIGNED FOR POOL POOLNAME1.	
	Meaning: No datalinks were assigned to the specified pool.
	Action: Verify the name of the pool using the ACDMR directory mrstat command.
NO DATALINKS HAVE BEEN STARTED FOR POOL POOLNAME1.	
	Meaning: The datalink assigned to the specified pool was in the dead or disconnected state.
	Action: Reissue the init command.

mrstat

Function

Use the mrstat command to display ACDMR status information.

mrstat command parameters and variables	
Command	**Parameters and variables**
mrstat	all
	group *groupname*
	pool *poolname*

Parameters and variables	Description
all	This parameter displays ACDMR information for all ACD groups.
group	This parameter displays ACDMR information for a particular group.
groupname	This variable specifies the name of the ACD group.
pool	This parameter displays information about a specified pool.
poolname	This variable specifies the name of the pool.

Qualifications

None

Examples

The following table provides examples of the mrstat command.

Examples of the mrstat command	
Example	**Task, response, and explanation**
mrstat all ↵	
	Task: Display all ACD groups with their associated pool.
	Response: THE FOLLOWING ACD GROUPS ARE ROUTED TO POOL POOLNAME1: GROUPNAME2 GROUPNAME8 GROUPNAME9

THE FOLLOWING ACD GROUPS ARE ROUTED TO POOL POOLNAME5: GROUPNAME4   GROUPNAME5   GROUPNAME6 |
| | **Explanation:** This command displays all ACD groups with their associated pool. |
| **-continued-** | |

mrstat (continued)

Examples of the mrstat command (continued)	
Example	**Task, response, and explanation**
mrstat group groupname5 ↵ *where* groupname5 specifies the ACD group for which information is requested	
	Task: Display the name of the pool to which the ACD groups are routed.
	Response: `ACD GROUP GROUPNAME5 IS ROUTED TO POOL POOLNAME5.`
	Explanation: This command results in a message that the ACD group named groupname5 is routed to the pool named poolname5.
mrstat pool poolname1 ↵ *where* poolname1 specifies the pool for which information is requested	
	Task: Display the ACD groups routed to a specified pool.
	Response: `THE FOLLOWING ACD GROUPS ARE ROUTED TO` `POOLNAME1: GROUPNAME2    GROUPNAME8    GROUPNAME9`
	Explanation: This command displays ACD groups routed to poolname1.
End	

Responses

The following table provides explanations of the responses to the mrstat command.

Responses for the mrstat command	
MAP output	**Meaning and action**
`ACD GROUP GROUPNAME1 HAS NOT BEEN ROUTED.`	
	Meaning: The ACD group was not assigned to a pool.
	Action: Assign the ACD group to a pool using the LNKUTIL directory devcon command.
-continued-	

mrstat (end)

Responses for the mrstat command (continued)	
MAP output	**Meaning and action**
`NO ACD GROUPS HAVE BEEN ROUTED.`	
	Meaning: None of the pools contained ACD groups.
	Action: Verify the name of the ACD group. Route the ACD groups to a pool using the LNKUTIL directory devcon command followed by using the ACDMR directory send command.
`NO ACD GROUPS HAVE BEEN ROUTED TO POOL POOLNAME2.`	
	Meaning: No ACD groups are routed to the specified pool.
	Action: Verify the name of the pool in Table SLLNKDEV.
End	

quit

Function

Use the quit command to exit the ACDMR directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⎡ *1 level* all *name* *n_levels* ⎦

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	CI:
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut		
	specifies a directory	
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

send

Function

Use the send command to route management reports for specific ACD groups to a particular pool.

send command parameters and variables		
Command	**Parameters and variables**	
send	*poolname* ⎡ all ⎤ ⎣ *groupname* ⎦	
Parameters and variables	**Description**	
all	This parameter routes all ACD groups to a specified pool.	
groupname	This variable specifies the name or names of the ACD group or groups datafilled in Table ACDGRP and routed to a specified pool If you enter more than one ACD group, separate each ACD group name from the next by a single space.	
poolname	This variable specifies the name of the pool created by the LNKUTIL directory devcon command.	

Qualifications

The send command is qualified by the following exceptions, restrictions, and limitations:

- The ACD groups must be datafilled in Table ACDGRP with the management reports (mgtrpt) option.
- The device to be used as a datalink must be added to Table SLLNKDEV.
- You must use the LNKUTIL directory commands devcon (in which the pool names are assigned), devstart, devstop, and devdisc to initiate and control the datalink.

Examples

The following table provides examples of the send command.

send (continued)

Examples of the send command
Example   **Task, response, and explanation**

send pool1 all ↵
where

pool1   specifies the name of the pool to which the groups are routed

Task:	Route all ACD groups to a specified pool.
Response:	ACD GROUP ACDGRP1 HAS BEEN ASSIGNED TO POOL POOL1.
	ACD GROUP ACDGRP2 HAS BEEN ASSIGNED TO POOL POOL1.
	ACD GROUP ACDGRP3 HAS BEEN ASSIGNED TO POOL POOL1.
	ACD GROUP ACDGRP4 HAS BEEN ASSIGNED TO POOL POOL1.
	ACD GROUP ACDGRP5 HAS BEEN ASSIGNED TO POOL POOL1.
	ACD GROUP ACDGRP6 HAS BEEN ASSIGNED TO POOL POOL1.
	ACD GROUP ACDGRP7 HAS BEEN ASSIGNED TO POOL POOL1.
	ACD GROUP ACDGRP8 HAS BEEN ASSIGNED TO POOL POOL1.
	ACD GROUP ACDGRP9 HAS BEEN ASSIGNED TO POOL POOL1.
	ROUTE LIST HAS BEEN UPDATED FOR POOL POOL1.
Explanation:	This command routes ACD groups named acdgrp1, acdgrp2, acdgrp3, acdgrp4, acdgrp5, acdgrp6, acdgrp7, acdgrp8, and acdgrp9 to the pool named pool1.

send   pool1 acdgrp1 ↵
where

pool1   specifies the name of the pool to which the specified group is routed
acdgrp1   specifies the name of the ACD group

Task:	Route one ACD group to a specified pool.
Response:	ACD GROUP ACDGRP1 HAS BEEN ASSIGNED TO POOL POOL1.
Explanation:	This command routes the ACD group named acdgrp1 to the pool named pool1.

-continued-

send (continued)

Examples of the send command (continued)

Example   Task, response, and explanation

send pool1 acdgrp1 acdgrp2 acdgrp3 acdgrp4 acdgrp9 ↵
where

pool1	specifies the name of the pool to which the specified group is routed
acdgrp1	specifies one of five ACD group names
acdgrp2	specifies one of five ACD group names
acdgrp3	specifies one of five ACD group names
acdgrp4	specifies one of five ACD group names
acdgrp9	specifies one of five ACD group names

Task:	Route multiple ACD groups to a specified pool.
Response:	ACD GROUP ACDGRP1 HAS BEEN ASSIGNED TO POOL POOL1. ACD GROUP ACDGRP2 HAS BEEN ASSIGNED TO POOL POOL1. ACD GROUP ACDGRP3 HAS BEEN ASSIGNED TO POOL POOL1. ACD GROUP ACDGRP4 HAS BEEN ASSIGNED TO POOL POOL1. ACD GROUP ACDGRP9 HAS BEEN ASSIGNED TO POOL POOL1.
Explanation:	This command routes ACD groups named acdgrp1, acdgrp2, acdgrp3, acdgrp4, and acdgrp9 to the pool named pool1.

End

send (continued)

Responses

The following table provides explanations of the responses to the send command.

Responses for the send command	
MAP output	**Meaning and action**
| ```
ACD GROUP GROUPNAME1 HAS BEEN ASSIGNED TO POOL POOLNAME1
ACD GROUP GROUPNAME2 HAS BEEN ASSIGNED TO POOL POOLNAME1
ACD GROUP GROUPNAME3 HAS BEEN ASSIGNED TO POOL POOLNAME1

GROUP GROUPNAME_ABCDEF DOES NOT EXIST.

REMAINING GROUPS WILL BE IGNORED.
``` | |
| | **Meaning:** Multiple ACD groups were entered. One of the ACD groups does not exist. The remaining ACD groups are ignored.

Action: Datafill the specified ACD group in Table ACDGRP. |
| ```
ACD GROUP GROUPNAME1 HAS BEEN REMOVED FROM POOL OLD_POOLNAME1.
ACD GROUP GROUPNAME1 HAS BEEN ASSIGNED TO POOL NEW_POOLNAME.

ACD GROUP GROUPNAME2 DOES NOT HAVE MGTRPT OPTION.
ACD GROUP GROUPNAME2 HAS NOT BEEN ASSIGNED TO POOL NEW_POOLNAME.

ACD GROUP GROUPNAME3 HAS BEEN REMOVED FROM POOL OLD_POOLNAME2.
ACD GROUP GROUPNAME3 HAS BEEN ASSIGNED TO POOL NEW_POOLNAME.

ACD GROUP GROUPNAME4 DOES NOT HAVE MGTRPT OPTION.
ACD GROUP GROUPNAME4 HAS NOT BEEN ASSIGNED TO POOL NEW_POOLNAME.

ROUTE LIST HAS BEEN UPDATED FOR POOL NEW_POOLNAME.
``` | |
| | **Meaning:** Multiple ACD groups were entered. Some ACD groups do not have the mgtrpt option applied and the rest of the groups currently are assigned to another pool.

Action: Add the mgtrpt option in Table ACDGRP for the ACD groups specified in the display. Otherwise, the route list is updated for the new pool. |
| ```
AT LEAST ONE ACD GROUP MUST BE SPECIFIED.
``` | |
| | **Meaning:** No ACD groups were entered and the command aborted.

Action: Enter a valid ACD group name from Table ACDGRP. |
| **-continued-** | |

send (end)

Responses for the send command (continued)	
MAP output	**Meaning and action**
`NO ACD GROUPS WERE ROUTED TO POOL POOLNAME2.`	
	Meaning: The names of the ACD groups were entered and the command aborted.
	Action: Verify the name of the pool using the mrstat command in this directory. Verify that all of the ACD groups are datafilled in Table ACDGRP. Verify that the mgtrpt option is added to each ACD group in Table ACDGRP.

`THE NUMBER OF DATALINKS ASSIGNED TO POOL POOLNAME1 IS 1.`
`NO MORE DATALINKS MAY BE ASSIGNED.`
`NO ACTION TAKEN.`	

	Meaning: The system attempted to assign a datalink to a pool that already is assigned to a datalink by the LNKUTIL directory devcon command.
	Action: Use the datalink that already is assigned to the pool or assign that datalink to a new pool.
End	

stop

Function

Use the stop command to disable routing of management reports for ACD groups assigned to a specified pool.

stop command parameters and variables	
Command	**Parameters and variables**
stop	all *groupname*

Parameters and variables	**Description**
all	This parameter disables all ACD groups routed to the pool specified by the devcon command in the LNKUTIL directory.
groupname	This variable specifies the name or names of the ACD group routed to a valid pool. If you enter more than one ACD group, separate each ACD group name from the next by a single space.

Qualifications

The stop command is qualified by the following exceptions, restrictions, and limitations:

- When the datalinks transfer, related operational measurement (OM) data displays with appropriate register peggings.
- The OM registers are incremented depending on the number of messages unprocessed prior to routing cancellation.

stop (continued)

Examples

The following table provides examples of the stop command.

Examples of the stop command		
Example	**Task, response, and explanation**	
stop all ↵		
	Task:	Prevent routing for all ACD groups in the same pool.
	Response:	ROUTING INFORMATION FOR ALL ACD GROUPS HAS BEEN DELETED.
	Explanation:	This command disables routing for all ACD groups.
stop groupname1 ↵ *where* groupname1 specifies the ACD group that no longer is routed to the specified pool		
	Task:	Prevent routing for one ACD group.
	Response:	ACD GROUP GROUPNAME1 HAS BEEN REMOVED FROM POOL POOLNAME1.
	Explanation:	This command removed the ACD group named groupname1 from the pool named poolname1.
-continued-		

stop (continued)

Examples of the stop command (continued)

Example	Task, response, and explanation

stop groupname1 groupname2 groupname3 ↵
where

groupname1	specifies one of three ACD groups that no longer are routed to the specified pool
groupname2	specifies one of three ACD groups that no longer are routed to the specified pool
groupname3	specifies one of three ACD groups that no longer are routed to the specified pool

Task:		Prevent routing for multiple ACD groups in the same pool.
Response:		ACD GROUP GROUPNAME1 HAS BEEN REMOVED FROM POOL POOLNAME1. ACD GROUP GROUPNAME2 HAS BEEN REMOVED FROM POOL POOLNAME1. ACD GROUP GROUPNAME3 HAS BEEN REMOVED FROM POOL POOLNAME1.
Explanation:		This command removed ACD groups named groupname1, groupname2, and groupname3 from the pool named poolname1.

End

Responses

The following table provides explanations of the responses to the stop command.

Responses for the stop command

MAP output	Meaning and action

```
ACD GROUP GROUPNAME1 HAS BEEN REMOVED FROM POOL POOLNAME1.
ACD GROUP GROUPNAME1 HAS BEEN REMOVED FROM POOL POOLNAME2.
ACD GROUP GROUPNAME1 HAS BEEN REMOVED FROM POOL POOLNAME3.
GROUP ABCDEF DOES NOT EXIST.
REMAINING GROUPS WILL BE IGNORED.
```

Meaning:	Multiple ACD groups were entered. Although one of the specified ACD groups does not exist, the command executed successfully.
Action:	Verify the name of the pool using the ACDMR directory mrstat command. Verify the name of the group in Table ACDGRP.

-continued-

stop (end)

Responses for the stop command (continued)	
MAP output	**Meaning and action**
ACD GROUP GROUPNAME1 IS NOT ASSIGNED TO A POOL. NO ACTION TAKEN.	
	Meaning: The name of an ACD group that was entered is not assigned to a pool. The command aborted.
	Action: Verify the name of the ACD group in Table ACDGRP and assign the ACD group to a pool using the LNKUTIL directory devcon command.
AT LEAST ONE ACD GROUP MUST BE SPECIFIED. NO ACTION TAKEN.	
	Meaning: No ACD group was entered and the command aborted.
	Action: Enter a valid ACD group name from Table ACDGRP.
NO ACD GROUPS HAVE ROUTING INFORMATION SPECIFIED. NO ACTION TAKEN.	
	Meaning: Multiple ACD groups were entered and the command aborted.
	Action: Enter valid ACD group names from Table ACDGRP.
End	

ACDPOOLS level commands

Use the ACDPOOLS level of the MAP to display pool configurations and current status of Automatic Call Distribution (ACD) pools.

The ACDPOOLS directory provides commands for the ACD management information system that partition ACD groups into data streams. This allows the down stream processor (DSP) to access data and receive call event messages for only the ACD groups within the selected data stream.

Accessing the ACDPOOLS level

To access the ACDPOOLS level, enter the following command from the CI level:

acdpools ↵

ACDPOOLS commands

The commands available at the ACDPOOLS MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

ACDPOOLS commands	
Command	**Page**
acdgrps	A-79
help	A-83
pools	A-85
quit	A-91
status	A-95
subpools	A-97

acdgrps

Function

Use the acdgrps command to display the requested ACD group information.

acdgrps command parameters and variables	
Command	**Parameters and variables**
acdgrps	all
	one *acdgroup*
Parameters and variables	**Description**
acdgroup	This variable specifies the name of the ACD group for which the requested information is to be listed.
all	This parameter displays the requested information for all ACD groups.
one	This parameter displays the requested information for one ACD group.

Qualifications

None

acdgrps (continued)

Examples

The following table provides examples of the acdgrps command.

Examples of the acdgrps command	
Example	**Task, response, and explanation**
acdgrps all ↵	
	Task: List ACD group information for all ACD groups on the switch.

	Response: `ACD GROUP :    USAA1`
	`POOL: AAA        SUBPOOL:    AAA1`
	
	`ACD GROUP :    USAA2`
	`POOL: AAA        SUBPOOL:    AAA1`
	
	`ACD GROUP :    USAA3`
	`POOL: AAA        SUBPOOL:    AAA2`
	
	`ACD GROUP :    ACDLOND`
	`POOL: BBB        SUBPOOL:    BBB2`

	Explanation: This command lists information for all ACD groups on the switch.
acdgrps one usaa1 ↵ *where* usaa1 specifies the particular ACD group for which information is to be listed	
	Task: List information for a specified ACD group.
	Response: `ACD GROUP :    USAA1` `POOL: AAA        SUBPOOL:    AAA1`
	Explanation: This command lists information for the ACD group named usaa1.

Response

The following table provides an explanation of the response to the acdgrps command.

acdgrps (end)

Response for the acdgrps command	
MAP output	**Meaning and action**
| ```
***  INVALID ACD GROUP NAME.
         VALID NAMES ARE:
     ACD1, ACD2, ACD3, ACD4,
     CORRECT FORMAT:    <ACD_GROUP> STRING
     ENTER:             <ACD_GROUP>
``` | |
| | **Meaning:** The specified ACD group name was entered incorrectly or an invalid group name was used. |
| | **Action:** Enter a valid ACD group name or abort the command. |

help

Function

Use the help command to receive online documentation for the ACDPOOLS directory.

help command parameters and variables	
Command	**Parameters and variables**
help	<u>*all*</u> acdpools

Parameters and variables	Description
<u>*all*</u>	Omitting this entry forces the system to default to displaying online documentation for this directory.
acdpools	This parameter produces a function summary of all ACDPOOLS directory commands.

Qualifications

None

Examples

The following table provides examples of the help command.

Examples of the help command		
Example	**Task, response, and explanation**	
help ↵		
	Task:	Access online documentation.
	Response:	```The available commands are:```

		```POOL -- Displays ACD pool information```
		```SUBPOOLS -- Displays ACD subpool information```
		```ACDGRPS -- Displays ACD group pool information```
		```STATUS -- Displays ACD pool status```
		``` information```
		```QUIT -- Quits from ACDPOOLS environment```
		```HELP -- Displays available commands```

	Explanation:	This example typifies a response for the help command string.
	-continued-	

help (end)

Examples of the help command (continued)

Example   **Task, response, and explanation**

---

help   acdpools ↵
where

acdpools   specifies the help query

Task:	Access online documentation.	

Response: The available commands are:

```
          POOL     -- Displays ACD pool information
          SUBPOOLS -- Displays ACD subpool information
          ACDGRPS  -- Displays ACD group pool information
          STATUS   -- Displays ACD pool status
                       information
          QUIT     -- Quits from ACDPOOLS environment
          HELP     -- Displays available commands
```

Explanation: This example typifies a response for the help command string.

End

Response

The following table provides an explanation of the response to the help command.

Response for the help command

MAP output   **Meaning and action**

MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.

Meaning: The directory you are trying to access is not loaded or must be accessed
through another directory.

Action: None

---

pools

Function

Use the pools command to display the requested ACD pool information.

pools command parameters and variables	
Command	**Parameters and variables**

pools all *nn* ⎡ n ⎤ ⎡ *n* ⎤
                      ⎢ y ⎥ ⎢ y ⎥
                      ⎣ *subpool* ⎦

        one *acdpool* all ⎡ one *subpool* ⎤

Parameters and variables	Description
acdpool	This variable specifies the name of the ACD pool for which information is to be listed.
all	This parameter indicates that the requested information is to be listed for all ACD pools, subpools, or groups.
n	This parameter indicates that ACD subpools are not to be included in the display.
n	Omitting this entry forces the system to default to not displaying ACD groups.
nn	Omitting this entry forces the system to default to not displaying ACD groups or subpools.
one	This parameter indicates that the requested information is to be listed for a specific subpool or group.
subpool	This variable specifies the name of the ACD subpool for which information is to be listed.
y	This parameter indicates that ACD groups or ACD subpools are to be included in the display.

Qualifications

None

pools (continued)

Examples

The following table provides examples of the pools command.

Examples of the pools command	
Example	**Task, response, and explanation**
pools all ↵	
	Task: List all pools.
	Response: POOLS : AAA,BBB,CCC
	Explanation: This command displays all pools on the switch. The system defaults to no subpools or ACD groups in the display.
pools all y ↵	
	Task: List all pools and subpools on the switch.

	Response: POOL : AAA
	    SUBPOOL: AAA1, AAA2, AAA3, AAA4
	POOL : BBB
	    SUBPOOL: BBB1, BBB2
	POOL : CCC
	    SUBPOOL: NO SUBPOOLS HAVE BEEN ASSIGNED

	Explanation: This command displays all of the pools and subpools on the switch, but the system defaults to not displaying ACD groups.
pools all y y ↵	
	Task: List all pools, subpools, and groups on the switch.

	Response: POOL : AAA1
	    ACD GROUPS: ACD1, ACD2
	      SUBPOOL: AAA2
	    ACD GROUPS: ACD3
	      SUBPOOL: AAA3
	    ACD GROUPS: NONE
	      SUBPOOL: AAA4
	    ACD GROUPS: NONE

	Explanation: This command displays all pools, subpools, and ACD groups on the switch.
-continued-	

pools (continued)

Examples of the pools command (continued)

Example	Task, response, and explanation

pools all n y ↵

	Task:	List all pools and ACD groups on the switch.
| | **Response:** | ```
POOL :    AAA
   ACD GROUPS:  ACD1, ACD2, ACD3
POOL :    BBB
   ACD GROUPS:  ACD4
POOL :    CCC
   ACD GROUPS:  NONE
``` |
| | **Explanation:** | This command displays all pools and ACD groups on the switch, but does not display subpools. |

pools one ccc all ↵
where

ccc   specifies the name of the pool

	Task:	List status of all ACD subpools for a specified pool.
| | **Response:** | ```
POOL:   CCC
   SUBPOOL:   NO SUBPOOLS HAVE BEEN ASSIGNED
``` |
| | **Explanation:** | This command lists the status of subpools for the pool named ccc. The system defaults to no ACD groups in the display. |

pools one bbb all y ↵
where

bbb   specifies the name of the pool

	Task:	List status of all subpools and ACD groups for a specified pool.
| | **Response:** | ```
POOL:   BBB
   SUBPOOL:   BBB1
      ACD GROUPS:   NONE
   SUBPOOL:   BBB2
      ACD GROUPS:  ACD4
``` |
| | **Explanation:** | This command lists the status of subpools for the pool named bbb and the ACD groups associated with each subpool. |

-continued-

pools (continued)

Examples of the pools command (continued)	
Example	**Task, response, and explanation**

pools **one** **bbb** **one** **bbb1** ↵ *where* bbb specifies the name of the pool bbb1 specifies the name of the subpool for which information is to be listed	

Task:	List the status of a specified subpool for a specified pool.
Response:	POOL: BBB SUBPOOL: BBB1
Explanation:	This command lists the status of the subpool named bbb1 for the pool named bbb. The system defaults to no ACD groups in the display.

pools **one** **aaa** **one** **aaa1** **y** ↵ *where* aaa specifies the name of the pool aaa1 specifies the name of the subpool for which information is to be listed	

Task:	List the status of ACD groups for a specified subpool.

Response:	POOL: AAA
	     SUBPOOL: AAA1
	         ACD GROUPS: NONE

Explanation:	This command lists the status of ACD groups associated with the subpool named aaa1 for the pool named aaa.

End

Response

The following table provides an explanation of the response to the pools command.

pools (end)

Response for the pools command

MAP output   Meaning and action

```
***   INVALID ACD MIS POOL NAME:
            VALID NAMES ARE:
AAA,BBB,CCC,POOL,
CORRECT FORMAT:    <ACD_POOL> STRING
ENTER:     <ACD_POOL> <SUBPOOLS?>
```

Meaning: This response indicates that the specified pool is not in the system.

Action: Enter a valid pool name.

quit

Function

Use the quit command to exit the ACDPOOLS directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	$\begin{bmatrix} \underline{\textit{1 level}} \\ \text{all} \\ \textit{name} \\ \textit{n\_levels} \end{bmatrix}$
Parameters and variables	**Description**
<u>*1 level*</u>	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
CI:	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
QUIT -- Increment not found	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
QUIT -- Unable to quit requested number of levels	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

status

Function

Use the status command to display status information for one or all ACD pools. The status of ACD pools is reported as initializing, transferring, init and transfer, or wait.

status command parameters and variables	
Command	**Parameters and variables**
status	all one *acdpool*

Parameters and variables	Description
acdpool	This variable specifies the name of the ACD pool selected for a status display.
all	This parameter lists status information for all ACD pools.
one	This parameter lists status information for a specified ACD pool.

Qualifications

None

Examples

The following table provides examples of the status command.

Examples of the status command		
Example	**Task, response, and explanation**	
status all ↵		
	Task:	List status for all ACD pools.
	Response:	```POOL STATE THROTTLE THROTTLE STATUS
AAA   INITIALIZING N/A   N/A
BBB   NOT ASSOCIATED WITH DSP
CCC   TRANSFERRING N/A   N/A```
	Explanation:	This command lists the status of all ACD pools.
	-continued-	

status (end)

Examples of the status command (continued)	
Example	**Task, response, and explanation**
status one aaa ↵ *where* aaa specifies the name of the ACD pool	
	Task: List status for a specified ACD pool.
	Response: `POOL        STATE        THROTTLE   THROTTLE STATUS` `AAA         INITIALIZING N/A           N/A`
	Explanation: This command queries the status of the ACD pool named aaa.
End	

Response

The following table provides an explanation for the response to the status command.

Response for the status command	
MAP output	**Meaning and action**
`***  INVALID SYMBOL: <POOLS?> (ALL, ONE, <ACD_POOL> STRING)`	
	Meaning: This message indicates that incorrect command syntax was used.
	Action: Enter the command correctly or enter abort.

subpools

Function

Use the subpools command to display the requested ACD subpool information.

subpools command parameters and variables			
Command	**Parameters and variables**		
subpools	all	$\begin{bmatrix} \underline{n} \\ y \end{bmatrix}$	
	one	*subpool*	$\begin{bmatrix} \underline{n} \\ y \end{bmatrix}$

Parameters and variables	Description
n	Omitting this entry forces the system to default to not displaying ACD groups.
all	This parameter lists the requested information for all ACD subpools.
one	This parameter indicates that the requested information is to be listed for a specific subpool.
subpool	This variable specifies the name of the ACD subpool for which information is to be listed.
y	This parameter displays ACD groups.

Qualifications

None

Examples

The following table provides examples of the subpools command.

subpools (continued)

Examples of the subpools command	
Example	**Task, response, and explanation**
subpools all ↵	
	Task: List all the subpools on the switch.

	Response:
	`POOL:    AAA`
	`     SUBPOOL:  AAA1`
	`POOL:    BBB`
	`     SUBPOOL:  BBB1`
	`POOL:    CCC`
	`     SUBPOOL:  CCC1`

	Explanation: This command displays the requested information for all ACD subpools. Since the command entry does not specify that ACD groups are to be included in the display, the system defaults to showing no ACD groups in the display.
subpools all y ↵	
	Task: List all the subpools and ACD groups on the switch.

	Response:
	`POOL:    AAA`
	`     SUBPOOL:  AAA1`
	`       ACD GROUPS:  ACD1, ACD2`
	`POOL:    AAA`
	`     SUBPOOL:  AAA2`
	`       ACD GROUPS:  ACD3`
	`POOL:    AAA`
	`     SUBPOOL:  AAA3`
	`       ACD GROUPS:  ACD1, ACD2`
	`POOL:    BBB`
	`     SUBPOOL:  BBB1`
	`       ACD GROUPS: NONE`
	`POOL:    CCC`
	`     SUBPOOL:  CCC1`
	`       ACD GROUPS:  NONE`

	Explanation: This command displays the requested information for all ACD subpools and groups.
-continued-	

subpools (continued)

Examples of the subpools command (continued)

Example   **Task, response, and explanation**

---

subpools one ccc4 ↵
where

ccc4   specifies the name of the subpool

Task:	List information on a specified subpool.
Response:	POOL: CCC SUBPOOL: CCC4
Explanation:	This command displays the requested information for the subpool named ccc4. Since the command entry did not specify that ACD groups are to be included in the display, the system defaults to showing no ACD groups in the display.

---

subpools one ccc4 y ↵
where

ccc4   specifies the name of the subpool

Task:	List information on a specified subpool as well as ACD group information.

Response:	POOL: CCC
	      SUBPOOL: CCC4
	        ACD GROUPS: NONE

Explanation:	This command displays the requested information for the subpool named ccc4 and associated ACD groups.

End

Response

The following table provides an explanation for the response to the subpools command.

subpools (end)

Response for the subpools command	
MAP output	**Meaning and action**

```*** INVALID ACD MIS SUBPOOL NAME.```
```   VALID NAMES ARE:```
```ABCD, BBB1,```
```CORRECT FORMAT: <ACD_SBPL> STRING```
```ENTER: <ACD_SBPL>```	

	Meaning: This message indicates that the specified subpool is not in the system. **Action:** Enter the correct subpool name and reissue the command.

ACDRTDIS level commands

Use the ACDRTDIS level of the MAP to obtain a simple management report for Automatic Call Distribution (ACD) groups. Statistics for the specified ACD groups are gathered and displayed at selected time intervals.

Accessing the ACDRTDIS level

To access the ACDRTDIS level, enter the following command from the CI level:

> **acdrtdis** ↵

ACDRTDIS commands

The commands available at the ACDRTDIS MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

ACDRTDIS commands	
Command	**Page**
help	A-103
quit	A-105
rtdstat	A-109
send	A-113
startrpt	A-115
stop	A-119
stoprpt	A-121

help

Function

Use the help command to receive online documentation for the ACDRTDIS directory.

help command parameters and variables	
Command	**Parameters and variables**
help	acdrtdis *command_nam*
Parameters and variables	**Description**
acdrtdis	This parameter produces summary documentation for the commands in the ACDRTDIS directory.
command_nam	This variable specifies a valid ACDRTDIS directory command name. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help startrpt ↵ *where*	
startrpt	specifies a valid command for the ACDRTDIS directory
	Task: Access online documentation.

	Response: `Start Real Time Display at the specified time`
	`interval in seconds for a specifed pool.`
	`Parms:  <pool> STRING`
	`[<Interval> {1 to 255}]`

	Explanation: This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

quit

Function

Use the quit command to exit the ACDRTDIS directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	$\begin{bmatrix} \textit{1 level} \\ \text{all} \\ \textit{name} \\ \textit{n\_levels} \end{bmatrix}$

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut	specifies a directory	
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
	End	

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

rtdstat

Function

Use the rtdstat command to query a particular pool, query one or more ACD groups, or query the time interval at which reports are generated for each specified pool.

rtdstat command parameters and variables		
Command	**Parameters and variables**	
rtdstat	all	
	group	*group*
	interval	all
		pool *pool*
	pool	*datalink*

Parameters and variables	Description
all	This parameter displays information for all ACD group pools.
datalink	This variable specifies the name of the datalink pool of ACD groups for which the information is requested.
group	This parameter displays ACD group information.
group	This variable specifies the name of one or more ACD groups for which the information is requested.
interval	This parameter displays the report generation time interval for one or all pools.
pool	This parameter displays ACD pool information.
pool	This variable specifies the name of the pool for which report generation time interval information is requested.

Qualifications

None

Examples

The following table provides examples of the rtdstat command.

rtdstat (continued)

Examples of the rtdstat command	
Example	**Task, response, and explanation**
rtdstat group group1 ↵ *where* group1 specifies the name of the ACD group for which the information is requested	
	Task: Display the status of a specified ACD group.
	Response: `ACD GROUP GROUP1 IS ROUTED TO POOL:  PNAME B`
	Explanation: This command displays the routing status of the ACD group named group1.
rtdstat interval pnamea ↵ *where* pnamea specifies the name of the pool	
	Task: Display the report generation time interval for a pool.
	Response: `POOL PNAMEA HAS A TRANSMISSION INTERVAL OF 20` `SECONDS.`
	Explanation: This command displays the report generation time interval for the pool named pnamea.

Responses

The following table provides explanations of the responses to the rtdstat command.

Responses for the rtdstat command	
MAP output	**Meaning and action**
`ACD GROUP GROUP1 IS ROUTED TO POOL:  PNAMEA`	
	Meaning: This message displays the status of the ACD group named group1.
	Action: None
-continued-	

Responses for the rtdstat command (continued)

MAP output   Meaning and action

```
THE FOLLOWING ACD GROUPS HAVE BEEN ROUTED TO POOL PNAMEA WITH A
TRANSMISSION INTERVAL OF 20 SECONDS:
GROUP0, GROUP2, GROUP9
```

Meaning: This message provides report generation information for the pool named pnamea.

Action: None

End

send

Function

Use the send command to route ACD real time display data for particular ACD groups to the specified datalink pool.

send command parameters and variables		
Command	**Parameters and variables**	
send	*pool*	⎡ all ⎤ ⎣ *group* ⎦

Parameters and variables	**Description**
all	This parameter routes all ACD groups to the pool.
group	This variable specifies one or more ACD groups to be routed to the pool.
pool	This variable specifies the name of the the datalink pool to which the ACD group or groups are to be routed.

Qualification

To route the ACD group data to the specified pool, the pool must be known to the system.

Examples

The following table provides examples of the send command.

Examples of the send command	
Example	**Task, response, and explanation**
send pname group1 ↵ *where* pname specifies the datalink pool where the ACD group will be routed group1 specifies the ACD group whose data is to be routed to the pool	

	Task:	Route report data for a specified ACD group to a specified pool.
	Response:	`ACD GROUP GROUP1 HAS BEEN ASSIGNED TO POOL PNAME.` `ROUTE LIST HAS BEEN UPDATED FOR POOL PNAME.`
	Explanation:	This command routes ACD group data for group1 to the pool named pname for report display.

-continued-

send (end)

Examples of the send command (continued)	
Example	**Task, response, and explanation**

send pname group1 group5 ↵
where

pname	specifies the datalink pool where the ACD group will be routed
group1	specifies one of two ACD groups whose data is to be routed to the pool
group5	specifies one of two ACD groups whose data is to be routed to the pool

Task:	Route report data for more than one ACD group to a specified pool.
Response:	`ACD GROUP GROUP1 HAS BEEN ASSIGNED TO POOL PNAME.` `ACD GROUP GROUP5 HAS BEEN ASSIGNED TO POOL PNAME.` `ROUTE LIST HAS BEEN UPDATED FOR POOL PNAME.`
Explanation:	This command routes ACD group data for group1 and group5 to the pool named pname for report display.

End

Responses

The following table provides explanations of the responses to the send command.

Responses for the send command	
MAP output	**Meaning and action**

`GROUP GROUP3 DOES NOT EXIST.  REMAINING GROUPS WILL BE IGNORED.`	
Meaning:	When the nonexistent ACD group is entered for routing, any groups subsequently entered for routing are ignored.
Action:	Reissue the command with a valid ACD group.
`TWO DATALINKS ARE CURRENTLY ASSIGNED TO POOL PNAME.  ONLY ONE DATALINK MAY BE ASSIGNED TO A POOL USED FOR THE ACD REAL TIME DISPLAY.`	
Meaning:	Duplicate datalinks were entered for the pool named pname. Only one datalink is allowed for each pool.
Action:	Ensure that only one datalink is assigned for each pool.

startrpt

Function

Use the startrpt command to start the ACD real time report display for a specified pool.

startrpt command parameters and variables		
Command	**Parameters and variables**	
startrpt	*pool*	$\begin{array}{l} \underline{255} \\ interval \end{array}$

Parameters and variables	Description
255	Omitting this entry forces the system to default to an interval frequency value of 255.
pool	This variable specifies the name of the pool on which the ACD report generation is to start.
interval	This variable specifies the interval frequency for report generation. The valid entry range is 1-255 seconds.

Qualifications

The startrpt command is qualified by the following exceptions, restrictions, and limitations:

- To route the ACD group data to the specified pool, the pool must be known to the system.

- The pool designated as the starting pool for report generation must be assigned to a datalink or you cannot generate the report.

Examples

The following table provides examples of the startrpt command.

startrpt (continued)

Examples of the startrpt command	
Example	**Task, response, and explanation**
startrpt pname ↵ *where* pname specifies the name of the pool	
	Task: Start a report on a specified pool.
	Response: ACD REAL TIME DISPLAY HAS BEEN STARTED ON POOL PNAME.
	Explanation: This command displays an ACD report for the pool named pname. Since no interval frequency is specified, the system defaults to an interval of 255 seconds.
startrpt pname 30 ↵ *where* pname specifies the name of the pool 30 specifies a 30-second time interval frequency for the generation of the report	
	Task: Start a report on a specified pool for a particular time interval.
	Response: ACD REAL TIME DISPLAY HAS BEEN STARTED ON POOL PNAME.
	Explanation: This command displays an ACD report for the pool named pname with a 30-second time interval specified.

Responses

The following table provides explanations of the responses to the startrpt command.

Responses for the startrpt command	
MAP output	**Meaning and action**
ACD REAL TIME DISPLAY HAS BEEN STARTED ON POOL PNAME.	
	Meaning: The ACD real time display report on the pool named pname has been started.
	Action: None
-continued-	

startrpt (end)

Responses for the startrpt command (continued)

MAP output   **Meaning and action**

```
NO ACD GROUPS ARE ROUTED TO POOL PNAME.

NO ACTION TAKEN.
```

Meaning: If there are no ACD groups routed to the pool named pname, the report cannot be started on that pool.

Action: Ensure that starting pool has ACD groups routed to it.

```
NO DATALINKS ARE ASSIGNED TO POOL PNAME.
```

Meaning: The report cannot start on this pool because the pool named pname has not been assigned to a datalink.

Action: Before generating reports, assign the pool to a datalink.

End

stop

Function

Use the stop command to disable routing ACD real time display data to the specified pool.

stop command parameters and variables		
Command	**Parameters and variables**	
stop	*pool*	$\begin{bmatrix} \text{all} \\ \textit{group} \end{bmatrix}$

Parameters and variables	Description
all	This parameter removes all ACD groups from the pool.
group	This variable specifies one or more ACD groups to be removed from the pool.
pool	This variable specifies the name of the datalink pool from which the group data is to be removed.

Qualifications

None

Examples

The following table provides examples of the stop command.

Examples of the stop command	
Example	**Task, response, and explanation**
stop pname group1 ↵ *where* pname specifies the pool group1 specifies the name of the ACD group to be removed from the pool	

	Task:	Remove a group from a pool.
	Response:	`ACD GROUP GROUP1 HAS BEEN REMOVED FROM POOL PNAME`
	Explanation:	This command removes the ACD group named group1 from the routing list for the pool named pname and prevents the data for this ACD group from appearing on an ACD report for the specified pool.

-continued-

stop (end)

Examples of the stop command (continued)	
Example	**Task, response, and explanation**
stop pname all ↵ *where* pname specifies the pool	
	Task: Remove all ACD groups from pool pname.
	Response: `ALL ACD GROUPS HAVE BEEN REMOVED FROM POOL PNAME.`
	Explanation: This command prevents data for all ACD groups from appearing on an ACD report for the pool named pname.
End	

Response

The following table provides an explanation of the response to the stop command.

Response for the stop command	
MAP output	**Meaning and action**
`GROUP GROUP9 DOES NOT EXIST.` `REMAINING GROUPS WILL BE IGNORED.`	
	Meaning: Multiple ACD groups were entered and a specific group does not exist.
	Action: Reissue the command with a valid ACD group name or datafill the specified ACD group in Table ACDGRP.

stoprpt

Function

Use the stoprpt command to stop the generation of an ACD real time display report for a specified pool.

stoprpt command parameters and variables	
Command	**Parameters and variables**
stoprpt	*pool*
Parameters and variables	**Description**
pool	This variable specifies the name of the pool on which the ACD report generation is to be stopped.

Qualifications

None

Example

The following table provides an example of the stoprpt command.

Example of the stoprpt command		
Example	**Task, response, and explanation**	
stoprpt pname ↵ *where* pname specifies the pool		
	Task:	Stop the report on a specified pool.
	Response:	ACD REAL TIME DISPLAY HAS BEEN STOPPED FOR POOL.
	Explanation:	This command stops the generation of an ACD report for pool pname.

stoprpt (end)

Responses

The following table provides explanations of the responses to the stoprpt command.

Responses for the stoprpt command	
MAP output	**Meaning and action**
ACD REAL TIME DISPLAY HAS BEEN STOPPED FOR POOL PNAME.	
	Meaning: The generation of the ACD real time display report has been stopped for the pool named pname.
	Action: None
ACD REAL TIME DISPLAY HAS NOT BEEN STARTED FOR POOL PNAME.	
	Meaning: The real time display report was not started for the pool named pname at the time the stoprpt command was entered.
	Action: None

ACDSHOW level commands

Use the ACDSHOW level of the MAP to display the following information about the current configuration of Automatic Call Distribution (ACD) groups and subgroups:

- administration groups
- agent positions
- directory numbers
- queue sizes
- supervisors
- routing information
- audio information
- logon IDs and passwords

You can use the ACDSHOW directory commands to display information before using the LOADMGMT commands to tailor the ACD configurations. You can also verify the execution of LOADMGMT commands.

Accessing the ACDSHOW level

To access the ACDSHOW level, enter the following command from the CI level:

acdshow ↵

Upon entering the ACDSHOW directory, you are automatically placed in brief mode. Commands display agent information only.

ACDSHOW commands

The commands available at the ACDSHOW MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

ACDSHOW commands	
Command	**Page**
acddns	A-127
admingroup	A-131
agtpos	A-137
audiogroup	A-145
clrroute	A-147
counts	A-149
fiaudgrp	A-151
foaudgrp	A-155
groupinfo	A-159
groupname	A-169
help	A-173
loadmgmt	A-177
loginid	A-179
mode	A-185
nsaudgrp	A-187
nsroute	A-189
ovflroute	A-191
password	A-193
q	A-197
quit	A-199
status	A-203
supervisor	A-207
tabentry	A-215
threshold	A-219
-continued-	

ACDSHOW commands (continued)	
Command	**Page**
throute	A-223
validaudio	A-225
validroutes	A-229
End	

acddns

Function

Use the acddns command to display a list of Automatic Call Distribution directory numbers (ACDDN) and their priorities for ACD groups. The ACDDISP option must be datafilled in Table ACDGRP to display the ACDDN name. Each ACD group can have one primary ACDDN and up to 16 supplementary ACDDNs.

acddns command parameters and variables		
Command	**Parameters and variables**	
acddns	all	
	group	*acdgroup*
Parameters and variables	**Description**	
acdgroup	This variable specifies the ACD group.	
all	This parameter displays all the ACDDNs according to the ACD group where they are assigned.	
group	This parameter displays the list of directory numbers for a particular ACD group.	

Qualifications

None

Examples

The following table provides examples of the acddns command.

acddns (continued)

```
┌─────────────────────────────────────────────────────────────────────────┐
│ Examples of the acddns command                                          │
│                                                                         │
│ Example       Task, response, and explanation                          │
├─────────────────────────────────────────────────────────────────────────┤
│ acddns  all  ↵                                                          │
│                                                                         │
├─────────────────────────────────────────────────────────────────────────┤
│              Task:          Display all ACDDNs.                          │
│                                                                         │
│              Response:      ACD Directory Numbers For ACD Group NTTACD1 │
│                                                                         │
│                             Primary ACDDN:              003 001 1100    │
│                               Trunk Priority:           0               │
│                               Line Priority:            0               │
│                             =========================================== │
│                                                                         │
│                             ACD Directory Numbers For ACD Group NTTACD2 │
│                                                                         │
│                             Primary ACDDN:              003 001 1200    │
│                               Trunk Priority:           0               │
│                               Line Priority:            0               │
│                             =========================================== │
│                                                                         │
│                             ACD Directory Numbers For ACD Group GOTACD1 │
│                                                                         │
│                             Primary ACDDN:              003 001 2100    │
│                               Trunk Priority:           0               │
│                               Line Priority:            0               │
│                             =========================================== │
│                                                                         │
│                             ACD Directory Numbers For ACD Group GOTACD2 │
│                                                                         │
│                             Primary ACDDN:              003 001 2200    │
│                               Trunk Priority:           0               │
│                               Line Priority:            0               │
│                                                                         │
│              Explanation:   This command displays all of the ACD groups.│
├─────────────────────────────────────────────────────────────────────────┤
│                             -continued-                                  │
└─────────────────────────────────────────────────────────────────────────┘
```

acddns (continued)

Examples of the acddns command (continued)

Example	**Task, response, and explanation**

acddns group plan1 ↵
where

plan 1   specifies the ACD group name

Task:	Display the directory numbers associated with an ACD group.
| **Response:** | ```
ACD Directory Numbers for ACD Group PLAN1
    Primary ACDDN:              613 722 4449
    ACDDN Name:                 ACDGROUP_1
      Trunk Priority:           0
      Line Priority:            0

    Supplementary ACDDN:        613 722 4447
    ACDDN Name:                 SUPP_ACDGRP_1
      DN Priority:              1
    Supplementary ACDDN:        613 722 1234
    *** No Name Associated ***
      DN Priority:              1
``` |
| **Explanation:** | This command displays the directory numbers in the plan1 ACD group. The ACDDISP option is datafilled in Table ACDGRP. |

acddns group plan2 ↵
where

plan 2   specifies the ACD group name

Task:	Display the directory numbers associated with an ACD group.
| **Response:** | ```
ACD Directory Numbers for ACD Group PLAN2
    Primary ACDDN:              613 734 4456
      Trunk Priority:           0
      Line Priority:            0

    Supplementary ACDDN:        613 734 4454
      DN Priority:              1
    Supplementary ACDDN:        613 734 1252
      DN Priority:              1
``` |
| **Explanation:** | This command displays the directory numbers in the plan2 ACD group. The ACDDISP option is not datafilled in Table ACDGRP. The ACDDN name field does not appear in the display. |

End

acddns (end)

Responses

The following table provides explanations of the responses to the acddns command.

Responses for the acddns command	
MAP output	**Meaning and action**
```*** Invalid ACD group name. Valid names are: <acd_grp> <acd_grp> <acd_grp>```	
	Meaning: You entered an invalid group name.
	Action: Enter one of the group names listed to continue or abort to cancel.
```Invalid symbol: <Which_Groups?> {ALL, GROUP <Acd_grp?> STRING} Enter: <Which_Groups?>```	
	Meaning: You entered the command without the required parameter.
	Action: Enter the parameter desired to continue or abort to cancel.

admingroup

Function

Use the admingroup command to display the Automatic Call Distribution (ACD) groups in one or more administration groups and the senior supervisors associated with each administration group.

admingroup command parameters and variables	
Command	**Parameters and variables**
admingroup	acdgroup ⌈ all ⌊ group *acdgroup* ⌋ admingroup *admingroup* adminsup *adminsup* all myadmingroup

Parameters and variables	**Description**
acdgroup	This parameter displays the administration groups and associated senior supervisors for a particular ACD group.
acdgroup	This variable identifies the ACD group.
admingroup	This parameter displays information for an ACD administration group.
admingroup	This variable identifies the administration group. The valid entry range is 1-255.
adminsup	This parameter displays information for the supervisor of an administration group.
adminsup	This variable identifies the supervisor.
all	This parameter displays the administration groups and associated senior supervisors for all the ACD groups when used with only the command. This parameter displays the administration group and associated senior supervisors for a specific ACD group when used with the command and the acdgroup parameter.
group	This parameter displays information for an ACD group.
myadmingroup	This parameter displays information about your ACD groups.

Qualifications

None

admingroup (continued)

Examples

The following table provides examples of the admingroup command.

Examples of the admingroup command	
Example	**Task, response, and explanation**
admingroup adminsup aaaa ↵ *where* aaaa specifies supervisor assigned to acdgroups	
	Task: Display information for a supervisor.
	Response: ```Admingrp: 2 Adminsup:aaaa``` ```Acdgroups: PLAN2, PLAN3```
	Explanation: This command displays the administration group and the ACD groups assigned to senior supervisor aaaa.
admingroup admingroup 2 ↵ *where* 2 specifies the admingroup	
	Task: Display information for an administration group.
	Response: ```Admingrp: 2 Adminsup: aaaa``` ```Acdgroups: ACD2, ACD3```
	Explanation: This command displays the ACD groups and the senior supervisor associated with administration group 2.
admingroup myadmingroup ↵	
	Task: Display information for your groups.
	Response: ```Admingrp: 2 Adminsup: AAAA``` ```Acdgroups: ACD2, ACD3```
	Explanation: This command displays the administration group and the ACD groups associated with you.
-continued-	

admingroup (continued)

Examples of the admingroup command (continued)

Example   Task, response, and explanation

---

admingroup all ↵

Task:	Display all the administration groups.	

Response:
```
Admingrp:  2              Adminsup:   AAAA
Acdgroups:  ACD2, ACD3
Admingrp:  3              Adminsup:   BBBB
Acdgroups:  ACD5
Admingrp:  4              Adminsup:   CCCC
Acdgroups:  ACD6, ACD7
```

Explanation: This command displays all of the administration groups, their associated senior supervisors, and ACD groups.

---

admingroup acdgroup group plan1 ↵
where

plan1   specifies the group name

Task: Display the administration groups and the ACD group.

Response:
```
Acd group:  PLAN1
Admingrp:  2              Adminsup:   AAAA
```

Explanation: This command displays the administration group and the ACD group plan1.

-continued-

---

admingroup (continued)

Examples of the admingroup command (continued)	
Example	**Task, response, and explanation**
admingroup acdgroup all ↵	
	Task: Display all the ACD groups.
| | **Response:**<br>```
Acd group: ACD2
Admingrp:  2              Adminsup:  AAAA
Acd group: ACD3
Admingrp:  2              Adminsup:  AAAA
Acd group: ACD4
Admingrp:  4              Adminsup:  CCCC
Acd group: ACD5
Admingrp:  3              Adminsup:  BBBB
Acd group: ACD6
Admingrp:  4              Adminsup:  CCCC
Acd group: ACD7
Admingrp:  4              Adminsup:  CCCC
``` |
| | **Explanation:** This command displays the administration groups and the senior supervisors associated with all the ACD groups. |
| **End** | |

Responses

The following table provides explanations of the responses to the admingroup command.

Responses for the admingroup command	
MAP output	**Meaning and action**
| ```
Admingrp:  admingrp    Adminsup:  adminsup
Acdgrp: acdgrps
``` | |
	Meaning: The system displayed the ACD group and senior supervisor for the adminstration group specified in the command. The admingroup admingroup *admingroup* command string and the admingroup myadmingroup command string generate the same display.
	Action: None
-continued-	

admingroup (continued)

Responses for the admingroup command (continued)	
MAP output	**Meaning and action**
`Acdgrp: acdgrp` `Admingrp:  admingrp     Adminsup:  adminsup`	
	Meaning: The system displayed the administration group and senior supervisor for the ACD group specified in the command. **Action:** None

`*** Invalid ACD group name.`
`    Valid names are:`
`<Acd_grp> <Acd_grp> <Acd_grp> <Acd_grp>`	

	Meaning: You entered an invalid ACD group name. **Action:** Enter one of the group names listed to continue or abort to cancel.

`Invalid symbol: <How?> {ALL,`
`                    MYADMINGROUP,`
`                    ADMINGROUP <Acd_admngr?> {1 to 255},`
`                    ACDGROUP <Groups?> {ALL,`
`                                        GROUP <Acd_grp?> STRING},`
`                    ADMINSUP <Acd_admnsp?> STRING}`
`Enter: <How?>`	

	Meaning: You entered the command with an invalid parameter. **Action:** Check the spelling and enter the parameter to continue or abort to cancel.
`Out of range: <Acd_admngr?> {1 to 255}` `Enter: <Acd_admngr?>`	
	Meaning: You entered a value lower than one or higher than 255. **Action:** Enter the correct value to continue or abort to cancel.

`This Adminsup is not in table ACDADMIN`
`Parameter 1 is of wrong type.`
`<Acd_admnsp?> STRING`
`Enter: <Acd_admnsp?>`	

	Meaning: You entered a value that is not in Table ACDADMIN. **Action:** Enter a correct adminsup to continue or abort to cancel.
-continued-	

admingroup (end)

Responses for the admingroup command (continued)	
MAP output	**Meaning and action**
```This Adminsup is not in table ACDADMIN Parameter 2 is of wrong type. <Acd_admnsp?> STRING Enter: <Acd_admnsp?>```	
	Meaning: You entered a numeric value for an alphabetic field. **Action:** Enter an alphabetic adminsup to continue or abort to cancel.
```Wrong type: <Acd_admngr?> {1 to 255} Enter: <Acd_admngr?>```	
	Meaning: You entered alphabetic characters for a numeric field. **Action:** Enter the correct number to continue or abort to cancel.
End	

agtpos

Function

Use the agtpos command to display the following information on Automatic Call Distribution (ACD) agent positions:

- all agents for all subgroups of an ACD group
- all agents for one subgroup number, across all ACD groups
- all agents for one subgroup of an ACD group
- all agents, listed by ACD group
- the line equipment number (LEN) and directory number (DN) of an agent, given the position ID
- if the position is in use
- login ID of agent logged in at the position
- an agent position, given its LEN
- an agent position, given its DN

agtpos command parameters and variables		
Command	**Parameters and variables**	

agtpos	all ⌈ all ⌉
	        ⌊ subgroup *subgrp* ⌋
	 clli *clli* *num*
	 dn *dn*
	 group *acdgroup* ⌈ all ⌉
	                  ⌊ subgroup *subgrp* ⌋
	 len ⌈ *site* *frame* *bay* *drawer* *card* ⌉
	 posid *posid*
		brief
		
		 full

Parameters and variables	Description
acdgroup	This variable identifies the ACD group.
all	This parameter displays information about all the agent positions in an ACD group.
all all	This parameter displays information about all the agent positions in all the ACD groups.
bay	This variable specifies the bay number. The valid entry range is 0-1.
-continued-	

agtpos (continued)

agtpos command parameters and variables (continued)	
Parameters and variables	**Description**
brief	This parameter displays the login ID of the agent and the agent position where the agent is currently logged in. Omitting this entry forces the system to default to display the current login ID and position when mode is set to brief, which is the setting when entering the ACDSHOW directory.
card	This variable specifies the card number. The valid entry range is 0-32.
clli	This parameter displays information about the CLLI in the ACD group.
clli	This variable specifies the CLLI to display.
dn	This parameter displays the list of agent positions associated with a particular DN.
dn	This variable identifies the DN. DNs are fully displayed as ten-digit numbers, which include the area code.
drawer	This variable specifies the drawer number. The valid entry range is 0-19.
frame	This variable specifies the frame number. The valid entry range is 0-2047.
full	This parameter displays the login ID of the agent, the agent position where the agent is currently logged in, the supervisor position ID, and the LEN and DN of the supervisor who is logged in. Omitting this entry forces the system to default to display full information when mode is set to full, which must be set with the mode command in the ACDSHOW directory.
group	This parameter displays information for a particular ACD group.
len	This parameter displays information for the agent position associated with a particular LEN.
num	This variable further specifies the CLLI. The valid entry range is 0-9999.
posid	This parameter displays information for the supervisor or agent associated with a particular position ID.
posid	This variable identifies the position ID. The valid entry range is 1-9999.
site	This variable specifies the site location.
-continued-	

agtpos (continued)

agtpos command parameters and variables (continued)	
Parameters and variables	**Description**
subgroup	This parameter displays information requested for a particular ACD subgroup.
subgroup	This variable identifies the ACD subgroup. The valid entry range is 0-255.
End	

Qualifications

None

Examples

The following table provides examples of the agtpos command.

Examples of the agtpos command		
Example	**Task, response, and explanation**	
agtpos dn 7226098 ↵ *where* 7226098	identifies the directory number	
	Task:	Display information about a particular agent position.
	Response:	`Agent Position ID: 6098` ` Agent Login ID: 5432`
	Explanation:	This command displays information about a particular agent position identified by the DN 7226098.
-continued-		

agtpos (continued)

Examples of the agtpos command (continued)

Example	Task, response, and explanation

agtpos all all ↵

	Task:	Display information about all the agent positions in all the ACD groups.
|  | **Response:** | ```
=================================================

Agent In ACD Group:  NTTACD1

 -------------------------------------------------
  Agent Position ID: 1121
  Agent Login ID:    1121

Table ACDLOGIN related information:
  No Password assigned.
  Customer Group: ACDDEMO

 -------------------------------------------------
  Agent Position ID: 1120
  Agent Login ID:    1120

Table ACDLOGIN related information:
  No Password assigned.
  Customer Group: ACDDEMO

Total Number of Agent Positions: 2
=================================================
   .
   .
   .
``` |
| | **Explanation:** | This command is used to display information about all the agent positions in all the ACD groups. |

-continued-

---

agtpos (continued)

Examples of the agtpos command (continued)	
Example	**Task, response, and explanation**

agtpos group plan1 all ↵
where

plan 1   identifies the ACD group

Task:	Display information on all agent positions in an ACD group.
Response:	AGENTS IN ACD GROUP: PLAN1 Agent Position ID: 5000 Agent Login ID: 1000 Agent Position ID: 3420 * *Agent not Logged in* * Agent Position ID: 5005 Agent Login ID: 7890
Explanation:	This command is used to display information on all the agent positions in ACD group plan1.

agtpos group plan1 subgroup 3 full ↵
where

plan 1   identifies the ACD group
3   identifies the ACD subgroup

Task:	Display full information for a subgroup.
Response:	Agent LEN & DN: LEN HOST 02 0 00 04 DN 7221244 Agent Position ID: 6098 Agent Group: PLAN1 ACD Subgroup: 3 Agent Login ID: 5432 Associated Supervisor Info: Supervisor LEN & DN: LEN HOST 02 0 01 09 DN 7226000 Supervisor Position ID: 6000
Explanation:	This command displays information on all the agent positions and the associated supervisor for subgroup 3 of ACD Group PLAN1.

-continued-

agtpos (continued)

Examples of the agtpos command (continued)	
Example	**Task, response, and explanation**
agtpos len 0200108 full ↵ *where* 0200108 identifies the LEN	
	Task: Display full information for a LEN.

	Response: `Agent Position ID: 6098`
	`Agent Login ID:    5432`
	` Agent LEN & DN: LEN HOST 02 0 01 08 DN 7226000`
	` ACD Group:       PLAN1`
	` ACD Subgroup:   3`

	Explanation: This command displays detailed information about the agent position serviced by the LEN 0200108.
agtpos posid 5005 ↵ *where* 5005 identifies the position ID	
	Task: Display information for a position ID.
	Response: `Agent LEN & DN: LEN HOST 02 0 01 14 DN 7227600` `Agent Login ID: 7890`
	Explanation: This command displays information about the agent position for position ID 5005.
End	

Responses

The following table provides explanations of the responses to the agtpos command.

Responses for the agtpos command	
MAP output	**Meaning and action**
`ACD GROUP NOT FOUND` `CHECK TABLE ACDGRP`	
	Meaning: You entered an invalid group name.
	Action: Reenter the command with an appropriate group name.
-continued-	

agtpos (continued)

Responses for the agtpos command (continued)	
MAP output	**Meaning and action**
| ```
ACD GROUP NOT FOUND
CHECK TABLE ACDSGRP
``` | |
| | **Meaning:** You entered an invalid subgroup name. |
| | **Action:**   Reenter the command with an appropriate subgroup name. |
| ```
Agent Position ID: posid
Agent Login ID: loginid
``` | |
| | **Meaning:** The system displays the login ID and the ID of the agent position for the agent specified in the command string agtpos dn dn, or agtpos posid posid brief. |
| | **Action:**   None |
| ```
Agent Position ID: posid
* *Agent not Logged in* *
``` | |
| | **Meaning:** No agent is logged in at the agent position identified by the parameter posid. |
| | **Action:**   None |
| ```
AGENTS IN ACD GROUP:   acdgroup
Agent Position ID:   posid
Agent Login ID:   loginid

Agent Position ID:   posid
* *Agent not Logged in* *

Agent Position ID:  posid
Agent Login ID: loginid
``` | |
	Meaning: For the group specified in the command string agtpos group acdgroup all, the system displays all the agent position IDs in the group, and the login ID of the agents currently using them. With the command string agtpos all all, this information is displayed for each accessible ACD group.
	Action: None
-continued-	

agtpos (end)

Responses for the agtpos command (continued)	
MAP output	**Meaning and action**
| ```
Agent LEN & DN: LEN HOST len DN dn
Agent Position ID:  posid
Agent Group:  grpnme
ACD Subgroup:  subgrp
Agent Login ID:  loginid

Associated Supervisor Info:
Supervisor LEN & DN: len  dn
Supervisor Position ID: posid
``` | |
| | **Meaning:** The system displays detailed information about the agent position specified in the command string agtpos posid posid full. With the command string agtpos group grpnme subgroup subgrp full, this information is displayed for each agent position in the specified group. |
| | **Action:**   None |
| ```
Invalid ACD groupname
Valid names are: <grpname>
``` | |
	Meaning: You entered an invalid group name.
	Action: Valid group names are listed. Reenter a valid group name or enter abort to cancel the command.
End	

audiogroup

Function

Use the audiogroup command to display the name of the audio group used to give a recorded announcement to callers in the incoming call queue. The selection of announcements available to an Automatic Call Distribution (ACD) group depends on the datafill in Table AUDIO.

audiogroup command parameters and variables		
Command	**Parameters and variables**	
audiogroup	all	
	group	*acdgroup*
Parameters and variables	**Description**	
acdgroup	This variable identifies the ACD group.	
all	This parameter displays the name of the audio group associated with each of the ACD groups.	
group	This parameter displays the name of the audio group associated with the specified ACD group.	

Qualifications

None

Example

The following table provides an example of the audiogroup command.

Example of the audiogroup command	
Example	**Task, response, and explanation**
audiogroup group acdgrp1 ↵ *where*	
acdgroup 1 identifies the ACD group	
	Task: Display the audio information for an ACD group.
	Response: `AUDIO group for ACDGRP1:  AUDIO2`
	Explanation: This command displays the audio information for the ACD group acdgrp1.

audiogroup (end)

Responses

The following table provides explanations of the responses to the audiogroup command.

Responses for the audiogroup command	
MAP output	**Meaning and action**
`*** ACDGRP1 does not have AUDIO option.`	
	Meaning: You specified an ACD group that does not have the audio option.
	Action: None
`AUDIO group for acdgroup:   audio`	
	Meaning: The system displays the name of the audio group assigned to the ACD group specified in the command string audiogroup acdgroup.
	Action: None

clrroute

Function

Use the clrroute command to display the name of the clearing route used for forced night service.

clrroute command parameters and variables	
Command	**Parameters and variables**
clrroute	all
	group *acdgroup*

Parameters and variables	**Description**
acdgroup	This variable identifies the Automatic Call Distribution (ACD) group.
all	This parameter displays ACD subgroup information for all the ACD groups.
group	This parameter displays the ACD subgroup information for a particular ACD group.

Qualifications

None

Example

The following table provides an example of the clrroute command.

Example of the clrroute command		
Example	**Task, response, and explanation**	
clrroute all ↵		
	Task:	Display the name of the clearing route.
	Response:	CLRROUTE For Group: TSTIBNACD No Forced Night Service . . . CLRROUTE For Group: S3NACD10 No Forced Night Service
	Explanation:	This command displays the name of the clearing route.

clrroute (end)

Responses

The following table provides explanations of the responses to the clrroute command.

Responses for the clrroute command	
MAP output	**Meaning and action**
``` *** Invalid ACD group name. Valid names are: <Acd_grp?> <Acd_grp?> <Acd_grp?> ```	
	Meaning: You entered an invalid ACD group name.
	Action: Enter one of the group names listed to continue or abort to cancel.
``` Invalid symbol: <Group?> {ALL, GROUP <Acd_grp?> STRING} Enter: <Groups?> ```	
	Meaning: You entered an invalid parameter.
	Action: Enter a correct parameter to continue or abort to cancel.

counts

Function

Use the counts command to display the Automatic Call Distribution (ACD) subgroup breakdown (name, supervisor position and number of agent positions) for one or all ACD groups.

counts command parameters and variables		
Command	**Parameters and variables**	
counts	all	
	group	*acdgroup*
Parameters and variables	**Description**	
acdgroup	This variable identifies the ACD group.	
all	This parameter displays ACD subgroup information for all the ACD groups.	
group	This parameter displays the ACD subgroup information for a particular ACD group.	

Qualifications

None

Example

The following table provides an example of the counts command.

counts (end)

Example of the counts command

Example   **Task, response, and explanation**

---

counts group plan1 ↵
where

plan 1   identifies the ACD group

            Task: Display the number of agent positions for an ACD group.

            Response:
```
            Group PLAN1
ACD Subgroup:               1
Supervisor Position:        9999
Agent Positions in this Subgroup: 1
ACD Subgroup:               2
Supervisor Position:        9997
Agent Positions in this Subgroup: 1
```

            Explanation: This command displays the number of agent positions in each subgroup of ACD group plan1.

Responses

The following table provides explanations of the responses to the counts command.

Responses for the counts command

MAP output   **Meaning and action**

```
*** Invalid ACD group name.
    Valid names are:
<Acd_grp?> <Acd_grp?> <Acd_grp?>
```

            Meaning: You entered an invalid ACD group name.

            Action: Enter one of the group names listed to continue or abort to cancel.

```
Invalid symbol: <Group?> {ALL,
                     GROUP <Acd_grp?> STRING}
Enter: <Groups?>
```

            Meaning: You entered an invalid parameter.

            Action: Enter a correct parameter to continue or abort to cancel.

---

fiaudgrp

Function

Use the fiaudgrp command to display the audio group played to all incoming calls that are queued or presented.

fiaudgrp command parameters and variables	
Command	**Parameters and variables**
fiaudgrp	all
	group *acdgroup*

Parameters and variables	**Description**
acdgroup	This variable defines the Automatic Call Distribution (ACD) group to display.
all	This parameter displays all ACD groups.
group	This parameter displays a particular ACD group.

Qualifications

None

Examples

The following table provides examples of the fiaudgrp command.

fiaudgrp (continued)

Examples of the fiaudgrp command	
Example	**Task, response, and explanation**
fiaudgrp all ↵	
	Task: Display fiaudio information for all ACD groups.

	Response: `FI AUDIO GROUP for ACDGRP1  :   AUDIO7`
	`FI AUDIO GROUP for ACDGRP2  :   AUDIO7`
	`ACDGRP3 does not have FIAUDIO option`
	`FI AUDIO GROUP for ACDGRP4   :   AUDIO5`

	Explanation: This command displays fiaudio information for all ACD groups.
fiaudgrp group acdgrp1 ↵ *where* acdgrp1 specifies the group	
	Task: Display fiaudio information for an ACD group.
	Response: `FI AUDIO GROUP for ACDGRP1   :   AUDIO7`
	Explanation: This command displays fiaudio information for the ACD group acdgrp1.

Responses

The following table provides explanations of the responses to the fiaudgrp command.

Responses for the fiaudgrp command	
MAP output	**Meaning and action**

`FI AUDIO GROUP for ACDGRP1   :   AUDIO7`
`FI AUDIO GROUP for ACDGRP2   :   AUDIO7`
`ACDGRP3 does not have FIAUDIO option`
`FI AUDIO GROUP for ACDGRP4   :   AUDIO5`	

	Meaning: The system displays information concerning the fiaudio group assigned to all ACD groups. **Action:** None
-continued-	

fiaudgrp (end)

Responses for the fiaudgrp command (continued)
MAP output   **Meaning and action**

`FI AUDIO GROUP for ACDGRP1   :   AUDIO7`
Meaning: The system displays information concerning the fiaudio group assigned to a specific ACD group. **Action:**   None

| ```
*** Invalid ACD group name.
    Valid names are:
<Acd_grp?> <Acd_grp?> <Acd_grp?>
Correct Format: <acd_grp?> STRING
Enter: <Acd_grp?>
``` |
| **Meaning:** You entered an invalid ACD group name.

Action: Enter one of the group names listed to continue or abort to cancel. |

| ```
Invalid symbol: <Which_Groups?> {ALL,
                        GROUP <Acd_grp?> STRING}
Enter: <Which_Groups?>
``` |
| **Meaning:** You entered an invalid parameter.

Action: Check the spelling and enter the correct parameter to continue or abort to cancel. |
| **End** |

foaudgrp

Function

Use the foaudgrp command to display the audio group played to all
threshold overflow calls prior to rerouting.

foaudgrp command parameters and variables	
Command	**Parameters and variables**
foaudgrp	all
	group *acdgroup*

Parameters and variables	**Description**
acdgroup	This variable defines the Automatic Call Distribution (ACD) group to display.
all	This parameter displays all ACD groups.
group	This parameter displays a particular ACD group.

Qualifications

None

Examples

The following table provides examples of the foaudgrp command.

Examples of the foaudgrp command	
Example	**Task, response, and explanation**
foaudgrp all ↵	
	Task: Display foaudio information for all ACD groups.

	Response: `FO AUDIO GROUP for ACDGRP1   :   AUDIO7`
	`FO AUDIO GROUP for ACDGRP2   :   AUDIO7`
	`ACDGRP3 does not have FOAUDIO option`
	`FO AUDIO GROUP for ACDGRP4   :   AUDIO5`

	Explanation: This command displays foaudio information for all ACD groups.
-continued-	

foaudgrp (continued)

Examples of the foaudgrp command (continued)	
Example	**Task, response, and explanation**
foaudgrp group acdgrp1 ↵ *where* acdgrp 1 specifies the ACD group	
	Task: Display foaudio information for an ACD group. **Response:** FO AUDIO GROUP for ACDGRP1 : AUDIO7 **Explanation:** This command displays foaudio information for the ACD group acdgrp1.
End	

Responses

The following table provides explanations of the responses to the foaudgrp command.

Responses for the foaudgrp command	
MAP output	**Meaning and action**

```FO AUDIO GROUP for ACDGRP1   :   AUDIO7```
```FO AUDIO GROUP for ACDGRP2   :   AUDIO7```
```ACDGRP3 does not have FOAUDIO option```
```FO AUDIO GROUP for ACDGRP4 : AUDIO5```	

	Meaning: The system displays information concerning the foaudio group assigned to all ACD groups. **Action:** None
```FO AUDIO GROUP for ACDGRP1 : AUDIO7```	
	Meaning: The system displays information concerning the foaudio group assigned to a specific ACD group. **Action:** None
-continued-	

Responses for the foaudgrp command (continued)

MAP output   Meaning and action

```
*** Invalid ACD group name.
    Valid names are:
<Acd_grp?> <Acd_grp?> <Acd_grp?>
Correct Format: <acd_grp?> STRING
Enter: <Acd_grp?>
```

Meaning: You entered an invalid ACD group name.

Action: Enter one of the group names listed to continue or abort to cancel.

```
Invalid symbol: <Which_Groups?> {ALL,
                                 GROUP <Acd_grp?> STRING}
Enter: <Which_Groups?>
```

Meaning: You entered an invalid parameter.

Action: Check the spelling and enter the correct parameter to continue or abort to cancel.

End

groupinfo

Function

Use the groupinfo command to display the following information about the Automatic Call Distribution (ACD) setup for a particular ACD group or ACD directory number (ACDDN):

- ACD group
- primary ACDDN and its incoming call priorities
- supplementary ACDDN and its incoming call priority
- customer group
- logon ID partition
- ringing threshold
- threshold route
- night service route
- maximum call queue size
- maximum call queue wait time
- maximum call transfer queue size
- audio group
- night service audio group
- whether or not the group has the management reports option
- administration group
- number of agents in the group
- ACD display digits
- recorded announcement threshold
- enhanced overflow route
- multi-stage queue status thresholds
- virtual facility group (VFG)
- wrap-up time
- whether or not the ACD group has a forced night service clearing route
- forced incoming audio group
- forced outgoing audio group

groupinfo (continued)

groupinfo command parameters and variables				
Command	**Parameters and variables**			
groupinfo	acddn all group	*npa* *acdgroup*	*oc*	*ext*

Parameters and variables	Description
acddn	This parameter displays information for the ACD group associated with a particular ACDDN.
acdgroup	This variable identifies the ACD group. The valid entry range is 0-256.
all	This parameter displays information for all the ACD groups.
ext	This variable specifies the extension number. The valid entry range is 0-9999.
group	This parameter displays ACD configuration information for a particular ACD group.
npa	This variable specifies the numbering plan area. The valid entry range is 0-999.
oc	This variable specifies the office code. The valid entry range is 0-999.

Qualifications

None

groupinfo (continued)

Examples

The following table provides examples of the groupinfo command.

Examples of the groupinfo command	
Example	**Task, response, and explanation**
groupinfo acddn 613 555 1212 ↵ *where* 613 specifies the numbering plan area 555 specifies the office code 1212 specifies the extension	

Task:		Display information about an ACD group using an ACDDN.

Response:		`ACD Group:  ACDGRP1`
		
		`ACD Group :                 ACDGRP1`
		`Primary ACDDN:              613 555 1212`
		`.  .  .`
		`.  .  . <- existing fields`
		`.  .  .`
		
		(No wrap-up time field is displayed)

Explanation:		This command displays information about an ACD group without the varwrap option, using ACDDN 6135551212.

-continued-

groupinfo (continued)

Examples of the groupinfo command (continued)

| **Example** | **Task, response, and explanation** |

groupinfo all ↵

Task:	Display information about all ACD groups.	
Response:	ACD Group: TSTIBNACD	
	No ACDDN's For This Group	
	Customer Group:	IBNTST
	Loginid Partition:	0
	Ringing Threshold:	50 Secs
	Threshold Route:	OFRT 60
	Night Service Route:	OFRT 60
	Max Call Queue Size:	5
	Max Call Queue Wait Time:	60 Secs
	Management Report Option:	NO
	SCAI Group Option:	NO
	Administration Group:	0
	Number of Agents in Group:	3
	OrgAnn:	NO
	Forced Night Service:	NO
	.	
	.	
	.	
	ACD Group: GRP6B	
	No ACDDN's For This Group	
	.	
	.	
	.	
	Forced Night Service:	NO
Explanation:	This command displays information about all ACD groups.	

-continued-

---

groupinfo (continued)

Examples of the groupinfo command (continued)

Example　　　**Task, response, and explanation**

groupinfo group usaa3 ↵
where

usaa3　　　identifies the ACD group

Task:	Display group information about an ACD group.	

Response:
```
ACD Group:  USAA3

No ACDDN's For This Group
      Customer Group:              COMKODAK
      Ringing Threshold:           20 Secs
      Threshold Route:             OFRT 100)
      Night Service Route:         OFRT 1001
      Max Call Queue Size:         10
      Max Call Queue Wait Time:    20 Secs
      Management Report Option:    NO
      Administration Group:        0
      Number of Agents in Group:   3
```

Explanation:　This command displays group information about ACD group usaa3. The group has no options or directory numbers.

-continued-

groupinfo (continued)

Examples of the groupinfo command (continued)

Example	Task, response, and explanation

groupinfo group usaa2 ↵
where

usaa2   specifies the group name

Task:	Display information about an ACD group.

Response:

```
ACD Group:   USAA2

              Primary ACDDN:               613 722 4447
                Trunk Priority:            0
                Line Priority:             0

              Customer Group:              COMKOKAK
              Ringing Threshold:           20 Secs
              Threshold Route:             OFRT 1000
              Night Service Route:         OFRT 1001
              Max Call Queue Size:         10
              Max Call Queue Wait Time:    20 Secs
              Audio Group:                 AUDIO5
              Ringing Before Announcement: YES
              Recorded Announcement Threshold: 15 SECS
              Management Report Option:    NO
              Administration Group:        0
              Number of Agents in Group:   2
              Enhanced Overflow Route:
                  USAA1    USAA3
```

Explanation: This command displays information about the ACD group usaa2.
The group has ACD primary directory numbers, and the Enhanced
Overflow and audio options.

-continued-

---

groupinfo (continued)

Examples of the groupinfo command (continued)

Example	Task, response, and explanation

groupinfo group usaa1 ↵
where

usaa1   identifies the ACD group

Task: Display information about an ACD group.

Response:
```
ACD Group:  USAA1
        Primary ACDDN:                  613 722 4449
          Trunk Priority:               0
          Line Priority:                0
          Acd-Dn Name:                  Joe Smith
        Supplementary ACDDN:            613 722 5020
          Dn Priority:                  3
          Acd-Dn Name:                  Jane Smith
        Supplementary ACDDN:            613 722 5021
          Dn Priority:                  3
        ***No Name Associated***
        Supplementary ACDDN:            613 722 5022
          Dn Priority:                  3
        ***No Name Associated***
        Customer Group:                 COMKODAK
        Loginid Partition               1
        Ringing Threshold:              20 Secs
        Threshold Route:                OFRT 1003
        Night Service Route:            OFRT 1002
        Max Call Queue Size:            10
        Max Call Queue Wait Time:       180 Secs
        Audio Group:                    AUDIO2
        Ringing Before Announcement:    YES
        Recorded Announcement Threshold: 30 SECS
        Management Report Option:       NO
        Administration Group:           0
        Number of Agents in Group:      3
        Enhanced Overflow Route:
            USAA2  USAA3  USAA4  USAA5
        Multi-Stage Queue Status Thresholds
          T1:  10   T2:  20   T3:  30
ACD Display Digits:
```

Explanation: This command displays information about the ACD group usaa1. The group has primary and supplementary directory numbers, Enhanced Overflow route, audio option, multi-stage queue status option, and ACD display option.

-continued-

groupinfo (continued)

Examples of the groupinfo command (continued)	
Example	**Task, response, and explanation**

groupinfo group grp1 ↵
where

grp 1   identifies the ACD group

	Task:	Display information about an ACD group.
	Response:	ACD Group: GRP1

		ACD Group: GRP1
		Primary ACDDN: 613 722 4449
		  Trunk Priority: 1
		  Line Priority: 0
		. . .
		. . . <- existing fields
		. . .
		. . .
		FI Audio Group: AUDIO2
		FO Audio Group: AUDIO3
		OrgAnn: YES

	Explanation:	This command displays information about the ACD group grp1 with the organn option.

groupinfo group acdgrp1 ↵
where

acdgrp 1   identifies the ACD group

	Task:	Display information about an ACD group.
	Response:	ACD Group: ACDGRP1
		ACD Group : ACDGRP1 Primary ACDDN: 613 555 1212 . . . . . . <- existing fields . . . Wrap-up Time: 30
	Explanation:	This command displays information about the ACD group acdgrp1 with the varwrap option of dwrptime 30 seconds.
End		

groupinfo (end)

Responses

The following table provides explanations of the responses to the groupinfo command.

Responses for the groupinfo command	
MAP output	**Meaning and action**
``` *** Invalid ACD group name Valid names are: <acdgroup> ```	
	Meaning: You entered an invalid ACD group name.
	Action: Reenter the command with an appropriate ACD group name.
``` No ACD Group Exists for this ACDDN ```	
	Meaning: You asked for groupinfo but there are no groups.
	Action: None

groupname

Function

Use the groupname command to display the Automatic Call Distribution (ACD) group name, directory number (DN) type, and priorities for a particular DN.

groupname command parameters and variables				
Command	**Parameters and variables**			
groupname	acddn	*npa*	*oc*	*ext*
	all			
	group	*acdgroup*		

Parameters and variables	Description
acddn	This parameter displays information for the ACD group associated with a particular ACDDN.
acdgroup	This variable identifies the ACD group.
all	This parameter displays information for all the ACD groups.
ext	This variable specifies the extension number. The valid entry range is 0-9999.
group	This parameter displays ACD information for a particular ACD group.
npa	This variable specifies the numbering plan area. The valid entry range is 0-999.
oc	This variable specifies the office code. The valid entry range is 0-999.

Qualifications

None

groupname (continued)

Examples

The following table provides examples of the groupname command.

Examples of the groupname command	
Example	**Task, response, and explanation**
groupname 613 722 4449 ↵ *where* 613 specifies the numbering plan area 722 specifies the office code 4449 specifies the extension number	

	Task: Display information about a DN.
	
	Response:
	`Directory Number:    613 722 4449`
	`ACD Group Name:      ABCD1`
	`DN Type:             PRIM`
	`   Trunk:            1`
	`   Line:             0`
	
	Explanation: This command displays information about the primary DN 613 722 4449.

groupname 613 722 4449 ↵ *where* 613 specifies the numbering plan area 722 specifies the office code 4449 specifies the extension number	

	Task: Display information about a DN.
	
	Response:
	`Directory Number:    613 722 4449`
	`ACD Group Name:      ABCD1`
	`DN Type:             SUPP`
	`  DN Priority        1`
	
	Explanation: This command displays information about the supplementary DN 613 722 4449.

groupname (end)

Responses

The following table provides explanations of the responses to the groupname command.

Responses for the groupname command	
MAP output	**Meaning and action**
```*** Invalid ACD group name.``` ``` Valid names are: <acdgroup>```	
	Meaning: You entered an invalid group name.
	Action: Reenter the correct group name to continue or abort to cancel the command.
```Wrong type: <Npa?> {0 to 999}``` ```Enter: <Npa?> <Office code?> <Extension?>```	
	Meaning: You entered inappropriate information for the DN.
	Action: Reenter the DN to continue or abort to cancel the command.

help

Function

Use the help command to receive online documentation for the ACDSHOW directory. Also see q.

help command parameters and variables	
Command	**Parameters and variables**
help	_all_ _command_nam_

Parameters and variables	Description
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
_command_nam_	This variable specifies a valid ACDSHOW directory command name. When the _command_nam_ variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

help (continued)

Examples

The following table provides examples of the help command.

Examples of the help command	
Example	**Task, response, and explanation**
help ↵	
	Task: Access online documentation.

	Response: `HELP:  is a subcommand of ACDSHOW which displays general information about ACDSHOW and its various commands.  To obtain information about a given command, enter the following:  HELP <command_name>.`
	`To determine the actual syntax of a particular command enter the following:`
	`Q <command_name>,  ie. Q AGTPOS`
	`Available commands are:`

	`ACDDNS        HELP        QUIT`
	`ACDSHOW       LOADMGMT    STATUS`
	`ADMINGROUP    LOGINID     SUPERVISOR`
	`AGTPOS        MODE        TABENTRY`
	`AUDIOGROUP    NSAUDGRP    THRESHOLD`
	`COUNTS        NSROUTE     THROUTE`
	`GROUPINFO     OVFLROUTE   VALIDAUDIO`
	`GROUPNAME     PASSWORD    VALIDROUTES`
	`FIAUDGRP      FOAUDGRP    CLRROUTE`

	Explanation: This example typifies a response for the help command string.
help admingroup ↵ *where* admingroup specifies the command name	
	Task: Access online documentation for a command.
	Response: `Command to display administration group information.`
	Explanation: This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

loadmgmt

Function

Use the loadmgmt command to access the load management directory (LOADMGMT). Enter command acdshow at the CI level. When the system responds with the acdshow prompt, enter loadmgmt.

loadmgmt command parameters and variables	
Command	**Parameters and variables**
loadmgmt	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the loadmgmt command.

Example of the loadmgmt command		
Example	**Task, response, and explanation**	
loadmgmt ↵		
	Task:	Access the load management directory.
	Response:	LOADMGMT>>
	Explanation:	This command accesses the load management directory.

Response

The following table provides an explanation of the response to the loadmgmt command.

Response for the loadmgmt command	
MAP output	**Meaning and action**
LOADMGMT>>	
	Meaning: You have accessed the load management directory.
	Action: None

loginid

Function

Use the loginid command to display information on a particular agent login identification (ID).

loginid command parameters and variables		
Command	**Parameters and variables**	
loginid	*logid* [brief / full] *custgrp*	

Parameters and variables	**Description**
brief	This parameter displays the position ID and Table ACDLOGIN information.
custgrp	This variable specifies the customer group used to determine the partition number for the login ID. The ACD group must be assigned the ENLOG option in T able CUSTACD or the partition number defaults to zero.
full	This parameter displays the position ID, T able ACDLOGIN information, and the LEN, DN, and ACD group and subgroup of the agent who is currently using the login ID.
logid	This variable identifies the agent login ID. The valid entry range is 0001-9999.

Qualifications

None

Examples

The following table provides examples of the loginid command.

loginid (continued)

<table>
<tr><td colspan="2">Examples of the loginid command</td></tr>
<tr><td>Example</td><td>Task, response, and explanation</td></tr>
<tr>
<td colspan="2">
loginid   3409 ↵
<i>where</i>

3409   identifies the agent login ID
</td>
</tr>
<tr>
<td></td>
<td>

Task: Display information for an agent login ID.

Response:
```
Agent Position ID:   0020
Agent Login ID:      3409
Table ACDLOGIN related information:
  Password: 2345
  Customer Group: CUSTGRP1
```

Explanation: This command displays information about agent login ID 3409.
</td>
</tr>
<tr>
<td colspan="2">
loginid 3409   full ↵
<i>where</i>

3409   identifies the agent login ID
</td>
</tr>
<tr>
<td></td>
<td>

Task: Display full information for an agent login ID.

Response:
```
Agent Position ID: 0020
Agent Login ID:    3409
Agent LEN & DN:    LEN HOST 02 1 01 14 DN 7225005
ACD Group:         GROUPA
ACD Subgroup:      1
Call Forcing:      NO
Table ACDLOGIN related information:
  Password: 2345
  Customer Group: CUSTGRP1
  PAQ Size:      15
```

Explanation: This command displays agent and supervisor information about agent login ID 3409.
</td>
</tr>
<tr><td colspan="2" align="center">-continued-</td></tr>
</table>

loginid (continued)

Examples of the loginid command (continued)	
Example	**Task, response, and explanation**

loginid 2222 full ↵
where

2222	identifies the agent login ID	
	Task:	Display full information for an agent login ID.

	Response:	Agent Position ID: 1234
		   Agent Login ID: 2222
		
		         . . .
		         . . .
		         . . .
		      Call Forcing: YES
		      Wrap-Up Time: 45
		      MIS Group: ACDGRP2

	Explanation:	This command displays an agent position ID with a variable wrap-up time and with the misgroup option.

loginid 2222 full ↵
where

2222	identifies the agent login ID	
	Task:	Display full information for an agent login ID.

	Response:	Agent Position ID: 1234
		   Agent Login ID: 2222
		
		         . . .
		         . . .
		         . . .
		      Call Forcing: YES
		      MIS Group: ACDGRP2

	Explanation:	This command displays an agent position ID without a variable wrap-up time and with the misgroup option.
-continued-	

loginid (continued)

Examples of the loginid command (continued)	
Example	**Task, response, and explanation**
loginid 2222 full ⏎ *where*	
2222	identifies the agent login ID

	Task:	Display full information for an agent login ID.

	Response:	`Agent Position ID:        1234`
		 `  Agent Login ID:          2222`
		
		 `    . . .`
		 `    . . .`
		 `    . . .`
		 `  Call Forcing:           NO`
		 `  Wrap-Up Time:           45`

	Explanation:	This command displays agent position ID information for an agent with a variable wrap-up time.

End

Responses

The following table provides an explanation of the response to the loginid command.

Responses for the loginid command	
MAP output	**Meaning and action**

`Agent Position ID:    posid`
 `Agent Login ID:       logid`
 `Table ACDLOGIN related information:`
 `Password:             nnnn`
 `Customer Group:       custgrp`
 `PAQ Size:             nn`	

	Meaning: You entered the command correctly.
	Action: None

-continued-

loginid (end)

Responses for the loginid command (continued)	
MAP output	**Meaning and action**

`*** Invalid Customer Group name.`
`    Valid names are:`
`<custgrp> <custgrp> <custgrp> <custgrp>`
`EITHER incorrect optional parameter(s) OR too many parameters.`	

	Meaning: You entered an invalid customer group name.
	Action: Reenter the command with one of the listed customer groups.
`No Agent Logged In with Login ID <id> Partno <partno>`	
	Meaning: You entered the command correctly, but no agent was found.
	Action: None
`No customer group has been specified.  Partition number will default to 0`	
	Meaning: You left out the customer group variable.
	Action: Reenter the command if the information is not found and supply a customer group.
`Out of range: <Id?> {1 to 9999}`	
	Meaning: You entered an invalid login ID.
	Action: Reenter the login ID to continue or abort to cancel.
`The FORM parameter, BRIEF or FULL is required.`	
	Meaning: You left out the parameter brief or full.
	Action: Reenter the command.
End	

mode

Function

Use the mode command to determine how much information is displayed in system responses. The mode setting determines the default mode for commands that have brief and full parameters.

mode command parameters and variables	
Command	**Parameters and variables**
mode	brief full

Parameters and variables	Description
brief	This parameter causes commands to display agent information. The brief format is the current mode when entering the ACDSHOW directory.
full	This parameter requires commands to show associated agent and supervisor information.

Qualifications

None

Examples

The following table provides examples of the mode command.

Examples of the mode command		
Example	**Task, response, and explanation**	
mode brief_ ↵		
	Task:	Set the system to display agent information only.
	Response:	`Display mode has been set to brief.`
	Explanation:	This command sets the system to display agent information only when responding to commands.
	-continued-	

mode (end)

Examples of the mode command (continued)		
Example	**Task, response, and explanation**	
mode full ↵		
	Task:	Set the system to display agent and supervisor information.
	Response:	`Display mode has been set to FULL.`
	Explanation:	This command sets the system to display agent and supervisor information when responding to commands.
End		

Responses

The following table provides explanations of the responses to the mode command.

Responses for the mode command		
MAP output	**Meaning and action**	
`Display mode has been set to BRIEF.`		
	Meaning:	The command mode brief has been executed. The system responses display agent information.
	Action:	Enter commands as required.
`Display mode has been set to FULL`		
	Meaning:	The system executed the mode full command. While mode is set to full, the system responds to all commands and displays information on agents and supervisors.
	Action:	Enter commands as required.

nsaudgrp

Function

Use the nsaudgrp command to display the name of the audio group assigned to the Automatic Call Distribution (ACD) group for night service announcement to the callers before routing the call to night service.

nsaudgrp command parameters and variables	
Command	**Parameters and variables**
nsaudgrp	all
	group *acdgroup*

Parameters and variables	**Description**
acdgroup	This variable specifies the ACD group for the night service audio group.
all	This parameter displays the night service audio group assigned to all ACD groups
group	This parameter indicates the ACD group is specified.

Qualifications

None

Examples

The following table provides examples of the nsaudgrp command.

Examples of the nsaudgrp command		
Example	**Task, response, and explanation**	
nsaudgrp group abcgrp23 ↵		
where		
abcgrp23	specifies the ACD group	
	Task:	Display the night service audio group assigned to an ACD group.
	Response:	NS AUDIO GROUP FOR ABCGRP23: AUDIO98
	Explanation:	This command displays the night service audio group assigned to ACD group abcgrp23.
-continued-		

nsaudgrp (end)

Examples of the nsaudgrp command (continued)	
Example	**Task, response, and explanation**
nsaudgrp all ↵	
	Task: Display the night service audio group assigned to all ACD groups.
	Response: NS AUDIO GROUP FOR ABCGRP1: AUDIO98 NS AUDIO GROUP FOR ABCGRP2: AUDIO13 ABCGRP3 DOES NOT HAVE NSAUDIO OPTION NS AUDIO GROUP FOR ABCGRP4: AUDIO1
	Explanation: This command displays the night service audio group assigned to all ACD groups.
End	

Responses

The following table provides explanations of the responses to the nsaudgrp command.

Responses for the nsaudgrp command	
MAP output	**Meaning and action**
NS AUDIO GROUP FOR acd_grp: nsaudio group	
	Meaning: The system displays the night service audio group assigned to the specified ACD group.
	Action: None
acd_grp DOES NOT HAVE NSAUDIO OPTION	
	Meaning: The specified ACD group does not have the NSAUDIO option assigned to it.
	Action: None

nsroute

Function

Use the nsroute command to display the night service route for inactive Automatic Call Distribution (ACD) groups. An ACD group is considered in night service mode when all agents have activated the MakeBusy feature or when the ACD supervisor has activated the Night Service feature.

nsroute command parameters and variables	
Command	**Parameters and variables**
nsroute	all group *acdgroup*
Parameters and variables	**Description**
acdgroup	This variable identifies the ACD group.
all	This parameter displays the night service routes for all the ACD groups.
group	This parameter displays the night service route for a particular ACD group.

Qualifications

None

Examples

The following table provides examples of the nsroute command.

Examples of the nsroute command	
Example	**Task, response, and explanation**
nsroute acdpsaph1 ↵	
	Task: Display the night service routes for a specific ACD group.
	Response: NSROUTE For Group: ACDPSAPH1 Route: OFRT 911 Routes To Trunk Group E911OGES.
	Explanation: This command displays the night service routes for the ACD group acdpsaph1.
-continued-	

nsroute (end)

Examples of the nsroute command (continued)

Example	Task, response, and explanation

nsroute all ↵

	Task:	Display the night service routes for all ACD groups.

	Response:	`NSROUTE For Group: ACD1`
		`Route: IBNRTE 290`
		`Routes To Trunk Group IBNRTE 226.`
		
		`NSROUTE For Group: ACD2`
		`Route: OFRT 924`
		`Routes To Trunk Group OLAMADCM.`
		
		`NSROUTE For Group: ACD3`
		`Route: IBNRTE 226`
		`Routes to Trunk Group E911OGES.`

	Explanation:	This command displays the night service routes for all ACD groups.

End

Responses

The following table provides explanations of the responses to the nsroute command.

Responses for the nsroute command

MAP output	Meaning and action

```
*** Invalid ACD group name
    Valid names are: <acdgroup>
```

	Meaning: You entered an invalid group name or left out the group parameter.
	Action: Enter an appropriate group name to continue or abort to cancel.

