

# Networks • Communications



## DECnet-RSX

Guide to  
Network Management Utilities

software **digital**

# DECnet-RSX

## Guide to Network Management Utilities

Order No. AA-EB30A-TC

September 1985

This *DECnet-RSX Guide to Network Management Utilities* describes the utilities related to DECnet-RSX network management.

Supersession/Update Information: This is a new manual.

Operating System and Version: RSX-11M V4.2  
RSX-11S V4.2  
RSX-11M-PLUS V3.0  
Micro/RSX V3.0

Software Version: DECnet-11M V4.2  
DECnet-11S V4.2  
DECnet-11M-PLUS V3.0  
DECnet-Micro/RSX V1.0

**digital**

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may only be used or copied in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital or its affiliated companies.

Copyright © 1985 by Digital Equipment Corporation

The postage-prepaid Reader's Comments form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

DEC	MASSBUS	RT
DECmate	PDP	UNIBUS
DECnet	P/OS	VAX
DECUS	Professional	VAXcluster
DECwriter	Rainbow	VMS
DIBOL	RSTS	VT
<b>digital</b>	RSX	Work Processor

Ethernet is a trademark of Xerox Corporation.

This manual was produced by Networks and Communications Publications.

# Contents

## Preface

## 1 Network Management Utilities (CFE, NCP, VNP)

1.1	Configuration File Editor (CFE).....	1-1
1.1.1	Invoking CFE .....	1-1
1.1.2	Exiting CFE .....	1-2
1.2	Network Control Program (NCP) .....	1-3
1.2.1	Invoking NCP .....	1-3
1.2.2	Exiting NCP .....	1-4
1.2.3	Executing NCP Remotely .....	1-4
1.3	Virtual Network Processor (VNP).....	1-5
1.3.1	Invoking VNP .....	1-5
1.3.2	Exiting VNP .....	1-6
1.4	Using the Network Management Utility Commands .....	1-6
1.4.1	General Command Format .....	1-6
1.4.2	Issuing Commands .....	1-7
1.4.3	Using the Help Facility .....	1-8
1.4.4	Command Prompting .....	1-8
1.4.5	Using the LIST and SHOW Commands .....	1-9
1.4.6	Exit Status and Error Handling .....	1-9
1.5	Command Descriptions .....	1-10
1.5.1	Graphic Conventions .....	1-10
1.5.2	Standard Definitions .....	1-11
1.5.3	RSX-11M/M-PLUS CFE, NCP, and VNP Command Summary .....	1-13
	CLEAR ALIAS .....	1-14
	CLEAR CIRCUIT .....	1-15
	CLEAR EXECUTOR .....	1-16
	CLEAR EXECUTOR NODE .....	1-17
	CLEAR LINE .....	1-18
	CLEAR LOGGING .....	1-19
	CLEAR MODULE X25-ACCESS .....	1-22
	CLEAR MODULE X25-PROTOCOL .....	1-23
	CLEAR MODULE X25-SERVER .....	1-24
	CLEAR MODULE X29-SERVER .....	1-24
	CLEAR NODE .....	1-25

CLEAR OBJECT .....	1-28
CLEAR PROCESS .....	1-29
CLEAR SYSTEM .....	1-30
CLEAR TRACE.....	1-31
DEFINE CIRCUIT .....	1-32
DEFINE EXECUTOR .....	1-36
DEFINE LINE .....	1-40
DEFINE LOGGING .....	1-44
DEFINE MODULE X25-ACCESS .....	1-46
DEFINE MODULE X25-PROTOCOL.....	1-47
DEFINE MODULE X25-SERVER .....	1-52
DEFINE MODULE X29-SERVER .....	1-52
DEFINE NODE .....	1-54
DEFINE OBJECT .....	1-57
DEFINE PROCESS .....	1-59
DEFINE SYSTEM .....	1-61
KMX-DUMP .....	1-63
LIST CIRCUIT .....	1-64
LIST EXECUTOR .....	1-65
LIST LINE .....	1-66
LIST LOGGING.....	1-67
LIST MODULE X25-ACCESS .....	1-68
LIST MODULE X25-PROTOCOL .....	1-69
LIST MODULE X25-SERVER.....	1-70
LIST MODULE X29-SERVER.....	1-70
LIST NODE .....	1-71
LIST OBJECT .....	1-72
LIST PROCESS.....	1-73
LIST SYSTEM.....	1-74
LOAD NODE .....	1-75
LOAD VIA .....	1-78
LOOP CIRCUIT .....	1-80
LOOP EXECUTOR + .NODE.....	1-83
LOOP LINE .....	1-85
PURGE CIRCUIT .....	1-86
PURGE LINE.....	1-87
PURGE LOGGING.....	1-88
PURGE MODULE X25-ACCESS .....	1-89
PURGE MODULE X25-PROTOCOL.....	1-90
PURGE MODULE X25-SERVER .....	1-91
PURGE MODULE X29-SERVER .....	1-91
PURGE NODE .....	1-92
PURGE OBJECT .....	1-94
SET ALIAS .....	1-95
SET CIRCUIT .....	1-96
SET EXECUTOR.....	1-98
SET EXECUTOR NODE .....	1-102
SET KNOWN LINES.....	1-103
SET LINE.....	1-106

	SET LOGGING .....	1-110
	SET MODULE X25-ACCESS .....	1-113
	SET MODULE X25-PROTOCOL .....	1-114
	SET MODULE X25-SERVER .....	1-116
	SET MODULE X29-SERVER .....	1-116
	SET NODE .....	1-119
	SET NODE CIRCUIT .....	1-122
	SET OBJECT .....	1-123
	SET PROCESS .....	1-125
	SET SYSTEM .....	1-127
	SET TRACE .....	1-128
	SHOW ALIAS .....	1-129
	SHOW AREA .....	1-130
	SHOW CIRCUIT .....	1-131
	SHOW EXECUTOR .....	1-132
	SHOW LINE .....	1-133
	SHOW LOGGING .....	1-134
	SHOW MODULE X25-ACCESS .....	1-136
	SHOW MODULE X25-PROTOCOL .....	1-137
	SHOW MODULE X25-SERVER .....	1-139
	SHOW MODULE X29-SERVER .....	1-139
	SHOW NODE .....	1-141
	SHOW OBJECT .....	1-143
	SHOW PROCESS .....	1-144
	SHOW SYSTEM .....	1-145
	SHOW TRACE .....	1-146
	TELL .....	1-147
	TRIGGER NODE .....	1-148
	TRIGGER VIA .....	1-150
	ZERO CIRCUIT .....	1-152
	ZERO EXECUTOR .....	1-153
	ZERO LINE .....	1-154
	ZERO MODULE X25-PROTOCOL .....	1-155
	ZERO MODULE X25-SERVER .....	1-156
	ZERO MODULE X29-SERVER .....	1-156
	ZERO NODE .....	1-157
	ZERO SYSTEM .....	1-158
1.5.4	RSX-11S NCP Command Summary .....	1-159
	LOOP EXECUTOR/NODE .....	1-160
	SET CIRCUIT .....	1-162
	SET EXECUTOR HOST .....	1-163
	SET LINE .....	1-164
	SET LOGGING CONSOLE .....	1-165
	SHOW CIRCUIT .....	1-166
	SHOW EXECUTOR .....	1-167
	SHOW LINE .....	1-168
	SHOW LOGGING CONSOLE .....	1-169
	SHOW NODE .....	1-170
	SHOW SYSTEM .....	1-171

ZERO CIRCUIT .....	1-172
ZERO EXECUTOR .....	1-173
ZERO LINE .....	1-174
ZERO NODE .....	1-175
ZERO SYSTEM .....	1-176

## **2 Console Carrier Requester (CCR)**

2.1	Running CCR .....	2-1
2.2	CCR Special Characters .....	2-2
2.3	CCR Error Handling .....	2-2

## **3 Event File Interpreter (EVF)**

3.1	Invoking EVF .....	3-1
3.2	Exiting EVF .....	3-2
3.3	EVF Command Format .....	3-2
3.3.1	Using the EVF Default Command Line .....	3-3
3.4	EVF Option Descriptions .....	3-4
3.4.1	Using Multiple Option Arguments .....	3-6
3.4.2	File Control Options .....	3-6
3.4.2.1	The Output File Option .....	3-7
3.4.2.2	The Spooling Option .....	3-7
3.4.2.3	The Delete Option .....	3-8
3.4.3	Output File Format Control .....	3-8
3.4.4	Event Selection Options .....	3-9
3.4.4.1	The ALL Event Option .....	3-9
3.4.4.2	The DATE Options .....	3-9
3.4.4.3	The Entity Filter Option .....	3-10
3.4.4.4	The Event Specification Filter Option .....	3-12
3.4.4.5	The Source Node Filter Option .....	3-13
3.4.4.6	/SO Switch – Specify Source Node Filter .....	3-13
3.5	EVF Error Messages .....	3-14

## **4 KMX/KMY Dump Analyzer (KDA)**

4.1	Invoking and Exiting KDA .....	4-1
4.2	Using the KDA Command .....	4-2

## **5 LAT Control Program (LCP)**

5.1	INVOKING and EXITING LCP .....	5-2
5.2	LAT Characteristics .....	5-2
5.3	LCP COMMANDS: Format and Syntax .....	5-3
5.3.1	CREATE .....	5-5

5.3.2	DEFINE .....	5-6
5.3.3	EXIT .....	5-9
5.3.4	HELP .....	5-9
5.3.5	LIST CHARACTERISTICS .....	5-10
5.3.6	SET .....	5-11
5.3.7	SHOW CHARACTERISTICS .....	5-14
5.3.8	SHOW COUNTERS .....	5-16
5.3.9	START .....	5-18
5.3.10	STOP .....	5-21
5.3.11	ZERO [COUNTERS] .....	5-22

## 6 Network Crash Dump Analyzer (NDA)

6.1	NDA Operation .....	6-2
6.2	System Requirements .....	6-3
6.3	Obtaining a Crash Dump .....	6-3
6.4	Running NDA .....	6-4
6.4.1	Running NDA as an Installed Task .....	6-4
6.4.2	Running NDA as an Uninstalled Task .....	6-4
6.5	NDA Command Syntax .....	6-5
6.6	Indirect Command Files .....	6-8
6.7	NDA Switches .....	6-8
6.7.1	Analysis Control Switches .....	6-8
6.7.2	Function Control Switches .....	6-11
6.8	NDA Error Messages .....	6-14

## 7 Network Display Program (NTD)

7.1	Invoking NTD .....	7-2
7.2	NTD Commands .....	7-3
7.2.1	Immediate Mode Commands .....	7-3
7.2.2	Command Mode Commands .....	7-5
7.3	NTD Resource Display Format .....	7-6
7.4	NTD Node Summary Display Format .....	7-10
7.5	NTD Area Summary Display Format .....	7-11