```
Invalid symbol: <Which_Groups?> {ALL,
                      GROUP <Acd_grp?> STRING}
```

	Meaning: You entered an invalid parameter.
	Action: Enter the appropriate parameter to continue or abort to cancel.

---

ovflroute

Function

Use the ovflroute command to display the route of Automatic Call Distribution (ACD) groups to which calls overflowing from an ACD group queue are directed.

ovflroute command parameters and variables		
Command	**Parameters and variables**	
ovflroute	all	
	group	*acdgroup*

Parameters and variables	**Description**
acdgroup	This variable specifies the ACD group.
all	This parameter displays the overflow routes for all the ACD groups.
group	This parameter displays the overflow route for a specific ACD group.

Qualifications

None

Examples

The following table provides examples of the ovflroute command.

Examples of the ovflroute command		
Example	**Task, response, and explanation**	
ovflroute all ↵		
	Task:	Display the overflow routes for all ACD groups.

	Response:	```OVFLROUTE For Group: PLAN1```
		```1) Group PLAN2```
		```OVFLROUTE For Group: PLAN2```
		```1) Group PLAN1```
		```2) Group PLAN3```
		```OVFLROUTE For Group: PLAN3```
		```1) Group PLAN2```

	Explanation:	This command displays the overflow routes for all ACD groups.
	-continued-	

ovflroute (end)

Examples of the ovflroute command (continued)
Example   **Task, response, and explanation**
ovflroute   **group plan1** ↵ *where* plan 1   identifies the ACD group

Task: Display the overflow routes for an ACD group.
 	
      Response: `OVFLROUTE For Group:       PLAN1`
                    `1) Group PLAN2`
                    `2) Group PLAN3`
 	
      Explanation: This command displays the overflow routes for ACD group plan1.

End

Responses

The following table provides explanations of the responses to the ovflroute command.

Responses for the ovflroute command
MAP output   **Meaning and action**
`*** Invalid ACD group name` `    Valid names are: <acdgroup>`
Meaning: You entered an invalid group name or left out the group parameter.   **Action:**   Enter an appropriate group name from the list displayed to continue or abort to cancel.
`Invalid symbol: <Which_Groups?> {ALL,` `                      GROUP <Acd_grp?> STRING}`
Meaning: You entered an invalid parameter.   **Action:**   Enter the appropriate parameter to continue or abort to cancel.

password

Function

Use the password command to display the password for a given login ID.

password command parameters and variables		
Command	**Parameters and variables**	
password	all	
	loginid *loginid*	$\left[\begin{array}{c} \underline{0} \\ custgrp \end{array}\right]$

Parameters and variables	Description
0	Omitting this entry forces the system to default to using Table ACDLOGIN.
all	This parameter shows all login IDs.
custgrp	This variable specifies the customer group used to determine the partition number into Table ACDENLOG, if the customer group is assigned the ENLOG option.
loginid	This parameter indicates a login ID is specified.
loginid	This variable specifies the ACD agent login ID. The valid entry range is 0001-9999.

Qualifications

The password command is qualified by the following exceptions, restrictions and limitations:

- The login ID must be in Table ACDLOGIN.
- The customer group must be in Table ACDENLOG if the enlog option is active in Table CUSTACD.

password (continued)

Example

The following table provides an example of the password command.

Example of the password command	
Example	**Task, response, and explanation**
password loginid 1234 e911 ↵ *where* 1234 specifies the login ID e911 specifies the customer group	
	Task: Display a login ID. **Response:** Password for ACD Login Id 1234 Login_Partno 1 is: 2222 **Explanation:** This command displays the password for login ID 1234 and customer group e911.

Responses

The following table provides explanations of the responses to the password command.

Responses for the password command	
MAP output	**Meaning and action**
No password associated with ACD Login ID: <partno> <loginid>	
	Meaning: You entered a loginid that does not have the password option datafilled. The command aborts. **Action:** Datafill the password option in Table ACDENLOG and reenter the command.
Password for ACD Login ID <loginid> Partno <partno> is: <password>	
	Meaning: You entered the command successfully. **Action:** None
-continued-	

password (end)

Responses for the password command (continued)	
MAP output	**Meaning and action**
`Password for ACD Login ID <partno> <loginid> is not datafilled in table ACDENLOG.`	
	Meaning: You entered information for an ACD Agent whose customer group has the ENLOG option active but the partition for that group has not been datafilled in Table ACDENLOG. The command aborts.
	Action: Add the loginid and datafill the password option in Table ACDENLOG.
End	

q

Function

Use the q command to receive online documentation for the ACDSHOW directory.

q command parameters and variables	
Command	**Parameters and variables**
q	*command_name*

Parameters and variables	Description
command_name	This variable specifies a valid ACDSHOW directory command name. When the *command_name* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Examples

The following table provides an examples of the q command.

Examples of the q command	
Example	**Task, response, and explanation**
q admingroup ↵ *where* admingroup specifies the command name	

Task:	Access online documentation.
Response:	

```
Command to display administration group information
Parms: <How?> {ALL,
               MYADMINGROUP,
               ADMINGROUP <Acd_admngr?> {1 to 255},
               ACDGROUP <Groups?> {ALL,
                                   GROUP <Acd_grp?> STRING},
               ADMINSUP <Acd_admnsp?> STRING}
```

Explanation:	This example typifies a response for the query command string.

<div align="center">-continued-</div>

q (end)

Examples of the q command (continued)
Example　　　　**Task, response, and explanation**
q　**groupname** ↵ *where* groupname　specifies the command name

	Task:	Access online documentation.

	Response:	`Command to display name, DN, and priority`
		`Parms: <Npa?> {0 to 999}`
		`        <Office code?> {0 to 999}`
		`        <Extension number?> {0 to 9999}`

	Explanation:	This example typifies a response for the query command string.

End

Response

The following table provides an explanation of the response to the q command.

Responses for the q command
MAP output　　　**Meaning and action**
`MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.`

	Meaning:	The directory you are trying to access is not loaded or must be accessed through another directory.
	Action:	None

quit

Function

Use the quit command to exit the ACDSHOW directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	1 level all name n_levels
Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)	
Example	**Task, response, and explanation**
quit all ↵	
	Task: Exit from all levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory	
	Task: Exit from a specified directory without leaving any other directories.
	Response: AMADUMP>>> >
	Explanation: The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵	
	Task: Exit from a specified number of levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End	

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

status

Function

Use the status command to display the current configuration and activity levels of one or all of the Automatic Call Distribution (ACD) groups assigned to you.

status command parameters and variables		
Command	**Parameters and variables**	
status	all	
	group	*acdgroup*
Parameters and variables	**Description**	
acdgroup	This variable identifies the ACD group.	
all	This parameter displays status information for all the ACD groups assigned to you.	
group	This parameter displays status information for a particular ACD group.	

Qualifications

None

status (continued)

Examples

The following table provides examples of the status command.

Examples of the status command	
Example	**Task, response, and explanation**
status all ↵	
	Task: Display status information for all ACD groups.

	Response: Status For ACD Group NTTACD1
	   Call Transfer Recall Time: 0 Secs
	   Call Park Recall Time: 0 Secs
	   Current Call Queue Size: 0
	   Number of Calls Receiving FIAUDIO: 4
	   Max Call Queue Size: 10
	   Current Call Wait Time: 0 Secs
	   Max Call Wait Time: 1800 Secs
	   Number of Agent Positions Assigned: 2
	   Agents logged In: 2
	   Agents Not Logged In: 0
	   Agtpos On ACD Calls: 0
	   Agtpos Idle: 2
	   Agtpos In Not Ready Mode: 0
	===
	Status for ACD Group NTTACD2
	    .
	    .
	    .

	Explanation: This command displays status information for all ACD groups.
status group plan1 ↵ *where* plan 1 identifies the ACD group	
	Task: Display the status of an ACD group.

	Response: Status For ACD Group PLAN1
	Current Call Queue Size: 2
	    .
	    .
	    .
	Agtpos Idle: 1
	Agtpos in Not Ready Mode: 1

	Explanation: This command displays the status of ACD group plan1.

status (end)

Responses

The following table provides explanations of the responses to the status command.

Responses for the status command	
MAP output	**Meaning and action**
`*** Invalid ACD group name` `    Valid names are: <acdgroup>`	
	Meaning: You entered an invalid group name or left out the group parameter.
	Action: Enter an appropriate group name to continue or abort to cancel.
`Invalid symbol: <Which_Groups?> {ALL,` `                          GROUP <Acd_grp?> STRING}`	
	Meaning: You entered an invalid parameter.
	Action: Enter the appropriate parameter to continue or abort to cancel.

supervisor

Function

Use the supervisor command to display a list of one or more supervisor positions and the associated subgroups. Information can be requested for:

- an Automatic Call Distribution (ACD) group

- an ACD subgroup with an ACD group

- a line equipment number (LEN)

- a directory number (DN)

- all ACD groups

- all ACD subgroups within all the ACD groups

- common language location identifier (CLLI)

---

supervisor command parameters and variables

Command	Parameters and variables						
supervisor	all	all					
		subgroup	*subgrp*				
	clli	*clli*	*num*				brief
	dn	*dn*					
	group	*acdgroup*	all				full
			subgroup	*subgrp*			
	len	*site*	*frame*	*bay*	*drawer*	*card*	
	posid	*posid*					

Parameters and variables	Description
acdgroup	This variable identifies the ACD group.
all	This parameter displays the supervisors in all the ACD groups.
all all	This parameter displays the supervisors in all the subgroups of all the ACD groups.
bay	This variable specifies the bay number. The valid entry range is 0-1.
brief	This parameter displays the login ID of the agent and the agent position where the agent is currently logged in. Omitting this entry forces the system to default to display the current login ID and position when mode is set to brief, which is the setting when entering the ACDSHOW directory.
card	This variable specifies the card number. The valid entry range is 0-32.
clli	This parameter displays information about the CLLI in the ACD group.
-continued-	

---

supervisor (continued)

supervisor command parameters and variables (continued)	
Parameters and variables	**Description**
dn	This parameter displays the supervisor associated with a particular DN.
dn	This variable identifies the DN. Directory numbers are displayed as seven or ten-digit numbers.
drawer	This variable specifies the drawer number. The valid entry range is 0-19.
frame	This variable specifies the frame number. The valid entry range is 0-2047.
full	This parameter displays detailed information about all ACD supervisors.
group	This parameter displays a list of supervisors for a particular ACD group.
len	This parameter displays the supervisor associated with a particular LEN.
num	This variable further specifies the CLLI. The valid entry range is 0-9999.
posid	This parameter displays the supervisor associated with a particular position.
posid	This variable is the ID of the ACD supervisor position.
site	This variable specifies the site location.
subgroup	This parameter displays a list of supervisors for a particular ACD subgroup.
subgrp	This variable identifies the ACD subgroup. The valid entry range is 0-255.
End	

Qualifications

None

Examples

The following table provides examples of the supervisor command.

Examples of the supervisor command	
Example	**Task, response, and explanation**
supervisor all all ↵	
	Task: Display a list of supervisors for all the ACD groups.
	Response:
| | ```
Supervisors for ACD Group PLAN1
    Supervisor Position ID:   9999
    ACD Subgroup:             1
    Supervisor Position ID:   9997
    ACD Subgroup:             2
    Supervisor Position ID:   6000
    ACD Subgroup:             3
    ACD Subgroup:             4
    No Supervisor for Subgroup
Supervisors for ACD Group PLAN2
    Supervisor Position ID:   9998
    ACD Subgroup:             1
    Supervisor Position ID:   9996
    ACD Subgroup:             2
``` |
	Explanation: This command displays a list of supervisors for all the ACD groups.
supervisor dn 7226000 ↵ *where* 7226000 specifies the directory number	
	Task: Display the supervisor associated with a DN.
	Response:
| | ```
Supervisor Position ID:   6000
ACD Group:                PLAN1
ACD Subgroup:             2
``` |
| | **Explanation:** This command displays the supervisor associated with DN 7226000. |
| **-continued-** | |

supervisor (continued)

Examples of the supervisor command (continued)	
Example	**Task, response, and explanation**
supervisor group nttacd1 subgroup 1 ↵ *where* nttacd1 specifies the ACD group 1 specifies the subgroup	

	Task:	Display a list of supervisors for a particular ACD subgroup.
	Response:	`Supervisor Position ID: 1129` `ACD Subgroup:          1`
	Explanation:	This command displays a list of supervisors for the ACD group nttacd1 subgroup 1.

supervisor len ↵
where

         specifies the site
         specifies the frame
         specifies the bay
         specifies the drawer
         specifies the card

	Task:	Display the supervisor associated with a particular LEN.
	Response:	`this command not documented in 509 on-line text`
	Explanation:	This command displays the supervisor associated with a particular LEN.

supervisor posid ↵
where

         specifies the position ID of the ACD supervisor

	Task:	Display the supervisor associated with a particular position.
	Response:	`this command not documented in 509 on-line text`
	Explanation:	This command displays the supervisor associated with a particular position.
	End	

supervisor (continued)

Responses

The following table provides explanations of the responses to the supervisor command.

Responses for the supervisor command
MAP output   **Meaning and action**
command "all all" not documented in on-line text.
Meaning: The system displays the supervisors in all the subgroups of all the ACD groups. **Action:**   None
DN must be an ACD Incalls or Trunk DN
Meaning: You entered a directory number that is not in the appropriate table. **Action:**   Enter an appropriate DN to continue or abort to cancel.
EITHER incorrect optional parameter(s) OR too many parameters.
Meaning: You entered the command incorrectly. **Action:**   Check the command syntax and reenter the command.
*** Invalid ACD group name   Valid names are: <acdgroup>
Meaning: You entered an invalid group name or left out the group parameter. **Action:**   Enter an appropriate group name to continue or abort to cancel.
Invalid CLLI: <clli?> STRING Enter: <clli?><num?>[<Form?>]
Meaning: You entered an invalid CLLI. **Action:**   Enter an appropriate CLLI to continue or abort to cancel.
Invalid symbol: <Which_Groups?> {ALL,   GROUP <Acd_grp?> STRING}
Meaning: You entered an invalid parameter. **Action:**   Enter the appropriate parameter to continue or abort to cancel.
-continued-

supervisor (continued)

Responses for the supervisor command (continued)	
MAP output	**Meaning and action**
`Out of range:`	
	Meaning: You entered a number that was too large or too small.
	Action: Reenter the appropriate number to continue or abort to cancel.

`Supervisor LEN and DN:  LEN HOST len DN dn`
`ACD Group:              acdgroup`
`ACD Subgroup:           subgrp`	

	Meaning: The system displays information on the supervisor position associated with the specified position ID.
	Action: None

`Supervisors for ACD Group acdgroup`
`Supervisor Position ID:   posid`
`ACD Subgroup:             subgrp`
`Supervisor Position ID:   posid`
`ACD Subgroup:             subgrp`
`Supervisor Position ID:   posid`
`ACD Subgroup:             subgrp`
`ACD Subgroup:             posid`
`No Supervisor for Subgroup`	

	Meaning: The system displays a list of supervisor positions associated with the specified group. The parameters of this display are defined in the parameter list for command syntax.
	Action: None

`Supervisor Position ID:   posid`
`ACD Group:                acdgroup`
`ACD Subgroup:             subgrp`	

	Meaning: The system displays information on the supervisor position associated with the specified DN.
	Action: None
-continued-	

supervisor (end)

Responses for the supervisor command (continued)	
MAP output	**Meaning and action**
`The directory number should consist of 7 digits.`	
	Meaning: You entered a ten-digit number or characters.
	Action: Reenter an appropriate DN to continue or abort to cancel.
`This Supervisor POSID has not been assigned.`	
	Meaning: You entered the command correctly but there is no data available.
	Action: None
`Wrong type:`	
	Meaning: You entered a number in an alphabetic field or characters in a numeric field.
	Action: Check the command syntax and reenter the appropriate value to continue or abort to cancel.
End	

tabentry

Function

Use the tabentry command to display the routing information found in the specified entry within Tables IBNRTE, IBNRT2, IBNRT3, IBNRT4, OFRT OFR2, OFR3, or OFR4, as well as the audio information found in Table AUDIO.

tabentry command parameters and variables		
Command	**Parameters and variables**	
tabentry	audio ibnrte ibnrt2 ibnrt3 ibnrt4 ofrt ofr2 ofr3 ofr4	*audio_group* *index*

Parameters and variables	**Description**
audio	This parameter displays audio information found in Table AUDIO.
audio_group	This variable identifies the group datafilled in Table AUDIO. Possible groups range from AUDIO1 to AUDIO512.
ibnrt2	This parameter displays routing information found in Table IBNRT2.
ibnrt3	This parameter displays routing information found in Table IBNRT3.
ibnrt4	This parameter displays routing information found in Table IBNRT4.
ibnrte	This parameter displays routing information found in Table IBNRTE.
index	This variable indicates which entry in the table to display. The valid entry range is 0-1023.
ofr2	This parameter displays routing information found in Table OFR2.
ofr3	This parameter displays routing information found in Table OFR3.
ofr4	This parameter displays routing information found in Table OFR4.
ofrt	This parameter displays routing information found in Table OFRT.

tabentry (continued)

Qualification

This command displays the first element in a route list.

Examples

The following table provides examples of the tabentry command.

Examples of the tabentry command	
Example	**Task, response, and explanation**
tabentry audio audio1 ↵ *where* *audio1* specifies the audio group	
	Task: Display the audio information in an audio group.
	Response: `Ann:  ANN1   Silence: 25   Ann: ANN2` `Music:  MUSIC1 30   Ann: ANN2   Music: MUSIC1 0`
	Explanation: This command displays the audio information in audio group AUDIO1 of Table AUDIO. The caller listening to this call treatment will hear announcement 1 followed by 25 seconds of silence, followed by announcement 2, followed by 30 seconds of music, followed again by announcement 2 and, finally, continuous music.
tabentry ibnrte 12 ↵ *where* 12 specifies the index	
	Task: Display routing information in Table IBNRTE.
	Response: Currently not available
	Explanation: This command displays routing information found in index 12 of Table IBNRTE.
-continued-	

tabentry (continued)

Examples of the tabentry command (continued)
Example   **Task, response, and explanation**
tabentry ofrt 128 ↵ *where* 128   specifies the index

	Task:	Display routing information in Table OFRT.

	Response:	Depending on the information in the entry, the system response is:
		`Route:                    OFRT 128`
		`Routes To 613 722 1525`
		or
		`Route:                    OFRT 128`
		`Conditionally Routes to OFRT 135`
		or
		`Route:                    OFRT 128`
		`Routes To Virtual Trunk Group VIRTTRKGRP4`
		or
		`Route:                    OFRT 128`
		`Route Information is Non-Displayable`

	Explanation:	This command displays the routing information found in entry 128 of Table OFRT.

End

Responses

The following table provides explanations of the responses to the tabentry command.

Responses for the tabentry command
MAP output   **Meaning and action**

`*** Invalid AUDIO group name.`
`    Valid names are:`
`<Audio_group?> <Audio_group?> <Audio_group?>`
`Invalid Audio Group: <Audio_group?> STRING`

Meaning: You entered an invalid audio group name.	
Action: Enter an audio group from the list to continue or abort to cancel.	

-continued-

tabentry (end)

Responses for the tabentry command (continued)	
MAP output	**Meaning and action**
`Invalid symbol: <Table?>`	
	Meaning: You entered an invalid table name.
	Action: Enter the correct table name to continue or abort to cancel.
`Out of range: <Index?> {0 to 1023}`	
	Meaning: You entered an invalid index value.
	Action: Enter the correct index to continue or abort to cancel.
`Wrong type: <Index?> {0 to 1023}`	
	Meaning: You entered an alphabetic index value.
	Action: Enter the correct numeric index to continue or abort to cancel.
End	

threshold

Function

Use the threshold command to display threshold limits for a particular Automatic Call Distribution (ACD) group, or for all the ACD groups assigned to you.

threshold command parameters and variables		
Command	**Parameters and variables**	
threshold	all	
	group	*acdgroup*
Parameters and variables	**Description**	
acdgroup	This variable identifies the ACD group.	
all	This parameter displays the threshold limits of all the ACD groups.	
group	This parameter displays threshold limits for a particular ACD group.	

Qualifications

None

threshold (continued)

Examples

The following table provides examples of the threshold command.

Examples of the threshold command	
Example	**Task, response, and explanation**
threshold all ↵	

	Task:	Display the threshold limits of all the ACD groups.
	Response:	`==================================================`
		`Thresholds for ACD Group:  NTTACD1`
		`  Ring Threshold:                    0 Secs`
		`  Max Call Queue Wait Time:          1800 Secs`
		`  Max Call Queue Size:               10`
		`  Max Call Transfer Queue Size:    10`
		`  Recorded Announcement Threshold: 0 Secs`
		`  Multi-Stage Queue Status Thresholds`
		`       T1: 2   T2: 3   T3: 4`
		`==================================================`
		`Thresholds for ACD Group:  NTTACD2`
		`  Ring Threshold:                    12 Secs`
		`  .`
		`  .`
		`  .`
		`  Recorded Announcement Threshold: 0 Secs`
		`  Multi-Stage Queue Status Thresholds`
		`       T1: 2   T2: 3   T3: 4`
	Explanation:	This command displays the threshold limits of all the ACD groups.
-continued-		

threshold (continued)

Examples of the threshold command (continued)	
Example	**Task, response, and explanation**
threshold group plan1 ↵ *where* plan 1 identifies the ACD group	

	Task:	Display the threshold limits for a particular ACD group.
|  | **Response:** | ```
Thresholds for ACD Group:  PLAN 1
Ring Threshold:            20 sec
Max Call Queue Wait Time:  30 sec
Max Call Queue Size:       10
Recorded Announcement Threshold: 12 sec
``` |
| | | If the ACDGRP Multi-Stage Queue Status (MSQS) Thresholds option has been datafilled, the system response is: |
|  |  | ```
Thresholds for ACD Group:  PLAN1
Ring Threshold:            20 sec
Max Call Queue Wait Time:  30 sec
Max Call Queue Size:       10
Multi-Stage Queue Status Thresholds
  T1:   105   T2:   209   T3:   1800
``` |
| | | If the group has the ACD Call Transfer (ACDXFER) option assigned to it, the output contains the following additional value: |
|  |  | ```
Max Call Transfer Queue Size:   xx
``` |
| | **Explanation:** | This command displays the threshold limits for the ACD group plan1. |
| | **End** | |

threshold (end)

Responses

The following table provides explanations of the responses to the threshold command.

Responses for the threshold command	
MAP output	**Meaning and action**
``` *** Invalid ACD group name Valid names are: <acdgroup> ```	
	Meaning: You entered an invalid group name or left out the group parameter.
	Action: Enter an appropriate group name to continue or abort to cancel.
``` Invalid symbol: <Which_Groups?> {ALL, GROUP <Acd_grp?> STRING} ```	
	Meaning: You entered an invalid parameter.
	Action: Enter the appropriate parameter to continue or abort to cancel.

throute

Function

Use the throute command to display the threshold route for an Automatic Call Distribution (ACD) group. The threshold route is the destination to which overflowed calls for an ACD group are directed if the queue is full or the maxwait time has been exceeded.

throute command parameters and variables	
Command	**Parameters and variables**
throute	all group *acdgroup*

Parameters and variables	**Description**
acdgroup	This variable identifies the ACD group.
all	This parameter displays the threshold routes for all the ACD groups.
group	This parameter displays the threshold route for a particular ACD group.

Qualifications

None

Examples

The following table provides examples of the throute command.

Examples of the throute command		
Example	**Task, response, and explanation**	
throute all ↵		
	Task:	Display the threshold routes for all ACD groups.
	Response:	THROUTE For Group: PLAN1 Route: OFRT 167 Routes To A Directory Number THROUTE For Group: PLAN2 Route: IBNRTE 290 Routes Information Is Non-Displayable THROUTE For Group: ACDMIS1 Route: IBNRTE 1 Routes To Trunk Group OGDP
	Explanation:	This command displays the threshold routes for all ACD groups.
	-continued-	

throute (end)

Examples of the throute command (continued)	
Example	**Task, response, and explanation**
throute group plan1 ↵ *where* plan 1 identifies the ACD group	
	Task: Display the threshold route for an ACD group.

	Response: `THROUTE For Group:        PLAN1`
	`Route:                    OFRT 167`
	`Routes To A Directory Number`

	Explanation: This command displays the threshold route for ACD group plan1.
End	

Responses

The following table provides explanations of the responses to the throute command.

Responses for the throute command	
MAP output	**Meaning and action**
`*** Invalid ACD group name` `    Valid names are: <acdgroup>`	
	Meaning: You entered an invalid group name or left out the group parameter.
	Action: Enter an appropriate group name to continue or abort to cancel.
`Invalid symbol: <Which_Groups?> {ALL,` `                    GROUP <Acd_grp?> STRING}`	
	Meaning: You entered an invalid parameter.
	Action: Enter the appropriate parameter to continue or abort to cancel.

validaudio

Function

Use the validaudio command to display a list of the audio groups assigned to give recorded announcements to callers in an incoming call queue. This information can be displayed for one or all Automatic Call Distribution (ACD) groups in an administration group. This command is valid only if the AUDIO option is datafilled in Table ACDRTE.

validaudio command parameters and variables		
Command	**Parameters and variables**	
validaudio	all	
	group	*acdgroup*
Parameters and variables	**Description**	
acdgroup	This variable identifies the ACD group.	
all	This parameter displays the data for all the ACD groups assigned to you.	
group	This parameter displays the valid AUDIO groups for an ACD group.	

Qualifications

None

validaudio (continued)

Examples

The following table provides examples of the validaudio command.

Examples of the validaudio command	
Example	**Task, response, and explanation**
validaudio all ↵	
	Task: Display the valid audio groups for all the ACD groups in your administration group.

	Response: `Valid AUDIO groups for group: ACDGRP1`
	    `AUDIO1`
	    `AUDIO2`
	    `AUDIO3`
	`Valid AUDIO groups for group: ACDGRP2`
	  `Group ACDGRP2 has no audio restrictions.`
	`Valid AUDIO groups for group: ACDGRP3`
	
	    `AUDIO1`
	    `AUDIO2`

	Explanation: This command displays the valid audio groups for all the ACD groups in your administration group.
validaudio group acdgrp2 ↵ *where* acdgrp2 identifies the ACD group	
	Task: Display a list of the valid audio groups for an ACD group.

	Response: `Valid AUDIO groups for group: ACDGRP2`
	
	    `AUDIO1`
	    `AUDIO2`
	    `AUDIO3`
	    `AUDIO4`
	    `AUDIO5`

	Explanation: This command displays a list of the valid audio groups for ACD group acdgrp2.

validaudio (end)

Responses

The following table provides explanations of the responses to the validaudio command.

Responses for the validaudio command	
MAP output	**Meaning and action**

`Valid AUDIO GROUPS for group acdgrp:`
`audgrp`
`audgrp`	

	Meaning: You executed the command correctly.
	Action: None

`*** Invalid ACD Group name,`
`Valid names are:`
`<list of acdgroup names>`	

	Meaning: You entered an ACD group that is not assigned to an administration group. A list of valid ACD groups is displayed.
	Action: Enter a valid ACD group name.
`Valid AUDIO GROUPS for group acdgroup:` `Group acdgroup has no audio restrictions.`	
	Meaning: You entered a group that is not assigned a audio option in the Table ACDRTE.
	Action: None
`Group acdgroup is outside of user's ADMIN group.`	
	Meaning: You entered an ACD group that is not assigned to your administration group.
	Action: Reenter the command using a valid ACD group name.

validroutes

Function

Use the validroutes command to allow senior supervisors to display a list of valid threshold and night service routes for one or all Automatic Call Distribution (ACD) groups in their administration groups. This command is valid only if the TABEN option is datafilled in Table ACDRTE.

validroutes command parameters and variables		
Command	**Parameters and variables**	
validroutes	all	
	group	*acdgroup*

Parameters and variables	Description
acdgroup	This variable identifies the ACD group.
all	This parameter displays the valid routes for each of the ACD groups assigned to you.
group	This parameter displays the valid routes for an ACD group.

Qualifications

None

validroutes (continued)

Examples

The following table provides examples of the validroutes command.

Examples of the validroutes command	
Example	**Task, response, and explanation**
validroutes all ↵	
	Task: Display the valid routes for all ACD groups in your administration group.

	Response: Valid nightservice/threshold routes for group ACDGRP1:
	
	          OFRT 100
	          OFRT 101
	          IBNRTE 52
	          IBNRTE 53
	Valid nightservice/threshold routes for group ACDGRP2:
	
	Group ACDGRP2
	has no routing restrictions.
	
	Valid nightservice/threshold routes for group ACDGRP3:
	          OFRT 101
	          OFRT 102
	          IBNRTE 53
	          IBNRTE 54

	Explanation: This command displays the valid routes for all the ACD groups in your administration group.
-continued-	

validroutes (continued)

Examples of the validroutes command (continued)	
Example	**Task, response, and explanation**
validroutes group acdgrp2 ↵ *where* acdgrp2 identifies the ACD group	

	Task:	Display a list of the valid night service or threshold routes for an ACD group.

	Response:	```Valid nightservice/threshold routes for group ACDGRP2:```
		
		``` OFRT 100```
		``` OFRT 101```
		``` OFRT 102```
		``` IBNRTE 52```
		``` IBNRTE 53```

	Explanation:	This command displays a list of the valid night service or threshold routes for the ACD group acdgrp2.
End		

Responses

The following table provides explanations of the responses to the validroutes command.

Responses for the validroutes command	
MAP output	**Meaning and action**

```*** Invalid ACD Group name,```
```Valid names are:```
```<list of acdgroup names>```	

	Meaning:	You entered an ACD group that is not assigned to an administration group. A list of valid ACD groups is displayed.
	Action:	Enter a valid ACD group name.
-continued-		

validroutes (end)

Responses for the validroutes command (continued)	
MAP output	**Meaning and action**
`Valid nightservice/threshold routes for group acdgroup:` `Group acdgroup has no routing restrictions.`	
	Meaning: You entered an ACD group with no routing restrictions. The TABEN option is not assigned in Table ACDRTE for this ACD group.
	Action: None
`Group acdgroup is outside of user's ADMIN group.`	
	Meaning: You entered an ACD group that is not assigned to your administration group.
	Action: Reenter the command using a valid ACD group name.
End	

AFTCI level commands

Use the AFTCI level of the MAP to control and monitor the automatic file transfer (AFT) system.

Accessing the AFTCI level

To access the AFTCI level, enter the following command from the CI level:

aft ↵

AFTCI commands

The commands available at the AFTCI MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

AFTCI commands	
Command	Page
copyaft	A-235
delaft	A-241
help	A-247
queryaft	A-251
quit	A-257
resetovr	A-261
resetpft	A-265
setaft	A-269
setovr	A-273
startaft	A-277
stopaft	A-279

copyaft

Function

Use the copyaft command to copy a Device Independent Recording Package (DIRP) transfer file to a tape drive.

copyaft command parameters and variables	
Command	**Parameters and variables**
copyaft	*fn* *device*

Parameters and variables	**Description**
device	This variable specifies the device name of the tape drive to which the file is copied.
fn	This variable specifies the full DIRP source file name of the file to be copied.

Qualifications

The copyaft command is qualified by the following exceptions, limitations, and restrictions:

- This command does not mount the tape drive. The tape drive must be mounted manually before this command is executed.
- This command does not allow an active file to be copied.
- This command only allows DIRP files to be copied.

Example

The following table provides an example of the copyaft command.

Example of the copyaft command	
Example	**Task, response, and explanation**
copyaft u880531141059occ t0 ↵ *where* u880531141059occ specifies the full DIRP file name of the file to be copied t0 specifies the tape drive device name	

	Task:	Copy a transfer file to a tape drive.
	Response:	`File Copied Successfully - check logs please`
	Explanation:	The system successfully copied a transfer file to a tape drive. Once a file is transferred and copied, it becomes a processed file and no longer appears in the AFT system directory.

copyaft (continued)

Responses

The following table provides explanations of the responses to the copyaft command.

Responses for the copyaft command	
MAP output	**Meaning and action**
`Copy completed`	
	Meaning: Copying the file to tape completed normally. The AFT system directory is updated to show a copy state of complete.
	Action: None
`Could not close source file`	
	Meaning: The file copied to tape but the file on tape drive could not close. A SWERR is generated.
	Action: Call the next level of support and notify them that files are not being closed.
`Could not close target file`	
	Meaning: The file copied to tape but the file on tape drive could not close. A SWERR is generated.
	Action: Call the next level of support and notify them that files are not being closed.
`Could not create tape file`	
	Meaning: The command could not open a file on the tape drive to which the DIRP file is to be copied. The file is not copied to tape and a SWERR is generated.
	Action: Verify that the tape drive still is online. If the tape drive is online and this response continues to display, contact the next level of support.
-continued-	

Responses for the copyaft command (continued)

MAP output   **Meaning and action**

```
Error getting from file
Copy aborted
```

Meaning: An error occurred while copying the file to tape. Copying to tape is aborted and a SWERR is generated.

Action: Call the next level of support and receive recovery instructions. When the problem is resolved, erase the partial tape file and recopy the file to tape.

```
Error getting source VOLUME INFO
No action taken - request aborted
```

Meaning: The system cannot determine the location of the disk volume housing the file to be copied. The file is not copied to tape and a SWERR is generated.

Action: Check the state of the disk volume, recover the disk volume if possible, and try again to copy the file to tape. If the disk volume cannot be recovered, contact the next level of support.

```
Error getting target VOLUME INFO
Request Aborted.
```

Meaning: The system cannot determine the location of the disk volume housing the file to be copied. The file is not copied to tape and a SWERR is generated.

Action: Verify that the tape drive is online and mounted. If the tape drive is online, mounted, and this response continues to display, contact the next level of support.

```
Error putting to tape
Copy aborted
```

Meaning: An error occurred while writing the file to tape. Copying to tape is aborted and a SWERR is generated.

Action: Look at the text explaining the reason for the error and take appropriate action. If the tape is full, erase the partial file from the tape and place a new tape on the drive. Recopy the file to the new tape. If you cannot recover from the error, contact the next level of support.

-continued-

copyaft (continued)

Responses for the copyaft command (continued)	
MAP output	**Meaning and action**
`Invalid file` `Request aborted`	
	Meaning: The SST system marked the file as invalid. The file is not copied to tape.
	Action: Call the next level of support.
`Source device must be a disk drive` `Request aborted`	
	Meaning: The name of a file you entered is not located on a disk drive. The file is not copied to tape.
	Action: Check the name of the file to be copied and reenter the command with the correct file name.
`System error` `Request aborted`	
	Meaning: The system could not obtain information on the file to be copied. The file is not copied to tape.
	Action: Call the next level of support.
`Target device is not a tape drive` `Request aborted.`	
	Meaning: The device name you entered is not a tape drive. The file is not copied to tape.
	Action: Determine the device name of the tape drive to receive the file and reenter the command with the correct device name.
`The disk file cannot be accessed` `Request aborted`	
	Meaning: The system cannot access the disk drive on which the file is located because the disk drive is not in an appropriate state. The file is not copied to tape and a SWERR is generated.
	Action: Check the state of the disk drive and recover the disk drive if possible. If the disk drive cannot be recovered (or if the disk appears to be in a ready state) and this response continues to display, contact the next level of support.
-continued-	

copyaft (end)

Responses for the copyaft command (continued)	
MAP output	**Meaning and action**
`This is the DIRP ACTIVE file` `It cannot be copied to tape until it is rotated and closed`	
	Meaning: The file to be copied is the DIRP active file. The file is not copied to tape because it is not closed.
	Action: Wait until the DIRP file has been rotated and closed before reentering the command.
`Unable to allocate an event` `Safe store directory copy state may not be accurate`	
	Meaning: The system could not allocate an SST event for copying the file to tape. The file is not copied to tape.
	Action: Retry the command a few times. If this response continues to display, contact the next level of support.
End	

delaft

Function

Use the delaft command to delete a file from the AFT system "safe store" directory and the DIRPHOLD table. This command does not erase the Device Independent Recording Package (DIRP) file. When the command is executed, the file is taken out of the DIRPHOLD table and renamed from a DIRP "U" file to a "P" file.

If this command is used to delete an unprocessed file (a file which has not been transferred and copied), a system message warns that the file is unprocessed and you are prompted to confirm whether or not the file should be deleted.

delaft command parameters and variables	
Command	**Parameters and variables**
delaft	*filename*
Parameters and variables	**Description**
filename	This variable specifies the file to be deleted. This is the file name as it appears in the AFT system directory.

Qualifications

The delaft command is qualified by the following exceptions, limitations, and restrictions:

- The delaft command cannot be used to delete an actively-transferring file, a file being copied, or a file that is the DIRP active file.

> **CAUTION**
> **Loss of data can occur.**
> Use caution when deleting unprocessed files. If an unprocessed file is deleted, loss of data can occur.

If an unprocessed file is deleted, loss of data can occur.

Example

The following table provides an example of the delaft command.

delaft (continued)

Example of the delaft command

Example	Task, response, and explanation

---

delaft   **u880531141059occ** ↵
where

u880531141059occ   specifies the file to be deleted

Task:	Delete a file from the AFT system directory and the DIRPHOLD table.
| **Response:** | ```
This file has not yet been completely processed
   Do you still want to delete it?
   YES or NO:
>YES
   File will be renamed to a P (PROCESSED) file
   Then it will be deleted from DIRPHOLD and from
   the SAFE STORE Directory
   Do you want to continue?
   YES or NO:
>YES
``` |
| **Explanation:** | When the command executes, the specified file is taken out of the DIRPHOLD table and renamed from a DIRP "U" file to a "P" file. |

Responses

The following table provides explanations of the responses to the delaft command.

delaft (continued)

Responses for the delaft command	
MAP output	**Meaning and action**
Could not close source file	
	Meaning: The system could not close the source file after it was renamed and deleted. The file is deleted, but left in an open state. A SWERR is generated.
	Action: Contact the next level of support and notify them that files are not closing properly.
Error accessing DIRP Delete command may be retried The current request has been aborted	
	Meaning: DIRP is not allowing the command to change the state of the file so that it can be deleted. The file is not deleted from the AFT system directory or the DIRPHOLD table.
	Action: Reissue the command several times. If the command continues to fail, contact the next level of support.
Error deleting file from table DIRPHOLD	
	Meaning: DIRP is not allowing the command to delete the file from the DIRPHOLD table. The file is not deleted from the AFT system directory or the DIRPHOLD table.
	Action: Contact the next level of support.
Error getting file information from Safe Store Request aborted or System error Request aborted	
	Meaning: These two responses have the same meaning. SST failed to provide the command with the information necessary to delete the file. The file is not deleted from the AFT system directory or the DIRPHOLD table. A SWERR is generated.
	Action: Contact the next level of support.
-continued-	

delaft (continued)

Responses for the delaft command (continued)	
MAP output	**Meaning and action**
`File in use` `Retry delete command later`	
	Meaning: The file you want to delete currently is opened by another user. The file cannot be deleted. The file is not deleted from the AFT system directory or the DIRPHOLD table.
	Action: Reenter the command after the file has been closed.
`Invalid file` `Request aborted`	
	Meaning: The command received and invalid file responded when it attempted to open the file to be deleted. The file is not deleted from the AFT system directory or the DIRPHOLD table.
	Action: Contact the next level of support.

`This file is currently being copied`
`It may not be deleted until its copy completes`
`Request aborted`	

	Meaning: The file you want to delete is being copied and cannot be deleted. The file is not deleted from the AFT system directory or the DIRPHOLD table.
	Action: Wait until the file copies completely and reenter the command.

`This file is currently being transferred.`
`It may not be deleted until its transfer completes.`
`Request aborted`	

	Meaning: The file name you entered is in the process of being transferred. The file is not deleted from the AFT system directory or the DIRPHOLD table.
	Action: Wait until the file transfers completely and reenter the command.
-continued-	

delaft (end)

Responses for the delaft command (continued)
MAP output   **Meaning and action**

```
This is the DIRP ACTIVE file
It cannot be deleted until it is rotated and closed and appears in
DIRPHOLD
Request aborted
```

Meaning:	The file you want to delete is the DIRP active file; you cannot delete the DIRP active file. The file is not deleted from the directory or the DIRPHOLD table.
Action:	Determine whether the file should really be deleted. If the the file does need to be deleted, it should be rotated out of the active DIRP position. When the file appears in the DIRPHOLD table after the rotate, delete the file.

```
Unable to allocate an event
Safe Store Directory Access may be affected
```

Meaning:	The system could not allocate an SST event in order to delete the file. The file is not deleted from the directory or the DIRPHOLD table. The SST directory could be corrupted.
Action:	Contact the next level of support.

End

help

Function

Use the help command to receive online documentation for the AFTCI directory.

help command parameters and variables	
Command	**Parameters and variables**
help	aft *command_nam*
Parameters and variables	**Description**
aft	This parameter produces summary documentation for the commands in the AFTCI directory.
command_nam	This variable specifies a valid AFTCI directory command name. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

help (continued)

Example of the help command	
Example	**Task, response, and explanation**
help aft ↵	
	Task: Access online documentation.

	Response: This CI level contains commands used to query and manipulate the AFT (Automatic File Transfer) System. AFT automatically transfers files recorded by DIRP. The following is a list of commands contained in this CI level:
	
	QUERYAFT - Query information about AFT files.
	SETAFT - Set the next AFT file to transfer.
	SETOVR - Set an Override file transfer.
	RESETPFT - Copy an AFT file to tape.
	RESETOVR - Resets the Override file pointer.
	COPYAFT - Copy an AFT file to tape.
	DELAFT - Delete a file from DIRPHOLD and the
	             directory.
	STARTAFT - Starts the AFT system transferring
	                 files.
	STOPAFT - Stops the AFT system transferring
	                 files.
	QUIT - Quit out of the CI level.

	Explanation: This example typifies a response for the help command string. (The description for the quit command is misleading. The quit command takes you out of the AFTCI level and returns you to the CI level only.)

Responses

The following table provides explanations of the responses to the help command.

Responses for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None
-continued-	

help (end)

Responses for the help command (continued)	
MAP output	**Meaning and action**
NO COMMAND IN LINE	
	Meaning: The command you entered is spelled incorrectly or does not exist. For example, entering the help aftci command string instead of the help aft command string produces this message.
	Action: Reenter the command using a valid command name.
Undefined command "HELP"	
	Meaning: Since there is no default parameter for the help command, entering the help command without additional parameters produces this message.
	Action: None
End	

queryaft

Function

Use the queryaft command to query DIRP files in the AFT system. When this command is executed, a message is sent to the AFT software asking it to request a new directory from the SST software. The result is a directory of the latest files.

Processed files are files which have been both transferred and copied to tape. These files are deleted from the directory automatically. Once the file has been transferred and copied to tape, the file no longer is listed in the AFT system directory.

queryaft command parameters and variables		
Command	**Parameters and variables**	

queryaft	*session name*	all act com err *fn* pen pft sys	short long

Parameters and variables	**Description**
all	This default parameter queries all files in the AFT system. Either enter the all parameter or omit this entry to query all files in the AFT system.
short	This default parameter causes the format of the display to include a one-line summary of information for each file. Either enter the short parameter or omit this entry to receive display data in summary format.
act	This parameter queries information for the AFT active file.
com	This parameter queries information for the AFT files which have completed transferring.
err	This parameter queries information for the AFT files which are in an error state.
fn	This variable specifies the AFT file name for which information displays.
long	This parameter causes the format of the display to include a longer, more complete set of information on each file.
-continued-	

queryaft (continued)

queryaft command parameters and variables (continued)	
Parameters and variables	**Description**
pen	This parameter queries information for all AFT files which are not transferred.
pft	This parameter queries information for the partial file transfer (PFT) files only.
session name	This variable specifies the name of the AFT session for which the directory is being requested. This is the name datafilled as the key in the Table GASINFO.
sys	This parameter queries information for the AFT system without displaying a directory.
End	

Qualification

The long format of the queryaft command takes longer to produce than the short format. It is better to use this option only when a smaller number of files are to be displayed.

Examples

The following table provides examples of the queryaft command.

Examples of the queryaft command
Example   **Task, response, and explanation**
queryaft   aft1 all short ↵ *where* aft1   specifies the name of the AFT session for which the directory is being requested

> **Task:**   Query files in the AFT system.
>
> **Response:**

```
==AFT SYSTEM: STARTED -AFT STATE: SENDING -AFT CLASS: GBKGCLASS==

    FILE NAME           FILE      LAST      XFER      COPY      START
                        SIZE      ACK       STATE     STATE     TIME
    ----------------  --------  --------  --------  --------  --------
O->U880531123056OCC      5000      5000  COMPLETE  PENDING   31/22/33
A->U880531125057OCC      4890      2380  ACTIVE    PENDING   31/22/45
N->U880531133558OCC     12900         0  PENDING   PENDING
    U880531141059OCC    12900         0  PENDING   PENDING
    A880531143360OCC    12839         0  PENDING   PENDING
```

> **Explanation:** The system displays information on all files in the AFT system using short format. When the queryaft command is executed, a message is sent to the AFT software requesting a directory. The AFT system makes a directory request to SST. When SST receives the latest directory, it sends in the queryaft command software.
>
> The first line of the display provides the AFT system status and state. If the system is started, files begin transferring and continue to do so. If the system is stopped, files do not continue to transfer. Starting and stopping the AFT system is controlled using the AFTCI directory commands startaft and stopaft.
>
> Some of the file names in the response display are preceded by N->, A->, and O-> symbols. These flags point to the next file to transfer, the actively-transferring file, and the override file respectively.

-continued-

queryaft (continued)

Examples of the queryaft command (continued)	
Example	**Task, response, and explanation**

queryaft aft1 long ↵
where

aft1   specifies the name of the AFT session for which the directory is being requested

u880531133558occ   specifies the file name

Task: Display information on a specified file in long format.

Response:
```
==AFT SYSTEM: STARTED -AFT STATE: SENDING
                              -AFT CLASS: GBKGCLASS==

N->FILE NAME:    U880531133558OCC
 Far End Name:
Subsystem:               OCC
session:                 AFT1
Last ACK:                  0
File Size:             12900
XFER State:          PENDING
COPY State:          PENDING
Start Time:
Stop Time:
DIRPHOLD ID:              94
Retry:                     0
```

Explanation: The system displays information on a specific file in the AFT system in long format. When the queryaft command is executed, a message is sent to the AFT software requesting a directory. AFT makes a directory request to SST. When SST receives the latest directory, it sends in the queryaft software.

The long format provides all the information displayed using the short format as well as the DIRP subsystem, the session, the retry count, the stop timer, and the DIRPHOLD ID.

-end-

queryaft (continued)

Responses

The following table provides explanations of the responses to the queryaft command.

Responses for the queryaft command	
MAP output	**Meaning and action**
`AFT is not connected to SST!`	
	Meaning: The connection between the AFT system and SST is lost.
	Action: The AFT system attempts to bring up the connection every minute. Check the AFT system logs and reenter the command when the connection is reestablished. If all attempts to connect continue to fail, contact the next level of support.
`Bad AFT message received`	
	Meaning: The command received a corrupt message.
	Action: Examine the AFT system logs. If this response happens frequently, contact the next level of support.
`Could not contact AFT system` `reason = (integer)`	
	Meaning: This response displays if the command cannot contact the AFT system.
	Action: Contact the next level of support.
`Could not obtain directory from SST - See AFT logs`	
	Meaning: This response displays if an error occurs while obtaining a directory.
	Action: Examine the AFT system logs. If this response happens frequently, contact the next level of support.
`Directory not available at this time` `Try again in a few minutes`	
	Meaning: A restart reload occurred within the past five minutes and the directory has not stabilized.
	Action: Wait a few minutes and try to query a directory again.
-continued-	

queryaft (end)

Responses for the queryaft command (continued)	
MAP output	**Meaning and action**
`Parameter 3 – ( ) is invalid`	
	Meaning: An invalid parameter was entered.
	Action: Examine the command and reenter the correct parameter.
`<session name> is invalid`	
	Meaning: An invalid AFT session name has been entered.
	Action: Refer to the GASINFO table and retry the command with the correct session name.
`There are no files in the AFT directory`	
	Meaning: This response displays if there are no files in the AFT system directory.
	Action: None
`There is no file transferring at this time`	
	Meaning: This response displays if you request information for the actively-transferring file and no file is transferring.
	Action: None
End	

quit

Function

Use the quit command to exit the AFTCI directory.

quit command parameters and variables	
Command	**Parameters and variables**

quit	⌐ *1 level* ¬
	   all
	   name
	 ∟ *n_levels* ⌐

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)	
Example	**Task, response, and explanation**
quit all ↵	
	Task: Exit from all levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory	
	Task: Exit from a specified directory without leaving any other directories.
	Response: AMADUMP>>> >
	Explanation: The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵	
	Task: Exit from a specified number of levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End	

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

resetovr

Function

Use the resetovr command to reset the override status for a currently-selected AFT file so that you can select another file to override all others for transfer. The resetovr command causes the system to remove the override status pointer (O->) from the currently-selected file in the AFT system directory. The file with the N-> pointer now will be the next file to transfer.

resetovr command parameters and variables	
Command	**Parameters and variables**
resetovr	*session_name*
Parameters and variables	**Description**
session_name	This variable specifies the name of the AFT session.

Qualifications

The resetovr command is qualified by the following exceptions, restrictions, and limitations:

- This command does not work if the currently-marked file is in the process of transferring.
- When this command is executed, the system does not delete the file previously selected for override.

Example

The following table provides an example of the resetovr command.

Example of the resetovr command	
Example	**Task, response, and explanation**
resetovr aft1 ↵ *where* aft1 specifies the name of the AFT session	
	Task: Reset the override status of an AFT file.
	Response: `The override file has been reset.`
	Explanation: This command resets the override pointer in the AFT system directory.

resetovr (continued)

Responses

The following table provides explanations of the responses to the resetovr command.

Responses for the resetovr command	
MAP output	**Meaning and action**
`Bad AFT message received`	
	Meaning: The system did not recognize the message.
	Action: If this happens frequently, contact the next level of support.
`Bad send of message to AFT session - See SWERR`	
	Meaning: The command cannot contact the AFT software. The override file is not reset and a SWERR is generated.
	Action: Contact the next level of support and provide them with the data output in the SWERR.
`Could not contact AFT system`	
	Meaning: The AFT software does not respond to the command.
	Action: Contact the next level of support.
`<session name> is invalid`	
	Meaning: You entered an invalid AFT session name. The override file is not reset.
	Action: Check the GASINFO table and reenter the command with a valid session name.
`<session name> is not an AFT session`	
	Meaning: You entered a session name which is in the GASINFO table, but is not an AFT session. The override file is not reset.
	Action: Check the GASINFO table and reenter the command with a valid session name.
-continued-	

resetovr (end)

Responses for the resetovr command (continued)
MAP output   **Meaning and action**

`The override file has been reset.`
 	
                Meaning: The command executed normally. The override status pointer is removed from the current file. This file is not deleted from the directory.
 	
                Action: Query the AFT system directory to confirm that the file no longer is marked by the override status pointer.

`The override file is transferring and cannot be reset`
 	
                Meaning: You entered the command while the override file was transferring. The override file is not reset.
 	
                Action: Wait and try again.

`There is no override file to reset`
 	
                Meaning: You executed the command on an AFT session with no override files in its directory. The override file is not reset.
 	
                Action: None

End

resetpft

Function

Use the resetpft command to change the transfer state of a file from partial to pending. When this command is executed, the partial file transfer (PFT) file specified by the command has a transfer state of pending, the last acknowledged block set to zero, and the retry count set to zero. When a file reset by this command is transferred, the transfer starts at the beginning of the file.

resetpft command parameters and variables		
Command	**Parameters and variables**	
resetpft	*session_name*	*fn*
Parameters and variables	**Description**	
fn	This variable specifies the file which is to be reset. Enter the file name exactly as it appears in the AFT system directory.	
session_name	This variable specifies the name of the AFT session.	

Qualification

> **CAUTION**
> **Do not attempt to reset a currently-transferring file**.
> The resetpft command cannot be used to reset a currently-transferring file.

The resetpft command cannot be used to reset a currently-transferring file.

resetpft (continued)

Examples

The following table provides examples of the resetpft command.

Examples of the resetpft command	
Example	**Task, response, and explanation**
resetpft aft1 ↵ *where* aft1 specifies the name of the AFT session	

Task:	Reset the transfer state of an AFT session.
Response:	

```
==AFT SYSTEM: STARTED -AFT STATE: SENDING -AFT CLASS: GBKGCLASS==

     FILE NAME          FILE      LAST      XFER      COPY      START
                        SIZE      ACK       STATE     STATE     TIME
    ---------------- -------- -------- -------- -------- --------
     U880531141055OCC    12900    12900 COMPLETE PENDING  31/12/45
     U880531123056OCC     5000     5000 COMPLETE PENDING  31/22/33
  N->U880531125057OCC     4890     2380 PARTIAL  PENDING  31/22/45
     U880531133558OCC    12900        0 PENDING  PENDING
     U880531141059OCC    12900        0 PENDING  PENDING
     A880531143360OCC    12839        0 PENDING  PENDING
```

Explanation:	The system resets the transfer state of all files in the AFT1 session.

resetpft aft1 u880531125057occ ↵ *where* aft1 specifies the name of the AFT session u880531125057occ specifies the file which is to be reset	

Task:	Reset the transfer state of a specified file.
Response:	File U880531141055OCC has been reset.
Explanation:	The system resets the transfer state of the file named u880531125057occ.

resetpft (continued)

Responses

The following table provides explanations of the responses to the resetpft command.

Responses for the resetpft command	
MAP output	**Meaning and action**
`Bad send of message to AFT session - See SWERR`	
	Meaning: The command cannot contact the AFT software. The AFT system is not updated and a SWERR also is generated.
	Action: Contact the next level of support and provide them with the data in the SWERR.
`Bad AFT message received`	
	Meaning: The command received a message which it did not recognize. The AFT system is not updated.
	Action: Contact the next level of support.
`Could not contact AFT system`	
	Meaning: The command does not receive a response from the AFT software. The AFT system is not updated.
	Action: Contact the next level of support.
`File <file name> is ACTIVE and cannot be reset`	
	Meaning: You specified the currently-transferring file when executing this command. The AFT system is not updated.
	Action: Take down the transfer and reenter the command.
`File <file name> is not in the AFT directory`	
	Meaning: You entered a file name which is not in the AFT system directory. The AFT system is not updated.
	Action: Reenter the command with the correct file name.
-continued-	

resetpft (end)

Responses for the resetpft command (continued)
MAP output   **Meaning and action**
`File <file name> is not a PFT file`
Meaning: You entered a file name which is not a partial file. The AFT system is not updated. **Action:** Enter the file name of a partial file.
`<session name> is invalid`
Meaning: You supplied an invalid session name. The AFT system is not updated. **Action:** Check the GASINFO table and reenter the command with the correct session name.
`<session name> is not an AFT session`
Meaning: You entered session name which is in the GASINFO table, but is not an AFT session. The AFT system is not updated. **Action:** Refer to the GASINFO table and reenter the command with the correct session name.
`The <file name> has been reset`
Meaning: The command executed normally. The file specified by the command has its transfer state changed to pending, its last acknowledged block changed to "0," and its retry count set to "0." The AFT system is not updated. **Action:** Query the AFT system directory to confirm that the file information has been updated.
End

setaft

Function

Use the setaft command to set the next AFT file to transfer. Setting the next file to transfer is similar to setting the transfer point of the AFT system. The file set to transfer next is designated by the "next file pointer" (N->) symbol that displays beside the specified file. The file set to transfer next does not begin transferring until the currently-transferring file completes; subsequent files transfer sequentially after the file specified by this command transfers.

setaft command parameters and variables		
Command	**Parameters and variables**	
setaft	*session*	*fn*
Parameters and variables	**Description**	
fn	This variable specifies the name of the AFT file you want to designate as the next file to transfer. This entry must duplicate the Device Independent Recording Package (DIRP) file name as listed in the display of the AFTCI directory queryaft command.	
session	This variable specifies the name of the AFT session.	

Qualifications

The setaft command is qualified by the following exceptions, limitations, and restrictions:

- This command will not interrupt the currently-transferring file.
- If the command is executed when the system is disabled, the first file to transfer when the AFT system restarts is the file specified by the command.
- A file in the complete, active, or error state cannot be designated as the next file to transfer.
- If there is an override file in the AFT system directory at the time this command is executed, the override file takes precedence. The file specified by this command transfers after the completion of the override file.

setaft (continued)

Example

The following table provides an example of the setaft command.

Example of the setaft command	
Example	**Task, response, and explanation**
setaft aft1 u880531141059occ ↵ *where* aft1 specifies the name of the AFT session u880531141059occ specifies the name of the AFT file to transfer	
	Task: Set the next AFT file to transfer. **Response:** `The next file AFT will transfer is AFT file name` **Explanation:** The next file to be transferred is specified.

Responses

The following table provides explanations of the responses to the setaft command.

Responses for the setaft command	
MAP output	**Meaning and action**
`Bad AFT message received`	
	Meaning: This response displays if the command receives a corrupt message from the AFT system. The AFT system is not updated. **Action:** Contact the next level of support.
`BAD send of message to AFT system - See SWERR`	
	Meaning: The AFT system cannot be contacted. The AFT system is not updated. **Action:** Record the information in the SWERR and contact the next level of support.
`Could not contact AFT system`	
	Meaning: The AFT system cannot be contacted. The AFT system is not updated. **Action:** Contact the next level of support.
-continued-	

Responses for the setaft command (continued)	
MAP output	**Meaning and action**
File is already transferring	
	Meaning: You tried to set the next file while it is transferring. The AFT system is not updated.
	Action: Wait and try again later.
File is in a non-transferrable state	
	Meaning: This response displays if the file you specify is in the manual, complete, active, or error state. The AFT system is not updated.
	Action: Enter the command again with the correct file name.
File is not in the AFT directory	
	Meaning: You supplied a file name that is not in AFT system directory. The AFT system is not updated.
	Action: Enter the command again with the correct file name.
\<file name\> will transfer next	
	Meaning: The command completed normally. The AFT system modifies its internal data structures to reflect the change in the file transfer order.
	Action: Examine the file name in the response to make certain that the next file the AFT system transfers is the correct file.
Session name is invalid	
	Meaning: You entered an invalid AFT session name. The AFT system is not updated.
	Action: Refer to the GASINFO table and retry the command with the correct session name.
End	

setovr

Function

Use the setovr command to set an override file transfer. The file set to override is designated by the "override file pointer" (O->) symbol that displays beside the specified file. The file set to override begins transferring when the AFT system restarts. After the files set to override transfer properly, the next sequential file transfers.

setovr command parameters and variables		
Command	**Parameters and variables**	
setovr	*session*	*fn*

Parameters and variables	Description
fn	This variable specifies the name of the AFT file you want to designate as the next file to transfer. This entry must duplicate the Device Independent Recording Package (DIRP) file name as listed in the display of the AFTCI directory queryaft command.
session	This variable specifies the name of the AFT session.

Qualifications

The setovr command is qualified by the following exceptions, limitations, and restrictions:

- This command be used to transfer any file without a transfer state of manual.

- The setovr command can be reversed by executing the resetovr command.

> **CAUTION**
> **Improper use can result in duplicate files.**
> Using this command improperly can result in duplicate files on the remote processor.

Using this command improperly can result in duplicate files on the remote processor.

setovr (continued)

Example

The following table provides an example of the setovr command.

Example of the setovr command	
Example	**Task, response, and explanation**
setovr aft1 u880531141055occ ↵ *where* aft1 specifies the name of the session u880531141055occ specifies the file which is to be transferred next	
	Task: Set an override file transfer.
	Response: File U880531141055OCC has been set to Override.
	Explanation: When the command is executed, the override pointer is moved to the file specified by the command.

Responses

The following table provides explanations of the responses to the setovr command.

Responses for the setovr command	
MAP output	**Meaning and action**
Bad AFT message received	
	Meaning: This response displays if the command receives a corrupt message from the AFT system. The AFT system does not set the override pointer.
	Action: Contact the next level of support.
BAD send of message to AFT system - See SWERR	
	Meaning: The AFT system cannot be contacted. The AFT system does not set the override pointer.
	Action: Record the information in the SWERR and contact the next level of support.
-continued-	

Responses for the setovr command (continued)

MAP output   **Meaning and action**

```
Could not contact AFT system
```

> **Meaning:** The AFT system cannot be contacted.   The AFT system does not set the override pointer.
>
> **Action:**   Contact the next level of support.

```
File <file name> is already transferring
```

> **Meaning:** You tried to set the file to override while it is transferring. The AFT system does not set the override pointer.
>
> **Action:**   Wait and try again later.

```
File <file name> is in a non-overridable state
```

> **Meaning:** You tried to set a file to override when the file is in a manual transfer state.   A file in a manual transfer state cannot be designated as the override file. The AFT system does not set the override pointer.
>
> **Action:**   Select another file or continue with the next task.

```
File <file name> is not in the AFT directory
```

> **Meaning:** You supplied a file name that is not in AFT system directory. The AFT system does not set the override pointer.
>
> **Action:**   Enter the command again with the correct file name.

```
File <file name> is set for override transfer
```

> **Meaning:** The command executed normally. The AFT system sets the override pointer on the file specified in the command.
>
> **Action:**   Query the AFT system directory to confirm that the override pointer is on the correct file.

```
<session name> is invalid
```

> **Meaning:** You entered an invalid AFT session name. The AFT system does not set the override pointer.
>
> **Action:**   Refer to the GASINFO table and reenter the command with the correct session name.

-continued-

setovr (end)

Responses for the setovr command (continued)	
MAP output	**Meaning and action**
`<session name> is not an AFT session`	
	Meaning: You entered a session name which is in the GASINFO table, but is not an AFT session. The AFT system does not set the override pointer.
	Action: Refer to the GASINFO table and reenter the command with the correct session name.
End	

startaft

Function

Use the startaft command to start transferring AFT system files.   A
one-minute timer is started when this command is executed. When the timer
expires, the file transfer starts. If there is an override file in the AFT system,
the override file is transferred first. Otherwise, the next file is transferred.

startaft command parameters and variables	
Command	**Parameters and variables**
startaft	*session*
Parameters and variables	**Description**
session	This variable specifies the name of the AFT session.

Qualification

This command can be reversed by executing the AFTCI directory stopaft
command before the one-minute start timer expires.

Example

The following table provides an example of the startaft command.

Example of the startaft command	
Example	**Task, response, and explanation**
startaft aft1 ↵ *where* aft1 specifies the name of the session	
	Task: Start the AFT session.
	Response: `AFT session AFT1 has been started.`
	Explanation: The specified AFT session started.

startaft (end)

Responses

The following table provides explanations of the responses to the startaft command.

Responses for the startaft command	
MAP output	**Meaning and action**
`AFT is not connected with SST - Cannot Start`	
	Meaning: This response displays if the connection between the AFT system and SST is down. The AFT system attempts to bring up the connection every minute. The AFT system is not started.
	Action: Contact the next level of support.
`AFT session <session name> has been started`	
	Meaning: This is the normal response for the startaft command. The startaft command sets a one-minute start timer. When this timer expires, the AFT system starts transferring files.
	Action: To disable the AFT session before files begin transferring, the AFTCI directory stopaft command must be executed before the one-minute timer expires.
`Bad send of message to AFT system - See SWERR`	
	Meaning: This response displays if the AFT system cannot be contacted. The AFT system is not started.
	Action: Contact the next level of support.
`<session name> is invalid`	
	Meaning: The session name you supplied is not a valid AFT session. The AFT system is not started.
	Action: Refer to Table GASINFO for a valid AFT session name and reenter the command.
`Session <session name> is already started`	
	Meaning: This response displays if the AFT session specified in the command already is started. The AFT system is not restarted.
	Action: None

stopaft

Function

Use the stopaft command to stop transferring AFT system files. This command does not interrupt the actively-transferring file. The AFT system waits until the actively-transferring file completes before halting the transfer process.

stopaft command parameters and variables	
Command	**Parameters and variables**
stopaft	*session*
Parameters and variables	**Description**
session	This variable specifies the name of the AFT session.

Qualification

Use the AFTCI directory startaft command to resume file transfer.

Example

The following table provides an example of the stopaft command.

Example of the stopaft command		
Example	**Task, response, and explanation**	
stopaft aft1 ↵ *where*		
aft1	specifies the name of the AFT session	
	Task:	Stop the AFT system from transferring files.
	Response:	`AFT session AFT1 has been stopped`
	Explanation:	The AFT system stops transferring files after the current session.

stopaft (end)

Responses

The following table provides explanations of the responses to the stopaft command.

Responses for the stopaft command	
MAP output	**Meaning and action**
`AFT session <session name> has been stopped`	
	Meaning: This command executed correctly. The stopaft command stops transferring files after the completion of the actively-transferring file.
	Action: When you want the AFT system to begin transferring files again, the AFTCI directory startaft command must be executed.
`Bad send of message to AFT system - See SWERR`	
	Meaning: This response displays if the AFT system cannot be contacted. The AFT system is not stopped.
	Action: Contact the next level of support.
`<session name> is invalid`	
	Meaning: The AFT session name you specified is not valid. The AFT system is not stopped.
	Action: Refer to Table GASINFO for a valid AFT session name and reenter the command.
`Session <session name> is already stopped`	
	Meaning: This response displays if the AFT session specified in the command already is stopped.
	Action: Go to the next task.

AMADUMP level commands

Use the AMADUMP level of the MAP to display or print the contents of the automatic message accounting (AMA) files produced in local or centralized AMA offices.

The following formats can be used:

- block-by-block hexadecimal dump of the file contents for a specified range of blocks
- record-by-record dump of AMA call entries, data entries, or header entries in an AMA file with or without screening specified

Accessing the AMADUMP level

To access the AMADUMP level, you must specify the format and a file name in addition to the directory entry code. The general syntax of the command string you enter from the CI level is as follows:

> **amadump *format file_name* ↵**

The *format* variable specifies the form in which the data are transmitted and stored. Valid formats are nt, intl, cdr, cdra, cdrb, cdrc, cdrctemp, cdrd, vcdrucs26, and bc. The most common format is Bellcore (bc).

The *file_name* variable replacement value is used when entering the AMADUMP director to specify the name of the file or calldump. The file name may be any AMA, SMDR, or other billable file resident on the volume.   In addition, two special file name values are included in list of valid entries. The ama_active parameter specifies a file name which opens the currently mounted active AMA file. The ama_parallel parameter specifies a file name which opens the currently mounted parallel AMA file.

Refer to the PROG directory amadump command for a formal description of the syntax required to enter the AMADUMP directory.

AMADUMP commands

The commands available at the AMADUMP MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

AMADUMP commands	
Command	**Page**
dump	A-283
filter	A-291
help	A-301
quit	A-303

dump

Function

Use the dump command to display call information from DIRP. The display appears either in a simple or detailed format.

Note: Using the new parameter quickly produces a display of records generated for test calls just placed. This prevents the system from searching the entire open file.

dump command parameters and variables	
Command	**Parameters and variables**
dump	call data hdr hex new [*nodetails* / details / summary] [*all blocks* / startblks] [*first block* / numblks]

Parameters and variables	Description
all blocks	Omitting this entry forces the system to default to all blocks as the number of blocks to be dumped.
first block	Omitting this entry forces the system to default to the first block as the starting number of the blocks to be displayed
nodetails	Omitting this entry forces the system to default to using the simple format with each field in the record, separated by a space.
call	This parameter dumps header entries, control entries, and call entries in the C1C1 call record block. The block is dumped in ASCII. The data also includes associated call extension entries when they are present.
data	This parameter dumps header entries, control entries, and data entries in the C2C2 data blocks. This information is not call-related; it generally represents operational measurements (OMs) taken over a set period of time.
details	This parameter dumps records in detailed format as follows: ▪ provides each field in the record associated with the field name ▪ appends additional information to the end of each record indicating if the call type was direct dialing overseas and if the call was answered. (If a value is not specified, the system defaults to using simple format with each field in the record, separated by a space.)
hdr	This parameter dumps the block header entries and control entries.
-continued-	

dump (continued)

dump command parameters and variables (continued)	
Parameters and variables	**Description**
hex	This parameter causes an unformatted hexadecimal dump of block data and is applicable to all tape formats.
new	This parameter displays all new call records on an open file since the last dump was initiated, or since entering the AMADUMP level. This entry is valid only when you enter the AMADUMP level by specifying the special filename ama_active or ama_parallel in the entry command string.
numblcks	This variable specifies the number of blocks to be dumped. The valid entry range is 1-32767. (If a value is not specified, the system defaults to displaying all blocks.)
startblk	This variable specifies the starting number of the blocks to be displayed. The count is from the beginning of the file and not the block count shown in the C1C1 header. The valid entry range is 1-32767. (If a value is not specified, the system defaults to the first block.)
summary	This parameter displays a record count of each block and total record counts as well as block counts for the file. Actual records are not displayed.
End	

Qualifications

The dump command is qualified by the following exceptions, restrictions, and limitations:

- Call entries printed using the dump call command string or dump data command string can be screened based on fields that satisfy certain data patterns or time constraints. The AMADUMP directory filter command is used for this purpose and must be specified prior to using the dump command.

- The dump data command string cannot be used to obtain AMA record information with the Bellcore format.

- The dump command cannot be used if the calldump command was specified. The dump command only can be used with a DIRP file.

- Errors that occur while using the dump command are signaled by error messages that detail the reason for the failure.

- The new parameter cannot be used unless you enter the AMADUMP level by specifying the special filename ama_active or ama_parallel in the entry command string.

- If the dump new command string is being used for the first time, all AMA records recorded in the open file since you accessed the AMADUMP directory display (subject to filtering).

dump (continued)

Example

The following table provides an example of the dump command.

Example of the dump command	
Example	**Task, response, and explanation**
dump call details ↵	
	Task: Display call entry records in detailed format.
	Response:
| | ```
A BC AMA FILE IS BEING PROCESSED.
*HEX ID=AA   STRUCT CODE:00020C CALL TYPE:001C
     SENSOR TYPE:036C
SENSOR ID:0000000C  REC OFC TYPE:036C  REC OFC
     ID:0000000C
DATE:60422C   TIMING IND:00000C  STUDY
     IND:2300000C  ANSWER:0C
SERV OBSERVED:0C  OPER ACTION:0C SERV FEAT:000C
     ORIG NPA:613C
ORIG NO:6211092C  OVERSEAS IND:1C  TERM
     NPA:00613C
TERM NO:6211234C  CONN TIME:0043506C  ELAPSED
     TIME:000000051C  WATS IND:0C  WATS BAND
     IND:020C
``` |
| | **Explanation:**   This example provides one call entry record of the detailed call record format. |

Responses

The following table provides explanations of the responses to the dump command.

Responses for the dump command	
MAP output	**Meaning and action**
BLOCK READ FAILED	
	Meaning: Either an input/output (I/O) error has occurred or an incorrect block size was entered.
	Action: Reissue the command. If the command fails again, contact the next level of maintenance. Note the file name, device, the date that the file was created, and the block that failed.
	-continued-

dump (continued)

Responses for the dump command (continued)

MAP output	Meaning and action

`COMMAND ABORTED.  COULD NOT OPEN FILE: <filename>`

> **Meaning:** The specified file does not exist.
>
> **Action:** Open an existing file with the necessary information or create a new file.

`COMMAND ABORTED.  INVALID DETAILS PARAMETER: <parameter>`

> **Meaning:** The parameter was specified incorrectly.
>
> **Action:** Reissue the command using the correct parameter.

`COMMAND ABORTED.  UNKNOWN DUMP FUNCTION: <function>`

> **Meaning:** The appropriate dump parameters are hdr, hex, call, and data. (Entering the dump data command string produces a message referring you to the dump call command string).
>
> **Action:** Reissue the command using the appropriate dump function.

`CORRUPT RECORD ENCOUNTERED.`

> **Meaning:** One of the records being displayed is corrupt. The command was aborted.
>
> **Action:** Pinpoint the records being displayed to avoid the corrupt records and retry the command with the necessary parameters.

`CORRUPTED BDW <bdw> FOUND IN BLOCK NO: <block number> BDW SIZE COMPUTED AS: <size> BLOCK READ FAILED FAILURE OCCURRED WHILE DUMPING FILE: <filename>`

> **Meaning:** The block descriptor word of the record is corrupted.
>
> **Action:** Reissue the command.

`CORRUPTED RDW <rdw> FOUND IN BLOCK NO: <block number> RDW SIZE COMPUTED AS: <size> BAD RECORD ENCOUNTERED PROCEEDING TO NEXT BLOCK`

> **Meaning:** The record descriptor word, which is the first four bytes of a record, is corrupted.
>
> **Action:** Reissue the command.

-continued-

dump (continued)

Responses for the dump command (continued)

MAP output	Meaning and action

COULD NOT FORMAT BLOCK CONTAINING DEFERRED DATA. ERROR OCCURRED IN BLOCK
NO: <block number> PROCEEDING TO NEXT BLOCK.

> **Meaning:** The specified block contains unformatted AMA data. The data could be corrupted or the tape came from a switch that does not have the optional deferred AMA feature.
>
> **Action:** Ensure that the switch software can handle the information on the tape. Reissue the command. If the command fails again, contact the next level of maintenance.

END OF RECORD WAS ENCOUNTERED WITHOUT DETECTING MODULE CODE ZERO. RECORD
SIZE (NOT INCLUDING RDW) IS: <size> IN BLOCK NO: <block number record
contents> BAD RECORD ENCOUNTERED PROCEEDING TO NEXT RECORD

> **Meaning:** Module code 000 is the final module in any set of modules in an AMA record. If module code 000 is missing, this error message occurs.
>
> **Action:** Ensure that module code 000 is the final module in the AMA record.

ERROR ENCOUNTERED WHILE CLOSING FILE: <filename>

> **Meaning:** An internal software or hardware error of unknown origin has occurred.
>
> **Action:** Contact the next level of maintenance.

FAILURE OCCURRED WHILE DUMPING FILE: <filename>

> **Meaning:** This message accompanies other error messages explaining the failure.
>
> **Action:** Refer to accompanying error messages to determine fault causes and corrective actions.

NIL ASPECT INVOKED FOR BLOCK READ

> **Meaning:** An internal software or hardware error of unknown origin has occurred. This message could appear not only for block read, but also for record read, filter, and dump.
>
> **Action:** Contact the next level of maintenance.

-continued-

dump (continued)

Responses for the dump command (continued)	
MAP output	**Meaning and action**
NO RECORDS MATCHING FILTER DATA WERE FOUND	
	Meaning: A dump of call records was requested, but based on previously requested filter values, no matching call records were found.
	Action: None
REMINDER WARNING: FILTER FUNCTION IS ENABLED.	
	Meaning: The filter function is active. Only specific AMA records and associated field values display. (The filter function restricts the AMA records and field display to those associated with entered values.)
	Action: None
SOFTWARE ERROR. RECORD IN BLOCK <block number> EXCEEDS MAXIMUM SIZE OF A FILTER TABLE ENTRY <size> RECORD CONTENTS: <record contents> FILTER PROCESSING IS UNAFFECTED AND CONTINUES NORMALLY	
	Meaning: The specified record is longer than the maximum size of a filter table entry.
	Action: Reduce the size of the record.
START BLOCK WAS NOT FOUND	
	Meaning: The start block number exceeds the number of blocks on the tape.
	Action: Reissue the command with a valid start block number.
THE DUMP DATA COMMAND IS NOT APPLICABLE TO BELLCORE AMA FORMAT. ALL BELLCORE AMA RECORDS ARE DISPLAYED WITH THE DUMP CALL SUBCOMMAND.	
	Meaning: The dump data command string cannot be used to obtain AMA record information with the Bellcore format.
	Action: Use the dump call command string to obtain AMA record information.
TRUNCATED RECORD ENCOUNTERED RECORD SIZE (NOT INCLUDING RDW) IS: <size> IN BLOCK NO: <block number record contents> BAD RECORD ENCOUNTERED PROCEEDING TO NEXT RECORD	
	Meaning: One of the records in the file was truncated.
	Action: None
	-continued-

dump (end)

Responses for the dump command (continued)

MAP output   Meaning and action

```
UNRECOGNIZED MODULE CODE <module code> ENCOUNTERED IN BLOCK NO: <block
number record contents> BAD RECORD ENCOUNTERED PROCEEDING TO NEXT RECORD
```

> **Meaning:** Specific module code numbers can be appended to AMA records. This error message indicates that a specified module code is nonexistent or is not recognized by the software in this feature package.
>
> **Action:** Reissue a correct module code.

```
UNRECOGNIZED STRUCTURE CODE <sc> IN BLOCK NO: <block number record
contents> BAD RECORD ENCOUNTERED PROCEEDING TO NEXT RECORD
```

> **Meaning:** Specific structure codes exist for AMA records. This error message indicates that a specified structure code is nonexistent or is not recognized by the software in this feature package.
>
> **Action:** Reissue a correct structure code.

```
UNSUCCESSFUL ATTEMPT TO READ BLOCK NO: <block number> FILESYS RETURN CODE:
<return code> BLOCK READ FAILED FAILURE OCCURRED WHILE DUMPING FILE:
<filename>
```

> **Meaning:** Either an I/O error has occurred or an incorrect block size was entered.
>
> **Action:** Reissue the command. If the command fails again, contact the next level of maintenance. Note the file name, device, the date that the file was created, and the block that failed.

```
***** WARNING ***** THIS FILE CONTAINS UNFORMATTED DATA WHICH CANNOT BE
READ BY DOWNSTREAM RAO PROCESSORS
```

> **Meaning:** The file contains unformatted AMA data. The data could be corrupted or the tape came from a switch that does not have the optional deferred AMA feature.
>
> **Action:** Ensure that the switch has software capable of handling the information on the tape. Reissue the command. If the command fails again, contact the next level of maintenance.

End

---

filter

Function

Use the filter command to screen call billing records. Calls can be screened by specific fields and displayed with the dump command.

filter command parameters and variables	
Command	**Parameters and variables**
filter	add *field_name* [* / *value*] [*and* / *logical*] [*eq* / *range*] delete [0 / *entry*] display [*all fields* / *field_name*] [*_* / *value*] [*and* / *logical*] [*eq* / *range*] disable enable

Parameters and variables	Description
all fields	Omitting this entry forces the system to default to displaying all possible fields names. This is the the default value for the filter display command string only.
and	Omitting this entry forces the system to default to using a value of and as the logical operation to use with multiple filters. A record displays if it passes all the filters with a value of and. This is the the default value for both the filter add command string and the filter display command string.
eq	Omitting this entry forces the system to default to using a value of eq as the numerical range by which to filter the value. This is the the default value for both the filter add command string and the filter display command string.
_	Omitting this entry forces the system to default to using a wildcard for the screening value for the filter display command string only.
*	This parameter acts as a wildcard for the screening value.
0	This parameter removes all entries from the filter table.
add	This parameter adds a filter to the filter table. Once the filter table has one or more filters, it can be enabled and the filters will be applied to the records.
delete	This parameter removes the specified filter from the filter table.
-continued-	

filter (continued)

filter command parameters and variables (continued)	
Parameters and variables	**Description**
disable	This parameter disables filter screening. This parameter has no effect on the filter table.
display	This parameter displays the contents of the filter table. If the *field_name* variable replacement value is appended to the display command, all possible field names display.
enable	This parameter enables record screening. Screening is based on the contents of the filter table.
entry	This variable specifies the entry to remove from the filter table. This entry correlates to the number shown on the left edge of the filter table. The valid entry range is 1-50. (To remove all of the entries, enter a 0.)
field_name	This variable specifies the Bellcore field name by which to filter. The field name should be entered as a character string in the exact form shown when the fields are displayed using the filter display fields command string. In the first position, the *field_name* variable value replacement is used to specify a field when adding a filter to the filter table. In the second position, the *field_name* variable value replacement is used to display all possible field names in the filter table.
logical	This variable specifies the logical operation to use with multiple filters. The valid entry values are either the and value or the or value. A record will be displayed if it passes all the filters with the value of and, or if at least one filter passes with the value of or. (If no logical is specified, the system defaults to using a value of and.)
range	This variable specifies the numerical range by which to filter the value. It may be eq, neq, lte, or gte. (If no numerical range is specified, the system defaults to using a value of eq.)
value	This variable specifies the screening values. The valid entry values are in the range from either 0-9 or A-F and the asterisk (*). (The asterisk (*) entry acts as a wildcard.)
End	

Qualifications

The filter command is qualified by the following exceptions, restrictions, and limitations:

- Adding an entry to the filter table does not enable the filter command. To enable the screening process, you first must add an entry to the filter table and then enable the filter.

filter (continued)

- The filter command can define screening for call records when using the dump call command string, dump data command string, or calldump option.

Examples

The following table provides examples of this command.

Examples of the filter command		
Example	**Task, response, and explanation**	
filter display ↵		
	Task:	Display the contents of the filter table.
| | **Response:** | ```
Filter   Field              Filter      Filter
entry    name               value       attributes
-------------------------------------------------
   1     STRUCTURE_CODE     00076       AND EQ
   2     CALL_CODE          003         AND EQ

Note: Filtering is currently enabled.
``` |
	Explanation:	This command displays the contents of the filter table. The filter table contains two entries. The first entry screens for a structure_code of 00076, while the second entry screens for a call_code of 003.
filter add structure_code 00625 ↵ *where* structure_code 00625	specifies the field to which the filter is applied specifies the particular structure code value	
	Task:	Add an entry to the filter table.
| | **Response:** | ```
>>>Filter successfully added as filter entry 1
>>>NOTE: Filter function is currently disabled.
``` |
| | **Explanation:** | This command adds structure code 00625 to the filter table. When the process is complete, the system displays a confirmation message indicating that the function is complete. The status also displays. |

filter (continued)

Responses

The following tables explain the responses to the filter command. The command responses are grouped as follows:

- all formats
- filter add command string (Bellcore format)
- filter delete command string
- filter display command string (Bellcore format)
- filter enable command string and filter disable command string (Bellcore format)

The following table explains the responses to any filter command string for all formats.

Responses for the filter command (all formats)	
MAP output	**Meaning and action**
COMMAND ABORTED. ERROR ENCOUNTERED WHILE READING FILTER FUNCTION PARAMETER.	
	Meaning: Abort was entered in response to the prompt for a parameter.
	Action: None
COMMAND ABORTED. RETURN CODE ERROR: <return code>	
	Meaning: An internal software or hardware error of unknown origin occurred.
	Action: Contact the next level of maintenance.
COMMAND ABORTED. UNKNOWN FILTER FUNCTION: <function>	
	Meaning: An invalid filter function was entered. Valid filter functions are add, delete, display, enable, and disable.
	Action: Reissue the command using a known function.

filter (continued)

The following table explains the responses to the filter add command string (Bellcore format).

Responses for the filter add command string (Bellcore format)	
MAP output	**Meaning and action**
`ADDING..... STRUCTURE CODE <sc>`	
	Meaning: The specified structure code is being added to the filter table.
	Action: None
`BAD CHARACTERS <characters>.` `ENTER AGAIN:`	
	Meaning: Incorrect characters were entered with the filter command.
	Action: Reissue the command.
`COMMAND REJECTED. INVALID PARAMETER: <parameter>`	
	Meaning: An invalid parameter was entered with the filter command.
	Action: Enter a valid parameter.
`COMMAND REJECTED.  INVALID STRUCTURE CODE: <sc>`	
	Meaning: An invalid structure code was entered.
	Action: Enter a valid structure code.
`COMMAND REJECTED STRUCTURE CODE <sc> IS ALREADY IN ENTRY <entry number>` `OF THE FILTER TABLE.`	
	Meaning: You tried to add a structure code using the filter command, but the code for the entry number already is present in the filter table.
	Action: Reissue the command using a structure code that is not known to the filter table.
`COMMAND REJECTED.  THE FILTER TABLE IS FULL. A MAXIMUM OF 10 ENTRIES ARE` `ALLOWED.`	
	Meaning: You tried to add more than ten entries using the filter add command string.
	Action: Reissue the filter add command string with no more than ten entries.
-continued-	

filter (continued)

Responses for the filter add command string (Bellcore format) (continued)	
MAP output	**Meaning and action**
FILTER ADDITION ABORTED. THE MAXIMUM LENGTH OF A FILTER ENTRY (304) HAS BEEN EXCEEDED.	
	Meaning: A filter addition was aborted because too many fields were added to an entry.
	Action: Enter the filter add command string again, ensuring that the entries do not exceed the specified maximum length.
INVALID MODULE CODE <mc>. ENTER AGAIN:	
	Meaning: An invalid module code was entered.
	Action: Enter a valid module code.
SIZE OF DATA ENTERED <size> EXCEEDS SIZE OF FIELD <size> ENTER AGAIN:	
	Meaning: Information entered using the filter add command string exceeded the size of a particular field.
	Action: Reissue data that does not exceed field size.
STRUCTURE CODE <sc> HAS BEEN ADDED AS ENTRY <entry number> FILTER ADDITION COMPLETE.	
	Meaning: The specified structure code has been entered using the filter add command, and the structure code has been added to the filter table
	Action: None
THE FILTER FUNCTION IS CURRENTLY: ON COMMAND REJECTED. THE FILTER FUNCTION MUST BE DISABLED BEFORE ENTRIES CAN BE DELETED.	
	Meaning: To delete entries from the file, the filter function must be disabled.
	Action: Disable the filter function by entering the filter disable command string.
End	

filter (continued)

The following table explains the responses to the filter delete command string.

Responses for the filter delete command string	
MAP output	**Meaning and action**
COMMAND ACCEPTED. ALL ENTRIES IN THE FILTER TABLE HAVE BEEN REMOVED or COMMAND ACCEPTED. STRUCTURE CODE <sc> HAS BEEN REMOVED FROM ENTRY <entry number> IN THE FILTER TABLE.	
	Meaning: The filter delete command string executed successfully.
	Action: None
COMMAND ACCEPTED. STRUCTURE CODE <sc> HAS BEEN REMOVED FROM ENTRY <entry number> IN THE FILTER TABLE.	
	Meaning: The filter delete command executed successfully.
	Action: None
COMMAND REJECTED. STRUCTURE CODE <sc> WAS NOT FOUND IN THE FILTER TABLE.	
	Meaning: An attempt was made to delete an entry from the filter table, but the table contained no entries.
	Action: None
COMMAND REJECTED. THE FILTER TABLE IS EMPTY.	
	Meaning: An invalid parameter was entered with the filter command.
	Action: Reissue the command with a valid parameter.
THERE ARE NO MORE ENTRIES IN THE FILTER TABLE.	
	Meaning: Once enabled, the filter function remains enabled until the filter disable command string is entered.
	Action: None

filter (continued)

The following table explains the responses to the filter display command string (Bellcore format).

Responses for the filter display command string (Bellcore format only)	
MAP output	**Meaning and action**
COMMAND REJECTED. INVALID STRUCTURE CODE: <sc> END OF FILTER DISPLAY OR FILTER ENTRY: <entry number> UNRECOGNIZED STRUCTURE CODE <sc> ENCOUNTERED <record contents> OR FILTER ENTRY: <entry number> *EMPTY OR FILTER ENTRY: <entry number entry contents> END OF FILTER DISPLAY	
	Meaning: You entered an invalid structure code.
	Action: Reissue the command with a valid structure code.
THE FILTER FUNCTION IS CURRENTLY: OFF	
	Meaning: The filter function is disabled.
	Action: None
THE FILTER FUNCTION IS CURRENTLY: ON	
	Meaning: You activated the filter function by using the filter enable command string. Now, when call records display, only those call records and fields specified in the filter table appear.
	Action: None

filter (end)

The following table explains the responses to the filter enable command string and filter disable command string (Bellcore format).

Responses for the filter disable command string (Bellcore format only)
MAP output   Meaning and action

`COMMAND ACCEPTED.  THE FILTER FUNCTION IS NOW: ON` `DUMP COMMANDS WILL SCREEN ON THE BASIS OF DATA IN THE FILTER TABLE.`
Meaning: The filter function has been enabled with the filter enable command string. **Action:**   None
`COMMAND REJECTED.  THERE ARE NO ENTRIES IN THE FILTER TABLE.`
Meaning: An attempt was made to delete an entry from the filter table, but the table contained no entries. **Action:**   None
`THIS COMMAND WILL NOT DO ANYTHING.  THE FILTER FUNCTION IS ALREADY` `ENABLED.`
Meaning: The filter command string was entered more than once. **Action:**   None

help

Function

Use the help command to receive online documentation for the AMADUMP
directory.

help command parameters and variables	
Command	**Parameters and variables**
help	_all_ amadump _command_nam_

Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
amadump	This parameter produces a description of the AMADUMP directory.
_command_nam_	This variable specifies a valid AMADUMP directory command name. When the _command_nam_ variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

help (end)

Example of the help command	
Example	**Task, response, and explanation**
help amadump ↵	
	Task: Access online documentation.
	Response: This tool allows the user to display the contents of files containing billing records by using the "DUMP" subcommand. Billing records of various formats can be handled. Record search capabilities are also provided with the "FILTER" subcommand. Enter "HELP DUMP", or "HELP FILTER" from the AMADUMP level for more information. Parms: <FORMAT> STRING <FILENAME> STRING Special purpose filenames: AMA_ACTIVE - Opens the currrently mounted active AMA file. AMA_PARALLE - Opens the currently mounted parallel AMA file. CALLDUMP - Allows use of AMADUMP filters for CALLDUMP.
	Explanation: This example typifies a response for the help command string.

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

quit

Function

Use the quit command to exit the AMADUMP directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⎡ *1 level* ⎤ ⎢ all ⎥ ⎢ *name* ⎥ ⎣ *n_levels* ⎦
Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
CI:	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
QUIT -- Increment not found	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
QUIT -- Unable to quit requested number of levels	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

AMREPCI level commands

Use the AMREPCI of the MAP to query and change the central processing unit (CPU) occupancy threshold. In addition, the AMREPCI directory amreped command produces the maintenance manager's morning report.

Accessing the AMREPCI level

To access the AMREPCI level, enter the following command from the CI level:

amrepci ↵

AMREPCI commands

The commands available at the AMREPCI MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

AMREPCI commands	
Command	Page
amreped	A-309
help	A-313
querycputhresh	A-315
quit	A-317
setcputhresh	A-321

amreped

Function

Use the amreped command to include or exclude an item from the
maintenance manager's morning report. The amreped command also can be
used to list the items that currently are part of the report.

amreped command parameters and variables		
Command	**Parameters and variables**	
amreped	add del list	$\lceil$ *itemname* $\rceil$ $\lfloor$ $\rfloor$

Parameters and variables	**Description**
add	This parameter includes the specified item in the maintenance manager's morning report.
del	This parameter excludes the specified item from the maintenance manager's morning report.
-continued-	

amreped (continued)

amreped command parameters and variables (continued)	
Parameters and variables	**Description**
itemname	This variable specifies the name of the item to added or deleted from the maintenance manager's morning report. The valid entry values include the following:
	▪ spms switch performance monitoring system (SPMS) indicators
	▪ cpperf call processing performance
	▪ cpu CPU occupancy
	▪ swact peripheral module (PM) switch of activity (SWACT) count
	▪ swertrap trap/software error (SWERR) counts
	▪ logs focus maintenance and operational measurements (OM) count
	▪ netinteg network integrity failure count
	▪ alt automatic line testing (ALT) result
	▪ att automatic trunk testing (ATT) result
	▪ image central controller (CC) image dump result for NT40 switch or computing module image dump result for Supernode switch
	▪ patch patch summary information
	▪ xpmrex XMS-based PM (XPM) routine exercise test (REX) information
	▪ checktab table data information
	▪ cctst CC tests
	▪ outage outage information
list	This parameter displays a list of items currently in the maintenance manager's morning report as well as a list of items that you can add to the list.
End	

Qualifications

None

Example

The following table provides an example of the amreped command.

amreped (continued)

Example of the amreped command	
Example	**Task, response, and explanation**
amreped add ccperf ↵ *where* ccperf specifies the item to be added to the maintenance manager's morning report	
	Task: Add an item to the maintenance manager's morning report.
	Response: `*** CCPERF IS ADDED TO THE REPORT ***`
	Explanation: This command adds the CCPERF item to the maintenance manager's morning report.

Responses

The following table provides explanations of the responses to the amreped command.

Responses for the amreped command	
MAP output	**Meaning and action**
`*** <valid item> IS ADDED TO THE REPORT ***`	
	Meaning: The valid item was added to the maintenance manager's morning report.
	Action: None
`*** <valid item> IS DELETED FROM THE REPORT ***`	
	Meaning: The valid item was deleted from the maintenance manager's morning report.
	Action: None
`*** NO ACTION TAKEN – ITEM IS ALREADY DELETED ***`	
	Meaning: An attempt was made to delete an item that already has been deleted from the report.
	Action: None
-continued-	

amreped (end)

Responses for the amreped command (continued)	
MAP output	**Meaning and action**
`*** NO ACTION TAKEN - ITEM IS ALREADY INCLUDED ***`	
	Meaning: An attempt was made to add an item that already has been added to the report.
	Action: None
End	

help

Function

Use the help command to receive online documentation for the AMREPCI directory.

help command parameters and variables	
Command	**Parameters and variables**
help	_all_ _command_nam_
Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
_command_nam_	This variable specifies a valid AMREPCI directory command name. When the _command_nam_ variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help ↵	
	Task: Access online documentation.

	Response: ```AMREPCI Program - Subcommands are:```
	```*** Setcputhresh```
	```*** Querycputhresh```
	```*** Amreped```
	```*** Quit```

	Explanation: This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

querycputhresh

Function

Use the querycputhresh command to query the active CPU threshold value.

querycputhresh command parameters and variables	
Command	**Parameters and variables**
querycputhresh	There are no parameters or variables.

Qualification

If the CPU threshold value has been changed in the last 24 hours, the new value will not be reflected.

Example

The following table provides an example of the querycputhresh command.

Example of the querycputhresh command		
Example	**Task, response, and explanation**	
querycputhresh ↵		
	Task:	Query the active CPU threshold value.
	Response:	CPU THRESHOLD IS 60 %.
	Explanation:	The active CPU threshold value that is set using the setcputhresh command displays. In this example, the CPU threshold is set at the default value of 60%.

querycputhresh (end)

Response

The following table provides an explanation of the response to the querycputhresh command.

Response for the querycputhresh command	
MAP output	**Meaning and action**
UNDEFINED COMMAND "QUERYCPUTHRESHSH"	
	Meaning: The command was misspelled when it was entered.
	Action: Reissue the command.

quit

Function

Use the quit command to exit the AMREPCI directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	*1 level* all *name* *n_levels*

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)	
Example	**Task, response, and explanation**
quit all ↵	
	Task: Exit from all levels.
	Response: `CI:`
	Explanation: You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory	
	Task: Exit from a specified directory without leaving any other directories.
	Response: `AMADUMP>>>` `>`
	Explanation: The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵	
	Task: Exit from a specified number of levels.
	Response: `CI:`
	Explanation: You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End	

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
CI:	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
QUIT -- Increment not found	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
QUIT -- Unable to quit requested number of levels	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

setcputhresh

Function

Use the setcputhresh command to change the CPU occupancy threshold for the maintenance manager's morning report. The setcputhresh command sets the threshold to reflect the status of the switch.

setcputhresh command parameters and variables	
Command	**Parameters and variables**
setcputhresh	_60_ _percentage_
Parameters and variables	**Description**
_60_	Omitting this entry forces the system to default to a value of 60%.
percentage	This variable specifies the CPU occupancy threshold. The valid entry range is 0-100%.

Qualification

When the CPU occupancy threshold value is changed, the new value is not reflected until the next report is generated.

Example

The following table provides an example of the setcputhresh command.

Example of the setcputhresh command		
Example	**Task, response, and explanation**	
setcputhresh 75 ↵ _where_		
75	specifies the CPU occupancy threshold percentage	
	Task:	Set the CPU threshold.
	Response:	THRESHOLD VALUE HAS BEEN CHANGED TO 75% FROM 60%
	Explanation:	This command resets the CPU threshold from the default value to 75%.

setcputhresh (end)

Response

The following table provides an explanation of the response to the setcputhresh command.

Response for the setcputhresh command	
MAP output	**Meaning and action**
`OUT OF RANGE: <percentage> 1 TO 100%   ENTER: <percentage>`	
	Meaning: The specified percentage is not within the valid entry range.
	Action: Reissue the command specifying a percentage within the range of one to 100%.

AUTOPATCH level commands

Use the AUTOPATCH level of the MAP to control automatic application of patches.

Accessing the AUTOPATCH level

To access the AUTOPATCH level, enter the following command from the CI level:

autopatch ↵

AUTOPATCH commands

The commands available at the AUTOPATCH MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

AUTOPATCH commands	
Command	**Page**
cancel	A-325
delay	A-327
help	A-329
inhibit	A-331
query	A-335
quit	A-337
schedule	A-341
start	A-345
stop	A-349

cancel

Function

Use the cancel command to cancel autopatch sessions.

cancel command parameters and variables	
Command	**Parameters and variables**
cancel	There are no parameters or variables.

Qualification

If the cancel command is entered while the autopatch process is running, the current autopatch session continues to apply patches; future autopatch sessions are canceled.

Example

The following table provides an example of the cancel command.

Example of the cancel command		
Example	**Task, response, and explanation**	
cancel ↵		
	Task:	Cancel autopatch sessions.
	Response:	THE AUTOPATCH PROCESS IS RUNNING... THIS AUTOPATCH SESSION WILL CONTINUE... FUTURE AUTOPATCHING SESSIONS WILL BE CANCELED... PLEASE CONFIRM Y/N. >Y FUTURE AUTOPATCHING SESSIONS HAVE BEEN CANCELED.
	Explanation:	If the autopatch process is running and the cancel command is entered, future autopatch sessions are canceled when confirmation is received from the user.

cancel (end)

Responses

The following table provides explanations of the responses to the cancel command.

Responses for the cancel command
MAP output **Meaning and action**
| ```
THE AUTOPATCH PROCESS IS RUNNING... THIS AUTOPATCH SESSION WILL
CONTINUE... FUTURE AUTOPATCHING SESSIONS WILL BE CANCELED... PLEASE
CONFIRM    Y/N.
>N
FUTURE AUTOPATCHING SESSIONS ARE NOT CANCELED.
``` |
| **Meaning:** When the cancel command is entered, the system prompts for a confirmation of this action. |
| **Action:** Respond to the confirmation prompt. Entering Y cancels autopatch sessions. Entering N aborts the command and does not cancel autopatch sessions. |
| ```
THE AUTOPATCH PROCESS SCHEDULED TO RUN AT 09:00 WILL BE CANCELED...
FUTURE AUTOPATCHING SESSIONS WILL BE CANCELED. PLEASE CONFIRM    Y/N.
>Y
AUTOPATCHING SESSIONS HAVE BEEN CANCELED.
``` |
| **Meaning:** When the cancel command is entered, the system prompts for a confirmation of this action. |
| **Action:** Respond to the confirmation prompt. Entering Y cancels autopatch sessions. Entering N aborts the command and does not cancel autopatch sessions. |

delay

Function

Use the delay command to cancel the next scheduled wakeup of the auto apply feature.

delay command parameters and variables	
Command	**Parameters and variables**
delay	There are no parameters or variables.

Qualification

The delay command is accumulative. If the auto apply feature is scheduled to run on Monday, Wednesday, and Friday, and the delay command is entered twice on Monday, the auto apply feature will not run again until Friday.

Example

The following table provides an example of the delay command.

Example of the delay command	
Example	**Task, response, and explanation**
delay ↵	
	Task: Cancel the next scheduled run of the auto apply feature.

	Response: `THE AUTOPATCH PROCESS IS SCHEDULED TO RUN ON`
	`WEDNESDAY...DO YOU WISH TO DELAY THE PROCESS`
	`UNTIL FRIDAY INSTEAD?`
	`PLEASE CONFIRM ("YES" OR "NO"):`
	`>YES`
	`THE AUTOPATCH PROCESS HAS BEEN RESCHEDULED TO RUN`
	`ON FRIDAY.`

	Explanation: The command executed properly. The auto apply process does not run until the next scheduled day as defined by DAYOFWK field in table PATSET.

Responses

The following table provides explanations of the responses to the delay command.

delay (end)

Responses for the delay command	
MAP output	**Meaning and action**
``` THE AUTOPATCH PROCESS IS SCHEDULED TO RUN ON WEDNESDAY...DO YOU WISH TO DELAY THE PROCESS UNTIL FRIDAY INSTEAD? PLEASE CONFIRM ("YES" OR "NO"): >NO THE AUTOPATCH PROCESS HAS NOT BEEN DELAYED. ```	
	Meaning: You aborted the delay command before it executed.
	Action: None
`Autopatched is not scheduled to run, cannot be delayed`	
	Meaning: The autopatch process is not scheduled to run.
	Action: None

help

Function

Use the help command to receive online documentation for the
AUTOPATCH directory.

help command parameters and variables	
Command	**Parameters and variables**
help	_all_ autopatch _command_nam_
Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
autopatch	This parameter produces a short description of the function of the AUTOPATCH directory and lists the valid commands.
_command_nam_	This variable specifies a valid AUTOPATCH directory command name. When the _command_nam_ variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

help (end)

Example of the help command	
Example	**Task, response, and explanation**
help ↵	
	Task: Access online documentation.
	Response: Autopatch Utility -- (beside quit) subcommands are: START STOP QUERY DELAY CANCEL SCHEDULE INHIBIT
	Explanation: This example typifies a response for the help command string.

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

inhibit

Function

Use the inhibit command to control the maintenance sanity check for one log or for a group of log reports. The maintenance sanity check ignores any log that has the inhibit status turned on.

inhibit command parameters and variables	
Command	**Parameters and variables**
inhibit	*all*
	log $\begin{bmatrix} query\_log \\ inhibit\_log \begin{bmatrix} on \\ off \end{bmatrix} \end{bmatrix}$
	loggrp $\begin{bmatrix} query\_grp \\ inhibit\_grp \begin{bmatrix} on \\ off \end{bmatrix} \end{bmatrix}$
Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying all logs that have the inhibit status turned on.
inhibit_grp	This variable specifies the log report group to inhibited.
inhibit_log	This variable specifies the log report to inhibited.
log	This parameter either queries or activates the inhibit status of one log report.
loggrp	This parameter either queries or activates the inhibit status of a log report group.
off	This parameter turns off the inhibit status for a specified log report or log report group.
on	This parameter turns on the inhibit status for a specified log report or log report group.
query_grp	This variable queries the log report group and displays whether the inhibit status is turned on or off.
query_log	This variable specifies a particular log report and displays whether the inhibit status is turned on or off.

Qualifications

None

inhibit (continued)

Examples

The following table provides examples of the inhibit command.

Examples of the inhibit command	
Example	**Task, response, and explanation**
inhibit ↵	

	Task:	Display logs that have the inhibit status turned on.
	Response:	THE INHIBIT STATUS IS ON FOR THE FOLLOWING LOG REPORTS: PM 100 PM 101 PM 102 PM 103 PM 104 PM 105 PM 106 PM 107 PM 108 PM 109 PM 110 PM 111 PM 112 PM 113 PM 114 PM 115 PM 116 PM 117 PM 118 PM 119 PM 120 PM 121 PM 122 PM 124 PM 125 PM 126 PM 127 PM 128 PM 129 PM 130 PM 230 PM 131 PM 132 PM 139 PM 171 PM 179 PM 180 PM 181 PM 182 PM 183 PM 184 PM 185 PM 186 PM 187 PM 188 PM 189 PM 271 PM 777 PM 221 PM 222 PM 223 PM 190 PM 191 PM 192 PM 193 PM 194 PM 195 PM 196 PM 197 PM 198 PM 199 PM 200 PM 235 PM 270 PM 231 PM 232 PM 233 PM 234 PM 236 PM 210 PM 211 PM 212 PM 213 PM 214 PM 215 PM 216 PM 217 PM 218 PM 219 PM 220 PM 140 PM 141 PM 150 PM 151 PM 152 PM 153 PM 154 PM 160 PM 161 PM 165 PM 162 PM 163 PM 164 PM 170 PM 166 PM 167 PM 168 PM 169
	Explanation:	This command defaults to displaying all logs that have the inhibit status turned on.

inhibit log pm102 on ↵ *where*		
pm102	specifies the log report group to be inhibited	
	Task:	Turn on the inhibit status for a log report.
	Response:	PM 102 INHIBIT STATUS IS ALREADY ON.
	Explanation:	The inhibit status already is turned on for the log report named PM 102.
	End	

Response

The following table provides an explanation of the response to the inhibit command.

inhibit (end)

Response for the inhibit command	
MAP output	**Meaning and action**
THE INHIBIT STATUS IS ON FOR THE FOLLOWING LOG REPORTS: NO LOG REPORTS ARE INHIBITED.	
	Meaning: The autopatch maintenance sanity check ignores logs with an active inhibit status. This message indicates that no log reports are inhibited and that all log reports are included in the sanity check. **Action:** None

query

Function

Use the query command to query the status of the autopatch process. The query command produces a display that shows whether the autopatch process is running or scheduled to run. If the process is scheduled to run, this command displays the scheduled start time. If the full parameter is used, the results from the previous autopatch session display.

query command parameters and variables	
Command	**Parameters and variables**
query	*status* full
Parameters and variables	**Description**
status	Omitting this entry forces the system to default to displaying whether the autopatch process is running.
full	This parameter displays results from the previous autopatch session. Do not enter the query full command string until the autopatch process executes.

Qualification

An error message is produced if you use the full parameter before the autopatch process executes.

Example

The following table provides an example of the query command.

query (end)

Example of the query command	
Example	**Task, response, and explanation**
query full ↵	
	Task: Display the results from the last autopatch session.

	Response: `AUTOPATCH SUMMARY OF SESSION RAN ON`
	`THU.02/AUG/1990  13:30:56`
	`FIRST APPLIED PATCH:     BNG03C32   13:10:27`
	` LAST APPLIED PATCH:     SFD82C32   13:18:09`
	`FULLY APPLIED:  2`
	`FAILED:  2    BAK30C32`
	`                NOP26C32`
	`PARTIALLY APPLIED:  0`
	`APPLY MANUALLY:  1    GHV04C32`
	`PENDING:  2    BNG56C32`
	`                YUI09C32`

	Explanation: This response provides a summary of sample autopatch sessions from August 2, 1990.

Response

The following table provides an explanation of the response to the query command.

Response for the query command	
MAP output	**Meaning and action**
`THE AUTOPATCH PROCESS IS RUNNING. USE THE FULL OPTION AFTER IT HAS` `COMPLETED EXECUTION.`	
	Meaning: The query full command string was entered during an active autopatch session.
	Action: Wait for the autopatch session to finish before entering the query full command string.

quit

Function

Use the quit command to exit the AUTOPATCH directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⎡ *1 level* ⎤ ⎢ all ⎥ ⎢ *name* ⎥ ⎣ *n_levels* ⎦

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
CI:	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
QUIT -- Increment not found	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
QUIT -- Unable to quit requested number of levels	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

schedule

Function

Use the schedule command to run the autopatch process every 24 hours based on specified start and end times. Although there are no parameters or variables for this command, the system prompts for activity confirmation responses after the schedule command is entered. The responses include the start and end times of the new schedule. The start and end times entered with this command are stored in Table PATSET.

schedule command parameters and variables	
Command	**Parameters and variables**
schedule	There are no parameters or variables.

Qualifications

The schedule command is qualified by the following exceptions, restrictions, and limitations:

- If an autopatch session is active when this command is entered, the session continues to run and the next autopatch session takes place during the new scheduled time.

- If the clock time is beyond the new start and end times applied by the schedule command, the schedule change takes place the following day.

Example

The following table provides an example of the schedule command.

schedule (continued)

Example of the schedule command	
Example	**Task, response, and explanation**
schedule ↵	
	Task: Request an autopatch schedule change.
	Response: THE AUTOPATCH PROCESS IS RUNNING...THIS AUTOPATCH SESSION WILL CONTINUE...START AND END TIMES WILL BE CHANGED. DO YOU WISH TO CONTINUE? Y/N >Y START: 0100 >0300 END: 0200 >0600 START 0300 END 0600 PLEASE CONFIRM Y/N >Y START AND END TIMES CHANGED.
	Explanation: An autopatch schedule change was requested. In response to the prompts, new start and end times were entered. The new time changes were made to Table PATSET.

Responses

The following table provides explanations of the responses to the schedule command.

Responses for the schedule command	
MAP output	**Meaning and action**
THE AUTOPATCH PROCESS IS RUNNING...THIS AUTOPATCH SESSION WILL CONTINUE...START AND END TIMES WILL BE CHANGED. DO YOU WISH TO CONTINUE? Y/N >N	
	Meaning: The autopatch session was running, but the requestor decided not to continue this action.
	Action: Entering Y in response to the activity confirmation prompt allows you to set new start and end times. Entering N in response to the activity confirmation prompt aborts this action.
	-continued-

schedule (end)

Responses for the schedule command (continued)

MAP output   Meaning and action

```
THE AUTOPATCH PROCESS SCHEDULED TO RUN AT 09:00 WILL BE CANCELED...
FUTURE AUTOPATCHING SESSIONS WILL BE CANCELED. PLEASE CONFIRM Y/N.
>Y
AUTOPATCHING SESSIONS HAVE BEEN CANCELED.
```

Meaning: The autopatch process was not running, the schedule command was entered, and the requestor confirmed the cancellation of future autopatch sessions.

Action: Entering Y in response to the activity confirmation prompt allows you to set new start and end times. Entering N in response to the activity confirmation prompt aborts this action.

End

start

Function

Use the start command to bypass the scheduled start time in Table PATSET and begin the auto apply feature immediately. If an auto apply session was scheduled to run later the same day, it is cancelled and rescheduled for the next scheduled auto apply session. The system prompts for activity confirmation responses after the start command is entered.

start command parameters and variables	
Command	**Parameters and variables**
start	*apply patches* now

Parameters and variables	**Description**
apply patches	Omitting this entry forces the system to begin immediately, ignoring the time specified in table PATSET. Once started, the end time also is ignored and the process runs until all indicated patches are applied unless the process is stopped by command or an error condition. The process runs exactly as it does when it runs automatically. That is, maintenance sanity prechecks are performed, broadcast start-up warning messages are displayed, the front-end sync status is changed (if necessary), and all other activities are performed according to the contents of table PATSET.
now	This parameter causes the auto apply feature to begin patch application. No warning messages are displayed, no maintenance sanity checks are performed, and the CC/CM sync status is not altered. You are responsible for verifying that the condition of the office is appropriate.

Qualifications

None

Examples

The following table provides examples of the start command.

start (continued)

Examples of the start command	
Example	**Task, response, and explanation**
start now ↵	
	Task: Run the autopatch process without warnings.

	Response: `THE AUTOPATCH PROCESS IS SCHEDULED TO RUN AT`
	`09:00... DO YOU WISH TO RUN THE PROCESS NOW`
	`INSTEAD?  Y/N`
	`>Y`
	
	`THE AUTOPATCH PROCESS WILL RUN AGAIN ON MAY 9 AT`
	`09:00.`

	Explanation: The autopatch start now command string was entered and the response to the confirmation prompt was affirmative. The autopatch process applied patches listed within table PATCTRL.
start ↵	
	Task: Run the autopatch process with warnings.

	Response: `THE AUTOPATCH PROCESS SCHEDULED TO RUN AT 23:00`
	`WILL INSTEAD BE RUN NOW.`
	`THE FULL OPTION WILL INVOKE ALL AUTOPATCH`
	`FUNCTIONS.`
	`DO YOU WISH TO RUN THE PROCESS NOW?`
	`PLEASE CONFIRM ("YES" or "NO"):`
	`>YES`
	`THE AUTOPATCH PROCESS HAS BEEN STARTED..PLEASE`
	`MONITOR THE LOGUTIL PCH LOGS... THE AUTOPATCH`
	`PROCESS HAS BEEN COMPLETED.`
	
	`THE AUTOPATCH PROCESS WILL RUN AGAIN ON MAY 9 AT`
	`09:00.`

	Explanation: The autopatch start command was entered. This wakes up the auto apply process and causes execution. The autopatch process executed properly. Monitor all applicable logs and screen messages.

start (end)

Responses

The following table provides explanations of the responses to the start command.

Responses for the start command
MAP output   **Meaning and action**
THE AUTOPATCH PROCESS IS SCHEDULED TO RUN AT 09:00... DO YOU WISH TO RUN THE PROCESS NOW?   Y/N   >N   AUTOPATCH PROCESS NOT STARTED. THE AUTOPATCH PROCESS WILL RUN ON MAY 8 AT 09:00.
Meaning: The response to the process confirmation prompt was negative. The autopatch process did not start. **Action:**   None
Autopatcher is not scheduled to run today.   You must alter the DAYOFWK field in table PATSET.
Meaning: The response indicates that the autopatch process is not scheduled to run. The autopatch process did not start. **Action:**   Correct the DAYOFWK field in table PATSET or take no action.

stop

Function

Use the stop command to stop the autopatch process from running. Although there are no parameters or variables for this command, the system prompts for activity confirmation responses after the stop command is entered.

stop command parameters and variables	
Command	**Parameters and variables**
stop	There are no parameters or variables.

Qualification

The stop command prevents patches from being applied only while the autopatch process is running.

Example

The following table provides an example of the stop command.

Example of the stop command		
Example	**Task, response, and explanation**	
stop ↵		
	Task:	Stop the autopatch process.
	Response:	THE AUTOPATCH PROCESS IS RUNNING...THE AUTOPATCH PROCESS WILL BE HALTED...PLEASE CONFIRM Y/N. >Y THE AUTOPATCH PROCESS HAS BEEN HALTED. THE AUTOPATCH PROCESS WILL RUN AGAIN ON MAY 9 AT 09:00.
	Explanation:	The stop command was entered during an active autopatch session and the requestor responded to the confirmation prompt with an affirmative entry. The autopatch process stopped.

stop (end)

Responses

The following table provides explanations of the responses to the stop command.

Responses for the stop command	
MAP output	**Meaning and action**
THE AUTOPATCH PROCESS IS RUNNING...THE AUTOPATCH PROCESS WILL BE HALTED... PLEASE CONFIRM Y/N. >N THE AUTOPATCH PROCESS WILL NOT BE HALTED. AUTOPATCHING WILL CONTINUE. THE AUTOPATCH PROCESS WILL RUN AGAIN ON MAY 9 AT 09:00.	
	Meaning: The stop command was entered during an active autopatch session and the requestor responded to the confirmation prompt with a negative entry.
	Action: Entering Y causes the autopatch process to stop. Entering N causes the command to abort and the autopatch process continues.
THE AUTOPATCH PROCESS IS NOT RUNNING...THE AUTOPATCH PROCESS IS SCHEDULED TO RUN AT 09:00...USE THE CANCEL COMMAND TO CANCEL AUTOPATCH SESSIONS.	
	Meaning: The autopatch process was not running and the stop command was entered.
	Action: Use the cancel command to cancel autopatch sessions.

AUTOTABAUDIT level commands

Use the AUTOTABAUDIT level of the MAP to to check data integrity without external guidance. AUTOTABAUDIT produces three types of reports. These reports consist of generic table checks, syntax checks, and table-specific data checks including routing checks. The reports are generated for each table as it is being verified. The reports are maintained and displayed using a report utility.

AUTOTABAUDIT provides similar functions to those commands provided by the TABAUDIT directory. In addition, the AUTOTABAUDIT subdirectory provides two additional commands, the terminate command the the timeframe command. The terminate command terminates the AUTOTABAUDIT session before table verification continues. The timeframe command allows you to define the start time for the system to begin processing data integrity checks for specified tables automatically.

Note: AUTOTABAUDIT cannot execute at the same time as TABXFR or an image dump. Also, only one AUTOTABAUDIT session can execute at a time.

Accessing the AUTOTABAUDIT level

To access the AUTOTABAUDIT level, enter the following from the TABAUDIT level:

auto ↵

AUTOTABAUDIT commands

The commands available at the AUTOTABAUDIT MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

AUTOTABAUDIT commands	
Command	**Page**
clear	A-353
exclude	A-355
-continued-	

AUTOTABAUDIT commands (continued)	
Command	**Page**
execute	A-357
help	A-361
include	A-365
info	A-367
quit	A-371
report	A-375
status	A-379
terminate	A-383
timeframe	A-385
End	

clear

Function

Use the clear command to reset protected variables for AUTOTABAUDIT. This command either clears the AUTOTABAUDIT schedule that was previously defined by the timeframe command, clears the included tables list that was previously defined using the include command, or clears both.

clear command parameters and variables	
Command	**Parameters and variables**
clear	all schedule *mode* included

Parameters and variables	**Description**
all	This parameter clears both the schedule data and the included table data's protected store.
included	This parameter clears the list of included tables.
mode	This variable specifies which schedule or schedules to clear. The valid entry values are one and all.
schedule	This parameter clears the AUTOTABAUDIT schedule data from protected store.

Qualifications

None

Example

The following table provides an example of the clear command.

Example of the clear command		
Example	**Task, response, and explanation**	
clear all ↵		
	Task:	Clear both the schedule data and the included table data's protected store.
	Response:	None
	Explanation:	You issued the clear all command.
	-continued-	

clear (end)

Example of the clear command (continued)	
Example	**Task, response, and explanation**
clear schedule one ↵	
	Task: Clear specified schedule data.
	Response: -- Select on of the following timeframes to be cleared. -- 1. 1993/10/31 1993/10/31 12:00 14:00 -- Please select a timeframe to be cleared [1 - 1] >1
	Explanation: This display lists the existing schedules and prompts you to specify the time frame to clear.
End	

Response

The following table provides an explanation of the response to the clear command.

Response for the clear command
MAP output   Meaning and action
-- Select on of the following timeframes to be cleared. -- No Timeframes Specified. --
Meaning: There are no AUTOTABAUDIT sessions scheduled.
Action: None.

exclude

Function

Use the exclude command to specify a table that is not to be checked by AUTOTABAUDIT. This command also provides the ability to remove a table or all tables from the list of executed tables.

exclude command parameters and variables	
Command	**Parameters and variables**
exclude	*tablename*
Parameters and variables	**Description**
tablename	This variable specifies the table that should not be verified.

Qualification

You cannot change an active AUTOTABAUDIT session's parameters without first terminating the session.

Example

The following table provides an example of the exclude command.

Example of the exclude command		
Example	**Task, response, and explanation**	
exclude trkmem ↵ *where*		
trkmem	specifies the name of the table that should not be verified	
	Task:	Exclude a specified table.
	Response:	None
	Explanation:	The command executed properly. The system does not produce a response display.

Responses

The following table provides explanations of the responses to the exclude command.

exclude (end)

Responses for the exclude command	
MAP output	**Meaning and action**
``` Restored 928 tuples into DART Table DART is now sorted The following is a conflict between the current table order in DART and the order according to the table uses_list. --- HUNTMEM needs to be datafilled prior to ESAPXLA --- 928 tables compared. Error: Not a valid table name: RTCINV ```	
	Meaning: The system identified a table order error and the display specifies the source. In addition, you entered an invalid table name.
	Action: Correct the datafill error, enter a valid table name in the command string, or abort this command.
`Table TRKMEM is already excluded.`	
	Meaning: The table you specified already is excluded.
	Action: Exclude another table or abort this action.

execute

Function

Use the execute command to activate the AUTOTABAUDIT scheduler. (The AUTOTABAUDIT scheduler starts the verification process at the start time and stop time specified by the timeframe command.)

The current process parameters display and you are prompted to confirm. If information is missing, you are prompted to enter the appropriate command to specify the required data before continuing.

execute command parameters and variables	
Command	**Parameters and variables**
execute	There are no parameters or variables.

Qualification

The execute command is qualified by the following exceptions, restrictions, and limitations:

- Once the AUTOTABAUDIT session is active, you must terminate the session before changes can be made to the session parameters.
- An AUTOTABAUDIT session cannot execute at the same time as TABXFR or an image dump.
- No two sessions of AUTOTABAUDIT can execute at a time.

Example

The following table provides an example of the execute command.

execute (continued)

Example of the execute command	
Example	**Task, response, and explanation**
execute ↵	
	Task: Activate AUTOTABAUDIT with the current range of tables to be verified.
	Response:

```
                   AUTOMATED TABAUDIT STATUS

Active Timeframe                    Executing Timeframe
Start           Stop                 Start           Stop
Date            Date                 Time            Time

1992/06/30      1992/07/05           1:00            5:00

Current time              :1992/06/23   16:32:05
Automated Tabaudit        :Inactive

The following tables are INCLUDED
 From table LTCINV (52) to table KSETFEAT (420)

The following tables are EXCLUDED
  From table DART (0)     to table XLIUMAP (51)
  From table DNINV (421) to table CNMBTST (1115)

Please confirm ("YES", "Y", "NO" or "N"):
>y

Automated Tabaudit has been activated.
```

| | **Explanation:** The AUTOTABAUDIT session is activated. The scheduler uses supplied data to perform data integrity checks from table LTCINV to table KSETFEAT between 1:00 and 5:00 starting on June 30th, 1992, and every day there after between the same time frame until the last table in the range is verified or 5:00 July 7th, 1992. |

Response

The following table provides an explanation of the response to the execute command.

execute (end)

Response for the execute command	
MAP output	**Meaning and action**
`ERROR:  No tables have been included.` `Request aborted.`	
	Meaning: No tables had been included when you issued the execute command. **Action:** Specify the missing data using the AUTOTABAUDIT directory include command and reissue the execute command.

help

Function

Use the help command to generate a list of all the commands added by the
AUTOTABAUDIT directory. Or, use the help command to display help on
a single command added by the AUTOTABAUDIT directory.

help command parameters and variables	
Command	**Parameters and variables**
help	_all_ autotabaudit _command_nam_

Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
autotabaudit	This parameter produces a short description of the function of the AUTOTABAUDIT directory and lists the valid commands.
_command_nam_	This variable specifies the command for which help is required.

Qualifications

None

Examples

The following table provides examples of the help command.

Examples of the help command	
Example	**Task, response, and explanation**
help execute ↵	
	Task: Display the help for the execute command.

	Response: `EXECUTE command`
	`-------------`
	`Command to activate AUTOTABAUDIT using the user`
	`specified parameters.`
	`     eg1:    EXECUTE`

	Explanation: The definition of the execute command is displayed.
-continued-	

help (continued)

Examples of the help command (continued)

Example   **Task, response, and explanation**

---

help autotabaudit ↵

Task:	Display the commands added by the AUTOTABAUDIT directory.
Response:	The AUTO command is used to enter the AUTOTABAUDIT increment from the TABAUDIT increment.
	The AUTOTABAUDIT increment is used to perform automated data integrity checks on a DMS switch.
	The following is a typical scenario: >INCLUDE FROM custflds TO ovcar >TIMEFRAME 2:20 22:11:1993 5:30 28:11:1993 >EXECUTE
	These commands will cause all tables between an including CUSTFLDS and OVCAR to be verified between the specified hours.
	The increment consists of the following subcommands:
	INCLUDE EXCLUDE STATUS REPORT CLEAR TIMEFRAME EXECUTE TERMINATE QUIT HELP INFO
Explanation:	The commands for the AUTOTABAUDIT increment are displayed.

-continued-

---

help (end)

Examples of the help command (continued)		
Example	**Task, response, and explanation**	
help ↵		
	Task:	Display the commands added by the AUTOTABAUDIT directory.
	Response:	The AUTO command is used to setup a scheduled session of AUTOTABAUDIT. The increment consists of the following subcommands: INCLUDE EXCLUDE STATUS REPORT CLEAR TIMEFRAME EXECUTE TERMINATE QUIT HELP INFO From within the AUTOTABAUDIT increment type: HELP <subcommand> for further help on subcommand.
	Explanation:	The commands for the AUTOTABAUDIT increment are displayed.
End		

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded.
	Action: None

include

Function

Use the include command to specify the range of tables to be verified by AUTOTABAUDIT. This command allows the specification of a single table to verify, a range of tables according to DART, or all tables on a DMS switch.

The include command is cumulative in the sense that the successive includes can be used to generate a complex includes list. In other words, including tables does not clear the previous includes list. In this manner, you can generate a list of tables to verify that do not necessarily have to follow a given sequence of tables in Table DART.

include command parameters and variables	
Command	**Parameters and variables**
include	all *tablename(s)* from *start_table* ⌈ *last dart table* ⌊ to *stop_table*

Parameters and variables	Description
last dart table	Omitting this entry forces the system to default to using the last table in DART if the last table in the range of tables to verify it is not specified.
all	This parameter verifies all tables on a DMS switch.
from	This parameter verifies a range of tables beginning with the table specified after this parameter.
start_table	This parameter specifies the table with which to start data verification. (The tables are verified following the order in Table DART.)
stop_table	This parameter specifies the last table to verify in the range of tables. If this table name is not specified, the system defaults to using the last table in DART.
tablename(s)	This parameter specifies the table or tables to verify. A series of tables to be included can be specified on a single line with each table name being separated by a space.
to	This parameter indicates a range of tables will be verified by separating the *start_table* variable replacement value from the *stop_table* variable replacement value.

include (end)

Qualification

Once an AUTOTABAUDIT session is active, you may not change a session's parameters without first terminating the session.

Example

The following table provides an example of the include command.

Example of the include command	
Example	**Task, response, and explanation**
include **ofcopt** ↵ *where* ofcopt specifies the name of the table	
	Task: Verify a single table.
	Response: None
	Explanation: This command adds table OFCOPT to the list of those to be verified.

Response

The following table provides an explanation of the response to the include command.

Response for the include command	
MAP output	**Meaning and action**
ERROR: Not a valid end table name.	
	Meaning: You entered an invalid end table name while verifying a range of tables.
	Action: Reissue the include command to verify a range of tables and include a valid end table name.

info

Function

Use the info command to provide a reference where more information about the AUTOTABAUDIT tool is available. This command also provides a general background of what the AUTOTABAUDIT tool is intended to do.

info command parameters and variables	
Command	**Parameters and variables**
info	There are no parameters or variables

Qualifications

None

Example

The following table provides an example of the info command.

info (continued)

Example of the info command	
Example	**Task, response, and explanation**
info ↵	
	Task: Display information describing TABAUDIT.

	Response: `Information about TABAUDIT`
	`------------------------`
	`TABAUDIT performs the following checks with the`
	`DMS switch in sync:`
	`  Generic table checks:  (Performed on a per`
	`  table basis.)  Verify that a table is not`
	`  corrupt.`
	
	`  Syntax checks:  (Performed on a per tuple`
	`  basis.)  Verify that data contained within a`
	`  tuple's fields is consistent syntactically.`
	
	`  Table specific checks:  (Performed on a per`
	`  tuple basis.)  Verify data consistency on the`
	`  tuple.`
	
	`TABAUDIT is intended as a replacement for`
	`CHECKTAB.  CHECKTAB needed to be executed on the`
	`inactive side with the switch out of sync in`
	`order to effectively verify all data in the order`
	`to effectively verify all data in the switch.`
	`AUTOTABAUDIT does not have this limitation.  This`
	`is accomplished by verifying data integrity`
	`without performing nil-writes.`
	
	`For more information, please refer to NTP`
	`297-1001-303, the One Night Process and Hybrid`
	`Software Delivery Procedures document.`
	
	`The increment consists of the following`
	`subcommands:`
	
	`INCLUDE EXCLUDE STATUS REPORT CLEAR TIMEFRAME`
	`EXECUTE TERMINATE QUIT HELP INFO`
	                                           (cont.)

-continued-	

info (end)

Example of the info command (continued)	
Example	**Task, response, and explanation**

	Response:	From within the AUTOTABAUDIT increment type: HELP <subcommand> for further help on subcommand. To set up the AUTOTABAUDIT process: 1. Use the INCLUDE command to specify the range of tables to be verified. 2. Use the TIMEFRAME command to specify when the AUTOTABAUDIT process is to verify the specified range of table. 3. Use the EXECUTE command to activate the AUTOTABAUDIT process. The following is a typical scenario: >INCLUDE all >TIMEFRAME 2:20 22:11:1993 5:30 28:11:1993 >EXECUTE These commands will cause all tables on the DMS to be verified betwen the specified hours during the specified range of days.
	Explanation:	TABAUDIT information displays.

End

Response

The following table provides an explanation of the response to the info command.

Response for the info command	
MAP output	**Meaning and action**
Undefined command "INFORMATION"	
	Meaning: This variation of the info command is invalid.
	Action: Enter the command correctly or abort this action.

quit

Function

Use the quit command to exit the AUTOTABAUDIT directory. Leaving the increment will not effect AUTOTABAUDIT.

quit command parameters and variables	
Command	**Parameters and variables**
quit	1 level all n_levels

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. From the AUTOTABAUDIT directory, one level takes you to the TABAUDIT level where you need to use the quit command again in order to reach the CI level.
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit.

Qualification

The quit autotabaudit command string is invalid for this directory. This is because the AUTOTABAUDIT directory is built on top of the TABAUDIT directory and because the AUTOTABAUDIT directory is accessed using the TABAUDIT auto command instead of the directory level name.

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	TABAUDIT:
	Explanation:	You entered the quit command to exit a directory that is accessed from a directory, not directly from the CI level. The system assumes the default value of one directory level and returns you to the TABAUDIT level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

Responses for the quit command	
MAP output	**Meaning and action**
CI:	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
-continued-	

quit (end)

Responses for the quit command (continued)	
MAP output	**Meaning and action**
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.
End	

report

Function

Use the report command to display the data integrity checks performed by an AUTOTABAUDIT session.

---

report command parameters and variables

Command	Parameters and variables

| **report** | all
errors
included
checked
notchecked
tablename | *local window*
devicename filename

local window
filename |

Parameters and variables	Description
local window	Omitting this entry forces the system to default to using the window from which the report command was issued as the destination.
all	This parameter indicates that a data integrity report of all tables in Table DART is to be generated. For each table with tuples failing AUTOTABAUDIT's data integrity checks, the keys of the failed tuples are listed.
checked	This parameter indicates that a report of all the tables that have been checked by AUTOTABAUDIT is to be generated.
devicename	This variable specifies the device to which the data integrity report is to be routed. If no device name and file name are provided, the default destination is the window from which the report command was issued.
errors	This parameter indicates that the reports of all the tables containing errors are to be generated. This report is identical to the report that is generated when the *tablename* variable is used, except that a report is generated for each table that has known errors.
filename	This variable specifies the report file name. If no device name and file name are provided, the default destination is the window from which the report command was issued.
included	This parameter indicates that the reports of all the included tables are to be generated. The report is identical to the report that is generated when the *tablename* variable is used, except that a report is generated for each table in the list of included tables.

-continued-

---

report (continued)

report command parameters and variables (continued)	
Parameters and variables	**Description**
notchecked	This parameter indicates that a report of all the tables that have not been checked by AUTOTABAUDIT is to be generated.
tablename	This variable specifies a single table for which the data integrity report is to be routed.
End	

Qualifications

None

Example

The following table provides an example of the report command.

Example of the report command	
Example	**Task, response, and explanation**
report **ofcopt** ↵ *where* ofcopt specifies a table name	

	Task:	Generate a report for a specified table.
| | **Response:** | ```
DART
table      Table                        Start     Elapsed
name       control|chcked  Pass   Fail  time      time
-------------------------------------------------------
17:OFCOPT  New|  95      95       0  1993/10/30
                                        12:47:31.70   :02.359
-------------------------------------------------------

Total # of Tables Reported for this report option: 1
Total # of Passed Tuples for this report option: 95
Total # of Failed Tuples for this report option: 0
Total # of Verified Tuples for this report option: 95
Total Elapsed time to verify the above tables: :02.359
``` |
| | **Explanation:** | The report ofcopt command string produced a summary of Table OFRT's data integrity. No tuples failed the syntax check. |

report (end)

Response

The following table provides an explanation of the responses to the report command.

Response for the report command	
MAP output	**Meaning and action**
`Error: Not a valid table name: OFCPTT`	
	Meaning: The specified table name is invalid or spelled incorrectly.
	Action: Reissue this command with a valid table name or abort this action.

status

Function

Use the status command to display the included table range as well as their DART indices, the start time and stop time, when the report data was last cleared, whether or not AUTOTABAUDIT is active, and if active, the current table being checked.

status command parameters and variables	
Command	**Parameters and variables**
status	There are no parameters or variables

Qualifications

None

Example

The following table provides an example of the status command.

status (continued)

Example of the status command

Example	Task, response, and explanation

status ↵

	Task:	Display the current information.
	Response:	AUTOTABAUDIT is scheduled to execute during the start and stop period between the start and stop dates.

```
-------------------------------------------------

                 AUTOTABAUDIT STATUS
-------------------------------------------------
  Active Timeframe    |   Executing timeframe
-------------------------------------------------
  Start      Stop     |    Start       Stop
  Date       Date     |    Time        Time
-------------------------------------------------
1993/10/31 1993/10/31 |   12:00       15:00

Current time          : 1993/10/30   11:46:23
AUTOTABAUDIT          :  Inactive

The following tables are included
table AUTOEXEC (920)

The following tables are excluded
From table ACTPATCH (0)  to table NPASPLIT (919)
From table AUTOHIB (921) to table OCFPORT (927)
```

	Explanation:	AUTOTABAUDIT is scheduled to run but is not currently running.

Response

The following table provides an explanation of the response to the status command.

---

status (end)

Response for the status command

MAP output   Meaning and action

```
AUTOTABAUDIT is scheduled to execute during the start and stop period
between the start and stop dates.
---------------------------------------------------

          AUTOTABAUDIT STATUS
---------------------------------------------------
  Active Timeframe     |   Executing timeframe
---------------------------------------------------
  Start       Stop |    Start       Stop
  Date        Date |    Time        Time
---------------------------------------------------
No Timeframe Specified
---------------------------------------------------


Current time      : 1993/10/30  11:46:23
AUTOTABAUDIT      :  Inactive

The following tables are included
table AUTOEXEC (920)

The following tables are excluded
From table ACTPATCH (0)  to table NPASPLIT (919)
From table AUTOHIB (921) to table OCFPORT (927)
```

Meaning: No AUTOTABAUDIT session is scheduled.

Action: None

terminate

Function

Use the terminate command to deactivate an AUTOTABAUDIT session.

terminate command parameters and variables	
Command	**Parameters and variables**
terminate	There are no parameters or variables.

Qualification

This command only is available from within the AUTOTABAUDIT directory.

Example

The following table provides an example of the terminate command.

Example of the terminate command		
Example	**Task, response, and explanation**	
terminate ↵		
	Task:	Deactivate AUTOTABAUDIT.
	Response:	`AUTOTABAUDIT has been deactivated.`
	Explanation:	AUTOTABAUDIT no longer is performing data integrity checks during the specified time frame.

Responses

The following table provides explanations of the responses to the terminate command.

Responses for the terminate command	
MAP output	**Meaning and action**
`AUTOTABAUDIT is not active.` `Request aborted.`	
	Meaning: You tried to terminate an AUTOTABAUDIT session when no session was active.
	Action: None

timeframe

Function

Use the timeframe command to specify the start and stop time for AUTOTABAUDIT. You specify the date, hour, and minutes that AUTOTABAUDIT is to start and stop processing tables. In addition, you provide a start date and stop date. AUTOTABAUDIT's scheduler uses this information to schedule the next AUTOTABAUDIT process activation.

timeframe command parameters and variables		
Command	**Parameters and variables**	
timeframe	*start_time* <u>*next instance*</u> *start_date* *stop_time* <u>*next instance plus one year*</u> *stop_date*	

Parameters and variables	**Description**
<u>*next instance*</u>	Omitting this entry forces the system to default to executing AUTOTABAUDIT the next instance of the specified start time.
<u>*next instance plus one year*</u>	Omitting this entry forces the system to default to stopping execution of AUTOTABAUDIT the next instance of the specified start time plus one year.
start_date	This variable specifies the day on which TABAUDIT is to start execution. The valid entry value is three sets of two integers separated by a colon in a DD:MM:YY format, such as 15:06:92.
start_time	This variable specifies the hour and minute when TABAUDIT is to start execution. The valid entry value is two sets of integers separated by a colon, such as 2:03.
stop_date	This variable specifies the day on which TABAUDIT is to stop execution. The valid entry value is three sets of two integers separated by a colon in a DD:MM:YY format, such as 15:06:92.
stop_time	This variable specifies the hour and minute when TABAUDIT is to stop execution. The valid entry value is two sets of integers separated by a colon, such as 2:03.

Qualification

This command only is available from within the AUTOTABAUDIT directory.

Example

The following table provides an example of the timeframe command.

timeframe (end)

Example of the timeframe command

Example   **Task, response, and explanation**

---

timeframe 15:30 17:07:92 18:00 17:07:92 ↵

	Task:	Set a timeframe for AUTOTABAUDIT execution.
	Response:	Is the following schedule correct?
		Automated Tabaudit is to execute from 15:30 to 18:00 between the following dates:
		Start date: 1993/07/17 Stop date: 1993/07/17
		NOTE: Please enter EXECUTE to activate AUTOTABAUDIT once all timeframes have been specified.
		Please confirm ("YES", "Y", "NO", or "N"): >Y
	Explanation:	You confirmed that you want to initiate AUTOTABAUDIT at 3:30 P.M. on July 17th, 1993, and terminate processing at 6:00 P.M. on July, 17th, 1993.

Response

The following table provides an explanation of the response to the timeframe command.

Response for the timeframe command

MAP output   **Meaning and action**

The TIMRFRAME has been cleared.

	Meaning: You responded no or n to the confirmation prompt.
	Action: None

BCSMON level commands

Use the BCSMON level of the MAP to dump batch change supplement (BCS) monitoring data.

Accessing the BCSMON level

To access the BCSMON level, enter the following command from the CI level:

bcsmon ↵

Note: You cannot run the BCSMON level commands during a dump. You see the message `BCSMON -- COMMAND DISALLOWED DURING DUMP`.

BCSMON commands

Tthe commands available at the BCSMON MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

BCSMON commands	
Command	**Page**
assess	B-3
dblocks	B-7
dumpall	B-9
eqpcounts	B-11
help	B-15
highcpocc	B-17
highlogs	B-19
highparms	B-21
logbuffer	B-23
logcount	B-27
-continued-	

assess

Function

Use the assess command to display operational measurement (OM) peg data normalized perr 10 000 calls.

assess command parameters and variables	
Command	**Parameters and variables**
assess	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the assess command.

Example of the assess command	
Example	**Task, response, and explanation**
assess ↵	
	Task: Assess the OM peg data.

```
Response:
Office CLLI: COMD
BCS Release: 36
Polled from: 1993/02/12 09:00 to 1993/02/12 10:30
Duration   : 0 days 01 hr. 30 min.
***********************
*       ASSESS        *
***********************

Trunk Originated Calls 1, Line Originated Calls 65.

Notes:
 1. Peg Count are Normalized Per 10000 Calls
 2. Target Peg Count are Normalized Per 10000 Calls
 3. Faults are not normalized.
 4. NM is composed on NMC, NMMSG and NMSPCH.
 5. CMC Err is composed of CMC and CMCLINK.
```

-continued-

assess (continued)

Example of the assess command (continued)

Example	Task, response, and explanation

Response:

```
          MTM Err  MTM Flt  STM Err  STM Flt  LCM Err  LCM Flt

Peg Counts:    0        0        0        0        0       20
Normalized:  0.0      ---      0.0      ---      0.0      ---
Targets   :  1.0      0.0      1.0      0.0      1.0      0.0

          DTC Err  DTC Flt  LTC Err  LTC Flt MSB7 Err MSB7 Flt

Peg Counts:    4        0      554        0        1        0
Normalized:606.0      --- 83939.3      ---    151.5      ---
Targets   :  1.0      0.0      1.0      0.0      1.0      0.

          RMM Err  RMM Flt  RCC Err  RCC Flt LCMI Err LCMI Flt

Peg Counts:    0        2        2        0        2        4
Normalized:  0.0      ---    303.0      ---    303.0      ---
Targets   :  1.0      0.0      1.0      0.0      1.0      0.0

           NM Err   NM Flt  CMC Err  CMC Flt  CPTraps CPSuicds

Peg Counts:    3        0        0        0        0        0
Normalized:454.5      ---      0.0      ---      0.0      0.0
Targets   :  1.0      0.0      1.0      0.0      1.0      1.0
```

Explanation: This command displays the assessment of OM peg data.

End

assess (end)

Response

The following table provides an explanation of the response to the assess command.

Response for the assess command

MAP output   Meaning and action

```
Office CLLI: COMD
BCS Release: 36
Polled from: 1993/02/12 09:00 to 1993/02/12 10:30
Duration   : 0 days 01 hr. 30 min.
***********************
*        ASSESS        *
***********************

Trunk Originated Calls 1, Line Originated Calls 65.

Notes:
 1. Peg Count are Normalized Per 10000 Calls
 2. Target Peg Count are Normalized Per 10000 Calls
 3. Faults are not normalized.
 4. NM is composed on NMC, NMMSG and NMSPCH.
 5. CMC Err is composed of CMC and CMCLINK.
      MTM Err  MTM Flt  STM Err  STM Flt  LCM Err  LCM Flt

Peg Counts:   0        0        0        0        0       20
Normalized: 0.0      ---      0.0      ---      0.0      ---
Targets  : 1.0      0.0      1.0      0.0      1.0      0.0


.
.
.
         NM Err   NM Flt  CMC Err  CMC Flt  CPTraps CPSuicds

Peg Counts:   3        0        0        0        0        0
Normalized:454.5      ---      0.0      ---      0.0      0.0
Targets  : 1.0      0.0      1.0      0.0      1.0      1.0
```

Meaning: You successfully executed the command.

Action: None

dblocks

Function

Use the dblocks command to display the number of digit blocks used by each digilator pool and the number of digilator pools allocated for one or all of the tables.

dblocks command parameters and variables	
Command	**Parameters and variables**
dblocks	*all* *table*

Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying block information from all valid tables.
table	This variable specifies the table containing the block information. The valid entry values are bglocn, conscrn, hnpacont, nctprt, stdprtct, msgrte, and scrnclas.

Qualifications

None

Example

The following table provides an example of the dblocks command.

Example of the dblocks command	
Example	**Task, response, and explanation**
dblocks conscrn ↵ *where* conscrn	specifies the table name

	Task: Display block information from a specific table.
	
	Response:
	`Number of digit blocks being used by CONSCRN:   226`
	`Number of digit blocks allocated for CONSCRN: 32766`
	`Percent Used:  0.7    Percent available:  99.3`
	
	Explanation: This command shows the number of digit blocks from the CONSCRN Table.

dblocks (end)

Response

The following table provides an explanation of the response to the dblocks command.

Response for the dblocks command	
MAP output	**Meaning and action**
`Specify a valid table name`	
	Meaning: You specified an invalid table name.
	Action: Verify the table name and retry the command.

dumpall

Function

Use the dumpall command to display all BCS monitoring data.

dumpall command	
Command	**Parameters and variables**
dumpall	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the dumpall command.

Example of the dumpall command	
Example	**Task, response, and explanation**
dumpall ↵	
	Task: Display all BCS monitoring data.
	Response: OFFICE ID: CINT OFFICE CLLI: CINT BCS RELEASE: 30 POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30 DURATION: 0 days 05 hr. 00 min. ********************** * Major OMs * ********************** AVN\$ORIG = 5 AVN\$TERM = 1 AVN\$NOUT = 1 CMC\$ERRR = 12 CMC\$LERR = 4 CP1\$LPOV = 123 CP1\$ODEN = 12 CP1\$CCBO = 5 CP1\$WAKO = 44 CPU\$MTCH = 51 CPU\$SYNC = 12 CPU\$WARM = 53 CPU\$COLD = 131 EXT\$OVFL = 51 NMC\$MERR = 12 NMC\$SERR = 24 NMC\$SFLT = 86
	Explanation: You see a dump of all BCS monitoring data.

dumpall (end)

Response

The following table provides an explanation of the response to the dumpall command.

Response for the dumpall command

MAP output	Meaning and action

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
DURATION: 0 days 05 hr. 00 min.
***********************
*      Major OMs      *
***********************
AVN$ORIG = 5    AVN$TERM = 1    AVN$NOUT = 1   CMC$ERRR = 12
CMC$LERR = 4    CP1$LPOV = 123  CP1$ODEN = 12  CP1$CCBO = 5
CP1$WAKO = 44   CPU$MTCH = 51   CPU$SYNC = 12  CPU$WARM = 53
CPU$COLD = 131  EXT$OVFL = 51   NMC$MERR = 12  NMC$SERR = 24
NMC$SFLT = 86
```

Meaning: You executed the command successfully.

Action: None

eqpcounts

Function

Use the eqpcounts command to display equipment counts for various types of equipment. This report can be several pages long, depending on your equipment.

eqpcounts command	
Command	**Parameters and variables**
eqpcounts	*brief* all
Parameters and variables	**Description**
all	This parameter displays all equipment information available.
brief	Omitting this entry forces the system to default to displaying the equipment counts.

Qualifications

None

eqcounts (continued)

Example

The following table provides an example of the eqcounts command.

Example of the eqcounts command	
Example	**Task, response, and explanation**
eqcounts ↵	
Task:	Display equipment counts.
Response:	OFFICE ID: CINT OFFICE CLLI: CINT BCS RELEASE: 30 POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30 DURATION: 0 days 05 hr. 00 min. ************** * Counts * ************** Number of nodes: 111 Number of networks: 4 Number of MTM PMs: Insv: 4 Comm: 8 Number of OAU PMs: Insv: 0 Comm: 1 Number of LCM PMs: Insv: 4 Comm: 11 Number of MSB7 PMs: Insv: 1 Comm: 0 Number of RMM PMs: Insv: 1 Comm: 1 Number of IDTC PMs: Insv: 2 Comm: 4 Number of ESA PMs: Insv: 2 Comm: 0 Number of ILGC PMs: Insv: 2 Comm: 2 Number of ILCM PMs: Insv: 1 Comm: 1 Number of DP_POTS lines: 260 Number of DGT_POTS lines: 440
Explanation:	You see equipment counts for the system.

eqpcounts (end)

Response

The following table provides an explanation of the response to the eqpcounts command.

Response for the eqpcounts command	
MAP output	**Meaning and action**
`EITHER incorrect optional parameter(s) OR too many parameters.` `EQPCOUNTS -- Wrong number of parameters`	
	Meaning: You executed the command with a parameter other than all.
	Action: Reenter the command without a parameter or reenter the command with the all parameter.

help

Function

Use the help command to receive online documentation for the BCSMON directory.

help command parameters and variables	
Command	**Parameters and variables**
help	*command_nam*
Parameters and variables	**Description**
command_nam	This variable specifies a valid BCSMON directory command name. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help highcpocc ↵ *where* highcpocc specifies the command name	
	Task: Access online documentation.
	Response: `HIGHCPOCC  :  Displays the high water CP occupancy.`
	Explanation: This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

highcpocc

Function

Use the highcpocc command to display the high water call processing occupancy of the switch for the last thirty days.

highcpocc command	
Command	**Parameters and variables**
highcpocc	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the highcpocc command.

Example of the highcpocc command	
Example	**Task, response, and explanation**
highcpocc ↵	

> **Task:**   Display the high water call processing occupancy.
>
> **Response:**
> ```
> OFFICE ID: CINT
> OFFICE CLLI: CINT
> BCS RELEASE: 30
> POLLED FROM 1989/06/01 22:30 TO 1989/06/29 03:30
> DURATION: 28 days 05 hr. 00 min.
> **************************
> *  HIGH WATER CP OCCUPANCY *
> **************************
> SCAN RATE: 1/MINUTE
>                              TIME
> DATE             0-1   1-2   2-3   3-4   4-5   5-6   ...
> 1989/05/20       15    23    13    16    16    21
> 1989/05/19       23    17    15    19    17    23
> 1989/05/18       19    24    23    12    24    19
> 1989/05/17       26    14    11    17    11    23
> 1989/05/16       29    28    14    13    31    31
> 1989/05/15       19    20    12    11    20    16
> ------------------------------------------------------
> ```
>
> **Explanation:**   This command displays the high water call processing occupancy of the switch for the past thirty days.

highcpocc (end)

Response

The following table provides an explanation of the response to the highcpocc command.

Response for the highcpocc command

MAP output   Meaning and action

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/06/01 22:30 TO 1989/06/29 03:30
DURATION: 28 days 05 hr. 00 min.
**************************
*  HIGH WATER CP OCCUPANCY *
**************************
SCAN RATE: 1/MINUTE
                          TIME
DATE            0-1   1-2   2-3   3-4   4-5   5-6   ...
1989/05/20      15    23    13    16    16    21
1989/05/19      23    17    15    19    17    23
1989/05/18      19    24    23    12    24    19
1989/05/17      26    14    11    17    11    23
1989/05/16      29    28    14    13    31    31
1989/05/15      19    20    12    11    20    16
-------------------------------------------------------
```

Meaning: You successfully executed the command.

Action: None

---

highlogs

Function

Use the highlogs command to display the high runner logs.

highlogs command	
Command	**Parameters and variables**
highlogs	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the highlogs command.

Example of the highlogs command	
Example	**Task, response, and explanation**
highlogs ↵	

Task:　　　　Display the high runner log count.

Response:
```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/06/01 22:30 TO 1989/06/29 03:30
DURATION: 28 days 05 hr. 00 min.
*************************
*  HIGH RUNNER LOG COUNT *
*************************
Last reset : 1989/04/29 00:00
Data dumped: 1989/04/30 03:36

1)  AUDT..109: 14011        11) CMC...105:   3721
2)  PM....181: 12397        12) CMC...100:   3696
3)  IOAU..106: 11965        13) AUDT..183:   1611
4)  LOST..103: 10372        14) IOD...105:    972
5)  CMC...107:  9350        15) IOD...115:    941
6)  AUDT..128:  9299        16) PM....106:    624
7)  PM....102:  8114        17) LINE..138:    516
8)  IOD...102:  6964        18) AUDT..207:    554
9)  MTR...107:  6846        19) IOD...305:    349
10) DDU...205:  6731        20) CC....115:    299
```

Explanation:　This command displays the high runner log count.

highlogs (end)

Response

The following table provides an explanation of the response to the highlogs command.

Response for the highlogs command

MAP output   **Meaning and action**

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/06/01 22:30 TO 1989/06/29 03:30
DURATION: 28 days 05 hr. 00 min.
*************************
*  HIGH RUNNER LOG COUNT *
*************************
Last reset : 1989/04/29 00:00
Data dumped: 1989/04/30 03:36

1)   AUDT..109: 14011          11)  CMC...105:  3721
2)   PM....181: 12397          12)  CMC...100:  3696
3)   IOAU..106: 11965          13)  AUDT..183:  1611
4)   LOST..103: 10372          14)  IOD...105:   972
5)   CMC...107:  9350          15)  IOD...115:   941
6)   AUDT..128:  9299          16)  PM....106:   624
7)   PM....102:  8114          17)  LINE..138:   516
8)   IOD...102:  6964          18)  AUDT..207:   554
9)   MTR...107:  6846          19)  IOD...305:   349
10) DDU...205:  6731           20)  CC....115:   299
```

Meaning: You executed the command successfully.

Action: None

---

highparms

Function

Use the highparms command to display the high water mark values for critical office parameters.

highparms command	
Command	**Parameters and variables**
highparms	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the highparms command.

Example of the highparms command	
Example	**Task, response, and explanation**
highparms ↵	

	Task:	Display the high water mark values for critical office parameters.

Response:
```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
DURATION: 0 days 05 hr. 00 min.
********************************************************
* TABLES OF DAILY USAGE FOR CRITICAL OFFICE PARAMETERS *
********************************************************
30 days of high water mark values are printed with the most
current one (yesterday) being printed first
DATE          NUMCPLETTERS    NCCBS    NUMCALLPROCES
NUMOUTBUF
1989/04/29         0              0         60             3
                   1              1         60             1
                   0              0         60             2
DATE          FTRQAGENTS    FTRQ0AREAS   FTRQ2WAREAS
1989/04/29         0              0         0
                   0              0         0
                   0              0         0
```

Explanation: This command displays the high water mark values for critical office parameters.

highparms (end)

Response

The following table provides an explanation of the response to the highparms command.

Response for the highparms command

MAP output   Meaning and action

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
DURATION: 0 days 05 hr. 00 min.
*******************************************************
* TABLES OF DAILY USAGE FOR CRITICAL OFFICE PARAMETERS *
*******************************************************
30 days of high water mark values are printed with the most
current one (yesterday) being printed first
DATE          NUMCPLETTERS      NCCBS    NUMCALLPROCES    NUMOUTBUF
1989/04/29         0              0           60             3
                   1              1           60             1
                   0              0           60             2
DATE          FTRQAGENTS     FTRQ0AREAS   FTRQ2WAREAS
1989/04/29         0              0            0
                   0              0            0
                   0              0            0
```

Meaning: You successfully executed the command.

Action: None

logbuffer

Function

Use the logbuffer command to display TRAP, SWERR, and MISMATCH log information.

logbuffer command	
Command	**Parameters and variables**
logbuffer	There are no parameters or variables.

Qualifications

None

logbuffer (continued)

Example

The following table provides an example of the logbuffer command.

Example of the logbuffer command	
Example	**Task, response, and explanation**
logbuffer ↵	
	Task: Display the TRAP, SWERR, and MISMATCH log information.

	Response:
	OFFICE ID: CINT
	OFFICE CLLI: CINT
	BCS RELEASE: 30
	POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
	DURATION: 0 days 05 hr. 00 min.
	**************************************
	* TRAPS, SWERRS, and MISM Buffers *
	**************************************
	Trap number 27, Data Store Parity, Data Store Timeout
	At 4AAE1E=IYFMCOOR.AA03:TOPS_10_+#0240
	PTA= 4AADE0=IYFMCOOR.AA03:TOPS_10_+#0202,
	PROCID= #2107 #4076: dead, Entry Module: IYFMCOOR SSTI: #1040
	Current count of this trap type: 8
	Traceback:
	    4AA772=IYFMCOOR.AA03:FM_COORDINATOR+#03B0
	    01D8F7=MODULES.CA10:INITIALIZEP+#0009
	    011C21=PROCS.EB11:LIVEANDD+#0007
	
	DSholdregister: FDFE0A (No Owner)
	FIR= #0280
	TRAP on active CPU.
	CPU number 1, CMC 0 OFFLINE, CMC 1 ONLINE

	Explanation: This command displays the TRAP, SWERR, and MISMATCH log information.

logbuffer (end)

Response

The following table provides an explanation of the response to the logbuffer command.

Response for the logbuffer command

MAP output   Meaning and action

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
DURATION: 0 days 05 hr. 00 min.
*************************************
*   TRAPS, SWERRS, and MISM Buffers  *
*************************************
Trap number 27, Data Store Parity, Data Store Timeout
At 4AAE1E=IYFMCOOR.AA03:TOPS_10_+#0240
PTA= 4AADE0=IYFMCOOR.AA03:TOPS_10_+#0202,
PROCID= #2107 #4076: dead, Entry Module: IYFMCOOR SSTI: #1040
Current count of this trap type: 8
Traceback:
   4AA772=IYFMCOOR.AA03:FM_COORDINATOR+#03B0
   01D8F7=MODULES.CA10:INITIALIZEP+#0009
   011C21=PROCS.EB11:LIVEANDD+#0007

DSholdregister: FDFE0A (No Owner)
FIR= #0280
TRAP on active CPU.
CPU number 1, CMC 0 OFFLINE, CMC 1 ONLINE
```

Meaning: You executed the command correctly.

Action: None

---

logcount

Function

Use the logcount command to display the log count pegs.

logcount command	
Command	**Parameters and variables**
logcount	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the logcount command.

Example of the logcount command	
Example	**Task, response, and explanation**
logcount ↵	

Task:	Display the log count pegs.	
Response:	OFFICE ID: CINT OFFICE CLLI: CINT BCS RELEASE: 30 POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30 DURATION: 0 days 05 hr. 00 min. ********************** * Log Report Counts * ********************** LAST RESET: 1989/04/29 00:00 DATA DUMPED: 1989/04/30 03:36 LOST.101: 4 LOST.103: 137 LOST.106: 1 CC...130: 1 CC...115: 4 CC...116: 1 CC...103: 1 CC...107: 1 CMC..102: 1 CMC..105: 126 CMC..107: 135 CMC..111: 1 IOD..102: 1 IOD..103: 1 IOD..105: 65 IOD..115: 65 IOD..302: 2 IOD..303: 2 IOD..305: 2 IOD..312: 12	
Explanation:	This command displays the log count pegs.	

logcount (end)

Response

The following table provides an explanation of the response to the logcount command.

Response for the logcount command

MAP output   Meaning and action

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
DURATION: 0 days 05 hr. 00 min.
***********************
*   Log Report Counts  *
***********************
LAST RESET:  1989/04/29 00:00
DATA DUMPED: 1989/04/30 03:36

LOST.101: 4    LOST.103: 137    LOST.106: 1    CC...130: 1
CC...115: 4    CC...116: 1      CC...103: 1    CC...107: 1
CMC..102: 1    CMC..105: 126    CMC..107: 135  CMC..111: 1
IOD..102: 1    IOD..103: 1      IOD..105: 65   IOD..115: 65
IOD..302: 2    IOD..303: 2      IOD..305: 2    IOD..312: 12
```

Meaning: You executed the command correctly.

Action: None

---

memory

Function

Use the memory command to display memory usage information.

memory command	
Command	**Parameters and variables**
memory	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the memory command.

Example of the memory command	
Example	**Task, response, and explanation**
memory ↵	

Task:	Display memory usage information.	
| **Response:** | ```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
DURATION: 0 days 05 hr. 00 min.
***************
*   Memory    *
***************
                   ------------------
                    Number of Cards
                   ------------------
PS SHELF:         64K    256K   1M
--------          ---    ----   ---
    0              0      16     0
    1              0      14     0
                   ------------------
Total PS
  CARDS:           0      30     0
``` | |
| **Explanation:** | This command displays memory usage information. | |

memory (end)

Response

The following table provides an explanation of the response to the memory command.

Response for the memory command

MAP output   Meaning and action

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
DURATION: 0 days 05 hr. 00 min.
***************
*   Memory    *
***************
                  ------------------
                   Number of Cards
                  ------------------
PS SHELF:        64K    256K   1M
--------         ---    ----   ---
     0            0      16     0
     1            0      14     0
                  ------------------
Total PS
  CARDS:          0      30     0
```

Meaning: You successfully executed the command.

Action: None

---

newpatch

Function

Use the newpatch command to display recently applied patches.

newpatch command	
Command	**Parameters and variables**
newpatch	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the newpatch command.

Example of the newpatch command	
Example	**Task, response, and explanation**
newpatch ↵	
	Task: Display recently applied patches.

	Response:
	OFFICE ID: CINT
	OFFICE CLLI: CINT
	BCS RELEASE: 30
	POLLED FROM 1989/06/01 22:30 TO 1989/06/29 03:30
	DURATION: 28 days 05 hr. 00 min.
	******************
	* NEW PATCHES *
	******************
	Date last reset : 1989/06/01
	
	PATCHID INFO A TP MODULES TYPE DATE TIME R ST
	------- ---- - -- ------- ---- ---- ---- - --
	GAA12A24 Y CC IOUI ZC03 SRC 89/06/08 15:14:50 UN A
	RON53A24 Y CC IOUI ZC02 SRC 89/06/10 18:05:36 UN A
	LEM12A24 N CC IOUI ZC01 SRC 89/06/11 12:39:28 UN A
	    .
	    .
	    .

	Explanation: This command displays recently applied patches.

newpatch (end)

Response

The following table provides an explanation of the response to the newpatch command.

Response for the newpatch command

MAP output   Meaning and action

```
    OFFICE ID: CINT
    OFFICE CLLI: CINT
    BCS RELEASE: 30
    POLLED FROM 1989/06/01 22:30 TO 1989/06/29 03:30
    DURATION: 28 days 05 hr. 00 min.
    ******************
    *  NEW PATCHES   *
    ******************
    Date last reset : 1989/06/01

    PATCH ID   INFO A TP   MODULES   TYPE   DATE      TIME        R   ST
    --------   ---- - --   -------   ----   ----      ----        -   --
    GAA12A24        Y CC   IOUI ZC03   SRC  89/06/08  15:14:50    UN   A
    RON53A24        Y CC   IOUI ZC02   SRC  89/06/10  18:05:36    UN   A
    LEM12A24        N CC   IOUI ZC01   SRC  89/06/11  12:39:28    UN   A
      .
      .
      .
```

Meaning: You successfully executed the command.

Action: None

---

oms

Function

Use the oms command to display only non-zero OM peg counts.

oms command	
Command	**Parameters and variables**
oms	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the oms command.

Example of the oms command	
Example	**Task, response, and explanation**
oms ↵	

Task: Display non-zero peg counts.

Response:
```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
DURATION: 0 days 05 hr. 00 min.
***********************
*      Major OMs       *
***********************
AVN$ORIG = 5    AVN$TERM = 1    AVN$NOUT = 1   CMC$ERRR = 12
CMC$LERR = 4    CP1$LPOV = 123  CP1$ODEN = 12  CP1$CCBO = 5
CP1$WAKO = 44   CPU$MTCH = 51   CPU$SYNC = 12  CPU$WARM = 53
CPU$COLD = 131  EXT$OVFL = 51   NMC$MERR = 12  NMC$SERR = 24
NMC$SFLT = 86
```

Explanation: This command displays non-zero peg counts.

oms (end)

Response

The following table provides an explanation of the response to the oms command.

Response for the oms command
MAP output   **Meaning and action**

```
    OFFICE ID: CINT
    OFFICE CLLI: CINT
    BCS RELEASE: 30
    POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
    DURATION: 0 days 05 hr. 00 min.
    ***********************
    *       Major OMs        *
    ***********************
    AVN$ORIG = 5     AVN$TERM = 1     AVN$NOUT = 1    CMC$ERRR = 12
    CMC$LERR = 4     CP1$LPOV = 123   CP1$ODEN = 12   CP1$CCBO = 5
    CP1$WAKO = 44    CPU$MTCH = 51    CPU$SYNC = 12   CPU$WARM = 53
    CPU$COLD = 131   EXT$OVFL = 51    NMC$MERR = 12   NMC$SERR = 24
    NMC$SFLT = 86
```

Meaning: You executed the command correctly.

Action: None

opr

Function

Use the opr command to display an office performance report which includes:

- equipment counts
- total memory available
- restart history
- new patches added
- overall office performance using SPMS
- critical SPMS indexes (below 80 percent)
- unacceptable SPMS indexes (between 80-90 percent)
- high runner logs

opr command	
Command	**Parameters and variables**
opr	There are no parameters or variables.

Qualifications

None

opr (continued)

Example

The following table provides an example of the opr command.

Example of the opr command

Example	Task, response, and explanation

opr ↵

    Task: Display an office performance report.

    Response:

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/10/10 00:00 TO 1989/10/20 12:17
DURATION: 10 days 12 hrs. 17 min.
EQUIPMENT COUNT
---------------
Number of DGT_IBN lines:
Number of DGT_POTS lines:
Number of DP_IBN lines:
Number of DP_POTS lines:
Number of networks:
Number of nodes:
Number of LGC PMs:       Insv:      Comm:
Number of DTC PMs:       Insv:      Comm:
Number of RLCM PMs:       Insv:       Comm:
Number of TOPS positions:
Number of trunk groups:
Number of trunks:
Number of attendant consoles: Memory
--------
Data Store    : Total= 12030K    Available= 1083K
Program Store : Total=  7678K    Available= 2876K

New Patches
-----------
PATCHID INFO A TP   MODULES   TYPE   DATE      TIME       R  ST
------- ---- - --   -------   ----   ----      ----       -  --
HAM98A24     Y CC   IOUI ZC03 SRC    89/06/08  15:14:50   UN  A
RON53A24     Y CC   IOUI ZC02 SRC    89/06/10  18:05:36   UN  A
```

    Explanation: This command displays an office performance report.