## 8 Queue Manager (QUE)

8.1	FTS and QUE Components .....	8-1
8.2	Invoking QUE .....	8-3
8.3	QUE Command Syntax .....	8-3
8.3.1	Initializing/Deleting and Starting/Stopping the Queue and the Processor .....	8-4
8.3.2	Assigning and Deassigning the Processor .....	8-5
8.3.3	Holding, Releasing, and Deleting FTS User Jobs .....	8-6
8.4	Related Queue Management Functions .....	8-7

8.4.1	Purging FTSSYS.LOG and FAL.LOG .....	8-7
8.4.2	Listing FTSQUE and FTS Jobs .....	8-8
8.4.3	Listing and Purging URB Files .....	8-9

## 9 Trace Interpreter Task (TRI)

9.1	Invoking TRI .....	9-1
9.2	Exiting TRI .....	9-2
9.3	TRI Command Format .....	9-2
9.4	Contents of Trace Output .....	9-4
9.5	TRI Error Messages .....	9-5

## 10 Utility Error Messages

10.1	CCR Error Messages .....	10-3
10.2	CFE Error Messages .....	10-5
10.3	EVF Error Messages .....	10-24
10.4	HLD Error Messages .....	10-29
10.5	NCP Error Messages .....	10-36
10.5.1	RSX-11M/-PLUS Error Message Format .....	10-36
10.5.2	RSX-11S Error Message Format .....	10-38
10.5.3	NCP Error Messages .....	10-38
10.6	NDA Error Messages .....	10-56
10.7	NTD Error Messages .....	10-61
10.8	NTINIT Error Messages .....	10-62
10.9	NTL Error Messages .....	10-66
10.10	TRI Error Messages .....	10-79
10.11	VNP Error Messages .....	10-82

## A Process Names

## B Object Type Codes

### Examples

7-1	Sample Default Resource Display .....	7-9
7-2	Sample Old Resource Display .....	7-9
7-3	Sample Node Summary Display .....	7-11
7-4	Sample Area Summary Display .....	7-12

### Figure

8-1	FTS and QUE Components .....	8-3
-----	------------------------------	-----

## Tables

3-1	EVF Output Options .....	3-4
3-2	EVF Input Options .....	3-5
4-1	KDA Command Switches .....	4-2
5-1	LAT Command Format .....	5-3
6-1	NDA File Default Values .....	6-6
6-2	NDA Analysis Control Switches .....	6-9
6-3	NDA Function Control Switches .....	6-11
7-1	Immediate Mode Commands .....	7-3
7-2	Command Mode Commands .....	7-5
8-1	FTS and QUE Components .....	8-2
9-1	Trace Interpreter Switches .....	9-3
A-1	DECnet-RSX Processes .....	A-1
B-1	Object Type Codes .....	B-1



# Preface

The *DECnet-RSX Guide to Network Management Utilities* presents information needed to use the utilities that manage a DECnet-RSX node within a DECnet network. The term DECnet-RSX collectively refers to four DECnet products:

- DECnet-11M, which runs on RSX-11M
- DECnet-11M-PLUS, which runs on RSX-11M-PLUS
- DECnet-Micro/RSX, which runs on Micro/RSX
- DECnet-11S, which runs on RSX-11S

This manual also describes network management utilities for RSX-11 PSI and DECnet-RSX/PSI. The term DECnet-RSX/PSI refers to an RSX system that runs RSX-11 PSI and DECnet-RSX simultaneously.

The utilities described in this manual are used to manage DECnet-11M V4.2, DECnet-11S V4.2, DECnet-11M-PLUS V3.0, and DECnet-Micro/RSX V1.0 nodes. You can use these utilities to manage both local and remote DECnet systems, which may be other than RSX based. Although Phase IV DECnet provides for cross-system network command operation, there are also some system-specific commands. RSX system-specific commands are flagged in the *DECnet-RSX Network Management Reference Card*. For implementations other than DECnet-RSX, refer to the network documentation for that system.

## **Intended Audience**

This manual is intended for anyone who is responsible for building, maintaining, and managing the network. In this manual, all such people are collectively referred to as the network manager.

## **Structure of This Manual**

This manual is tab divided into four distinct parts. The parts and their contents are summarized below.

### **Part I CFE, NCP, and VNP**

Chapter 1      Network Management Utilities (CFE, NCP, VNP)

Describes the format, parameters, and usage of all CFE, NCP, and VNP commands.

### **Part II Network Management Tools**

Each of the chapters in Part II describes the operation of a network management tool, including commands for its operation and examples of its use and/or output (where applicable).

Chapter 2      Console Carrier Requester (CCR)

Chapter 3      Event File Interpreter (EVF)

Chapter 4      KMX/KMY Dump Analyzer (KDA)

Chapter 5      LAT Control Program (LCP)

Chapter 6      Network Crash Dump Analyzer (NDA)

Chapter 7      Network Display Program (NTD)

Chapter 8      Queue Manager (QUE)

Chapter 9      Trace Interpreter Task (TRI)

## **Part III Error Messages**

### Chapter 10 Utility Error Messages

Summarizes all error messages for the following tools and utilities: CCR, CFE, EVF, HLD, LCP, NCP, NDA, NTD, NTINIT, NTL, TRI, VNP.

## **Part IV Appendixes**

### Appendix A Process Names

Lists the valid process names for DECnet-RSX/PSI.

### Appendix B Object Type Codes

Lists the object type codes currently in use.

## **Associated Documents**

Before reading this manual, you should have a working knowledge of DECnet and the RSX-11 operating system you are using. A prerequisite to the effective use of this manual is familiarity with the overall character of DECnet as described in the following manuals:

*Introduction to DECnet*

*DECnet-RSX Network Management Concepts and Procedures*

Network generation and postinstallation checkout procedures are described in the following manuals:

*DECnet-RSX Network Generation and Installation Guide*

*RSX-11 PSI Generation Guide*

*LAT Network Manager's Guide*

## Acronyms

The following acronyms for DECnet-RSX and PSI components are used in this manual:

ACK	Positive acknowledge message
BCUG	Bilateral closed user group
CCB	Communication control buffer
CCITT	Comite Consultatif International Telephonique et Telegraphique
CCR	Console carrier requester
CCS	Console carrier server
CDA	Crash Dump Analyzer
CEX	Communications Executive
CFE	Configuration File Editor
CSR	Control status register
CUG	Closed user group
DAP	Data Access Protocol
DCB	Device control block
DCE	Data circuit-terminating equipment
DDCMP	Digital Data Communications Message Protocol
DDM	Device driver module
DECS	Digital Ethernet Communications Server
DLC	Data link control process

DLL	Down-line system loader
DLM	Data link mapping
DLX	Direct line access controller
DNA	Digital Network Architecture
DSR	Dynamic storage region
DTE	Data terminal equipment
DTR	DECnet test receiver
DTS	DECnet test sender
DUK	Dump KMX task
DUM	Up-line system dumper
ECL	End Communication layer
EVF	Event File Interpreter
EVL	Event Logger
EVR	Event-logging receiver
FAL	File Access Listener
FRMR	Frame reject error
FTQ	File transfer queue manager
FTS	File Transfer Spooler
HLD	Host Task Loader utility
ICB	Interrupt Control Block

KDA	KMS-11 microcode dump analyzer
KRB	Controller request block
LAPB	Link access procedure, Version B (CCITT recommendation for frame level protocol)
LAT	Local Area Transport
LCN	Logical channel number
LDB	Large data buffer
LLC	Logical link control process
LUN	Logical Unit Number
MIR	Loopback Mirror
MOP	Maintenance Operation Protocol
NAK	Negative acknowledgment message
NCP	Network Control Program
NDA	Network Crash Dump Analyzer
NFT	Network File Transfer utility
NICE	Network Information and Control Exchange Protocol
N(R)	Next expected sequence number
NS	DECnet user interface pseudodevice
NTD	Network Display Program
NTDEMO	Network Display Server
NTINIT	Network initializer

NTL	Network loader
NW	PSI user interface pseudodevice
ODT	On-line debugging tool
PIP	Peripheral Interchange Program
PLI	Packet level interface
PSDN	Packet Switching Data Network
PSI	PSI Packetnet System Interface
PVC	Permanent virtual circuit
QUE	Queue manager
RAM	Random access memory
RDB	Receive data buffer
RNR	Receive not ready
SCB	Status Control Block
SDB	Small data buffer
SLD	Satellite task loader
SVC	Switched virtual circuit
TDM	Time-division-multiplexed bus
TKB	Task builder
TRI	X.25 trace interpreter task
UCB	Unit Control Block

UFD	User file directory
UIC	User identification code
UMR	UNIBUS mapping register
URB	User Request Block
VMR	Virtual Monitor Console Routine
VNP	Virtual Network Processor
XDT	Executive Debugging Tool
XPT	Routing layer

## Conventions Used in This Guide

Convention	Meaning
Monospace	Monospace type indicates examples of system output or user input. System output is in black; user input is in red.
[ ]	Square brackets indicate that the enclosed data is optional. If a vertical list of options is enclosed, you can specify only one option. Do not type the brackets when you enter the command.
{ }	Braces indicate that you must choose one, and only one, of the enclosed options. Do not type the braces when you enter the command.
( )	Parentheses enclose a set of options that must be specified together or not at all.
Options in lists	The absence of brackets around vertical lists indicates that the items are optional.
UPPERCASE LETTERS	Indicate text that must be entered as shown. Uppercase words can be abbreviated to the first 3 or more unique characters.
<i>italicized lowercase words</i>	Indicate generic terms that must be replaced with specific data. NCP/VNP commands that are system specific are printed in red. All CFE commands are RSX system specific.
<KEY>	Indicates that you should press the specified key. <CTRL/X> indicates that you should press the <CTRL> key simultaneously with the X key, X being the desired letter. Note that unless otherwise specified, every command line is terminated by pressing the <RET> key.
	All numbers are decimal unless otherwise noted. All Ethernet addresses are given in hexadecimal.



## Network Management Utilities (CFE, NCP, VNP)

This chapter describes the network management utilities and summarizes their commands. CFE commands modify the permanent database, while NCP commands modify the volatile (running) database. VNP commands are used to modify the system image file. VNP is not supported for Micro/RSX systems. Each utility is described below, including how to invoke it and how to exit it. Details on how to use the commands and individual command descriptions follow.

### 1.1 Configuration File Editor (CFE)

The Configuration File Editor (CFE) is a privileged utility program that accepts commands from a terminal or from an indirect command file to modify or display parameters in the configuration file, CETAB.MAC, which is produced during network generation. CFE commands operate exclusively on the permanent database. Note that all CFE commands are privileged. To invoke CFE, you must be privileged to access the CETAB.MAC file.