---

opr (end)

Response

The following table provides an explanation of the response to the opr command.

Response for the opr command

MAP output   Meaning and action

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/10/10 00:00 TO 1989/10/20 12:17
DURATION: 10 days 12 hrs. 17 min.
EQUIPMENT COUNT
---------------
Number of DGT_IBN lines:
Number of DGT_POTS lines:
Number of DP_IBN lines:
Number of DP_POTS lines:
Number of networks:
Number of nodes:
Number of LGC PMs:       Insv:      Comm:
Number of DTC PMs:       Insv:      Comm:
Number of RLCM PMs:       Insv:       Comm:
Number of TOPS positions:
Number of trunk groups:
Number of trunks:
Number of attendant consoles: Memory
--------
Data Store    : Total= 12030K    Available= 1083K
Program Store : Total=  7678K    Available= 2876K

New Patches
-----------
PATCH ID    INFO A TP    MODULES    TYPE    DATE        TIME        R   ST
--------    ---- - --    -------    ----    ----        ----        -   --
HAM98A24        Y CC    IOUI ZC03    SRC    89/06/08    15:14:50    UN   A
RON53A24        Y CC    IOUI ZC02    SRC    89/06/10    18:05:36    UN   A
```

Meaning: You successfully executed the command.

Action: None

pmconfig

Function

Use the pmconfig command to display the peripheral module (PM) configuration.

pmconfig command	
Command	**Parameters and variables**
pmconfig	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the pmconfig command.

Example of the pmconfig command	
Example	**Task, response, and explanation**
pmconfig ↵	
	Task: Display the PM configuration.
	Response:

```
***********************************
*      Office Configuration       *
***********************************
The configuration of (I)LTC/(I)LGC, RCC and (I)LCMs,
DLMs and IPEs ONLY is displayed
  --------------           -------------
 |LTC HOST  0 | Links 5,3  |RCC REM2  0|  Links 14,16
  --------------           -------------  LCM REM2 00 0
means: LTC P-side link 5 is connected to RCC C-side link 0
       LTC P-side link 3 is connected to RCC C-side link 1
       RCC P-side link 14 is connected to LCM C-side link 0
```

-continued-

pmconfig (continued)

Example of the pmconfig command (continued)

Example   **Task, response, and explanation**

Response:

```
HOST XPM    LINKS        PSIDE PM(S)   LINKS       PSIDE PM(S)
_____    _____        _____   _____       _____

_____
|LTC HOST 0|
|          |
|          |  3,6,5,7,  |‾‾‾‾‾‾‾‾‾‾|
|          |  8,9,10,   |RCC REM2 0|
|          |  11        |          |10,6,5      LCM REM2 2 0
|          |            |          |2,4,3       LCM REM2 2 1
|          |            |          |14,16       LCM REM2 0 0
|          |            |          |18,17       LCM REM2 0 1
|          |            |          |
|          |  4,12,13   |_____|
|          |  0,2,1       LCM HOST 1 0
|_____|  19,18,17    LCM HOST 0 0
              _____    LCM HOST 0 1
_____
|LTC HOST 1|
|          |
|          |  3,6,5,7,  |‾‾‾‾‾‾‾‾‾‾|
|          |  8,9,10    |RCC REM2 0|
|          |            |          |4,5         LCM REM3 3 0
|          |            |          |9,8,        LCM REM3 3 1
|          |            |          |2,6,10      LCM REM4 0 0
|          |            |          |12,13       LCM REM3 1 1
|          |            |          |
|          |  17,18,19  |_____|
|          |  12,13       LCM HOST 2 0
|_____|  14,16,15    LCM HOST 2 1
              _____    LCM HOST 1 0
_____
|LTC HOST 2|
|          |  1,2       LCM HOST 3 0
|          |  3,0       LCM HOST 3 1
|          |  14,16,15  LCM HOST 4 0
|          |  18,17     LCM HOST 4 1
|          |  4,5,7,9,  LCM REM1 0 0
|          |  12,13     LCM REM1 2 0
|_____|  6,8,10
_____
|LTC HOST 4|
|          |  12,14,16, LCMI HOST 5 0
|_____|  18
```

Explanation: This command displays the PM configuration.

End

---

pmconfig (end)

Response

The following table provides an explanation of the response to the pmconfig command.

Response for the pmconfig command

MAP output   Meaning and action

```
*************************************
*      Office Configuration         *
*************************************
The configuration of (I)LTC/(I)LGC, RCC and (I)LCMs,
DLMs and IPEs ONLY is displayed
--------------          -------------
|LTC HOST  0 | Links 5,3  |RCC REM2  0|  Links 14,16
--------------          ------------- LCM REM2 00 0
means: LTC P-side link 5 is connected to RCC C-side link 0
       LTC P-side link 3 is connected to RCC C-side link 1
       RCC P-side link 14 is connected to LCM C-side link 0
.
.
.
_____
|LTC HOST 2|
|          |1,2       LCM HOST 3 0
|          |3,0       LCM HOST 3 1
|          |14,16,15  LCM HOST 4 0
|          |18,17     LCM HOST 4 1
|          |4,5,7,9,  LCM REM1 0 0
|          |12,13     LCM REM1 2 0
_____ 6,8,10

_____
|LTC HOST 4|
|          |12,14,16, LCMI HOST 5 0
_____ 18
```

Meaning: You successfully executed the command.

Action: None

---

pmloads

Function

Use the pmloads command to display all peripheral module (PM) loads available in an office.

pmloads command	
Command	**Parameters and variables**
pmloads	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the pmloads command.

Example of the pmloads command	
Example	**Task, response, and explanation**
pmloads ↵	
	Task: Display all PM loads.

	Response:
	`OFFICE ID: CINT`
	`OFFICE CLLI: CINT`
	`BCS RELEASE: 30`
	`POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30`
	`DURATION: 0 days 05 hr. 00 min.`
	`**************`
	`*  PM Loads  *`
	`**************`
	`PM LOADS IN THIS OFFICE:`
	`TKTMKA02`
	`MTMKA02`
	`RMMDA01`
	`NID26BG`
	`IDT29AR`
	`IDT29AQ`
	`NIL25BL`
	`RCL29AR4`
	`LCM28C`
	`IR28AB`
	`MC729AQ`
	`IES29AQ`
	`END OF PM LOADS`

	Explanation: This command displays all PM loads available in an office.

pmloads (end)

Response

The following table provides an explanation of the response to the pmloads command.

Response for the pmloads command

MAP output   Meaning and action

```
OFFICE ID: CINT
OFFICE CLLI: CINT
BCS RELEASE: 30
POLLED FROM 1989/05/29 22:30 TO 1989/05/30 03:30
DURATION: 0 days 05 hr. 00 min.
***************
*  PM Loads   *
***************
PM LOADS IN THIS OFFICE:
TKTMKA02
MTMKA02
RMMDA01
NID26BG
IDT29AR
IDT29AQ
NIL25BL
RCL29AR4
LCM28C
IR28AB
MC729AQ
IES29AQ
END OF PM LOADS
```

Meaning: You successfully displayed all PM loads available in your office.

Action: None

---

quit

Function

Use the quit command to exit the BCSMON directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⌐ *1 level* ⌐ all *name* ⌐ *n_levels* ⌐

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ⏎		
	Task:	Exit from this directory.
	Response:	CI:
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
CI:	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
QUIT -- Increment not found	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
QUIT -- Unable to quit requested number of levels	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

reset

Function

Use the reset command to reset the BCS monitoring data.

reset command	
Command	**Parameters and variables**
reset	all logcount newpatch oms restartinfo

Parameters and variables	Description
all	This parameter resets all of the BCS monitoring data.
logcount	This parameter resets the logcount data only.
newpatch	This parameter resets the new patch data only.
oms	This parameter resets the oms data only.
restartinfo	This parameter resets the restart information data only.

Qualifications

None

reset (end)

Example

The following table provides an example of the reset command.

Example of the reset command	
Example	**Task, response, and explanation**
reset **newpatch** ↵	
	Task: Reset the new patch data.
	Response: W A R N I N G : This command will destroy non regeneratable long-term switch performance data. DO YOU REALLY WISH TO RESET? Please confirm ("YES", "Y", "NO", or "N"): >y The new patch date has been set to: 1989/06/02
	Explanation: This command resets the new patch data.

Response

The following table provides an explanation of the response to the reset command.

Response for the reset command	
MAP output	**Meaning and action**
The new patch date has been set to: 1989/06/02	
	Meaning: You successfully reset the new patch data.
	Action: None

restartinfo

Function

Use the restartinfo command to display warm and cold restart information.

restartinfo command	
Command	**Parameters and variables**
restartinfo	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the restartinfo command.

Example of the restartinfo command	
Example	**Task, response, and explanation**
restartinfo ↵	
	Task: Display warm and cold restart information.

	Response:
	`***********************`
	`*  Restart Information *`
	`***********************`
	`LAST RESET:  1989/04/29 22:03`
	`DATA DUMPED: 1989/04/30 03:36`
	
	`MAN$COLD    0    SYS$COLD   0    MAN$WARM    0    SYS$WARM    0`
	`SYSTEM DOWNTIME: 0 hr., 0 min., 0 sec.`

	Explanation: This command displays warm and cold restart information.

restartinfo (end)

Response

The following table provides an explanation of the response to the restartinfo command.

Response for the restartinfo command

MAP output   **Meaning and action**

```
***********************
*  Restart Information *
***********************
LAST RESET:  1989/04/29 22:03
DATA DUMPED: 1989/04/30 03:36

MAN$COLD     0    SYS$COLD   0    MAN$WARM    0     SYS$WARM    0
SYSTEM DOWNTIME: 0 hr., 0 min., 0 sec.
```

Meaning: You executed the command correctly.

Action: None

BCSUPDATE level commands

Use the BCSUPDATE level of the MAP to access batch change supplement (BCS) process driver commands.

Accessing the BCSUPDATE level

To access the BCSUPDATE level, enter the following command from the CI level:

bcsupdate ↵

BCSUPDATE commands

The commands available at the BCSUPDATE MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

BCSUPDATE commands	
Command	**Page**
datadump	B-55
device	B-59
help	B-61
logcheck	B-63
override	B-65
pmaudit	B-67
postswact	B-69
precheck	B-71
preswact	B-75
quit	B-79
reset	B-83
runstep	B-85
-continued-	

BCSUPDATE commands (continued)	
Command	**Page**
status	B-87
swactci	B-91
End	

datadump

Function

Use the datadump command to display office information.

datadump command parameters and variables	
Command	**Parameters and variables**
datadump	There are no parameters or variables.

Qualifications

None

datadump (continued)

Example

The following table provides an example of the datadump command.

Example of the datadump command	
Example	**Task, response, and explanation**
datadump ↵	

	Task:	Display office information.
| | **Response:** | ```
Do you want to display INB, MB and RES trunks?
Please confirm ("YES", "Y", "NO", or "N"):
>y
Do you want to display 1MR registers?
Please confirm ("YES", "Y", "NO", or "N"):
>y
Do you want to display INB lines?
Please confirm ("YES", "Y", "NO", or "N"):
>y
Do you want to display tables?
Please confirm ("YES", "Y", "NO", or "N"):
>y
Do you want to display hardware?
Please confirm ("YES", "Y", "NO", or "N"):
>y

Data Dump for CO: GRBLDMSTOPS  BCS35  Date: DEC
14, 1992  09:29:28
==================================================
                    STORAGE USAGE
                   _____
      .
      .
      .
VFG NAME BILLING NUMBER INWATS ATMPT INWATS OVERF
-------- -------------- ------------ ------------
BNRIBN    6197202701         0            0
BNRPOT    6197202701         0            0
NTIIBN    6197203701         0            0
EWAT1     6197202705         0            0
EWAT4     6197202705         0            0
EWAT8     6197202705         0            0
==================================================
                  DATA DUMP COMPLETE
                  ------------------
``` |
| | **Explanation:** | This command displays office information. |

datadump (end)

Response

The following table provides an explanation of the response to the datadump command.

Response for the datadump command	
MAP output	**Meaning and action**
SYNTAX: DATA_DUMP does not require any parameters.	
	Meaning: You entered the command with parameters.
	Action: Reenter the command without parameters.

device

Function

Use the device command to display device and user information.

device command parameters and variables	
Command	**Parameters and variables**
device	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the device command.

Example of the device command	
Example	**Task, response, and explanation**
device ↵	
	Task: Display device and user information.
	Response:
| | ```
    Device: MODEM
       IOC: 0
      Card: 4
      Port: 0
  Comclass: ALL
      User: ADMIN
  Priority: 4
     Stack: 10000
 Privclass: ALL
``` |
| | **Explanation:** This command displays device and user information. |

device (end)

Response

The following table provides an explanation of the response to the device command.

Response for the device command	
MAP output	**Meaning and action**
SYNTAX: DEVICE does not require any parameters.	
	Meaning: You entered the command with parameters.
	Action: Reenter the command without parameters.

help

Function

Use the help command to receive online documentation for the
BCSUPDATE directory.

help command parameters and variables	
Command	**Parameters and variables**
help	_all_ _command_nam_

Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
_command_nam_	This variable specifies a valid BCSUPDATE directory command name. When the _command_nam_ variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help **pmaudit** ↵ _where_	
pmaudit	specifies the command name

	Task:	Access online documentation.
	Response:	PMAUDIT: Creates a file containing peripheral load names and patches. If a file name is not specified, PMAUDIT$FILE will be used. Parms: [<file name> STRING]
	Explanation:	This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

logcheck

Function

Use the logcheck command to display traps and specified logs.

logcheck command parameters and variables	
Command	**Parameters and variables**
logcheck	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the logcheck command.

Example of the logcheck command	
Example	**Task, response, and explanation**
logcheck ↵	

	Task:	Display traps and specified logs.

	Response:	---------------------------------CM 119 Logs
		No logs of this type found.
		
		---------------------------------CM 120 Logs
		RTPC *** CM120 DEC14 07:59:00 0200 INIT
		      RELOAD Restart no. 1 at ???-00 00:00:00.
		       System Image Reload.
		
		1 log(s) displayed.
		
		---------------------------------CM 122 Logs
		No logs of this type found.

	Explanation:	This command displays traps and specified logs.

logcheck (end)

Response

The following table provides an explanation of the response to the logcheck command.

Response for the logcheck command	
MAP output	**Meaning and action**
SYNTAX: LOGCHECK does not require any parameters.	
	Meaning: You entered the command with parameters.
	Action: Reenter the command without parameters.

override

Function

Use the override command to override individual steps of the preswact command. The command disallows an override for a step which is executing.

override command parameters and variables	
Command	**Parameters and variables**
override	*procedure_name*

Parameters and variables	**Description**
procedure _name	This variable is the name of a valid step executed by the preswact command.

Qualifications

None

Example

The following table provides an example of the override command.

Example of the override command		
Example	**Task, response, and explanation**	
override apply_patches ↵ *where* apply_patches	is the name of a valid step executed by the preswact command	
	Task:	This command is used to override a specified preswact command.
	Response:	`APPLY_PATCHES has been set to complete`
	Explanation:	You have overridden a specified preswact command.

override (end)

Response

The following table provides an explanation of the response to the override command.

Response for the override command	
MAP output	**Meaning and action**
This step is currently executing, it may not be overridden.	
	Meaning: This message is displayed if the frame relay billing generation step is currently executing when an override of this step is requested. If the status is failed, the override is allowed.
	Action: None

pmaudit

Function

Use the pmaudit command to create a file containing peripheral module (PM) load names and patches.

pmaudit command parameters and variables	
Command	**Parameters and variables**
pmaudit	*pmaudit$file* *filename*

Parameters and variables	**Description**
pmaudit$file	Omitting this entry forces the system to default to creating the file named pmaudit$file to hold the PM load names and patches.
filename	This variable specifies the file name that contains the PM load names and patches.

Qualifications

None

Examples

The following table provides examples of the pmaudit command.

Examples of the pmaudit command		
Example	**Task, response, and explanation**	
pmaudit ↵		
	Task:	Create the file using the default.
	Response:	File PMAUDIT$FILE created in SFDEV.
	Explanation:	This command creates the file using the pmaudit$file name.
	-continued-	

pmaudit (end)

Examples of the pmaudit command (continued)		
Example	**Task, response, and explanation**	
pmaudit audi ↵ *where* audi specifies the file name		
	Task:	Create the file using a file name.
	Response:	File AUDI created in SFDEV.
	Explanation:	This command creates the file using the file name audi.
End		

Response

The following table provides an explanation of the response to the pmaudit command.

Response for the pmaudit command	
MAP output	**Meaning and action**
File PMAUDIT$FILE created in SFDEV.	
	Meaning: You entered the command correctly.
	Action: None

postswact

Function

Use the postswact command to display postswact and recovery functions.

postswact command parameters and variables	
Command	**Parameters and variables**
postswact	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the postswact command.

Example of the postswact command	
Example	**Task, response, and explanation**
postswact ↵	
	Task: Display postswact and recovery functions.

	Response: `DIRP_RECOVERY                   executing`
	`***Could not find DIRP recovery file "DIRP_REC".`
	`   This file was created by the PRESWACT process.`
	`   data-fill table DIRPPOOL manually or call`
	`support.`
	`DIRP_RECOVERY                    not complete`
	
	`ACT - Error: Dirp subsystems not recovered.`
	`Correct error condition.`

	Explanation: This command displays postswact and recovery functions.

postswact (end)

Response

The following table provides an explanation of the response to the postswact command.

Response for the postswact command
MAP output   **Meaning and action**

```
DIRP_RECOVERY                    executing
***Could not find DIRP recovery file "DIRP_REC".
   This file was created by the PRESWACT process.
   data-fill table DIRPPOOL manually or call support.
DIRP_RECOVERY                    not complete

ACT – Error: Dirp subsystems not recovered.
Correct error condition.
```

Meaning: You entered the command correctly, but the system revealed an error.

Action: Datafill the Table DIRPPOOL or call support.

precheck

Function

Use the precheck command to perform a batch change supplement (BCS) precheck process.

precheck command parameters and variables	
Command	**Parameters and variables**
precheck	final first
Parameters and variables	**Description**
final	This parameter performs a final BCS precheck.
first	This parameter performs a beginning BCS precheck.

Qualifications

None

Example

The following table provides an example of the precheck command.

precheck (continued)

Example of the precheck command

Example **Task, response, and explanation**

---

precheck first ↵

       Task: Perform a beginning BCS precheck.

       Response: `DISPLAY_DEVICE_AND_USER`   `executing`

```
      Device: MODEM
         IOC: 0
        Card: 4
        Port: 0
    Comclass: ALL
        User: ADMIN
    Priority: 4
       Stack: 10000
    Privclass: ALL

    DISPLAY_DEVICE_AND_USER          complete

    TABLE_COUNTS                     executing
        .
        .
        .
    DEVICE_CHECK                     executing

    Displaying bad nodes:
    Node Device
    ---- ------
    39   SMA 1
    50   IDT 3
    DEVICE_CHECK                     not complete

    Investigate and correct if necessary.
```

     Explanation: This command performs a beginning BCS precheck.

Responses

The following table provides explanations of the responses to the precheck command.

precheck (end)

Responses for the precheck command

MAP output   Meaning and action

```
DISPLAY_DEVICE_AND_USER          executing

   Device: MODEM
      IOC: 0
     Card: 4
     Port: 0
 Comclass: ALL
     User: ADMIN
 Priority: 4
    Stack: 10000
Privclass: ALL


DISPLAY_DEVICE_AND_USER          complete

TABLE_COUNTS                     executing
    .
    .
    .
DEVICE_CHECK                     executing

Displaying bad nodes:
Node Device
---- ------
39   SMA 1
50   IDT 3
DEVICE_CHECK                     not complete

Investigate and correct if necessary.
```

Meaning: You entered the command correctly, but an error was revealed.

Action: Investigate and correct if necessary.

```
Invalid symbol: <Which process to execute:> {FIRST,
                                             FINAL}
```

Meaning: You entered the command with an invalid parameter.

Action: Enter the parameter to continue, or abort to cancel.

preswact

Function

Use the preswact command to perform a series of steps required for preparation of a DMS office before execution of a central control (CC) warm switch of activity (SWACT).

preswact command parameters and variables	
Command	**Parameters and variables**
preswact	There are no parameters or variables

Qualifications

None

Example

The following table provides an example of the preswact command.

Example of the preswact command		
Example	**Task, response, and explanation**	
preswact ↵		
	Task:	Run the application drivers.

	Response:	VERIFY_STORE Executing
		VERIFY_STORE Complete
		CHECK_DSLIMIT Executing
		CHECK_DSLIMIT Complete
		         * *
		         * *

	Explanation:	This command runs the application drivers.

Responses

The following table provides explanations of the responses to the preswact command.

preswact (continued)

Responses for the preswact command	
MAP output	**Meaning and action**
``` FRAME_RELAY_BILLING_GENERATION Executing FRAME_RELAY_BILLING_GENERATION Continuing to execute or FRAME_RELAY_BILLING_GENERATION Executing FRAME_RELAY_BILLING_GENERATION Failed ```	
	Meaning: The frame relay billing system collects all billing data of the FRS internal database into billing records for Automatic Message Accounting (AMA). You see that the frame relay billing generation step has begun collecting frame relay billing data and is continuing that task as other steps within preswact are run. The failure response indicates there was some problem setting up the frame relay billing generation step.
	Action: Your only required action would be upon failure. As with failure in other preswact steps, the preswact command terminates and you have the opportunity to override the frame relay billing generation step or to reenter the preswact command and retry execution of the step.
``` FRAME_RELAY_BILLING_GENERATION continuing to execute xxx items have been completed, there are yyy total items. or FRAME_RELAY_BILLING_GENERATION FAILED ```	
	Meaning: You see this status message when the preswact command is reentered while the frame relay billing generation step is still executing. You see that the frame relay billing generation step is continuing to collect all frame relay billing data. You can tell the percentage of processing completed from the xxx items completed versus yyy total items. The numbers of items correspond to the number of frame relay endpoints needing billing aggregation (they do not correspond to the actual number of AMA records to generate).
	Action: You need not do anything if the frame relay billing generation step is continuing. If the step fails, the preswact command terminates and you may override this step or reenter the preswact command to execute the step again.
-continued-	

preswact (end)

Responses for the preswact command (continued)
MAP output   Meaning and action
| ```
The following steps are still executing:

FRAME RELAY BILLING GENERATION
xxx items have been completed, there are yyy total items.
Enter one of the following: (YES/NO)
``` |

Meaning:	You see that the frame relay billing generation step has not completed collecting all of the frame relay billing data at the end of executing all of the other steps of preswact. You can tell the progress by the items completed versus items total. You are prompted to enter yes for immediate exiting of the preswact command or no to delay for approximately 15 minutes. The executing steps are not stopped or overridden. You can monitor the status by entering the preswact command again or by using the status command.
Action:	When you enter yes, the preswact command is exited immediately after giving a status display of all the steps.
	When you enter no, a delay of approximately 15 minutes is allowed in order that the frame relay billing generation step may execute to completion. At the end of the 15 minute period, if the step is still not finished, the same prompt sequence is given; otherwise the preswact command prints the status of all the steps and exits.

End

quit

Function

Use the quit command to exit the BCSUPDATE directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⎡ *1 level* all *name* *n_levels* ⎤

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
CI:	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
QUIT -- Increment not found	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
QUIT -- Unable to quit requested number of levels	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

reset

Function

Use the reset command to reset the status of all preswact steps back to needed. A step that is currently executing is not reset to needed status.

reset command parameters and variables	
Command	**Parameters and variables**
reset	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the reset command.

Example of the reset command	
Example	**Task, response, and explanation**
reset ↵	
	Task: Reset the status of all preswact steps.

	Response: `**Warning - All non-EXECUTING procedures will be set to "NEEDED"`
	
	 `Do you wish to continue?`
	
	 `Enter one of the following: (YES/NO)`

	Explanation: Enter yes to set all of the non-executing steps. Enter no to exit with no resets done.

reset (end)

Response

The following table provides an explanation of the response to the reset command.

Response for the reset command	
MAP output	**Meaning and action**

```**Warning - All non-EXECUTING procedures will be set to "NEEDED"```
 	
``` Do you wish to continue?```
 	
```Enter one of the following: (YES/NO)```	

	Meaning: You entered the command correctly. The preswact step frame relay billing generation is the only preswact step that could be executing. **Action:** Enter yes to set all of the non-executing steps to needed status. Enter no to exit with no resets done. To verify the current status of all preswact steps, use the status command.

runstep

Function

Use the runstep command to run an individual step of the preswact command.

runstep command parameters and variables	
Command	**Parameters and variables**
runstep	*procedure*
Parameters and variables	**Description**
procedure	This variable specifies the procedure name of a valid step executed by the preswact command.

Qualifications

None

Example

The following table provides an example of the runstep command.

runstep (end)

Example of the runstep command

Example **Task, response, and explanation**

---

runstep device_check ↵
where

device_check specifies the procedure name

	Task:	Run a step of the preswact command.

Response:
```
This step exists within PRECHECK FIRST and
PRECHECK FINAL.
Which process should the step be executed from?
Next par is: <Which process:> {FIRST,
                                      FINAL}
Enter:  <Which process:>
>first
**Warning - Procedure will be executed out of
sequence.
  Do you wish to continue?
Please confirm ("YES", "Y", "NO", or "N"):
>y
DEVICE_CHECK                    executing
Displaying bad nodes:
Node Device
---- ------
39   SMA 1
50   IDT 3
DEVICE_CHECK                    not complete
```

Explanation: You ran the device_check step of the preswact command.

Response

The following table provides an explanation of the response to the runstep command.

Response for the runstep command

MAP output **Meaning and action**

```
This step is currently executing.
```

Meaning: The step is currently running or executing. The command aborts.

Action: None

---

status

Function

Use the status command to provide a status display of all the steps of the preswact or postswact command.

status command parameters and variables	
Command	**Parameters and variables**
status	postswact preswact
Parameters and variables	**Description**
postswact	This parameter specifies the postswact command.
preswact	This parameter specifies the preswact command.

Qualifications

None

Examples

The following table provides examples of the status command.

Examples of the status command	
Example	**Task, response, and explanation**
status **postswact** ↵	
	Task: Check the status of the postswact command.
	Response:

	DIRP_RECOVERY Complete 00.00.000
	6X45_PECS Complete 00.00.000
	        * *
	        * *
	        * *

	Explanation: This command checks the status of the postswact command.
-continued-	

status (continued)

Examples of the status command (continued)	
Example	**Task, response, and explanation**
status preswact ↵	
	Task: Check the status of the preswact command.
	Response:
| | ```
VERIFY_STORE        Complete    00.00.000
CHECK_DSLIMIT       Complete    00.00.000
APPLY_PATCHES       Needed      Time unavailable
        *               *
        *               *
        *               *
``` |
| | **Explanation:** This command checks the status of the preswact command. |
| **End** | |

status (end)

Response

The following table provides an explanation of the response to the status command.

Response for the status command	
MAP output	**Meaning and action**
FRAME_RELAY_BILLING_GENERATION Executing xxx items have been completed, there are yyy total items. or FRAME_RELAY_BILLING_GENERATION Complete or FRAME_RELAY_BILLING_GENERATION Failed or FRAME_RELAY_BILLING_GENERATION Needed	
	Meaning: The status of the step may be executing, complete, failed, or needed. In the case of the executing status, an additional display is provided so that you have an idea of the progress of the executing frame relay billing aggregation for preswact. The items in the display refer to the number of frame relay endpoints for which billing is aggregated; not the exact number of billing records generated for Automatic Message Accounting (AMA) formatting.
	Action: None

swactci

Function

Use the swactci command to enter the switch of activity (SwAct) CI environment.

swactci command parameters and variables	
Command	**Parameters and variables**
swactci	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the swactci command.

Example of the swactci command	
Example	**Task, response, and explanation**
swactci ↵	
	Task: Enter the SWACT CI environment.
	Response: `SWACTCI:`
	Explanation: This command enters the SWACT CI environment.

Responses

The following table provides explanations of the responses to the swactci command.

Responses for the swactci command	
MAP output	**Meaning and action**
`MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.`	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None
-continued-	

swactci (end)

Responses for the swactci command (continued)	
MAP output	**Meaning and action**
SWACTCI:	
	Meaning: You entered the SWACT CI environment. See the SWACTCI directory for available commands.
	Action: None
End	

C7MON level commands

Use the C7MON level of the MAP to trace common channel signaling 7 (CCS7) messages passing through an message switch buffer 7 (MSB7) or link interface unit 7 (LIU7). When you enter search criteria, a template is created and stored in a match table. The system searches the message table to locate messages that match the template. If a match is found, a message dump is directed to either the MAP, logs, or to a specified disk file.

	CAUTION
	You cannot enter message tracing criteria during another message tracing session.
	Entering C7MON directory message tracing criteria is prohibited when any other message tracing is in progress. The reverse also is true. This restriction applies to C7TULINK, C7MON, and SIGRTU directories.

Accessing the C7MON level

To access the C7MON level, enter the following command from the CI level:

> **c7mon** ↵

C7MON commands

The commands available at the C7MON MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

C7MON commands	
Command	**Page**
delete	C-3
display	C-5
monitor	C-13
-continued-	

C7MON commands (continued)	
Command	**Page**
q	C-21
qc7mon	C-23
quit	C-25
start	C-29
stop	C-33
End	

delete

Function

Use the delete command to delete one or all the templates defined by the user.

delete command parameters and variables	
Command	**Parameters and variables**
delete	all *template*

Parameters and variables	**Description**
all	This parameter deletes all templates owned by the user.
template	This variable specifies a valid routeset name to delete. Routeset names are datafilled in the C7RTESET table. The valid entry range is 0-7.

Qualifications

None

Examples

The following table provides examples of the delete command.

Examples of the delete command		
Example	**Task, response, and explanation**	
delete all ↵		
	Task:	Delete all templates defined by the user.
	Response:	Delete done
	Explanation:	The system deletes all templates defined by the user.
-continued-		

delete (end)

Examples of the delete command (continued)	
Example	**Task, response, and explanation**
delete 4 ↵ *where* 4 specifies a valid routeset name	
	Task: Delete a specific template. **Response:** Delete done **Explanation:** The system deleted the template identified as number 4.
End	

Responses

The following table provides explanations of the responses to the delete command.

Responses for the delete command	
MAP output	**Meaning and action**
ERROR: Template is not owned by you.	
	Meaning: There is a template with the specified index, but it is not owned by the terminal. The command halts execution. **Action:** None
ERROR: Template not defined.	
	Meaning: There is no template defined with the specified index. **Action:** Retry the delete command with a valid index.
Trace must be stopped to delete.	
	Meaning: The trace must be stopped to delete a template. The command halts execution. **Action:** If the template must be deleted, the trace must be stopped first.

display

Function

Use the display command to display the messages stored in a disk file.

display command parameters and variables	
Command	**Parameters and variables**
display	*filename* ⎡ short ⎤ ⎢ line ⎥ ⎣ long ⎦

Parameters and variables	**Description**
short	This default parameter produces the display in short format. Short format messages display in binary format and are decoded as bits in the CCITT recommendations. Either omit the format entry to force the default action or enter the display *filename* short command string.
filename	This variable specifies the name of the file in which messages are stored.
line	This parameter produces a display in line format. ***Note:*** The format of SCCP data depends on the message type; only Unitdata (UDT) and Unitdata Service (UDTS) SCCP message types are decoded. All other SCCP message types are not decoded.
long	This parameter produces a display in long format. Long parameters display in hexadecimal form with spaces between bytes.

Qualification

> **CAUTION**
> **The disk file must be closed.**
> The disk file must be closed before using the display command.

The disk file must be closed before using the display command.

Examples

The following table provides examples of the display command.

display (continued)

Examples of the display command
Example   **Task, response, and explanation**

display trace1 short ↵
where

trace1   specifies the file name

> **Task:**   Display a specified file in short format.
>
> **Response:**
> ```
> C7MN100 date time lognums INFO INCOMING_LINK_MSG
> C7 HEADER: LEN=nn MSG=#nn LINK=n SLS=n CLLI=linkset_name
> C7 SIO:    NETWORK ID=n PRIORITY=n SERV IND=nn
> C7 LABEL:  DPC-dpc        OPC=opc        SLS=nn
> N7DATA FOLLOWING ROUTING LABEL:#xx xx xx xx xx xx xx xx xx xx
>             SCCP MSG TYPE = sccp_msg_types
> ```
>
> **Explanation:**   This command displays the file named trace1 in short format. This example message file contains SCCP data.

display trace2 long ↵
where

trace2   specifies the file name

> **Task:**   Display a specified file in long format for SCCP UDT or UDTS message type traces.
>
> **Response:**
> ```
> C7MN100 date time lognums INFO INCOMING_LINK_MSG
> C7 HEADER: LEN=nn MSG=#nn LINK=n SLS=n CLLI=linkset_name
> C7 SIO:    NETWORK ID=n PRIORITY=n SERV IND=nn
> C7 LABEL:  DPC-dpc        OPC=opc        SLS=nn
> N7DATA FOLLOWING ROUTING LABEL:#xx xx xx xx xx xx xx xx xx xx
>   SCCP MSG TYPE = UDT
>    PROT_CLASS=prot_clas_type[0000]MSG_HDL_OPT=hdl_opt_type[0000]
>    CALLED PARTY ADDR
>     .........
>    CALLING PARTY ADDR
>     .........
>    SCCP DATA
>     #xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx
> ```
>
> **Explanation:**   This command displays the file named trace2 in long format. The example file represents the format for SCCP UDT or UDTS message type traces. (The N7 field indicates this is a CCITT message.)

<center>**-continued-**</center>

display (continued)

Examples of the display command (continued)

Example   **Task, response, and explanation**

---

display trace2a long ↵
where

trace2a   specifies the file name

Task: Display a specified file in long format for SCCP message types
                   other than UDT or UDTS.

Response:
```
C7MN100 date time lognums INFO INCOMING_LINK_MSG
C7 HEADER: LEN=nn MSG=#nn LINK=n SLS=n CLLI=linkset_name
C7 SIO:    NETWORK ID=n PRIORITY=n SERV IND=nn
C7 LABEL:  DPC-dpc        OPC=opc          SLS=nn
N7DATA FOLLOWING ROUTING LABEL:#xx xx
xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx
xx xx xx xx xx xx xx xx xx xx xx xx
 SCCP MSG TYPE = sccp_msg_types
  SCCP DATA
    #xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx
     xx xx xx xx xx xx xx xx xx xx xx xx xx xx xx
```

Explanation: This command displays the file named trace2a in long format. The
                   example file contains traces for SCCP message types other than
                   UDT or UDTS. (The N7 field indicates this is a CCITT message.)

-continued-

---

display (continued)

Examples of the display command (continued)

Example   **Task, response, and explanation**

---

display trace0001 line ↵
where

trace0001   specifies the file name

Task: Display a specified file in line format.

Response:
```
C7MN100 date time lognums INFO INCOMING_LINK_MSG
SCCP sccp_msg_type MSG TYPE DPC = dpc OPC = opc
  CALLED PARTY ADDR
   ADDR INDC = #xx
   PCI  = bool PC = point_code
   SSNI = bool SSN = ssn_type                              (#xx)
   GTI  = gt_type                                         [0001]
   RTG  = rtg-type                                           [0]
   NUI  = bool
  ADDR NATURE   = addr_nat_type                            (#xx)
  ODD/EVEN IND  = odd_even_digits_indicator_type            [0]
  DIGITS        = #xx xx xx xx xx xx xx
```

Explanation: This command displays the file named trace0001 in line format.
                This example message file contains SCCP data. The called and
                calling party address information display in one of four available
                formats depending on the value of the global title indicator (GTI).
                This example illustrates a format for a GTI equal to [0001].

-continued-

---

display (continued)

Examples of the display command (continued)

Example   **Task, response, and explanation**

---

display trace0010 line ↵
where

trace0010   specifies the file name

        Task: Display a specified file in line format.

        Response:
```
C7MN100 date time lognums INFO INCOMING_LINK_MSG
SCCP sccp_msg_type MSG TYPE DPC = dpc OPC = opc
  CALLED PARTY ADDR
   ADDR INDC = #xx
   PCI  = bool PC = point_code
   SSNI = bool SSN = ssn_type                               (#xx)
   GTI  = gt_type                                          [0010]
   RTG  = rtg-type                                            [0]
   NUI  = bool
  TRANS TYPE     = #xx
  DIGITS         = #xx xx xx xx xx xx xx
```

        Explanation: This command displays the file named trace0010 in line format.
                        The called and calling party address information can have one of
                        four available formats depending on the value of the GTI. This
                        example illustrates a format for a GTI equal to [0010].

-continued-

display (continued)

Examples of the display command (continued)

Example   **Task, response, and explanation**

---

display trace0011 line ↵
where

trace0011   specifies the file name

       Task: Display a specified file in line format.

       Response:
```
C7MN100 date time lognums INFO INCOMING_LINK_MSG
SCCP sccp_msg_type MSG TYPE DPC = dpc OPC = opc
  CALLED PARTY ADDR
   ADDR INDC = #xx
   PCI  = bool PC = point_code
   SSNI = bool SSN = ssn_type                            (#xx)
   GTI  = gt_type                                       [0011]
   RTG  = rtg-type                                         [0]
   NUI  = bool
  TRANS TYPE    = #xx
  NUM_PLAN      = nm_plan_type [0000] ENCODING = [0000]
  DIGITS        = #xx xx xx xx xx xx xx
```

       Explanation: This command displays the file named trace0011 in line format.
                  The called and calling party address information can have one of
                  four available formats depending on the value of the GTI. This
                  example illustrates a format for a GTI equal to [0011].

-continued-

---

display (continued)

Examples of the display command (continued)	
Example	**Task, response, and explanation**

display trace0100 line ↵
where

trace0100   specifies the file name

Task: Display a specified file in line format.

Response:
```
C7MN100 date time lognums INFO INCOMING_LINK_MSG
SCCP sccp_msg_type MSG TYPE DPC = dpc OPC = opc
  CALLED PARTY ADDR
   ADDR INDC = #xx
   PCI  = bool PC = point_code
   SSNI = bool SSN = ssn_type                            (#xx)
   GTI  = gt_type                                       [0100]
   RTG  = rtg-type                                          [0]
   NUI  = bool
  TRANS TYPE     = #xx
  NUM_PLAN       = nm_plan_type [0000] ENCODING = [0000]
  ADDR NAT       = addr_nat_type                         (#xx)
  DIGITS         = #xx xx xx xx xx xx xx
```

Explanation: This command displays the file named trace0100 in line format.
                  The called and calling party address information can have one of
                  four available formats depending on the value of the GTI. This
                  example illustrates a format for a GTI equal to [0100].

End

Response

The following table provides an explanation of the response to the display
command.

display (end)

Response for the display command	
MAP output	**Meaning and action**
File cannot be found.	
	Meaning: The file name that was entered cannot be found in the current symbol dictionary. Either the wrong file name was entered or the file name was not in the symbol dictionary. The command halts execution.
	Action: Verify the file name and enter the correct name, or list files in the volumes where the file is likely to be.

monitor

Function

Use the monitor command to monitor messages on a CCS7 link. The C7MON directory monitor command supports three message classes:

- signalling network management messages (selected by the mtp parameter)
- SCCP node messages (selected by the sccp parameter)
- trunk signalling messages messages generated by User Parts (UP) such as TUP+, BTUP, TUPs, and ISUP (selected by the up parameter)

monitor command parameters and variables

Command	Parameters and variables
monitor	mtp *lnkset_nam link_num*

sccp   *ni*   ccitt   basic   *pc*
austria   region
china   *zone*   exchange   *sigpoint*
intl   areantw

ttc   *mainarea   subarea   unit*
jpn

up   clli   *clli   trunk*

dp   *ni*   ccitt   basic   *pc*
austria   region
china   *zone*   exchange   *sigpoint*   *cic*
intl   areantw

jpn   *mainarea   subarea   unit*
ttc

Parameters and variables	Description
areantw	This variable specifies the area network of the point code (PC), in intl format, of the CCITT message to monitor. The valid entry range is 0-255. Entering 0 monitors for all area networks.
areaunit	This variable specifies DPC area unit number of the TTC or JPN message to monitor. The valid entry range is 0-127. Entering 0 monitors for all main units.

-continued-

monitor (continued)

monitor command parameters and variables (continued)	
Parameters and variables	**Description**
austria	This parameter identifies the format of the CCITT test message to monitor.
basic	This parameter identifies the DPC format of the CCITT test message to monitor.
ccitt	This parameter identifies the network type of the message.
china	This parameter identifies the format of the CCITT test message to monitor.
cic	This variable specifies the unique circuit identification code (CIC). The CIC is datafilled in Table C7TRKMEM. The valid entry range is 0-16383.
clli	This parameter indicates that the CLLI trunk name and the external trunk number will be used with the up parameter to specify messages to monitor.
clli	This variable specifies the CLLI name of the trunk to be monitored. The valid entry is a string.
dp	This parameter indicates that the network indicator, the network type, PC entries, and the CIC entries will be used with the up parameter to specify messages to monitor.
exchange	This variable specifies the PC of the exchange, in china format, of the CCITT message to monitor. The valid entry range is 0-127. Entering 0 monitors for all exchanges.
intl	This parameter identifies the format of the CCITT test message to monitor.
jpn	This parameter identifies the network type of the message.
link_num	This variable specifies the number of the linkset to be monitored. The valid entry range is 0-15.
lnkset_nam	This variable specifies the name of the linkset to be monitored. The valid entry is a string.
mainarea	This variable specifies the DPC main area number of the TTC or JPN message to monitor. The valid entry range is 0-31. Entering 0 monitors for all main areas.
mtp	This parameter indicates the network type for which to monitor. The mtp parameter monitors signalling network management messages on a signalling link by entering the linkset name and link number.
-continued-	

monitor (continued)

monitor command parameters and variables (continued)	
Parameters and variables	**Description**
ni	This variable specifies the network indicator of the message. The valid entries are as follows: • intl • intlsp • natl • natlsp
pc	This variable specifies the PC, in basic format, of the CCITT message to monitor. The valid entry range is 0-16383. Entering 0 monitors for all PCs.
region	This variable specifies the region of the PC, in austria format, of the CCITT message to monitor. The valid entry range is 0-15. Entering 0 monitors for all regions.
sccp	This parameter indicates the network type for which to monitor. The sccp parameter traces messages to a SCCP node specified by the network indicator, network type, and PC entries.
sigpoint	This variable specifies the signal point of the PC, in the specified format, of the CCITT message to monitor. The valid entry range for austria format is 0-31. The valid entry range for china or intl format is 0-7. Entering 0 monitors for all signal points.
subarea	This variable specifies the DPC subarea number of the TTC or JPN message to monitor. The valid entry range is 0-15. Entering 0 monitors for all subareas.
trunk	This variable specifies the external trunk number of the trunk to be monitored. The valid entry range is 0-2047.
ttc	This parameter identifies the network type of the message.
up	This parameter indicates the network type for which to monitor. The up parameter monitors trunk signalling messages generated by the UP such as TUP+, BTUP , TUPs, and ISUP. Trunks are specified by the CLLI name and external trunk number, or trunks are specified by the network indicator, network type, PC entries and CIC entries.
zone	This variable specifies the zone of the PC, in the specified format, of the CCITT message to monitor. The valid entry range for intl format is 0-7. The valid entry range for austria format is 0-31. The valid entry range for china format is 0-15. Entering 0 monitors for all zones.
End	

monitor (continued)

Qualifications

The monitor command is qualified by the following limitations, exceptions, and restrictions:

- The C7MON and SIGRTU directories can run concurrently on the same PM, but the maximum, aggregate number of traces allowed for both tools is eight. Allocation of these eight trace criteria between C7MON and SIGRTU will be performed on a "first-come-first-served" basis.

- The number of C7MON directory users in a switch is limited to 20.

- If either the C7TU or SIGRTU directory is tracing, C7MON tracing will be prohibited. (C7MON tracing is not considered to be in progress until the C7MON directory start command is issued.) A warning message will display.

- Tracing will be stopped by the system under heavy traffic on the MSB7s.

- Switch of activity (SwAct) on an MSB7 will terminate the C7MON trace for the peripheral module (PM).

- Any type of central controller (CC) restart will reset C7MON and all trace criteria will be lost. (That is, the templates will have to be reentered and the trace will have to be restarted again.)

- SCCP message header decoding only will be preformed if the display long command string is used and the SCCP message type is either UDT or UDTS. Other SCCP messages types will be decoded and displayed in hex format. Decoding also will be performed on CCITT message types.

- C7MON cannot be used to monitor ANSI network type messages.

- C7MON only is available in the 8 MByte LIU7 loads.

Examples

The following table provides examples of the monitor command.

monitor (continued)

Examples of the monitor command
Example Task, response, and explanation

monitor mtp lonlks1 0 ↵
where

lonlks1	specifies the linkset name
0	specifies the linkset number

	Task:	Monitor signalling network messages on a specified signalling link.
	Response:	Not currently available
	Explanation:	This command monitors signalling network messages on signalling link number 0 (named LONLKS1).

monitor sccp intl ccitt7 intl 1 1 002 4 ↵
where

intl	specifies the network indicator of the CCITT message to monitor
intl	specifies the network type of the CCITT message to monitor
1	specifies the PC member of the CCITT message to monitor
002	specifies the PC cluster of the CCITT message to monitor
4	specifies the PC network of the CCITT message to monitor

	Task:	Trace messages to a SCCP node specified by the network indicator, network type, and PC entries.
	Response:	Not currently available
	Explanation:	This command traces CCITT messages to a SCCP node specified by the network indicator, network type, and PC entries.

monitor up clli lon12wn7 0 ↵
where

lon12wn7	specifies the CLLI name of the trunk to monitor
0	specifies the external trunk number of the trunk to monitor

	Task:	Monitor messages related to a trunk specified by the CLLI name and external trunk number.
	Response:	Not currently available
	Explanation:	This command monitors messages related to the LON12WN7 trunk.

-continued-

monitor (continued)

Examples of the monitor command (continued)	
Example	**Task, response, and explanation**

monitor up dp intl ccitt7 intl 1 002 3 0 ↵
where

intl	specifies the network indicator of the CCITT message to monitor
intl	specifies the network type of the CCITT message to monitor
1	specifies the PC of the zone of the CCITT message to monitor
002	specifies the PC of the area network of the CCITT message to monitor
3	specifies the PC of the signal point of the CCITT message to monitor
0	specifies the CIC of the CCITT message to monitor

Task:	Monitors messages related to a trunk specified by the network indicator, network type, PC specification, and CIC entries.
Response:	Not currently available
Explanation:	This command string monitors messages on the specified trunk.

End

Responses

The following table provides explanations of the responses to the monitor command.

Responses for the monitor command	
MAP output	**Meaning and action**

`Error: Invalid link number`

Meaning:	You entered a link number that is not datafilled for that linkset in the C7LINK table. The command halts execution. The monitor command is not added to the match table.
Action:	Verify the link number and retry the monitor command with the correct link number.

-continued-

monitor (end)

Responses for the monitor command (continued)	
MAP output	**Meaning and action**
Error: Invalid linkset name	
	Meaning: You entered a linkset name that is not datafilled in the C7LKSET table. The command halts execution. The monitor command is not added to the match table.
	Action: Verify the linkset name and retry the monitor command with the correct linkset name.
Error: Invalid trunk clli.	
	Meaning: The CLLI you specified cannot be found in Table CLLI. The command halts execution.
	Action: Verify the CLLI name and retry the monitor command.
Error: Match Table full.	
	Meaning: There already are eight entries in the match table. The command halts execution.
	Action: Remove a monitor or intercept request before retrying the monitor command.
Error: Trunk is not #7.	
	Meaning: The trunk you specified is not a CCS7 trunk. The command halts execution.
	Action: Do not try to monitor non-CCS7 trunks.
Error: Trunk not found.	
	Meaning: The trunk you specified is unequipped. The command halts execution.
	Action: Verify the trunk and retry the monitor command.
End	

q

Function

Use the q command to receive online documentation for the C7MON directory.

q command parameters and variables	
Command	**Parameters and variables**
q	c7mon *command_nam*

Parameters and variables	**Description**
c7mon	This parameter lists the commands in the C7MON directory and provides a short description of the function of each command.
command_nam	This variable specifies a valid C7MON directory command name. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the q command.

Example of the q command	
Example	**Task, response, and explanation**
q c7mon ↵	
	Task: Access online documentation.

	Response COMMANDS AVAILABLE IN C7MON ENVIRONMENT.
	    MONITOR - DEFINE A TRACE ENTRY.
	    QC7MON - LIST THE TRACE ENTRIES.
	    DELETE - DELETE A TRACE ENTRY.
	    START - START TRACE.
	    STOP - STOP TRACE.
	    DISPLAY - DISPLAY THE MESSAGES STORED IN A
	              FILE.
	ENTER "Q <COMMAND NAME>" FOR MORE INFORMATION.

	Explanation: This example typifies a response for the q command.

q (end)

Response

The following table provides an explanation of the response to the q command.

Response for the q command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

qc7mon

Function

Use the qc7mon command to display CCS7 templates stored in the match table.

qc7mon command parameters and variables	
Command	**Parameters and variables**
qc7mon	*users* all
Parameters and variables	**Description**
users	Omitting this entry forces the system to default to displaying only those templates owned by the user.
all	This parameter displays all templates in the match table.

Qualifications

None

Examples

The following table provides examples of the qc7mon command.

Examples of the qc7mon command	
Example	**Task, response, and explanation**
qc7mon all ↵ *where* all	specifies that all templates in the match table are to be displayed

Task:	Display all templates in the match table.	
Response:	Template User Templatenumber name -------- --------- ------------------------ 1 MAP1 UP TORTOMONTCCS7 5 2 MAP2 SCCP INTL CCITT BASIC 1222 3 MAP4 MTP C7LKSET 4 MAP1 UP NATL ANSI 12 11 1 304 5 MAP1 UP TORTOOTWCCS7 8	
Explanation:	All templates in the match table display.	

-continued-

qc7mon (end)

Examples of the qc7mon command (continued)

Example   **Task, response, and explanation**

---

qc7mon ↵

Task:	Display templates owned by the user.

Response:
```
Template    User       Templatenumber name
--------    --------    ------------------------
1           MAP1       UP   TORTOMONTCCS7 5
4           MAP1       UP   NATL ANSI 12 11 1 304
5           MAP1       UP   TORTOOTWCCS7 8
```

Explanation: The system defaults to displaying all templates owned by the user named MAP1.

End

Response

The following table provides an explanation of the response to the qc7mon command.

Response for the qc7mon command

MAP output   **Meaning and action**

```
No templates to display.
```

Meaning: If the all parameter is specified, there are no templates in the match table. If the all parameter is not specified, the user has no templates.

Action: None

---

quit

Function

Use the quit command to exit the C7MON directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⌐ *1 level* ¬ │ all │ │ *name* │ └ *n_levels* ┘

Parameters and variables	Description
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualification

> **CAUTION**
> **Do not leave the trace on when you quit the C7MON level.**
> This command must be used with extreme caution. If the trace is left on when you quit this tool, the only way to turn it off is by entering the C7MON directory on the same terminal.

This command must be used with extreme caution. If the trace is left on when you quit this tool, the only way to turn it off is by entering the C7MON directory on the same terminal.

Example

The following table provides an example of the quit command.

quit (continued)

Example of the quit command	
Example	**Task, response, and explanation**
quit ↵	
	Task: Exit from this directory.
	Response: `There are templates defined by you.  Do you want` `to delete them?  yes/no`
	Explanation: This command gives you the option of deleting templates before exiting this directory and returning to the CI MAP level.

Responses

The following table provides explanations of the responses to the quit command.

Responses for the quit command	
MAP output	**Meaning and action**
`There are templates defined by you.` `To delete you must reenter C7MON on this terminal.`	
	Meaning: This is a reminder that the templates must be deleted before other users can use them.
	Action: Access the C7MON directory on this terminal to free the templates.
`Trace is on. Do you want to leave it on?  Yes/No`	
	Meaning: The trace was not turned off before entering the quit command. Respond to the prompt with yes to quit this tool and leave the trace on. Respond to the prompt with not to turn the trace off. (If no is entered, the system prompts you to choose whether or not to delete the templates.)
	Action: Supply the appropriate answer.
	-continued-

quit (end)

Responses for the quit command (continued)	
MAP output	**Meaning and action**
`Trace is on. To turn it off you must reenter C7MON on this terminal.`	
	Meaning: This is a reminder that tracing is on.
	Action: Access the C7MON directory on this terminal to stop the trace.
End	

start

Function

Use the start command to start the trace.

start command parameters and variables		
Command	**Parameters and variables**	
start	map	short / long
	disk	
	log	short / line / long

Parameters and variables	**Description**
map	This default parameter displays messages on the MAP. Either omit this entry to force the default action or enter the start map command string.
short	This default parameter produces the display in short format. Enter omit this entry to force the default action or enter the start log command string. (This default parameter is valid with the log device only.)
short	Omitting this entry forces the system to default to setting the display format to short. Enter either the start command or the start map command string.
disk	This parameter indicates the device on which to display messages. No subsequent parameters are entered, but the system will prompt for a logical device and file name.
line	This parameter produces a display in line format. This parameter is valid with the log device only.
log	This parameter indicates the device on which to display messages. This parameter specifies that the output is directed to a log.
long	This parameter produces a display in long format. For User Part (UP) traces, this parameter decodes the UP header and displays UP data. For SCCP traces, only Unitdata (UDT) and Unitdata Services (UDTS) data are decoded. The calling and called party address data are decoded according to the global title indicator (GTI) field. Subsequent UDT or UDTS data displays in hex format. All other SCCP message types are identified and subsequent trace data are displayed in hex format.

start (continued)

Qualifications

The start command is qualified by the following exceptions, restrictions, and limitations:

- The C7MON and SIGRTU directories can run concurrently on the same PM, but the maximum, aggregate number of traces allowed for both tools is eight. Allocation of these eight trace criteria between C7MON and SIGRTU will be performed on a "first-come-first-served" basis.

- If either the C7TU or SIGRTU directory is tracing, C7MON tracing will be prohibited. (C7MON tracing is not considered to be in progress until the C7MON directory start command is issued.) A warning message will display.

- The number of C7MON directory users in a switch is limited to 20.

- Tracing will be stopped by the system under heavy traffic on the MSB7s.

- Switch of activity (SwAct) on an MSB7 will terminate the C7MON trace for the peripheral module (PM).

- Any type of central controller (CC) restart will reset C7MON and all trace criteria will be lost. (That is, the templates will have to be reentered and the trace will have to be restarted again.)

- SCCP message header decoding only will be performed if the display long command string is used and the SCCP message type is either UDT or UDTS. Other SCCP messages types will be decoded and displayed in hex format. Decoding also will be performed on CCITT message types.

- C7MON only is available in the 8 MByte LIU7 loads.

Examples

The following table provides examples of the start command.

Examples of the start command		
Example	**Task, response, and explanation**	
start ↵		
	Task:	Direct the output to default device in default format.
	Response:	`Trace started.`
	Explanation:	The output has been directed to the MAP and displays in short format.
	-continued-	

start (continued)

Examples of the start command (continued)		
Example	**Task, response, and explanation**	
start disk ↵		
	Task:	Direct the output to a disk.
	Response:	Enter device and file name: sfdev trace0100
	Explanation:	The system prompts for a device and file name. The output of the trace will be sent to a specified disk and file.
start log ↵		
	Task:	Direct the output to a specified device.
	Response:	Trace started.
	Explanation:	The output has been directed to a log. Since no format was specified in the command string, the system assumes the short format default.
End		

Responses

The following table provides explanations of the responses to the start command.

Responses for the start command	
MAP output	**Meaning and action**
`<directory>` IS IN USE. CANNOT START C7MON.	
	Meaning: Tracing already is in progress for either the C7TU or the SIGMON directory.
	Action: None
-continued-	

start (end)

Responses for the start command (continued)	
MAP output	**Meaning and action**
Enter device and file name.	
	Meaning: The disk option has been selected. A device and file name must be specified to continue.
	Action: Reenter the start command with a valid device name and file name.
File already exists.	
	Meaning: A file with the specified name already exists. The system halts execution of the command.
	Action: Reenter the start command with a new file name.
Invalid device name.	
	Meaning: The device name entered is not found. The system halts execution of the command.
	Action: Verify the device name and reenter the start command.
No templates specified.	
	Meaning: No templates have been specified. At least one template must be specified before starting the trace.
	Action: Specify a template and reenter the start command.
End	

stop

Function

Use the stop command to stop the tracing function. A log is produced if there are no other active users.

stop command parameters and variables	
Command	**Parameters and variables**
stop	There are no parameters or variables.

Qualification

WARNING
The test message is modified using this command.
The test message is modified using this command. The old message is overwritten with the altered version.

The test message is modified using this command. The old message is overwritten with the altered version.

Example

The following table provides an example of the stop command.

Example of the stop command	
Example	**Task, response, and explanation**
stop ↵	
	Task: Stop the C7MON trace.
	Response: Tracing stopped.
	Explanation: C7MON tracing has been stopped.

Response

The following table provides an explanation of the response to the stop command.

stop (end)

Response for the stop command	
MAP output	**Meaning and action**
Tracing stopped	
	Meaning: C7MON tracing has been stopped.
	Action: None

C7TU level commands

Use the C7TU (CCS7 test utility) level of the MAP to access directories and commands that monitor CCS7 messages or links on both message switch and buffer 7 (MSB7) and link interface units 7 (LIU7). The C7TU directory commands can be used on the Service Switching Point (SSP), Signal Transfer Point (STP), and Service Control Point (SCP) of the Digital Multiplex System (DMS) product line.

The C7TU tool has a throttling mechanism to prevent a large number of log messages to be sent from any single link. In this way, the message path between the peripheral and the central controller (CC) as well as the computing module (CM) are not congested with C7TU messages. The output from message monitoring sessions can be directed to a selected device.

CAUTION
There is a real time impact on the node using C7TU.
Use the C7TU tool with caution. There is a real time impact on the node using C7TU. This impact has been minimized by limiting the number of links that can be monitored, as well as the number of messages that can be entered into the match table for monitoring.

In addition to the C7TU directory commands c7turec, c7tuprt, dpc, and msgcode, this directory provides commands which access subdirectories. The C7TU directory must be entered in order to access the C7TUDTC directory, the C7TURFC directory, and both the basic and password-protected versions of C7TULINK directory.

Accessing the C7TU level

All commands and directories available at the C7TU MAP level are accessed in the usual manner except the password-protected C7TULINK commands. To access the C7TU level for all commands except the password-protected C7TULINK commands, enter the following command from the CI level:

c7tu ↵

The password-protected C7TULINK environment is identified as the C7TULINK_ILPT7 (Integrated Link Protocol Test Tool) environment. In order to gain access to password-protected C7TULINK commands, you must provide a valid password that resides in the tool supervisor (TOOLSUP). The TOOLSUP also records each time the tool is used. The entry sequence for accessing the password protected C7TULINK environment is described in the C7TULINK directory introduction beginning on page C-86.

C7TU commands

The commands available at the C7TU MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

C7TU commands	
Command	**Page**
c7tudtc	C-37
c7tulink	C-39
c7tuprt	C-41
c7turec	C-45
c7turfc	C-49
dpc	C-51
help	C-55
msgcode	C-57
q	C-59
quit	C-61

c7tudtc

Function

Use the c7tudtc command to access the C7TUDTC directory.

c7tudtc command parameters and variables	
Command	**Parameters and variables**
c7tudtc	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the c7tudtc command.

Example of the c7tudtc command	
Example	**Task, response, and explanation**
c7tudtc ↵	
	Task: Access the C7TUDTC directory.
	Response: `C7TUDTC:`
	Explanation: You have accessed the C7TUDTC directory.

Responses

The following table provides explanations of the responses to the c7tudtc command.

Responses for the c7tudtc command	
MAP output	**Meaning and action**
`MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.`	
	Meaning: The C7TUDTC directory is not loaded or must be accessed through another directory.
	Action: None
-continued-	

c7tudtc (end)

Responses for the c7tudtc command (continued)	
MAP output	**Meaning and action**
`Undefined command "<command>".`	
	Meaning: The command you entered is spelled incorrectly, this directory is accessed using another entry code, or the C7TUDTC directory is not included in this software load.
	Action: None
End	

c7tulink

Function

Use the c7tulink command to access the C7TULINK directory. There are two versions of C7TULINK. The basic version (identified as the C7TULINK_PMT7 environment) accesses commands that monitor messages only; access to commands for building, sending, or intercepting messages is not allowed.

The C7TULINK directory also has a password-protected version identified as the C7TULINK_ILPT7 environment. This password-protected version not only accesses the same basic commands as C7TULINK_PMT7 but also accesses commands used to build, send, and intercept messages.

The C7TULINK directory version you access depends on the entries you made at the C7TU MAP level.

c7tulink command parameters and variables	
Command	**Parameters and variables**
c7tulink	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the c7tulink command.

Example of the c7tulink command		
Example	**Task, response, and explanation**	
c7tulink ↵		
	Task:	Access the C7TULINK environment.
	Response:	C7TULINK:
	Explanation:	You have accessed the basic C7TULINK environment.

Responses

The following table provides explanations of the responses to the c7tulink command.

c7tulink (end)

Responses for the c7tulink command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The C7TULINK directory is not loaded or must be accessed through another directory.
	Action: None
Undefined command "<command>".	
	Meaning: The command you entered is spelled incorrectly, this directory is accessed using another entry code, or the C7TULINK directory is not included in this software load.
	Action: None

c7tuprt

Function

Use the c7tuprt command to output all CCS7 messages saved in a specified file. The messages are sent to the file using the C7TU directory c7turec command.

The c7tuprt command examines the device and file names and verifies that the files are in the correct format. This check consists of reading in the first line of the file and confirming that this line matches the string that always is written when the c7turec command opens the file. If the file is in the correct format, the c7tuprt command reads in each message in the file and formats the messages using the same routines formerly used by the log system to print C7TU logs.

c7tuprt command parameters and variables		
Command	**Parameters and variables**	
c7tuprt	*file*	<u>*screen*</u>
Parameters and variables	**Description**	
<u>*screen*</u>	Omitting this entry forces the system to default to display the messages on the screen unless the SYS directory commands send or record are used to redirect the formatted output to another device.	
file	This variable specifies a valid file name.	

Qualifications

None

Example

The following table provides an example of the c7tuprt command.

c7tuprt (continued)

Example of the c7tuprt command

Example	Task, response, and explanation

c7tuprt   **tempfile** ↵
where

tempfile   specifies the file name

Task: Print all CCS7 messages in the specified file.

Response:
```
TIME: 09:14:37 INCOMING LINK MSG
C7 HEADER: LEN= 34 MSG= 2 LINK= 1 SLC= 0 CLLI=
C7LKSET
C7 SIO: NETWORK= 2 PRIORITY= 2 SERV IND= 5
C7 LABEL: DPC = 001-001-001 OPC = 002-002-002 SLS
= 2
C7 DATA FOLLOWING HEADER: 01 01 01 01 01 01 01 01
01 01

TIME: 09:14:37 OUTGOING LINK MSG
C7 HEADER: LEN= 32 MSG= 2 LINK= 2 SLC= 1 CLLI=
C7LKSET2
C7 SIO: NETWORK= 2 PRIORITY= 2 SERV IND= 2
C7 LABEL: DPC = 003-003-003 OPC = 001-001-001 SLS
= 2
BODY: 02 03 04 05 06 07 08 01 02

TIME: 09:14:38 INVALID MESSAGE
INVALID MESSAGE TYPE
1909 55 07 E4 FF 32 45 09 A4 D2 FF C3 E9 D0 AA ED

TIME: 09:15:01 C7TU TRACING ON LIU7 201
```

Explanation: This command string displays all CCS7 messages saved in the file named tempfile.

Responses

The following table provides explanations of the responses to the c7tuprt command.

c7tuprt (continued)

Responses for the c7tuprt command

MAP output	Meaning and action
`Error: File is not in C7TU format.`	
	Meaning: The specified file is not a valid C7TU log file. The command halts execution. No C7TU log messages will be interpreted and displayed.
	Action: Retry the command with a valid C7TU log file.
`Error: While opening file.`	
	Meaning: A system error occurred. The C7TU was unable to open a file on the specified device. The command halts execution. No messages will be recorded.
	Action: Retry the command.
`Error while reading file header.`	
	Meaning: An error occurred when trying to read the file header of the specified file. The command halts execution. The file will be closed.
	Action: None
`Error while reading next record.`	
	Meaning: An error occurred when trying to read a C7TU log record from the specified file. The command halts execution. The file will be closed.
	Action: None

`TIME:   <time>`
`C7TU TRACING ON`
`C7TU TRACING OFF   <pm>    <num>`	

	Meaning: This report is produced when a peripheral is selected or removed by the user.
	Action: None
-continued-	

c7tuprt (end)

Responses for the c7tuprt command (continued)

MAP output   **Meaning and action**

```
TIME:<timestamp> INVALID MESSAGE INVALID MESSAGE TYPE <msgtype> <hexbyte>
```

> **Meaning:** This is the response to a message that the C7TU is unable to interpret. The invalid message type and the complete message in hexadecimal format follow the time stamp of the message.
>
> **Action:**   None

End

---

c7turec

Function

Use the c7turec command to specify whether to use the log system to display messages as they occur, or to send all messages from the peripheral modules (PMs) to a file. (It is more efficient to send a large number of messages to a file rather than to flood the log system.)

c7turec command parameters and variables		
Command	**Parameters and variables**	
c7turec	query start *device_name* *file_name* stop	
Parameters and variables	**Description**	
device_name	This variable specifies the name of the device where the CCS7 messages will be stored.	
file_name	This variable specifies the name of the file where the CCS7 messages will be stored.	
query	This parameter queries the active recording device and file.	
start	This parameter starts recording the CCS7 messages on a specified device and file.	
stop	This parameter stops recording the CCS7 messages on the specified device and file.	

Qualification

The c7turec command does not provide display functions; use the C7TU directory c7tuprt command to display all messages in the file.

Examples

The following table provides examples of the c7turec command.

c7turec (continued)

Examples of the c7turec command	
Example	**Task, response, and explanation**
c7turec query ↵	
	Task: Query the active recording device and file.
	Response: C7TU RECORD onto SFDEV TEMPFILE
	Explanation: The active recording device and file are identified.
c7turec start sfdev tempfile ↵ *where* sfdev specifies the name of the device where the messages will be stored tempfile specifies the name of the file where the messages will be stored	
	Task: Start recording on a specified file.
	Response: C7TU RECORD START onto sfdev tempfile
	Explanation: The file has been opened successfully by the C7TU and is ready to save messages.
c7turec stop ↵	
	Task: Stop recording.
	Response: C7TU RECORD STOP
	Explanation: The system stopped recording messages.

Responses

The following table provides explanations of the responses to the c7turec command.

Responses for the c7turec command	
MAP output	**Meaning and action**
C7TU RECORD START onto device file	
	Meaning: The file has been opened successfully by the C7TU.
	Action: None
-continued-	

c7turec (end)

Responses for the c7turec command (continued)	
MAP output	**Meaning and action**
C7TU RECORD STOP.	
	Meaning: The file has been closed successfully by the C7TU. This message is displayed in response to the query command when the C7TU is not recording to a file.
	Action: None
Error: Device is not valid.	
	Meaning: You specified a device name that is not valid or is not recognized. The command halts execution. No messages will be recorded.
	Action: Retry the command with a valid device name.
Error: Unable to get file information.	
	Meaning: You specified a file name that is not valid or is not recognized. The command halts execution. No messages will be recorded.
	Action: Retry the command with a valid file name.
Error: Unable to get volume information.	
	Meaning: You specified a device name that is not valid or is not recognized. The command halts execution. No messages will be recorded.
	Action: Retry the command with a valid device name.
Error: While creating the file.	
	Meaning: The system failed to create the specified file at the specified device. The command halts execution. No messages will be recorded.
	Action: Retry the command with a different device name.
Recording already started.	
	Meaning: This response indicates that the C7TU already is recording.
	Action: None
End	

c7turfc

Function

Use the c7turfc command to access the C7TURFC directory.

c7turfc command parameters and variables	
Command	**Parameters and variables**
c7turfc	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the c7turfc command.

Example of the c7turfc command		
Example	**Task, response, and explanation**	
c7turfc ↵		
	Task:	Access the C7TURFC directory.
	Response:	C7TURFC:
	Explanation:	You have accessed the C7TURFC directory.

Responses

The following table provides explanations of the responses to the c7turfc command.

Responses for the c7turfc command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The C7TURFC directory is not loaded or must be accessed through another directory.
	Action: None
-continued-	

c7turfc (end)

Responses for the c7turfc command (continued)	
MAP output	**Meaning and action**
`Undefined command "<command>".`	
	Meaning: The command you entered is spelled incorrectly, this directory is accessed using another entry code, or the C7TURFC directory is not included in this software load.
	Action: None
End	

dpc

Function

Use the dpc command to monitor a routeset for changes in availability and congestion or to query a routeset state. The responses from the dpc command are produced as a C7TU log.

dpc command parameters and variables		
Command	**Parameters and variables**	
dpc	query report	*routset* ⌈ off ⌉ ⌊ on ⌋

Parameters and variables	**Description**
off	This parameter disables reporting for routeset state or congestion level changes.
on	This parameter enables reporting for routeset state or congestion level changes.
query	This parameter queries a routeset state.
report	This parameter reports any routeset state changes or changes in congestion level
routset	This variable specifies a valid routeset name that is datafilled in able C7RTESET.

Qualifications

None

Examples

The following table provides examples of the dpc command.

Examples of the dpc command	
Example	**Task, response, and explanation**
dpc report on ↵	
	Task: Enable routeset state reporting.
	Response: Not currently available
	Explanation: This command enables routeset state reporting.
-continued-	

dpc (continued)

Examples of the dpc command (continued)	
Example	**Task, response, and explanation**
dpc query c7rteset1 ↵ *where* c7rteset1　　specifies a valid routeset name	
	Task:　　　Query the routeset state of a specified route. **Response:**　　Not currently available **Explanation:**　This command queries the routeset state of the c7rteset1 route.
End	

Responses

The following table provides explanations of the responses to the dpc command.

Responses for the dpc command	
MAP output	**Meaning and action**
`ERROR: CANNOT FIND DPC`	
	Meaning: The dpc command was unable to find the destination point code (DPC) associated with the routeset. The command halts execution. No messages will be printed. **Action:** Verify that the routeset name ID datafilled in the C7RTESET table. Retry the command with the correct routeset name.
`ERROR: INVALID ROUTESET NAME`	
	Meaning: The user specified a routeset name that is not datafilled in the C7RTESET table. The command halts execution. No messages will be printed. **Action:** Retry the command with a valid routeset name.
-continued-	

dpc (end)

Responses for the dpc command (continued)	
MAP output	**Meaning and action**
ERROR: QUERY FAILED	
	Meaning: The query command was unable to query the DPC associated with the routeset. The command halts execution. No messages will be printed.
	Action: Verify the routeset and retry the query command.
INVALID DPC OPERATION	
	Meaning: The user has specified an operation that is not allowed with the dpc command. The command halts execution. No messages will be printed.
	Action: Retry the dpc command with the correct options.
End	

help

Function

Use the help command to receive online documentation for the C7TU directory.

help command parameters and variables	
Command	**Parameters and variables**
help	c7tu

Parameters and variables	**Description**
c7tu	This parameter produces summary documentation for the commands in the C7TU directory.

Qualifications

None

Examples

The following table provides examples of the help command.

Examples of the help command		
Example	**Task, response, and explanation**	
help c7tu ↵		
	Task:	Access online documentation.

	Response:	COMMON CHANNEL SIGNALLING #7 TEST UTILITY
		C7TULINK - access the C7TU LINK test environment
		C7TUDTC - access the C7TU DTC test environment
		C7TUTRFC - access the C7TU TRAFFIC simulation
		   environment
		C7TUREC - record C7TU reports from pms to a
		   device
		C7TUPRT - print C7TU reports recorded on a device
		DPC - turn on/off routeset status change report
		MSGCODE - list C7TU message codes
		QUIT - exit C7TU
		ENTER "Q <command name> for more information.

	Explanation:	This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

msgcode

Function

Use the msgcode command to print a list of valid message codes that are available for use in the message code fields prompted for in the build and monitor commands. The message codes display in a hierarchical format. The hierarchy is distribution identification (DI), service indicator (SI), h0, and h1 (H0H1). Each level in the hierarchy has its own three- or four-letter message code.

msgcode command parameters and variables	
Command	**Parameters and variables**
msgcode	*msgcode*
Parameters and variables	**Description**
msgcode	This variable specifies a message code in a three- or four-letter format.

Qualifications

None

Example

The following table provides an example of the msgcode command.

msgcode (end)

Example of the msgcode command

Example   **Task, response, and explanation**

---

msgcode   **ext** ↵
where

ext   specifies the message code

               Task: Print a list of valid message codes.

               Response:

```
MSG CODE   DESCRIPTION                  DI  SI  H1H0
--------   ---------------------------  --  --  ----
EXT        C7 EXTERNAL                  04  X   XX
     SNM   SIGNALLING NETWORK MGT.      -   00  XX
       CHM CHANGEOVER/BACK MSGS         -   -   01
         COO CHANGEOVER ORDER           -   -   11
         COA CHANGEOVER ACK             -   -   21
         CBD CHANGEBACK DECLARATION     -   -   51
         CBA CHANGEBACK ACK             -   -   61
   .
   .
   .
```

               Explanation: This command string produces a list of message code fields.

Response

The following table provides an explanation of the response to the msgcode command.

Response for the msgcode command

MAP output   **Meaning and action**

`INVALID MSGCODE: ZPF`

        Meaning: You entered a message code that is not recognized by the C7TU. No message codes are displayed.

        Action: Check the message code entered to ensure it is correct and retry the command. Otherwise, display the entire message code table by entering the msgcode command with no parameters.

---

q

Function

Use the q command to receive online documentation for the C7TU directory.

q command parameters and variables	
Command	**Parameters and variables**
q	*command_nam*
Parameters and variables	**Description**
command_nam	This variable specifies a valid C7TU directory command. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Examples

The following table provides examples of the q command.

Examples of the q command	
Example	**Task, response, and explanation**
q c7turec ↵ *where* c7turec	 specifies a valid C7TU directory command
	Task: Access online documentation.

	Response: `START OR STOP RECORDING C7TU MESSAGES.  MAXIMUM`
	`32 CHARS FOR FILENAME.`
	`Parms:  <CMD>  {START <DEVICE> DEVICE name`
	`        <FILE> STRING,`
	`     STOP,`
	`     QUERY}`

	Explanation: This example typifies a response for the help command string.

Response

The following table provides an explanation of the response to the q command.

q (end)

Response for the q command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

quit

Function

Use the quit command to exit the C7TU directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	*1 level* all *name* *n_levels*

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	CI:
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)	
Example	**Task, response, and explanation**
quit all ↵	
	Task: Exit from all levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory	
	Task: Exit from a specified directory without leaving any other directories.
	Response: AMADUMP>>> >
	Explanation: The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵	
	Task: Exit from a specified number of levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End	

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

C7TUDTC level commands

Use the C7TUDTC (CCS7 test utility digital trunk controller) level of the MAP to enter the digital trunk controller (DTC) test environment.

Accessing the C7TUDTC level

To access the C7TUDTC level, enter the following command string from the CI level:

c7tu;c7tudtc ↵

C7TUDTC commands

The commands available at the C7TUDTC MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.   These commands are listed for reference only. These C7TUDTC commands are laboratory test commands and not intended for general field use.

C7TUDTC commands	
Command	**Page**
help	C-67
intercept	C-69
loop	C-71
monitor	C-73
quit	C-75
remove	C-79
restore	C-81
status	C-83

help

Function

Use the help command to receive online documentation for the C7TUDTC directory.

help command parameters and variables	
Command	**Parameters and variables**
help	*all* *command_nam*

Parameters and variables	Description
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
command_nam	This variable specifies a valid C7TUDTC directory command. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help ↵	
	Task: Access online documentation. **Response:** Not currently available. **Explanation:** This example typifies a response for the help command.

Response

The following table provides an explanation of the response to the help command.

help (end)

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

intercept

Function

Use the intercept command to intercept DTC messages.

intercept command parameters and variables	
Command	**Parameters and variables**
intercept	This command is a laboratory test command and is not intended for general field use.

loop

Function

Use the loop command to loop messages between two trunks.

loop command parameters and variables	
Command	**Parameters and variables**
loop	This command is a laboratory test command and is not intended for general field use.

monitor

Function

Use the monitor command to monitor DTC messages.

monitor command parameters and variables	
Command	**Parameters and variables**
monitor	This command is a laboratory test command and is not intended for general field use.

quit

Function

Use the quit command to exit the C7TUDTC directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⎡ *1 level* ⎤ ⎢ all ⎥ ⎢ *name* ⎥ ⎣ *n_levels* ⎦

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

remove

Function

Use the remove command to cancel C7TUDTC commands loop, intercept, or monitor.

remove command parameters and variables	
Command	**Parameters and variables**
remove	This command is a laboratory test command and is not intended for general field use.

restore

Function

Use the restore command to retransmit all loop, intercept, or monitor commands.

restore command parameters and variables	
Command	**Parameters and variables**
restore	This command is a laboratory test command and is not intended for general field use.

status

Function

Use the status command to display the status of the C7TUDTC MAP level.

status command parameters and variables	
Command	**Parameters and variables**
status	This command is a laboratory test command and is not intended for general field use.

C7TULINK level commands

Use the C7TULINK level of the MAP to access commands for monitoring CCS7 messages. Links can be monitored as well.

The C7TULINK directory select command is used to identify a maximum of four links to be monitored. The display command displays links and current entries in the match table. The commands dump, monitor, remove, and restore are used for specific messages. The commands build and alter are used to create or modify messages, and commands match and mask allow you to change mask and match entries beginning at a specified byte. The send command is used to take a message and inject it into a link. The intercept command is used to intercept messages coming off the link.

There are two versions of the C7TULINK environment. The basic C7TULINK environment (C7TULINK_PMT7) allows you to access commands that monitor messages only; you are not allowed to build, send, or intercept messages unless you provide a valid password when accessing the C7TU MAP level. The password-protected C7TULINK environment (C7TULINK_ILPT7) allows you to use the basic C7TULINK commands as well as commands used to build, send, or intercept messages.

CAUTION
You cannot enter message tracing criteria during another message tracing session.
Entering C7TULINK directory message tracing criteria is prohibited when any other message tracing is in progress. The reverse also is true. This restriction applies to C7TULINK, C7MON, and SIGRTU directories.

Accessing the C7TULINK level

To access the basic (C7TULINK_PMT7) C7TULINK level, enter the following command string from the CI level:

c7tu;c7tulink ↵

Accessing the password-protected C7TULINK monitoring environment

The password-protected C7TULINK environment is identified as the
C7TULINK_ILPT7 (Integrated Link Protocol Test Tool) environment. In
order to gain access to password-protected C7TULINK commands, you
must provide a valid password that resides in the tool supervisor
(TOOLSUP). The tool supervisor also provides a history of when the tool
was used.

To access the password-protected commands in the C7TULINK level, enter
the following commands from the CI level:

toolsup ↵

date ↵
where:
> *date* represents day/date/month/year/time

access on c7tu_ilpt7 ↵

valid_password ↵
where:
> *valid_password* represents a valid password

Note: If a valid password is entered, the system provides access and displays
these messages:

```
C7TU_ILPT7 permitted
C7TU_ILPT7 access will expire 48 hours from now.

                    ** WARNING **

You have permitted access to command(s) that require
skilled and knowledgable users. Proper use is required
to avoid possible service degradations. Please ensure
that only fully trained and qualified personnel proceed.
```

c7tu ↵

```
** ILPT7 - INTEGRATED LINK PROTOCOL TEST TOOL **

ILPT7 allows messages to be monitored or intercepted
on a CCS7 signaling link. In addition, messages may be
sent in or out on a CCS7 signaling link.

      ** WARNING ** WARNING ** WARNING **

C7TU should only be used under the strict supervision of
TAS or TELCO personnel who completely understand the
ramifications of using C7TU on a switch carrying traffic.
Improper use of C7TU can seriously degrade C7 traffic
```

```
capacity and/or cause total C7 or office failure.
DO YOU WISH TO CONTINUE ? Please confirm (YES or NO):
```

yes ↵

c7tulink ↵

C7TULINK commands

The commands available to you depend on whether or not you entered a
valid password at the C7TU level. All of the C7TULINK commands,
including those that are password-protected, are described in this chapter
and arranged in alphabetical order. The page number for each command is
listed in the following table.

C7TULINK commands	
Command	**Page**
alter	C-89
build	C-95
display	C-103
dump	C-105
help	C-109
intercept	C-113
mask	C-121
match	C-125
monitor	C-129
q	C-139
quit	C-141
remove	C-143
restore	C-145
select	C-147
send	C-151
status	C-155

alter

Function

Use the alter command to modify a test message that was added to the C7TU message table by the build command. Message length can be modified. The routing label can be changed by identifying a new network type, destination point code (DPC) and origination point code (OPC), and signalling link selector (SLS). Specific bytes in the message can be changed using the data parameter associated with a message code and message type. Or, the message format can be defined by binding it against a message type.

alter command parameters and variables	
Command	**Parameters and variables**
alter	*mssg_num* ⎡ data ⎡ *offset* ⎤ (1) ⎣ *hexbytes* ⎦ (2) length *length* (3) parms *prompt_ans* (4) routing *label ni* ⎡ *default* (5) ⎣ *prio dpc_mbr dpc_cls dpc_ntw* (6)
alter (continued)	(1) (2) (3) (4) (5) (6) *opc_mbr opc_cls opc_ntw sls* (end)

Parameters and variables	Description
default	Omitting this entry forces the system to default to settings for the priority, DPC, OPC, and SLS data. The default states include the following: • The default OPC is datafilled in Table C7NETWRK. • The default DPC is datafilled in Table C7RTESET. • The default priority is 0. • The default SLS is 0.
data	This parameter changes specific message bytes. If the data parameter is used, you must enter the necessary hex bytes w.r.t. the message code. The hex bytes are defaulted to zero. The message code may be one or two bytes long and does not necessarily follow the routing label immediately.
	-continued-

alter (continued)

alter command parameters and variables (continued)	
Parameters and variables	**Description**
dpc_cls	This variable specifies the DPC cluster number of the message to alter. The valid entry range is 0-255. Entering 0 alters all clusters.
dpc_mem	This variable specifies the DPC member number of the message to alter. The valid entry range is 0-255. (Entering 0 alters all members.)
dpc_netwk	This variable specifies the DPC area network of the message to alter. The valid entry range is 0-255. (Entering 0 alters all area networks.)
hexbytes	This variable string specifies the new hex bytes of the message body. The existing bytes are overridden in the message.
label	This variable specifies the routing label used in the CCS7 message. The valid values are ansi, ccitt, jpn, or ttc.
length	This parameter changes the length of a message identified by the message number.
length	This variable specifies the new length of the message identified by message number. The valid entry range is 0-256.
mssg_num	This variable specifies the message number of the selected message. The valid entry range is 0-7.
ni	This variable specifies the network indicator of the message. The valid entries are as follows: • intl • intlsp • natl • natlsp
offset	This variable specifies the starting offset of the CCS7 message bytes to be altered. The valid entry range is 0-256.
opc_cls	This variable specifies the OPC cluster number of the message to alter. The valid entry range is 0-255. (Entering 0 alters all clusters.)
opc_mem	This variable specifies the OPC member number of the ANSI/routing message to alter. The valid entry range is 0-255. (Entering 0 alters all members.)
opc_netwk	This variable specifies the OPC area network of the message to alter. The valid entry range is 0-255. (Entering 0 alters all area networks.)
-continued-	

alter (continued)

alter command parameters and variables (continued)	
Parameters and variables	**Description**
parms	This parameter changes the specified message in readable format.
prio	This variable specifies the CCS7 priority to alter. The valid entry range is 0-4. Entering 4 indicates a priority of all.
prompt_ans	This variable represents the system action when the parms parameter is used to alter the message in readable format. The system produces the current value of valid parameters and prompts only for parameters that are valid for the message you want to alter. When you complete the changes, enter the word "done" to signal completion. The message is updated when the new value is received.
routing	This parameter changes the routing label of the message.
sls	This variable specifies the SLS of the ANSI, CCITT, JPN, or TTC test message. The valid entry range for ANSI and JPN messages is 0-31. The valid entry range for TTC and CCITT messages is 0-15.
End	

Qualifications

The alter command is qualified by the following exceptions, restrictions, and limitations:

- Once saved, messages are retained in the message table even if you exit this MAP level.

- The old test message is overwritten with the altered version.

alter (continued)

Example

The following table provides an example of the alter command.