#### 1.1.1 Invoking CFE

Before invoking CFE, you must know the device and UIC under which CETAB.MAC resides. After network generation, CETAB.MAC resides under one of the following UICs:

DECnet-RSX-11M/M-PLUS      [x,54]

DECnet-RSX-11S              [x,64]

where *x* is the network UIC selected during network generation.

There are two ways to invoke CFE:

**1. MCR>RUN CFE**

CFE then prompts

**Enter filename:**

Enter the name of the configuration file you want to modify. If you omit any part of the file specification, the default is SY:[*current-uic*]CETAB.MAC. The CFE utility reads the specified file and constructs a memory-resident version of this file. For large configuration files this process takes several seconds. CFE then prompts

**CFE>**

Enter your CFE command after the prompt and press RETURN.

**2. MCR>CFE @*command file* [/TR]**

where *command file* is the name of an indirect command file containing CFE commands and /TR is an optional switch that allows you to trace the commands as they execute. You can nest command files.

### **1.1.2 Exiting CFE**

There are three ways to exit CFE:

1. Use the EXIT command or <CTRL/Z> in response to the CFE prompt to terminate an editing session and update the configuration file.

**CFE>EXIT**  
**MCR>**

2. Use the EXIT PURGE command in response to the CFE prompt to terminate an editing session, update the configuration file, and delete the old file.

**CFE>EXIT PURGE**  
**MCR>**

3. Use the KILL command in response to the CFE prompt to abort an editing session. The configuration file is not updated, and the old file is retained.

```
CFE>KILL  
MCR>
```

## 1.2 Network Control Program (NCP)

The Network Control Program (NCP) is a utility program that accepts commands from a terminal or from an indirect command file to load, control, monitor, and test the network software, as well as to down-line load an RSX-11S system.

There are two versions of NCP: a full set of commands for DECnet-11M and M-PLUS systems and a subset of these for DECnet-11S. Both sets of commands are described in detail in this chapter and are summarized in the *DECnet-RSX Network Management Reference Card*.

You can execute most NCP commands both locally and remotely (see Section 1.2.3). The *DECnet-RSX Network Management Reference Card* lists all NCP commands supported by DECnet-RSX and flags those that are not executable remotely. It also highlights those commands that are specific to DECnet-RSX only.

### 1.2.1 Invoking NCP

There are three ways to invoke NCP:

#### 1. MCR>NCP

NCP then prompts

```
NCP>
```

Enter your NCP command after the prompt and press RETURN.

#### 2. MCR>NCP *command*

where *command* is any valid NCP command. With this method, you return to MCR after the command is executed.

### 3. MCR>NCP @*command file* [/TR]

where *command file* is the name of an indirect command file containing NCP commands and /TR is an optional switch that allows you to trace the commands as they execute. You can nest command files.

#### 1.2.2 Exiting NCP

Use either the EXIT command or <CTRL/Z> in response to the NCP prompt to exit NCP.

#### 1.2.3 Executing NCP Remotely

Most DECnet-RSX NCP commands can be issued at your terminal for execution at any Phase IV or Phase III remote node. In addition, you can issue many NCP commands not executable by DECnet-RSX. This allows you to use NCP commands that are supported by other DECnet implementations but not by DECnet-RSX. For more information about these other NCP commands, see the DECnet documentation of the remote node.

Remote command execution is useful in many ways; for example, you can use it to modify parameters or display information at a remote node, or you can down-line load a node that is not adjacent to your system. Remote execution can be accomplished in two ways:

- To execute a single NCP command at a remote node, issue it with the TELL prefix (see the TELL command description in this chapter). The *DECnet-RSX Network Management Reference Card* summarizes all NCP commands and flags the few that cannot be executed with the TELL prefix.
- To execute a series of NCP commands at a remote node, use the SET EXECUTOR NODE command to temporarily set a specified remote node as the executor. Any subsequent commands that you issue are executed at that node until you return execution to your own node by issuing a CLEAR EXECUTOR NODE command. (Both commands are described in this chapter.) If you exit NCP, control automatically reverts to the local node.

When you issue commands to be executed at a remote node, you may have to specify access control information to access the node. You can append the access control information to the node name, or you can specify an alias node name that includes access control information. Section 1.5.2 provides more information on the format of access control information.

The remote node interprets all commands according to its own network management software and uses any required default parameters from its own database. Therefore, you must issue the command as if it were originating at the remote node. Consult the appropriate DECnet documentation if the node is other than DECnet-RSX.

## 1.3 Virtual Network Processor (VNP)

The Virtual Network Processor (VNP) is a privileged utility program. VNP is not supported on Micro/R SX systems. It accepts commands from a terminal or from an indirect command file to modify or display the DECnet-RSX software configuration as contained in an RSX-11 system image file. All VNP commands are privileged and operate exclusively on the system image file. VNP does not support commands that cause displays to be written to a file

Before using VNP, you must use the Virtual Monitor Console Routine (VMR) to set the NETUIC in the RSX system image file. Use one of the following commands:

```
VMR>SET /NETUIC =[xxx,54] (for RSX-11M/M-PLUS)
```

```
VMR>SET /NETUIC =[xxx,64] (for RSX-11S)
```

VNP uses the NETUIC when it searches for the network software on the LB: device. Therefore, it is imperative that the proper NETUIC be set in the system image file and that LB: be assigned.

### 1.3.1 Invoking VNP

There are two ways to invoke VNP:

1. MCR>RUN VNP

VNP prompts

```
Enter filename:
```

Enter the name of the operating system file you want to modify. If you do not specify the device and UIC, the SY: device and the current UIC are used. (There is no default file name.) VNP then prompts

```
VNP>
```

Enter your VNP command after the prompt and press RETURN.

2. MCR>VNP @*command file* [/TR]

where *command file* is the name of an indirect command file containing VNP commands and /TR is an optional switch that allows you to trace the commands as they execute. You can nest command files.

### 1.3.2 Exiting VNP

Use either the EXIT command or <CTRL/Z> in response to the VNP prompt to exit VNP.

## 1.4 Using the Network Management Utility Commands

The following sections contain general information on command usage.

### 1.4.1 General Command Format

Most commands consist of three parts: the command verb, a component on which the command operates, and one or more parameters that further qualify the action to be taken on the component. Examples of the general command format follow:

Command Verbs	Components	Parameter Lists
CLEAR	EXECUTOR	HOST RECEIVE PASSWORD TRANSMIT PASSWORD
CLEAR	{LINE <i>line-id</i> KNOWN LINES}	{ALL COUNTER TIMER}

For each command, you must supply the verb, one component option, and (usually) one or more of the parameters. Generally, the order in which you specify parameters does not matter. However, some commands have parameters that are mutually exclusive for the component. For example, in the CLEAR LINE command above, you must select either ALL or COUNTER TIMER (as indicated by the braces), whereas you can select one or more of the parameters for CLEAR EXECUTOR. On many of the SHOW and LIST commands, you do not have to specify any parameters because there is a default. The system will prompt you if you omit a required parameter.

### 1.4.2 Issuing Commands

Enter the command keywords separated by spaces.

**Continuing a command.** To continue a long command to the next line, insert a hyphen (-) where you wish to break the line; then enter <RET>. You can continue the command following the prompt on the next line.

#### Example:

```
CFE>DEFINE LINE DZ-1-2 DUPLEX FULL SPEED 9600 -<RET>  
CFE>VECTOR 360 CSR 160120 <RET>  
CFE>
```

Any necessary spaces in the command must be entered even if one occurs just before the hyphen.

**Abbreviating command keywords.** When entering a command, you can abbreviate any keyword to the first three or more unique characters. For example, the following two versions of the same command are equally valid:

```
NCP>SHOW MODULE X25-ACCESS ALL DESTINATIONS CHARACTERISTICS
```

```
NCP>SHO MOD X25-A ALL DES CHA
```

Note that X25-ACCESS is abbreviated to the first five unique characters to differentiate it from other entities beginning with the X25- character string. For clarity and consistency, all examples in this manual show the full command syntax.

**Entering comment lines.** You can enter comment lines by prefacing each line of text with a semicolon (;).

### 1.4.3 Using the Help Facility

Use the HELP command whenever you need assistance in selecting network management commands and parameter options. Execution of this command by any of the three utilities causes a summary of help text to be displayed.

#### Example:

```
>NCP  
NCP>HELP
```

Use the HELP keyword with a selected command and/or component to display the portion of the help file relevant to that command.

#### Example:

```
>NCP  
NCP>HELP LINE
```

### 1.4.4 Command Prompting

Command prompting provides additional assistance when using network management commands. At any point where you can type a space, type <RET> instead. This produces a prompt requesting the required information. If you need additional information after a prompt, you can type <ESC> to display additional help information.

#### Example:

```
NCP>SET <RET>  
Set component: <ESC>
```

Options are:

ALIAS	KNOWN CIRCUITS	LOGGING	PROCESS
CIRCUIT	KNOWN LINES	NODE	SYSTEM
EXECUTOR	LINE	OBJECT	

### 1.4.5 Using the LIST and SHOW Commands

NCP, CFE, and VNP provide commands that enable you to display information about network components on your terminal. NCP provides the additional capability to send display information to a file.

For more information about the SHOW and LIST commands and examples of the type of information displayed by these commands, see the *DECnet-RSX Network Management Concepts and Procedures* manual.

### 1.4.6 Exit Status and Error Handling

When CFE, NCP, or VNP terminates, it sets a bit that can be inspected using the normal RSX exit status handling techniques described in RSX-11M/M-PLUS operating system documentation. The utility sets the exit status to indicate the worst case for the commands executed. The exit status is especially useful when used in conjunction with the indirect command file processor's support of this operating system feature. The possible values and their meanings are summarized below.

Status	Numeric Value	Meaning
Success	1	The command executed as requested.
Warning	0	Minor error. Processing continues.
Error	2	The command was accepted, but failed to execute completely. The system waits for a new command.
Severe Error	4	The command was not accepted. Processing stops and control returns to the monitor.

#### NOTE

The following NCP commands do not affect the exit status:

```
CLEAR EXECUTOR NODE  
EXIT (or <CTRL/Z>)  
HELP  
SET EXECUTOR NODE
```

If the command executes successfully, you receive the utility prompt on the next line, whereupon you can issue another command. If the command is unsuccessful, you receive an error message on the next line indicating the reason for the error, followed by the utility prompt.

**Example:**

```
MCR>NCP
NCP>SET LINE DMC-O TOP
NCP>LOAD NODE XYZ
NCP -- Load failed, line communication error
NCP>
```

Error messages for all three utilities are provided in Chapter 10.

## 1.5 Command Descriptions

The rest of this chapter is devoted to descriptions of the individual commands and their parameters. If you need more information on the command components or their parameters, refer to the *DECnet-RSX Network Management Concepts and Procedures* for a more in-depth discussion.

Commands are presented in alphabetical order. The command name and the utility (or utilities) to which it applies are printed at the top of the page (many NCP and VNP commands have the same format and are described together). The full DECnet-RSX-11M/M-PLUS set is described first. The RSX-11S subset is described at the end of the chapter (NCP/S appears at the top of those pages). All commands are designated as privileged (P) or nonprivileged (NP) at the top of the page next to the command name. On combined NCP/VNP commands, an NP designator applies to NCP commands only, as all VNP commands are privileged.

### 1.5.1 Graphic Conventions

Review the graphic conventions outlined at the front of this manual, especially the usage of braces and brackets [ ]. **Shaded text** indicates commands or parameters that are valid for PSI users only.

## 1.5.2 Standard Definitions

The following parameters and user-supplied variables are frequently used in the command descriptions. They are described here to avoid needless repetition. Exceptions to these definitions are indicated in individual parameter descriptions. More information about most of these parameters can be found in the *DECnet-RSX Network Management Concepts and Procedures* manual.

*acc-con-info* Access control information in one of the following forms:

*luser-id[/password[/account]]*

USER *user-id* [PASSWORD *password* [ACCOUNT *account*]]

NOTE: DECnet-RSX does not use the account field.

*alias-name* An alternative name for a network node. 1 to 6 alphanumeric characters including at least 1 alphabetic character.

*area-number* A decimal integer in the range of 1 to 63.

*circuit-id* A DECnet circuit identification in the form *dev-c[-u][.t]* or a PSI circuit identification consisting of 1 to 6 alphanumeric characters.

*csr-address* CSR address in the range of 160000 to 177774 (octal).

*dest-name* An X.25 destination name consisting of 1 to 6 alphanumeric characters.

*dte-address* A string of 1 to 15 decimal digits forming a DTE address.

*E-address* An Ethernet address consisting of a string of 12 hexadecimal digits in the form *nn-nn-nn-nn-nn-nn*.

<i>event-list</i>	A list of events that belong to the same class, such as, <i>event-class.type</i> .
<i>file</i>	A file specification, which is interpreted according to the executor node's file system. The standard format for RSX is <i>dev:[uic]filename.type.ver</i> .
<i>group-name</i>	An X.25 group name consisting of 1 to 6 alphanumeric characters.
<i>line-id</i>	A line identification in the form <i>dev-c[-u]</i> .
<i>node-address</i>	An area-number followed by a period and a decimal integer in the range of 1 to 1023. When referring to the local area, the area number and period are optional.
<i>node-id</i>	A node name (see <i>node-name</i> ) or node address (see <i>node-address</i> ).
<i>node-name</i>	1 to 6 alphanumeric characters, including at least 1 alphabetic character.
<i>object-id</i>	An RSX task name (1 to 6 alphanumeric characters) or an object type code (see <i>type-code</i> ).
<i>service-password</i>	A hexadecimal number used as a password for certain service requests. For DDCMP circuits, it is in the range of 0 to FFFFFFFF; for Ethernet circuits, it is in the range of 0 to FFFFFFFFFFFFFFFF (up to 16 hexadecimal digits).
<i>process-name</i>	A name of a DECnet-RSX process. The process name is 1 to 3 characters in length. For a list of valid process names, see Appendix A.
<i>range</i>	A subaddress range consisting of one or two 1- to 4-digit integers in the range of 0 to 9999. If you specify two subaddresses, they must be separated by a hyphen (-), and the second must be greater than the first (e.g., 10-100). See <i>PSI Network-specific Information</i> for the maximum length of a subaddress on your network.

<i>type-code</i>	An object type code in the range of 1 to 255. Use an object type code from Appendix B.
<i>vector-address</i>	A vector address in the range of 0 to 774 (octal).
CHARACTERISTICS	(Default for CFE) Displays static information for the specified component. This type of information is automatically displayed for all CFE LIST commands.
COUNTERS	(NCP only) Displays error and performance statistics for the specified component.
STATUS	(NCP only) Displays dynamic information for the specified component.
SUMMARY	(Default for VNP and NCP, invalid for CFE) Displays only the most useful information for the specified component.
SCOPE <i>scope</i>	(Valid for privileged users only) Qualifies the extent to which the specified operation applies (default: requestor's terminal). Possible values for <i>scope</i> are:  GLOBAL     The operation applies to all terminals.  TERMINAL   The operation applies to the specified <i>term-id</i> terminal only, where <i>term-id</i> is a standard RSX terminal device identification.
TO <i>file</i>	(Valid for NCP only) Directs output to the specified file or device at the local node (default: information displayed at the calling terminal). If you specify an existing file as output, the information is appended to that file. Information is transferred in ASCII in the same format as it is displayed at the terminal. (See the description of <i>file</i> , above, for format.)

### 1.5.3 RSX-11M/M-PLUS CFE, NCP, and VNP Command Summary

This section describes all of the CFE and NCP commands supported by RSX-11M/M-PLUS. VNP commands are described in tandem with the corresponding NCP command. RSX-11S NCP commands are described in Section 1.4.5.

**CLEAR ALIAS**

Use the NCP CLEAR ALIAS command to remove the specified alias node name(s) from the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
CLEAR { ALL ALIASES }      [[SCOPE] scope]
      { ALIAS alias-name }
      { KNOWN ALIASES }
```

where

ALL ALIASES            (Valid for privileged users only) Removes all alias names.

ALIAS                         Removes only the specified alias name.  
*alias-name*

KNOWN ALIASES                 Removes all alias names for the specified scope.

SCOPE *scope*                 (Invalid for ALL ALIASES) See definition in Section 1.5.2.

**Example:**

```
NCP>CLEAR ALIAS NY5 SCOPE TERMINAL TT2:
```

This command removes the association of alias NY5 with terminal TT2:.

---

**CLEAR CIRCUIT**

Use the CLEAR CIRCUIT command to zero the specified PSI circuit counter timer(s) in the volatile database.

**Format:**

```
CLEAR {CIRCUIT circuit-id } COUNTER TIMER  
      {KNOWN CIRCUITS }
```

where

CIRCUIT                      Zeroes the counter timer for the specified circuit  
*circuit-id*                      only.

KNOWN                      Zeroes the counter timers for all known circuits.  
CIRCUITS

**Example:**

```
NCP>CLEAR CIRCUIT PVC1 COUNTER TIMER
```

This command zeroes the counter timer for PSI circuit PVC1.

## **NCP/VNP**

**P**

### **CLEAR EXECUTOR**

Use the NCP CLEAR EXECUTOR command to remove specified executor node parameters from the volatile database. Use the VNP command to do the same with the system image file.

#### **Format:**

```
CLEAR EXECUTOR  HOST
                  RECEIVE PASSWORD
                  TRANSMIT PASSWORD
```

where

**HOST**                      Removes identification of the host for the executor node. The executor then defaults to itself as host for service operations.

**RECEIVE  
PASSWORD**                  Removes the executor node's receive password.

**TRANSMIT  
PASSWORD**                  Removes the executor node's transmit password.

#### **Example:**

```
NCP>CLEAR EXECUTOR RECEIVE PASSWORD
```

This command removes the receive password.

---

**CLEAR EXECUTOR  
NODE**

Use the **CLEAR EXECUTOR NODE** command to return NCP command execution to the local node after issuing an **NCP SET EXECUTOR NODE** command.

**Restriction:**

You cannot use the **TELL** prefix with this command.

**Format:**

```
CLEAR EXECUTOR NODE
```

**Example:**

```
NCP>SET EXECUTOR NODE BOS  
. . .  
NCP>CLEAR EXECUTOR NODE
```

The first command causes subsequent NCP commands to be executed at node **BOS**. The second command returns NCP command execution to the local node.

**CLEAR LINE**

Use the NCP CLEAR LINE command to unload the specified line(s) and to remove all associated parameters from the volatile database. Use the VNP command to do the same with the system image file. You can also use the NCP command to zero the counter timer(s) for the specified PSI line(s).

**Restriction:**

Before you unload a DECnet line, all circuits associated with the line must be in the OFF state (see SET CIRCUIT). Any PSI line to be unloaded must also be in the OFF state (see SET LINE and SET KNOWN LINES).

**Format:**

$$\text{CLEAR } \left\{ \begin{array}{l} \text{LINE } \textit{line-id} \\ \text{KNOWN LINES} \end{array} \right\} \left\{ \begin{array}{l} \text{ALL} \\ \text{COUNTER TIMER} \end{array} \right\}$$

where

LINE <i>line-id</i>	Clears the specified line only.
KNOWN LINES	Clears all known lines.
ALL	Unloads the specified line(s) and removes all associated line parameters.
COUNTER TIMER	(Invalid for VNP) Zeroes the counter timer(s) for the specified PSI line(s).

**Examples:**

```
NCP>CLEAR LINE DMC-0 ALL
```

This command unloads DECnet line DMC-0 and removes all parameters associated with it, thus making DMC-0 unusable.

```
NCP>CLEAR LINE SDP-1 COUNTER TIMER
```

This command zeroes the counter timer for PSI line SDP-1.

## CLEAR LOGGING

Use the NCP CLEAR LOGGING command to remove logging parameters from the volatile database. Use the VNP command to do the same with the system image file.

**Restriction:**

Whenever a circuit, line, module, node, or sink is specified in a CLEAR LOGGING command, an event list or KNOWN EVENTS parameter must also be included.

**Format:**

```
CLEAR { KNOWN LOGGING
        LOGGING CONSOLE
        LOGGING FILE
        LOGGING MONITOR } [NAME]
        [ EVENTS event-list
          KNOWN EVENTS
          [ CIRCUIT circuit-id
            LINE line-id
            MODULE { X25-PROTOCOL
                   X25-SERVER
                   X29-SERVER }
            NODE node-id
            SINK { EXECUTOR
                  NODE { node-id
                        $HOST } } ] ] ]
```

where

KNOWN LOGGING	Removes parameters for all known logging components.
LOGGING CONSOLE	Removes parameters for console logging component.
LOGGING FILE	Removes parameters for the file logging component.
LOGGING MONITOR	Removes parameters for the monitor logging component.
NAME	Returns the specified logging component to its default name (console: CO0;; file: LB:[1.6]EVENTLOG.SYS; monitor: MON...).

EVENTS <i>event-list</i>	Removes the event class and type(s) specified in <i>event-list</i> for the specified component.
KNOWN EVENTS	Removes all known events that DECnet-RSX can generate for the specified component.
CIRCUIT <i>circuit-id</i>	(Invalid for VNP) Inhibits event logging for the specified circuit (see restriction).
LINE <i>line-id</i>	Inhibits event logging for the specified line (see restriction).
MODULE	(Invalid for VNP) Inhibits event logging for the specified module (see restriction).
NODE <i>node-id</i>	Inhibits event logging for the specified node (see restriction).
SINK	Inhibits logging of the specified event(s) at the specified node (see restriction).
EXECUTOR	(Default) Events are not to be logged at the executor node.
NODE <i>node-id</i>	Events are not to be logged at the specified node.
NODE \$HOST	(Invalid for NCP) Events are not to be logged at the host node.

## Examples:

```
NCP>CLEAR LOGGING CONSOLE EVENT 2 1
```

This command ceases logging of event 2.1 to the console.