Example of the alter command

Example	Task, response, and explanation

alter 0 length 200 ↵
where

0	specifies the message number of the selected message
200	specifies the new length of the CCS7 test message being altered

Task: Change the length of a specified message.

Response:

```
C7TU MESSAGE        SIO            DPC          OPC        SLS
  num type length  ni pr  si    mem clu net  mem clu net
   0  SLTM   9       2  3   2    001 001 001  002 002 002    0
  Message bytes:
   0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
  -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
  00 00 09 00 00 02 01 B2 01 01 01 02 02 02 00 11 01 01

C7TU MESSAGE        SIO            DPC          OPC        SLS
  num type length  ni pr  si    mem clu net  mem clu net
   0  SLTM  200      2  3   2    001 001 001  002 002 002    0
  Message bytes:
   0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
  -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
  00 00 C8 00 00 02 01 B2 01 01 01 02 02 02 00 11 01 01
```

Explanation: The system displays the original message number 0 and the subsequent display of the altered message number 0.

Responses

The following table provides explanations of the responses to the alter command.

alter (end)

Responses for the alter command	
MAP output	**Meaning and action**
MESSAGE NUMBER <num> HAS NOT BEEN BUILT YET	
	Meaning: You entered a message number that has not been built in the message table. The alter command exits.
	Action: Retry the alter command with a valid message number.
THE OFFSET DOES NOT FALL WITHIN THE DEFINED MESSAGE AREA	
	Meaning: You entered a bytes offset that is outside the current length of the test message. The test message displays in the same format as the display command. The alter command exits.
	Action: Retry the command with the correct offset.

build

Function

Use the build command to add a test message to the C7TU message table. The routing label is identified by network type, destination point codes (DPC) and origination point codes (OPC), and the signaling link selector (SLS). Specific bytes in the message can be specified using the data parameter associated with a message code and message type. Or, the message format can be defined by binding it against a message type.

build (continued)

build command parameters and variables	
Command	**Parameters and variables**

build   *mssg*

```
         ┌ ansi  ┌ default settings                                         ┐ (1)
         │       └ dmem dcluster dnet omem ocluster onet ansisls            ┘ (2)
         │                                                                    (3)
         │ ccitt ┌ default settings                                         ┐ (4)
         │       │ ni  prio ┌ basic    dpc    opc                           │ (5)
         │       │          │ austria ┌    dregn ┐     ┌ oregn  ┐           │ (6)
         │       │          │ china   │ dz dexch │ dsig│ oz oexch│ osig ccittsls (7)
         │       │          └ intl    └    dnetwk┘     └ onetwk ┘           │ (8)
         │                                                                    (9)
         │                                                                    (10)
         │ jpn   ┌ default settings                                         ┐ (11)
         └ ttc   └ dmain dsub dunit omain osub ounit sls                    ┘ (12)
```

build
(continued)

```
 (1)  ┌                                              ┐
 (2)  │                                              │
 (3)  │                                              │
 (4)  │                                              │
 (5)  │                                              │
 (6)  │ msgtype ┌ data ┌ zero filled       ┐        │
 (7)  │         │      └ offset    hexbytes ┘        │
 (8)  │         └ parms  user_defined                │
 (9)  │                                              │
 (10) │                                              │
 (11) │                                              │
 (12) └                                              ┘  (end)
```

Parameters and variables	Description
default settings	Omitting this entry forces the system to default to settings for the priority, OPC, DPC, and SLS data. The default states are as follows: • The default OPC is datafilled in Table C7NETWRK. • The default DPC is datafilled in Table C7RTESET. • The default priority is 0. • The default SLS is 0.
zero filled	Omitting this entry forces the system to default to zero-filling all data bytes after the routing label (up to the truncating limit of 16 bytes for a match entry).

-continued-

build (continued)

build command parameters and variables (continued)	
Parameters and variables	**Description**
ansi	This parameter identifies the network type of the message. The network type is the first parameter in the routing label string.
ansisls	This variable specifies the SLS of the ANSI/routing test message. The valid entry range for ANSI messages is 0-31.
austria	This parameter identifies the format of the CCITT test message to build.
basic	This parameter identifies the DPC format of the CCITT test message to build.
ccitt	This parameter specifies the network type of the message. The network type is the first parameter in the routing label string.
ccittsls	This variable specifies the SLS of the CCITT test message. The valid entry range for CCITT messages is 0-15.
china	This parameter identifies the format of the CCITT test message to build.
data	This parameter changes specific message bytes. If the data parameter is used, you must enter the necessary hex bytes w.r.t. the message code. The hex bytes are defaulted to zero. The message code may be one or two bytes long and does not necessarily follow the routing label immediately.
dcluster	This variable specifies the DPC cluster number of the ANSI/routing message to build. The valid entry range is 0-255.
dexch	This variable specifies the DPC exchange, in china format, of the CCITT message to build. The valid entry range is 0-127.
dmain	This variable specifies the DPC main area number of the TTC or JPN message to build. The valid entry range is 0-31.
dmem	This variable specifies the DPC member number of the ANSI/routing message to build. The valid entry range is 0-255.
dnet	This variable specifies the DPC network number of the ANSI/routing message to build. The valid entry range is 0-255.
dnetwk	This variable specifies the DPC area network, in intl format, of the point code of the CCITT message to build. The valid entry range is 0-255.
dpc	This variable specifies the DPC of the CCITT message to build, in basic format. The valid entry range is 0-16383.
-continued-	

build (continued)

build command parameters and variables (continued)	
Parameters and variables	**Description**
dregn	This variable specifies the DPC region, in austria format, of the CCITT message to build. The valid entry range is 0-15.
dsig	This variable specifies the DPC signal point of the CCITT message to build. The valid entry range for austria format is 0-31. The valid entry range for china or intl format is 0-7.
dsub	This variable specifies the DPC subarea number of the TTC or JPN message to build. The valid entry range is 0-15.
dunit	This variable specifies the DPC area unit number of the TTC or JPN message to build. The valid entry range is 0-127.
dz	This variable specifies the DPC zone of the CCITT message to build. The valid entry range for intl format is 0-7. The valid entry range for austria format is 0-31. The valid entry range for china format is 0-15.
hexbytes	This variable string specifies the new hex bytes for the message body. The existing bytes are overridden in the message.The hex bytes are defaulted to zero.
intl	This parameter identifies the format of the CCITT test message to build.
jpn	This parameter identifies the network type of the message.
msgtype	This variable specifies the message type of the test message being altered. The valid entry value is a message code that represents a message type currently understood by C7TU. The message code is one or two bytes long and does not necessarily follow the routing label immediately. (Refer to the C7TU directory msgcode command documentation. This command is preset to allow you to list the CCS7 message codes that can be altered.)
mssg	This variable specifies the message number of the selected message. The valid entry range is 0-7.
ni	This variable specifies the network indicator of the message. The valid entries are as follows: • intl • intlsp • natl • natlsp
-continued-	

build (continued)

build command parameters and variables (continued)	
Parameters and variables	**Description**
ocluster	This variable specifies the OPC cluster number of the ANSI/routing message to build.The valid entry range is 0-255.
oexch	This variable specifies the OPC exchange, in china format, of the CCITT message to build. The valid entry range is 0-127. Entering 0 builds a test message for all exchanges.
offset	This variable specifies the starting offset of the CCS7 message bytes to be altered. The valid entry range is 0-256.
omain	This variable specifies the OPC main area number of the TTC or JPN message to build. The valid entry range is 0-31.
omem	This variable specifies the OPC member number of the ANSI/routing message to build.The valid entry range is 0-255.
onet	This variable specifies the OPC network number of the ANSI/routing message to build. The valid entry range is 0-255.
onetwk	This variable specifies the OPC area network, in intl format, of the CCITT message to build. The valid entry range is 0-255.
opc	This variable specifies the OPC of the CCITT message to build, in basic format. The valid entry range is 0-16383.
oregn	This variable specifies the OPC region, in austria format, of the CCITT message to build. The valid entry range is 0-15.
osig	This variable specifies the OPC signal point of the CCITT message to build. The valid entry range for austria format is 0-31. The valid entry range for china or intl format is 0-7.
osub	This variable specifies the OPC subarea number of the TTC or JPN message to build.The valid entry range is 0-15.
ounit	This variable specifies the OPC area unit number of the TTC or JPN message to build. The valid entry range is 0-127.
oz	This variable specifies the OPC zone of the CCITT message to build. The valid entry range for intl format is 0-7. The valid entry range for austria format is 0-31. The valid entry range for china format is 0-15.
-continued-	

build (continued)

build command parameters and variables (continued)	
Parameters and variables	**Description**
parms	This parameter changes the specified message using a user-defined message format bound against a message type.
prio	This variable specifies the CCS7 priority to build. The valid entry range is 0-3.
sls	This variable specifies the SLS of the TTC or JPN test message. The valid entry range for TTC messages is 0-15. The valid entry range for JPN messages is 0-31.
ttc	This parameter specifies the network type of the message. The network type is the first parameter in the routing label string.
user_defined	This variable binds user-defined formats. These procedures are associated with the specified network type (ANSI, CCITT, JPN, or TTC), the message code which represents the service indicators, and the message type.
End	

Qualification

> **CAUTION**
> **The system cannot distinguish CCS7 test messages and normal CCS7 messages.**
> The system cannot distinguish between CCS7 test messages and normal CCS7 messages once they are sent into the network.

The system cannot distinguish between CCS7 test messages and normal CCS7 messages once they are sent into the network.

build (continued)

Example

The following table provides an example of the build command.

Example of the build command	
Example	**Task, response, and explanation**
build 0 ansi natl 0 1 2 3 6 7 8 0 sltm parms 01 01 ↵ *where*	
0	specifies the test message number of the selected message
ansi	specifies the network type of the message
natl	specifies the type of routing label used in the CCS7 message
0	specifies the CCS7 priority to intercept
1	specifies the DPC member number of the ANSI test message
2	specifies the DPC cluster number of the ANSI test message
3	specifies the DPC network number of the ANSI test message
6	specifies the OPC member number of the ANSI test message
7	specifies the OPC cluster number of the ANSI test message
8	specifies the OPC network number of the ANSI test message
0	specifies the SLS of the ANSI test message
sltm	specifies the message type
01	specifies a user-defined parameter
01	specifies a user-defined parameter
	Task: Build a specified message.
	Response: MESSAGE 0 WAS BUILT SUCCESSFULLY
	Explanation: The specified message was successfully built.

Responses

The following table provides explanations of the responses to the build command.

Responses for the build command	
MAP output	**Meaning and action**
ERROR: CANNOT BUILD AN ECA MESSAGE	
	Meaning: You entered a recognizable code, but the utility cannot build a message for the specified code. The build command exits.
	Action: Retry the build command with a valid message code.
	-continued-

build (end)

Responses for the build command (continued)	
MAP output	**Meaning and action**
ERROR: INVALID MESSAGE CODE ZPF	
	Meaning: You entered a message code that is not recognized by C7TU. The build command exits.
	Action: Retry the build command with a valid message code.
MESSAGE 0 WAS BUILT SUCCESSFULLY	
	Meaning: The message was built by the C7TU and stored in the message table with message number
	Action: None
MESSAGE WAS NOT BUILT SUCCESSFULLY	
	Meaning: You entered an invalid message number. The build command exits.
	Action: Retry the build command with a valid message number.
MESSAGE 0 WAS NOT BUILT SUCCESSFULLY	
	Meaning: You entered errors in the message input. The message number is echoed in the error message. The build command exits.
	Action: Retry the build command with a valid message input.
Warning: Msg type has been overwritten	
	Meaning: The command executed and the message is added to the match table. The message code in the message body (entered after the routing label) has been overwritten with the entered hex bytes or default data parameter entry. The default entry zero-fills all the data bytes after the routing label, up to the 16-byte truncating limit for a match entry.
	Action: None
End	

display

Function

Use the display command to display the newly-built test messages.

display command parameters and variables		
Command	**Parameters and variables**	
display	all *msg_num*	⌈ verbose ⌉
Parameters and variables	**Description**	
all	This parameter displays all C7TU test messages that exist in the message table.	
msg_num	This variable specifies the test message number for the message to be displayed.	
verbose	This parameter displays the complete C7TU message including the internal header. This is not a required entry.	

Qualifications

None

Examples

The following table provides examples of the display command.

Examples of the display command
Example   **Task, response, and explanation**
display   all ↵

Task: Display all C7TU test messages.
 	
Response:
<pre>C7TU MESSAGE   SIO   DPC   OPC   SLS
  num type length ni pr si   mem clu net mem clu net
   0 SLTM   9   2 3   2   001 001 001 002 002 002   0
 	
C7TU MESSAGE   SIO   DPC   OPC   SLS
  num type length ni pr si   mem clu net mem clu net
   0 UDT   56   2 3   2   003 004 005 006 007 008   2</pre>
Explanation: The system displays all C7TU test messages in the message table.

-continued-

display (end)

Examples of the display command (continued)	
Example	**Task, response, and explanation**
display 0 ↵ *where* 0	displays a specified C7TU test message

<div>

Task: Display a specified test message.

Response:
```
C7TU MESSAGE         SIO            DPC           OPC        SLS
  num type length  ni pr  si   mem clu net  mem clu net
   0  SLTM   9       2  3   2   001 001 001  002 002 002   0
  Message bytes:
   0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
  -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
  00 00 09 00 00 02 01 B2 01 01 01 02 02 02 00 11 01 01
```

Explanation: The system displays the test message for test message number 0.

</div>

End

Responses

The following table provides explanations of the responses to the display command.

Responses for the display command	
MAP output	**Meaning and action**
`MESSAGE NUMBER <msg> HAS NOT BEEN BUILT YET`	
	Meaning: You specified a message number for a message that has not been built. The display command exits.
	Action: None
`THERE ARE NO C7TU MESSAGES BUILT`	
	Meaning: You attempted to display C7TU messages when none are in the message table. The display command exits.
	Action: None

dump

Function

Use the dump command to display the match table and examine the criteria used in monitor and intercept requests for C7TU messages.

dump command parameters and variables	
Command	**Parameters and variables**
dump	*start* *stop*

Parameters and variables	**Description**
start	This variable specifies an entry number defining the starting position for the display in the match table. The valid entry range is 0-7.
stop	This variable specifies an entry number defining where the display stops in the match table. The valid entry range is 0-7.

Qualifications

None

Example

The following table provides an example of the dump command.

dump (continued)

Example of the dump command

Example	Task, response, and explanation

---

dump 0 1 ↵
where

0	defines the starting position for the display in the match table
1	defines where the display stops in the match table

Task: Display a selected portion of the match table.

Response:
```
Number of valid match entries = 3
C7TU MON            SIO            DPC          OPC
NUM  DIR   NET    NI PR  SI MEM CLU NET  MEM CLU NET   SLS   TYPE
 0   BOTH ANSI    2 00   5   1   2   3    0   0   0     0    IAM
            0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
            -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
Match:     00 04 00 00 82 01 01 01 00 00 00 00 01
Mask:      00 FF 00 00 CF FF FF FF 00 00 00 00 1F

C7TU INT            SIO            DPC          OPC
NUM  DIR   NET    NI PR  SI MEM CLU NET  MEM CLU NET   SLS   TYPE
 1   BOTH ANSI    0  0   2   0   0   0    0   0   0     0    SLTM
            0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
            -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
Match:     00 04 00 00 02 00 00 00 00 00 00 00 11
Mask:      00 FF 00 00 0F 00 00 00 00 00 00 00 1F
```

Explanation: The system provides a display of the match table from 0 to 1.

Responses

The following table provides explanations of the responses to the dump command.

Responses for the dump command

MAP output	Meaning and action

ERROR: FIRST ITEM MUST NOT BE GREATER THAN LAST ITEM

Meaning: You attempted to display a range where the first item had a larger entry number in the match table than the last item. The dump command exits.

Action: Verify the start and stop numbers, and retry the command with a correct range.

-continued-

---

dump (end)

Responses for the dump command (continued)
MAP output   **Meaning and action**
There are no valid match entries in the specified range.
Meaning: There are no match entries in the specified range. The command halts execution and no messages display. **Action:**   None
End

help

Function

Use the help command to receive online documentation for the C7TULINK directory.

help command parameters and variables	
Command	**Parameters and variables**
help	c7tulink
Parameters and variables	**Description**
c7tulink	This parameter produces online documentation for the C7TULINK directory.

Qualifications

None

Examples

The following table provides examples of the help command.

Examples of the help command	
Example	**Task, response, and explanation**
help c7tulink ↵	
Task:	Access online documentation.

Response:	`**********C7TU LINK PMT7 ENVIRONMENT*********`
	`MASK      -set three MASK bytes of an entry`
	`MATCH     -set the MATCH bytes of an entry`
	`DUMP      -display MATCH table in hex format`
	`HELP      -generate this text`
	`MONitor   -monitor messages at the ST interface`
	`REMOVE    -cancel an intercept/monitor request or`
	`           build`
	`RESTORE   -send the MATCH table entries to MSB`
	`SELECT    -select PMs and attributes`
	`QUIT      -exit C7TULINK environment`
	`STATUS    -display the status of the C7TULINK`
	`           environment`
	`Enter "Q <command name>" for more information.`

Explanation:	You entered the C7TU directory, accessed the basic C7TULINK directory, and performed a help query.
-continued-	

help (continued)

Examples of the help command (continued)

Example	Task, response, and explanation

help c7tulink ↵

	Task:	Access online documentation.
	Response:	```**********C7TULINK ILPT7 ENVIRONMENT*********```

```
ALTER      -alter the bytes in the build message
BUILD      -build a CCS7 message to be sent
DISPLAY    -display the built message
DUMP       -display MATCH table in hex format
HELP       -generate this text
INTercept  -intercept messages at the ST interface
MASK       -set three MASK bytes of an entry
MATCH      -set the MATCH bytes of an entry
MONitor    -monitor messages at the ST interface
QUIT       -exit C7TULINK environment
REMOVE     -cancel an intercept/monitor request or
            build
RESTORE    -send the MATCH table entries to MSB
SELECT     -select PMs and attributes
SEND       -insert the message at ST interface
STATUS     -display the status of the C7TULINK
            environment
Enter "Q <command name>" for more information.
```

	Explanation:	You entered a valid password prior to entering the C7TU directory, accessed the password-protected C7TULINK directory, and performed a help query.

End

Response

The following table provides an explanation of the response to the help command.

help (end)

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

intercept

Function

Use the intercept command to intercept CCS7 messages coming off the link. The intercept command accesses the match table to remove the message from the link. The CCS7 system never sees the message.

intercept command parameters and variables	
Command	**Parameters and variables**

intercept	*l s d* ansi *dmem dcluster dnet omem ocluster onet ansisls* (1) (2) (3) ccitt *ni prio* basic *dpc opc* (4) austria *dregn* *oregn* *ccsls* (5) china *dz* *dexch* *dsig* *oz* *oexch* *osig* (6) intl *dnetwk* *onetwk* (7) (8) (9) jpn *dmain dsub dunit omain osub ounit sls* (10) ttc (11)
intercept (continued)	(1) (2) (3) (4) (5) (6) *msgtype* data *zero_filled* (7) *offset hexbytes* (8) *parms* (9) (10) (11) (end)

Parameters and variables	**Description**
zero filled	Omitting this entry forces the system to default to zero-filling all data bytes after the routing label (up to the truncating limit of 16 bytes for a match entry).
ansi	This parameter specifies the network type of the message to be intercepted. The network type is the first parameter in the routing label string.
	-continued-

intercept (continued)

intercept command parameters and variables (continued)	
Parameters and variables	**Description**
ansisls	This variable specifies the SLS of the ANSI/routing test message. The valid entry range for ANSI messages is 0-32. Entering 32 intercepts all messages.
austria	This parameter identifies the format of the CCITT test message to intercept.
basic	This parameter identifies the destination point code (DPC) format of the CCITT test message to intercept.
ccitt	This parameter identifies the network type of the message to be intercepted. The network type is the first parameter in the routing label string.
ccittsls	This variable specifies the signalling link selector (SLS) of the CCITT test message. The valid entry range for CCITT messages is 0-16. Entering 16 intercepts all messages.
china	This parameter identifies the format of the CCITT test message to intercept.
d	This variable specifies the direction of the message to be intercepted. The valid entry values include the following: ▪ in ▪ out ▪ both
data	This parameter changes specific message bytes. If the data parameter is used, you must enter the necessary hex bytes w.r.t. the message code. The hex bytes are defaulted to zero. The message code may be one or two bytes long and does not necessarily follow the routing label immediately.
dcluster	This variable specifies the DPC cluster number of the ANSI message to intercept. The valid entry range is 0-255. Entering 0 intercepts all clusters.
dexch	This variable specifies the DPC exchange, in china format, of the CCITT message to intercept. The valid entry range is 0-127. Entering 0 intercepts all exchanges.
dmain	This variable specifies the DPC main area number of the TTC or JPN message to intercept. The valid entry range is 0-31. Entering 0 intercepts all main area numbers.
dmem	This variable specifies the DPC member number of the ANSI/routing message to intercept. The valid entry range is 0-255. Entering 0 intercepts all members.
-continued-	

intercept (continued)

intercept command parameters and variables (continued)	
Parameters and variables	**Description**
dnet	This variable specifies the DPC network number of the ANSI/routing message to intercept. The valid entry range is 0-255. Entering 0 intercepts all networks.
dnetwk	This variable specifies the DPC area network, in intl format, of the CCITT message to intercept. The valid entry range is 0-255. Entering 0 intercepts all area networks.
dpc	This variable specifies the DPC of the CCITT message to intercept, in basic format. The valid entry range is 0-16383. Entering 0 sets the dpc variable to a value of all.
dregn	This variable specifies the DPC region, in austria format, of the CCITT message to intercept. The valid entry range is 0-15. Entering 0 intercepts all exchanges.
dsig	This variable specifies the DPC signal point of the CCITT message to intercept. The valid entry range for austria format is 0-31. The valid entry range for china or intl format is 0-7. Entering 0 intercepts all signal points.
dsub	This variable specifies the DPC subarea number of the TTC or JPN message to intercept. The valid entry range is 0-15. Entering 0 intercepts all subareas.
dunit	This variable specifies the DPC area unit number of the TTC or JPN message to intercept. The valid entry range is 0-127. Entering 0 intercepts all area units.
dz	This variable specifies the DPC zone of the CCITT message to intercept. The valid entry range for intl format is 0-7. The valid entry range for austria format is 0-31. The valid entry range for china format is 0-15. Entering 0 intercepts all zones.
hexbytes	This variable string specifies the new hex bytes of the message body. The existing bytes are overridden in the message. Any extra bytes beyond the 16-byte match table limit will be discarded. The hex bytes are defaulted to zero.
intl	This parameter identifies the format of the CCITT test message to intercept.
jpn	This parameter identifies the network type of the message.
l	This variable specifies the name of the linkset to be intercepted. The valid entry is a string.
-continued-	

intercept (continued)

intercept command parameters and variables (continued)	
Parameters and variables	**Description**
msgtype	This variable specifies the message type of the test message being altered. The valid entry value is a message code that represents a message type currently understood by C7TU. The message code is one or two bytes long and does not necessarily follow the routing label immediately. (Refer to the C7TU directory msgcode command documentation. This command is preset to allow you to list the CCS7 message codes that can be altered.)
ni	This variable specifies the network indicator of the message. The valid entries are as follows: • intl • intlsp • natl • natlsp • all
ocluster	This variable specifies the origination point code (OPC) cluster number of the ANSI/routing message to intercept.The valid entry range is 0-255. Entering 0 intercepts all clusters.
oexch	This variable specifies the OPC exchange, in china format, of the CCITT message to intercept. The valid entry range is 0-127. Entering 0 intercepts all exchanges.
offset	This variable specifies the starting offset of the CCS7 message bytes to be altered. The valid entry range is 0-256.
omain	This variable specifies the OPC main area number of the TTC or JPN message to intercept. The valid entry range is 0-31. Entering 0 intercepts all main areas.
omem	This variable specifies the OPC member number of the ANSI/routing message to intercept.The valid entry range is 0-255. Entering 0 intercepts all members.
onet	This variable specifies the OPC network number of the ANSI/routing message to intercept. The valid entry range is 0-255. Entering 0 intercepts all networks.
onetwk	This variable specifies the OPC area network, in intl format, of the CCITT message to intercept. The valid entry range is 0-255. Entering 0 intercepts all area networks.
opc	This variable specifies the OPC of the CCITT message to intercept, in basic format. The valid entry range is 0-16383.
oregn	This variable specifies the OPC region, in austria format, of the CCITT message to intercept. The valid entry range is 0-15. Entering 0 intercepts all regions.
-continued-	

intercept (continued)

intercept command parameters and variables (continued)	
Parameters and variables	**Description**
osig	This variable specifies the OPC signal point of the CCITT message to intercept. The valid entry range for austria format is 0-31. The valid entry range for china or intl format is 0-7. Entering 0 intercepts all signal points.
osub	This variable specifies the OPC subarea number of the TTC or JPN message to intercept.The valid entry range is 0-15. Entering 0 intercepts all subareas.
ounit	This variable specifies the OPC area unit number of the TTC or JPN message to intercept. The valid entry range is 0-127. Entering 0 intercepts all area units.
oz	This variable specifies the OPC zone of the CCITT message to intercept. The valid entry range for intl format is 0-7. The valid entry range for austria format is 0-31. The valid entry range for china format is 0-15. Entering 0 intercepts all zones.
parms	This variable allows parameters for specified message types only. For example, most significantly, the CIC for ISUP messages.
prio	This variable specifies the CCS7 priority to intercept. The valid entry range is 0-4. Entering 4 sets a priority of all.
s	This variable specifies the link number of the linkset to be intercepted. The valid entry range is 0-15.
sls	This variable specifies the SLS of the TTC or JPN test message. The valid entry range for TTC messages is 0-16. (Entering 16 intercepts all messages.) The valid entry range for JPN messages is 0-31. (Entering 32 intercepts all messages.)
ttc	This parameter specifies the network type of the message to be intercepted. The network type is the first parameter in the routing label string.
End	

intercept (continued)

Qualification

⚠ **CAUTION**

Removing a CCS7 message may have consequences for the
node and the network.

Removing a CCS7 message may have consequences for the node and the
network.

Example

The following table provides an example of the intercept command.

Example of the intercept command

Example	Task, response, and explanation
intercept *where*	**c7lkset2 1 both ansi all natl 3** ↵
c7lkset2 1 3	specifies the name of the linkset to be intercepted specifies the link number of the linkset to be intercepted specifies the priority of the linkset to be intercepted

	Task:	Intercept CCS7 messages coming off the link.
	Response:	Currently not available
	Explanation:	This command intercepts CCS7 messages coming from and going to linkset 1 (C7LKSET2).

Responses

The following table provides explanations of the responses to the intercept
command.

intercept (continued)

Responses for the intercept command	
MAP output	**Meaning and action**
`ALL IS NOT PERMITTED IN A FIELD ENVIRONMENT`	
	Meaning: You entered all as a linkset name. Since only four entries are allowed in the match table, the intercept command exits.
	Action: Retry the command with a linkset name in place of the all parameter, or release a link before entering the intercept command string.
`ERROR: INVALID LINK NUMBER`	
	Meaning: You specified a link number that is not datafilled for the specified linkset in the C7LINK table. The intercept command exits.
	Action: Verify the link number and retry the intercept command with the correct number.
`ERROR: INVALID LINKSET NAME`	
	Meaning: You specified a linkset name that does not appear in Table C7LKSET.
	Action: Verify the linkset name and retry the intercept command with the correct linkset.
`ERROR: INVALID MSGCODE ZPF`	
	Meaning: You entered a message code that is not recognized by C7TU. The intercept command exits.
	Action: Verify the message code and retry the command.
`ERROR: MATCH TABLE FULL`	
	Meaning: You attempted to intercept a message when the match table already had eight entries. No further requests can be made. The intercept command exits.
	Action: Remove an existing entry from the match table and retry the command.
`ERROR: MATCHING ECM MESSAGES IS NOT PERMITTED`	
	Meaning: You entered a message code that was recognized by C7TU, but a match is not allowed for the specified code. The intercept command exits.
	Action: Verify the message code, and retry the intercept command with a correct message code.
-continued-	

intercept (end)

Responses for the intercept command (continued)	
MAP output	**Meaning and action**
ONLY FOUR MONITORS ALLOWED IN FIELD ENVIRONMENT	
	Meaning: You attempted to intercept when four entries were already in the match table. The field environment allows only four entries in the match table. The intercept command exits.
	Action: Remove an existing intercept or monitor request and retry the intercept command.
SLS IS UPDATED WITH THE LEAST SIGNIFICANT 4 BITS OF THE CIC	
	Meaning: This command executes. The message appears when Telephone User Part (TUP), TUPP, or BTUP messages are involved. In these message types, the SLS value is derived from the least significant four bits of the CIC value. The SLS value entered in the routing label is overwritten.
	Action: None
USER ENTERED DATA LONGER THAN MATCH ENTRY SIZE TRUNCATING USER DATA.	
	Meaning: The match entry has been truncated to contain bytes 3-18 of the message unit, inclusively. Each match entry is 16 bytes long.
	Action: None
WARNING: C7TU IS NOT ENABLED IN ANY PMs	
	Meaning: The command is executed, and the request is added to the match table.
	Action: Use the select command to enable the C7TU in the peripheral modules that are to be used.
WARNING: C7TU IS NOT ENABLED ON pm num WHERE THIS LINK RESIDES	
	Meaning: The intercept command executed, and the request is added to the match table.
	Action: Use the select command to enable the C7TU in the peripheral module that is to be used.
End	

mask

Function

Use the mask command to mask bytes in a monitor or intercept entry. The masked bytes are not used to compare for matching messages. This command is used to customize a monitor or intercept entry. Previously, the entire 16 bytes of a mask entry had to be specified in order to change one byte. This command allows you to specify the starting byte of the change.

mask command parameters and variables		
Command	**Parameters and variables**	
mask	*item_no* *byte_offset* *mask_bytes*	
Parameters and variables	**Description**	
byte_offset	This variable specifies the number of bytes used to describe the type of CCS7 messages to monitor or intercept. In a monitor or intercept entry, the valid entry range is 0-15.	
item_no	This variable specifies the number of the monitor or intercept entry. The valid entry range is 0-7.	
mask_bytes	This variable specifies how the bytes, starting at the byte_offset, should be masked.	

Qualifications

None

Example

The following table provides an example of the mask command.

mask (continued)

Example of the mask command

Example	Task, response, and explanation

---

mask 0 01 aa ↵
where
0   specifies the number of the monitor or intercept entry
01   specifies the number of bytes to describe the type of CCS7 messages to monitor or intercept
aa   describes how the bytes, starting at the byte_offset, should be masked

---

Task: Change a mask offset value.

Response:
```
C7TU  MON       SIO                  DPC         OPC     SLS TYPE
NUM   DIR   NET NI  PR  SI  MEM CLU NET MEM CLU NET
 0    BOTH ANSI ALL XX  SNTS XXX XXX XXX XXX XXX XXX   XXX SLTM
           0   1 2 3 4 5   6 7 8    9 10 11 12 13 14 15
           -   - - - - -   - - -    - --  -- - -- -- --
MATCH:     00 05 00 00 02 00 00 00 00 00 05 11 00 00 00
MASK:      00 FF 00 00 0F 00 00 00 00 00 00 FF 00 00 00


C7TU  MON       SIO                  DPC         OPC     SLS TYPE
NUM   DIR   NET NI  PR  SI  MEM CLU NET MEM CLU NET
 0    BOTH ANSI ALL XX  SNTS XXX XXX XXX XXX XXX XXX   XXX SLTM
           0   1 2 3 4 5   6 7 8    9 10 11 12 13 14 15
           -   - - - - -   - - -    - --  -- - -- -- --
MATCH:     00 05 00 00 02 00 00 00 00 00 05 11 00 00 00
MASK:      00 AA 00 00 0F 00 00 00 00 00 00 FF 00 00 00
```

Explanation: The mask offset value is changed.

Responses

The following table provides explanations of the responses to the mask command.

Responses for the mask command

MAP output	Meaning and action

ERROR: MASK ENTRY 1 IS NOT IN USE

        Meaning: Mask entry 1 has not been assigned.

        Action: Assign the entry before attempting to alter or use it.

                                  -continued-

---

mask (end)

Responses for the mask command (continued)	
MAP output	**Meaning and action**
`Out of range: <BYTE OFFSET> (0 TO 15)` `Enter:  <BYTE OFFSET> [<MASK BYTES>] . . .`	
	Meaning: The byte offset value was entered incorrectly. **Action:** Reenter the command.
End	

match

Function

Use the match command to specify bytes to match on in a monitor or intercept entry. (The match bytes are used to compare for matching messages.) This command is used to customize a monitor or intercept entry. Previously, the entire 16 bytes of a match entry had to be specified in order to change one byte. This command allows you to specify the starting byte of the change.

match command parameters and variables		
Command	**Parameters and variables**	
match	*item_no* *byte_offset* *mask_bytes*	
Parameters and variables	**Description**	
byte_offset	This variable specifies the number of bytes used to describe the type of CCS7 messages to monitor or intercept. In a monitor or intercept, the valid entry range is 0-15.	
item_no	This variable specifies the number of the monitor or intercept entry. The valid entry range is 0-7.	
mask_bytes	This variable is a string that describes the bytes to match messages against.	

Qualifications

None

Example

The following table provides an example of the match command.

match (continued)

Example of the match command

Example	Task, response, and explanation

---

match 0 01 06 ↵
where

0	specifies the number of the monitor or intercept entry
01	specifies the number of bytes to describe the type of CCS7 messages to monitor or intercept
06	describes how the bytes, starting at the byte_offset, should be matched.

Task: Change a match offset value.

Response:
```
C7TU  MON       SIO                   DPC           OPC      SLS TYPE
NUM   DIR    NET NI   PR  SI   MEM CLU NET MEM CLU NET
 0    BOTH  ANSI ALL  XX   SNTS XXX XXX XXX XXX XXX XXX   XXX SLTM
            0    1  2  3  4  5    6  7  8    9  10 11  12 13 14 15
            -    -  -  -  -  -    -  -  -    -  --  --  -  -- -- --
MATCH:      00 05 00 00 02 00 00 00 00 00 05 11 00 00 00
MASK:       00 FF 00 00 0F 00 00 00 00 00 00 FF 00 00 00


C7TU  MON       SIO                   DPC           OPC      SLS TYPE
NUM   DIR    NET NI   PR  SI   MEM CLU NET MEM CLU NET
 0    BOTH  ANSI ALL  XX   SNTS XXX XXX XXX XXX XXX XXX   XXX SLTM
            0    1  2  3  4  5    6  7  8    9  10 11  12 13 14 15
            -    -  -  -  -  -    -  -  -    -  --  --  -  -- -- --
MATCH:      00 06 00 00 02 00 00 00 00 00 05 11 00 00 00
MASK:       00 AA 00 00 0F 00 00 00 00 00 00 FF 00 00 00
```

Explanation: The match offset value is changed.

Responses

The following table provides explanations of the responses to the match command.

match (end)

Responses for the match command	
MAP output	**Meaning and action**
`ERROR: MATCH ENTRY 1 IS NOT IN USE`	
	Meaning: Match entry 1 has not been assigned.
	Action: Assign the entry before attempting to alter or use it.
`Out of range: <BYTE OFFSET> (0 TO 15)` `Enter:   <BYTE OFFSET> [<MATCH BYTES>] . . .`	
	Meaning: The byte offset value was entered incorrectly.
	Action: Retry the command.

monitor

Function

Use the monitor command to monitor a test message in the C7TU message table. The routing label identifies a network type, network indicator, priority, destination point code (DPC), origination point code (OPC), and signaling link selector (SLS). Specific bytes in the message can be identified by the data parameter associated with a message code and message type, or parameters can be specified for selected message types such as the CIC for ISUP messages.

monitor (continued)

monitor command parameters and variables	
Command	**Parameters and variables**

monitor *l s d* | ansi *dmem dcluster dnet omem ocluster onet ansisls* (1)
(2)
ccitt [*ni prio* basic *dpc opc* (3) (4)
austria [*dregn* *oregn* *ccsls* (5)
china *dz* dexch *dsig* *oz* oexch *osig* (6)
intl *dnetwk* *onetwk* (7)
(8)
(9)
jpn [*dmain dsub dunit omain osub ounit sls*] (10)
ttc (11)

monitor (1)
(continued) (2)
(3)
(4)
(5)
(6) *msgtype* data *zero_filled*
(7) *offset hexbytes*
(8) *parms*
(9)
(10)
(11) (end)

Parameters and variables	Description
default settings	Omitting this entry forces the system to default to settings for the OPC, DPC, priority, and SLS data. The default states are as follows: • The default OPC is datafilled in Table C7NETWRK. • The default DPC is datafilled in Table C7RTESET. • The default priority is 0. • The default SLS is 0.
zero filled	Omitting this entry forces the system to default to zero-filling all data bytes after the routing label (up to the truncating limit of 16 bytes for a match entry).
ansi	This parameter specifies the network type of the message to be monitored. The network type is the first parameter in the routing label string.

-continued-

monitor (continued)

monitor command parameters and variables (continued)	
Parameters and variables	**Description**
ansisls	This variable specifies the SLS of the ANSI/routing test message. The valid entry range for ANSI/routing messages is 0-32. Entering 32 selects all messages.
austria	This parameter identifies the format of the CCITT test message to monitor.
basic	This parameter identifies the DPC format of the CCITT test message to monitor.
ccitt	This parameter identifies the network type of the message to be monitored. The network type is the first parameter in the routing label string.
ccsls	This variable specifies the SLS of the CCITT test message. The valid entry range for CCITT messages is 0-16. Entering 16 selects all messages.
china	This parameter identifies the format of the CCITT test message to monitor.
d	This variable specifies the direction of the message to be monitored. The valid entry values include the following: ▪ in ▪ out ▪ both
data	This parameter changes specific message bytes. If the data parameter is used, you must enter the necessary hex bytes w.r.t. the message code. The hex bytes are defaulted to zero. The message code may be one or two bytes long and does not necessarily follow the routing label immediately.
dcluster	This variable specifies the DPC cluster number of the ANSI/routing message to monitor. The valid entry range is 0-255. Entering 0 monitors all clusters.
dexch	This variable specifies the DPC exchange, in china format, of the CCITT message to monitor. The valid entry range is 0-127. Entering 0 monitors all exchanges.
dmain	This variable specifies the DPC main area number of the TTC or JPN message to monitor. The valid entry range is 0-31. Entering 0 monitors all main area numbers.
dmem	This variable specifies the DPC member number of the ANSI/routing message to monitor. The valid entry range is 0-255. Entering 0 monitors all members.
dnet	This variable specifies the DPC network number of the ANSI/routing message to monitor. The valid entry range is 0-255. Entering 0 monitors all networks.
-continued-	

monitor (continued)

monitor command parameters and variables (continued)	
Parameters and variables	**Description**
dnetwk	This variable specifies the DPC area network, in intl format, of the CCITT message to monitor. The valid entry range is 0-255. Entering 0 monitors all area networks.
dpc	This variable specifies the DPC of the CCITT message to monitor, in basic format. The valid entry range is 0-16383. Entering 0 sets the dpc variable to a value of all.
dregn	This variable specifies the DPC region, in austria format, of the CCITT message to monitor. The valid entry range is 0-15. Entering 0 monitors all regions.
dsig	This variable specifies the DPC signal point of the CCITT message to monitor. The valid entry range for austria format is 0-31. The valid entry range for china or intl format is 0-7. Entering 0 monitors all signal points.
dsub	This variable specifies the DPC subarea number of the TTC or JPN message to monitor. The valid entry range is 0-15. Entering 0 monitors all subareas.
dunit	This variable specifies the DPC area unit number of the TTC or JPN message to monitor. The valid entry range is 0-127. Entering 0 monitors all area units.
dz	This variable specifies the DPC zone of the CCITT message to monitor. The valid entry range for intl format is 0-7. The valid entry range for austria format is 0-31. The valid entry range for china format is 0-15. Entering 0 monitors all zones.
hexbytes	This variable string specifies the new hex bytes in the message body. The existing bytes are overridden in the message.The hex bytes are defaulted to zero.
intl	This parameter identifies the format of the CCITT test message to monitor.
jpn	This parameter identifies the network type of the message.
l	This variable specifies the name of the linkset to be monitored. The valid entry is a string.
msgtype	This variable specifies the message type of the test message being altered. The valid entry value is a message code that represents a message type currently understood by C7TU. The message code is one or two bytes long and does not necessarily follow the routing label immediately. (Refer to the C7TU directory msgcode command documentation. This command is preset to allow you to list the CCS7 message codes that can be altered.)
-continued-	

monitor (continued)

monitor command parameters and variables (continued)	
Parameters and variables	**Description**
ni	This variable specifies the network indicator of the message. The valid entries are as follows: • intl • intlsp • natl • natlsp • all
ocluster	This variable specifies the OPC cluster number of the ANSI/routing message to monitor. The valid entry range is 0-255. Entering 0 monitors all clusters.
oexch	This variable specifies the OPC exchange, in china format, of the CCITT message to monitor. The valid entry range is 0-127. Entering 0 monitors all exchanges.
offset	This variable specifies the starting offset of the CCS7 message bytes to be altered. The valid entry range is 0-256.
omain	This variable specifies the OPC main area number of the TTC or JPN message to monitor. The valid entry range is 0-31. Entering 0 monitors all main areas.
omem	This variable specifies the OPC member number of the ANSI/routing message to monitor. The valid entry range is 0-255. Entering 0 monitors all members.
onet	This variable specifies the OPC network number of the ANSI/routing message to monitor. The valid entry range is 0-255. Entering 0 monitors all networks.
onetwk	This variable specifies the OPC area network, in intl format, of the CCITT message to monitor. The valid entry range is 0-255. Entering 0 monitors all area networks.
opc	This variable specifies the OPC, in basic format, of the CCITT message to monitor. The valid entry range is 0-16383.
oregn	This variable specifies the OPC region, in austria format, of the CCITT message to monitor. The valid entry range is 0-15. Entering 0 monitors all regions.
osig	This variable specifies the OPC signal point of the CCITT message to monitor. The valid entry range for austria format is 0-31. The valid entry range for china or intl format is 0-7. Entering 0 monitors all signal points.
osub	This variable specifies the OPC subarea number of the TTC or JPN message to monitor. The valid entry range is 0-15. Entering 0 monitors all subareas.
-continued-	

monitor (continued)

monitor command parameters and variables (continued)	
Parameters and variables	**Description**
ounit	This variable specifies the OPC area unit number of the TTC or JPN message to monitor. The valid entry range is 0-127. Entering 0 monitors all area units.
oz	This variable specifies the OPC zone of the CCITT message to monitor. The valid entry range for intl format is 0-7. The valid entry range for austria format is 0-31. The valid entry range for china format is 0-15. Entering 0 monitors all zones.
parms	This variable allows parameters for specified message types only. For example, most significantly, the CIC for ISUP messages.
prio	This variable specifies the CCS7 priority to monitor. The valid entry range is 0-4. Entering 4 sets a priority of all.
s	This variable specifies the link number of the linkset to be monitored. The valid entry range is 0-15.
sls	This variable specifies the SLS of the TTC or JPN test message. The valid entry range for TTC messages is 0-16. (Entering 16 selects all TTC messages.) The valid entry range for JPN messages is 0-31. Entering 32 selects all JPN messages.
ttc	This parameter specifies the network type of the message to be monitored. The network type is the first parameter in the routing label string.
End	

Qualifications

The monitor command is qualified by the following exceptions, restrictions, and limitations:

- If the message code used used in a monitor command string is not in the list of valid message codes, the monitor command will not perform. (If you need to review a list of current, valid message codes, use the C7TU directory msgcode command.)

- Match entries are matched from the first entry in the match table. The process stops when there is a valid match entry or when the search does not produce a match. After the first suitable match, the rest of the entries are not evaluated for a match. You will need to ensure that your match entries are not screened by another user's entry.

- Before you use the monitor command, use the select command to enable message monitoring on the appropriate peripheral modules (PM).

monitor (continued)

Example

The following table provides an example of the monitor command.

Example of the monitor command	
Example	**Task, response, and explanation**
monitor c7lkset2 1 in ansi all intl 2 ↵ *where* c7lkset2 specifies the name of the linkset to be monitored 1 specifies the link number of the linkset to be monitored 2 specifies the priority of the linkset to be monitored	
	Task: Monitor CCS7 messages on a specified linkset. **Response:** Not currently available **Explanation:** This command monitors linkset 1 (C7LKSET2).

Responses

The following table provides explanations of the responses to the monitor command.

Responses for the monitor command	
MAP output	**Meaning and action**
`ALL IS NOT PERMITTED IN  FIELD ENVIRONMENT`	
	Meaning: The field environment allows only four entries in the match table; therefore, the all parameter is not valid. The monitor command exits. **Action:** None
`ERROR: INVALID LINK NUMBER`	
	Meaning: You specified a link number that is not datafilled for the specified linkset in the C7LINK table. The monitor command exits. **Action:** Verify the link number and retry the monitor command with the correct number.
-continued-	

monitor (continued)

Responses for the monitor command (continued)	
MAP output	**Meaning and action**
ERROR: INVALID LINKSET NAME	
	Meaning: You specified a linkset name that does not appear in Table C7LKSET.
	Action: Verify the linkset name, and retry the monitor command with the correct linkset.
ERROR: INVALID MSGCODE ZPF	
	Meaning: You entered a message code that is not recognized by C7TU. The monitor command exits.
	Action: Verify the message code and retry the command.
ERROR: MATCH TABLE FULL	
	Meaning: You attempted to monitor a message when the match table already had eight entries. No further requests can be made. The monitor command exits.
	Action: Remove an existing entry from the match table and retry the command.
ERROR: MATCHING ECM MESSAGES IS NOT PERMITTED	
	Meaning: You entered a message code that was recognized by C7TU, but a match is not allowed for the specified code. The monitor command exits.
	Action: Verify the message code and retry the monitor command with a correct message code.
ONLY FOUR MONITORS ALLOWED IN FIELD ENVIRONMENT	
	Meaning: You attempted to monitor when four entries were already in the match table. The field environment allows only four entries in the match table. The monitor command exits.
	Action: Remove an existing monitor, or monitor request, and retry the monitor command.
-continued-	

monitor (end)

Responses for the monitor command (continued)	
MAP output	**Meaning and action**
SLS IS UPDATED WITH THE LEAST SIGNIFICANT 4 BITS OF THE CIC	
	Meaning: This command executes. The message appears when (Telephone User Part) TUP, TUPP, or BTUP messages are involved. In these message types, the SLS value is derived from the least significant four bits of the CIC value. The SLS value entered in the routing label is overwritten.
	Action: None
USER ENTERED DATA LONGER THAN MATCH ENTRY SIZE TRUNCATING USER DATA.	
	Meaning: The match entry has been truncated to contain bytes 3-18 of the message unit, inclusively. Each match entry is 16 bytes long.
	Action: None
WARNING: C7TU IS NOT ENABLED IN ANY PMS	
	Meaning: The command is executed, and the request is added to the match table.
	Action: Use the select command to enable the C7TU in the peripheral modules that are to be used.
WARNING: C7TU IS NOT ENABLED ON <pm num> WHERE THIS LINK RESIDES	
	Meaning: The monitor command executed, and the request is added to the match table.
	Action: Use the select command to enable the C7TU in the peripheral module that is to be used.
WARNING: MSG TYPE HAS BEEN OVERWRITTEN	
	Meaning: The command executed. The message body that comes after the routing label has been overwritten with the entries or default for the data parameter. The default entry zero-fills all data bytes after the routing label (up to the truncating limit of 16 bytes for a match entry).
	Action: None
End	

q

Function

Use the q command to receive online documentation for the C7TULINK directory.

q command parameters and variables	
Command	**Parameters and variables**
q	*command_nam*

Parameters and variables	**Description**
command_nam	This variable specifies a valid C7TULINK directory command. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the q command.

Example of the q command	
Example	**Task, response, and explanation**
q dump ↵ *where*	
dump	specifies a valid C7TULINK directory command

	Task:	Access online documentation.
	Response:	DUMP MATCH TABLE IN HEX FORMAT UP TO THE ENTRY NUMBER SPECIFIED PARMS: <FIRST ITEM> {0 TO 7} [<LAST ITEM> {0 TO 7}]
	Explanation:	This example typifies a response for the help command string.

Response

The following table provides an explanation of the response to the q command.

q (end)

Response for the q command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

quit

Function

Use the quit command to exit the C7TULINK environment and return to the C7TU level. You can clear the C7TULINK environment before exiting.

quit command	
Command	**Parameters and variables**
quit	clear noclear

Parameters and variables	**Description**
clear	This parameter clears the C7TULINK environment before quitting and the system returns you to the CI level.
noclear	This parameter leaves the C7TULINK environment intact and the system returns you to the CI level.

Qualifications

None

Example

The following table provides an example of the quit command.

Example of the quit command	
Example	**Task, response, and explanation**
quit clear ↵	
	Task: Clear the C7TULINK environment before quitting.
	Response: C7TU:
	Explanation: This command exits the C7TULINK directory and returns you to the C7TU MAP level. Entering the quit command again returns you to the CI MAP level.

Response

The following table provides an explanation of the response to the quit command.

quit (end)

Response for the quit command	
MAP output	**Meaning and action**
`C7TU:`	
	Meaning: You have returned to the C7TU MAP level.
	Action: Enter the quit command again to return to the CI MAP level.

remove

Function

Use the remove command to remove the following:

- a monitor or intercept request from the match table
- a link that was previously selected with the select command
- a message that was entered by the build command from the message table

remove command parameters and variables		
Command	**Parameters and variables**	
remove	match message	all *number*

Parameters and variables	Description
all	This parameter specifies that all entries be removed.
match	This parameter removes an entry from the match table.
message	This parameter removes an entry from the message table.
number	This variable is the number of the entry to be removed. The valid entry value is 0-7.

Qualifications

None

Example

The following table provides an example of the remove command.

Example of the remove command	
Example	**Task, response, and explanation**
remove match all ↵	
	Task: Remove all entries from the match table.
	Response: Currently not available
	Explanation: This command removes all entries from the match table.

remove (end)

Responses

The following table provides explanations of the responses to the remove command.

Responses for the remove command	
MAP output	**Meaning and action**
`Item number <num> is currently not defined in the match table`	
	Meaning: You entered a match entry number not yet built using the monitor or intercept commands. The command does not execute.
	Action: Validate the match entry number and reissue the command.
`Message number <num> has not been built yet`	
	Meaning: You entered a message number not yet built using the build command. The command does not execute.
	Action: Validate the message number and reissue the command.
`Must specify Entry Number or ALL for REMOVE`	
	Meaning: You entered an invalid value. The command does not execute.
	Action: Reenter the command with a valid entry number or all.

restore

Function

Use the restore command to restore the match table to the MSB7s and LIU7s following a restart in the central controller (CC).

restore command parameters and variables	
Command	**Parameters and variables**
restore	There are no parameters or variables.

Qualifications

None

Example

Currently not available

Responses

Currently not available

select

Function

Use the select command to select an LIU7 or MSB7 in order to monitor messages. If the peripheral is not selected, CCS7 messages will not be matched against the match table for that link. This command also can be used to release a link that currently is selected.

Varying the log throttle is a password-protected function. If allowed, the select command can set the log throttle to between one and 60 logs for each LIU7. This log throttle can be changed when an LIU7 is selected or deselected.

---

select command parameters and variables

Command	Parameters and variables

| **select** | all

liu7   all
          l_num   *log_throt_set*   <u>*on*</u>
                                          off

msb7   all
         m_num |

Parameters and variables	Description
<u>*on*</u>	Omitting this entry forces the system to default to selecting the specified link.
all	When used in the first position, this parameter selects all datafilled MSB7s and LIU7s as the link specification. When used in the second position, this parameter selects all datafilled MSB7s or all datafilled LIU7s as the link specification.
liu7	This parameter selects one or all LIU7s as the link specification.
l_num	This variable is the LIU7 number. The valid entry range is 0-511.
log_throt_set	This variable is used to vary the C7TU log throttle in the selected LIU7. The valid entry range is 1-60. This entry is password-protected.
msb7	This parameter selects one or all MSB7s as the link specification.
m_num	This variable is the MSB7 number. The valid entry range is 0-9.
off	This variable releases the specified link.

---

select (continued)

Qualification

Only four entries are allowed in the match table at a time. If an attempt is made to use the select command when four entries already are in the match table, the command does not execute. You have to release a link that currently is selected in order to execute this command.

Examples

The following table provides examples of the select command.

Examples of the select command	
Example	**Task, response, and explanation**
select msb7 0 ↵ *where* 0 specifies the MSB7 number	
	Task: Select a specified MSB7 link.
	Response: `Select done`
	Explanation: The MSB7 link 0 has been selected.
select msb7 1 off ↵ *where* 1 specifies the MSB7 number	
	Task: Release a specified MSB7 link.
	Response: `Release done`
	Explanation: The MSB7 link 1 has been released.

Responses

The following table provides explanations of the responses to the select command.

select (continued)

Responses for the select command	
MAP output	**Meaning and action**
ERROR: WRONG PM SELECTION	
	Meaning: You entered a value that was not among the valid choices for PM selection. The select command exits.
	Action: Retry the command with a valid parameter.
LIU7 2 IS NOT DEFINED FOR THIS OFFICE	
	Meaning: You attempted to select an LIU7 that is not datafilled for this office. The select command exits.
	Action: Retry the select command specifying an LIU7 that is datafilled for this office.
LIU7 2 IS NOT INSERVICE THE LIU7 HAS BEEN DESELECTED	
	Meaning: You released an LIU7 number that currently is not in service. Monitoring does not start when the LIU7 comes in service because the link is released. The select command continues execution.
	Action: None
LIU7 1 IS NOT INSERVICE TRACING WILL BE ENABLED WHEN THE LIU7 GOES INSERVICE	
	Meaning: You selected an LIU7 number that is not in service. Monitoring starts when the LIU7 comes in service.The select command continues execution.
	Action: None
MSB7 3 IS NOT DEFINED FOR THIS OFFICE	
	Meaning: You attempted to select an MSB7 that is not datafilled for this office. The select command exits.
	Action: Retry the select command specifying an MSB7 that is datafilled for this office.
-continued-	

select (end)

Responses for the select command (continued)	
MAP output	**Meaning and action**
`MSB7 3 IS NOT INSERVICE`	
	Meaning: You specified an MSB7 that currently is not in service. The select command exits.
	Action: Check the status of the MSB7 and retry the select command.
`ONLY FOUR SELECTS ALLOWED IN FIELD ENVIRONMENT`	
	Meaning: You attempted to select when four entries were already in the match table. The field environment allows only four entries in the match table. The select command exits.
	Action: Release a link that is currently selected and retry the select command.
End	

send

Function

Use the send command to take the specified message from the message table and inject it into the given link. Once a message is sent, the system treats it the same way as any other CCS7 message.

send command parameters and variables		
Command	**Parameters and variables**	
send	*msg_num* [in / out] *linkset* *link_num*	

Parameters and variables	Description
in	This variable specifies the direction for the message. The direction is in on the link, into the node.
link_num	This variable specifies the link number of the linkset on which to send the test message.
linkset	This variable specifies the name of the linkset on which to send the test message.
msg_num	This variable specifies the message number of the test message to be sent.
out	This variable specifies the sending direction for the specified message. The direction is out on the link, into the network.

Qualifications

None

Example

The following table provides an example of the send command.

send (continued)

Example of the send command	
Example	**Task, response, and explanation**
send 0 in c7lkset1 0 ↵ *where* 0 specifies the message number of the test message to be sent c7lkset1 specifies the name of the linkset on which to send the test message 0 specifies the link number of the linkset on which to send the test message	

	Task:	Take the specified message from the message table and inject it into the given link.
	Response:	`Insert done`
	Explanation:	The system took test message number 1 and injected it into link number 0 in linkset C7LKSET1.

Responses

The following table provides explanations of the responses to the send command.

Responses for the send command	
MAP output	**Meaning and action**
`ERROR:  INVALID LINK NUMBER`	

	Meaning:	You entered a link number that is not datafilled for the specified linkset in the C7LINK table. The send command exits.
	Action:	Verify the link number and retry the send command.

`ERROR: pm num IS NOT INSERVICE`	

	Meaning:	The peripheral that is attached to the specified link, in the linkset, is not inservice. The send command exits.
	Action:	Assure that the peripheral is inservice or choose a different link, then retry the send command.

-continued-

send (continued)

Responses for the send command (continued)	
MAP output	**Meaning and action**
INVALID LINKSET NAME	
	Meaning: You entered a linkset name that is not datafilled in the C7LKSET table.The send command exits.
	Action: Verify the linkset name and retry the send command.
INVALID NETWORK TYPE IN DPC	
	Meaning: You specified a network type other than ANSI, CCITT, or TTC. The send command exits.
	Action: Change the network type of the message to one of the three valid network types, then retry the send command.
MESSAGE NUMBER msg HAS NOT BEEN BUILT YET	
	Meaning: You attempted to send a message specifying a message number that has not been built with the build command. The send command exits.
	Action: Verify the message number and retry the send command.
THE LIU7 IS NOT IN USE BY C7TU	
	Meaning: The LIU7 attached to the specified link is not in use by C7TU. The send command exits.
	Action: Use the select command to select the LIU7, then retry the send command.
UNABLE TO RESOLVE POINT CODES	
	Meaning: You attempted to send a message using a default linkset. The error occurred either because this linkset is not part of a routeset, or because the routeset is not a valid network. The send command exits.
	Action: Verify that the specified linkset is part of a routeset, and that the routeset is part of a valid network. Retry the send command.
-continued-	

send (end)

Responses for the send command (continued)	
MAP output	**Meaning and action**
WARNING: LINK MUST BE IN SYNC STATE FOR MESSAGE INJECTION WARNING: MESSAGE WILL BE SENT ANYWAY	
	Meaning: The peripheral that is attached to the specified link in the linkset is in service, but the link state is not set to synchronized. The message is sent to the peripheral. **Action:** None
End	

status

Function

Use the status command to display the current status of the C7TULINK environment. The display includes links and a shortened dump of the entries in the match table.

status command parameters and variables	
Command	**Parameters and variables**
status	brief verbose
Parameters and variables	**Description**
brief	This parameter displays only the links that are selected.
verbose	This parameter displays all links, marking the ones that are selected.

Qualifications

None

Examples

The following table provides examples of the status command.

Examples of the status command	
Example	**Task, response, and explanation**
status brief ↵	
	Task: Display selected links.

	Response:
	`LIU7            FTA        TRACING    THROTTLE`
	`201       4248 1000      ENABLE        20`
	`ITEM DISP NI Nettype DIR LINK Dist MSG SI H0H1`
	`0 MON ALL ANSI BOTH LS001 1 EXT XXX ISUP XXXXXX`

	Explanation: The system displays selected links.
	-continued-

status (end)

Examples of the status command (continued)

Example   **Task, response, and explanation**

---

status   verbose ↵

             Task: Display all links.

             Response:

```
LIU7           FTA         TRACING    THROTTLE
201       4248 1000     ENABLE         20
205       4248 1000     DISABLE        10
207       4248 1000     DISABLE        10
ITEM DISP NI Nettype DIR LINK Dist MSG SI H0H1
0 MON ALL ANSI BOTH LS001 1 EXT XXX ISUP XXXXXX
```

             Explanation: The system displays all links, marking those that are selected.

End

Response

The following table provides an explanation of the response to the status command.

Response for the status command

MAP output   Meaning and action

```
********** C7TU LINK ENVIRONMENT ***********
 MSB7  NODE    TRACING   MSGS   NACK
 msb   node    trace     msg    nack

 LIU7         FTA      TRACING
 liu          fta      trace

 ITEM  DISP NETW  DIR  ST  DIST  MSG  SI  H0  HI
 num   disp net   dir  st  dist  msg  si  h0  h1
```

             Meaning: The display includes links that currently are selected and a shortened dump of the entries in the match table.

             Action: None

C7TUTRFC level commands

Use the C7TUTRFC (CCS7 test utility traffic simulation test environment) level of the MAP to enter the traffic command environment.

Accessing the C7TUTRFC level

To access the C7TUTRFC level, enter the following command string from the CI level:

c7tu;c7tutrfc ↵

C7TUTRFC commands

The commands available at the C7TUTRFC MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table. These commands are listed for reference only.   The C7TUTRFC commands are laboratory test commands and are not intended for general field use.

C7TUTRFC commands	
Command	Page
cancel	C-159
help	C-161
modify	C-163
quiet	C-165
quit	C-167
report	C-171
reset	C-173
setup	C-175
start	C-177
status	C-179
-continued-	

C7TUTRFC commands (continued)	
Command	**Page**
stop	C-181
verbose	C-183
End	

cancel

Function

Use the cancel command to terminate a traffic test.

cancel command parameters and variables	
Command	**Parameters and variables**
cancel	This command is a laboratory test command and is not intended for general field use.

help

Function

Use the help command to receive online documentation for the C7TUTRFC directory.

help command parameters and variables	
Command	**Parameters and variables**
help	*all* *command_nam*
Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
command_nam	When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command		
Example	**Task, response, and explanation**	
help ↵		
	Task:	Access online documentation.
	Response:	Not currently available
	Explanation:	This example typifies a response for the help command.

Response

The following table provides an explanation of the response to the help command.

help (end)

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

modify

Function

Use the modify command to change the traffic test parameters.

modify command parameters and variables	
Command	**Parameters and variables**
modify	This command is a laboratory test command and is not intended for general field use.

quiet

Function

Use the quiet command to turn off sequence error logs.

quiet command parameters and variables	
Command	**Parameters and variables**
quiet	This command is a laboratory test command and is not intended for general field use.

quit

Function

Use the quit command to exit the C7TUTRFC directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⎡ *1 level* ⎤ ⎢ all ⎥ ⎢ *name* ⎥ ⎣ *n_levels* ⎦

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command	
Example	**Task, response, and explanation**
quit ↵	
	Task: Exit from this directory.
	Response: `CI:`
	Explanation: You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

report

Function

Use the report command to log the current status of the traffic test.

report command parameters and variables	
Command	**Parameters and variables**
report	This command is a laboratory test command and is not intended for general field use.

reset

Function

Use the reset command to reset all traffic counters to zero.

reset command parameters and variables	
Command	**Parameters and variables**
reset	This command is a laboratory test command and is not intended for general field use.

setup

Function

Use the setup command to set the traffic test parameters.

setup command parameters and variables	
Command	**Parameters and variables**
setup	This command is a laboratory test command and is not intended for general field use.

start

Function

Use the start command to start the traffic test.

start command parameters and variables	
Command	**Parameters and variables**
start	This command is a laboratory test command and is not intended for general field use.

status

Function

Use the status command to display the status of the C7TU traffic level.

status command parameters and variables	
Command	**Parameters and variables**
status	This command is a laboratory test command and is not intended for general field use.

stop

Function

Use the stop command to stop the C7TU traffic test temporarily.

stop command parameters and variables	
Command	**Parameters and variables**
stop	This command is a laboratory test command and is not intended for general field use.

verbose

Function

Use the verbose command to turn on sequence error logs.

verbose command parameters and variables	
Command	**Parameters and variables**
verbose	This command is a laboratory test command and is not intended for general field use.

CLOG level commands

Use the CLOG level of the MAP to access the switch-based Incoming Callers List (ICL). The ICL provides the subscriber with information pertaining to a limit of thirty-one of their unanswered, busy, or forwarded calls.

The ICL is transparent to the caller but provides the subscriber with the directory number (DN) and name of the calling party, the time and date of the call, the number of times the calling party called, and the status of the subscriber's line when the call was logged. The caller's name and number is dependent on the availability of Calling Name Delivery in the subscriber's area.

As the CLOG directory user, you see a blank field for any calls that do not make the name or number available. A blank name field is converted to Unknown Name or Private Name for the subscriber. A blank number field is converted to Unknown Number or Private Number for the subscriber.

> **CAUTION**
> **Could corrupt subscriber session**
> Do not use the CLOG directory commands to delete or change a customer's ICL entries.

If the Call Logging subscriber is in a Call Logging session and you are using the CLOG command on the subscriber's DN, there is the potential to corrupt the Call Logging subscriber's current session if ICL entries are changed or deleted through CLOG. Therefore, it is recommended that you not use the CLOG utility to delete or change customer's ICL entries.

If you do not enter all the required data necessary to complete the command, you are prompted for the fields. Those fields that accept default data are displayed for you to change. If you choose not to make changes, the default is accepted. While you are being prompted, you may terminate the command execution at any given time by using the abort command.

Accessing the CLOG level

To access the CLOG level, enter the following command from the CI level:

clog ↵

CLOG commands

The commands available at the CLOG MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

CLOG commands	
Command	**Page**
deq	C-187
help	C-191
queue	C-195
quit	C-203
reset	C-207
status	C-209

deq

Function

Use the deq command to delete the entry in the incoming callers list (ICL).

deq command parameters and variables		
Command	**Parameters and variables**	

deq	*requestee*	$\begin{bmatrix} \text{l} & requestor\_intra \\ \text{n} & requestor\_inter \end{bmatrix}$ $\begin{bmatrix} \underline{no\_name} \\ \text{nm} & requestor\_name \end{bmatrix}$

Parameters and variables	**Description**
no_name	Omitting this entry forces the system to default to expecting a blank requestor name when identifying the call.
l	This parameter describes the call as a intraoffice call.
n	This parameter describes the call as a interoffice call.
nm	This parameter specifies that the caller name follows to aid in identifying the incoming call.
requestee	This variable specifies the directory number (DN) of the requestee who owns the ICL.
requestor_inter	This variable specifies the DN of the interoffice requestor who is calling. The number must be 10 digits.
requestor_intra	This variable specifies the DN of the intraoffice requestor who is calling. The number must be only 7 digits.
requestor_name	This variable specifies the name of the requestor who is calling. The name may be up to 15 characters long.

Qualifications

If you do not enter all the required data necessary to complete the command, you are prompted for the fields. While you are being prompted, you may terminate the command execution at any given time by using the abort command.

You must match the requestor name exactly as it appears in the incoming call list. If the name is blank, you need not provide the requestor name.

The caller's name is dependent on the availability of Calling Name Delivery in the subscriber's area. Fifteen characters are available for the name.

deq (continued)

Examples

The following table provides examples of the deq command.

Examples of the deq command	
Example	**Task, response, and explanation**
deq 6211234 l 6214321 ↵ *where* 6211234 specifies the requestee number 6214321 specifies the requestor number	
	Task: Delete an intraoffice call. **Response:** `Request DEQUEUED.` **Explanation:** This command deletes the intraoffice call from 6214321 from the ICL. The requestor name field is blank.
deq 6211234 n 7043669566 ↵ *where* 6211234 specifies the requestee number 7043669566 specifies the requestor number	
	Task: Delete an interoffice call. **Response:** `Request DEQUEUED.` **Explanation:** This command deletes the interoffice call from 7043669566 from the ICL. The requestor name field is blank.
-continued-	

Examples of the deq command (continued)	
Example	**Task, response, and explanation**
deq 6211234 n 6136432312 nm John Doe ↵ *where* 6211234 specifies the requestee DN 6136432312 specifies the requestor DN John Doe specifies the requestor name	
	Task: Delete a interoffice call. **Response:** `Request DEQUEUED.` **Explanation:** This command deletes the interoffice call from John Doe at 6136432312 from the incoming call list. The requestor name is in the ICL.
End	

Response

The following table provides an explanation of the response to the deq command.

Response for the deq command	
MAP output	**Meaning and action**
`Requestor is NOT queued against requestee.`	
	Meaning: The requestor DN was not found in the ICL. **Action:** None

help

Function

Use the help command to receive online documentation for the CLOG directory.

help command parameters and variables	
Command	**Parameters and variables**
help	*all* *command_nam*
Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
command_nam	This variable specifies a valid CLOG directory command name. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command		
Example	**Task, response, and explanation**	
help ↵		
	Task:	Access online documentation.

	Response:	```CLOG: Call Logging Query Command```
		```Options: STATUS: <Requestee DN>```
		``` Display the Call Logging```
		``` Entries for a particular line```
		``` on the screen.```
		``` RESET: <Requestee DN>```
		``` Dequeue all messages from the```
		``` specified Requestee.```
		
		``` (cont.)```

	-continued-	

help (continued)

Example of the help command (continued)

Example	Task, response, and explanation

Response:

```
              DEQ:  <Requestee DN>
                    DEQueue the requestor from the
                    specified Requestee.  The
                    Requestor DN can be intraoffice
                    L plus the DN, or interoffice,
                    N plus the DN and an optional
                    Requestor Name.
                    <Requestee DN>
                        L <DN> |
                        N <DN: 10 digit TCAP>
                            <Name>
                                {NM
                                <Requestor Name
                                up to 15 chars>}
              QUEUE:  <Requestee DN>
                    QUEUE the Requestor for the
                    specified Requestee. The
                    Requestor can be intraoffice
                    L plus the DN, or interoffice,
                    N plus the DN and an optional
                    Requestor Name.
                    <Requestee DN>
                        L <DN> |
                        N <DN: 10 digit TCAP>
                            <Name>
                                {NM
                                <Requestor Name
                                up to 15 chars>}
                    <Options>
                        {O
                            <Date: 000101-991231>
                            <Time: 0-2359>
                            <Status: Auto|Savd>
                            <Suppress Name:
                                True|False>
                            <Suppress DN:
                                True|False>
                            <Repeat Calls: 1-15>
                            <State: Bsy|Uan|Fwd>}
```

Explanation: This example typifies a response for the help command string.

End

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

queue

Function

Use the queue command to add a specified entry to the incoming callers list (ICL).

queue command parameters and variables					
Command	**Parameters and variables**				
queue	*requestee* ⌈ l *requestor_intra* ⌉ ⌊ n *requestor_inter* ⌋		⌈ <u>*no_name*</u> ⌉ ⌊ nm *requestor_name* ⌋		(1) (2)
queue (continued)	(1) (2) ⌈ <u>*no_opt*</u> ⌊ o	<u>*sys_date*</u> *date*	<u>*sys_time*</u> *time*	<u>*auto stat*</u> *savd_stat*	<u>*f_s_name*</u> *t_s_name* (1) (2)
queue (continued)	(1) <u>*f s no*</u> (2) *t_s_no*	<u>*1*</u> *reps*	<u>uan</u> *state* ⌋		(end)

Parameters and variables	Description
<u>*1*</u>	Omitting this entry forces the system to default to specifying that the requestor has called once. This value is added to the total number of calls from this directory number (DN).
<u>*auto stat*</u>	Omitting this entry forces the system to default to saving the status of the current entry.
<u>*f s name*</u>	Omitting this entry forces the system to default to not suppressing the requestor name.
<u>*f s no*</u>	Omitting this entry forces the system to default to not suppressing the requestor DN.
<u>*no name*</u>	Omitting this entry forces the system to default to looking for the requestor name in Table NETNAMES for intraoffice calls and leaving the requestor name blank for interoffice calls.
<u>*no_opt*</u>	Omitting this entry forces the system to default to accepting the default information as correct.
<u>*sys_date*</u>	Omitting this entry forces the system to default to accepting the system date.
<u>*sys_time*</u>	Omitting this entry forces the system to default to accepting the system time.

-continued-

queue (continued)

queue command parameters and variables (continued)	
Parameters and variables	**Description**
<u>uan</u>	This default parameter indicates the last call was unanswered. Omitting this entry forces the system to default to displaying the unanswered state for the requestee line.
date	This variable specifies the year, month and day the requestor was enqueued. The valid range is 000101-991231. The date is not checked for validity of day within month.
l	This parameter describes the call as a intraoffice call.
n	This parameter describes the call as a interoffice call.
nm	This parameter specifies that the requestor name follows to aid in identifying the incoming call.
o	This parameter indicates that you want to change some or all of the remaining parameters.
reps	This variable counts the number of times that the requestor called. The valid entry range is 1-15. Once the counter has reached 15, it remains at 15 regardless of additional calls from the requestor.
requestee	This variable specifies the DN of the requestee subscriber who owns the ICL.
requestor_inter	This variable specifies the DN of the interoffice requestor who is calling. The number must be 10 digits.
requestor_intra	This variable specifies the DN of the intraoffice requestor who is calling. The number must be only seven digits.
requestor_name	This variable specifies the name of the requestor who is calling. The name may be up to 15 characters long.
savd_stat	This variable specifies the current entry status. This status indicates that the requestee has viewed the requestor call entry. The valid entry value is savd.
state	This variable specifies the state of the requestee line. The valid entries are bsy (busy), fwd (forward), and uan (unanswered).
t_s_name	This variable indicates that the requestor name is suppressed. The valid entry value is true.
-continued-	

queue (continued)

queue command parameters and variables (continued)	
Parameters and variables	**Description**
t_s_no	This variable indicates that the requestor number is suppressed. The valid entry value is true.
time	This variable specifies the time of the requestor call. The valid entry range is 0-2359.
End	

Qualification

If you do not enter all the required fields, you are prompted. Those fields that accept default data are displayed, and if you choose not to make changes, the default is accepted. You may terminate the prompted command execution at any given time by using the abort command.

Examples

The following table provides examples of the queue command.

Examples of the queue command	
Example	**Task, response, and explanation**
queue 6211234 l 6212125 o 910217 1510 auto false false 1 bsy ↵ *where*	
6211234	specifies the requestee number
6212125	specifies the requestor number
910217	specifies the date
1510	specifies the time
auto	specifies the status
false	specifies the name is not suppressed
false	specifies the number is not suppressed
1	specifies the repetitions
bsy	specifies the state
	Task: Enqueue an intraoffice call with options.
	Response: `Message queued on line.`
	Explanation: This command adds an intraoffice call to the ICL of February 17, 1991, at 3:10 in the afternoon as an incomplete busy call with the requestor name and number.
-continued-	

queue (continued)

Examples of the queue command (continued)		
Example	**Task, response, and explanation**	
queue 6211234 l 6215548 ↵ *where* 6211234 specifies the requestee number 6215548 specifies the requestor number		
	Task:	Enqueue an intraoffice call without options.
	Response:	`Message queued on line.`
	Explanation:	This command adds an intraoffice call to the ICL. Default data is added for all options. The switch looks for the requestor name in Table NETNAMES.
queue 6211234 n 9197378888 ↵ *where* 6211234 specifies the requestee number 9197378888 specifies the requestor number		
	Task:	Enqueue an interoffice call without options.
	Response:	`Message queued on line.`
	Explanation:	This command adds an interoffice call to the ICL. Default data is added for all options. The requestor name is blank.
queue 6211234 n 7043663921 nm David Charles ↵ *where* 6211234 specifies the requestee number 7043663921 specifies the requestor number David Charles specifies the requestor name		
	Task:	Enqueue an interoffice call with the requestor name but without options.
	Response:	`Message queued on line.`
	Explanation:	This command adds an interoffice call to the ICL. Default data is added for all options. The requestor name is David Charles.
-continued-		

queue (continued)

Examples of the queue command (continued)

Example	**Task, response, and explanation**

queue   6211234 n 7043663921 o 910219 2345 savd false false 3 fwd ↵
where

6211234	specifies the requestee number
7043663921	specifies the requestor number
910219	specifies the date
2345	specifies the time
savd	specifies the status
false	specifes the requestor name is not suppressed
false	specifies the requestor number is not suppressed
3	specifies the repetitions
fwd	specifies the state

Task:	Enqueue an interoffice call without the requestor name but with options.
Response:	`Message queued on line.`
Explanation:	This command adds an interoffice call to the ICL. The requestor name is blank. The call was placed on February 19, 1991 at 11:45 in the evening and the requestee viewed the data showing that an unidentified requestor at an unidentified number called three times and the calls were forwarded.

-continued-

queue (continued)

Examples of the queue command (continued)	
Example	**Task, response, and explanation**

queue 6211234 n 7043663921 nm Charles David o 910219 312 auto true false 2 uan ↵	
where	
6211234	specifies the requestee number
7043663921	specifies the requestor number
Charles David	specifies the requestor name
910219	specifies the date
312	specifies the time
auto	specifies the status
true	specifies the requestor name is suppressed
false	specifies the requestor number is not suppressed
2	specifies the repetitions
uan	specifies the state

	Task:	Enqueue an interoffice call with requestor name and options.
	Response:	`Message queued on line.`
	Explanation:	This command adds an interoffice call to the ICL. The requestor name is Charles David. The call was placed on February 19, 1991 at 3:12 in the morning and the requestee has not seen the data showing the requestor name and number who called twice and the calls were unanswered.

End

Responses

The following table provides explanations of the responses to the queue command.

Responses for the queue command	
MAP output	**Meaning and action**

`EITHER incorrect optional parameter(s) OR too many parameters.`	
	Meaning: You entered the command with an incorrect number of parameters.
	Action: Check the command syntax and reenter the command.

-continued-

queue (end)

Responses for the queue command (continued)	
MAP output	**Meaning and action**
ERROR: For Network DNs, must enter 10 digits in range from 0-9.	
	Meaning: You entered a seven digit number for an interoffice call.
	Action:　Reenter the command using a 10 digit directory number (DN).
Invalid REQUESTEE DN	
	Meaning: You specified an invalid DN.
	Action:　Reenter the command with a valid DN.
REQUESTEE does not have CALLOG.	
	Meaning: You specified a DN that does not have the call log feature.
	Action:　None
End	

quit

Function

Use the quit command to exit the CLOG directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⎡ *1 level* ⎤ ⎢ all ⎥ ⎢ *name* ⎥ ⎣ *n_levels* ⎦

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)	
Example	**Task, response, and explanation**
quit all ↵	
	Task: Exit from all levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory	
	Task: Exit from a specified directory without leaving any other directories.
	Response: AMADUMP>>> >
	Explanation: The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵	
	Task: Exit from a specified number of levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End	

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

reset

Function

Use the reset command to delete all the entries enqueued.

reset command parameters and variables	
Command	**Parameters and variables**
reset	*requestee*
Parameters and variables	**Description**
requestee	This variable specifies the requestee directory number (DN).

Qualifications

None

Example

The following table provides an example of the reset command.

Example of the reset command	
Example	**Task, response, and explanation**
reset 6211234 ↵ *where*	
6211234	specifies the requestee DN
	Task: Delete all entries for the requestee.
	Response: All messages are dequeued from the line.
	Explanation: This command deletes all the entries for the requestee DN 6211234.

reset (end)

Responses

The following table provides explanations of the responses to the reset command.

Responses for the reset command	
MAP output	**Meaning and action**
Invalid REQUESTEE DN	
	Meaning: You specified an invalid DN.
	Action: Reenter the command with a valid DN.
REQUESTEE does not have the Call Logging feature	
	Meaning: You specified a DN that does not have the call log feature.
	Action: None

status

Function

Use the status command to display all the entries and pertinent data associated with each entry.

status command parameters and variables	
Command	**Parameters and variables**
status	*requestee*
Parameters and variables	**Description**
requestee	This variable specifies the requestee directory number (DN).

Qualifications

None

status (continued)

Example

The following table provides an example of the status command.

Example of the status command	
Example	**Task, response, and explanation**

status   6211234 ↵
where

6211234   specifies the requestee DN

	Task:	Display the entries.
	Response:	`-------------------------------------------------`
		`REQUESTEE:  6211234    TYPE:  NCL`
		`REQUESTEE TYPE:  EMW   MCOS:  CLASSP`
		`FTRQ FORMAT:  NETWORK   TERM STATE:  BSY`
		`MSG STATE:  AUTO`
		`SUPPRESS NM:  N    SUPPRESS DN:  N`
		`REQUESTOR:  6136432312`
		`REQUESTOR NAME:  Jonathan Doe`
		`YEAR:  91  MONTH:  FEB  DAY:  05  TIME:  10:23`
		`REPEAT CALLS:  1`
		`NETWORK TYPE:  EMW`
		`NETWORK NAME:  PUBLIC`
		`-------------------------------------------------`
		`-------------------------------------------------`
		`REQUESTEE:  6211234    TYPE:  CL`
		`REQUESTEE TYPE:  EMW   MCOS:  CLASSP`
		`FTRQ FORMAT:  NODAL    TERM STATE:  UAN`
		`MSG STATE:  AUTO`
		`SUPPRESS NM:  N    SUPPRESS DN:  N`
		`REQUESTOR:  8134212321`
		`REQUESTOR NAME:  Thomas Petty`
		`YEAR:  91  MONTH:  JAN  DAY:  31  TIME:  23:18`
		`REPEAT CALLS:  4`
		`-------------------------------------------------`
		`End of requests.`
	Explanation:	The subscriber received one interoffice call from Jonathan Doe on February 5, 1991 at 10:23 in the morning while on the phone. The subscriber received four intraoffice calls from Thomas Petty, the last one occurring January 31, 1991 at 11:18 in the evening, which the subscriber did not answer.

status (end)

Responses

The following table provides explanations of the responses to the status command.

Responses for the status command	
MAP output	**Meaning and action**
Invalid DN	
	Meaning: You specified an invalid DN.
	Action: Reenter the command with a valid DN.
REQUESTEE does not have CALLOG option.	
	Meaning: You specified a DN that does not have the call log feature.
	Action: None

CPSTATUS level commands

Use the CPSTATUS level of the MAP to access the CPSTATUS tool to:

- measure all CPU occupancies including call processing occupancy.
- measure of additional CPU time available for call processing work.
- indicate overload and switch performance with respect to the switch's engineering.

Accessing the CPSTATUS level

To access the CPSTATUS level, enter the following command from the CI level:

cpstatus ↵

CPSTATUS commands

The commands available at the CPSTATUS MAP level are described in this chapter and are arranged in alphabetical order. The page number for each command is listed in the following table.

Note: The non-menu command cpstat can be used at the CI level to perform the same functions as the CPSTATUS level.

CPSTATUS commands	
Command	Page
quit	C-215

quit

Function

Use the quit command to exit the CPSTATUS directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	$\begin{bmatrix} \textit{1 level} \\ \text{all} \\ \textit{name} \\ \textit{n\_levels} \end{bmatrix}$

Parameters and variables	Description
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command	
Example	**Task, response, and explanation**
quit ↵	
	Task: Exit from this directory.
	Response: `CI:`
	Explanation: You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where*		
dskut	specifies a directory	
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

CUTOVER level commands

Use the CUTOVER level of the MAP to control the cutover mode for digital trunk controllers (DTC), carriers, and CICs that have been swung over from the old switch to the Digital Multiplex System (DMS).

Accessing the CUTOVER level

To access the CUTOVER level, enter the following command from the CI level:

cutover ↵

CUTOVER commands

The commands available at the CUTOVER MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

CUTOVER commands	
Command	**Page**
disable	C-221
emulate	C-223
flash	C-225
help	C-227
query	C-229
quit	C-231
testoff	C-235
teston	C-237

disable

Function

Use the disable command to deactivate the cut-over mode. When cut-over is disabled, the link peripheral processor (LPP) on the DMS no longer intercepts messages for the old switch.

disable command parameters and variables	
Command	**Parameters and variables**
disable	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the disable command.

Example of the disable command		
Example	**Task, response, and explanation**	
disable ↵		
	Task:	Deactivate the cut-over mode.
	Response:	CUTOVER IS DISABLED.
	Explanation:	This command deactivates the cut-over mode.

Response

The following table provides an explanation of the response to the disable command.

Response for the disable command	
MAP output	**Meaning and action**
COMMAND IGNORED, CUTOVER IS ALREADY OFF	
	Meaning: The emulate command has not been entered to enable cut-over.
	Action: Enter the emulate command with the appropriate PC to emulate.

emulate

Function

Use the emulate command to enable cut-over mode causing all messages for the old switch (specified by the PC number) to go through the LPP on the DMS.

emulate command parameters and variables			
Command	**Parameters and variables**		
emulate	ans17	*network*	
		network *cluster*	
		full *network* *cluster* *member*	
Parameters and variables	**Description**		
ans17	This parameter indicates the network type.		
cluster	This variable specifies the cluster. The valid entry range is 0-255.		
full	This parameter indicates the scope of the cut-over. This parameter is associated with values for the variables *network*, *cluster*, and *member*.		
member	This variable specifies the member. The valid entry range is 0-255.		
network	This variable specifies the network and defines which PC will be emulated by the DMS. The valid entry range is 0-255.		

Qualifications

None

emulate (end)

Example

The following table provides an example of the emulate command.

Example of the emulate command	
Example	**Task, response, and explanation**
emulate ans17 full 1 3 5 ↵ *where* 1 specifies the network and defines which PC will be emulated by the DMS 3 specifies the cluster 5 specifies the member	
	Task: Enable the DMS to emulate a PC. **Response:** `CUT-OVER ENABLED` **Explanation:** The emulate command was successful. Information is downloaded to the LPP so that messages for the old switch are routed through the LPP. The STP/DMS signaling link must be activated.

Responses

The following table provides explanations of responses to the emulate command.

Responses for the emulate command	
MAP output	**Meaning and action**
`CUT-OVER HAS ALREADY BEEN STARTED.`	
	Meaning: Cut-over mode already is enabled. **Action:** None
`PC OUT OF RANGE`	
	Meaning: The PC is not within the range. **Action:** Check the value entered for the *network* variable and enter the emulate command again.

flash

Function

Use the flash command to flash-cut all trunks to the DMS. The DMS then assumes the previously-ascertained PC.

flash command parameters and variables	
Command	**Parameters and variables**
flash	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the flash command.