```
NCP>SET LOGGING CONSOLE EVENT 2.1 SINK NODE PHL  
NCP>SET LOGGING CONSOLE NAME TT1: EVENT 2.1 STATE ON
```

```
.  
. .  
.
```

```
NCP>CLEAR LOGGING CONSOLE EVENT 2.1 SINK NODE PHL
```

The first command in this example causes access control rejection events (2.1) to be logged on the console on node PHL. The second command enables logging of 2.1 events to TT1: on the local node. The third command reverts event logging to the local node only, and 2.1 events are no longer logged to the console on node PHL.

---

**CLEAR MODULE  
X25-ACCESS**

Use the CLEAR MODULE X25-ACCESS command to remove destination names from the volatile database.

**Format:**

```
CLEAR MODULE X25-ACCESS { ALL DESTINATIONS
                        DESTINATION dest-name
                        KNOWN DESTINATIONS } [[SCOPE]scope]
```

where

ALL DESTINATIONS	Removes all destination names, regardless of scope.
DESTINATION <i>dest-name</i>	Removes only the specified destination name.
KNOWN DESTINATIONS	Removes all destination names for the specified scope.
SCOPE <i>scope</i>	See definition in Section 1.5.2.

**Examples:**

```
NCP>CLEAR MODULE X25-ACCESS ALL DESTINATIONS
```

This command removes all destination names.

```
NCP>CLEAR MODULE X25-ACCESS DESTINATION CUG111 TERMINAL TT2:
```

This command removes destination name CUG111 associated with terminal TT2:.

---

**CLEAR MODULE  
X25-PROTOCOL**

Use the CLEAR MODULE X25-PROTOCOL command to zero the counter timer(s) for the specified DTE(s) or to remove from the volatile database all parameters for the specified DTE(s) or group(s).

**Format:**

```

CLEAR MODULE X25-PROTOCOL { DTE dte-address { ALL
                           KNOWN DTES      { COUNTER TIMER }
                           }
                           { GROUP group-name ALL
                           KNOWN GROUPS }

```

where

DTE	<i>dte-address</i>	Clears the specified DTE only.
KNOWN DTES		Clears all known DTEs.
GROUP <i>group-name</i>		Removes parameters for the specified group only.
KNOWN GROUPS		Removes parameters for all known groups.
ALL		Removes all parameters for the specified DTE(s) or group(s).
COUNTER TIMER		Zeroes the counter timer(s) for the specified DTE(s).

**Examples:**

```
NCP>CLEAR MODULE X25-PROTOCOL DTE 123456789 COUNTER TIMER
```

This command zeroes the counter timer for local DTE 123456789.

```
NCP>CLEAR MODULE X25-PROTOCOL KNOWN GROUPS ALL
```

This command removes parameters for all known groups.

## NCP

P

CLEAR MODULE  
X25-SERVER  
CLEAR MODULE  
X29-SERVER

Use the CLEAR MODULE X25/X29-SERVER commands to zero the counter timer or to remove the specified destination name(s) and all associated parameters from the volatile database.

### Format:

```
CLEAR MODULE { X25-SERVER } { COUNTER TIMER }  
              { X29-SERVER } { DESTINATION dest-name }  
                              { KNOWN DESTINATIONS }
```

where

COUNTER TIMER	Zeroes the server module counter timer.
DESTINATION <i>dest-name</i>	Removes parameters for the specified destination only.
KNOWN DESTINATIONS	Removes parameters for all known destinations.

### Example:

```
NCP>CLEAR MODULE X25-SERVER DESTINATION CHI205 ALL
```

This command removes all parameters for destination CHI205.

---

**CLEAR NODE**

Use the CLEAR NODE command to remove specified node parameters from the volatile database on the local node. Use the VNP command to do the same with the system image file.

**Restriction:**

NAME is the only valid parameter for VNP.

**Format:**

```
CLEAR  NODE node-id    ALL

                                CIRCUIT
                                DIAGNOSTIC FILE
                                DUMP ADDRESS
                                DUMP COUNT
                                DUMP FILE
                                HARDWARE ADDRESS
                                HOST
                                LOAD FILE
                                NAME
                                SECONDARY LOADER
                                SERVICE CIRCUIT
                                SERVICE DEVICE
                                SERVICE PASSWORD
                                TERTIARY LOADER
```

where

NODE <i>node-id</i>	Specifies the node whose parameters are to be removed. You must use a node name if you are removing the circuit identification (see CIRCUIT parameter below).
ALL	Removes all parameters for the specified node. If you specify ALL, you cannot include any other parameters.

CIRCUIT	Removes the circuit identification associated with the node name for node level loopback testing.
DIAGNOSTIC FILE	Removes the identification of the down-line load diagnostics file.
DUMP ADDRESS	Removes the node's up-line dump address.
DUMP COUNT	Removes the up-line dump count.
DUMP FILE	Removes the up-line dump file identification.
HARDWARE ADDRESS	Removes the Ethernet address of the system hardware.
HOST	Removes the host node identification.
LOAD FILE	Removes the down-line load file identification.
NAME	Removes the node name associated with the specified node.
NODE CIRCUIT	Clears the assignment of a node <i>id</i> to a circuit used for loopback testing. This command does not apply to Ethernet nodes.
SECONDARY LOADER	Removes the parameter associated with the file containing secondary loader software for down-line loading to the node.
SERVICE CIRCUIT	Removes the circuit parameter associated with the node for down-line load operations.

SERVICE DEVICE	Removes the service device type.
SERVICE PASSWORD	Removes the password parameter required to trigger the bootstrap mechanism.
TERTIARY LOADER	Removes the parameter associated with the file containing tertiary loader software for down-line loading to the node.

**Example:**

```
NCP>CLEAR NODE BOS ALL
```

This command removes all information for the node BOS from the volatile database.

**CLEAR OBJECT**

Use the NCP CLEAR OBJECT command to remove all object parameters from the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
CLEAR {OBJECT type-code} ALL
      {KNOWN OBJECTS}
```

where

**OBJECT**                      Removes parameters for the specified object only.  
*type-code*

**KNOWN OBJECTS**              Removes parameters for all known objects.

**ALL**                          (Required for NCP; optional for VNP) Specifies that all parameters are to be removed.

**Example:**

```
NCP>CLEAR OBJECT 245 ALL
```

This command removes all parameters for object 245.

---

**CLEAR PROCESS**

Use the NCP CLEAR PROCESS command to unload a process and remove all process parameters from the volatile database. Use the VNP command to do the same with the system image file.

**Restrictions:**

- You can unload a process only when it is not being used.
- If you clear NW from the volatile database and then reload it using NCP SET PROCESS, you must also issue the SET MODULE X25-PROTOCOL command (first with STATE OFF and then with STATE ON) to update the process pointers to the database.

**Format:**

**CLEAR PROCESS** *process-name*

where

**PROCESS**                      Specifies the process to unload.  
*process-name*

**Example:**

```
NCP>CLEAR PROCESS DLX
```

This command unloads the DLX process and removes all parameters associated with it.

**CLEAR SYSTEM**

Use the NCP CLEAR SYSTEM command to unload the Communications Executive, release any resources being used by the network, and remove all system parameters from the volatile database. Use the VNP command to unload the Communications Executive in the system image file.

**Restrictions:**

- You cannot use the TELL prefix with this command.
- Before you issue NCP CLEAR SYSTEM, the executor must be in the OFF state (see NCP SET EXECUTOR).
- If you are running PSI, the X.25 server module must be in the OFF state (see SET MODULE X25-SERVER).

**Format:****CLEAR SYSTEM**

---

**CLEAR TRACE**

Use the **CLEAR TRACE** command to stop tracing of PSI frames on the specified line(s). For more information on the trace interpreter task, see Chapter 9.

**Format:**

```
CLEAR TRACE {LINE line-id
             {ACTIVE LINES}}
```

where

**LINE *line-id***            Disables tracing on the specified line only.

**ACTIVE LINES**            Disables tracing on all active lines.

**Example:**

```
NCP>CLEAR TRACE LINE SDP-0
```

This command disables tracing of PSI frames on line SDP-0.

**DEFINE CIRCUIT**

Use the DEFINE CIRCUIT command to modify circuit parameters in the permanent database. You can also use this command to create parameters for PVCs.

**Restriction:**

The values used for MAXIMUM DATA and MAXIMUM WINDOW must be the same as the packet size and window size agreed upon by the network authorities for the PVC.

**Format:**

```

DEFINE {CIRCUIT circuit-id
        {KNOWN CIRCUITS}}
CHANNEL channel-number
COST cost
COUNTER TIMER seconds
DTE dte-address
HELLO TIMER seconds
MAXIMUM ROUTERS number
MAXIMUM DATA byte-count
MAXIMUM RECALLS retry-count
MAXIMUM WINDOW block-count
MULTIPOINT ACTIVE active-ratio
NUMBER dte-address
RECALL TIMER seconds
ROUTER PRIORITY priority
SERVICE {DISABLE}
          {ENABLE}
STATE {OFF}
       {ON}
TRIBUTARY trib-address
USAGE {INCOMING}
      {OUTGOING}

```

where

CIRCUIT  
*circuit-id*

Modifies or creates parameters for the specified circuit only.

KNOWN CIRCUITS	Modifies (or creates) parameters for all known circuits.
CHANNEL <i>channel-number</i>	(DECnet DLM and PSI circuits only) Specifies the logical channel number for a PVC (range: 0 to 4095).
COST <i>cost</i>	(DECnet circuits only) Specifies the routing cost of the circuit. Messages will travel between nodes along the path with the smallest total cost (range: 1 to 25; default: varies according to circuit type).
COUNTER TIMER <i>seconds</i>	(PSI circuits only) Sets a timer whose expiration causes the circuit counters to be logged and zeroed (range: 0 to 65,535).
DTE <i>dte-address</i>	(DECnet DLM and PSI circuits only) Specifies a local DTE for a PVC.
HELLO TIMER <i>seconds</i>	(DECnet circuits only) Specifies the frequency of routing hello messages sent to an adjacent node on the circuit (range: 1 to 8191).
MAXIMUM ROUTERS <i>number</i>	(DECnet Ethernet circuits only) Specifies the maximum number of routing nodes that an Ethernet circuit can maintain (range: 0 to 32; default: 5).
MAXIMUM DATA <i>byte-count</i>	(DECnet DLM and PSI circuits only) Specifies the maximum packet size for a PVC. For DLM SVCs, this parameter specifies the maximum packet size to be negotiated with the network, if allowed. This parameter value must be at least 5 bytes less than the maximum data size specified for PSI lines (see DEFINE LINE). By default, packet size takes the network value specified by the MAXIMUM DATA parameter in the DEFINE MODULE X25-PROTOCOL command. See <i>PSI Network-specific Information</i> for the network value of this parameter (range: 16 to 4054).