Example of the flash command		
Example	**Task, response, and explanation**	
flash ↵		
	Task:	Activate the flash-cut process.
	Response:	FLASHCUT MUST BE PERFORMED IMMEDIATELY AFTER THIS COMMAND ENTER Y TO CONTINUE FLASHCUT >y
	Explanation:	This command initiates the flash-cut process. A positive response to the activity confirmation prompt causes the flash-cut to occur.

flash (end)

Response

The following table provides an explanation of the response to the flash command.

Response for the flash command	
MAP output	**Meaning and action**
CUT-OVER IS OFF, TURN ON FIRST WITH EMULATE COMMAND	
	Meaning: Cut-over must be active and the point code to be emulated must be defined.
	Action: Use the emulate command before issuing the flash command.

help

Function

Use the help command to receive online documentation for the CUTOVER directory.

help command parameters and variables	
Command	**Parameters and variables**
help	_all_ _command_nam_

Parameters and variables	Description
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
_command_nam_	This variable specifies a valid CUTOVER directory command name. When the _command_nam_ variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help ↵	
	Task: Access online documentation.

	Response: `CUTOVER Environment - The available commands are:`
	
	`*** DISABLE`
	`*** EMULATE`
	`*** FLASH`
	`*** QUERY`
	`*** QUIT`
	`*** TESTOFF`
	`*** TESTON`

	Explanation: This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

query

Function

Use the query command to provide information on the status of the cut-over. The display indicates whether or not cut-over mode is enabled. If cut-over mode is enabled, the display identifies the DTCs, carriers, and any CICs that have been swung-over.

query command parameters and variables	
Command	**Parameters and variables**
query	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the query command.

Example of the query command	
Example	**Task, response, and explanation**
query ↵	

	Task:	Query the status of the cut-over.
	Response:	CUTOVER IS ON, TRUNKS SWUNG OVER AS INDICATED DTC # 0 CARRIER # 2 ISUP TRUNKS (CICS) SWUNG OVER 824 825 826
	Explanation:	This command produces a summary of the cut-over status.

query (end)

Response

The following table provides an explanation of the response to the query command.

Response for the query command	
MAP output	**Meaning and action**
CUT-OVER IS OFF, TURN ON FIRST WITH EMULATE COMMAND	
	Meaning: Cut-over is not enabled.
	Action: None

quit

Function

Use the quit command to exit the CUTOVER directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⎡ *1 level* ⎤ ⎢ all ⎥ ⎢ *name* ⎥ ⎣ *n_levels* ⎦

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	`CI:`
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where*		
dskut	specifies a directory	
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	`AMADUMP>>>` `>`
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	`CI:`
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

testoff

Function

Use the testoff command to specify the DTC and carrier for the CICs that are being swung over. Messages for these CICs are routed to the old switch again.

testoff command parameters and variables		
Command	**Parameters and variables**	
testoff	*dtctype* *dtcnum* *carrnum*	
Parameters and variables	**Description**	
carrnum	This variable specifies the carrier number for swung-over trunks. The valid entry range is 0-19.	
dtcnum	This variable specifies the DTC number for swung-over trunks. The valid entry range is 0-127.	
dtctype	This variable specifies the type of DTC. The valid entry range for the DTC7 is 0-1. (The valid entry value for the PCM-30 digital trunk controller (PDTC) is 1.)	

Qualifications

None

Example

The following table provides an example of the testoff command.

testoff (end)

Example of the testoff command	
Example	**Task, response, and explanation**
testoff dtc6 3 ↵ *where* dtc6 specifies the DTC number for swung-over trunks 6 specifies the carrier number for swung-over trunks	

	Task:	Confirm that the DMS receives messages from CICs on the DTC.
	Response:	`CICS ON DTC AND CARRIER SPECIFIED HAVE BEEN SWUNG` `OVER.`
	Explanation:	This commands informs the link peripheral processor (LPP) that CICs on DTC 6 and carrier 3 have been swung over, and therefore messages for these CICs should now be sent to the DMS rather than the old switch.

Responses

The following table provides explanations of responses to the testoff command.

Responses for the testoff command	
MAP output	**Meaning and action**
`CARRIER DOES NOT EXIST`	
	Meaning: The carrier number is not datafilled.
	Action: Check the carrier number and reenter a valid number.
`CUT-OVER HAS NOT BEEN STARTED (USE EMULATE COMMAND)`	
	Meaning: The emulate command has not been entered.
	Action: Enter the emulate command.
`DTC DOES NOT EXIST`	
	Meaning: The DTC number is not datafilled.
	Action: Check the DTC number and re-enter a valid number.

teston

Function

Use the teston command to enter the DTC and carrier for the CICs that are being swung over. Messages for these CICs then will be received by the DMS.

teston command parameters and variables		
Command	**Parameters and variables**	
teston	*dtctype* *dtcnum* *carrnum*	
Parameters and variables	**Description**	
carrnum	This variable specifies the carrier number for swung-over trunks. The valid entry range is 0-19.	
dtcnum	This variable specifies the DTC number for swung-over trunks. The valid entry range is 0-127.	
dtctype	This variable specifies the type of DTC. The valid entry range for the DTC7 is 0-1. (The valid entry value for the PCM-30 digitial trunk controller (PDTC) is 1.)	

Qualifications

None

Example

The following table provides an example of the teston command.

teston (end)

Example of the teston command	
Example	**Task, response, and explanation**
teston dtc6 3 ↵ *where* dtc6 specifies the DTC number for swung-over trunks 3 specifies the carrier number for swung-over trunks	
	Task: Confirm that the DMS receives messages from CICs on the DTC.
	Response: CICS ON DTC AND CARRIER SPECIFIED HAVE BEEN SWUNG OVER.
	Explanation: This command informs the link peripheral processor (LPP) that CICs on DTC6 and carrier 3 have been swung over. Messages for these CICs should now be sent to the DMS rather than the old switch.

Responses

The following table provides explanations of responses to the teston command.

Responses for the teston command	
MAP output	**Meaning and action**
CARRIER DOES NOT EXIST	
	Meaning: The carrier number is not datafilled.
	Action: Check the carrier number and reissue a valid number.
CUT-OVER HAS NOT BEEN STARTED (USE EMULATE COMMAND)	
	Meaning: The emulate command has not been entered.
	Action: Enter the emulate command.
DTC DOES NOT EXIST	
	Meaning: The DTC number is not datafilled.
	Action: Check the DTC number and reissue a valid number.

DASIM level commands

Use the directory assistance simulator (DASIM) level of the MAP to set up parameters to control the simulator and monitor the messages between traffic operator position system (TOPS) call processing and the simulator. The DASIM directory commands can be used as a designer tool when testing call-processing messages to the DAS data base.

If you do not enter the required data to complete the command, you will be prompted for the fields. At any time while in the prompt mode, you may terminate the command by using the abort command.

Accessing the DASIM level

To access the DASIM level, enter the following command from the CI level:

dasim ↵

DASIM commands

The commands available at the DASIM MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

DASIM commands	
Command	**Page**
ann	D-3
ccannopt	D-7
ccbiltype	D-9
ccpoolid	D-11
clear	D-13
data	D-15
dump	D-19
help	D-21
-continued-	

DASIM commands (continued)	
Command	**Page**
intdn	D-23
lang	D-25
leave	D-27
listing	D-29
logdtl	D-35
poolid	D-37
posrsn	D-39
q	D-41
reqdn	D-43
rfpdata	D-45
rst	D-49
scencci	D-51
scenibm	D-59
servnum	D-65
setlink	D-69
sim	D-71
trace	D-73
vendor	D-75
End	

ann

Function

Use the ann command to display or set the announcement returned by the simulator in an audio response unit (ARU) request message. You can set a unique announcement for each scenario.

ann command parameters and variables	
Command	**Parameters and variables**
ann	display set $\begin{bmatrix} ann\_num \\ scen\_num \end{bmatrix}$

Parameters and variables	**Description**
ann_num	This variable specifies the announcement number. The valid entry range is 0-254.
display	This parameter displays the announcement data returned by the simulator.
scen_num	This variable specifies the scenario number. The valid entry range is 0-23.
set	This parameter sets the scenario number and announcement number returned by the simulator.

Qualification

None

ann (continued)

Examples

The following table provides examples of the ann command.

Examples of the ann command	
Example	**Task, response, and explanation**
ann **display** ↵	
	Task: Display the announcement data returned by the simulator.
	Response:

```
ANNOUNCEMENT DATA:
SCEN INDEX    SIM DESCRIPTION                      ANN NUMBER
         0    AUTO QUOTE DA CALL                        0
         1    AUTO QUOTE AUTO-INTERCEPT CALL            0
         2    AUTO QUOTE ONI-INTERCEPT CALL             0
         3    AUTO QUOTE MULTIPLE REQ DA CALL           0
         4    AUTO QUOTE WITH RECALL DA CALL            0
         5    VOICE QUOTE MEMORY CALL                   0
         6    VOICE QUOTE DA CALL                       0
         7    VOICE QUOTE AUTO-INTERCEPT CALL           0
         8    VOICE QUOTE AUTO-INTER, NO ARU            0
```

	Explanation: This command displays the announcement data settings for the simulator.
ann **set 2 0**↵ *where* 2 specifies the index entry 0 specifies the announcement number	
	Task: Set the scenario and announcement numbers.
	Response: None
	Explanation: This command sets the scenario and announcement numbers of the announcement returned by the simulator.

ann (end)

Responses

The following table provides explanations of the responses to the ann command.

Responses for the ann command	
MAP output	**Meaning and action**
`Undefined command`	
	Meaning: You entered the command incorrectly.
	Action: Reenter the command.
`Wrong type`	
	Meaning: You entered an invalid parameter.
	Action: Reenter the command with an appropriate parameter.

ccannopt

Function

Use the ccannopt command to set the call completion and announcement.

ccannopt command parameters and variables	
Command	**Parameters and variables**
ccannopt	ccann ccnoann nocc
Parameters and variables	**Description**
ccann	This parameter indicates that call completion uses an announcement.
ccnoann	This parameter indicates that call completion does not use an announcement.
nocc	This parameter indicates that there is no call completion.

Qualifications

None

Examples

The following table provides examples of the ccannopt command.

Examples of the ccannopt command		
Example	**Task, response, and explanation**	
ccannopt ccann ↵		
	Task:	Set call completion to an announcement.
	Response:	CALL COMPLETION AND ANNOUNCEMENT OPTION IS SET TO: CCANN
	Explanation:	This command sets call completion to an announcement.
	-continued-	

ccannopt (end)

Examples of the ccannopt command (continued)		
Example	**Task, response, and explanation**	
ccannopt ccnoann ↵		
	Task:	Set call completion to no announcement.
	Response:	CALL COMPLETION AND ANNOUNCEMENT OPTION IS SET TO: CCNOANN
	Explanation:	This command sets call completion to no announcement.
ccannopt nocc ↵		
	Task:	Set call completion to no completion.
	Response:	CALL COMPLETION AND ANNOUNCEMENT OPTION IS SET TO: NOCC
	Explanation:	This command sets call completion to no completion.
End		

Response

The following table provides an explanation of the response to the ccannopt command.

Response for the ccannopt command	
MAP output	**Meaning and action**
`Invalid symbol`	
	Meaning: You entered an invalid parameter.
	Action: Enter a valid parameter to continue, or abort to cancel.

ccbiltype

Function

Use the ccbiltype command to set the call completion bill type.

ccbiltype command parameters and variables	
Command	**Parameters and variables**
ccbiltype	altbil autocol contbil none sentpd
Parameters and variables	**Description**
altbil	This parameter indicates that the directory assistance is toll free or chargeable and the call completion is billable to a credit card or third party number.
autocol	This parameter indicates that the party who ordered call completion pays for the call.
contbil	This parameter indicates that the calling party pays for the entire call.
none	This parameter indicates that there is no billing for call completion.
sentpd	This parameter indicates that the directory assistance is toll free but the call completion is charged to the customer.

Qualifications

None

ccbiltype (end)

Examples

The following table provides examples of the ccbiltype command.

Examples of the ccbiltype command	
Example	**Task, response, and explanation**
ccbiltype altbil ↵	
	Task: Set call completion to alternate billing.
	Response: CC BILL TYPE IS SET TO: ALTBIL
	Explanation: This command sets call completion to alternate billing.
ccbiltype none ↵	
	Task: Set call completion to no billing.
	Response: CC BILL TYPE IS SET TO: NONE
	Explanation: This command sets call completion to no billing.

Response

The following table provides an explanation of the response to the ccbiltype command.

Response for the ccbiltype command	
MAP output	**Meaning and action**
Invalid symbol	
	Meaning: You entered an invalid parameter.
	Action: Enter a valid parameter to continue, or abort to cancel.

ccpoolid

Function

Use the ccpoolid command to change the call completion pool ID. The call completion pool can be different from the regular pool.

ccpoolid command parameters and variables	
Command	**Parameters and variables**
ccpoolid	*pool_id*
Parameters and variables	**Description**
pool_id	This variable specifies the pool identification. The valid entry range is 0-15.

Qualifications

None

Example

The following table provides an example of the ccpoolid command.

Example of the ccpoolid command	
Example	**Task, response, and explanation**
ccpoolid 0 ↵ *where* 0 specifies the pool identification	
	Task: Set the pool identification.
	Response: CC POOLID IS SET TO 0
	Explanation: This command sets the pool identification to zero.

ccpoolid (end)

Responses

The following table provides explanations of the responses to the ccpoolid command.

Responses for the ccpoolid command	
MAP output	**Meaning and action**
`Out of range`	
	Meaning: You entered a pool identification of less than 0 or more than 15.
	Action: Enter an appropriate pool identification to continue or abort to cancel.
`Wrong type`	
	Meaning: You entered an alphabetic identification.
	Action: Enter an appropriate pool identification to continue or abort to cancel.

clear

Function

Use the clear command to clear all saved response messages in the trace file, and reset the counter used to index the message file to zero.

clear command	
Command	**Parameters and variables**
clear	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the clear command.

Example of the clear command		
Example	**Task, response, and explanation**	
clear ↵		
	Task:	Clear the trace file and reset the message file counter.
	Response:	None
	Explanation:	This command clears the trace file and resets the message file counter.

Response

The following table provides an explanation of the response to the clear command.

Response for the clear command	
MAP output	**Meaning and action**
`undefined command`	
	Meaning: The command is not part of the load.
	Action: None

data

Function

Use the data command to display the current value of all simulator data as set by the setlink, vendor, lang, posrsn, poolid, ann, reqdn, trace, and scencci select commands.

data command	
Command	**Parameters and variables**
data	There are no parameters or variables.

Qualifications

None

data (continued)

Example

The following table provides an example of the data command.

Example of the data command	
Example	**Task, response, and explanation**
data ↵	
	Task: Show the values of simulator data.

	Response: SIMULATOR LINK IS SET TO: FAIL
	 VENDOR IS SET TO: NONE
	 THE LANGUAGE IS SET TO: DEFAULT
	 THE POS REQUEST REASON IS SET TO: MISCELLANEOUS
	 THE LOGIN LOGOUT DETAILS IS SET TO: SUCCESS
	 THE POOLID IS SET TO: 0
	
	LISTING STATUS FIELDS:
	LISTING STATUS: 15-NIL
	SUBSTATUS1: 15-NIL
	SUBSTATUS2: 15-NIL
	SUBSTATUS3: 15-NIL
	SUBSTATUS4: 15-NIL
	SUBSTATUS5: 15-NIL
	SUBSTATUS6: 15-NIL
	SUBSTATUS7: 15-NIL
	
	ANNOUNCEMENT DATA:
	SCEN INDEX SIM DESCRIPTION ANN NUMBER
	       0 AUTO QUOTE DA CALL 0
	       1 AUTO QUOTE AUTO-INTERCEPT CALL 0
	       2 AUTO QUOTE ONI-INTERCEPT CALL 0
	       3 AUTO QUOTE MULTIPLE REQ DA CALL 0
	       4 AUTO QUOTE WITH RECALL DA CALL 0
	       5 VOICE QUOTE MEMORY CALL 0
	       6 VOICE QUOTE DA CALL 0
	       7 VOICE QUOTE AUTO-INTERCEPT CALL 0
	       8 VOICE QUOTE AUTO-INTER, NO ARU 0
	THE SCENARIO SELECTED IS: 0
	
	THE REQUESTED DN IS: 0000000000
	
	TRACING IS NOT ACTIVATED
	END OF SIMULATOR DATA

	Explanation: This command displays the current value of all simulator data.

Response

The following table provides an explanation of the response to the data command.

Response for the data command	
MAP output	**Meaning and action**
`undefined command`	
	Meaning: You entered a command which is not part of this load.
	Action: None

dump

Function

Use the dump command to display all saved messages (up to 20 stored in buffer at one time) stored in the trace file.

dump command	
Command	**Parameters and variables**
dump	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the dump command.

Example of the dump command	
Example	**Task, response, and explanation**
dump ↵	
	Task: Display all saved messages stored in the trace file.

	Response: `CCI MSG TYPE: POS STATUS`
	`SWITCH ID     15`
	`DAS AREA      FFFF`
	`DETAIL        1`
	`S/W GENERIC   1`
	`TERMINAL ID   312`
	`MONITOR ID    -1`
	`OPERATOR ID   312`
	`MESSAGE: 1.`
	`CCI MSG TYPE: POS STATUS REPLY`
	`SWITCH ID     15`
	`DETAIL        0`
	`DAS CALLID    1`
	`TERMINAL ID   17976`
	`OPERATOR ID   312`
	`MESSAGE: 2.`
	`END OF MESSAGES`

	Explanation: This command displays all saved messages stored in the trace file.

dump (end)

Response

The following table provides an explanation of the response to the dump command.

Response for the dump command	
MAP output	**Meaning and action**
`undefined command`	
	Meaning: The command is not part of the load.
	Action: None

help

Function

Use the help command to list the commands. See the query command.

help command parameters and variables	
Command	**Parameters and variables**
help	There are no parameters or variables.

Qualification

The commands listed vary depending on the vendor type specified.

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help ↵	
	Task: List the directory commands.

	Response: `TOPSVR VENDOR & DATALINK CONTROLLING COMMANDS:`
	`    DATA`
	`    SETLINK`
	`    VENDOR`
	`    SIM`
	`    LOGDTL`
	`    LANG`
	`    POSRSN`
	`    CCANNOPT`
	`    POOLID`
	`    CCPOOLID`
	`    LISTING`
	`    ANN`
	`    REQDN`
	`    INTDN`
	`    CCBILTYPE`
	`    SCENCCI`
	`    SERVNUM`
	`    CLEAR`
	`    TRACE`
	`    DUMP`
	`    RST`
	`    HELP`

	Explanation: This command lists the commands in the DASIM directory.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded.
	Action: None

intdn

Function

Use the intdn command to change the intercept ONI/ANIF directory number.

intdn command parameters and variables	
Command	**Parameters and variables**
intdn	*int_dn*

Parameters and variables	**Description**
int_dn	This variable specifies the number to intercept when using call completion.

Qualifications

None

Example

The following table provides an example of the intdn command.

Example of the intdn command	
Example	**Task, response, and explanation**
intdn 1234567 ↵ *where*	
1234567	specifies the directory number
	Task: Set a call completion directory number.
	Response: None
	Explanation: This command sets a call completion for directory number 1234567.

intdn (end)

Response

The following table provides an explanation of the response to the intdn command.

Response for the intdn command	
MAP output	**Meaning and action**
INTERCEPTED NUMBER INVALID	
	Meaning: You entered a directory number that is not valid for call completion.
	Action: Reenter the command with a valid directory number.

lang

Function

Use the lang command to specify the language identifier returned in the details field of the audio response unit (ARU) request message by the simulator.

lang command parameters and variables	
Command	**Parameters and variables**
lang	default primary secondary

Parameters and variables	Description
default	This parameter is used to request the default language identifier.
primary	This parameter is used to request the first choice language identifier.
secondary	This parameter is used to request the second choice language identifier.

Qualifications

None

Examples

The following table provides examples of the lang command.

Examples of the lang command	
Example	**Task, response, and explanation**
lang default ↵	
	Task: Set the language identifier to the default.
	Response: LANGUAGE IS SET TO: DEFAULT
	Explanation: This command sets the language identifier to default.
-continued-	

lang (end)

Examples of the lang command (continued)	
Example	**Task, response, and explanation**
lang primary ↵	

	Task:	Set the language identifier to the primary language.
	Response:	LANGUAGE IS SET TO: PRIMARY
	Explanation:	This command sets the language identifier to the primary language identifier.
End		

Response

The following table provides an explanation of the response to the lang command.

Response for the lang command	
MAP output	**Meaning and action**
INVALID SYMBOL	

	Meaning: You entered the language identifier incorrectly.
	Action: Reenter the correct language identifier.

leave

Function

Use the leave command to exit from the DASIM directory.

leave command parameters and variables	
Command	**Parameters and variables**
leave	_current_ all
Parameters and variables	**Description**
all	This parameter leaves all directories and returns you to the CI MAP level.
current	Omitting this entry forces the system to default to leave the current DASIM directory.

Qualifications

None

Example

The following table provides an example of this command.

Example of the leave command		
Example	**Task, response, and explanation**	
leave ↵		
	Task:	Quit this directory.
	Response:	`CI:`
	Explanation:	This command exits this directory and returns to the CI MAP level.

leave (continued)

Response

The following table provides a common response to this command.

Response for the leave command	
MAP output	**Meaning and action**
CI:	
	Meaning: This prompt indicates that you have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.

listing

Function

Use the listing command to display or set the listing status fields returned by the simulator in the AMA transfer, ARU request, and POS release messages.

listing command parameters and variables	
Command	**Parameters and variables**
listing	display
	status *listing*
	substatus1 *dir_info*
	substatus2 *pub_info*
	substatus3 *list_pres*
	substatus4 *list_post*
	substatus5 *lsdb_bill*
	substatus6 *oper_bill*
	substatus7 *auto_colct*

Parameters and variables	**Description**
auto_colct	This variable specifies auto collection. The valid values are:
	1 not marked auto collect
	2 listing marked auto collect
	3 listing marked auto collect for local only
	4 listing marked auto collect for toll only
	9 unknown
	15 nil
dir_info	This variable specifies directory information. The valid values are:
	1 listing found in local directory
	2 listing not found in local directory
	9 unknown
	15 nil
display	This parameter displays the listing status fields returned by the simulator.
listing	This variable specifies the listing status. The valid values are:
	1 listing found
	2 listing not found
	3 LSDB QUERY not made
	9 unknown
	15 nil
	-continued-

listing (continued)

listing command parameters and variables (continued)	
Parameters and variables	**Description**
list_post	This variable specifies listing posted. The valid values are: 1 listing posted 2 listing pre-posted 9 unknown 15 nil
list_pres	This variable specifies listing presence. The valid values are: 1 listing 2 listing 9 unknown 15 nil
lsdb_bill	This variable specifies LSDB billing. The valid values are: 1 listing not marked free by LSDB 2 listing marked free by LSDB 9 unknown 15 nil
oper_bill	This variable specifies the operator billing. The valid values are: 1 listing not marked free by the operator 2 listing marked free by the operator 9 unknown 15 nil
pub_info	This variable specifies publishing information. The valid values are: 1 listing published 2 listing non-published 3 unknown 15 nil
status	This parameter sets the listing status value returned by the simulator.
substatus1	This parameter sets the directory information value returned by the simulator.
substatus2	This parameter sets the publishing information value returned by the simulator.
substatus3	This parameter sets the listing presence value returned by the simulator.
-continued-	

listing (continued)

listing command parameters and variables (continued)	
Parameters and variables	**Description**
substatus4	This parameter sets the listing posted value returned by the simulator.
substatus5	This parameter sets the LSDB billing value returned by the simulator.
substatus6	This parameter sets the operator billing value returned by the simulator.
substatus7	This parameter sets the auto collection value returned by the simulator.
End	

Qualifications

None

Examples

The following table provides examples of the listing command.

Examples of the listing command		
Example	**Task, response, and explanation**	
listing display ↵		
	Task:	Display the listing status fields.
	Response:	LISTING STATUS: 15-NIL SUBSTATUS1 15-NIL SUBSTATUS2 15-NIL SUBSTATUS3 15-NIL SUBSTATUS4 15-NIL SUBSTATUS5 15-NIL SUBSTATUS6 15-NIL SUBSTATUS7 15-NIL
	Explanation:	You see a listing of the status fields returned by the simulator.
-continued-		

listing (continued)

Examples of the listing command (continued)	
Example	**Task, response, and explanation**
listing status *3* ↵ *where*	
3	specifies the listing value
	Task: Specify a listing status value.
	Response: `LISTING STATUS:  3`
	Explanation: You set the listing status to 3, which means 'LSDB QUERY not made'.
listing substatus6 *2* ↵ *where*	
2	specifies the operator billing value
	Task: Specify the operator billing value.
	Response: `LISTING SUBSTATUS6 2`
	Explanation: You set the operator billing to 2, which means 'listing marked free by the operator'.
End	

Responses

The following table provides explanations of the responses to the listing command.

Responses for the listing command	
MAP output	**Meaning and action**
`LISTING SUBSTATUS6 2`	
	Meaning: You set a status field correctly.
	Action: None
-continued-	

listing (end)

Responses for the listing command (continued)	
MAP output	**Meaning and action**
OUT OF RANGE	
	Meaning: You entered a parameter that exceeds the correct range.
	Action: Reenter a valid parameter within the correct range.
UNDEFINED COMMAND	
	Meaning: You entered the command incorrectly.
	Action: Reenter the correct command.
End	

logdtl

Function

Use the logdtl command to set the login and logout success or failure details returned by the simulator in the position status reply message.

logdtl command parameters and variables	
Command	**Parameters and variables**
logdtl	failure success
Parameters and variables	**Description**
failure	This parameter sets login or logout to simulator failure.
success	This parameter sets login or logout to simulator success.

Qualifications

None

Example

The following table provides an example of the logdtl command.

Example of the logdtl command		
Example	**Task, response, and explanation**	
logdtl success ↵		
	Task:	Set the login and logout details.
	Response:	LOG DETAILS SET TO SUCCESS
	Explanation:	This command sets the login and logout success details returned by the simulator in the position status reply message.

logdtl (end)

Responses

The following table provides explanations of the responses to the logdtl command.

Responses for the logdtl command	
MAP output	**Meaning and action**
INVALID SYMBOL	
	Meaning: You entered an incorrect parameter.
	Action: Reenter the command using an appropriate parameter.
LOG DETAILS ALREADY SET TO \<setting\>	
	Meaning: You already set the login or logout details to this setting.
	Action: None
LOG DETAILS SET TO \<setting\>	
	Meaning: You set the login and logout details.
	Action: None
UNDEFINED COMMAND	
	Meaning: The logdtl command is not part of the load.
	Action: None

poolid

Function

Use the poolid command to set the pool ID returned by the simulator.

poolid command parameters and variables	
Command	**Parameters and variables**
poolid	*poolid*
Parameters and variables	**Description**
poolid	This variable specifies the pool identification. The valid entry range is 0-15.

Qualifications

None

Example

The following table provides an example of the poolid command.

Example of the poolid command		
Example	**Task, response, and explanation**	
poolid 15 ↵ *where*		
15	specifies the pool identification	
	Task:	Set the poolid.
	Response:	POOLID IS SET TO: 15
	Explanation:	This command sets the poolid returned by the simulator to 15.

poolid (end)

Responses

The following table provides explanations of the responses to the poolid command.

Responses for the poolid command	
MAP output	**Meaning and action**
OUT OF RANGE	
	Meaning: You entered an invalid value for the pool identification.
	Action: Reenter the command with an appropriate value.
POOLID IS SET TO: 15	
	Meaning: You successfully executed the command.
	Action: None
UNDEFINED COMMAND	
	Meaning: You entered the command incorrectly.
	Action: Reenter the command correctly.
WRONG TYPE	
	Meaning: You entered an invalid character for the pool identification.
	Action: Reenter the command with an appropriate value.

posrsn

Function

Use the posrsn command to determine the position request reason returned by the simulator in the detail field of the POS request message.

posrsn command parameters and variables	
Command	**Parameters and variables**
posrsn	misc priority split
Parameters and variables	**Description**
misc	This parameter sets the position request reason to miscellaneous report.
priority	This parameter sets the position request reason to priority status code.
split	This parameter sets the position request reason to split referral.

Qualifications

None

Examples

The following table provides examples of the posrsn command.

Examples of the posrsn command		
Example	**Task, response, and explanation**	
posrsn split ↵		
	Task:	Set the position request reason.
	Response:	POS REQUEST REASON IS SET TO: SPLIT
	Explanation:	This command sets the position request reason to split referral.
	-continued-	

posrsn (end)

Examples of the posrsn command (continued)	
Example	**Task, response, and explanation**
posrsn priority ↵	
	Task: Set the position request reason.
	Response: `POS REQUEST REASON IS SET TO:  PRIORITY`
	Explanation: This command sets the position request reason to priority status.
End	

Responses

The following table provides explanations of the responses to the posrsn command.

Responses for the posrsn command	
MAP output	**Meaning and action**
`INVALID SYMBOL`	
	Meaning: You entered the reason name incorrectly.
	Action: Reenter the correct reason name.
`POS REQUEST REASON IS SET TO:  PRIORITY`	
	Meaning: You successfully executed the command.
	Action: None
`UNDEFINED COMMAND`	
	Meaning: You entered the command incorrectly.
	Action: Reenter the correct command.

q

Function

Use the q command to access online documentation for the DASIM directory commands.

q command parameters and variables	
Command	**Parameters and variables**
q	*cmd_name*

Parameters and variables	**Description**
cmd_name	This variable specifies a valid DASIM directory command.

Qualifications

None

Example

The following table provides an example of the q command.

Example of the q command	
Example	**Task, response, and explanation**
q ccannopt ↵ *where* ccannopt specifies the command name	

	Task:	Access online documentation.

	Response:	COMMAND TO SET THE CALL COMPLETION AND
		ANNOUNCEMENT OPTION
		Parms: <CC/ANN OPTION> {NOCC,
		                                      CCANN,
		                                      CCNOANN}

	Explanation:	This command provides a short description of the ccannopt command.

q (end)

Response

The following table provides an explanation of the response to the q command.

Response for the q command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded.
	Action: None

reqdn

Function

Use the reqdn command to set the directory number returned by the simulator in the DN field of the response message.

reqdn command parameters and variables	
Command	**Parameters and variables**
reqdn	*dn*
Parameters and variables	**Description**
dn	This variable identifies the directory number returned by the simulator in response messages. The valid entry values are ten digits for non-local directory numbers and seven digits for local directory numbers.

Qualifications

None

Examples

The following table provides examples of the reqdn command.

Examples of the reqdn command		
Example	**Task, response, and explanation**	
reqdn **1234567** ↵ *where*		
1234567	specifies a local directory number	
	Task:	Set the directory number.
	Response:	None
	Explanation:	This command identifies the directory number returned by the simulator in response messages as 1234567.
-continued-		

reqdn (end)

Examples of the reqdn command (continued)		
Example	**Task, response, and explanation**	
reqdn **2121234567** ↵ *where*		
2121234567	specifies a non-local directory number	
	Task:	Set the directory number.
	Response:	None
	Explanation:	This command identifies the directory number returned by the simulator in response messages as 2121234567.
End		

Responses

The following table provides explanations of the responses to the reqdn command.

Responses for the reqdn command	
MAP output	**Meaning and action**
REQUESTED NUMBER INVALID	
	Meaning: You entered invalid parameters.
	Action: Reenter a valid number.
REQUESTED NUMBER IS WRONG LENGTH	
	Meaning: You entered the area code for a local directory number or failed to enter the area code for a non-local directory number.
	Action: Reenter the appropriate number of digits for the requested number.

rfpdata

Function

Use the rfpdata command to specify a function and release code with an event number. The function and release code specified is sent in the relay from position message when it is present in a given call scenario.

rfpdata command parameters and variables			
Command	**Parameters and variables**		
rfpdata	*syntax* dispall display modify	 *event_no* *event_no*	 *function*	 *release_code*

Parameters and variables	**Description**
syntax	Omitting this entry forces the system to default to displaying the syntax for this command.
dispall	This parameter displays all the events and their associated values for the scenario.
display	This parameter shows the specified event.
event_no	This variable indicates the event number. The valid entry range is 0-19.
function	This variable represents the function specification. The valid entry values are nil or rlsrqst.
modify	This parameter allows you to specify the function and release code to be associated with a particular event.
release_code	This variable specifies the release code associated with a particular event. The valid entry range is 0-255. A release code value of 254 indicates position release.

Qualification

The rfpdata command is not available unless the DA simulator has the vendor value set to IBM.

rfpdata (continued)

Examples

The following table provides examples of the rfpdata command.

Examples of the rfpdata command	
Example	**Task, response, and explanation**
rfpdata dispall ↵	
	Task: Display all events.

	Response: IBM RELAY FROM POSITION DATA:
	
	EVENT# = 0 FUNCTION = NIL DATA = 255
	EVENT# = 1 FUNCTION = RLSRQST DATA = 1
	EVENT# = 2 FUNCTION = RLSRQST DATA = 0
	  .
	  .
	  .
	EVENT# = 18 FUNCTION = NIL DATA = 255
	EVENT# = 19 FUNCTION = NIL DATA = 255

	Explanation: This command displays all events and their associated values.
rfpdata display 1 ↵ *where* 1 specifies the event number	
	Task: Display a specific event.
	Response: EVENT # = 1 FUNCTION = RLSRQST DATA = 1
	Explanation: This response displays the rfpdata information for event 1.
-continued-	

Examples of the rfpdata command (continued)

Example	Task, response, and explanation

rfpdata modify 3 rlsrqst 254 ↵
where

3	specifies the event number
rfsrqst	specifies the function
254	specifies the release code

> **Task:** Modify an event.
>
> **Response:** RFPDATA ENTRY IS PROGRAMMED
>
> **Explanation:** This response indicates that the command executed properly. You specified that when the relayfromposmessage parameter has been defined as event 3 in a given scenario, an rlsrqst will be composed with a scenario of 254 (position release).

End

Responses

The following table provides explanations of the responses to the rfpdata command.

Responses for the rfpdata command

MAP output	Meaning and action

EITHER incorrect optional parameter(s) OR too many parameters.
ERROR IN INPUT STRING: <position_number>

> **Meaning:** You entered an invalid parameter or too many parameters.
>
> **Action:** Check the command syntax and reenter the command.

Wrong type

> **Meaning:** You entered an invalid parameter.
>
> **Action:** Enter an appropriate value to continue, or abort to cancel.

rst

Function

Use the rst command to reset all parameters used by the simulator to their initial state. The reset stops all call processing.

rst command	
Command	**Parameters and variables**
rst	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the rst command.

Example of the rst command		
Example	**Task, response, and explanation**	
rst ↵		
	Task:	Reset all parameters and stop call processing.
	Response:	ALL PARAMETERS ARE RESET
	Explanation:	This command resets all parameters to their initial state and stops call processing.

Response

The following table provides an explanation of the response to the rst command.

Response for the rst command	
MAP output	**Meaning and action**
UNDEFINED COMMAND	
	Meaning: The command is not part of the load.
	Action: None

scencci

Function

Use the scencci command to display the available call scenario list.

scencci command parameters and variables	
Command	**Parameters and variables**

scencci	dispall
	display *current no* / *index_no*
	modify_descript *descript*
	modify_event *event_no* *event* *time_value* *time_units*
	select *index_no*

Parameters and variables	**Description**
current no	Omitting this entry forces the system to default to displaying the current internal scenario list.
descript	This variable specifies a descriptive character string. If the string contains spaces you must enclose the string in single quotes ('').
dispall	This parameter displays all the indices and their associated values for the scenario.
display	This parameter shows the specified internal scenario list.
event	This variable indicates the name of the event. The valid entry values are: amatransfer aruconnect arurequest auditrequest auditreply callbegin callend callfloat callstatus ccaruconnect completecall extendarureq nil posconnect posdisconnect posrelease posreply posrequest posstatus relayfrompos srvrequest
	-continued-

scencci (continued)

scencci command parameters and variables (continued)	
Parameters and variables	**Description**
event_no	This variable indicates the event number. The valid entry range is 0-19.
index_no	This variable indicates the index number. The valid entry range is 0-23.
modify_descript	This parameter modifies the description of a selected scenario.
modify_event	This parameter modifies the description of selected time value for a selected scenario. To create a new scenario, this command is used once for each event or message the scenario is to send or receive.
select	This parameter specifies the scenario to run.
time_units	This variable indicates the units of time. The valid entry values are: tenms secs mins hrs aeons
time_value	This variable indicates the amount of time. The valid entry range is 1-255.
End	

Qualification

The scencci command is not available unless the DA simulator has the vendor value set to CCI.

Examples

The following table provides examples of the scencci command.

Examples of the scencci command	
Example	**Task, response, and explanation**
scencci dispall ↵	
	Task: Show the available internal scenario list.
	Response:
-continued-	

Examples of the scencci command (continued)

Example   Task, response, and explanation

```
SCENARIO DATA:

SCENARIO INDEX: 0  SCENARIO DESCRIPTION: AUTO QUOTE DA CALL
  MESSAGES:   TIME:

     CALLBEGI   1 AEONS
     ARUREQUE   2 SECS
     ARUCONNE   2 SECS
     CALLEND    1 AEONS


SCENARIO INDEX: 1  SCENARIO DESCRIPTION: AUTO QUOTE AUTO-INTERCEPT CALL
  MESSAGES:   TIME:

     CALLBEGI   1 AEONS
     ARUREQUE   2 SECS
     ARUCONNE   2 SECS
     CALLEND    1 AEONS


SCENARIO INDEX: 2  SCENARIO DESCRIPTION: AUTO QUOTE ONI-INTERCEPT CALL
  MESSAGES:   TIME:

     CALLBEGI   1 AEONS
     CALLFLOA   2 SECS
     POSDISCO   2 SECS
     ARUREQUE   2 SECS
     ARUCONNE   2 SECS
     CALLEND    1 AEONS


SCENARIO INDEX: 3 SCENARIO DESCRIPTION: AUTO QUOTE MULTIPLE REQ DA CALL
  MESSAGES:   TIME:

     CALLBEGI   1 AEONS
     AMATRANS   2 SECS
     CALLSTAT   2 SECS
     ARUREQUE   2 SECS
     ARUCONNE   2 SECS
     CALLEND    1 AEONS
```

-continued-

scencci (continued)

Examples of the scencci command (continued)

Example	Task, response, and explanation

```
SCENARIO INDEX: 4  SCENARIO DESCRIPTION: AUTO QUOTE WITH RECALL DA CALL
  MESSAGES:   TIME:

     CALLBEGI   1 AEONS
     ARUREQUE   2 SECS
     ARUCONNE   2 SECS
     POSCONNE   2 SECS
     ARUREQUE   2 SECS
     ARUCONNE   2 SECS
     CALLEND    1 AEONS


SCENARIO INDEX: 5  SCENARIO DESCRIPTION: VOICE QUOTE MEMORY CALL
  MESSAGES:   TIME:

     CALLBEGI   1 AEONS
     CALLEND    1 AEONS


SCENARIO INDEX: 6  SCENARIO DESCRIPTION: VOICE QUOTE DA CALL
  MESSAGES:   TIME:

     CALLBEGI   1 AEONS
     POSRELEA   2 SECS
     CALLEND    1 AEONS


SCENARIO INDEX: 7  SCENARIO DESCRIPTION: VOICE QUOTE AUTO-INTERCEPT CALL
  MESSAGES:   TIME:

     CALLBEGI   1 AEONS
     POSREQUE   2 SECS
     POSCONNE   2 SECS
     POSRELEA   2 SECS
     CALLEND    1 AEONS


SCENARIO INDEX: 8  SCENARIO DESCRIPTION: VOICE QUOTE AUTO-INTER, NO ARU
  MESSAGES:   TIME:

     CALLBEGI   1 AEONS
     ARUREQUE   2 SECS
     POSCONNE   2 SECS
     POSRELEA   2 SECS
     CALLEND    1 AEONS
```

Explanation: This command displays all available scenario lists.

-continued-

---

scencci (continued)

Examples of the scencci command (continued)

Example   **Task, response, and explanation**

scencci   display 0 ↵
where

0   specifies the index number

> **Task:**   Display the data of a particular scenario.
>
> **Response:**
> ```
> SCENARIO INDEX: 0  SCENARIO DESCRIPTION: AUTO QUOTE DA CALL
>   MESSAGES:   TIME:
>
>        CALLBEGI   1 AEONS
>        ARUREQUE   2 SECS
>        ARUCONNE   2 SECS
>        CALLEND    1 AEONS
> ```
>
> **Explanation:**   This command displays the scenario for index 0.

scencci   select 3 ↵
where

3   specifies the index number

> **Task:**   Specify a scenario to run.
>
> **Response:**   None
>
> **Explanation:**   This command specifies the scenario to run.

scencci   modify_descript 'voice quote memory call' ↵
where
'voice quote memory call'   specifies the description

> **Task:**   Modify the description of a selected scenario.
>
> **Response:**   None
>
> **Explanation:**   This command modifies the description of the current scenario to 'voice quote memory call'.

-continued-

scencci (continued)

Examples of the scencci command (continued)	
Example	**Task, response, and explanation**

scencci modify_event 3 callbegin 2 secs ↵ *where* 3 specifies the event number callbegin specifies the event 2 specifies the time value secs specifies the time units	

	Task:	Modify an event.
	Response:	None
	Explanation:	This command adds the event to the current scenario.

scencci modify_event 0 srvrequest 1 aeons ↵ *where* 0 specifies the event number srvrequest specifies the event 1 specifies the time value aeons specifies the time units	

	Task:	Modify an event.
	Response:	None
	Explanation:	This command adds the service request message to the current scenario at event 0 and specifies the time (1 aeons) to be used.

End

Responses

The following table provides explanations of the responses to the scencci command.

Responses for the scencci command	
MAP output	**Meaning and action**
`INVALID SYMBOL`	
	Meaning: You entered an invalid parameter.
	Action: Enter the appropriate parameter to continue, or abort to cancel.
-continued-	

scencci (end)

Responses for the scencci command (continued)	
MAP output	**Meaning and action**
UNDEFINED COMMAND	
	Meaning: You entered the command incorrectly.
	Action: Reenter the command correctly.
WRONG TYPE	
	Meaning: You entered alphabetic characters in a numeric field.
	Action: Enter the appropriate value to continue, or abort to cancel.
End	

scenibm

Function

Use the scenibm command to display the available call scenario list.

scenibm command parameters and variables	
Command	**Parameters and variables**
scenibm	*syntax* dispall display ⎡ *current no* ⎤ ⎣ *index_no* ⎦ modify_descript *descript* modify_event *event_no* *event* *time_value* *time_units* *bill* select *index_no*

Parameters and variables	Description
current no	Omitting this entry forces the system to default to displaying the current internal scenario list.
syntax	Omitting this entry forces the system to default to displaying the syntax for this command.
bill	This variable specifies if a billing appendage is to be applied. The valid entry values are yes and no.
descript	This variable specifies a descriptive character string. If the string contains spaces, you must enclose the string in single quotes (').
dispall	This parameter displays all the indices and their associated values for the scenario.
display	This parameter shows the specified internal scenario list.
-continued-	

scenibm (continued)

scenibm command parameters and variables (continued)	
Parameters and variables	**Description**
event	This variable indicates the name of the event. The valid entry values are: aruselect reqaruststus arustatus nil4 posstatus posbusyin subscribedisconnect transferabort posrelease transfer immedtransfer transfercancel reqposstatus callarrival referral delaytransfer nil17 sysidresponse sysidreq speak speakcomplete speakreject newreq nil24 cctransfer ccaruselect completecall callrelease detexception nil30 amaupdate nil32 nil33 nil34 relayfrompos
event_no	This variable indicates the event number. The valid entry range is 0-19.
index_no	This variable indicates the index number. The valid entry range is 0-23.
modify_descript	This parameter modifies the description of a selected scenario.
-continued-	

scenibm (continued)

scenibm command parameters and variables (continued)	
Parameters and variables	**Description**
modify_event	This parameter modifies the description of selected time value for a selected scenario. To create a new scenario, this parameter is used with the scenibm command once for each event or message the scenario is to send or to receive.
select	This parameter specifies the scenario to run.
time_units	This variable indicates the units of time. The valid entry values are: tenms secs mins hrs aeons
time_value	This variable indicates the amount of time. The valid entry range is 1-255.
End	

Qualification

The scenibm command is not available unless the DA simulator has the vendor value set to IBM.

Examples

The following table provides examples of the scenibm command.

scenibm (continued)

Examples of the scenibm command
Example   **Task, response, and explanation**

scenibm   **display**   **18** ↵
where

18   specifies the index number

> **Task:**   Display the data of a particular scenario.
>
> **Response:**
> ```
> SCENARIO INDEX: 18   SCENARIO DESCRIPTION: VQUOTE, SERV
> CHANGE,  DA AND TA
>   MESSAGES:   TIME:
>
>         POSBUSYI   1 AEONS
>         RELAYFRO   2 SECS
>         RELAYFRO   2 SECS
>         SUBSCRIB   1 AEONS
> ```
>
> **Explanation:**   This example shows the response to the scenibm display
> command string when scenario 18 is specified. Scenario 18 is
> automatically implemented with the DA simulator. It will perform
> the service changes described in the scenario description when the
> rfpdata command has been used to specify service 1 and 0 for
> events 1 and 2 respectively.

-continued-

Examples of the scenibm command (continued)

Example   **Task, response, and explanation**

scenibm   display 19 ↵
where

19   specifies the index number

> **Task:**   Display the data of a particular scenario.
>
> **Response:**
> ```
> SCENARIO INDEX: 19  SCENARIO DESCRIPTION: VQUOTE, POS RLS VIA
> RELAY
>   MESSAGES:   TIME:
>
>        POSBUSYI   1 AEONS
>        RELAYFRO   2 SECS
>        SUBSCRIB   1 AEONS
> ```
>
> **Explanation:**   This example shows the response to the scenibm display
>   command when scenario 19 is specified. Scenario 19 is
>   automatically implemented with the DA simulator. It will perform a
>   positon busy in followed by a position release, whe the rfpdata
>   command has been used ot specify position release (254) for event
>   1.

scenibm   modify_event 0 relayfrompos 1 aeons ↵
where
o   specifies the event number
relayfrompos specifies the event
1   specifies the time value
aeons   specifies the time units

> **Task:**   Modify an event.
>
> **Response:**   None
>
> **Explanation:**   This command adds the relay from position message to the
>   scenario at event 0 and specifies the time (1 aeons) to be used.
>   IBM is selected as the DA vendor.

End

Responses

The following table provides explanations of the responses to the scenibm
command.

scenibm (end)

Responses for the scenibm command	
MAP output	**Meaning and action**
INVALID SYMBOL	
	Meaning: You entered an invalid parameter.
	Action: Enter the appropriate parameter to continue, or abort to cancel.
UNDEFINED COMMAND	
	Meaning: You entered the command incorrectly.
	Action: Reenter the command correctly.
WRONG TYPE	
	Meaning: You entered alphabetic characters in a numeric field.
	Action: Enter the appropriate value to continue, or abort to cancel.

servnum

Function

Use the servnum command to modify the service number associated with SRVRQST.

servnum command parameters and variables		
Command	**Parameters and variables**	
servnum	dispall	
	display	*event_no*
	modify	*event_no* *service_no*

Parameters and variables	**Description**
dispall	This parameter displays all the events and their associated values for the scenario.
display	This parameter shows the specified event.
event_no	This variable indicates the event number. The valid entry range is 0-19.
modify	This parameter modifies the specified event and service number.
service_no	This variable specifies the service number to be assoicated with a particular event. The valid entry range is 0-255. The DMS only has service numbers of 0-62 available.

Qualification

The servnum command is not available unless the DA simulator has the vendor value set to CCI.

servnum (continued)

Examples

The following table provides examples of the servnum command.

Examples of the servnum command	
Example	**Task, response, and explanation**
servnum dispall ↵	
	Task: Display all events.
	Response: `CCI SERVICE REQUEST DATA:`

	`EVENT#: 0 REQUEST SERVICE: 255`
	`EVENT#: 1 REQUEST SERVICE: 255`
	`.`
	`.`
	`.`
	`EVENT#: 18 REQUEST SERVICE: 255`
	`EVENT#: 19 REQUEST SERVICE: 255`

	Explanation: This command displays all events and their associated values.
servnum display 5 ↵ *where* 5 specifies the event number	
	Task: Display a specific event.
	Response: `EVENT #: 5 REQUEST SERVICE: 0`
	Explanation: This command displays the event number 5.
servnum modify 5 0 ↵ *where* 5 specifies the event number 0 specifies the service number	
	Task: Modify an event.
	Response: `SERVNUM ENTRY IS PROGRAMMED`
	Explanation: This command modifies the event number 5 to a service number value of zero.

Responses

The following table provides explanations of the responses to the servnum command.

Responses for the servnum command	
MAP output	**Meaning and action**
EITHER incorrect optional parameter(s) OR too many parameters. ERROR IN INPUT STRING: <position_number>	
	Meaning: You entered an invalid parameter or too many parameters.
	Action: Check the command syntax and reenter the command.
Wrong type	
	Meaning: You entered an invalid parameter.
	Action: Enter an appropriate value to continue, or abort to cancel.

setlink

Function

Use the setlink command to activate the data link messaging mode.

setlink command parameters and variables	
Command	**Parameters and variables**
setlink	fail mpc stub

Parameters and variables	Description
fail	This parameter activates the message mode to simulate failure to receive messages.
mpc	This parameter activates the message mode to simulate sending outgoing messages over the multi-protocol controller (MPC) data link and looping them back.
stub	This parameter activates the message mode to simulate sending outgoing messages directly to TOPS call processing without going through the data link.

Qualifications

None

Example

The following table provides an example of the setlink command.

Example of the setlink command	
Example	**Task, response, and explanation**
setlink fail ↵	
	Task: Set simulation to fail.
	Response: `LINK IS SET TO:   FAIL`
	Explanation: This command sets the link to fail. The simulation appears to not receive messages.

setlink (end)

Responses

The following table provides explanations of the responses to the setlink command.

Responses for the setlink command	
MAP output	**Meaning and action**
`INVALID SYMBOL`	
	Meaning: You entered an invalid message mode.
	Action: Enter the correct message mode.
`LINK ALREADY SET TO <setting>`	
	Meaning: You already set the link to this setting.
	Action: None

sim

Function

Use the sim command to enable or disable the simulator.

sim command parameters and variables	
Command	**Parameters and variables**
sim	off on
Parameters and variables	**Description**
off	This parameter disables the simulator from receiving or processing messages.
on	This parameter enables the simulator to receive and process messages.

Qualifications

None

Example

The following table provides an example of the sim command.

Example of the sim command		
Example	**Task, response, and explanation**	
sim on ↵		
	Task:	Enable the simulator.
	Response:	SIMULATOR IS TURNED ON
	Explanation:	This command enables the simulator to receive and process messages.

sim (end)

Responses

The following table provides explanations of the responses to the sim command.

Responses for the sim command	
MAP output	**Meaning and action**
INVALID SYMBOL	
	Meaning: You entered an invalid parameter.
	Action: Reenter the command using a valid parameter.
UNDEFINED COMMAND	
	Meaning: The sim command is not part of the load.
	Action: None

trace

Function

Use the trace command to activate and deactivate the saving of all messages.

trace command parameters and variables	
Command	**Parameters and variables**
trace	off
	on
Parameters and variables	**Description**
off	This parameter stops saving messages that the simulator sends and receives.
on	This parameter saves messages that the simulator sends and receives.

Qualifications

None

Examples

The following table provides examples of the trace command.

Examples of the trace command		
Example	**Task, response, and explanation**	
trace on ↵		
	Task:	Activate the trace.
	Response:	TRACING IS ACTIVATED
	Explanation:	This command activates the saving of messages.
trace off ↵		
	Task:	Deactivate the trace.
	Response:	TRACING IS DEACTIVATED
	Explanation:	This command deactivates the saving of messages.

trace (end)

Response

The following table provides an explanation of the response to the trace command.

Response for the trace command	
MAP output	**Meaning and action**
`TRACING ALREADY <setting>`	
	Meaning: You already set tracing to this setting.
	Action: None

vendor

Function

Use the vendor command to specify which vendor to simulate.

vendor command parameters and variables	
Command	**Parameters and variables**
vendor	cci ibm lockheed none

Parameters and variables	**Description**
cci	This parameter simulates the Computer Consoles Incorporated (CCI) vendor.
ibm	This parameter simulates the International Business Machines (IBM) vendor.
lockheed	This parameter simulates the Lockheed vendor.
none	This parameter performs no simulation.

Qualifications

None

Example

The following table provides an example of the vendor command.

Example of the vendor command	
Example	**Task, response, and explanation**
vendor cci ↵	
	Task: Specify the vendor to simulate.
	Response: VENDOR IS SET TO: CCI
	Explanation: This command specifies to simulate CCI.

vendor (end)

Responses

The following table provides explanations of the responses to the vendor command.

Responses for the vendor command	
MAP output	**Meaning and action**
`INVALID SYMBOL`	
	Meaning: You entered the vendor name incorrectly.
	Action: Reenter the correct vendor name.

`UNDEFINED COMMAND`
`or`
`NO COMMAND IN LINE`	

	Meaning: You entered the name of the vendor without the preceding vendor command.
	Action: Reenter the command correctly.

DBUT level commands

Use the DBUT level of the MAP to backup and restore databases. The backup remembers all states of the database and retains all useful information of caches, log, database, and files of the database to a bookkeeping file. If the original state of the database is not in service (NIS) and there is a download, restore, or recovery in process, the database backup is not attempted.

⚠ **CAUTION**
Risk of data corruption
The setnode command must be done before any other commands in the DBUT directory can be issued.

You are prompted to do a setnode command if you violate the caution.

Accessing the DBUT level

To access the DBUT level, enter the following command from the CI level:

> **dbut** ↵

DBUT commands

The commands available at the DBUT MAP level are described in this chapter and are arranged in alphabetical order. The page number for each command is listed in the following table.

DBUT commands	
Command	**Page**
backupdb	D-79
backuplog	D-93
cancel	D-105
dbstatus	D-109
-continued-	

DBUT commands (continued)	
Command	**Page**
help	D-113
quit	D-115
restoredb	D-119
setnode	D-129
End	

backupdb

Function

Use the backupdb command to backup a database.

backupdb command parameters and variables							
Command	**Parameters and variables**						
backupdb	*dbname*	*dbinstance*	*destination*			*dbname.bak*	(1)
				ba		*backupname*	(2)
backupdb (continued)	(1) (2)	bdb	*.dbname* master *dbinstance* *backupdbname*		bsf	*.bak* suffix	(1) (2)
backupdb (continued)	(1) (2)	log	y n	dl	y n	wait	y n (end)

Parameters and variables	**Description**
.bak	Omitting this entry forces the system to default to using .bak for the file type extension.
dbname master *dbinstance*	Omitting this entry forces the system to default to naming the backup log from the original database name, __master__, and database instance. If the database instance has two digits, the name of the backup log is the original database name, __master_, and database instance.
ba	This parameter indicates the name of the database backup.
backupdbname	This variable specifies the output name for the database log backup. The database log backup name is limited to 32 characters.
backupname	This variable specifies the output name for the database backup. The database backup name is limited to 21 characters.
bdb	This parameter indicates the name of the database log backup.
bsf	This parameter indicates the suffix for the backup log file.
dbinstance	This variable specifies the database instance.
dbname	This variable specifies the name of the database to backup.
-continued-	

backupdb (continued)

backupdb command parameters and variables (continued)	
Parameters and variables	**Description**
dbname.bak	Omitting this entry forces the system to default to naming the backup database from the original database name and the .bak suffix.
destination	This variable specifies the destination for the backup. The destination is limited to 80 characters.
dl	This parameter indicates the delete log files flag.
log	This parameter indicates whether to backup the log of the database after the backup of the database.
n	This parameter does not backup the log of the database after the backup of the database when the log parameter is indicated.
	This parameter keeps the log files when the dl parameter is indicated.
	This parameter does not wait nor display progress messages when the wait parameter is indicated.
suffix	This variable specifies the output file type extension. The suffix is limited to three characters.
wait	This parameter indicates the wait flag.
y	This parameter backs up the log of the database after the backup of the database when the log parameter is indicated. Omitting this entry forces the system to default to backing up the log of the database after the backup of the database when the log parameter is indicated.
	This parameter deletes the log files when the dl parameter is indicated and log is yes. Omitting this entry forces the system to default to deleting the log files when the dl parameter is indicated and log is yes.
	This parameter waits and displays progress messages when the wait parameter is indicated. Omitting this entry forces the system to default to waiting until the backup is complete and display the progress as each ten percent is completed when the wait parameter is indicated.
End	

Qualifications

The backupdb command is qualified by the following exceptions, restrictions, and limitations:

- No other read or write operations are allowed during the execution of the backup on the database. Offline backup ensures consistency of the backup.

backupdb (continued)

- Upon completion, the command leaves database files opened for two minutes to allow the application to open the file and use the caches populated by the command. You can not manual busy the disk or stop the shadowset during these two minutes, because the files are not closed until the two minutes have expired.

Examples

The following table provides examples of the backupdb command.

Examples of the backupdb command	
Example	**Task, response, and explanation**
backupdb 800plus 0 fp00ct00 ↵ *where* 800plus specifies the database name 0 specifies the database instance fp00ct00 specifies the backup destination	
	Task: Backup a database and log. **Response:** Not currently available **Explanation:** You backed up the database to the file fp00ct00 under the default backup name 800plus__master__0.bak. The default log and delete log file options backup and delete the log files and the system displays progress messages.
backupdb accs 0 fp00ct00 bdb accsdb__jun10 log n wait y ↵ *where* accs specifies the database name 0 specifies the database instance fp00ct00 specifies the backup destination accslog__jun10 specifies the backup name n specifies the log state y specifies the wait state	
	Task: Backup a database to a specified file without backing up the log. **Response:** Not currently available **Explanation:** You backed up the database accs to the file fp00ct00 under the backup name accsdb__jun10. The log file is not backed up and the system displays progress messages.
-continued-	

backupdb (continued)

Examples of the backupdb command (continued)	
Example	**Task, response, and explanation**

backupdb 800plus 0 :/fp00dk01800pdb wait n ↵
where

800plus	specifies the database name
0	specifies the database instance
:/fp00dk01800pdb	specifies the backup destination
n	specifies the wait state

Task:	Backup a database to a directory path name.
Response:	Not currently available
Explanation:	You backed up the database and logs for 800plus to the directory path name :/fp00dk01800pdb under the backup log name of 800plus__master__0.bak. The backed up log files are deleted and the system does not display progress messages.

backupdb 800plus 0 :/ss00800pdb bdb '800plus.0201' ↵
where

800plus	specifies the database name
0	specifies the database instance
:/ss00800pdb	specifies the backup destination
'800plus.0201'	specifies the backup database name

Task:	Backup a database to a shadow set.
Response:	Not currently available
Explanation:	You backed up the database and logs to a shadow set named :/ss00 in the volume 800pdb under the backup database name of 800plus.0201. The backed up log files are deleted and the system displays progress messages.

-continued-

backupdb (continued)

Examples of the backupdb command (continued)	
Example	**Task, response, and explanation**
backupdb 800plus 0 :/fp00dk01800pdb log y ↵ *where* 800plus specifies the database name 0 specifies the database instance :/fp00dk01800pdb specifies the backup destination y specifies the log backup state	
	Task: Backup a database and log to a disk and volume.
	Response: Not currently available
	Explanation: You backed up the database and logs to a disk named :/fp00dk01 in the volume 800pdb under the backup database name of 800plus__master__0.bak. The backed up log files are deleted and the system displays progress messages.
backupdb 800plus 0 fp03ct10 log y dl n ↵ *where* 800plus specifies the database name 0 specifies the database instance fp03ct10 specifies the backup destination y specifies the backup log state n specifies the delete log state	
	Task: Backup a database and log to tape.
	Response: Not currently available
	Explanation: You backed up the database and logs to a tape named fp03ct10 under the backup database name of 800plus__master__0.bak. The backed up log files are not deleted and the system displays progress messages.
End	

Responses

The following table provides explanations of the responses to the backupdb command.

backupdb (continued)

Responses for the backupdb command (continued)
MAP output   **Meaning and action**

A database backup already in progress!

Meaning:	You attempted to run more than one backup or restore at a time. The command aborts.
Action:	Wait until the backup is completed and reenter the command. You can check the progress using the dbstatus command.

A database restore already in progress!

Meaning:	You attempted to run more than one backup or restore at a time. The command aborts.
Action:	Wait until the restore is completed and reenter the command. You can check the progress using the dbstatus command.

Backup database Name already in use.

Meaning:	You specified a backup database name that already exists on the output device. The command aborts.
Action:	Reenter the command using another database name or another output device.

Backup DB Name too long!

Meaning:	You specified a backup database of more than 32 characters. The command aborts.
Action:	Reenter the command using a shorter backup database name.

BFM file management system error.
(or) FTFS Err: <...>
(or) FS Err: <..>
    which has not been listed in previous responses.

Meaning:	The system encountered an error at the file management level. The command aborts.
Action:	Get help from an expert on the transaction record management system (TRMS) system.

-continued-

backupdb (continued)

Responses for the backupdb command (continued)	
MAP output	**Meaning and action**

`Completion msg received`
`Msg Time: <date> <hour: minute: second: msec>;`
`Report: Backup of log for <dbname> is Completed.`	

	Meaning: The backup process has completed. The Msg Time indicates the time of completion.
	Action: None
`DAT Tape device in use by other user!`	
	Meaning: You specified a tape device that is already in use. The command aborts.
	Action: Free up the required output device or select a different output device.
`Database Name Too Long!`	
	Meaning: You specified a database name with more than 21 characters. The command aborts.
	Action: Reenter the command using a correct database name.
`Database not created!`	
	Meaning: You specified a database that is not created. The command aborts.
	Action: Reenter the command using a created database name.
`Database under other operation!`	
	Meaning: You specified a database that is under some operation making it unavailable for backup. The command aborts.
	Action: If possible, check to see what is going on with the database. Reenter the command later.
`Destination device does not exist.`	
	Meaning: You specified a device that does not exist. The command aborts.
	Action: Reenter the command using a valid device name.
-continued-	

backupdb (continued)

Responses for the backupdb command (continued)	
MAP output	**Meaning and action**
`Device is not ready`	
	Meaning: You specified a device that exists but is not ready. The device may possibly be turned off. The command aborts.
	Action: Check the device status on the MAPCI, return the device to service, and reenter the command.
`Device Unavailable.`	
	Meaning: You specified a device that has gone out of service. The command aborts.
	Action: Check the device status on the MAPCI, return the device to service, and reenter the command.
`DMS system error, possibly from SOS.`	
	Meaning: The system encountered an error, possibly a software error at the support operating system (SOS) level. The command aborts.
	Action: Get help from an expert on the TRMS system.
`Dumping is in progress.`	
	Meaning: The system encountered an error, possibly trying to unprotect some store when dumping is in progress. The command aborts.
	Action: Wait for the dumping to complete. If the retry fails, get help from an expert on the TRMS system. Use a volume with no duplicated file names or clear the backup volume.
`Error : Bookkeeping file <filename> already exists`	
	Meaning: If the backup is to disk, the bookkeeping file is found to exist on the destination disk. If backup is to digital audio tape (DAT), the bookkeeping file is found on the source disk volume. The command aborts.
	Action: Clear the offending file and reenter the command.
-continued-	

backupdb (continued)

Responses for the backupdb command (continued)	
MAP output	**Meaning and action**
`Error : Database state not suitable for Backup.`	
	Meaning: The database state is in NIS_Recovery or NIS_Restore. A backup is not allowed. The command aborts.
	Action: The database state may change if it was under recovery. If the database was in restore state, it means a previous restore was not successful, and the database is not a good one to be backed up.

`Error: Device is out of service.`
`(or) Error: FS Err - Device is out of service.`
`Completion msg received`
`Msg Time: <date> <hr: min: sec: ms>;`
`Error : Error in backup of Database <dbname>.`	

	Meaning: You specified a destination device that is out of service. The command aborts.
	Action: Bring the device back into service and reenter the command.
`Error : File <filename> already exists on disk.` `Error : FTFS Err - Duplicate file name`	
	Meaning: A file with the same name as one of the files to be backed up already exists on the backup volume. The command aborts with all created files erased.
	Action: Use a volume with no duplicated file names. You may clear the backup volume or simply erase the duplicated file on the destination device.
`Error : File <filename> created on destination deleted`	
	Meaning: A file created on the destination has been deleted. This is part of the clean up on failure.
	Action: None
-continued-	

backupdb (continued)

Responses for the backupdb command (continued)

MAP output   Meaning and action

```
Error : TRMS log file <fname> cannot be found!
```

Meaning: A log file expected to exist is not found. This can mean that a previous backup of log has been killed before it can update the ArchiveNmb and LogTailAddress of the log so that these numbers indicate an older log address than what has been backed up and deleted. It is not treated as an error.

Action: Take note of the incidence. You may check the number of files opened by using the DISKUT directory commands.

```
Error : Volume not mounted
Completion msg received
Msg Time: <date> <hr: min: sec: ms>;
Error : Error in backup of Database <dbname>.
```

Meaning: The DAT tape is not mounted. The command aborts.

Action: Mount the DAT tape and reenter the command.

```
Error : Wait for files to be closed by previous backup.
```

Meaning: After a database is backed up, the backup process waits for some time before it closes the database files. You have the chance to open them before they are closed. A second backup or restore is not permitted until the previous backup has closed all these files, which may take a few minutes. The command aborts.

Action: Wait for a few minutes and reenter the command. You may check the number of files opened by using the DISKUT directory commands.

```
Error in Backup of TRMS file <fname> : <error id>
```

Meaning: You encountered a problem in backing up the particular TRMS file. The command aborts with back out. More information about the error may display.

Action: If more information is displayed, you may be able to recover. If not, get help from an expert on the TRMS system.

-continued-

backupdb (continued)

Responses for the backupdb command (continued)

MAP output	Meaning and action

`Messaging System problem.`

> **Meaning:** The system encountered an error at the messaging system. The command aborts.
>
> **Action:** Get help from an expert on the TRMS system.

`Msg Time: <time>`
`Report : Database <dbname> is Backed up`

> **Meaning:** You successfully completed the command.
>
> **Action:** None

`No Master Entry - DB not defined!`

> **Meaning:** You specified a database that is not defined. The command aborts.
>
> **Action:** Reenter the command using a database that is defined.

`No more storage space on output medium.`

> **Meaning:** You specified an output device that is full. The command aborts and the backup on the disk is erased.
>
> **Action:** Check the size of the backup and specify an output device that has enough storage.

`Path Name too Long!`

> **Meaning:** You specified a destination path of more than 80 characters. The command aborts.
>
> **Action:** Reenter the command using a correct path name.

`Report : Backing up system log:`

> **Meaning:** The system log backup is started.
>
> **Action:** Do nothing until the backup is completed or a failure occurs.

-continued-

---

backupdb (continued)

Responses for the backupdb command (continued)	
MAP output	**Meaning and action**
`Report : Backup for <dbname> is Completed.`	
	Meaning: All files of the database are backed up.
	Action: Do nothing until the backup is completed or a failure occurs.
`Report : Destination Device is OK.`	
	Meaning: The backup process is started.
	Action: Do nothing until the backup is completed or a failure occurs.
`Report : No log file needs to be backed up.`	
	Meaning: There is no log file in the database or system log that needs backed up.
	Action: Do nothing until the backup is completed or a failure occurs.
`Report : TRMS file <filename> is Backed up`	
	Meaning: The file is backed up.
	Action: Do nothing until the backup is completed or a failure occurs.
`Report : TRMS log file <filename> is Backed up`	
	Meaning: The log file is backed up.
	Action: Do nothing until the backup is completed or a failure occurs.
`STARTTIME <date> <hour: minute: second>;`	
	Meaning: The backup process has started.
	Action: If the wait option is issued, do nothing until the backup is completed or a failure occurs. If the no wait option is issued, use the dbstatus command to find out the status of the backup in regular intervals.
-continued-	

backupdb (end)

Responses for the backupdb command (continued)
MAP output   Meaning and action

| ```
StoreRC error (from STOR) : <number>
T_MTS_RC error (from MTSKERN) : <number>
MBRC error (from MESSAGES) : <number>
FlagRC error (from FLAGS) : <number>
SemaRC error (from SEMAS) : <number>
CIRetCode error (from CITYPES) : <number>
``` |
| **Meaning:** The system encountered an error at the corresponding system. The command aborts. |
| **Action:**   Get help from an expert on the TRMS system. |
| ```
Suffix Name too long!
``` |
| **Meaning:** You specified a suffix of more than three characters. The command aborts. |
| **Action:**   Reenter the command using a shorter suffix. |
| ```
TRMS system error.
``` |
| **Meaning:** The system encountered an error, possibly a software error in the transaction record management system. The command aborts. |
| **Action:**   Get help from an expert on the TRMS system. |
| ```
Volume not mounted.
``` |
| **Meaning:** You specified a DAT device that is not mounted. The command aborts. |
| **Action:**   Mount the DAT by putting the DAT into the device slot. |
| ```
Waiting for report Messages:
Warning - this may take some time!
``` |
| **Meaning:** Wait for more report messages to come from backup or restore. |
| **Action:**   Do nothing until the backup is completed or a failure occurs. |
| **End** |

backuplog

Function

Use the backuplog command to backup the transaction log for a database.

backuplog command parameters and variables							
Command	**Parameters and variables**						

backuplog	*dbname*	*dbinstance*	*destination*	bdb	*dbname* master *dbinstance*		(1)
					backupdbname		(2)

backuplog (continued)	(1) (2)	bsf	*.bak* *suffix*	nf	*all* *numfiles*	dl	*y* n	(1) (2)

backuplog (continued)	(1) (2)	wait	*y* n				(end)

Parameters and variables	**Description**
all	Omitting this entry forces the system to default to backing up all of the database files available.
.bak	Omitting this entry forces the system to default to using .bak for the file type extension.
dbname master *dbinstance*	Omitting this entry forces the system to default to naming the backup from the original database name, __master__, and database instance. If the database instance has two digits, the name of the backup is the original database name, __master_, and database instance.
backupdbname	This variable specifies the output name for the backup file. The backup database name is limited to 32 characters.
bdb	This parameter indicates the backup database is specified.
bsf	This parameter indicates the backup suffix for the file.
dbinstance	This variable specifies the database instance.
dbname	This variable specifies the database name for the log backup. The database name is limited to 21 characters.
destination	This variable specifies the destination for the backup. The destination is limited to 80 characters.
-continued-	

backuplog (continued)

backuplog command parameters and variables (continued)	
Parameters and variables	**Description**
dl	This parameter indicates the delete log files flag.
n	This parameter keeps the log files when the dl parameter is indicated.
	This parameter does not wait nor display progress messages when the wait parameter is indicated.
nf	This parameter indicates the number of files.
numfiles	This variable specifies the number of files to backup.
suffix	This variable specifies the output file type extension. The suffix is limited to three characters.
wait	This parameter indicates the wait flag.
y	This parameter deletes the log files when the dl parameter is indicated. Omitting this entry forces the system to default to deleting the log files when the dl parameter is indicated.
	This parameter waits and displays progress messages when the wait parameter is indicated. Omitting this entry forces the system to default to waiting until the backup is complete and displaying the progress as each ten percent is completed when the wait parameter is indicated.
End	

Qualifications

The backuplog command is qualified by the following exceptions, restrictions and limitations:

- If the wait option is chosen, the CI is not available during the backup process.
- If the nowait option is chosen, then you should monitor the status of the backup by using the dbstatus command frequently.

backuplog (continued)

Examples

The following table provides examples of the backuplog command.

Examples of the backuplog command	
Example	**Task, response, and explanation**
backuplog 800plus 0 fp00ct00 ↵ *where* 800plus specifies the database name 0 specifies the database instance fp00ct00 specifies the backup destination	
	Task: Dump all the log files into a backup. **Response:** Not currently available **Explanation:** You dumped all the log files that can be deleted into a file fp00ct00 under the backup log name 800plus.bak. The backed up files are deleted and the system displays progress messages.
backuplog accs 0 fp00ct00 bdb accslog__0101 nf 5 wait n ↵ *where* accs specifies the database name 0 specifies the database instance fp00ct00 specifies the backup destination accslog__0101 specifies the backup log name 5 specifies the number of files n specifies the no wait and no progress messages state	
	Task: Dump up to five log files into a specified log backup. **Response:** Not currently available **Explanation:** You dumped up to five of the log files that can be deleted into a file fp00ct00 under the backup log name accslog__0101.bak. The backed up files are deleted and the system does not display progress messages.
-continued-	

backuplog (continued)

Examples of the backuplog command (continued)
Example   **Task, response, and explanation**

backuplog 800plus 0 :/ss00800bak nf 5 ↵
where

800plus	specifies the database name
0	specifies the database instance
:/ss00800bak	specifies the backup destination
5	specifies the number of files

Task:	Dump up to five log files into a backup shadow set.
Response:	Not currently available
Explanation:	You dumped up to five of the log files that can be deleted into a shadow set named :/ss00 in the volume 800bak under the backup log name of 800plus__master__0.bak. The backed up files are deleted and the system displays progress messages.

backuplog 800plus 0 :/fp00dk01800bak bdb '800pluglog.0201' dl n ↵
where

800plus	specifies the database name
0	specifies the database instance
:/fp00dk01800bak	specifies the backup destination
'800pluslog.0201'	specifies the backup database name
n	specifies the delete log flag

Task:	Dump all log files into a backup without deleting the log.
Response:	Not currently available
Explanation:	You dumped all of the log files that can be deleted into a disk named :/fp00dk01 in the volume 800bak under the backup log name of 800pluslog.0201. The backed up files are not deleted and the system displays progress messages.

End

backuplog (continued)

Responses

The following table provides explanations of the responses to the backuplog command.

Responses for the backuplog command	
MAP output	**Meaning and action**
A database backup already in progress!	
	Meaning: You attempted to run more than one backup or restore at a time. The command aborts.
	Action: Wait until the backup is completed and reenter the command. You can check the progress using the dbstatus command.
A database restore already in progress!	
	Meaning: You attempted to run more than one backup or restore at a time. The command aborts.
	Action: Wait until the restore is completed and reenter the command. You can check the progress using the dbstatus command.
Backup database Name already in use.	
	Meaning: You specified a backup database name that already exists on the output device. The command aborts.
	Action: Reenter the command using another database name or another output device.
Backup Log Name too long!	
	Meaning: You specified a backup database of more than 32 characters. The command aborts.
	Action: Reenter the command using a shorter backup database name.
BFM file management system error.	
	Meaning: The system encountered an error at the file management level. The command aborts.
	Action: Get help from an expert on the transaction record management system (TRMS) system.
-continued-	

backuplog (continued)

Responses for the backuplog command (continued)

MAP output   Meaning and action

```
Completion msg received
Msg Time: <date> <hour: minute: second: msec>;
Report: Backup of log for <dbname> is Completed.
```

Meaning: The backup process has completed. The Msg Time indicates the time of completion.

Action: None

```
DAT Tape device in use by other user!
```

Meaning: You specified a tape device that is already in use. The command aborts.

Action: Free up the required output device or select a different output device.

```
Database Name Too Long!
```

Meaning: You specified a database name with more than 21 characters. The command aborts.

Action: Reenter the command using a correct database name.

```
Database not created!
```

Meaning: You specified a database that is not created. The command aborts.

Action: Reenter the command using a created database name.

```
Database under other operation!
```

Meaning: You specified a database that is under some operation making it unavailable for backup. The command aborts.

Action: If possible, check to see what is going on with the database. Reenter the command later.

```
Destination device does not exist.
```

Meaning: You specified a device that does not exist. The command aborts.

Action: Reenter the command using a valid device name.

-continued-

Responses for the backuplog command (continued)	
MAP output	**Meaning and action**
`Device is not ready`	
	Meaning: You specified a device that exists but is not ready. The device may possibly be turned off. The command aborts.
	Action: Check the device status on the MAPCI, return the device to service, and reenter the command.
`DMS system error, possibly from SOS.`	
	Meaning: The system encountered an error, possibly a software error at the support operating system (SOS) level. The command aborts.
	Action: Get help from an expert on the TRMS system.
`Dumping is in progress.`	
	Meaning: The system encountered an error, possibly while trying to unprotect some store when dumping is in progress. The command aborts.
	Action: Wait for the dumping to complete. If the retry fails, get help from an expert on the TRMS system. Use a volume with no duplicated file names or clear the backup volume.
`Error : Bookkeeping file <filename> already exists`	
	Meaning: If the backup is to disk, the bookkeeping file is found to exist on the destination disk. If backup is to digital audio tape (DAT), the bookkeeping file is found on the source disk volume. The command aborts.
	Action: Clear the offending file and reenter the command.

`Error: Device is out of service.`
`Completion msg received`
`Msg Time: <date> <hr: min: sec: ms>;`
`Error : Error in backup of Database <dbname>.`	

	Meaning: You specified a destination device that is out of service. The command aborts.
	Action: Bring the device back into service and reenter the command.
-continued-	

backuplog (continued)

Responses for the backuplog command (continued)	
MAP output	**Meaning and action**
`Error : File <filename> already exists on disk.` `Error : Duplicate file name on output volume!`	
	Meaning: A file with the same name as one of the files to back up already exists on the backup volume. The command aborts with all created files erased.
	Action: Use a volume with no duplicated file names. You may clear the backup volume or simply erase the duplicated file on the destination device.
`Error : File <filename> created on destination deleted`	
	Meaning: A file created on the destination has been deleted. This is part of the clean up on failure.
	Action: None
`Error : TRMS log file <fname> cannot be found!`	
	Meaning: A log file expected to exist is not found. This can mean that a previous backup of log has been killed before it can update the ArchiveNmb and LogTailAddress of the log so that these numbers indicate an older log address than what has been backed up and deleted. It is not treated as an error.
	Action: Take note of the incidence. You may check the number of files opened by using the DISKUT directory commands.

`Error : Volume not mounted`
`Completion msg received`
`Msg Time: <date> <hr: min: sec: ms>;`
`Error : Error in backup of Database <dbname>.`	

	Meaning: The DAT tape is not mounted. The command aborts.
	Action: Mount the DAT tape and reenter the command.
-continued-	

Responses for the backuplog command (continued)

MAP output	Meaning and action

Error : Wait for files to be closed by previous backup.

	Meaning:	After a database is backed up, the backup process waits for some time before it closes the database files. You have the chance to open them before they are closed. A second backup or restore is not permitted until the previous backup has closed all these files, which may take a few minutes. The command aborts.
	Action:	Wait for a few minutes and reenter the command. You may check the number of files opened by using the DISKUT directory commands.

Messaging System problem.

	Meaning:	The system encountered an error at the messaging system. The command aborts.
	Action:	Get help from an expert on the TRMS system.

No Master Entry - DB not defined!

	Meaning:	You specified a database that is not defined. The command aborts.
	Action:	Reenter the command using a database that is defined.

No more storage space on output medium.

	Meaning:	You specified an output device that is full. The command aborts and the backup on the disk is erased.
	Action:	Check the size of the backup and specify an output device that has enough storage.

Path Name too Long!

	Meaning:	You specified a destination path of more than 80 characters. The command aborts.
	Action:	Reenter the command using a correct path name.

Report : Backing up system log:

	Meaning:	The system log backup is started.
	Action:	Do nothing until the backup is completed or a failure occurs.

-continued-

backuplog (continued)

Responses for the backuplog command (continued)	
MAP output	**Meaning and action**
`Report : Backup for <dbname> is Completed.`	
	Meaning: All files of the database are backed up.
	Action: Do nothing until the backup is completed or a failure occurs.
`Report : Destination Device is OK.`	
	Meaning: The backup process is started.
	Action: Do nothing until the backup is completed or a failure occurs.
`Report : No log file needs to be backed up.`	
	Meaning: There is no log file in the database or system log that needs to be backed up.
	Action: Do nothing until the backup is completed or a failure occurs.
`Report : TRMS log file <filename> is Backed up`	
	Meaning: The log file is backed up.
	Action: Do nothing until the backup is completed or a failure occurs.
`STARTTIME <date> <hour: minute: second>;`	
	Meaning: The backup process has started.
	Action: If the wait option is issued, do nothing until the backup is completed or a failure occurs.
	If the no wait option is issued, use the dbstatus command to find out the status of the backup in regular intervals.
-continued-	

Responses for the backuplog command (continued)

MAP output   Meaning and action

```
StoreRC error (from STOR) : <number>
T_MTS_RC error (from MTSKERN) : <number>
MBRC error (from MESSAGES) : <number>
FlagRC error (from FLAGS) : <number>
SemaRC error (from SEMAS) : <number>
CIRetCode error (from CITYPES) : <number>
```

> **Meaning:** The system encountered an error at the corresponding system. The command aborts.
>
> **Action:**   Get help from an expert on the TRMS system.

```
Suffix Name too long!
```

> **Meaning:** You specified a suffix of more than three characters. The command aborts.
>
> **Action:**   Reenter the command using a shorter suffix.

```
TRMS system error.
```

> **Meaning:** The system encountered an error, possibly a software error in the transaction record management system. The command aborts.
>
> **Action:**   Get help from an expert on the TRMS system.

```
Volume not mounted.
```

> **Meaning:** You specified a DAT device that is not mounted. The command aborts.
>
> **Action:**   Mount the DAT by putting the DAT into the device slot.

```
Waiting for report Messages:
Warning - this may take some time!
```

> **Meaning:** Wait for more report messages to come from backup or restore.
>
> **Action:**   Do nothing until the backup is completed or a failure occurs.

End

cancel

Function

Use the cancel command to stop a backup or restore.

cancel command parameters and variables		
Command	**Parameters and variables**	
cancel	*dbname* *dbinstance* *operation*	
Parameters and variables	**Description**	
dbname	This variable specifies the database name.	
dbinstance	This variable specifies the database instance. The valid entry range is -1-31.	
operation	This variable specifies the operation you want to stop. The valid entry values are replicate, backup, and restore.	

Qualifications

The cancel command is qualified by the following exceptions, restrictions and limitations:

- You must re-initialize the volume whenever a restore is canceled and before the next attempt to restore the database.
- Cancellation of a restore does not return the environment to the state it was in before the restore. The cancel simply drops out of the restore at its current execution point.
- If the restoration of the database fails, the failure is reported and the cancel command is ignored.
- Cancellation of a backup to disk restores the destination disk to its pre-backup environment. Any component files that have been backed up prior to the issuance of the cancel command are erased from the backup disk.
- Cancellation of a backup to digital audio tape (DAT) does not restore the destination DAT to its pre-backup environment. Any component files that have been backed up prior to the issuance of the cancel command remain on the DAT.
- After the database has been backed up, the log files are deleted if the appropriate set or requirements have been met. If a cancel command is issued while the log files are being deleted, the cancel command is accepted and the deletion of the log files is stopped but the backed up files remain on the destination disk or DAT.

cancel (continued)

- If the backup of the database fails, the failure is reported and the cancel command is ignored.

Examples

The following table provides examples of the cancel command.

Examples of the cancel command	
Example	**Task, response, and explanation**
cancel 800plus 0 restore ↵ *where* 800plus specifies the database name 0 specifies the instance restore specifies the operation	
	Task: Cancel a database restore. **Response:** `Cancel could take up to 25 mins.` **Explanation:** You canceled the restore of 800plus database instance 0.
cancel 800plus 0 backup ↵ *where* 800plus specifies the database name 0 specifies the instance backup specifies the operation	
	Task: Cancel a database backup. **Response:** `Cancel could take up to 25 mins.` **Explanation:** You canceled the backup of 800plus database instance 0.

cancel (continued)

Responses

The following table provides explanations of the responses to the cancel command.

Responses for the cancel command	
MAP output	**Meaning and action**
Backup operation cancelled by user or device unavailable.	
	Meaning: You canceled a backup operation.
	Action: None
Error: Error in backup Database <database name>.	
	Meaning: Your backup operation failed.
	Action: None
Error: Error in restore Database <database name>.	
	Meaning: Your restore operation failed.
	Action: None
Report: Backup of Database <database name> was CANCELLED.	
	Meaning: You successfully canceled a backup operation.
	Action: None
Report: Restore of Database <database name> was CANCELLED.	
	Meaning: You successfully canceled a restore operation.
	Action: None
Restore operation cancelled by user or device unavailable.	
	Meaning: You canceled a restore operation.
	Action: None
-continued-	

cancel (end)

Responses for the cancel command (continued)

MAP output	Meaning and action
`There is currently no active database backup.`	
	Meaning: You attempted to cancel a backup that is not running.
	Action: None
`There is currently no active database restore.`	
	Meaning: You attempted to cancel a restore that is not running.
	Action: None
End	

dbstatus

Function

Use the dbstatus command to report the status of a backup or a restore.

dbstatus command parameters and variables	
Command	**Parameters and variables**

dbstatus	⌈ ba ⌉ ⌈ _recent_ ⌉
	⌊ re ⌋ all
	     db _dbname_
	     ⌊ id _backupid_ ⌋

Parameters and variables	**Description**
recent	Omitting this entry forces the system to default to displaying the most recent data.
all	This parameter specifies that all databases display.
ba	This parameter specifies a query of a backup.
backupid	This variable specifies the identification of the database backup.
db	This parameter specifies a particular database.
dbname	This variable specifies the database name.
id	This parameter identifies the database by an identification code.
re	This parameter specifies a query of a restore.

Qualification

Currently, only five report messages are stored for backup or restore. If
more than five errors have occurred, you may lose the information. For
example, if you fail to delete more than five log files, you may lose the
information that some log files are not deleted.

dbstatus (continued)

Example

The following table provides an example of the dbstatus command.

Example of the dbstatus command	
Example	**Task, response, and explanation**
dbstatus **ba** ↵	
	Task: Display the status of the most recent backup.

	Response: `Status report for BACKUP of database:`
	`-------------------------------------------------`
	`Backup status Report:`
	`-------------------------------------------------`
	`DBName : <DBName>`
	`*** Backup for Database ***`
	`  User Name: <user name>`
	`  StartTime: <start time>`
	`  EndTime  : <end time>`
	`  Status of Backup: *** <status> ***`
	
	`  Backup DB Name  : <DBname>`
	`  Backup Device: <DAT/DISK>`
	`  Backup Directory: <backup directory>`
	
	`Report Messages:`
	`Msg Time: <time>`
	`Report: <message>`
	`Msg Time: <time>`
	`Report: <message>`
	`...`
	`-------------------------------------------------`

	Explanation: You see all the information for the most recent backup.

dbstatus (continued)

Responses

The following table provides explanations of the responses to the dbstatus command.

Responses for the dbstatus command

MAP output   Meaning and action

```
Status report for BACKUP of database:
-----------------------------------------------------------
Backup status Report:
-----------------------------------------------------------
DBName : <DBName>
*** Backup for Database and Log. ***
  User Name: <user name>
  StartTime: <start time>
  EndTime  : <end time>
  Status of Backup: *** <status> ***

  Backup DB Name  : <DBname>
  Backup Device: <DAT/DISK>
  Backup Directory: <backup directory>

Report Messages:
Msg Time: <time>
Report: <message>
Msg Time: <time>
Report: <message>
 ...
-----------------------------------------------------------
```

Meaning: You see status information about the most recent backup on a database and log.

Action: If you see error messages, look to backupdb responses for the appropriate actions.

```
Status report for BACKUP:
No backup on record!
```

Meaning: You specified a backup that does not exist.

Action: None

-continued-

---

dbstatus (end)

Responses for the dbstatus command (continued)

MAP output   Meaning and action

```
----------------------------------------------------------
Restore status Report:
----------------------------------------------------------
DBName : <DBName>

  User Name: <user name>
  StartTime: <start time>
  EndTime  : <end time>
  Status of Backup: *** <status> ***

  Backup DB Name  : <DBname>
  Backup Device: <DAT/DISK>
  Backup Directory: <backup directory>

Report Messages:
Msg Time: <time>
Report: <message>
Msg Time: <time>
Report: <message>
...
----------------------------------------------------------
```

Meaning: You see information about the most recent restore of a database from a backup.

Action: If you see error messages, look to restoredb responses for the appropriate actions.

```
Status report for RESTORE:
No restore on record!
```

Meaning: You specified a restore that does not exist.

Action: None

End

help

Function

Use the help command to receive online documentation for the DBUT directory.

help command parameters and variables	
Command	**Parameters and variables**
help	<u>*all*</u> *command_nam*

Parameters and variables	**Description**
<u>*all*</u>	Omitting this entry forces the system to default to displaying online documentation for this directory.
command_nam	This variable specifies a valid DBUT directory command name. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command		
Example	**Task, response, and explanation**	
help ↵		
	Task:	Access online documentation.
	Response:	Not currently available
	Explanation:	This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

quit

Function

Use the quit command to exit the DBUT directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⌐ *1 level* ⌐ all *name* *n_levels* ⌐

Parameters and variables	Description
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command	
Example	**Task, response, and explanation**
quit ↵	
	Task: Exit from this directory.
	Response: `CI:`
	Explanation: You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

restoredb

Function

Use the restoredb command to restore a backup database.

restoredb command parameters and variables						
Command	**Parameters and variables**					
restoredb	*dbname*	*dbinstance*	*source_loc*	bdb	*dbname.bak* *backupdbname*	(1) (2)
restoredb (continued)	(1) (2)	wait	*y* n			(end)

Parameters and variables	Description
dbname.bak	Omitting this entry forces the system to default to using the original database name with the .bak suffix.
y	Omitting this entry forces the system to default to waiting until the restore is completed and display progress information.
backupdbname	This variable specifies the backup database name used when the backup was created.
bdb	This parameter indicates the backup database name is specified.
dbinstance	This variable specifies the instance of the database. The complete database name includes the original database name, __master__, and the dbinstance. If the dbinstance has two digits, the complete database name includes the original database name, __master_, and the two-digit dbinstance.
dbname	This variable specifies the database name to restore. Typically it is the local master database. It can also be the name of a disk-based backup database, in which case, the local master is not affected.
n	This parameter specifies the system not wait until the restore is completed when the wait parameter is indicated. You must use dbstatus to monitor the restore process frequently.
-continued-	

restoredb (continued)

restoredb command parameters and variables (continued)	
Parameters and variables	**Description**
source_loc	This variable specifies the device where the backup is stored. It can either be a digital audio tape (DAT) volume name or the name of a disk-based backup.
wait	This parameter indicates the wait state.
End	

Qualifications

The restoredb command is qualified by the following exceptions, restrictions, and limitations:

- Before doing a restore, you must re-initialize the volume.
- If the wait option is chosen, the CI is tied up during the restore process.
- If the nowait option is chosen, then you must monitor the status of the restore by using the dbstatus command frequently.

Examples

The following table provides examples of the restoredb command.

Examples of the restoredb command		
Example	**Task, response, and explanation**	
restoredb 800plus 0 fp00ct00 ↵ *where* 800plus 0 fp00ct00	 specifies the database name specifies the database instance specifies the source location	
	Task:	Restore a database.
	Response:	Not currently available
	Explanation:	You restored the database with name 800plus__master__0 from the DAT fp00ct00. The default backup database name is 800plus__master__0.bak. The system waits for the restore to complete and displays progress messages.
-continued-		

restoredb (continued)

Examples of the restoredb command (continued)

Example   Task, response, and explanation

---

restoredb 800plus 0 fp00ct00 bdb '800plus.june10' wait y ↵
where

800plus	specifies the database name
0	specifies the database instance
fp00ct00	specifies the source location
'800plus.june10'	specifies the backup database name
y	specifies the wait state

Task:	Restore a database with a specified backup name.
Response:	Not currently available
Explanation:	You restored the database with name 800plus__master__0 with the backup 800plus.june10 from the DAT fp00ct00. The system waits for the restore to complete and displays progress messages.

---

restoredb 800plus 0 :/fp00dk01800bak bdb 800plusdb_june7 wait n ↵
where

800plus	specifies the database name
0	specifies the database instance
:/fp00dk01800bak	specifies the source location
800plus_june7	specifies the backup database name
n	specifies the wait state

Task:	Restore a database with a specified directory path name.
Response:	Not currently available
Explanation:	You restored the database with name 800plus__master__0 with the backup 800plus_june7 from the disk directory :/fp00dk01800bak. The directory is :/fp00dk01 and the volume is 800bak. The system does not wait for the restore to complete and does not display progress messages.

-continued-

restoredb (continued)

Examples of the restoredb command (continued)

Example **Task, response, and explanation**

---

restoredb 800plus 0 :/ss00800bak bdb 800plusdb_june7 ↵
where

800plus	specifies the database name
0	specifies the database instance
:/ss00800bak	specifies the source location
800plus_june7	specifies the backup database name

Task:	Restore a database with a specified shadow set.
Response:	Not currently available
Explanation:	You restored the database with name 800plus__master__0 with the backup 800plus_june7 from the shadow set :/ss00800bak. The shadow set is :/ss00 and the volume is 800bak. The system waits for the restore to complete and displays progress messages.

End

Responses

The following table provides explanations of the responses to the restoredb command.

Responses for the restoredb command

MAP output **Meaning and action**

`A database backup already in progress!`

> **Meaning:** You attempted to run more than one backup or restore at a time. The command aborts.
>
> **Action:** Wait until the backup is completed and reenter the command. You can check the progress using the dbstatus command.

`A database restore already in progress!`

> **Meaning:** You attempted to run more than one backup or restore at a time. The command aborts.
>
> **Action:** Wait until the restore is completed and reenter the command. You can check the progress using the dbstatus command.

-continued-

---

restoredb (continued)

Responses for the restoredb command (continued)

MAP output   **Meaning and action**

```
Completion msg received
Msg Time: <date> <hour: minute: second: msec>;
Report: Database <dbname> is restored.
```

Meaning:	The restore process has completed. The Msg Time indicates the time of completion.
Action:	None

```
DAT Tape device in use by other user!
```

Meaning:	You specified a tape device that is already in use. The command aborts.
Action:	Free up the required output device or select a different source device.

```
Database must be NOT IN SERVICE for restoredb.
```

Meaning:	You specified a database that is not out of service. The command aborts.
Action:	If possible, take the database out of service. You may need to busy the update processing module (UPI). Reenter the command.

```
Device is not ready
```

Meaning:	You specified a device that exists but is not ready. The device may possibly be turned off. The command aborts.
Action:	Check the device status on the MAPCI, return the device to service, and reenter the command.

```
Device out of service.
```

Meaning:	You specified a source device that is out of service. The command aborts.
Action:	Check the device status on the MAPCI, return the device back to service, and reenter the command.

-continued-

restoredb (continued)

Responses for the restoredb command (continued)

MAP output   **Meaning and action**

`Duplicate file name on output volume.`

Meaning: A file with the same name as one of the files to restore already exists on the restore volume. The command aborts with all created files erased.

Action: Use a volume with no duplicated file names. You may clear the restore volume or simply erase the duplicated file on the destination device.

`Error : Backup <backupdbname> not found!`

Meaning: You specified a backup file that is not on the source directory. The command aborts.

Action: Check the source directory or DAT and the backup database name given in the restore command. Make sure the name matches the database stored on the source and reenter the command.

`Error : Bookkeeping file <filename> already exists`

Meaning: If restore is from DAT, the bookkeeping file is found on the destination disk volume. The command aborts. If restoring on top of an existing database, the database is still intact.

Action: Clear the offending file and reenter the command. If not successful, clear the volume and reenter the command.

`Error : Cache already exists in restore.`

Meaning: A cache with the same name as a cache of the database you are restoring already exists. The command aborts.

Action: You may clear the volume to clear up the cache in TRMS.

`Error : FTFS Err : File not found`

Meaning: A file is missing on the backup. The backup used is probably corrupted. The command aborts.

Action: You should clear up the volume and reenter the command with another backup.

-continued-

Responses for the restoredb command (continued)

MAP output	Meaning and action

Error : Log already exists in restore.

> **Meaning:** A log with the same name as the log of the database you are restoring exists already. The command aborts.
>
> **Action:** You may clear the volume to clear up the log in transaction record management system (TRMS).

Error : No more file on DAT tape

> **Meaning:** A file is missing on the backup on DAT. The backup used is probably corrupted. The command aborts.
>
> **Action:** You should clear up the volume and reenter the command with another backup.

Error : Problem with bookkeeping file in backup or restore.

> **Meaning:** The bookkeeping file stores information about the database in the backup. This error indicates that the backup is corrupted because the information in the bookkeeping file is not proper. The command aborts.
>
> **Action:** You should clear up the volume and reenter the command with another backup.

Error : Wait for files to be closed by previous backup.

> **Meaning:** After a database is backed up, the backup process waits for some time before it closes the database files. You have the chance to open them before they are closed. A second backup or restore is not allowed until the previous backup has closed all these files, which may take a few minutes. The command aborts.
>
> **Action:** Wait for a few minutes and reenter the command. You may check the number of files opened by using the DISKUT directory commands.

Invalid label on DAT tape.

> **Meaning:** You tried to restore from a DAT tape that has an invalid label. The command aborts.
>
> **Action:** Use the tape with the correct label.

-continued-

restoredb (continued)

Responses for the restoredb command (continued)

MAP output   Meaning and action

`No Master Entry - DB not defined!`

   Meaning: You specified a database that is not defined. The command aborts.

   Action: Reenter the command using a database that is defined.

`No more storage space on output medium.`

   Meaning: You specified an output device that is full. The command aborts and the restore on the disk is erased.

   Action: Check the size of the backup and specify an output device that has enough storage.

`Report : File <filename> is restored.`

   Meaning: The TRMS file is restored.

   Action: Do nothing until the restore is completed or a failure occurs.

`Report : Source Device is OK.`

   Meaning: The restore process is started.

   Action: Do nothing until the restore is completed or a failure occurs.

`Shall we destroy the existing database? (y/n)`

   Meaning: You attempted to restore a database that exists. The system waits for confirmation.

   Action: Enter n or carriage return to cancel the request. The command aborts.

   Enter y to proceed with the restore that destroys the existing database before recreating the database from the backup.

-continued-

restoredb (end)

Responses for the restoredb command (continued)	
MAP output	**Meaning and action**

`STARTTIME <date> <hour: minute: second>;`
`Waiting for report messages:`
`Warning - this may take some time!`	

	Meaning: The restore process has started.
	Action: If the wait option is issued, do nothing until the restore is completed or a failure occurs. If the no wait option is issued, use the dbstatus command to find out the status of the restore in regular intervals.
`Volume not mounted.`	
	Meaning: You specified a DAT device that is not mounted. The command aborts.
	Action: Mount the DAT by putting the DAT into the device slot.
End	

setnode

Function

Use the setnode command to set the node number for the master database.

setnode command parameters and variables	
Command	**Parameters and variables**
setnode	fp *nodenum*

Parameters and variables	Description
fp	This parameter indicates the file processor (FP).
nodenum	This variable specifies the node number where the master database resides. The valid entry range is 0-99.

Qualification

> **CAUTION**
> **Risk of data corruption**
> The setnode command must be done before any other commands in the DBUT directory can be issued.

You are prompted to do a setnode command if you violate the caution.

Example

The following table provides an example of the setnode command.

Example of the setnode command	
Example	**Task, response, and explanation**
setnode fp 6 ↵ *where*	
6	specifies the node number
	Task: Set the master database node number.
	Response: Not currently available
	Explanation: You set the master database node number to six.

setnode (end)

Response

The following table provides an explanation of the response to the setnode command.

Response for the setnode command	
MAP output	**Meaning and action**
Node does not exist.	
	Meaning: You specified a node number that does not exist.
	Action: Reenter the command with a valid node number.

DCTTOOL level commands

Use the DCTTOOL level of the MAP to access the data call tester (DCT) tool commands.

Accessing the DCTTOOL level

To access the DCTTOOL level, enter the following from the CI level:

dcttool ⏎

DCTTOOL commands

The commands available at the DCTTOOL MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

DCTTOOL commands	
Command	**Page**
delete	D-133
display	D-141
help	D-149
quit	D-151
testbook	D-155

delete

Function

Use the delete command to delete all or a subset of the test results belonging to the active testbook, a list of test numbers or a range of test numbers, or both. In addition, this command allows you to delete test results with no errors only.

---

delete command parameters and variables

Command	Parameters and variables

delete
del

all ⎡ *all* ⎤
     ⎣ noerr ⎦

t   ⎡ *test_num(s)*   ⎤   ⎡ *all* ⎤
    ⎢ *1st_num* to *lst_num* ⎥   ⎣ noerr ⎦
    ⎣ *both*   ⎦

     testbook

Parameters and variables	Description
all	Omitting this entry forces the system to default to deleting all test results, with or without errors.
1st_num	This variable specifies the first test number in a range of test numbers. Test numbers can be any number in the range of 1-65535.
all	This parameter deletes all tests in the active testbook. (Using the noerr parameter in the delete all noerr command string deletes all test results with no errors.)
both	This variable represents an entry of both a range and a list of test numbers. A list can be followed by a range which, in turn, can be followed by an additional list of test numbers. Lists can contain up to a maximum of ten test numbers, each separated by a space. A range is specified with the starting test number in the range, the to parameter, and the ending test number in the range. Test numbers can be any number in the range of 1-65535.

If the noerr parameter is specified with this command string, the system deletes test results with no errors for the test numbers in the list and range specified. If you do not specify the noerr parameter, the system defaults to deleting test results for all test numbers, with or without errors, in the list and range specified. |

-continued-

---

delete (continued)

delete command parameters and variables (continued)	
Parameters and variables	**Description**
last_num	This variable specifies the last test number in a range of test numbers. Test numbers can be any number in the range of 1-65535.
noerr	This parameter deletes only test results with no errors for the specified test numbers.
t	This parameter precedes a list or range of test numbers.
testbook	This parameter deletes the active testbook ID from the list of testbooks. Before using this command, all test results in the testbook have to be deleted using the delete all command string. (Normally, test results should be backed-up before being deleted.)
test_num(s)	This variable specifies a test number or test numbers to be deleted. The list may contain up to a maximum of ten test numbers, each separated by a space. Test numbers can be any number in the range of 1-65535.
to	This parameter must be used between the *1st_num* and *last_num* variables of a range of test numbers. The to parameter between test numbers distinguishes a range from a list.
End	

Qualifications

The delete command is qualified by the following exceptions, restrictions, and limitations:

- The maximum number of individual test numbers allowed in a delete command is 10.
- A testbook cannot be deleted if a dial command is executing in it, if test results exist that belong to it, if it is active on another MAP, or if it is a TERMRES TESTBOOK.
- Test number values cannot be greater than 65535.
- The delete command applies to the active testbook.
- If it is necessary to delete a testbook, all test results in the testbook have to be deleted first using the delete all command string. (Normally, test results should be backed-up before being deleted.)

delete (continued)

Examples

The following table provides examples of the delete command.

Examples of the delete command	
Example	**Task, response, and explanation**
delete all ↵	
	Task: Delete all test results in the active testbook.

	Response: `Please confirm ( y / n )`
	`>y`
	`DELETE command completed.`
	`37 test results DELETEd.`

	Explanation: The system deletes all test results, with or without errors, from the active testbook.
delete testbook ↵	
	Task: Delete the active testbook.

	Response: `Please confirm ( y / n )`
	`>y`
	`DELETE command not executed.`
	`Testbook HOMER1 contains test results.`

	Explanation: The active testbook cannot be deleted because it contains test results.
delete t 123 402 501 ↵ *where* 123, 402, 501 specifies test numbers	
	Task: Delete the test results for tests numbered 123, 402, and 501.

	Response: `Please confirm ( y / n )`
	`>y`
	`DELETE command completed.`
	`3 test results DELETEd.`

	Explanation: The test results for the three selected tests have been deleted.
-continued-	

delete (continued)

Examples of the delete command (continued)	
Example	**Task, response, and explanation**

delete t 400 401 406 408 noerr ↵
where

400, 401, 406, 408   specifies test numbers

Task:	Delete the test results without errors for the specified tests.
| **Response:** | ```
Please confirm ( y / n )
>y
DELETE command completed.
2 test results DELETEd.
``` |
| **Explanation:** | Only two test results were found to have no errors and were deleted. All four test results exist. |

delete t 640 to 659 ↵
where

640 and 659   specifies the first and last test numbers of a range

Task:	Delete the results of all tests in a range.
| **Response:** | ```
Please confirm ( y / n )
>y
DELETE command completed.
20 test results deleted
``` |
| **Explanation:** | The results of all 20 tests in the range have been deleted. |

End

Responses

The following table provides explanations of the responses to the delete command.

delete (continued)

Responses for the delete command	
MAP output	**Meaning and action**
A dial command is executing in testbook <id>.	
	Meaning: The delete testbook command is issued for a testbook with no test results in it while a dial command is executing, therefore, the delete command cannot execute.
	Action: The testbook can be deleted by aborting the dial command with the stop command, or deleting all test results the delete all command and then reissuing the command The testbook can also be deleted by waiting for the dial command to be completed, then deleting the test results and reentering the delete testbook command.
DELETE command completed.	
	Meaning: This message displays when the delete command executes successfully.
	Action: None
DELETE command failed. Try again Report swerr to the next level of support.	
	Meaning: An internal error was encountered.
	Action: Report any SWERRS generated.
DELETE command not executed.	
	Meaning: This message displays when the delete command does not execute successfully, for example, when you enter an invalid option.
	Action: None
<n> test results DELETEd.	
	Meaning: The delete command is completed and <n> is the number of test results deleted.
	Action: None
No Testbook is active.	
	Meaning: No testbook is active and the command cannot execute.
	Action: None
-continued-	

delete (continued)

Responses for the delete command (continued)

MAP output   **Meaning and action**

```
No test results exist in testbook <id>.
```

Meaning: You issued the delete t command string for a testbook without test results.

Action: None

```
Testbook <id> contains test results.
```

Meaning: The delete testbook command cannot be executed because the testbook, identified as <id>, contains test results.

Action: If the testbook is to be deleted, all test results in the testbook have to be deleted first using the delete all command. Test results should normally be backed up before being deleted.

```
Testbook <id> is being used by another MAP.
```

Meaning: The delete testbook command is issued for a testbook, identified by <id>, which is being used by another MAP user. The command cannot, therefore, be executed.

Action: Take no action or wait until no one is using the specified testbook and reissue the command.

```
Testbook <id> DELETEd.
```

Meaning: This responses indicates that the delete testbook command string was successful.

Action: None

```
Test results queue is busy.
A DELETE or DISPLAY command is being executed by another user.
Try again later.
```

Meaning: You attempted to delete test results when the test results queue is being used by another user.

Action: Wait a moment, then try again.

-continued-

Responses for the delete command (continued)

MAP output	Meaning and action
TERMRES testbooks cannot be DELETEd.	
	Meaning: You attempted to delete testbook TERMRES0 or TERMRES1. These terminating testbooks cannot be deleted.
	Action: None
End	

display

Function

Use the display command to display all or a subset of the test results belonging to the active testbook. You can display test results as they occur (real time) for the current and subsequent tests in an active originating testbook, or one of the current tests for TERMRES testbooks. In addition, this command allows you to display test summaries, or isolate test results occurring with errors or without errors.

display command parameters and variables		
Command	**Parameters and variables**	

display **dis**	all	[*both* err noerr] [*full results* summ]
	off on	[*all* *test_no*]
	t	[*tst_num(s)* *1st_num* to *last_num*] [*both* err noerr] [*full results* summ]

Parameters and variables	**Description**
all	Omitting this entry forces the system to default to activating or deactivating the continuous display for all tests. The only time you specify a test number is when you are activating or deactivating the continuous display in the TERMRES0 or TERMRES1 testbook.
both	Omitting this entry forces the system to default to displaying any test results, including tests with errors and tests without errors.
full results	Omitting this entry forces the system to default to displaying full test results for all tests because no summ, err, or noerr parameter is entered.
1st_num	This variable specifies the first test number in a range of test numbers. Test numbers can be any number in the range of 1-65535.
all	This parameter selects all test results in the active testbook.
err	This parameter displays test results for tests with errors.
last_num	This variable specifies the last test number in a range of test numbers.
-continued-	

display (continued)

display command parameters and variables (continued)	
Parameters and variables	**Description**
noerr	This parameter displays test results for tests with no errors.
off	This parameter deactivates continuous display mode. Real time error display for the currently running test will stop.
on	This parameter activates continuous display mode. Errors are displayed as they occur for all subsequent tests. In TERMRES0 or TERMRES1, this parameter has to be followed by a test number.
summ	This parameter causes a summary of test results for the selected tests to be displayed. *Note:* If the summm parameter is used with a display t *tst_number* command string, the five-minute bins do not display.
t	This parameter precedes a list or range of test numbers.
test_no	This variable is the number of a test for which results are to be displayed. This value is required only if working in the TERMRES0 or the TERMRES1 testbook. This is required because many tests can be running simultaneously and the requested test must be identified explicitly. T est numbers can be any number in the range of 1-65535.
tst_num(s)	This variable specifies a test number or test numbers to be displayed. The list can contain up to a maximum of ten test numbers, each separated by a space. The test results for each test number are separated by asterisks in the displayed. Test numbers can be any number in the range of 1-65535.
to	This parameter must be used between the *1st_num* and *last_num* variables of a range of test numbers. The to parameter between test numbers distinguishes a range from a list.
End	

Qualifications

The display command is qualified by the following exceptions, restrictions, and limitations:

- The difference between this command and the menu version of the display command entered from the DCTTTP and DCTLTP menu level is that the results from the menu command display only one type of summary and only the summm parameter is recognized. In addition, the err and noerr parameters are not allowed.

display (continued)

- The display command applies only to the active testbook. If no testbook is active, the command is ignored.

- The display command is ignored if a delete command is being executed in the same testbook by any DCT MAP user. The test result queue (one for each testbook) is unavailable for reading when a delete command is acting on it.

- Real time error cannot be displayed on two different MAPs for the same test; that is, if a display on command string currently is executing on one MAP terminal for a testbook, it is ignored if issued from another MAP terminal for the same testbook.

- The display on command string automatically stops if the active testbook is changed.

Examples

The following table provides examples of the display command.

Examples of the display command	
Examples	**Task, response, and explanation**
displayall err summ ↵	
	Task: Produce a summary display of all test results with errors in the active testbook.
	Response:
	``` Testbook ID: HOMER1 Test number: 10 Peer test number: 55 Call duration: 120 DN: 5186990701 Starttime: 1993/01/20 09:40:04 Stoptime: 1993/01/20 09:42:07 IBERT: 1 (HOST 00 0 00 00) Peer IBERT: 3 (HOST 00 0 01 15) Errors: 4 Errored seconds: 1 Error free seconds: 119 Number of sync losses: 0 Blocks received: 245520 Block size: 2047 bits Call status: completed Call setup time: 0 Call termination reason: normal termination DISPLAY command completed. 1 test results DISPLAYed. ```
	Explanation: This command produces a summary display of all test results with errors in the active testbook.
-continued-	

display (continued)

Examples of the display command (continued)

Examples	Task, response, and explanation

displayall ↵

Task:	Produce all test results in the active testbook.

Response:

```
Testbook ID: HOMER1          Test number: 2
                             Peer test number: 52
Call duration: 120           DN: 5186990701
Starttime: 1993/01/20 09:34:04 Stoptime: 1993/01/20 09:36:07
IBERT: 1 (HOST 00 0 00 00)    Peer IBERT: 3 (HOST 00 0 01 15)
Errors: 4                    Errored seconds: 1
Error free seconds: 119      Number of sync losses: 0
Blocks received: 245520      Block size: 2047 bits
Call status: completed       Call setup time: 0
Call termination reason: normal termination
5-MIN WINDOW START TIME  ERRORED-SECONDS  ERRORS
            09:34:04     1                4
************************************************************
Testbook ID: HOMER1          Test number: 7
                             Peer test number: 55
Call duration: 120           DN: 5186990701
Starttime: 1993/01/20 09:40:04 Stoptime: 1993/01/20 09:42:07
IBERT: 1 (HOST 00 0 00 00)    Peer IBERT: 3 (HOST 00 0 01 15)
Errors: 4                    Errored seconds: 1
Error free seconds: 119      Number of sync losses: 0
Blocks received: 245520      Block size: 2047 bits
Call status: completed       Call setup time: 0
Call termination reason: normal termination
5-MIN WINDOW START TIME  ERRORED-SECONDS  ERRORS
            09:40:04     1                4
DISPLAY command completed.
2 test results DISPLAYed.
```

Explanation:	This command produces all test results in the active testbook. Since the summ parameter is not specified to receive a summary display, the system assumes the default and displays full results.

display on ↵

Task:	Activate continuous display mode.
Response:	DISPLAY command completed. Continuous DISPLAY is ON.
Explanation:	Continuous display mode is activated.

-continued-

---

display (continued)

Examples of the display command (continued)	
Examples	**Task, response, and explanation**

display off ↵

Task:	Deactivate the continuous display mode.
Response:	DISPLAY command completed. Continuous DISPLAY is OFF.
Explanation:	Continuous display mode is deactivated.

display t 10 to 20 ↵

Task:	Display results for a range of test numbers.
Response:	DISPLAY command completed. 0 test results DISPLAYed.
Explanation:	Test numbers 10 to 20 do not exist so the command cannot execute successfully.

display t 1 to 8 noerr ↵

Task:	Display all tests results with no errors for range of test numbers.
Response:	

```
Testbook ID: HOMER1         Test number: 4
Call duration: 120          DN: 5186990701
Starttime: 1993/01/20 09:34:04 Stoptime: 1993/01/20 09:36:07
IBERT: 1 (DTU 1)            Trunk member:  ISUPIBNOG 2
Errors: 0                   Errored seconds: 0
Error free seconds: 120     Number of sync losses: 0
Blocks received: 245520     Block size: 2047 bits
Call status: completed      Call setup time: 0
Call termination reason: normal termination
5-MIN WINDOW START TIME  ERRORED-SECONDS   ERRORS
            09:34:04     1                 4
DISPLAY command completed.
1 test results DISPLAYed.
```

Explanation:	This command displays all test results with no errors for test numbers in the range of 1 to 8. Since the summ parameter is not specified to receive a summary display, the system assumes the default and displays full results.

End

display (continued)

Responses

The following table provides explanations of the responses to the display command.

Responses for the display command	
MAP output	**Meaning and action**
Continuous display is already off.	
	Meaning: This message indicates that the continuous display mode was off when the display off command string is entered.
	Action: None
Continuous DISPLAY is already ON.	
	Meaning: The continuous display mode already has been turned on.
	Action: None
Continuous DISPLAY is being used by another MAP.	
	Meaning: The continuous display mode has been turned on at another MAP position.
	Action: None
Continuous DISPLAY is ON.	
	Meaning: The display on command string executed successfully.
	Action: None
DELETE command is executing. Try again later.	
	Meaning: You tried to display test results while a delete operation is being performed by another user in the same testbook.
	Action: Try again later.
DCT is unavailable. System maintenance in progress.	
	Meaning: You entered a display on command string or display off command string when DCT processing software is unavailable to handle the command.
	Action: Try the command again later.
-continued-	

display (continued)

Responses for the display command (continued)	
MAP output	**Meaning and action**
DISPLAY command completed.	
	Meaning: The display command executed successfully.
	Action: None
DISPLAY command failed. Try again. Report swerr to the next level of support.	
	Meaning: The internal DCT MAP to DCT process message to initiate continuous display is not acknowledged and the command fails. A SWERR is generated indicating that a time-out has occurred.
	Action: Try the command again later, and report the SWERR.
DISPLAY command not executed.	
	Meaning: This message displays when the display command does not execute successfully. This message is followed by additional explanation.
	Action: None
Next summary unavailable	
	Meaning: In a dial sequence, the next summary cannot be allocated due to memory shortage or the DCT_MEM_LIMIT is reached.
	Action: Free some summaries and try again.
No testbook is active	
	Meaning: No testbook is active when the display command with parameters is issued. If no testbook is active, no test results can be displayed, and no real time display can be initiated or stopped.
	Action: None
No test results in testbook <id>.	
	Meaning: No test results exist in the active test book.
	Action: None
-continued-	

display (end)

Responses for the display command (continued)	
MAP output	**Meaning and action**
`Test call <number> is not running in testbook <TERMRES0 or TERMRES1>.`	
	Meaning: The test call number specified by the display on command string for the TERMRES testbook is not currently running.
	Action: Use the testbook command to determine what test are running.
`Test number is required in TERMRES testbooks.`	
	Meaning: You issued a display on command string in a TERMRES testbook. The test number must be specified in a TERMRES testbook.
	Action: Use the testbook command to see the list of current test calls running in the TERMRES testbook; then, reenter the display on command using the test number.
`Test number <n> not found.`	
	Meaning: The test number you entered is invalid.
	Action: Use the testbook command without any parameters or the use count command to determine what tests are running. Use the DCTTOOL level display all command string to display all test numbers.
`WARNING: Test number ignored.` `TEst <id> is an originating testbook.`	
	Meaning: You specified a test number for continuous display in an originating testbook. The test number is simply ignored and the command is executed as if no test number were specified.
	Action: None
End	

help

Function

Use the help command to receive online documentation for the DCTTOOL directory.

help command parameters and variables	
Command	**Parameters and variables**
help	dcttool

Parameters and variables	Description
dcttool	This parameter produces online documentation for this directory describing the function of this directory and listing the valid DCTTOOL directory commands.

Qualification

Use a q *command_nam* command string to obtain more detailed help on each of the DCTTOOL directory commands.

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
helpdcttool ↵	

	Task:	Access online documentation.
	Response:	DCTTOOL ---Enter DCTTOOL CI increment Data Call Tester Tool commands level Commands Available inside DCTTOOL: TESTBOOK--Select an Active Testbook DISPLAY--DISPLAY TestBooks, Test summaries, etc. DELETE--DELETE TestBooks and Test Summaries Use Q <command> to obtain more detailed help on each of the commands. e.g. Q TESTBOOK will provide detailed help for the testbook command.
	Explanation:	This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

quit

Function

Use the quit command to exit the DCTTOOL directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⎡ *1 level* ⎤ ⎢ all ⎥ ⎢ *name* ⎥ ⎣ *n_levels* ⎦

Parameters and variables	Description
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where*		
dskut	specifies a directory	
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

testbook

Function

Use the testbook command to do the following:

- access a testbook
- create a new testbook
- display information on the active testbook
- count the number of testbooks
- list a specified number of testbook IDs from the list of testbook IDs
- move the list index up or down a specified number of items in the list of testbook IDs

testbook command parameters and variables	
Command	**Parameters and variables**
testbook **tes**	*active* count query *tstbk_id* point *tstbk_index* down ⎡ *one* up ⎣ *num_positions* list ⎡ *three* all *selected_number*

Parameters and variables	**Description**
active	Omitting this entry forces the system to default to displaying information on the active testbook.
one	Omitting this entry forces the system to default to moving the index up or down one position when no value for the *offset* variable is entered.
three	Omitting this entry forces the system to default to listing three testbook IDs when no value is entered for the *quantity* variable.
all	This parameter displays all testbook IDs in the list of testbooks. The index is moved to the bottom of the list.
count	This parameter displays the number of testbooks that have been created.
-continued-	

testbook (continued)

testbook command parameters and variables (continued)	
Parameters and variables	**Description**
down	This parameter moves the index down in the list of testbooks. The system moves down the number of items indicated by the *offset* variable. The active test book is not affected.
list	This parameter displays consecutive testbook IDs from the list of testbooks, starting at the current index position for as many testbooks as are specified by the next entry in the command string. (The default value is 3 positions.) The active testbook is not affected.
num_positions	The variable indicates the number of positions up or down to move the index. The valid entry range is 1-256.
point	This parameter points to the testbook ID at the position in the list of testbooks indicated by the *position* variable. This list is kept by the DCT tool and is in chronological order. This command does not affect the active testbook.
query	This parameter provides brief information on testbooks.
selected_number	This variable indicates the number of testbook IDs to list. The valid entry range is 1-256.
tstbk_index	This variable indicates the number of positions down from the first position to point the index. The valid entry range is 1-256.
tstbk_id	This variable identifies the testbook that will be active. The valid entry value is a 1- to 8-character alphanumeric string without reserved keywords including count, up, down, list, and point. *Note:* Although you can create a new testbook using this variable while in the DCTTTP menu levels, you cannot create a new testbook while using the nonmenu version of the DCTTOOL directory.
up	This parameter moves the index up in the list of testbooks. The system moves up the number of items indicated by the *offset* variable. The active test book is not affected.
End	

testbook (continued)

Qualifications

The testbook command is qualified by the following exceptions, restrictions, and limitations:

- The testbook command in the DCTTOOL_DIR cannot create a new testbook. Use the DCTTTP menu level to create a new testbook.

- Real time error display, requested by the display on command, is stopped when a new active testbook is entered; that is, the testbook *tstbk_id* command string stops any previously requested real time error display unless the *tstbk_id* is the ID of the currently active testbook.

Examples

The following table provides examples of the testbook command.

Examples of the testbook command		
Example	**Task, response, and explanation**	
testbook ↵		
	Task:	Determine the active testbook.
	Response:	
| | ```
Active testbook bcs35ck
Number     Last       Current   Number    DIAL     Testbook
of test    test       test      of tests  status   created
results    performed  number    to go              by
---------------------------------------------------------
100        1011       1012      11        Testing  JeanP
``` | |
	Explanation:	Since no parameters or variables are entered, the system assumes the default value of active. The active testbook and brief summary of contents are displayed.
testbook count ↵		
	Task:	Determine the number of testbooks.
	Response:	Number of testbooks is 126
	Explanation:	The display lists the number of testbooks.
-continued-		

testbook (continued)

Examples of the testbook command (continued)	
Example	**Task, response, and explanation**

testbook query ↵

Task:	Receive brief information on testbooks.
Response:	`Currently not available.`
Explanation:	The system displays brief information on testbooks. This command string behaves similar to the testbook count command string.

testbook xb25 ↵
where

xb25   specifies the ID of an existing testbook

Task: Obtain a brief summary of information on a specified testbook.

Response:
```
TESTBOOK sb25 is now active.
Active testbook xb25
Number     Last       Current    Number    DIAL      Testbook
of test    test       test       of tests  status    created
results    performed  number     to go               by
-------------------------------------------------------------
25         50         51         200       Testing   HLNGUYEN
```

Explanation: A brief summary of the contents of xb25 is displayed.

testbook point 300 ↵
where

300   specifies the number of positions the index is to move down from the top

Task: Point index to ID of testbook 300 positions from the top of the list.

Response: `Pointing to testbook prob512`
```
Testbook  Number    Number    Last                 Created
ID        of users  of tests  used                 by
-------------------------------------------------------------
prob512   3         93        1993/03/09 11:24:42  HAIN
```

Explanation: The system displays the ID of the testbook 300 positions from the top.

-continued-

testbook (continued)

Examples of the testbook command (continued)

Example   **Task, response, and explanation**

---

testbook list all ↵

        Task: List all of the testbooks.

        Response:

```
Testbook  Number    Number    Last                    Created
ID        of users  of tests  used                    by
------------------------------------------------------------
TERMRESO  5         67        1993/03/09 11:24:42 DCT
TERMRES1  5         67        1993/03/09 11:24:42 DCT
Z1        1         12        1993/03/10 13:22:45 ZEGRAY
Z2        3         25        1993/03/10 13:22:45 ZEGRAY
Homer1    1         14        1993/03/15  9:21:35 HOMAYOON
Homer2    4         14        1993/03/15  9:21:35 HOMAYOON
Number of testbooks listed: 6.
Bottom of testbook list.
```

        Explanation: All testbooks are listed.

End

Responses

The following table provides explanations of the responses to the testbook command.

Responses for the testbook command

MAP output   **Meaning and action**

```
Bottom of testbook list
```

      Meaning: This message displays whenever the index to the list of testbooks is taken to the bottom of the list by a testbook point command string, testbook down command string, or testbook list command string.

      Action: None

-continued-

---

testbook (continued)

Responses for the testbook command (continued)
MAP output   **Meaning and action**
`Continuous DISPLAY OFF for previous testbook <ID>.`
Meaning: This response indicates that the display on the command issued from the previous testbook stopped when the testbook <ID> was issued. Refer to the DCTTOOL directory display command. **Action:**   None
`DCT Office parameters are being changed.  Try again later.`
Meaning: This message responds to a testbook being selected when DCT Office parameters are being changed. This is because changing the OFCENG parameter DCT_MEM_LIMIT may cause the testbook memory structure to be affected. No testbooks should be accessed while DCT_MEM_LIMIT is being changed. **Action:**   Wait, and try again later.
`No testbook is active.`
Meaning: This message appears when the testbook command with no parameters or with query keyword is issued before setting an active testbook. **Action:**   Access a testbook, and try the command again.
`Number of testbooks is <n>`
Meaning: This message displays the number of testbooks as requested by the testbook count command string or the testbook query command string, where <n> is the number of testbooks. **Action:**   None
`Number of testbooks listed : <n>`
Meaning: This message displays the number of testbooks listed as requested by the testbook list command string, where <n> is the number of testbooks. **Action:**   None
-continued-

testbook (continued)

Responses for the testbook command (continued)	
MAP output	**Meaning and action**

`Pointing to testbook <ID>.`

	Meaning: This message is part of the acknowledgement of testbook point, testbook up, or testbook down. It displays the testbook ID at the resulting position.
	Action: None

`Severe error condition.  Cannot create or delete testbooks.`

	Meaning: This message is in response to an attempt to create or delete a testbook when certain resources are unavailable. This is usually associated with the repeated failure of the test call DCT processing software. In this case, only existing test results can be displayed or dumped to other media, and no test calls will be possible.
	Action: Try again, and if the same message appears, report this error condition. Test results should be dumped to other media in case of corruption of results.

`TESTBOOK command failed.  Try again.`
`Report swerr to the next level of support`

	Meaning: This message is in response to an unknown internal software error. The system generates a SWERR report.
	Action: Try again later. If the condition persists, check and report any SWERRS generated.

`TESTBOOK command not executed.`

	Meaning: This message displays when the testbook command is not successful, for example, when an invalid option or keyword is entered.
	Action: None

`Testbook <ID> does not exist.`
`To create it, access the DCT MAP sublevel.`

	Meaning: This message appears when trying to create a new testbook in the DCTTOOL level.
	Action: None

-continued-

testbook (continued)

Responses for the testbook command (continued)

MAP output	Meaning and action

```
Testbook <ID> is now active.
```

Meaning: This message confirms that the testbook is now active.

Action: None

```
Testbook <ID> is the active testbook.
```

Meaning: This response indicates that the specified testbook already is active.

Action: None

```
TESTBOOK LIST index is unpositioned.
Use TESTBOOK POINT command to reposition.
```

Meaning: This message is the response to testbook up command string, testbook down command string, or testbook list command string when the testbook preciously indexed has been deleted. The index is unpositioned and must be repositioned.

Action: Reposition the list index using the testbook point command.

```
Testbook list is busy.  Try again later.
```

Meaning: This message is the response to an attempts to execute testbook commands at more than one MAP at a time.

Action: Wait a few moments, then try again.

```
There are no testbooks
```

Meaning: This response appears when you execute a testbook up command string, testbook down command string, testbook list command string, or testbook point command string and there are no testbooks. Because testbooks TERMRESO and TERMRES1 are always present, this response should not appear under normal conditions, but can appear if TERMRES testbooks were not created and no originating testbooks have been created yet.

Action: Try the command again, and report this error to the next level of support.

-continued-

testbook (end)

Responses for the testbook command (continued)	
MAP output	**Meaning and action**
`There is a maximum of 256 testbooks.`	
	Meaning: You tried to create a new testbook when the maximum of 256 testbooks, including TERMRES0 and TERMRES1, have already been created.
	Action: Delete unneeded testbooks using the DCTTOOL directory delete command.
`There is no memory available to create a testbook.`	
	Meaning: An attempt to create a testbook has occurred when no more memory resources have been allocated to the DCT.
	Action: Delete unneeded test results and testbooks, and try again.
`This testbook is being used for incoming tst call results.`	
	Meaning: This message displays when currently running terminating tests are stored in the currently active terminating testbook (TERMRES0 or TERMRES1). Note that only one terminating testbook can be used for incoming calls at a time.
	Action: None
`Top of testbook list`	
	Meaning: This message displays when the index to the list of testbooks is taken to the top of the list by a testbook point command string or testbook up command string.
	Action: None
End	

DISKADM level commands

Use the disk administration (DISKADM) level of the MAP to initialize, configure and administer the system load module (SLM) device. The SLM stores the image files of several processors of the enhanced core switch. The DISKADM directory allows you to manage and administer these image files on the SLM disks.

Accessing the DISKADM level

To access the DISKADM level, enter the following command string from the CI level:

> **diskadm *device* ↵**

For more information on the diskadm command, see page NO TAG.

<table>
<tr><td></td><td>CAUTION
Must be manual busy (MBsy)
The commands in this directory can be used only if the device
is in the MBsy state.</td></tr>
</table>

The commands in this directory can be used only if the device is in the MBsy state.

DISKADM commands

The commands available at the DISKADM MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

DISKADM commands	
Command	**Page**
createvol	D-167
deletevol	D-175
-continued-	

DISKADM commands (continued)	
Command	**Page**
displaydisk	D-179
displayvols	D-183
formatdisk	D-185
help	D-191
quit	D-193
reinitvol	D-197
End	

createvol

Function

Use the createvol command to create a new volume on the disk. The createvol command sets the space (specified in megabytes) from a pool of the disk free space and adjusts the system data structure to reflect the existence of a new volume. The new volume contains a volume label, a file directory, a volume free space map, and space to store files.

The createvol command also allows you to specify the maximum size of the file directory (FD) and how fragmented the volume can become as files are created, deleted and removed.

- Create at least one volume to store system image files for the computing module (CM) and the message switch (MS).

- Make the volume large enough to hold at least one CM system image file and one MS system image file.

  System image files use approximately 40 megabytes and MS system image files use 4 megabytes.

  Reserve at least 10 percent of the space in a volume for the fragmentation of disk space.

  A volume that holds CM and MS system image files should therefore be approximately 50 megabytes in size.

- Create separate volumes to store other types of files such as XMS-based peripheral module (XPM) and peripheral module (PM) load files and non-resident modules.

- Increase the volume size for each batch change supplement (BCS) to accommodate larger files. For example, the size of the volume used to store system image files should increase by 5 percent.

createvol (continued)

createvol command parameters and variables		
Command	**Parameters and variables**	

createvol **cv**	*vol_name* *vol_size* ⌈ std ⌈ files <u>*128*</u> ⌊ ⌊ files *max_files* ⌉ ⌈ ftfs files *avg_files*	(1) (2) (3) (4)

createvol (continued)	(1) ⌈ <u>*64*</u> ⌉ (2) ⌊ segments *max_segs* ⌋ (3) segments *max_segs* lblocks *blksize* ⌉ (4)	(end)

Parameters and variables	**Description**
<u>*64*</u>	Omitting this entry forces the system to default to reserving space for 64 segments.
<u>*128*</u>	Omitting this entry forces the system to default to reserving space for 128 files.
avg_files	This variable specifies the average number of files stored in one Fault Tolerant File System (FTFS) volume. The valid entry range is 16-2048.
blksize	This variable specifies the logical block size. The valid entry range is 1-64.
files	This parameter indicates that the maximum files in each volume is specified.
ftfs	This parameter specifies an FTFS volume.
lblocks	This parameter indicates that the logical block size is specified.
max_files	This variable specifies the number of files that can be stored on one volume. The valid entry range is 16-2048.
max_segs	This variable specifies the number of free or unallocated segments of unused data blocks that can be stored in this volume. The valid entry range is 16-2048.
segments	This parameter indicates that the maximum free segments per volume is specified.
std	This parameter specifies a standard volume.
vol_name	This variable identifies the volume to create on the system load module (SLM) disk. It can be a maximum of eight characters long.
vol_size	This variable determines the size of the created volume. The valid entry range is 1-256 megabytes.

createvol (continued)

Qualification

⚠	**CAUTION** **Must be manual busy (MBsy)** The commands in this directory can be used only if the device is in the MBsy state.

The createvol command can be used only if the device is in the MBsy state.

Examples

The following table provides examples of the createvol command.

Examples of the createvol command	
Example	**Task, response, and explanation**
createvol core21 5 std 127 90 ↵ *where* core21 specifies the volume name 5 specifies the volume size 127 specifies the maximum number of files 90 specifies the maximum number of segments	

	Task:	Create a volume.

	Response:	```Volume core21 will be created on S00D.```
		```Volume size: 5 megabytes```
		```File Directory size: 127 files```
		```Volume Free Space Map size: 90 segments```
		
		```Do you want to continue?```
		```Please confirm ("YES" or "NO"):```
		```>yes```
		```Creation of the volume is completed.```

	Explanation:	You created a 5 megabyte standard volume called core21 with a maximum of 127 files and 90 segments.

-continued-

createvol (continued)

Examples of the createvol command (continued)

Example	Task, response, and explanation

---

createvol   **image 70 std** ↵
where

image   specifies the volume name
70   specifies the volume size

Task: Create a volume using defaults.

Response:
```
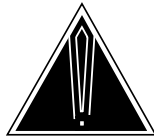Volume IMAGE will be created on S00D.
Volume size:                 70 megabytes
File Directory size:        128 files
Volume Free Space Map size:  64 segments

Do you want to continue?
Please confirm ("YES" or "NO"):
>yes
Creation of the volume is completed.
```

Explanation: You created a 70 megabyte standard volume called image with the defaults of a maximum of 128 files and 64 segments.

End

Responses

The following table provides explanations of the responses to the createvol command.

Responses for the createvol command

MAP output	Meaning and action

```
CREATEVOL command is aborted.
Duplicate volume name was specified.
```

Meaning: You specified a volume that has a duplicate volume name. The command aborts.

Action: Choose another name or delete the other volume of the same name.

-continued-

---

createvol (continued)

Responses for the createvol command (continued)

MAP output   Meaning and action

```
CREATEVOL command is aborted.
The command has been entered with the wrong parameters.
Check the SWERR generated and notify the appropriate people.
```

> **Meaning:** The software for the command encountered a software error and generated a SWERR. The command aborts.
>
> **Action:** Check the SWERR that is generated and inform the next level of support and ETAS of the problem. You may change the parameters and retry the command.

```
CREATEVOL command is aborted.
The device has encountered a hardware error.
Check the log generated and notify the appropriate people.
```

> **Meaning:** The device hardware is faulty. A log is generated indicating the nature of the problem and the command aborts. The device becomes system busy.
>
> **Action:** Check the log that was generated and Inform the next level of support. Maintenance personnel should attempt to return the device to service again.

```
CREATEVOL is aborted.
The device has encountered an error in doing a read after a write to the
medium on the disk.
Check the log generated and notify the appropriate people.
```

> **Meaning:** The device encountered an error in reading a block immediately after writing the medium on the disk. A log is generated indicating that there is a disk read after write error.
>
> **Action:** Check the log that was generated and inform the next level of support of the problem.

-continued-

createvol (continued)

Responses for the createvol command (continued)

MAP output   Meaning and action

```
CREATEVOL command is aborted.
The device has encountered an error in doing a read for verification
purposes on the medium on the disk.
Check the log generated and notify the appropriate people.
```

> **Meaning:** The device encountered an error doing a read for verification purposes on the medium on the disk. A log is generated indicating that there is a disk verification error.
>
> **Action:** Check the log that was generated and inform the next level of support of the problem.

```
CREATEVOL command is aborted.
The device has encountered an error in writing the medium on the disk.
Check the log generated and notify the appropriate people.
```

> **Meaning:** The device encountered an error in writing the medium on the disk. A log is generated indicating that there is a disk medium error. The system then attempts to read the alternate system data structure. If the alternate is read successfully, the command does not abort. If the alternate is not read successfully, the command aborts.
>
> **Action:** Check the log that was generated and inform the next level of support of the problem.

```
CREATEVOL command is aborted.
The MAX SEGMENT COUNT cannot exceed 16 times the MAX NO OF FILES.
```

> **Meaning:** You specified a maximum segment count is more than 16 times larger than the maximum number of files. The command aborts.
>
> **Action:** Reissue the command specifying a smaller number for the maximum segment count.

```
CREATEVOL command is aborted.
Volume exceeds the size of the disk.
```

> **Meaning:** You specified a volume size that exceeds the size of the disk. The command aborts.
>
> **Action:** Recalculate the size of the volume and ensure that the size is less than the size of the disk.

-continued-

Responses for the createvol command (continued)	
MAP output	**Meaning and action**
CREATEVOL command is aborted. Volume size exceeds the available space on the disk.	
	Meaning: You specified a volume size that exceeds the available space on the disk. The command aborts.
	Action: Recalculate the size of the volume and ensure that its size is less than or equal to the available contiguous space on the disk.
End	

deletevol

Function

Use the deletevol command to delete a volume from a system load module (SLM) disk. This operation deletes the entry for the volume from the volume directory. The space occupied by the volume is returned to the free space map for the disk. The disk label is updated to indicate that the disk was modified.

Volumes cannot be deleted if they contain system load files that are registered in the image table of contents (ITOC). These files must first be removed from the ITOC using the clearbootfl command. The displayvols command displays files registered in any ITOC. To ensure that the files are removed, use the listbootfl command in the DISKUT directory to display the ITOC.

deletevol command parameters and variables	
Command	**Parameters and variables**
deletevol **ddv**	*vol_name*
Parameters and variables	**Description**
vol_name	This variable indicates the name of the volume to delete. The volume name can have a maximum of eight characters.

Qualifications

CAUTION
Must be manual busy (MBsy)
The commands in this directory can be used only if the device is in the MBsy state.

CAUTION
Risk of service interruption
Deleting a volume can cause loss of service or data.

The deletevol command can be used only if the device is in a MBsy state. Deleting a volume can cause loss of service or data.

deletevol (continued)

Example

The following table provides an example of the deletevol command.

Example of the deletevol command
Example   **Task, response, and explanation**
deletevol core21 ↵   *where*   core21   specifies the volume name

Task: Delete a volume.
 	
 Response:
 ``` \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ **** WARNING **** ```
 `Deleting volume core21 on s00d will destroy the contents of`
 `the volume`
 `Do you want to continue?`
 `Please confirm ("YES" or "NO"):`
 `>yes`
 `Volume core21 has been deleted on s00d`
 `DISKADM:`
 	
 Explanation: You deleted volume core21 from s00d.

Responses

The following table provides explanations of the responses to the deletevol command.

Responses for the deletevol command
MAP output   **Meaning and action**

``` DELETEVOL command is aborted. ```
 ``` The command has been entered with the wrong parameters. ```
 ``` Check the SWERR generated and notify the appropriate people. ```

Meaning: The command encountered a software error and generated a SWERR. The command aborts.   **Action:**   Check the SWERR that is generated and inform the next level of support and ETAS of the problem. You may change the parameters and retry the command.
-continued-

deletevol (continued)

Responses for the deletevol command (continued)

MAP output	Meaning and action
| ```
DELETEVOL command is aborted.
The device has encountered a hardware error.
Check the log generated and notify the appropriate people.
``` | |
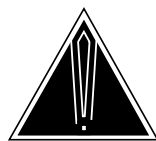| | **Meaning:** The device hardware is faulty. A log is generated indicating that there is a hardware error. The command aborts and the device is made system busy. |
| | **Action:** Check the log that is generated. Inform the next level of support. Maintenance personnel should attempt to return the device to service again. |
| ```
DELETEVOL is aborted.
The device has encountered an error in doing a read after a write to the
medium on the disk.
Check the log generated and notify the appropriate people.
``` | |
| | **Meaning:** The device encountered an error in reading a block immediately after writing the medium on the disk. A log is generated indicating that there is a disk read after write error. |
| | **Action:** Check the log that is generated. Inform the next level of support of the problem. |
| ```
DELETEVOL command is aborted.
The device has encountered an error in doing a read for verification
purposes on the medium on the disk.
Check the log generated and notify the appropriate people.
``` | |
	Meaning: The device encountered an error doing a read for verification purposes on the medium on the disk. A log is generated indicating that there is a disk verification error.
	Action: Check the log that is generated. Inform the next level of support of the problem.
-continued-	

deletevol (end)

Responses for the deletevol command (continued)

MAP output **Meaning and action**

```
DELETEVOL command is aborted.
The device has encountered an error in reading the medium on the disk.
Check the log generated and notify the appropriate people.
```

Meaning: The device encountered an error in reading the medium on the disk. A log is generated indicating that there is a disk medium error. The system then attempts to read the alternate system data structure. If the alternate is read successfully, the command does not abort. If the alternate is not read successfully, the command aborts.

Action: Check the log that is generated. Inform the next level of support of the problem.

```
DELETEVOL command is aborted.
Volume contains a registered bootable image file.
Enter HELP DELETEVOL FULL for more information.
```

Meaning: The volume cannot be deleted if it contains a file that is registered in an ITOC on the disk. The command aborts.

Action: Use the clearbootfl command in the DISKUT directory to remove the registration of the system load file from the ITOC.

```
DELETEVOL command is aborted. Volume <name> does not exist. Use
displayvols command to list the volumes.
```

Meaning: You specified a volume that does not exist. The command aborts.

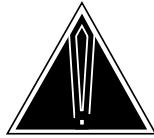**Action:** Reissue the command using an existing volume name.

End

displaydisk

Function

Use the displaydisk command to display information about the system load module (SLM) disk.

displaydisk command parameters and variables	
Command	**Parameters and variables**
displaydisk **dd**	There are no parameters or variables.

Qualification

> **CAUTION**
> **Must be manual busy (MBsy)**
> The commands in this directory can be used only if the device is in the MBsy state.

The displaydisk command can be used only if the device is in a MBsy state

displaydisk (continued)

Example

The following table provides an example of the displaydisk command.

Example of the displaydisk command

Example	Task, response, and explanation

displaydisk ↵

Task: Display information about the system load module (SLM) disk.

Response:
```
Disk drive information for SLM disk 1
-------------------------------------------------------------
DRIVE NAME:                       DSK1
VENDOR INFORMATION:               MAXTOR XT-8760S B5A
DATE LAST FORMATTED:              1989/11/23 17:19:22,065 WED
DATE LAST MODIFIED:               1989/11/23 17:20:23,812 WED
TOTAL BLOCKS ON DISK              1316242
TOTAL NUMBER OF BLOCKS FOR USE:   1316174
TOTAL NUMBER OF BLOCKS NOT USED:  8766
SIZE OF LARGEST UNUSED SEGMENT:   8766 BLOCKS
TOTAL NUMBER OF DEFECTIVE BLOCKS: 0
1 BLOCK = 512 bytes
-------------------------------------------------------------
```

Explanation: You see the system load module information.

Responses

The following table provides explanations of the responses to the displaydisk command.

Responses for the displaydisk command

MAP output	Meaning and action

```
DISPLAYDISK command is aborted.
The command has been entered with the wrong parameters.
Check the SWERR generated and notify the appropriate people.
```

Meaning: The command encountered a software error and generated a SWERR. The command aborts.

Action: Check the SWERR that is generated and inform the next level of support and ETAS of the problem. Change the parameters and retry the command.

-continued-

---

displaydisk (end)

Responses for the displaydisk command (continued)	
MAP output	**Meaning and action**
``` DISPLAYDISK command is aborted. The device has encountered a hardware error. Check the log generated and notify the appropriate people. ```	
	Meaning: The device hardware is faulty. A log is generated indicating the nature of the problem. The command aborts and the device is made system busy.
	Action: Check the log that was generated and inform the next level of support .
``` DISPLAYDISK command is aborted. The device has encountered an error in reading the medium on the disk. Check the log generated and notify the appropriate people. ```	
	Meaning: The device encountered an error in reading the medium on the disk. A log is generated indicating that there is a disk medium error. The system then attempts to read the alternate system data structure. If the alternate is read successfully, the command does not abort. If the alternate is not read successfully, the command aborts.
	Action: Check the log that was generated. Inform the next level of support of the problem.
End	

displayvols

Function

Use the displayvols command to display certain information about each volume on the system load module (SLM) disk currently in use.

displayvols command parameters and variables	
Command	**Parameters and variables**
displayvols **dv**	There are no parameters or variables.

Qualification

⚠	**CAUTION** **Must be manual busy (MBsy)** The commands in this directory can be used only if the device is in the MBsy state.

The displayvols command can be used only if the device is in a MBsy state.

Example

The following table provides an example of the displayvols command.

Example of the displayvols command	
Example	**Task, response, and explanation**
displayvols ↵	
	Task: Display information about each volume on the SLM disk currently in use.

	Response:
	<pre>Volume information for S00D
	Volume Create Modify Size Max. Max. No of
	Name Date Date Mega No of No of ITOC
	       Type Y/M/D Y/M/D Bytes Files Segments Files
	------ ---- ------ ------ ----- ----- -------- -----
	IMAGE1 STD 87/01/22 87/03/14 30 2048 2048 5
	IMAGE2 STD 86/11/21 87/03/11 3 16 50 0
	IMAGE3 STD 86/11/21 86/11/21 1 128 50 1</pre>

	Explanation: You see the SLM information.

displayvols (end)

Responses

The following table provides explanations of the responses to the displayvols command.

Responses for the displayvols command	
MAP output	**Meaning and action**
DISPLAYVOLS command is aborted. The command has been entered with the wrong parameters. Check the SWERR generated and notify the appropriate people.	
	Meaning: The command encountered a software error and generated a SWERR. The command aborts.
	Action: Check the SWERR that is generated and inform the next level of support and ETAS of the problem. You may change the parameters and retry the command.
DISPLAYVOLS command is aborted. The device has encountered a hardware error. Check the log generated and notify the appropriate people.	
	Meaning: The device hardware is faulty. A log is generated indicating that there is a hardware error. The command aborts and the device is made system busy.
	Action: Check the log that was generated and inform the next level of support. Maintenance personnel should attempt to return the device to service again.
DISPLAYVOLS command is aborted. The device has encountered an error in reading the medium on the disk. Check the log generated and notify the appropriate people.	
	Meaning: The device encountered an error in reading the medium on the disk. A log is generated indicating that there is a disk medium error. The system then attempts to read the alternate system data structure. If the alternate is read successfully, the command does not abort. If the alternate is not read successfully, the command aborts.
	Action: Check the log that was generated and inform the next level of support of the problem.

formatdisk

Function

Use the formatdisk command to format the system load module (SLM) disk. The format operation divides the tracks on the disk into sectors and assigns each sector a logical and sequential block number.

The disk is normally formatted only if it does not contain any active volumes. You may delete any active volumes with the deletevol command or you may override active volumes using the force parameter.

formatdisk command parameters and variables						
Command	**Parameters and variables**					
formatdisk **fd**	*disk_name*	*3* *sectors*	*16* *tracks*	g none	*noforce* force	*full* quick

Parameters and variables	Description
3	Omitting this entry forces the system to default to creating 3 sectors.
16	Omitting this entry forces the system to default to creating 16 tracks.
full	Omitting this entry forces the system to default to doing a full format. A full format creates tracks and sectors like the disk is brand new. A full format takes a long time to perform.
noforce	Omitting this entry forces the system to default to refusing to override active volumes, requiring you to delete any active volumes with the deletevol command.
disk_name	This variable identifies the SLM disk to format. Make the disk name as descriptive as possible to aid in distinguishing it from other disks and avoid disk names such as S01D which can be confused with the disk slot. The disk name may be a maximum of 17 characters long.
force	This parameter overrides any active volumes on the disk, which deletes all active volumes on the disk.
g	This parameter identifies the defective blocks discovered since the installation of the disk drive. Omitting this entry forces the system to default to indentifying the defective blocks discovered since the installation of the disk drive.
none	This parameter specifies that no defect list is requested.
quick	This parameter indicates that the disk is already initialized and you want to do a quick format procedure.
-continued-	

formatdisk (continued)

formatdisk command parameters and variables (continued)	
Parameters and variables	**Description**
sectors	This variable indicates the desired number of sectors. The valid entry range is 0-3.
tracks	This variable indicates the desired number of tracks. The valid entry range is 0-255.
End	

Qualifications

The formatdisk command is qualified by the following exceptions, restrictions, and limitations:

- The formatdisk command can be used only if the device is in a MBsy state.

CAUTION
Must be manual busy (MBsy)
The commands in this directory can be used only if the device is in the MBsy state.

- The formatdisk command removes all data stored on the disk. This could cause an interruption of service.

CAUTION
Risk of service interruption
The formatdisk command removes all data stored on the disk. This could cause an interruption of service.

- Insure that disk names accurately describe the disk. Avoid names such as S01D which can be confused with a disk slot.

Example

The following table provides an example of the formatdisk command.

Example of the formatdisk command	
Example	**Task, response, and explanation**
formatdisk **mydata** ↵ *where* mydata	specifies the disk name

	Task: Format a disk.
	
	Response:
	`          ***** WARNING *****`
	`Formatting of S00d will destroy the contents of the disk.`
	`The formatting will:`
	`    allocate 3 spare or alternate sectors per tracks`
	`    allocate 16 spare or alternate tracks per disk`
	`    use the G defect list`
	`    assign slm as the name for the disk`
	`    perform a full format,`
	`    exclude force option.`
	
	`Do you want to continue?`
	`Please confirm ("YES" or "NO"):`
	`>yes`
	`Formatting of disk has started.`
	`This may take 10 to 30 minutes.`
	`Formatting of disk has finished.`
	`<number> Defective Blocks were discovered.`
	
	Explanation: You see a warning message and the system prompts you to ensure that the correct parameters were specified. You see the progress of the execution, and the number of defective blocks that were discovered during format. This number includes the number of blocks specified in the defect list.

formatdisk (continued)

Responses

The following table provides explanations of the responses to the formatdisk command.

Responses for the formatdisk command	
MAP output	**Meaning and action**

`FORMATDISK command is aborted.`
`The command has been entered with the wrong parameters.`
`Check the SWERR generated and notify the appropriate people.`	

	Meaning: The command encountered a software error and generated a SWERR. The command aborts.
	Action: Check the SWERR that is generated and inform the next level of support of the problem. You may change the parameters and retry the command.

`FORMATDISK command is aborted.`
`The device has encountered a hardware error.`
`Check the log generated and notify the appropriate people.`	

	Meaning: The device hardware is faulty and a log is generated. The command aborts and the device is made system busy.
	Action: Check the log that is generated. Inform the next level of support. Maintenance personnel should attempt to return the device to service again.

`FORMATDISK is aborted.`
`The device has encountered an error in doing a read after a write to the`
`medium on the disk.`
`Check the log generated and notify the appropriate people.`	

	Meaning: The device encountered an error in reading a block immediately after writing the medium on the disk. A log is generated indicating that there is a disk read after write error. The command may or may not abort.
	Action: Check the log that is generated. Inform the next level of support of the problem.
-continued-	

formatdisk (end)

Responses for the formatdisk command (continued)

MAP output   Meaning and action

```
FORMATDISK command is aborted.
The device has encountered an error in doing a read for verification
purposes on the medium on the disk.
Check the log generated and notify the appropriate people.
```

> **Meaning:** The device encountered an error doing a read for verification purposes on the medium on the disk. A log is generated indicating that there is a disk verification error. The command may or may not abort.
>
> **Action:**   Check the log that is generated. Inform the next level of support of the problem.

```
FORMATDISK command is aborted.
The device has encountered an error in writing the medium on the disk.
Check the log generated and notify the appropriate people.
```

> **Meaning:** The device encountered an error in writing the medium on the disk and a log is generated. The command may or may not abort.
>
> **Action:**   Check the log that is generated. Inform the next level of support of the problem.

```
FORMATDISK command is aborted.
The disk contains a volume.
Disk cannot be formatted if it contains a volume.
```

> **Meaning:** The command aborts because volumes exist on the disk.
>
> **Action:**   Attempt to delete all volumes on the disk. If you encounter difficulties in deleting any of the volumes, contact the next level of support.

End

help

Function

Use the help command to receive online documentation for the DISKADM directory.

help command parameters and variables	
Command	**Parameters and variables**
help	*all* *command_nam*
Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
command_nam	This variable specifies a valid DISKADM directory command name. When the *command_nam* variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

help (end)

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help reintvol ↵ *where* reinitvol specifies the command name	
	Task: Access online documentation.
	Response: The REINITVOL (RV) command is useed to reinitialize a disk volume. This operation will delete all files on the volume, consolidating the space occupied by the files will the current free segments into a single freee segment, spanning the entire volume. This command will fail if the volume contains any system load files that are registered in the ITOC (Image Table Of Contents). These files must first be removed from the ITOC using the CLEARBOOTFL command. Parms: \<VOLUME NAME\> STRING
	Explanation: This example typifies a response for the help command string.

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

quit

Function

Use the quit command to exit the DISKADM directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	⌐ *1 level* ¬ │ all │ │ *name* │ └ *n_levels* ┘

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)	
Example	**Task, response, and explanation**
quit all ↵	
	Task: Exit from all levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory	
	Task: Exit from a specified directory without leaving any other directories.
	Response: AMADUMP>>> >
	Explanation: The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵	
	Task: Exit from a specified number of levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End	

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

reinitvol

Function

Use the reinitvol command to reinitialize the volume on the disk. The operation deletes all files on the specified volume and the space occupied by these files is returned to the free space map for the volume. All the segments of free space are grouped together to form one segment of free space covering the entire volume. The volume label is updated to indicate that the volume was modified.

Volumes can not be reinitialized if they contain system load files that are registered in the image table of contents (ITOC). These files must first be removed from the ITOC using the clearbootfl command in the DISKUT directory. The displayvols command displays files registered in any ITOC. To verify that these files are removed, use the listbootfl command in the DISKUT directory.

reinitvol command parameters and variables	
Command	**Parameters and variables**
reinitvol **rv**	*vol_name*
Parameters and variables	**Description**
vol_name	This variable indicates the name of the volume to be reinitialized. The volume name can have a maximum of eight characters.

Qualifications

	CAUTION **Must be manual busy (MBsy)** The commands in this directory can be used only if the device is in the MBsy state.

	CAUTION **Risk of service interruption** Reinitializing a volume can cause a loss of service or data.

The reinitvol command can be used only if the device is in a MBsy state. Reinitializing a volume can cause a loss of service or data.

reinitvol (continued)

Example

The following table provides an example of the reinitvol command.

Example of the reinitvol command	
Example	**Task, response, and explanation**
reinitvol core21 ↵ *where* core21 specifies the volume name	
	Task: Reinitialize a volume on the disk. **Response:** **** WARNING **** Re-initializing volume core21 on s00d will destroy the contents of the volume Do you want to continue? Please confirm ("YES" or "NO")? >yes VOLUME CORE21 HAS BEEN RE-INITIALIZED ON S00D. **Explanation:** You deleted all files on the volume core21. The space occupied by these files returns to the free space map for the volume.

Responses

The following table provides explanations of the responses to the reinitvol command.

Responses for the reinitvol command	
MAP output	**Meaning and action**
REINITVOL command is aborted. The command has been entered with the wrong parameters. Check the SWERR generated and notify the next level of support.	
	Meaning: The command encountered a software error and generated a SWERR. The command aborts. **Action:** Check the SWERR that is generated and inform the next level of support of the problem. You may change the parameters and retry the command.
-continued-	

reinitvol (continued)

Responses for the reinitvol command (continued)	
MAP output	**Meaning and action**
| ```
REINITVOL command is aborted.
The device has encountered a hardware error.
Check the log generated and notify the appropriate people.
``` | |
| | **Meaning:** The device hardware is faulty. A log is generated indicating that there is a hardware error. The command aborts and the device is made system busy. |
| | **Action:** Check the log that is generated. Inform the next level of support. Maintenance personnel should attempt to return the device to service again. |
| ```
REINITVOL is aborted.
The device has encountered an error in doing a read after a write to the
medium on the disk.
Check the log generated and notify the appropriate people.
``` | |
| | **Meaning:** The device encountered an error in reading a block immediately after writing the medium on the disk. A log is generated indicating that there is a disk read after write error. |
| | **Action:** Check the log that is generated. Inform the next level of support of the problem. |
| ```
REINITVOL command is aborted.
The device has encountered an error in doing a read for verification
purposes on the medium on the disk.
Check the log generated and notify the appropriate people.
``` | |
	Meaning: The device encountered an error doing a read for verification purposes on the medium on the disk. A log is generated indicating that there is a disk verification error.
	Action: Check the log that is generated. Inform the next level of support of the problem.
-continued-	

reinitvol (end)

Responses for the reinitvol command (continued)

MAP output	Meaning and action

```
REINITVOL command is aborted.
The device has encountered an error in reading the medium on the disk.
Check the log generated and notify the appropriate people.
```

Meaning: The device encountered an error in reading the medium on the disk and a log is generated. The system then attempts to read the alternate system data structure. If the alternate is read successfully, the command does not abort. If the alternate is not read successfully, the command aborts.

Action: Check the log that is generated. Inform the next level of support of the problem.

```
REINITVOL command is aborted.
The device has encountered an error in writing the medium on the disk.
Check the log generated and notify the appropriate people.
```

Meaning: The device encountered an error in writing the medium on the disk. A log is generated indicating that there is a disk medium error.

Action: Check the log that is generated. Inform the next level of support.

```
REINITVOL command is aborted.
Volume contains a registered bootable image file.
Enter HELP REINITVOL FULL for more information.
```

Meaning: The volume cannot be re-initialized if the volume contains a file that is registered in an ITOC on the respective disk. The command aborts.

Action: Use the clearbootfl command to remove the registration of the boot or system load file from the respective ITOC.

```
REINITVOL command is aborted.
Volume name does not exist.
Use DISPLAYVOLS command to list the volumes.
```

Meaning: You specified a volume that does not exist. The command aborts.

Action: Reissue the command specifying an existing volume name.

End

DISKUT level commands

Use the DISKUT level of the MAP to perform regular operations on the system load module (SLM), the volumes and files on the SLM disk, and the associated tape cartridge. The DISKUT level of the MAP stores image files on processors such as the message switch (MS) or the computing module (CM).

Most of the DISKUT directory commands operate on SLM disks only. The listfl, restorefl, backupfl, ejecttape, and inserttape commands apply to tapes also.

Accessing the DISKUT level

To access the DISKUT level, enter the following command from the CI level:

diskut ↵

```
            ⚠   CAUTION
                The SLM device must be in the OK state.
                The DISKUT directory commands can be used only if the
                SLM device is in the OK state.
```

The SLM device must be in the OK state.

DISKUT commands

The commands available at the DISKUT MAP level are described in this chapter and are arranged in alphabetical order. The page number for each command is listed in the following table.

DISKUT commands	
Command	Page
backup	D-203
clearbootfl	D-211
-continued-	

DISKUT commands (continued)	
Command	**Page**
clearvol	D-217
deletefl	D-221
duplicate	D-225
ejecttape	D-229
help	D-231
inserttape	D-233
listbootfl	D-237
listfl	D-241
listvols	D-245
quit	D-249
renamefl	D-253
restore	D-259
setbootfl	D-267
End	

backup

Function

Use the backup command to back up disk files onto tape cartridge. Backup copies of files are created by copying individual files or entire disk volumes.

backup command parameters and variables			
Command	**Parameters and variables**		
backup **ba**	file volume	*vol_name* *disk_fname* *vol_name* *tape_fname2*	*tape_fname1*

Parameters and variables	**Description**
disk_fname	This variable specifies the name of the disk file being archived. If this file name is an STD file, the volume name must be specified. If it is a fault tolerant file system (FTFS) file, the volume name may be omitted. The file name for an FTFS file is prefixed by a colon(:), and replaced by an FTFS path name.
file	This parameter indicates that a single file is copied to a tape.
tape_fname1	This variable specifies the name of the tape file and can be omitted. The file name can have a maximum of 32 characters and 17 characters for system image files.
tape_fname2	This variable specifies the name of the file to copy from disk to tape. The file name can have a maximum of 32 characters and 17 characters for system image files. This parameter can not be used for FTFS files.
vol_name	This variable specifies the name of the system load module (SLM) disk volume. The first four characters must be s00d or s01d for SLM disks 0 and 1. The remaining characters indicate the volume name. The volume name can have a maximum of 12 characters.
volume	This parameter indicates that all the files on a disk volume should be copied to a tape; however, this option is not supported for FTFS files.

Qualifications

The backup command is qualified by the following exceptions, restrictions and limitations:

- Both disk and tape cartridge devices must be in the same system load module (SLM) unit.
- A disk volume that is being backed up to tape must not contain more than 256 files and must not contain any open files.
- The tape to which disk files are copied must be mounted using the inserttape command.

backup (continued)

- Unless otherwise noted, backup of single FTFS files is supported.
- SLM can not be used while the backup command is in progress.

> **CAUTION**
> **Risk of service interruption**
> Files contained in backup volumes are not individually
> accessible on tape. If backed up by volume, these image files
> are usable only to restore archived files back to disk but the
> tape can not be booted. This could extend a short outage into
> a long one; therefore, it is recommended that the "filename"
> option of the backup command be used when backing up
> bootable image files for computing module (CM) and
> message switch (MS) to SLM tape.

Files contained in backup volumes are not individually accessible on tape.
If backed up by volume, these image files are usable only to restore archived
files back to disk but the tape can not be booted. This could extend a short
outage into a long one; therefore, it is recommended that the "filename"
option of the backup command be used when backing up bootable image
files for computing module (CM) and message switch (MS) to SLM tape.

Examples

The following table provides examples of the backup command.

Examples of the backup command
Example   **Task, response, and explanation**
backup   **file s00dvol1 myfile** ↵ *where* s00dvol1   specifies the volume name myfile   specifies the disk file name

Task: Copy a file from disk to tape.
 	
Response:
`Disk file is opened.`
`Tape file is created.`
`The copy operation may take several minutes.`
`Tape file is closed`
`Disk file is closed.`
`Disk file MYFILE in volume S00DVOL1 has been copied to tape.`
 	
Explanation: You copied the file myfile on volume s00dvol1 of SLM disk 0.

-continued-

backup (continued)

Examples of the backup command (continued)

Example   **Task, response, and explanation**

---

backup volume s01dvol1 myvolume ↵
where

s01dvol1   specifies the volume name
myvolume   specifies the tape file name

> **Task:**   Copy all the files from a disk volume to tape.
>
> **Response:**
> Tape file is created.
> The copy operation may take several minutes.
> Tape file is closed.
> n files from Volume S01DVOL1 have been copied to tape.
> The backup copy of the volume is called MYVOLUME.
>
> **Explanation:**   You copied all the files in volume s01dvol1 of SLM disk 1 to a tape
>   file named myvolume.

---

backup   file :/s00dftfs1/dira/dirb/myfile myfile&bak ↵
where
:/s00dftfs1/dira/dirb/myfile specifies the FTFS file path
myfile&bak   specifies the tape file name

> **Task:**   Backup an FTFS file.
>
> **Response:**   DISK FILE IS OPEN
>   TAPE FILE IS CREATED
>   DISK FILE IS COPIED TO MYFILE$BAK ON TAPE
>   DISK FILE IS CLOSED
>   TAPE FILE IS CLOSED
>
> **Explanation:**   You successfully backed up an FTFS file.

End

Responses

The following table provides explanations of the responses to the backup
command.

backup (continued)

Responses for the backup command	
MAP output	**Meaning and action**
`Attempt to write to read-only device.`	
	Meaning: You specified a disk drive or a tape cartridge that is write-protected. The command aborts.
	Action: If necessary, see if the write-protection can be safely removed.

`BACKUP command is aborted`
`Tape is not inserted for this unit.`
`You must insert the tape before issuing backup command.`	

	Meaning: You specified a tape drive where the tape is not inserted. The command aborts.
	Action: Use the inserttape command and reenter the backup command.
`BACKUP OF MULTIPLE FTFS FILES IS NOT YET SUPPORTED`	
	Meaning: You tried to backup an FTFS volume when only a single FTFS file can be archived. The command aborts.
	Action: You must backup each individual file for an FTFS volume.
`Device does not exist.`	
	Meaning: You specified a device name or a volume name that does not exist. The command aborts.
	Action: Reissue the command specifying the correct device.
`Device error.`	
	Meaning: You specified a device that encountered an input or output error. The command aborts.
	Action: Check the integrity of the disk data structures and files.
`Device is in use.`	
	Meaning: You specified a device that is being used by another process. The command aborts.
	Action: Reissue the command after the other process is finished using the device.
-continued-	

backup (continued)

Responses for the backup command (continued)	
MAP output	**Meaning and action**
`DIRECTORY FILES CANNOT BE BACKED UP`	
	Meaning: You tried to archive an FTFS directory file. These files can not be backed up. The command aborts.
	Action: None

`DISK FILE IS OPEN`
`TAPE FILE IS CREATED`
`DISK FILE IS COPIED TO <tape-name> ON TAPE`
`DISK FILE IS CLOSED`
`TAPE FILE IS CLOSED`	

	Meaning: You completed the backup successfully.
	Action: None
`End of file medium met.`	
	Meaning: You reached the end of the tape. The command aborts.
	Action: Insert a new tape.
`ERROR CALCULATING TAPE FILE SIZE.`	
	Meaning: The system calculations for the requested backup indicate an overflow will occur. The backup command aborts.
	Action: Report the problem to the next level of maintenance.
`ERROR CLOSING DISK FILE` `FILE SYSTEM ERROR <system-error-text>`	
	Meaning: You specified a disk file that failed to close. Previous faults during backup may have led to this fault. The backup command continues execution.
	Action: Verify the status of the SLM disk and the integrity of the disk file.
-continued-	

backup (continued)

Responses for the backup command (continued)	
MAP output	**Meaning and action**
`ERROR CLOSING TAPE FILE` `FILE SYSTEM ERROR: <system-error-text>`	
	Meaning: You specified a tape file that failed to close. Previous faults during backup may have led to this fault. The backup continues execution.
	Action: Verify the status of the SLM tape and the integrity of the tape file.
`ERROR COPYING DISK FILE TO TAPE`	
	Meaning: The system encountered an error while writing the disk file to the tape. The newly created tape file is deleted. The command aborts.
	Action: Check the tape cartridge for faults and see if the system load module (SLM) has gone system-busy. Contact next level of maintenance for further help.
`ERROR CREATING TAPE FILE`	
	Meaning: You specified a tape file for backup that the system could not create. The command aborts.
	Action: Verify write-access to the tape and correct insertion of the tape.
`ERROR CREATING USER HEADER LABEL`	
	Meaning: You specified a tape file but the system could not create the header label. The newly created tape file is deleted. The command aborts.
	Action: Report the error to the next level of maintenance.
`ERROR OPENING DISK FILE` `FILE SYSTEM ERROR: <system-error-text>`	
	Meaning: You specified an FTFS file that the system can not open. Possible causes are denied file access, or a file not found error. The command aborts.
	Action: Determine the cause of the system error and retry the command.
`File does not exist.`	
	Meaning: You specified a file that does not exist. The command aborts.
	Action: Reissue the command specifying the correct file name.
-continued-	

backup (continued)

Responses for the backup command (continued)
MAP output   **Meaning and action**

File name too long for device.

>**Meaning:** You specified a file name that exceeds 17 or 32 characters. The command aborts.
>
>**Action:**   Reissue the command specifying a shorter file name.

Illegal file system operation requested.

>**Meaning:** You requested an operation that is illegal in the context of the command. The command aborts.
>
>**Action:**   Reissue the command correctly.

Medium error.

>**Meaning:** You specified a device that encountered an error in reading or writing to the medium on the disk. The command aborts.
>
>**Action:**   Contact the next higher level of support.

Unit attention: tape is no longer available.

>**Meaning:** You specified a cartridge tape that was removed or replaced after the command was entered. The command aborts.
>
>**Action:**   Reinsert the cartridge tape.

Volume does not contain any files.

>**Meaning:** You specified an empty volume. The command aborts.
>
>**Action:**   None

Volume incorrectly formatted.

>**Meaning:** You specified a tape volume that is not stored in the correct format. The command aborts.
>
>**Action:**   Erase the contents of the tape.

-continued-

backup (end)

Responses for the backup command (continued)

MAP output   **Meaning and action**

```
Volume not mounted.
```

Meaning: You specified a tape cartridge that is not mounted. The command aborts.

Action: Mount the tape cartridge and reissue the command.

```
******* WARNING *******
Files or volumes on S00D (S01D) device can
be backed up only to S00T (S01T) cartridge.

Backup command is aborted
Please reinsert the cartridge tape for the appropriate
SLM device, before issuing BACKUP command
```

Meaning: You specified an empty tape device. The command aborts.

Action: Mount the tape, issue the inserttape command, and reenter the backup command.

End

clearbootfl

Function

Use the clearbootfl command to clear all or any of the system load files registered in the image table of contents (ITOC) of the system load module (SLM) disk. You can not boot or load those files to a target device.

The clearbootfl command can also be used to delete the default load file. The default load file is created when the load image is generated using the dump command with the update parameter.

clearbootfl command parameters and variables

Command	Parameters and variables				
clearbootfl **cbf**	*devname*	*boot_table*	all active file		
			file	*vol_name*	*file_name*

Parameters and variables	Description
active	This parameter specifies only the active boot entry in the boot table. The active boot entry is the default load file.
all	This parameter specifies all entries in the boot table. This includes the active boot entry.
boot_table	This variable specifies the target device ITOC. The boot table name can be CM (computing module) or MS (message switch). The valid entry values are cm and ms.
devname	This variable specifies the SLM device. The valid entry values are s00d or s01d.
file	This parameter specifies a single file to clear from the boot table.
file_name	This variable specifies the name of the file to clear from the ITOC. The file name has a maximum of 17 characters.
vol_name	This variable specifies the name of the volume where the file is located. The first four characters of the volume name specify the device (s00d, s01d, s00t or s01t). The last eight characters specify the volume.

clearbootfl (continued)

Qualification

CAUTION
Risk of service interruption
The clearbootfl command creates a potential for loss of
service or data.

The clearbootfl command creates a potential for loss of service or data.

Examples

The following table provides examples of the clearbootfl command.

Examples of the clearbootfl command	
Example	**Task, response, and explanation**
clearbootfl s00d cm file s00dloadnew image21 ↵ *where* s00d cm s00dloadnew image21	 specifies the device name specifies the boot table specifies the volume name specifies the file name

	Task: Clear a file in the ITOC from the CM.
	
	Response:
	`Volume S00DLOADNEW file IMAGE21 has been cleared in`
	`Image Table Of Contents for CM on SLM 0`
	`DISKUT:`
	
	Explanation: You cleared the system load file image21 in volume s00dloadnew
	in the ITOC for the CM on SLM 0.

-continued-	

clearbootfl (continued)

Examples of the clearbootfl command (continued)	
Example	**Task, response, and explanation**
clearbootfl s01d ms active ↵ *where* s01d specifies the device name ms specifies the boot table	

	Task:	Clear the active boot file for the MS.

	Response:	`The active boot file has been cleared for`
		 `Image Table Of Contents for MS on SLM 1`
		 `DISKUT:`

	Explanation:	You cleared the active boot file for the ITOC from the MS on SLM 1.

clearbootfl s01d ms all ↵ *where* s01d specifies the device name ms specifies the boot table	

	Task:	Clear the entire ITOC for the MS.