<b>MAXIMUM RECALLS</b> <i>retry-count</i>	(DECnet DLM circuits only) Specifies the maximum number of times that DECnet should attempt to place a call to establish an SVC (range: 0 to 255; default: 5).
<b>MAXIMUM WINDOW</b> <i>block-count</i>	(DECnet DLM and PSI circuits only) Specifies the maximum number of blocks allowed to be outstanding on the circuit for PSI PVCs. For DLM SVCs, this specifies the maximum number of outstanding blocks to be negotiated with the network, if allowed. By default, window size takes the network value specified by the <b>MAXIMUM WINDOW</b> parameter in the <b>DEFINE MODULE X25-PROTOCOL</b> command. See <i>RSX-11 PSI Network-specific Information</i> for the network value of this parameter (range: 1 to 7).
<b>MULTIPOINT ACTIVE</b> <i>active-ratio</i>	(DECnet non-DMP/DMV multipoint control circuits only) Specifies the rate at which a tributary in the active multipoint state is polled (range: 1 to 255).
<b>NUMBER</b> <i>dte-address</i>	(DECnet DLM circuits only) <b>For outgoing calls:</b> specifies a remote DTE address used by DECnet when trying to establish an SVC. <b>For incoming calls:</b> specifies the address the incoming call request must have in order for DECnet to establish the SVC. An alternate method is to use the <b>SUBADDRESSES</b> parameter of the <b>DEFINE EXECUTOR</b> command (if the network supports subaddresses).
<b>RECALL TIMER</b> <i>seconds</i>	(DECnet DLM circuits only) Sets the interval to wait after a failed call before attempting to place another call to establish an SVC (range: 0 to 65535; default: 30).
<b>ROUTER PRIORITY</b> <i>priority</i>	(DECnet Ethernet circuits only) Specifies the priority that the executor node on this circuit is to have in selecting a designated router for the circuit (range: 0 to 255; default: 64).

**SERVICE** (DECnet circuits only) Specifies whether the circuit is enabled or disabled for down-line loading and loopback testing.

**STATE** Sets the circuit's operational state.

**OFF** The circuit is not available for use.

**ON** The circuit is available for normal use. For circuits owned by the executor, the state of the corresponding line(s) must be ON.

**TRIBUTARY** (DECnet multipoint circuits only) Specifies the data link physical address of the multipoint circuit  
*trib-address* (range: 1 to 255).

**USAGE** (DECnet DLM circuits only) Specifies the availability of a DLM SVC for either incoming or outgoing switched calls (default: incoming).

**Examples:**

```
CFE>DEFINE CIRCUIT DZ-3 STATE OFF
```

This command makes circuit DZ-3 unavailable for use.

```
CFE>DEFINE KNOWN CIRCUITS MAXIMUM RECALLS 10
```

This command sets to 10 the maximum number of times DECnet should attempt to place a call on all known circuits.

**DEFINE EXECUTOR**

Use the DEFINE EXECUTOR command to modify executor node parameters in the permanent database.

**Format:**

```

DEFINE EXECUTOR    ADDRESS node-address
                   AREA MAXIMUM COST number
                   AREA MAXIMUM HOPS number
                   BROADCAST ROUTING TIMER seconds
                   HOST node-address
                   IDENTIFICATION id-string
                   INACTIVITY TIMER seconds
                   INCOMING TIMER seconds
                   MAXIMUM ADDRESS node-address
                   MAXIMUM AREAS number
                   MAXIMUM BROADCAST NONROUTERS number
                   MAXIMUM COST number
                   MAXIMUM HOPS number
                   MAXIMUM LINKS number
                   MAXIMUM NODE COUNTERS number
                   NAME node-name
                   OUTGOING TIMER seconds
                   RETRANSMIT FACTOR number
                   ROUTING TIMER seconds
                   SEGMENT BUFFER SIZE number
                   SUBADDRESSES range
                   VERIFICATION [STATE] {OFF}
                                       {ON }

```

where

ADDRESS <i>node-address</i>	Specifies the address of the executor node.
AREA MAXIMUM COST <i>number</i>	Specifies the total path cost allowed to any remote area (range: 1 to 1022; default: 1022).

<b>AREA MAXIMUM HOPS</b> <i>number</i>	Specifies the maximum number of routing hops allowed to any remote area (range: 1 to 30; default: 10).
<b>BROADCAST ROUTING TIMER</b> <i>seconds</i>	Sets a timer whose expiration forces a routing update on Ethernet circuits, which produces a routing configuration message for each adjacent node (range: 1 to 65535; default: 30)
<b>HOST</b> <i>node-address</i>	Specifies the host node address for down-line load, trigger, and up-line dump operations.
<b>IDENTIFICATION</b> <i>id-string</i>	Specifies the text identification string for the executor node (range: 1 to 32 characters). You must use quotation marks (") to delimit any string containing blanks or tabs. To indicate a quotation mark within a quoted string, use a double quotation mark (to distinguish it from a string delimiter).
<b>INACTIVITY TIMER</b> <i>seconds</i>	Specifies the maximum time the executor will allow a link to remain idle (no user data traffic) before it checks to see if the circuit still works (range: 1 to 255).
<b>INCOMING TIMER</b> <i>seconds</i>	Specifies the maximum time a process has to answer an incoming connect request. If the process does not answer the connect request within this time interval, the node will reject the connect request on behalf of the process (range: 1 to 255).
<b>MAXIMUM ADDRESS</b> <i>node-address</i>	Specifies the maximum node address of the executor or any remote node known to the executor (range: 1 to 1023).
<b>MAXIMUM BROADCAST NONROUTERS</b> <i>number</i>	Specifies the maximum number of nonrouting nodes that the executor node can have on all its Ethernet circuits (range: 1 to 1022; default: 64).

<b>MAXIMUM AREAS</b> <i>area-number</i>	Specifies the maximum area number of the executor or any remote area known to the executor (range: 1 to 64).
<b>MAXIMUM COST</b> <i>number</i>	Specifies the maximum total path cost allowed to any node (range: 1 to 1022; default: 1022).
<b>MAXIMUM HOPS</b> <i>number</i>	Specifies the maximum number of routing hops allowed from the executor node to any other node (range: 1 to 30; default: 10).
<b>MAXIMUM LINKS</b> <i>number</i>	Specifies the maximum number of active logical links for the executor node (range: 1 to 255).
<b>MAXIMUM NODE COUNTERS</b> <i>number</i>	Specifies the maximum number of node counters allowed. If you require more counters, the existing ones are reassigned on demand in the same order as they were originally assigned. Use a decimal integer greater than 1, but not exceeding the executor maximum address (range: 1 to 170).
<b>NAME</b> <i>node-name</i>	Specifies the node name of the executor.
<b>OUTGOING TIMER</b> <i>seconds</i>	Specifies the maximum time the executor node will wait for a pending connect request to be answered at a destination node. If the request is not answered in this time interval, the source process receives an error indication (range: 1 to 255).
<b>RETRANSMIT FACTOR</b> <i>number</i>	Specifies the number of times the executor will restart the retransmission timer before the logical link is disconnected (range: 1 to 255).
<b>ROUTING TIMER</b> <i>seconds</i>	Sets a timer whose expiration forces a routing update on non-Ethernet circuits (range: 1 to 65,535).

**SEGMENT BUFFER SIZE** *number* Specifies the maximum size of transmit buffers (in bytes), thereby controlling the size of the NSP message segment to be sent. This value is the maximum size message that the End Communications layer can transmit; it does not include routing or data link overhead (range: 1 to 4096; default: 18 bytes less than the large buffer size specified in the DEFINE SYSTEM command). The segment buffer size can never be greater than the large buffer size minus 18 bytes.

**SUBADDRESSES** *range* Specifies a range of local DTE subaddresses that are valid on any PSI circuit for incoming calls to the executor node.

**VERIFICATION STATE** Sets the logical link access control verification state for the executor node.

OFF The executor node does not verify access control on any incoming requests.

ON (Default) The executor node verifies access control on all incoming connect requests according to the options set for each object (see the discussion of access control in the *DECnet-RSX Network Management Concepts and Procedures* manual).

**Examples:**

```
CFE>DEFINE EXECUTOR ADDRESS 21
```

This command sets the executor address to 21.

```
CFE>DEFINE EXECUTOR MAXIMUM LINKS 20
```

This command sets the maximum number of active logical links to 20.

**DEFINE LINE**

Use the DEFINE LINE command to modify line parameters in the permanent database.

**Format:**

```

DEFINE { LINE line-id
        KNOWN LINES }
        [ CONTROLLER ] CSR csr-address
        CONTROLLER { LOOPBACK
                    NORMAL }
        COUNTER TIMER seconds
        DUPLEX { FULL
               HALF }
        HOLDBACK TIMER milliseconds
        MAXIMUM DATA byte-count
        MAXIMUM RETRANSMITS retry-count
        MAXIMUM WINDOW block-count
        MULTIPOINT DEAD, dead-ratio
        PRIORITY hardware-priority
        RETRANSMIT TIMER milliseconds
        SPEED baud-rate
        STATE { CLEARED
              OFF
              ON }
        UNIT CSR csr-address
        VECTOR vector-address

```

where

LINE <i>line-id</i>	Modifies parameters for the specified line only.
KNOWN LINES	Indicates that parameters for all known lines are to be modified.
CONTROLLER CSR <i>csr-address</i>	Specifies the address of the first CSR for the line controller.

CONTROLLER	(For Ethernet, DMC/DMR, DMP, DHU, and DHV lines) Specifies the controller mode for the line.
	LOOPBACK    Internal device loopback mode.
	NORMAL       Normal operating mode.
COUNTER TIMER <i>seconds</i>	(PSI lines only) Sets a timer whose expiration causes the line counters to be logged and zeroed (range: 0 to 65,535)
DUPLEX	(DECnet DDCMP lines only) Sets the hardware transmission mode to full duplex or half duplex.
HOLDBACK TIMER <i>milliseconds</i>	(PSI lines only) Specifies the length of time that a message acknowledgment can be held back in order to be included with another data message (range: 0 to 65,535).
MAXIMUM DATA <i>byte-count</i>	(PSI lines only) Specifies the maximum frame size (in bytes) for the line. This value must be at least 5 bytes larger than the maximum data size specified for PSI circuits (see the DEFINE MODULE X25-PROTOCOL command) (range: 21 to 1029).
MAXIMUM RETRANSMITS <i>retry-count</i>	(PSI lines only) Specifies the maximum number of retransmissions of a frame over the specified line (range: 0 to 255).
MAXIMUM WINDOW <i>block-count</i>	(PSI lines only) Specifies the maximum number of frames for which outstanding PSI acknowledgment messages are allowed (range: 1 to 7).

**MULTIPOINT DEAD** *dead-ratio* (DECnet non-DMP/DMV multipoint lines only) Specifies the polling rate for a tributary in the dead multipoint state (range: 1 to 255).

**PRIORITY** *hardware-priority* Specifies the hardware priority of the controller (range: 4 to 7).

**RETRANSMIT TIMER** *milliseconds* (PSI lines only) Specifies the time to elapse before a block is retransmitted. CFE rounds off all values within this range to the nearest hundred milliseconds (range: 1 to 10,049).

**SPEED** *baud-rate* Sets the line baud rate to one of the following:

50	300	2400
75	600	3600
110	1200	4800
134	1800	7200
150	2000	9600

To set a high speed device (that is, over 9600 baud), enter 9600 or the actual baud rate.

**STATE** Sets the line's operational state when the system is loaded.

**CLEARED** The line is not loaded. A DDCMP controller line may not be set to CLEARED if any circuits on it are in the ON state.

**OFF** The line is loaded, but not available for use. A DDCMP controller line may not be set to OFF if any circuits on it are in the ON state.

**ON** The line is loaded and available for normal use except during temporary overrides for service operations. All PSI lines are set to ON. If the line is a DDCMP point-to-point line or a DDCMP tributary, the corresponding circuit is also set to ON.

UNIT CSR                      Specifies the address of the first CSR for a  
*csr-address*                      DECnet device controlled by a KMC-11.

#### NOTE

With the KDP, a unit CSR can be specified for any unit on the controller.

With the KDZ, a unit CSR can be specified only for the first unit set in the associated DZ multiplex group.

VECTOR                      Specifies the vector address of the line  
*vector-address*                      controller.

#### Examples:

```
CFE>DEFINE KNOWN LINES MAXIMUM DATA 256
```

This command establishes the maximum frame size for all known lines as 256 bytes.

```
CFE>DEFINE LINE DZ-3-0 SPEED 3600
```

This command sets the baud rate of line DZ-3-0 at 3600.

**DEFINE LOGGING**

Use the DEFINE LOGGING command to create or modify logging parameters in the permanent database

**Format:**

```

DEFINE { KNOWN LOGGING
        LOGGING CONSOLE
        LOGGING FILE
        LOGGING MONITOR } [EVENTS event-list
                          KNOWN EVENTS
                          STATE {OFF}
                               {ON} ]

```

where

KNOWN LOGGING	Indicates that the specified parameters are to be created or modified for all known logging components.
LOGGING CONSOLE	Indicates that the specified parameters are to be created or modified for the console logging component.
LOGGING FILE	Indicates that the specified parameters are to be created or modified for the file logging component.
LOGGING MONITOR	Indicates that the specified parameters are to be created or modified for the monitor logging component.
EVENTS <i>event-list</i>	Specifies the event class and type(s) to be logged.

KNOWN  
EVENTS

Specifies that all events that DECnet-RSX can generate are to be logged.

STATE

Sets the operational state of the logging component when the system is loaded.

OFF

The logging component is disabled and events for that component are discarded.

ON

The logging component is enabled and events for that component are sent to a name currently defined for that component.

**Example:**

```
CFE>DEFINE LOGGING MONITOR STATE ON
```

This command changes the state of the logging monitor to ON.

CFE

P

**DEFINE MODULE  
X25-ACCESS**

Use the DEFINE MODULE X25-ACCESS command to specify a destination name for a remote DTE in the permanent database. Destinations defined in the permanent database apply to all terminals.

**Format:**

```
DEFINE MODULE X25-ACCESS DESTINATION dest-name NUMBER dte-address
```

where

DESTINATION <i>dest-name</i>	Specifies a destination name for the identified remote DTE address.
NUMBER <i>dte-address</i>	Specifies the remote DTE address to be associated with the specified destination name.

**Example:**

```
CFE>DEFINE MODULE X25-ACCESS DESTINATION PRS44 NUMBER 219746
```

This command establishes remote DTE 219746 as having the name PRS44.

---

**DEFINE MODULE  
X25-PROTOCOL**

Use the DEFINE MODULE X25-PROTOCOL command to create or modify DTE, group, or protocol module parameters in the permanent database. When using this command, you can choose one or no DTE component. You can also choose one or no GROUP component plus any of the other parameters listed, such as CALL TIMER *seconds*.

**Format:**

```

DEFINE MODULE  DTE dte-address  CHANNELS list
X25-PROTOCOL  KNOWN DTES          COUNTER TIMER seconds
                                         LINE line-id
                                         MAXIMUM CIRCUITS count
                                         STATE {OFF}
                                                {ON}

          GROUP group-name  DTE dte-address
          KNOWN GROUPS      NUMBER group-number
                              TYPE BILATERAL

          CALL TIMER seconds
          CLEAR TIMER seconds
          DEFAULT DATA byte-count
          DEFAULT WINDOW block-count
          MAXIMUM DATA byte-count
          MAXIMUM CLEARS count
          MAXIMUM RESETS count
          MAXIMUM RESTARTS count
          MAXIMUM WINDOW count
          RESET TIMER seconds
          RESTART TIMER seconds

```

where

**DTE-related parameters:**

DTE <i>dte-address</i>	Defines parameters for the specified local DTE only. See <i>RSX-11 PSI Network-specific Information</i> for the address format on your network.
KNOWN DTES	Defines parameters for all local DTEs.
CHANNELS <i>list</i>	Specifies a list of logical channel numbers to be used for outgoing calls or to be taken by incoming calls. Use one or more 1- to 4-digit channel numbers in the range of 1 to 4095. Use a hyphen (-) to indicate ranges; use a comma (,) to separate numbers within the list. Channel numbers are used in the order in which they are listed.
COUNTER TIMER <i>seconds</i>	Sets a timer whose expiration causes a DTE counter logging event (range: 1 to 65.535).
LINE <i>line-id</i>	Specifies a particular line for use by the protocol module.
MAXIMUM CIRCUITS <i>count</i>	Specifies the maximum number of circuits the module can use (range: 16 to 512).
STATE	Sets the local DTE's operational state to OFF or ON when the system is loaded.

### Group-related parameters:

<b>GROUP</b> <i>group-name</i>	Defines parameters for the specified group only.
<b>KNOWN GROUPS</b>	Defines parameters for all known groups.
<b>DTE</b> <i>dte-address</i>	Identifies the local DTE associated with the specified group. This parameter must be used with a group number.
<b>NUMBER</b> <i>group-number</i>	Specifies a 2-digit closed user group (CUG) number or a 4-digit bilateral closed user group (BCUG) number (range: 0 to 9999; you can omit leading zeroes). This parameter must be used with a DTE address.
<b>TYPE</b> <b>BILATERAL</b>	Required when the specified group is bilateral (BCUG).

### Protocol-related parameters:

<b>CALL TIMER</b> <i>seconds</i>	Sets a timer that starts when a request is transmitted to set up a virtual circuit and whose expiration clears the request if no response has been received (range: 0 to 255).
<b>CLEAR TIMER</b> <i>seconds</i>	Sets a timer that starts when a request is transmitted to clear a virtual circuit and whose expiration causes a retransmission of the clear request if no response has been received (range: 0 to 255).
<b>DEFAULT DATA</b> <i>byte-count</i>	Specifies the default packet size (in bytes) for all SVCs. This value must not exceed that specified for the <b>MAXIMUM DATA</b> parameter and must be a power of 2 (range: 16 to 1024).

<b>DEFAULT WINDOW</b> <i>block-count</i>	Specifies the default number of data packets that can be sent over an SVC before you have to wait for an acknowledgment (range: 1 to 7). This value must not exceed that specified for MAXIMUM WINDOW.
<b>MAXIMUM DATA</b> <i>byte-count</i>	Specifies the maximum packet size you expect to receive on any SVC. This value must be at least 5 bytes less than the maximum data size specified for PSI lines (see DEFINE LINE) and must be a power of 2 (range: 16 to 1024).
<b>MAXIMUM CLEARS</b> <i>count</i>	Specifies the maximum number of times that the protocol module can try clearing a virtual circuit (range: 0 to 255).
<b>MAXIMUM RESETS</b> <i>count</i>	Specifies the maximum number of times the protocol module can try resetting a virtual circuit (range: 0 to 255).
<b>MAXIMUM RESTARTS</b> <i>count</i>	Specifies the maximum number of times the protocol module can try restarting a DTE (range: 0 to 255).
<b>MAXIMUM WINDOW</b> <i>count</i>	Specifies the maximum number of data packets that can be sent over an SVC before you have to wait for an acknowledgment (range: 1 to 7).
<b>RESET TIMER</b> <i>seconds</i>	Sets a timer that starts when a reset is transmitted and whose expiration causes a retransmission of the reset message if no response has been received (range: 0 to 255).
<b>RESTART TIMER</b> <i>seconds</i>	Sets a timer that starts when a restart is transmitted and whose expiration causes a retransmission of the restart message if no response has been received (range: 0 to 255).

**Examples:**

```
CFE>DEFINE MODULE X25-PROTOCOL KNOWN DTES STATE ON
```

This command sets the state of all local DTEs to ON when the system is loaded.

```
CFE>DEFINE MODULE X25-PROTOCOL MAXIMUM RESTARTS 20
```

This command limits to 20 the number of times the protocol module is to try restarting a virtual circuit.

```

DEFINE MODULE
X25-SERVER
DEFINE MODULE
X29-SERVER

```

Use the DEFINE MODULE X25/X29-SERVER commands to create or modify server module parameters in the permanent database.

#### Restriction:

If you specify CALL MASK or CALL VALUE, you must specify both, and the *hex-value* of each must contain the same number of digits.

#### Format:

```

DEFINE MODULE {X25-SERVER} COUNTER TIMER seconds
               {X29-SERVER} MAXIMUM CIRCUITS count
                               [
                               DESTINATION dest-name CALL MASK hex-value
                               KNOWN DESTINATIONS CALL VALUE hex-value
                               GROUP group-name
                               NUMBER dte-address
                               OBJECT object-id
                               PRIORITY priority
                               SUBADDRESSES range
                               ]

```

where

**COUNTER TIMER** Sets a timer whose expiration causes a  
*seconds* server module counter logging event (range: 0 to 65,535).

**MAXIMUM  
CIRCUITS** Specifies the maximum number of circuits that the  
*count* module (that is, all destinations) can use (range:  
1 to 255).