	Response:	`The whole Image Table of Contents for MS`
		 `on SLM 1 has been cleared.`
		 `DISKUT:`

	Explanation:	You cleared the entire ITOC from the MS on SLM 1.

End

Responses

The following table provides explanations of the responses to the clearbootfl command.

clearbootfl (continued)

Responses for the clearbootfl command	
MAP output	**Meaning and action**
`CLEARBOOTFL command is aborted. Device does not exist`	
	Meaning: You specified a device that does not exist. The command aborts.
	Action: Reissue the command with the correct device name.
`CLEARBOOTFL command is aborted. Device IO error`	
	Meaning: You specified a device that encountered an input or output error. The command aborts leaving the input or output in an unfinished state.
	Action: Check the integrity of the file being cleared. If the integrity of the file has been lost, follow maintenance procedures.
`CLEARBOOTFL command is aborted. Device is in use`	
	Meaning: You specified a device that is being used by another process. The command aborts.
	Action: Wait for the other process to complete. Reissue the command.
`CLEARBOOTFL command is aborted. Device is not ready`	
	Meaning: You specified a device that is not ready or available. The command aborts.
	Action: Use the maintenance commands to return the device to service. Reissue the command.
`CLEARBOOTFL command is aborted. File does not exist`	
	Meaning: You specified a file that does not exist. The command aborts.
	Action: Display the files on the volume using the command listfl. Reissue the command with the correct file name.
`CLEARBOOTFL command is aborted. ITOC file does not exist`	
	Meaning: You specified a file that is not in the ITOC. The command aborts.
	Action: Reissue the command with the correct file name.
-continued-	

clearbootfl (end)

Responses for the clearbootfl command (continued)	
MAP output	**Meaning and action**
`CLEARBOOTFL command is aborted. ITOC table does not exist`	
	Meaning: You specified an image table of contents (ITOC) table that does not exist. The command aborts.
	Action: Reissue the command with the correct ITOC table name.
`CLEARBOOTFL command is aborted. Volume does not exist`	
	Meaning: You specified a volume that does not exist. The command aborts.
	Action: Reissue the command with the correct volume name.
End	

clearvol

Function

Use the clearvol command to delete all files on a system load module (SLM) disk volume, and all erasable files on fault tolerant file systems (FTFS) volumes. The volume must not contain any open files or files that are registered in an image table of contents (ITOC).

clearvol command parameters and variables	
Command	**Parameters and variables**
clearvol **cvol**	*vol_name*

Parameters and variables	Description
vol_name	This variable specifies the SLM or FTFS disk volume. The first four characters must be s00d or s01d for SLM disks 0 or 1. The remaining characters specify the volume name. The volume name must not exceed 12 characters.

Qualification

> ⚠ **CAUTION**
> **Risk of service interruption**
> The clearvol command erases files from the disk. As a result, all data is lost.

The clearvol command erases files from the disk. As a result, all data is lost.

Examples

The following table provides examples of the clearvol command.

clearvol (continued)

Examples of the clearvol command	
Example	**Task, response, and explanation**
clearvol s00dvol1 ↵ *where* s00dvol1 specifies the volume name	
	Task: Clear all the files in a volume.

	Response: `Do you want to clear Volume S00DVOL1 (Yes/No)?`
	`>yes`
	`Volume S00DVOL1 has been cleared.`

	Explanation: You cleared all the files in volume s00dvol1 on SLM 0.
clearvol s00dmyvolume ↵ *where* s00dmyvolume specifies the volume name	
	Task: Clear a volume of all files to which you have write access.
	Response: `CLEARVOL OF S00DMYVOLUME COMPLETE`
	Explanation: You cleared all files to which you have write access on volume s00dmyvolume on SLM 0.

Responses

The following table provides explanations of the responses to the clearvol command.

Responses for the clearvol command	
MAP output	**Meaning and action**
`CLEARVOL OF <volume_nm> COMPLETE`	
	Meaning: You cleared the volume successfully. You deleted all files and directories to which you had write access. You deleted those directories that have no file or directory entries left in their list.
	Action: None
-continued-	

clearvol (continued)

Responses for the clearvol command (continued)

MAP output	Meaning and action

COULD NOT DETERMINE ROOT DIRECTORY OF <volume_nm> FILE
SYSTEM ERROR: <system-error-text>

Meaning: You tried to clear an FTFS volume, but the system could not find the root directory for the specified volume. The command aborts.

Action: Contact the next level of maintenance.

Device does not exist.

Meaning: You specified a device or a volume that does not exist. The command aborts.

Action: Reissue the command specifying the correct device.

Device error.

Meaning: You specified a device that encountered an input or output error. The command aborts.

Action: Check the integrity of disk data structures and files being operated on.

Illegal file system operation requested.

Meaning: You requested an operation that is illegal in the context of the command. The command aborts.

Action: Check the syntax and reissue the command correctly.

Medium error.

Meaning: You specified a device that encountered an error in reading or writing to the medium on the disk. The command aborts.

Action: Consult the maintenance procedures for the disk.

Volume contains boot files that are registered in an ITOC.

Meaning: You tried to delete files on an SLM disk volume containing files registered in an ITOC. The command aborts.

Action: Remove all files registered in an ITOC before reissuing the command.

-continued-

clearvol (end)

Responses for the clearvol command (continued)	
MAP output	**Meaning and action**
`Volume contains open files.`	
	Meaning: You tried to delete files on an SLM disk volume which contains open files. The command aborts.
	Action: Close all the files in the volume.
`Volume <name> has been cleared.`	
	Meaning: You executed the command successfully. The file directory (FD) and volume free space map (VFSM) are re-initialized. The FD contains no entries. The VFSM contains one entry indicating that all the space on the volume is available for storing files.
	Action: None

`*** WARNING ***`
`Clearing a volume will destroy all files stored on the volume.`
`Do you want to clear Volume name (Yes/No)?`	

	Meaning: You must confirm the command before the system proceeds.
	Action: Enter yes to execute the command. Enter no to abort the command.
End	

deletefl

Function

Use the deletefl command to remove a file irrevocably from a volume on the system load module (SLM) disk or fault tolerant file system (FTFS) disk. If the file is marked as bootable and is registered in the image table of contents (ITOC) it can not be deleted. It must first be removed from the ITOC using the clearbootfl command. The file can not be deleted if it is marked as a read-only or non-erasable file.

deletefl command parameters and variables	
Command	**Parameters and variables**
deletefl **ddf**	*vol_name* *file_name*

Parameters and variables	**Description**
file_name	This variable is the name of the file to delete. If the file name is a standard (STD) file, the volume name must be specified. If the file name is an FTFS file, the volume name may be omitted. The file name for an FTFS file is prefixed by a colon(:) and replaced by an FTFS path name.
vol_name	This variable specifies the SLM or FTFS volume name. You may omit the volume name for FTFS only. The first four characters indicate the device. The disk devices are s00d and s01d. The last eight characters specify the name of the volume on the disk.

Qualifications

None

deletefl   (continued)

Examples

The following table provides examples of the deletefl command.

Examples of the deletefl command	
Example	**Task, response, and explanation**
deletefl s00dnewload myfile ↵ *where* s00dnewload specifies the volume name myfile specifies the file name	

	Task: Delete a file from a volume.
	
	Response:
	`File MYFILE has been deleted in volume S00DNEWLOAD`
	`DISKUT:`
	
	Explanation: You deleted the file myfile from the volume s00dnewload on SLM 0.

deletefl s00dstd1 myfile ↵ *where* s00dstd1 specifies the volume name myfile specifies the file name	
	Task: Delete an STD file. **Response:** `MYFILE DELETED` **Explanation:** You deleted the STD file myfile from volume s00dstd1 on SLM 0.
deletefl :/s00dftfs1/a/b/lastdir/myfile ↵ *where* :/s00dftfs1/a/b/lastdir/myfile specifies an FTFS file name	
	Task: Delete an FTFS file. **Response:** `/S00DFTFS1/A/B/LASTDIR/MYFILE DELETED` **Explanation:** You deleted the FTFS file :/s00dftfs1/a/b/lastdir/myfile.

deletefl   (continued)

Responses

The following table provides explanations of the responses to the deletefl command.

Responses for the deletefl command	
MAP output	**Meaning and action**
COULD NOT DELETE <file_nm> FILE SYSTEM ERROR: <system-error-text>	
	Meaning: You failed to delete the file. Possible causes are denied file access, or a mistyped file name. The command aborts.
	Action: Determine the cause of the system error, and retry the command.
DELETEFL command is aborted. Device does not exist	
	Meaning: You specified a device that does not exist. The command aborts.
	Action: Reissue the command with the correct device.
DELETEFL command is aborted. Device IO error	
	Meaning: You specified a device that encountered an input or output error. The command aborts. The input or output is left in an unfinished state.
	Action: Verify the integrity of the file. Follow maintenance procedures if the integrity of the file has been lost.
DELETEFL command is aborted. Device is not ready	
	Meaning: You specified a device that is not ready or available. The command aborts.
	Action: Use the maintenance commands to return the device to service. Reissue the command.
DELETEFL command is aborted. File does not exist	
	Meaning: You specified a file that does not exist. The command aborts.
	Action: Use the listfl command to display the files on the volume. Reissue the command with the correct file name.
-continued-	

deletefl (end)

Responses for the deletefl command (continued)	
MAP output	**Meaning and action**
DELETEFL command is aborted. File is a registered bootable image file. Enter HELP DELETEFL FULL for more information.	
	Meaning: You can not delete a file if it is registered in an ITOC on the disk. The command aborts.
	Action: Use the clearbootfl command to remove the registration of the boot or system load file from the ITOC. Reenter the command.
DELETEFL command is aborted. Volume does not exist	
	Meaning: You specified a volume that does not exist. The command aborts.
	Action: Reissue the command specifying the correct volume.
<file_nm> DELETED	
	Meaning: You deleted the specified file.
	Action: None
End	

duplicate

Function

Use the duplicate command to create an exact copy of a system load module (SLM) file. The file type can be either standard (STD) or fault tolerant file system (FTFS). Files can be duplicated across volumes and across volume types, with certain restrictions.

duplicate command parameters and variables			
Command	**Parameters and variables**		
duplicate	*src_vn* *src_fn* *dst_vn* *dst_fn*		
Parameters and variables	**Description**		
dst_fn	This variable defines the name of the destination file. If the destination file is an STD file, then the destination volume must be specified. If the destination file is an FTFS file, then the volume may be omitted. The destination file name for an FTFS file is prefixed by a colon(:) and replaced by an FTFS path name.		
dst_vn	This variable defines the volume name of the destination file.		
src_fn	This variable defines the name of the source file. If the source file is an STD file, the source volume must be specified. If the source file is an FTFS file, the source volume may be omitted. The source file for an FTFS file is prefixed by a colon(:) and replaced by an FTFS path name.		
src_vn	This variable defines the volume name of the source file.		

Qualifications

The duplicate command is qualified by the following exceptions, restrictions and limitations:

- The command allows the following transitions between volumes without restrictions:
    - STD -> STD
    - FTFS -> FTFS
- Restrictions apply as noted for the following transitions:
    - STD -> FTFS : if the STD file has fixed record format.
    - FTFS -> STD : if the FTFS file was originally created by the FILESYS (STD) interface.
- If a fatal error occurs during the duplication process, the destination file is deleted.

duplicate (continued)

Example

The following table provides an example of the duplicate command.

Example of the duplicate command	
Example	**Task, response, and explanation**
duplicate s00dstdvol myfile :/s00dftfsvol/myfiledup ↵ *where* s00dstdvol specifies the STD source volume name myfile specifies the STD source file name :/s00dftfsvol/myfiledup specifies the FTFS destination file path	
Task: Duplicate an STD file to an FTFS file. **Response:** DUPLICATION COMPLETE **Explanation:** You duplicated the STD file myfile on the volume s00dstdvol to an FTFS file.	

Responses

The following table provides explanations of the responses to the duplicate command.

Responses for the duplicate command	
MAP output	**Meaning and action**
CANNOT ALLOCATE STORE FOR STORAGE BUFFER	
	Meaning: The system could not allocate temporary storage for the copy buffer. The command aborts. **Action:** Retry the command later.
COPY COMPLETED, BUT COULD NOT COPY BLACKBOX 2 or WARNING: COULD NOT SET PARENT DIRECTORY IN BLACK BOX 2	
	Meaning: The system encountered an error while copying internal black box information. The command completes. **Action:** Check the status of the SLM disk drive. Contact the next level of maintenance if necessary.
-continued-	

duplicate (continued)

Responses for the duplicate command (continued)	
MAP output	**Meaning and action**
DIRECTORIES CANNOT BE DUPLICATED	
	Meaning: You tried to duplicate an FTFS directory. This activity is not supported. The command aborts.
	Action: None
DUPLICATION COMPLETE	
	Meaning: You duplicated the file as specified.
	Action: None
ERROR CLOSING INPUT FILE or ERROR CLOSING <src_fn>	
	Meaning: The system encountered an error while closing the source files. The command aborts.
	Action: Check the status of the SLM disk drive. Contact the next level of maintenance if necessary.
ERROR CLOSING OUTPUT FILE or ERROR CLOSING <dst_fn>	
	Meaning: The system encountered an error while closing the destination file. The command completes execution.
	Action: Check the status of the SLM disk drive and the destination file. Contact the next level of maintenance if necessary.
ERROR CREATING OUTPUT FILE FILE SYSTEM ERROR: <system_error_text>	
	Meaning: The system encountered an error while creating the specified output file. Possible causes are denied file access, or a mistyped file name. The command aborts.
	Action: Determine the cause of the system error. Retry the command, or contact the next level of maintenance.
-continued-	

duplicate (end)

Responses for the duplicate command (continued)

MAP output   **Meaning and action**

```
ERROR IN INPUT FILE - RECORD NUMBER ####
or
ERROR READING <src_filename> AT OFFSET #### ####
```

	Meaning: The system encountered an error while reading the source file. The command aborts.
	Action: Check the status of the SLM disk drive. Contact the next level of maintenance if necessary.

```
ERROR IN OUTPUT FILE - RECORD NUMBER ####
or
ERROR WRITING TO <dst_fn> AT OFFSET #### ####
```

	Meaning: The system encountered an error while writing to the new file. The command aborts.
	Action: Check the status of the SLM disk drive. Contact the next level of maintenance if necessary.

```
ERROR OPENING INPUT FILE
FILE SYSTEM ERROR: <system_error_text>
```

	Meaning: The system failed to open the specified input file. Possible causes are denied file access, or a mistyped file name. The command aborts.
	Action: Determine the cause of the system error. Retry the command, or contact the next level of maintenance.

```
FAILED TO GET LAST COMPONENT OF dst_fn
FILE SYSTEM ERROR: <system_error_text>
```

	Meaning: The system encountered an error in the destination file name. The command aborts.
	Action: Retry the command with a valid destination file name.

End

ejecttape

Function

Use the ejecttape command to signal to the file system that the tape device is no longer available and that the tape cartridge is ready to be removed from the drive.

ejecttape command parameters and variables	
Command	**Parameters and variables**
ejecttape **et**	*device_name*
Parameters and variables	**Description**
device_name	This variable indicates the name of the system load module (SLM) device to use. The valid entry values are s00t and s01t.

Qualifications

None

Example

The following table provides an example of the ejecttape command.

Example of the ejecttape command		
Example	**Task, response, and explanation**	
ejecttape s00t ↵ *where*		
s00t	specifies the device name	
	Task:	Prepare a tape for removal.
	Response:	`The EJECT operation may take up to 5 minutes to position the tape to the beginning.` `EJECTTAPE command aborted.` `Tape device S00T on node CM is no longer inserted.`
	Explanation:	This command prepares the tape cartridge on SLM number 0 for removal.

ejecttape (end)

Responses

The following table provides explanations of the responses to the ejecttape command.

Responses for the ejecttape command	
MAP output	**Meaning and action**
`EJECTTAPE command is aborted. Device does not exist`	
	Meaning: You specified a device that does not exist. The command aborts.
	Action: Reissue the command specifying the correct device.
`EJECTTAPE command is aborted. Device IO error`	
	Meaning: You specified a device that encountered an input or output error. The command aborts and leaves the input or output left in an unfinished state.
	Action: Follow maintenance procedures.
`EJECTTAPE command is aborted. Device is in use`	
	Meaning: You specified a device that is being used by another process. The command aborts.
	Action: Reissue the command when the device is no longer in use.

`EJECTTAPE command is aborted.`
`The device is not a SLM tape device.`
`Enter HELP EJECTTAPE FULL for more information.`	

	Meaning: You specified an invalid device. The command aborts.
	Action: Reissue the command with s00t or s01t.

help

Function

Use the help command to receive online documentation for the DISKUT directory.

help command parameters and variables	
Command	**Parameters and variables**
help	_all_ _command_nam_

Parameters and variables	Description
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
_command_nam_	This variable specifies a valid DISKUT directory command name. When the _command_nam_ variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

help (end)

Example of the help command	
Example	**Task, response, and explanation**
help **listfl** ↵ *where* listfl specifies a command name	
	Task: Access online documentation.

	Response: `The LISTFL (LF) command is used to display`
	`information about files on both the disk and tape`
	`devices.`
	`In order to list files on a remote volume (a`
	`volume that resides on a node different from the`
	`node that DISKUT is running on), the user must`
	`specify the remote volume name as well as the`
	`remote node name.`
	
	`Once a valid parameter or parameter combination`
	`has been accepted by the LF command, all`
	`subsequent input by the user will be ignored.`
	
	`Parms: [<LOCAL VOLUME NAME> DEVICE name]`
	`       [<FTFS PATHNAME> FILE name]`
	`       [<REMOTE VOLUME NAME> STRING]`
	`       [<REMOTE NODE NAME> STRING]`

	Explanation: This example typifies a response for the help command string.

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
`MODULE NOT LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.`	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

inserttape

Function

Use the inserttape command to signal the file system that the tape cartridge has been placed in the tape drive and is available for use. You can read, check or create a tape label. When this command is executed, you can write onto the tape or read from it. The tape is positioned at the first file on the tape, or positioned to write the first file.

inserttape command parameters and variables		
Command	**Parameters and variables**	
inserttape **it**	*device_name*	readlabel checklabel *label_name* writelabel *label_name*

Parameters and variables	**Description**
readlabel	This default parameter attempts to read the label on the tape. If the tape label is not found within a certain time, a timeout occurs and you are informed. No checking is done. Omitting this entry forces the system to default to reading the tape label.
checklabel	This parameter attempts to read the specified label on the tape. The tape is wound to the beginning of the tape (BOT) and scanned for the label. When the label is read, it is checked against the label specified in the command. If no match occurs, an error is reported.
device_name	This variable indicates the system load module (SLM) device used. The valid entry values are s00t or s01t.
label_name	This variable specifies the name of the label to write or read on the tape. The label name can have a maximum of six characters.
writelabel	This parameter writes a new label to the tape after rewinding the tape to BQTAny existing files on the tape are destroyed, and the tape is positioned to write the first file immediately after the label. The label name can have a maximum of six characters.

Qualifications

None

inserttape (continued)

Examples

The following table provides examples of the inserttape command.

Examples of the inserttape command	
Example	**Task, response, and explanation**
inserttape s00t writelabel load52 ↵ *where* s00t specifies the device name load52 specifies the label name	
	Task: Assign a label name.
	Response: `A tape is available in SLM 0` `Name LOAD52 has been written to the tape label.`
	Explanation: You placed a tape in SLM drive 0 and assigned the label name load52.
inserttape s00t checklabel load52 ↵ *where* s00t specifies the device name load52 specifies the label name	
	Task: Verify a label name.
	Response: `Tape LOAD52 is available in SLM 0` `DISKUT:`
	Explanation: You verified the label successfully. The tape labeled load52 is on tape drive 0.
-continued-	

inserttape (continued)

Examples of the inserttape command (continued)
Example   Task, response, and explanation

inserttape   s00t checklabel load52 ↵
where

s00t	specifies the device name
load52	specifies the label name

	Task:	Verify a label name.
	Response:	INSERTTAPE command is aborted. Tape LOAD52 is not available in SLM 0 the name on the tape is LOAD51 DISKUT:
	Explanation:	You verified the label unsuccessfully. The tape on tape drive 0 has a different label. The command aborts.

inserttape   s01t ↵
where

s01t	specifies the device name

	Task:	Read a tape label.
	Response:	The INSERT operation may take up to 5 minutes to tension the tape. Tape RTPB, unit 0, is now availabel to user on node CM.
	Explanation:	You read the tape label on tape drive 1 and the tape is available.

End

Responses

The following table provides explanations of the responses to the inserttape command.

Responses for the inserttape command
MAP output   Meaning and action

INSERTTAPE command is aborted. Device does not exist
Meaning: You specified a device that does not exist. The command aborts.
Action: Reissue the command specifying the correct device name.
-continued-

inserttape (end)

Responses for the inserttape command (continued)	
MAP output	**Meaning and action**
`INSERTTAPE command is aborted. Device IO error`	
	Meaning: You searched for the tape label, but either a timeout occurred or the end of the tape was reached before the label was found. The command aborts.
	Action: Follow maintenance procedures or refer to field support.
`INSERTTAPE command is aborted. Device is in Use`	
	Meaning: You specified a device that is being used by another process. The command aborts.
	Action: Reissue the command when no other process is using the device.
`INSERTTAPE command is aborted. Tape is not inserted`	
	Meaning: The tape cartridge is not inserted. The command aborts.
	Action: Insert the tape and reissue the command.

`INSERTTAPE command is aborted.`
`Tape <name> is not available in SLM <number>`
`The name on the tape is <name>.`	

	Meaning: You read the label on the tape but it is not the tape you specified. The command aborts.
	Action: Remove the tape from the drive and insert the correct tape, or reissue the command using the label name on the existing tape.
End	

listbootfl

Function

Use the listbootfl command to list all the files registered in the image table of contents (ITOC) of a system load module (SLM) disk. These are the system load files that can be loaded or booted into their target device. Each target device has its own ITOC. This command also displays the default boot file (the active boot file) for the target device.

listbootfl command parameters and variables		
Command	**Parameters and variables**	
listbootfl **lbf**	*device_name* *boot_table*	active all

Parameters and variables	**Description**
<u>active</u>	This default parameter displays entry zero in the boot table. This entry contains the active system load. Omitting this entry forces the system to default to displaying entry zero in the boot table.
all	This parameter displays entries 1-15 in the boot table. These entries contain the file names of valid boot or system load files.
boot_table	This variable specifies the target device ITOC to display. The target device can be either CM (computing module) or MS (message switch). The valid entry values are cm and ms.
device_name	This variable identifies the system load module (SLM) disk. The valid entry values are s00d and s01d.

Qualifications

None

listbootfl (continued)

Examples

The following table provides examples of the listbootfl command.

Examples of the listbootfl command	
Example	**Task, response, and explanation**
listbootfl s00d cm all ↵ *where* s00d specifies the device name cm specifies the boot table	

	Task:	Display every system load file for the CM.

	Response:	`Image Table of Contents for CM on SLM 0`
		
		 `ITOC   Volume    File`
		 `Entry  Name      Name`
		 `No`
		 `-----  ------    ----`
		 `1      IMAGE     BCS21ZELOAD`
		 `4      IMAGE     BCS21ZFLOAD`
		 `14     OLDIMAGE  BCS21ZFLOAD`
		
		 `DISKUT:`

	Explanation:	You see all the files registered in the ITOC of a SLM disk.

listbootfl s00d cm active ↵ *where* s00d specifies the device name cm specifies the boot table	

	Task:	Display the active system load file for the CM.

	Response:	`Image Table of Contents for CM on SLM 0`
		
		 `Active Boot File Device:  DISK`
		 `Active Boot File Volume:  IMAGE`
		 `Active Boot File Name:    BCS21ZELOAD`
		
		 `DISKUT:`

	Explanation:	You see the active system load file for the CM on SLM 0.

listbootfl (end)

Responses

The following table provides explanations of the responses to the listbootfl command.

Responses for the listbootfl command	
MAP output	**Meaning and action**
`LISTBOOTFL command is aborted. Device does not exist`	
	Meaning: You specified a device that does not exist. The command aborts.
	Action: Reissue the command specifying the correct device name.
`LISTBOOTFL command is aborted. Device IO error`	
	Meaning: You specified a device that encountered an input or output error. The command aborts.
	Action: Follow maintenance procedures.
`LISTBOOTFL command is aborted. Device is in use`	
	Meaning: You specified a device that is being used by another process. The command aborts.
	Action: Reissue the command when the other process is completed.
`LISTBOOTFL command is aborted. Device is not ready`	
	Meaning: You specified a device that is not ready or available. The command aborts.
	Action: Use the maintenance commands to return the device to service. Reissue the command.
`LISTBOOTFL command is aborted. ITOC table does not exist`	
	Meaning: You specified an ITOC that does not exist. The command aborts.
	Action: Reissue the command with the correct ITOC table.

listfl

Function

Use the listfl command to display information about files on the system load module (SLM) disk or tape volume.

listfl command parameters and variables			
Command	**Parameters and variables**		
listfl **lf**	*vol_name*	full	bvols

Parameters and variables	**Description**
bvols	This parameter displays the files contained in the backup volume. This parameter applies to tape devices only.
full	This parameter displays detailed information and can be omitted.
vol_name	This variable specifies the name of the volume on the SLM device. The first four characters indicate the device.
	The disk devices are s00d and s01d. If the volume is on a disk, its name appears in the eight characters following the device name.
	The tape devices are s00t and s01t. If the volume is on a tape, the volume name is limited to the device name.

Qualifications

None

Examples

The following table provides examples of the listfl command.

Examples of the listfl command
Example   **Task, response, and explanation**
listfl s00dimage1 full ↵   *where*   s00dimage1 specifies the volume name
-continued-

listfl (continued)

Examples of the listfl command (continued)

Example	Task, response, and explanation

Task: Display file information for a disk.

Response:
```
File information for volume S00DIMAGE1.
File             Create  Modify  File  File  In
Name             Date    Date    Org.  Code  ITOC
                 Y/M/D   Y/M/D
---------------- ------  ------  ----- ----  -----
SN100_26AU_MS    79/05/03 79/05/03 IMG    0
SN100_26AU_CM    79/05/03 79/05/03 IMG    0
COPYLAST$LD      84/11/20 84/11/20 OTH    0
BLMLA02          83/12/17 83/12/17 OTH    0
BRLMLA02         83/12/17 83/12/17 OTH    0
RDCMMA01         83/12/17 83/12/17 OTH    0
BTMKA02          83/12/17 83/12/17 OTH    0
BTMKA02          83/12/17 83/12/17 OTH    0      *OLD*
KTMKA02          83/12/17 83/12/17 OTH    0
BRLMLA02         83/12/17 83/12/17 OTH    0      *OLD*
S00DIMAGE        83/12/17 83/12/17 OTH    0      YES
MYFILE           78/09/04 78/09/04 IMG  205
M24BJ_EC100_MS   88/06/30 88/06/30 IMG    0
M26AX_SN100_MS   88/06/30 88/06/30 IMG    0

File             No of   No of        Record  Max.
Name             Blocks  Records      Format  Record
                                              Length
---------------- ------  -------      ------  ------
SN100_26AU_MS    5894    2992         FIX     1020
SN100_26AU_CM    75606   37803        FIX     1020
COPYLAST$LD         7       14        FIX      256
BLMLA02           212     1427        FIX       76
BRLMLA02          376     2527        FIX       76
RDCMMA01           78      523        FIX       76
BTMKA02            46      275        VAR       76
BTMKA02            45      302        FIX       76    *OLD*
KTMKA02            42      278        FIX       76
BRLMLA02          277     1863        FIX       76    *OLD*
S00DIMAGE12       114      690        VAR       76
MYFILE              4        2        FIX     1020
M24BJ_EC100_MS   6574     3278        FIX     1020
M26AX_SN100_MS   5984     2992        FIX     1020
```

Explanation: You see information about every file in the volume image1 on
                   SLM 0.

-continued-

Examples of the listfl command (continued)

Example   Task, response, and explanation

---

listfl   s00t ↵
where

s00t   specifies the tape volume name

        Task: Display file information for a tape.

        Response:

```
File                Create   File   File    In
Name                Date     Org.   Code    ITOC
                    Y/M/D
----------------    ------   -----  ----    -----
M26AV_MDEV_MS       78/09/04  IMG     0
M26AX_SN100_MS      78/09/04  IMG     0
VOL1                78/09/04  VOL    200
```

        Explanation: You see information about every file in the s00t tape volume.

End

Responses

The following table provides explanations of the responses to the listfl command.

Responses for the listfl command

MAP output   Meaning and action

```
LISTFL command is aborted. Device does not exist
```

        Meaning: You specified a device that does not exist. The command aborts.

        Action: Reissue the command specifying the correct volume.

```
LISTFL command is aborted. Device error
```

        Meaning: You specified a device that encountered an input or output error. The command aborts.

        Action: Follow maintenance procedures.

-continued-

---

listfl (end)

Responses for the listfl command (continued)	
MAP output	**Meaning and action**
`LISTFL command is aborted. Device is not ready`	
	Meaning: You specified a device that is not ready or available. The command aborts.
	Action: Use the maintenance commands to return the device to service. Reissue the command.
`LISTFL command is aborted. Parameters must include a valid volume name or path name.`	
	Meaning: You specified a volume or path name that is not valid. The command aborts.
	Action: Reissue the command with a valid volume name or path.
`LISTFL command is aborted. Volume does not exist`	
	Meaning: You specified a volume that does not exist. The command aborts.
	Action: Reissue the command with the correct volume.
End	

listvols

Function

Use the listvols command to display information about each volume on the
system load module (SLM) disk. The information is drawn from data
structures in memory, not directly from data on the disk.

listvols command parameters and variables		
Command	**Parameters and variables**	
listvols **lv**	*device_name*	<u>*brief*</u> full

Parameters and variables	**Description**
<u>*brief*</u>	This default parameter displays brief volume information. Omitting the full parameter forces the system to default to displaying brief volume information.
device_name	This variable indicates the device for which a volume display is requested. The valid entry values are s00d and s01d.
full	This parameter displays more detailed volume information.

Qualification

CAUTION

The DISKUT directory commands can be used only if the
SLM device is in the OK state.

The DISKUT directory commands can be used only if the SLM device is in
the OK state. If the device is not in service, attempting to list volumes on it
will result in the message

```
INVALID DEVICE
```

being displayed, even though the there is such a volume on the device.

listvols (continued)

Examples

The following table provides examples of the listvols command.

Examples of the listvols command	
Example	**Task, response, and explanation**
listvols s00d ↵ *where* s00d	specifies the device name

Task:	Display brief volume information.
| **Response:** | ```
Volume information for the SLM disk 0
Volume     Modify  Total   No. of  ITOC
Name       Date    No. of  Open    Files
           Y/M/D   Files   Files
--------   ------  ------  ------  ------
S00Dvol1   870122    876        2      15
S00Dvol2   861121     14       14       0
S00Dvol3   861121      0        0       0
S00Dvol4   861121   2048      100       0
S00Dvol5   861121      1        0       1
``` |
| **Explanation:** | You see brief volume information for disk s00d. |

-continued-

Examples of the listvols command (continued)

Example	Task, response, and explanation

listvols s00d full ↵
where

s00d   specifies the device name

Task: Display detailed volume information.

Response:
```
Volume information for the SLM disk 0

Volume    Volume     Create   Modify   Total   No. of   ITOC
Name      Status     Date     Date     No. of  Open     Files
                     Y/M/D    Y/M/D    Files   Files
--------  ------     ------   ------   ------  ------   ------
S00Dvol1  opened     870122   870314     876     2        15
S00Dvol2  deleted    861121   870314      14    14         0
S00Dvol3  indefinite 861121   861121       0     0         0
S00Dvol4  closed     861121   861121    2048   100         0
S00Dvol5  readonly   861121   861121       1     0         1

Volume    Total      Blocks   Total    Largest Number
Name      No. of     In       Free     Free    of Free
          Blocks     Use      Blocks   Segment Segments
--------  ------     ------   ------   ------- --------
S00Dvol1  102400      98765     3635     2345     2022
S00Dvol2  002048       2048        0      300       22
S00Dvol3  002048          8     2040     2040        1
S00Dvol4  002048          8     2040     2040      300
S00Dvol5  002048          8     2040     2040        1
```

Explanation: You see detailed volume information for disk s00d.

End

listvols (end)

Responses

The following table provides explanations of the responses to the listvols command.

Responses for the listvols command	
MAP output	**Meaning and action**
`LISTVOLS command is aborted. Device does not exist`	
	Meaning: You specified a device that does not exist. The command aborts.
	Action: Reissue the command specifying the correct device.
`LISTVOLS command is aborted. Device IO error`	
	Meaning: You specified a device that encountered an input or output error. The command aborts and leaves the input or output in an unfinished state.
	Action: Follow maintenance procedures.
`LISTVOLS command is aborted. Device is not ready`	
	Meaning: You specified a device that is not ready or available. The command aborts.
	Action: Use maintenance commands to return the device to service. Reissue the command.
`LISTVOLS command is aborted. Volume does not exist`	
	Meaning: You specified a volume that does not exist. The command aborts.
	Action: Reissue the command specifying the correct volume name.

quit

Function

Use the quit command to exit the DISKUT directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	$\begin{bmatrix} \textit{1 level} \\ \text{all} \\ \textit{name} \\ \textit{n\_levels} \end{bmatrix}$

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)		
Example	**Task, response, and explanation**	
quit all ↵		
	Task:	Exit from all levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory		
	Task:	Exit from a specified directory without leaving any other directories.
	Response:	AMADUMP>>> >
	Explanation:	The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵		
	Task:	Exit from a specified number of levels.
	Response:	CI:
	Explanation:	You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End		

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

renamefl

Function

Use the renamefl command to rename a fault tolerant file system (FTFS) file, or a file stored on a system load module (SLM) disk volume. The file name must be different than the name of the volume on which it is stored but duplicate file names on a volume are allowed.

The renamefl command can also be used to move FTFS files across directories, within an FTFS volume. To move files across volumes, see the duplicate command.

File names can have a maximum of 32 characters with the exception of system image file names which are restricted to 17 characters. The file to rename can not be registered as a boot file in the image table of contents (ITOC).

Two or more files on a volume can have the same name. The name of the most recently modified file is added to the CI directory and the remainder of the files of the same name are marked as old files. To eliminate duplicate file names, you must remove the old files.

renamefl command parameters and variables	
Command	**Parameters and variables**
renamefl **rnf**	*volume_name* *old_name* *new_name*

Parameters and variables	Description
new_name	This variable is the new file name. It has a maximum of 32 characters or 17 characters for system image files. If the old file name is an STD file, the volume name must be specified. If the old file name is an FTFS file, the volume name may be omitted. The old file name for an FTFS file is prefixed by a colon(:), and replaced by an FTFS path name.
old_name	This variable is the current file name. It has a maximum of 32 characters or 17 characters for system image files.
volume_name	This variable specifies the SLM or FTFS volume name. The volume name may be omitted for FTFS only. It cannot exceed 12 characters. The first four characters must be s00d or s01d for SLM disks 0 or 1. The remaining characters give the volume name. The volume name can have a maximum of 12 characters.

renamefl (continued)

Qualifications

The SLM device specified in the renamefl command must be in service. If it is ManB or OffL, the system will prompt for the correct old filename (*old_name*) even if the old filename specified is correct.

Examples

The following table provides examples of the renamefl command.

Examples of the renamefl command	
Example	**Task, response, and explanation**
renamefl s00dvol1 myfile yourfile ↵ *where* s00dvol1 specifies the volume name myfile specifies the old file name yourfile specifies the new file name	
	Task: Rename a file. **Response:** `RENAMEFL S00DVOL1 MYFILE has been renamed to YOURFILE.` **Explanation:** This command renames myfile to yourfile on volume s00dvol1 of SLM 0.
renamefl s00dstd1 oldname newname ↵ *where* s00dstd1 specifies the volume name oldname specifies the old file name newname specifies the new file name	
	Task: Rename an STD file. **Response:** `OLDNAME RENAMED TO NEWNAME` **Explanation:** This command renames the file oldname to newname.
-continued-	

renamefl (continued)

Examples of the renamefl command (continued)	
Example	**Task, response, and explanation**
renamefl :/s00dftfs1/baddir/oldname :/s00dftfs1/newname ↵ *where* :/s00dftfs1/baddir/oldname specifies the old file path :/s00dftfs1/newdir/newname specifies the new file path	
Task:	Rename an FTFS file.
Response:	`:/S00DFTFS1/BADDIR/OLDNAME   RENAMED TO` `/S00DFTFS1/NEWDIR/NEWNAME`
Explanation:	This command renames the FTFS file.
End	

Responses

The following table provides explanations of the responses to the renamefl command.

Responses for the renamefl command	
MAP output	**Meaning and action**
`<old_name> RENAMED TO <new_name>`	
Meaning:	You executed the command successfully. The old file name was replaced by the new file name for the specified file.
Action:	None
`Device does not exist.`	
Meaning:	You specified a device or volume that does not exist. The command aborts.
Action:	Reissue the command specifying the correct device.
`Device error.`	
Meaning:	You specified a device that encountered an input or output error. The command aborts.
Action:	Check the integrity of disk data structures and files being operated on.
-continued-	

renamefl (continued)

Responses for the renamefl command (continued)	
MAP output	**Meaning and action**
File <old_file> Volume <volume> has been renamed to <new_file>	
	Meaning: You executed the command successfully.
	Action: None
File cannot be found in user directory.	
	Meaning: You specified a file that cannot be found in the user directory. The command aborts.
	Action: List the files to place them in the user directory.
File does not exist.	
	Meaning: You specified a file that does not exist. The command aborts.
	Action: Reissue the command specifying the correct file name.
File is a registered bootable image file.	
	Meaning: You specified a file to rename that is registered as a boot file in an ITOC. The command aborts.
	Action: Remove the file from the ITOC. Reissue the command.
File name too long for device.	
	Meaning: You specified a file name that exceeds 32 characters or 17 characters for a system image file. The command aborts.
	Action: Reissue the command with a shorter file name.
FILE RENAMED, BUT COULD NOT DETERMINE PARENT OF <new_name> FILE SYSTEM ERROR: <file_system_text>	
	Meaning: You successfully renamed the file, but some header information may be lost.
	Action: Contact next level of maintenance.
-continued-	

renamefl (end)

Responses for the renamefl command (continued)

MAP output   Meaning and action

```
FILE RENAMED, BUT COULD NOT VERIFY FILE INFO FOR <new_name>
FILE SYSTEM ERROR: <system_error_text>
```

Meaning: You successfully renamed the file, but the clean up attempts failed.

Action: Contact next level of maintenance.

```
Illegal file system operation requested.
```

Meaning: You requested an operation that is illegal in the context of the command. The command aborts.

Action: Check the command and reissue the command correctly.

```
Medium error.
```

Meaning: You specified a device that encountered an error in reading or writing to the medium on the disk. The command aborts.

Action: Consult maintenance procedures for the disk.

```
RENAME <old_name> TO <new_name> FAILED
FILE SYSTEM ERROR: <system_error_text>
```

Meaning: The system encountered errors during the rename process. Possible causes are denied file access, a mistyped file name, or an invalid path. The command aborts.

Action: Determine the cause of the system error. Retry the command, or contact the next level of maintenance.

End

restore

Function

Use the restore command to restore system load module (SLM) and fault tolerant file system (FTFS) disk files from the tape cartridge. Both disk and tape cartridge devices must be in the same SLM unit.

Individual files or entire disk volumes may be restored from backup volumes on tape. Duplicate file names are permitted on tape and only the latest versions are used.

The tape from which disk files are to be copied must first be mounted using the command inserttape.

restore command parameters and variables			
Command	**Parameters and variables**		
restore **re**	file volume	*vol_name* *vol_name*	*tape_fname* *tape_vname*
Parameters and variables	**Description**		
file	This parameter specifies a single file in a backed up volume that is copied to a disk volume.		
tape_fname	This variable is the name of the tape file to restore to disk. If the tape file is an FTFS file, the volume name may be omitted. The tape file for an FTFS file is prefixed by a colon(:) and replaced by an FTFS path name. The file name can have a maximum of 17 characters.		
tape_vname	This variable is the name of the tape volume to restore to disk. It can not be used with FTFS files. The backup tape volume name can have a maximum of 17 characters.		
vol_name	This variable specifies the name of the SLM disk volume where the specified file resides. The volume name may be omitted for FTFS disk volumes. The first four characters must be s00d or s01d for SLM disks 0 or 1. The remaining characters specify the volume name. The volume name must not exceed 12 characters.		
volume	This parameter indicates that all the files in a backed up volume are copied to a disk volume. This parameter is not supported for FTFS files.		

Qualifications

None

restore (continued)

Examples

The following table provides examples of the restore command.

Examples of the restore command	
Example	**Task, response, and explanation**
restore file s00dvol1 myfile ↵ *where* s00dvol1 specifies the volume name myfile specifies the tape file name	

	Task:	Restore a file from tape.
	Response:	```Tape file has been opened. Disk file has been created. The copy operation may take several minutes. File has been copied. Tape file has been closed. Disk file has been closed. Tape file MYFILE has been copied to volume SOODVOL1.```
	Explanation:	You restored the file named myfile from the tape backup to the disk volume s00dvol1 on SLM 0.

restore volume s01dvol1 myvolume ↵ *where* s01dvol1 specifies the volume name myvolume specifies the tape volume name		
	Task:	Restore a volume from tape.
	Response:	```Tape file has been opened. The copy operation may take several minutes. File has been copied. Tape file has been closed. # files have been copied to volume SO1DVOL1 from backup volume MYVOLUME.```
	Explanation:	You restored all the files in the tape volume myvolume to the disk volume s01dvol1 on SLM 1.

<div align="center">

-continued-

</div>

restore (continued)

Examples of the restore command (continued)
Example   **Task, response, and explanation**
restore file s00dftfs1 mytapefile ↵ *where* s00dftfs1   specifies the volume name mytapefile   specifies the tape file name

Task:	Restore an FTFS file.	
Response:	TAPE FILE HAS BEEN OPENED DISK FILE IS CREATED FILE IS COPIED TAPEFILE IS CLOSED DISKFILE IS CLOSED	
Explanation:	You restored the file named mytapefile to an FTFS volume s00dftfs1 on SLM 0.	

End

Responses

The following table provides explanations of the responses to the restore command.

Responses for the restore command
MAP output   **Meaning and action**

Device does not exist.
Meaning: You specified a device or volume that does not exist. The command aborts.
Action: Reissue the command with a correct device.

Device error.
Meaning: You specified a device that encountered an input or output error. The command aborts.
Action: Check the integrity of the disk data structures and files.

-continued-

restore (continued)

Responses for the restore command (continued)	
MAP output	**Meaning and action**
`Device is in use.`	
	Meaning: You specified a device that is being used by another process. The command aborts.
	Action: Reissue the command after the other process is finished using the device.
`End of file medium met.`	
	Meaning: You reached the end of the tape. The command aborts.
	Action: Insert a new tape if data is being copied to tape.
`ERROR CLOSING DISK FILE` `FILE SYSTEM ERROR: <system_error_text>`	
	Meaning: You specified a disk file that the system could not close. This may be caused by other errors occurring during the copy process. The command aborts.
	Action: Check the status of the SLM disk drive. If necessary, contact the next higher level of support.
`ERROR CLOSING TAPE FILE` `FILE SYSTEM ERROR: <system_error_text>`	
	Meaning: You specified a tape file that the system could not close. This may be caused by other errors occurring during the copy process. The command aborts.
	Action: Check the status of the SLM tape drive, and the state of the cartridge tape. If necessary, contact the next higher level of support.
`ERROR COPYING FILE FROM TAPE TO DISK` `FILE SYSTEM ERROR: <system_error_text>`	
	Meaning: You specified a file that the system could not copy. The command aborts.
	Action: Check the status of the SLM tape drive, and the state of the cartridge tape. If necessary, contact the next higher level of support.
-continued-	

restore (continued)

Responses for the restore command (continued)

MAP output   **Meaning and action**

```
ERROR CREATING DISK FILE
FILE SYSTEM ERROR: <system_error_text>
```

Meaning: You specified a disk file that the system could not create. The command aborts.

Action: Determine the cause of the error. Check for write access to the target directory.

```
ERROR DETERMINING ROOT DIRECTORY OF <vol_nm>
FILE SYSTEM ERROR: <system_error_text>
```

Meaning: You specified a volume and the system failed to find the root directory. The command aborts.

Action: Check the status of the SLM. If necessary, contact the next higher level of support.

```
ERROR OPENING TAPE FILE
FILE SYSTEM ERROR: <system_error_text>
```

Meaning: You specified a tape file that the system could not open. The command aborts.

Action: Check that the cartridge tape is inserted correctly, and check the status of the tape drive. If necessary, contact the next higher level of support.

```
ERROR RETRIEVING USER HEADER LABEL
FILE SYSTEM ERROR: <system_error_text>
```

Meaning: You specified a tape file and the system could not read the header information. The command aborts.

Action: Check the status of the tape cartridge and the tape drive. If necessary, contact the next higher level of support.

```
ERROR SETTING BLACK BOX 2 INFORMATION
FILE SYSTEM ERROR: <system_error_text>
```

Meaning: The system failed to set the black box information, which is required by the internal file system routines. The command aborts.

Action: Contact the next higher level of support.

-continued-

restore (continued)

Responses for the restore command (continued)	
MAP output	**Meaning and action**
ERROR SETTING FILE SIZE FILE SYSTEM ERROR: <system_error_text>	
	Meaning: You specified a file and the system failed to set the size of the disk file. The command aborts.
	Action: Contact the next higher level of support.
ERROR WHILE DELETING BAD DISK FILE FILE SYSTEM ERROR: <system_error_text>	
	Meaning: You specified a file that caused the command to abort and the system failed to delete the bad disk file. The command completes execution.
	Action: Erase the bad disk file manually. This file should not be used.
File cannot be found in user directory.	
	Meaning: You specified a file that can not be found. The command aborts.
	Action: List the files to place them in the user directory.
File Directory is full.	
	Meaning: You specified a file directory that has no room for a new file. The command aborts.
	Action: Delete unwanted files from the disk volume to make space in the file directory, or create a disk volume with a larger file directory.
File does not exist.	
	Meaning: You specified a file that does not exist. The command aborts.
	Action: Reissue the command specifying the correct file name.
File name already exists in the FD.	
	Meaning: You specified a file name that already exists in the file directory. The command aborts.
	Action: Reissue the command specifying a new file name.
-continued-	

restore (continued)

Responses for the restore command (continued)	
MAP output	**Meaning and action**
`File name too long for device.`	
	Meaning: You specified a file name that exceeds 17 characters. The command aborts.
	Action: Reissue the command with a different file name.
`Illegal file system operation requested.`	
	Meaning: You requested an operation that is illegal in the context of the command. The command aborts.
	Action: Reissue the command correctly.
`Insufficient space in Free Space Map.`	
	Meaning: You specified a volume that does not have sufficient free space to store a file. The command aborts.
	Action: Delete unwanted files from the disk volume to free up space or create a larger disk volume.
`Medium error.`	
	Meaning: You specified a device that encountered an error in reading or writing to the medium on the disk. The command aborts.
	Action: Consult the maintenance procedures for the disk.
`RESTORE MULTIPLE FILES IS NOT SUPPORTED FOR FTFS VOLUMES`	
	Meaning: You tried to restore a volume for an FTFS volume; only single FTFS files can be restored. The command aborts.
	Action: Issue the restore file command to handle each individual file.
`TAPE FILE VOLUME TYPE DOES NOT MATCH DESTINATION`	
	Meaning: You tried to restore a file of one system type to a different volume type. This error occurs with an attempt to restore an FTFS file to an STD volume. The command aborts.
	Action: Retry restore using a destination volume of the same type as the original volume.
-continued-	

restore (end)

Responses for the restore command (continued)	
MAP output	**Meaning and action**
`Unit attention: tape is no longer available.`	
	Meaning: You specified a tape cartridge that has been removed or replaced after an inserttape command. The command aborts.
	Action: The tape is no longer accessible. Reinsert the cartridge tape.
`VOLUME <vol_nm> IS NOT SAME AS VOLUME OF WORKING DIRECTORY`	
	Meaning: You specified a volume that does not match the working directory name. The command aborts.
	Action: Retry the command with the volume name of the working directory as the volume name parameter of restore.
`Volume already contains files.`	
	Meaning: You specified a disk volume that already contains files. The command aborts.
	Action: Clear the files on the disk volume. Reissue the command.
`Volume incorrectly formatted.`	
	Meaning: You specified a tape volume that is not stored in the correct format. The command aborts.
	Action: Erase the contents of the tape.
`Volume not mounted.`	
	Meaning: You specified a tape cartridge that is not mounted. The command aborts.
	Action: Mount the tape cartridge and reissue the command.
End	

setbootfl

Function

Use the setbootfl command to add or set a file in the image table of contents (ITOC) so that it can be loaded into its target device. The setbootfl command also specifies this file as the default load file. The default load file is chosen by the system when the target device automatically boots itself.

setbootfl command parameters and variables					
Command	**Parameters and variables**				
setbootfl **sbf**	*vol_name*	*file_name*	*boot_table*	*entry_number*	regular active

Parameters and variables	**Description**
regular	This default parameter adds the file to the ITOC without making it active. Omitting this parameter forces the system to default to adding the file to the ITOC without making it active.
active	This parameter specifies that the file becomes the active boot file.
boot_table	This variable specifies the target device ITOC to display. The device can be either CM (computing module) or MS (message switch). The valid entry values are cm and ms.
entry_number	This variable specifies the entry in the ITOC to use for adding the file name and its respective volume name. The valid entry range is 1-15.
file_name	This variable specifies the name of the file in the volume on the system load module (SLM). It is a maximum of 17 characters long.
vol_name	This variable specifies the name of the volume on the SLM disk. The first four characters of the volume name refer to the name of the device. Disk names are s00d and s01d. The last eight characters specify the name of the volume on the disk.

Qualifications

None

setbootfl (continued)

Examples

The following table provides examples of the setbootfl command.

Examples of the setbootfl command	
Example	**Task, response, and explanation**
setbootfl s00dimage image21 cm 4 ↵ *where* s00dimage specifies the volume name image21 specifies the file name cm specifies the boot table 4 specifies the entry number	

	Task: Register a file in the ITOC and make it available to load to the CM.
	
	Response:
	`Volume S00DIMAGE file IMAGE2 has been registered in Image`
	`Table of Contents for CM on SLM 0 as entry number 4.`
	`DISKUT:`
	
	Explanation: You registered the file named image21 in volume s00dimage as entry number four of the ITOC table. You can load image21 to the CM on SLM 0.

setbootfl s01dimage image21 cm 14 active ↵ *where* s01dimage specifies the volume name image21 specifies the file name cm specifies the boot table 14 specifies the entry number	

	Task: Register a file in the ITOC as the active boot file.
	
	Response:
	`Volume S01DIMAGE IMAGE21 has been registered in Image Table`
	`Of contents for CM on SLM 1 as entry number 14.`
	`DISKUT:`
	
	Explanation: You registered the file name image 21 in volume s01dimage as entry number 14 of the ITOC table. File image21 automatically boots.

setbootfl (continued)

Responses

The following table provides explanations of the responses to the setbootfl command.

Responses for the setbootfl command	
MAP output	**Meaning and action**
SETBOOTFL command is aborted. Device does not exist	
	Meaning: You specified a device that does not exist. The command aborts.
	Action: Reissue the command with the correct device name.
SETBOOTFL command is aborted. Device IO error	
	Meaning: You specified a device that encountered an input or output error. The command aborts. The input or output is left in an unfinished state.
	Action: Verify the integrity of the file. If the integrity of the file has been lost, follow maintenance procedures.
SETBOOTFL command is aborted. Device is in use.	
	Meaning: You specified a device that is being used by another process. The command aborts.
	Action: Reissue the command when the other process is completed.
SETBOOTFL command is aborted. Device is not ready.	
	Meaning: You specified a device that is not ready or available. The command aborts.
	Action: Use the maintenance commands to return the device to service. Reissue the command.
SETBOOTFL command is aborted. File does not exist.	
	Meaning: You specified a file that does not exist. The command aborts.
	Action: Reissue the command with the correct file name.
-continued-	

setbootfl (end)

Responses for the setbootfl command (continued)	
MAP output	**Meaning and action**
`SETBOOTFL command is aborted. ITOC entry in use`	
	Meaning: You specified an ITOC entry number that is in use. The command aborts.
	Action: Display available entry numbers by using the listbootfl command. Reissue the command with another entry number.
`SETBOOTFL command is aborted ITOC table does not exist`	
	Meaning: You specified an ITOC table that does not exist. The command aborts.
	Action: Reissue the command with the correct ITOC table.
`SETBOOTFL command is aborted. Volume does not exist.`	
	Meaning: You specified a volume that does not exist. The command aborts.
	Action: Reissue the command with the correct volume name.
End	

DRAM level commands

Use the DRAM level of the MAP to inform the system of the pre-recorded phrases in programmable read-only memory (PROM), and to record phrases in random access memory (RAM) and erasable read-only memory (EEPROM).

Accessing the DRAM level

To access the DRAM level, enter the following command from the CI level:

dramrec ↵

If you encounter the message DRAMREC -- COMMAND DISALLOWED DURING DUMP or the message RECORDING FACILITY IN USE, wait and try again later. If you see any of these messages: CANNOT ALLOCATE DRAM DIRECTORY, CANNOT EXTEND DRAM ST, CANNOT FREE DRAM DIRECTORY, or COULD NOT ALLOCATE DRAM EVENT, check your hardware and software resources. If the problem persists, contact your maintenance support group.

DRAM commands

The commands available at the DRAM MAP level are described in this chapter and arranged in alphabetical order. The page number for each command is listed in the following table.

DRAM commands	
Command	**Page**
annsdebug	D-273
assign	D-275
assigndump	D-279
connect	D-281
debug	D-285
disconnect	D-289
display	D-291
-continued-	

DRAM commands (continued)	
Command	**Page**
erase	D-293
find	D-295
help	D-297
playback	D-299
position	D-301
quit	D-305
record	D-309
sitload	D-313
End	

annsdebug

Function

Use the annsdebug command to print the contents of internal announcement tables.

annsdebug command parameters and variables	
Command	**Parameters and variables**
annsdebug	There are no parameters or variables.

Qualifications

None

Example

The following table provides an example of the annsdebug command.

Example of the annsdebug command	
Example	**Task, response, and explanation**
annsdebug ↵	
	Task: Print ANNS tables.
	Response: `ANN  GRP  UNPROT`
	`LAST  CURCOUNT`
	`2      0`
	`2      0`
	`0      0`
	`.`
	`.`
	`.`
	`ANN  GRP  PROT`
	`MAXCO`
	`.`
	`.`
	`.`
	`ANN  TRK  PROT IS NIL`
	`ANN  TRK  UNPROT`
	`ANN  TRK  UNPROT IS NIL`
	`.`
	`.`
	`.`
	Explanation: This command prints the announcement tables.

annsdebug (end)

Response

The following table provides an explanation of the response to the annsdebug command.

Response for the annsdebug command		
MAP output	**Meaning and action**	

```
ANN   GRP   UNPROT

  LAST   CURCOUNT
    2       0
    2       0
    0       0
  .
  .
  .
ANN   GRP   PROT

MAXCO
  .
  .
  .
ANN   TRK   PROT IS NIL

ANN   TRK   UNPROT
ANN   TRK   UNPROT IS NIL
  .
  .
  .
```

Meaning: You entered the command correctly.

Action: None

assign

Function

Use the assign command to enable the datafilling of management tables. Programmable read-only memory (PROM) management tables are datafilled with speech and special information tone (SIT) data, and random access memory (RAM) management tables are datafilled with SIT data.

assign command parameters and variables				
Command	**Parameters and variables**			
assign	*dram* *phrasenm* *length* *block* *phraseno*			

Parameters and variables	Description
block	This variable specifies the speech block of the PROM card where the phrase resides. The valid entry range is 0-31.
dram	This variable specifies the number assigned to the digital recording announcement module (DRAM). The valid entry range is 0-63.
length	This variable specifies the operating time in seconds. The valid entry range is 1-31.
phrasenm	This variable string specifies the name given to a phrase which makes up all or part of an announcement.
phraseno	This variable specifies an integer identifying the phrase to the DRAM controller. The valid entry range is 0-63.

Qualification

This command should only be done once for each pre-recorded phrase when the PROM card is initially installed.

assign (continued)

Example

The following table provides an example of the assign command.

Example of the assign command	
Example	**Task, response, and explanation**
assign **0 vaceng 6 0 23** ↵ *where*	
0	specifies the number assigned to the DRAM
vaceng	specifies the name given to a phrase
6	specifies the operating time of the phrase in seconds
0	specifies the speech block of the PROM card in which the phrase resides
23	specifies an integer identifying the phrase to the DRAM controller
	Task: Assign a specified message in a specified location.
	Response: PHRASE ASSIGNED ON CARD n
	Explanation: This command assigns the phrase name vaceng on DRAM 0 block 0 with phrase number 23 a duration of 6 seconds.

Responses

The following table provides explanations of the responses to the assign command.

Responses for the assign command	
MAP output	**Meaning and action**
NO SPACE FOR NEW PHRASE	
	Meaning: Insufficient space exists on the specified DRAM for the phrase to be added.
	Action: None
PHRASE ASSIGNED ON CARD n	
	Meaning: The assignment was successful on the specified card.
	Action: None
-continued-	

assign (end)

Responses for the assign command (continued)	
MAP output	**Meaning and action**
SPECIFIED BLOCK DOES NOT RESIDE ON PROM CAN ONLY ASSIGN PHRASES TO PROM USE RECORD COMMAND FOR RAM	
	Meaning: The block number specified appears in Table DRAMS but is not on a RAM card.
	Action: Use the record command to assign the phrase to a RAM card.
UNKNOWN DRAM OR BLOCK	
	Meaning: The number specified for dram or block is not in Table DRAMS.
	Action: Verify the number specified and retry the command.
End	

assigndump

Function

Use the assigndump command to create a file or produce a display containing the commands used to assign the phrases existing in the present batch change supplement (BCS) load. The file is created before updating an office with a new BCS load.

When the new BCS package has been loaded, the file created by the commands parameter is read, causing automatic assignment of the phrases.

assigndump command parameters and variables	
Command	**Parameters and variables**
assigndump	anns commands
Parameters and variables	**Description**
anns	This parameter displays the valid digital recording announcement module (DRAM) phrases placed in the system through the assign command.
commands	This parameter creates a file containing all of the assign commands used in assigning DRAM phrases.

Qualifications

None

assigndump (end)

Example

The following table provides an example of the assigndump command.

Example of the assigndump command	
Example	**Task, response, and explanation**
assigndump commands ↵	
	Task: Display announcement strings in the assign command mode.

	Response:
	``
	DRAMREC
	ASSIGN 0 SILENCE 1 2 0
	ASSIGN 0 ACTS_PAUSE 1 0 0
	ASSIGN 0 ACTS_1 1 0 49
	        .
	        .
	        .
	ASSIGN 1 FFLTB8 1 7 52
	ASSIGN 1 FFLTB9 1 7 53
	``

	Explanation: This command displays announcement strings in the assign command mode.

Response

The following table provides an explanation of the response to the assigndump command.

Response for the assigndump command	
MAP output	**Meaning and action**
`Invalid symbol`	
	Meaning: You entered the command without an appropriate parameter.
	Action: Enter the appropriate parameter to continue or abort to cancel.

connect

Function

Use the connect command to connect a trunk to a digital recorded announcement module (DRAM) for recording.

connect command parameters and variables		
Command	**Parameters and variables**	
connect **conn**	*dram* *trunk_clli* *member*	

Parameters and variables	Description
dram	This variable specifies the DRAM where the recording is made. The valid entry range is 0-63.
member	This variable specifies the trunk member number assigned to the trunk. The valid entry range is 0-255.
trunk_clli	This variable specifies the common language location identifier (CLLI) of the trunk to connect to the DRAM.

Qualifications

The connect command is qualified by the following exceptions, restrictions and limitations:

- Only one trunk can be connected to a DRAM from a MAP at any given time.
- A headset (HSET) or JACK trunk with a jack appearance near the MAP being used is convenient for recording purposes. However, any trunk can be used.
- The connect command is usually followed by one or more uses of the record or position commands. If the DRAM has been recently reset, a default phrase can be heard playing over the trunk.

connect (continued)

Example

The following table provides an example of the connect command.

Example of the connect command	
Example	**Task, response, and explanation**
connect 0 hset 0 ↵ *where* 0 specifies the DRAM hset specifies the trunk CLLI 0 specifies the trunk member	
	Task: Connect a trunk to a DRAM.
	Response: CONNECTION MADE:
	Explanation: This command connects the trunk hset 0 to the DRAM 0 for recording.

Responses

The following table provides explanations of the responses to the connect command.

Responses for the connect command	
MAP output	**Meaning and action**
CONNECTION ALREADY MADE FROM THIS TERMINAL	
	Meaning: You have already made a successful connection from this terminal.
	Action: None
DRAM CURRENTLY IN USE BY RECORDING OR DIAGNOSTIC	
	Meaning: You specified a DRAM that has been posted in the trunk test position (TTP) and a diagnostic test is being run on it.
	Action: Wait until the diagnostic is complete and retry the connect command.
-continued-	

Responses for the connect command (continued)

MAP output	Meaning and action

DRAM NOT SEIZED

> **Meaning:** You specified a DRAM that is not idle and can not be seized.
>
> **Action:**   Wait for a period of time and retry the connect command.

INVALID DRAM CIRCUIT LOCATION IN TABLE DRAMS

> **Meaning:** You specified a DRAM trunk module circuit location shown in Table DRAMS that is incorrect.
>
> **Action:**   Contact the next level of support.

INVALID DRAM NUMBER

> **Meaning:** You specified a DRAM number that is out of range.
>
> **Action:**   Verify the DRAM number and retry the connect command.

NO DRAM CONTROLLER DATA

> **Meaning:** You specified a DRAM controller card that has not been defined in Table DRAMS.
>
> **Action:**   Contact the next level of support.

TRUNK NOT SEIZED

> **Meaning:** You specified a recording trunk that is not idle.
>
> **Action:**   Verify the desired trunk is idle and retry the connect command.

UNABLE TO FIND TRUNK TID

> **Meaning:** You specified an invalid trunk circuit number or trunk member number.
>
> **Action:**   Verify the trunk number and retry the connect command.

UNABLE TO MAKE NETWORK CONNECTION

> **Meaning:** You requested a connection that could not be made due to a problem in the network links, the peripheral modules, the junctors, the trunk, or the DRAM itself.
>
> **Action:**   Contact the next level of support.

-continued-

connect (end)

Responses for the connect command (continued)	
MAP output	**Meaning and action**
UNKNOWN TRUNK CLLI	
	Meaning: You specified a trunk CLLI that does not exist in Table CLLI.
	Action: Verify the trunk CLLI and retry the connect command.
UNKNOWN TRUNK MEMBER	
	Meaning: You specified a trunk member number that does not exist in Table TRKMEM.
	Action: Verify the trunk member number and retry the connect command.
End	

debug

Function

Use the debug command to display the contents of the digital recorded announcement module (DRAM) tables.

debug command parameters and variables	
Command	**Parameters and variables**
debug	⎡ all ⎤ *dram_ann* ⎢ drams ⎢ ⎢ phrase ⎢ ⎢ speech ⎢ ⎢ track ⎢ ⎣ unprot ⎦

Parameters and variables	Description
all	This parameter displays the contents of all the DRAM tables.
dram_ann	This variable specifies the DRAM announcement number. The valid entry range is 0-63.
drams	This parameter displays the contents of the DRAMS Table.
phrase	This parameter displays the contents of the PHRASE Table.
speech	This parameter displays the contents of the SPEECH Table.
track	This parameter displays the contents of the TRACK Table.
unprot	This parameter displays the contents of the UNPROT Table.

Qualifications

None

debug (continued)

Example

The following table provides an example of the debug command.

Example of the debug command	
Example	**Task, response, and explanation**

debug drams 1 ↵ *where*	
1	specifies the DRAM announcement number

	Task:	Display the contents of the DRAMS Table.
| | **Response:** | ```
TABLE DRAMS
  CTLR  0  CLLI  91
  COD TYP CKT STA ISV MB SB PB OFL NEQ BLK
   29   1   1  14   T   F  F  F   F   F
   58   2   3  14   T   F  F  F   F   F    0  1
   37   2   5  14   T   F  F  F   F   F    2  3
   69   2   7  14   T   F  F  F   F   F    4  5
   39   2   9  14   T   F  F  F   F   F    6  7
    0   0   0   0   F   F  F  F   F   T
    0   0   0   0   F   F  F  F   F   T
    0   0   0   0   F   F  F  F   F   T
    0   0   0   0   F   F  F  F   F   T
``` |
| | **Explanation:** | This command displays the contents of the DRAMS Table for the first announcement. |

Responses

The following table provides explanations of the responses to the debug command.

Responses for the debug command	
MAP output	**Meaning and action**

INVALID DRAM NUMBER	
	Meaning: You specified an entry in a table that does not exist.
	Action: Reenter the command using a valid DRAM number.
-continued-	

debug (end)

Responses for the debug command (continued)	
MAP output	**Meaning and action**
`Next par is: <DRAM/ANN> {0 to 63]`	
	Meaning: You did not specify a DRAM number.
	Action: Enter a valid DRAM number to continue or abort to cancel.
End	

disconnect

Function

Use the disconnect command to disconnect the recording trunk.

disconnect command parameters and variables	
Command	**Parameters and variables**
disconnect **disc**	There are no parameters or variables.

Qualification

When you exit the DRAM directory, any connected trunk is automatically disconnected.

Example

The following table provides an example of the disconnect command.

Example of the disconnect command		
Example	**Task, response, and explanation**	
disconnect	↵	
	Task:	Disconnect a recording trunk.
	Response:	TRUNK DISCONNECTED
	Explanation:	This command disconnects a recording trunk from the digital recorded announcement module (DRAM).

Response

The following table provides an explanation of the response to the disconnect command.

Response for the disconnect command	
MAP output	**Meaning and action**
DRAM NOT CURRENTLY CONNECTED	
	Meaning: No recording trunk has been connected to the DRAM.
	Action: None

display

Function

Use the display command to display the external and internal phrase names, and the use of recording space on one or all cards of the specified digital recorded announcement module (DRAM).

display command parameters and variables		
Command	**Parameters and variables**	
display	*dram*	*card*

Parameters and variables	**Description**
card	This variable specifies the number of the card where phrase contents are required. The valid entry range is 0-8.
dram	This variable specifies the number assigned to the DRAM. The valid entry range is 0-63.

Qualification

For programmable read-only memory (PROM) cards, the amount of recording space remaining is always zero, regardless of the phrases stored on it.

Example

The following table provides an example of the display command.

display (end)

Example of the display command

Example   **Task, response, and explanation**

---

display 0 2 ↵
where

0   specifies the number assigned to the DRAM
2   specifies the card number

          Task: Display the speech memory contents of a specified card of a
                            specified DRAM.

          Response:
```
CARD 2  PROM        SPACE:  MAX CONTIG  0   TOTAL   0

PHRASE EXT          PHRASE INT          LENGTH
-----------         -----------         ------
NCAFRE                   40                  10
PSPDFRE                  41                   9
VCAFRE                   42                  12
BLKDNFRE                 43                   6
```

          Explanation: This command displays the speech memory contents of card 2 of
                              DRAM 0.

Responses

The following table provides explanations of the responses to the display
command.

Responses for the display command

MAP output   **Meaning and action**

UNKNOWN CARD NUMBER

          Meaning: You specified a card number that is not in Table DRAMS.

          Action: Verify the card number and retry the display command.

UNKNOWN DRAM

          Meaning: You specified a DRAM number that is not in Table DRAMS.

          Action: Verify the DRAM number and retry the display command.

---

erase

Function

Use the erase command to erase a specified phrase from speech memory on a specified digital recorded announcement module (DRAM).

erase command parameters and variables	
Command	**Parameters and variables**
erase	*dram* *phrasename*

Parameters and variables	Description
dram	This variable specifies the number assigned to the DRAM. The valid entry range is 0-63.
phrasename	This variable string specifies the name given to a phrase that makes up all or part of an announcement.

Qualifications

None

Example

The following table provides an example of the erase command.

Example of the erase command	
Example	**Task, response, and explanation**
erase 0 vcaeng ↵ *where* 0 specifies the DRAM number vcaeng specifies the phrase name	

	Task:	Erase a specified phrase.
	Response:	PHRASE TO BE ERASED: VCAENG PLEASE CONFIRM (YES OR NO) yes
	Explanation:	This command erases the phrase vcaeng from the speech memory of DRAM 0.

erase (end)

Responses

The following table provides explanations of the responses to the erase command.

Responses for the erase command	
MAP output	**Meaning and action**
`INVALID DRAM NUMBER`	
	Meaning: You specified a DRAM number that is out of range.
	Action: Verify the DRAM number and retry the erase command.
`NO SUCH PHRASE EXISTS`	
	Meaning: You specified a phrase that does not reside on any programmable read-only memory (PROM) or random access memory (RAM) on the specified DRAM.
	Action: Verify the phrase name and retry the erase command.
`UNKNOWN DRAM`	
	Meaning: You specified a DRAM that is not listed in Table DRAMS.
	Action: Verify the DRAM number and retry the erase command.

find

Function

Use the find command to list all occurrences of the specified phrase on all digital recorded announcement modules (DRAMs) in the office.

find command parameters and variables	
Command	**Parameters and variables**
find	*phrasename*
Parameters and variables	**Description**
phrasename	This variable specifies the name of a phrase that makes up all or part of an announcement.

Qualifications

None

Example

The following table provides an example of the find command.

Example of the find command	
Example	**Task, response, and explanation**
find vcaeng ↵ *where* vcaeng specifies the phrase name	
	Task: List all occurrences of the specified phrase.

	Response: DRAM CARD TYPE LENGTH
	              ---- ---- ---- ------
	                0 2 PROM 12

	Explanation: This command lists all occurrences of the phrase vcaeng.

find (end)

Response

The following table provides an explanation of the response to the find command.

Response for the find command	
MAP output	**Meaning and action**
`PHRASE NOT FOUND`	
	Meaning: You specified a phrase that does not exist on any DRAM in the office.
	Action: None

help

Function

Use the help command to receive online documentation for the DRAM directory.

help command parameters and variables	
Command	**Parameters and variables**
help	_all_ _command_nam_

Parameters and variables	**Description**
all	Omitting this entry forces the system to default to displaying online documentation for this directory.
_command_nam_	This variable specifies a valid DRAM directory command name. When the _command_nam_ variable is replaced by a command name, online documentation for the specified command is provided.

Qualifications

None

Example

The following table provides an example of the help command.

Example of the help command	
Example	**Task, response, and explanation**
help assigndump ↵ _where_ assigndump specifies the command name	
	Task: Access online documentation.

	Response: `ASSIGNDUMP: DISPLAY ANN STRINGS IN ASSIGN COMMAND`
	`MODE`
	`Parms: <FUNCTION> {COMMANDS,`
	`                      ANNS}`

	Explanation: This example typifies a response for the help command string.

help (end)

Response

The following table provides an explanation of the response to the help command.

Response for the help command	
MAP output	**Meaning and action**
MODULE LOADED OR NEEDS OTHER CI INCREMENT TO BE BUILT.	
	Meaning: The directory you are trying to access is not loaded or must be accessed through another directory.
	Action: None

playback

Function

Use the playback command to play back a digital recorded announcement module (DRAM) phrase over a headset or jack. The phrase is repeated continuously until a new command is entered or a DRAM diagnostic is run.

playback command parameters and variables		
Command	**Parameters and variables**	
playback	*dram*	*phrasename*
Parameters and variables	**Description**	
dram	This variable specifies the DRAM number of the desired phrase. The valid entry range is 0-63.	
phrasename	This variable string specifies the name of the phrase to play back.	

Qualifications

None

Example

The following table provides an example of the playback command.

Example of the playback command	
Example	**Task, response, and explanation**
playback blkdneng ↵ *where* blkdneng specifies the phrase name	
	Task: Play back a specified phrase.
	Response: No display is provided by the MAP. The phrase is played through a headset until another command is given.
	Explanation: This command plays back the phrase blkdneng through a headset.

playback (end)

Responses

The following table provides explanations of the responses to the playback command.

Responses for the playback command	
MAP output	**Meaning and action**
FIRST USE CONNECT COMMAND TO CONNECT HSET OR JACK	
	Meaning: You must connect the headset (HSET) or jack by using the connect command.
	Action: Connect the headset or jack with the connect command and retry the playback command.
INVALID DRAM NUMBER	
	Meaning: You specified a DRAM number that is out of range.
	Action: Verify the DRAM number and retry the playback command.
UNKNOWN DRAM	
	Meaning: You specified a DRAM that is not in Table DRAMS.
	Action: None

position

Function

Use the position command to record a phrase at a given position in memory.

position command parameters and variables					
Command	**Parameters and variables**				
position	*phrasename*	*length*	⎡ pad ⎤ ⎣ nopad ⎦	*block*	*startpos*

Parameters and variables	**Description**
block	This variable specifies the digital recorded announcement module (DRAM) speech block number where the phrase is recorded. The valid entry range is 0-31.
length	This variable specifies the approximate length, in seconds, of the phrase to record. The valid entry range is 1-31.
nopad	This parameter specifies that the 6 db pad is not inserted.
pad	This parameter specifies that the 6 db pad is inserted in the recording circuit for optimum speech response.
phrasename	This variable string specifies the name of the phrase that makes up all or part of an announcement.
startpos	This variable specifies the number of seconds from the start of the speech block where the recorded phrase is positioned. The valid entry range is 1-31.

Qualifications

The position command is qualified by the following exceptions, restrictions and limitations.

- The connect command must be issued before the record command is entered.

> **CAUTION**
>
> The position command should only be used if there is a valid reason for recording on a specific card. An overwrite of existing speech memory can occur.

position (continued)

- Whenever a recorded announcement is longer than 18 seconds, you must record it as two or more phrases of equal length. Refer to Tables DRAMTRK and ANNMEMS for the stitching of phrases and tracks to create announcements.

- The DRAM commences to play back the recorded phrase continuously until the phrase is erased or over-recorded, the trunk is disconnected, or a diagnostic is run.

Example

The following table provides an example of the position command.

Example of the position command	
Example	**Task, response, and explanation**
position vaceng 10 nopad 3 10 ↵ *where* vaceng specifies the phrase name 10 specifies the length, in seconds 3 specifies the DRAM speech block number 10 specifies the number of seconds from the start of the speech block	

	Task: Set the position for a recording phrase.
	
	Response: RECORDING ON DRAM 0
	                 USING TRUNK HSET MEMBER 0
	
	Explanation: This command sets the position for the phrase vaceng for
	                 10 seconds at block 3 to start in 10 seconds.

Responses

The following table provides explanations of the responses to the position command.

Responses for the position command	
MAP output	**Meaning and action**
INSUFFICIENT SPACE FOR PHRASE AT SPECIFIED LOCATION	
	Meaning: The system cannot find sufficient space for the phrase at the location defined by the block and startpos numbers. **Action:** Shorten the phrase.
-continued-	

position (end)

Responses for the position command (continued)	
MAP output	**Meaning and action**
`INVALID BLOCK NUMBER`	
	Meaning: The specified block number is out of range.
	Action: Verify a valid block number and retry the position command.
`NO BLOCK FOUND WITH SUFFICIENT SPACE`	
	Meaning: The system is unable to find any recording blocks with sufficient space to accommodate the phrase of the defined length.
	Action: Shorten the phrase.
`NO TRUNK CURRENTLY CONNECTED TO THIS TERMINAL`	
	Meaning: The connect command has not been entered.
	Action: Enter the connect command and then retry the position command.
`PHRASE COULD NOT BE ADDED`	
	Meaning: The phrase has already been recorded on random access memory (RAM) elsewhere in the DRAM.
	Action: None
`RECORDING ON DRAM n` `USE TRUNK s MEMBER m`	
	Meaning: The recording process has started. The trunk, DRAM, and card used are shown.
	Action: None
`UNABLE TO ASSIGN SPACE`	
	Meaning: The specified space length exceeds the maximum continuous space available on the card.
	Action: None
End	

quit

Function

Use the quit command to exit the DRAM directory.

quit command parameters and variables	
Command	**Parameters and variables**
quit	*1 level* all *name* *n_levels*

Parameters and variables	**Description**
1 level	Omitting this entry forces the system to default to exiting one directory level. (This is the most common selection for exiting nonmenu directories.)
all	This parameter causes the system to exit all directories and returns you to the CI level.
n_levels	This variable specifies the number of directory levels to exit. The default value is 1.
name	This variable specifies the particular directory level from which you want to exit.

Qualifications

None

Examples

The following table provides examples of the quit command.

Examples of the quit command		
Example	**Task, response, and explanation**	
quit ↵		
	Task:	Exit from this directory.
	Response:	`CI:`
	Explanation:	You entered the quit command to exit a directory that is accessed directly from the CI level. The system assumes the default value of one directory level and returns you to the CI level.
	-continued-	

quit (continued)

Examples of the quit command (continued)	
Example	**Task, response, and explanation**
quit all ↵	
	Task: Exit from all levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit all levels and return to the CI level.
quit dskut ↵ *where* dskut specifies a directory	
	Task: Exit from a specified directory without leaving any other directories.
	Response: AMADUMP>>> >
	Explanation: The system exited the DSKUT directory without leaving any other directories. (In this example, the AMADUMP directory is still accessed.)
quit 2 ↵	
	Task: Exit from a specified number of levels.
	Response: CI:
	Explanation: You entered the quit command in order to exit from two levels. You were using a subdirectory accessed through another directory, so the system exits both directory levels and returns you to the CI level.
End	

Responses

The following table provides explanations of the responses to the quit command.

quit (end)

Responses for the quit command	
MAP output	**Meaning and action**
`CI:`	
	Meaning: You have returned to the CI MAP level.
	Action: Access another directory from the CI MAP level or end this session.
`QUIT -- Increment not found`	
	Meaning: The system did not recognize the *name* variable replacement value as a valid directory level.
	Action: Verify your entry. If the name you entered is incorrect, retry the command. If the name is correct, check to see if the environment is active or if you have already left that directory.
`QUIT -- Unable to quit requested number of levels`	
	Meaning: You entered an *n_levels* variable replacement value that is too large.
	Action: Enter the quit all command string or retry the command with a smaller number of levels.

record

Function

Use the record command to define and record a phrase.

record command parameters and variables								
Command	**Parameters and variables**							
record	*phrasename*	*length*	pad nopad	*dram*	*card*	*intphrase*	*noforce* force	

Parameters and variables	Description
noforce	Omitting this entry forces the system to default to not forcing the datafill during a dump or restore.
card	This variable specifies the number assigned to the digital recorded announcement module (DRAM) memory card. The valid entry range is 1-8.
dram	This variable specifies the number of the DRAM where the recording takes place. The valid entry range is 0-63.
force	This variable forces the system to datafill the DRAM software internal memory management tables.
intphrase	This variable specifies the internal phrase number. The valid entry range is 0-63.
length	This variable specifies the approximate length, in seconds, of the phrase to record. The valid entry range is 1-31.
nopad	This parameter specifies that the 6 db pad is not inserted.
pad	This parameter specifies that the 6 db pad is inserted for optimum speech response.
phrasename	This variable string specifies the name given to the phrase that makes up all or part of the announcement.

Qualifications

The record command is qualified by the following exceptions, restrictions and limitations.

- The connect command must be issued before the record command is entered.
- For normal recording, use only the parameters phrasename, length, and pad or nopad.

record (continued)

- Use the parameters dram, card, intphrase, and force only on a dump and restore when converting from one batch change supplement (BCS) to another.

- Use the position command if a recording is desired on a particular card.

- Whenever a recorded announcement is longer than 18 seconds, you must record it as two or more phrases of equal length. Refer to Tables DRAMTRK and ANNMEMS for the stitching of phrases and tracks to create announcements.

- The DRAM commences to play back the recorded phrase continuously until the phrase is erased or over-recorded, the trunk is disconnected, or a diagnostic is run.

Example

The following table provides an example of the record command.

Example of the record command		
Example	**Task, response, and explanation**	
record vcaeng 10 nopad ↵ *where*		
vcaeng 10	specifies the phrase name specifies the length, in seconds	
	Task:	Record a phrase.
	Response:	`RECORDING ON DRAM 0 CARD n` `USE TRUNK HSET MEMBER 0`
	Explanation:	This command records phrase vcaeng for 10 seconds with no pad to optimize the speech.

record (continued)

Responses

The following table provides explanations of the responses to the record command.

Responses for the record command	
MAP output	**Meaning and action**
INVALID CARD NUMBER	
	Meaning: You specified a card number that is out of range.
	Action: Verify the card number and retry the record command.
INSUFFICIENT SPACE FOR RECORDING OR PHRASE ALREADY RECORDED	
	Meaning: You specified a phrase length that exceeds the maximum continuous space left on the card, or a phrase by this name has already been recorded.
	Action: Shorten the phrase or rename the phrase.
INSUFFICIENT SPACE ON SPECIFIED DRAM	
	Meaning: You specified a phrase length that exceeds the maximum continuous space left on the card, or the card does not physically exist.
	Action: Shorten the phrase or specify a different card.
NO TRUNK CONNECTION FROM THIS TERMINAL	
	Meaning: You did not execute the connect command to connect a trunk for recording to the DRAM before the record command was executed.
	Action: Execute the connect command and retry the record command.
PHRASE COULD NOT BE ADDED	
	Meaning: You specified a phrase that has already been recorded on random access memory (RAM) within the DRAM.
	Action: None
-continued-	

record (end)

Responses for the record command (continued)	
MAP output	**Meaning and action**
`RECORDING ON DRAM n` `USE TRUNK HSET MEMBER n`	
	Meaning: You started the recording process on DRAM n using trunk headset (HSET) member number n. **Action:** None
`UNABLE TO ASSIGN SPACE OR PHRASE ALREADY EXISTS`	
	Meaning: A RAM card has been pulled out and a re-recording is in progress without having erased all phrases that existed on the RAM card. **Action:** None
`UNKNOWN CARD`	
	Meaning: You specified a card number that is not in Table DRAMS. **Action:** Reenter the command with an appropriate card number or datafill Table DRAMS.
End	

sitload

Function

Use the sitload command to put special information tones (SIT) in random access memory (RAM).

sitload command parameters and variables	
Command	**Parameters and variables**
sitload	*dram*

Parameters and variables	**Description**
dram	This variable specifies the number of the digital recorded announcement module (DRAM) where the SIT download is made. The valid entry range is 0-63.

Qualifications

The sitload command is qualified by the following exceptions, restrictions and limitations:

- Use of the sitload command is not required in offices equipped with programmable read-only memory (PROM) cards, as the SITs are already in PROM.
- The sitload command is used when an office goes into service and whenever RAM card memory containing the SIT data is corrupted or accidentally erased.
- Make sure that two physical RAM cards are in the shelf corresponding to the DRAM being used. Four seconds of speech time on each card are required for SIT data.
- Following successful use of the sitload command, use the assign command to make sure the system is aware of the existence of the SIT.

sitload (continued)

Example

The following table provides an example of the sitload command.

Example of the sitload command	
Example	**Task, response, and explanation**
sitload **0** ↵ *where*	
0	specifies the DRAM number
	Task: Record special information tones.
	Response: `SITDATA HAS BEEN SUCCESSFULLY LOADED`
	Explanation: This command records special information tones to DRAM 0.

Responses

The following table provides explanations of the responses to the sitload command.

Responses for the sitload command	
MAP output	**Meaning and action**
`INVALID CARDF TYPE IN TABLE DRAMS`	
	Meaning: The card data in Table DRAMS was probably altered through the table editor during the sitload attempt.
	Action: Verify Table DRAMS and retry the sitload command.
`NEGATIVE ACKNOWLEDGEMENT TO SITLOAD SETUP`	
	Meaning: The DRAM is responding to sitload, but is unable to comply with the command.
	Action: Diagnose the DRAM and retry the sitload command.
`NO CARD DATA`	
	Meaning: The card data in Table DRAMS was probably deleted through the table editor data modification during the sitload attempt.
	Action: Verify Table DRAMS and retry the sitload command.
-continued-	

sitload (end)

Responses for the sitload command (continued)	
MAP output	**Meaning and action**
NO DRAM CONTROLLER DATA	
	Meaning: The controller is not datafilled, or is improperly datafilled in Table DRAMS.
	Action: Verify Table DRAMS and retry the sitload command.
NO DRAM RESPONSE TO SITLOAD SETUP	
	Meaning: The DRAM is offline or undergoing diagnostics.
	Action: Verify the DRAM is online and in service and retry the sitload command.
NO RAM CARDS PRESENT	
	Meaning: The sitload command cannot execute without RAM cards.
	Action: Contact the next level of support.
SITLOAD UNSUCCESSFUL TRY AGAIN	
	Meaning: The system was unable to successfully execute the command.
	Action: Retry the sitload command
SIT TONES ALREADY EXIST ON PROM OR NEED TWO RAM CARDS FOR SITLOAD	
	Meaning: The system was unable to successfully execute the command.
	Action: Contact the next level of support.
End	

DMS-100 Family

Nonmenu Commands

Historical Reference Manual-ABBT Through DRAM
Volume 1 of 4

NØRTEL
NETWORKS

How the world shares ideas.