<b>DESTINATION</b> <i>dest-name</i>	Defines parameters for the specified destination only
<b>KNOWN DESTINATIONS</b>	Defines parameters for all known destinations.
<b>CALL MASK</b> <i>hex-value</i>	Specifies the mask applied to incoming call data before it is tested against the call value (default: no mask). Use a hexadecimal string of 2 to 32 digits (must be an even number of digits); see restriction.
<b>CALL VALUE</b> <i>hex-value</i>	Specifies a string of information used to test incoming call data (default: no test string). Use a hexadecimal string of 2 to 32 digits (must be an even number of digits); see restriction.
<b>GROUP</b> <i>group-name</i>	Specifies the name of a closed user group (CUG) or a bilateral closed user group (BCUG) (default: no group).
<b>NUMBER</b> <i>dte-address</i>	Identifies the remote DTE that originates the call for the specified destination (default: no DTE).
<b>OBJECT</b> <i>object-id</i>	Specifies the task that runs when an incoming call activates it; required when creating a destination.
<b>PRIORITY</b> <i>priority</i>	Specifies the priority of the destination. This is used to select one of a set of destinations for which the incoming call may be valid (range: 0 to 255, where 255 is the highest priority; default: 128).
<b>SUBADDRESSES</b> <i>range</i>	Specifies a range of local DTE subaddresses that identifies the destination for the incoming call.

**Examples:**

```
CFE>DEFINE MODULE X25-SERVER MAXIMUM CIRCUITS 20
```

This command limits the module to 20 circuits.

```
CFE>DEFINE MODULE X29-SERVER DESTINATION TKY029 PRIORITY 5
```

This command sets the priority of destination TKY029 to 5.

**DEFINE NODE**

Use the DEFINE NODE command to create or modify parameters in the permanent database for down-line loading and up-line dumping to the specified node (see the *DECnet-RSX Network Management Concepts and Procedures* manual for a discussion of down-line loading and up-line dumping). If you are not performing service operations on the node, NAME is the only meaningful parameter.

**Restriction:**

You must specify the NAME parameter if a node name has not previously been defined for the specified node.

**Format:**

```

DEFINE NODE node-id  DIAGNOSTIC FILE file
                      DUMP ADDRESS address
                      DUMP COUNT number
                      DUMP FILE file
                      HARDWARE ADDRESS E-address
                      HOST node-id
                      LOAD FILE file
                      NAME node-name
                      SECONDARY [LOADER] file
                      SERVICE  CIRCUIT circuit-id
                      SERVICE DEVICE device-type
                      SERVICE NODE VERSION {PHASE III }
                                           {PHASE IV  }
                      [SERVICE] PASSWORD password
                      TERTIARY  [LOADER] file

```

where

**NODE** *node-id*            Specifies the node for which parameters are to be created or modified. The *node-id* must be a node address if you wish to assign a name to an unnamed node.

**DIAGNOSTIC FILE** *file*        (For Ethernet nodes only) Specifies the file to be read when the node is down-line loaded and requests diagnostics.

**DUMP ADDRESS** Specifies an octal address in memory at which to  
*address* begin an up-line dump of the node; range: 0 to 77777  
(octal).

**DUMP COUNT** Specifies the default number of memory units to  
*blocks* up-line dump from the node (range: 0 to 77777)

**DUMP FILE** *file-id* Specifies the file that is to receive a copy of the system  
at the time of the crash when the node is up-line dumped.

**HARDWARE ADDRESS** Identifies the Ethernet hardware address that  
*E-address* was originally assigned to the DEUNA controller  
for the system on the node. This address is used during  
operations such as down-line load to communicate with  
the system before it has set up its physical address.

**HOST** *node-id* Specifies a host node for all service operations (default:  
executor node).

**LOAD FILE** *file* Specifies a file containing the system software for  
down-line loading to the node.

**NAME** Specifies a node name to be associated with the  
*node-name* node ID.

**SECONDARY LOADER** Specifies a file containing secondary loader  
*file* software for down-line loading to the node.

**SERVICE CIRCUIT** Specifies the circuit to be used for down-line  
*circuit-id* loading and up-line dumping. This circuit is the default  
value for the VIA parameter of the LOAD command.

**SERVICE DEVICE** Specifies the node's line controller for the  
*device-type* service line over which the operation is to take place.  
Possible devices are:

DA	DMC	DPV	DV	KDZ
DL	DMP	DU	DZ	QNA
DLV	DMV	DUP	KDP	UNA

**SERVICE NODE** Specifies the node as a Phase III or Phase IV  
**VERSION** (default) node.

**SERVICE** Specifies the password required to trigger the  
**PASSWORD** bootstrap mechanism on the node.  
*password*

**TERTIARY** Specifies a file containing tertiary loader  
**LOADER** software for down-line loading to the node.  
*file*

**Example:**

```
CFE>DEFINE NODE 205 NAME DALLAS
```

This command identifies node 205 as DALLAS.

---

**DEFINE OBJECT**

Use the DEFINE OBJECT command to create a new object or to modify object parameters in the permanent database.

**Format:**

```

DEFINE OBJECT type-code COPIES { number
                                SINGLE }
NAME object-name
USER { DEFAULT
      LOGIN }
VERIFICATION { INSPECT
               OFF
               ON }

```

where

OBJECT <i>type-code</i>	Specifies the object for which parameters are to be created or modified.
COPIES <i>number</i>	Specifies the maximum number of copies of a task that can be run at once (range: 2 to 64). Use the keyword SINGLE (default) if the task is not multicopy.
NAME <i>object-name</i>	Specifies a name to be associated with the object. Use any valid RSX task name. If the task is multicopy, the name must end with \$\$\$.
USER	Specifies the UIC under which a task is to run when the object's VERIFICATION option is set to ON or INSPECT.
DEFAULT	(Default) The task runs under the default UIC under which it was built or installed.
LOGIN	The task runs under the log-in UIC.

VERIFICATION	Specifies the degree of access to the controlled object.
INSPECT	Allows access to the object while indicating to it whether or not the access control information is valid.
OFF	(Default) Allows access to the object regardless of the validity of the access control information.
ON	Allows access to the object only for inbound connections with valid access control information.

**Example:**

```
CFE>DEFINE OBJECT 23 VERIFICATION ON
```

This command allows access to object 23 only to inbound connections with valid access control information.

---

**DEFINE PROCESS**

Use the DEFINE PROCESS command to modify process parameters in the permanent database.

**Restriction:**

Data space is allocated within the process's addressing space only.

**Format:**

```

DEFINE {PROCESS process-name } MAXIMUM CONTROLLERS count
      {KNOWN PROCESSES } MAXIMUM LINES number
                          STATE {CLEARED }
                               {ON }

```

where

PROCESS <i>process-name</i>	Modifies parameters for the specified process only.
KNOWN PROCESSES	Modifies parameters for all known processes.
MAXIMUM CONTROLLERS <i>count</i>	Specifies the maximum number of hardware controllers (range: 0 to 64) for which the process is to allocate data space (see restriction).
MAXIMUM LINES <i>number</i>	Specifies the maximum number of lines (range: 0 to 64) for which the process is to allocate data space (see restriction).
STATE	Sets the operational state of the process when the system is loaded.
	CLEARED Not available; must be loaded.
	ON Automatically loaded and available.

**Example:**

```
CFE>DEFINE PROCESS DMC MAXIMUM CONTROLLERS 5
```

This command specifies that process DMC allocate data space for five controllers.

---

DEFINE SYSTEM

Use the DEFINE SYSTEM command to modify Communications Executive parameters in the permanent database.

**Format:**

```

DEFINE SYSTEM  LARGE BUFFER SIZE number
                [LOCATION] {FIRSTFIT}
                        {TOPDOWN}
                MAXIMUM CONTROL BUFFERS number
                MAXIMUM LARGE BUFFERS number
                MAXIMUM SMALL BUFFERS number
                MINIMUM RECEIVE BUFFERS number
                POOL BYTE-AREA byte-count
                POOL NAME pool-name
                POOL PARTITION partition-name

```

where

LARGE BUFFER SIZE <i>number</i>	Specifies the size (in bytes) of large buffers (range: for DECnet, 192 to 4096; for PSI only, 128 to 4096).
LOCATION	Specifies the type of dynamic memory allocation for any process whose state is ON in the permanent database.
	FIRSTFIT (Default) Loads the processes at the first available space that is large enough.
	TOPDOWN Loads the processes at the top of the partition.
MAXIMUM CONTROL BUFFERS <i>number</i>	Specifies the maximum number of control buffers available for Communications Executive system use (range: 4 to 200).

MAXIMUM LARGE BUFFERS <i>number</i>	Specifies the maximum number of large buffers available for Communications Executive system use (range: 4 to 200).
MAXIMUM SMALL BUFFERS <i>number</i>	Specifies the maximum number of small buffers available for Communications Executive system use (range: 3 to the maximum number of control buffers).
MINIMUM RECEIVE BUFFERS <i>number</i>	Specifies the minimum number of receive buffers available for Communications Executive system use (range: 1 to <i>n</i> ; <i>n</i> is the maximum number of large buffers minus 1).
POOL BYTE-AREA <i>byte-count</i>	Specifies the number of bytes needed in the network pool (range: <i>n</i> to 64,572, where <i>n</i> is calculated by CFE and includes space for all currently defined remote nodes, objects, and so on; default: <i>m</i> , where <i>m</i> is calculated by CFE and is equal to the value <i>n</i> plus a variable amount of extra free space).
POOL NAME <i>pool-name</i>	Specifies the name of the subpartition to be created. Use a name consisting of 1 to 6 Radix-50 characters.
POOL PARTITION <i>partition-name</i>	Specifies the name of the partition into which the network pool is to be loaded. Use a name consisting of 1 to 6 Radix-50 characters.

**Example:**

```
CFE>DEFINE SYSTEM MAXIMUM SMALL BUFFERS 50
```

This command sets the maximum number of small buffers in the system to 50.

Use the KMX-DUMP command to dump KMX or KMY microcode to disk for analysis by the KMX-Dump analyzer task (KDA). For information on the KDA utility, see Chapter 4.

### NOTE

This command halts all lines on the KMX or KMY device. To bring the device back up after the dump, use the NCP SET LINE command to set the state of all lines first to OFF and then to ON.

#### Format:

```
KMX-DUMP LINE line-id [FILENAME file]
```

where

LINE <i>line-id</i>	Specifies a KMX or KMY line (see note above). Specify <i>line-id</i> in the form KMX- <i>c-u</i> or KMY- <i>u</i> .
FILENAME <i>file</i>	Names the file to hold the binary dump (default: LB:[1,6]PSIKMXMEM.SYS).

#### Example:

```
NCP>KMX-DUMP LINE KMX-0-1 LB:[1,6]KMXMC.DMP
```

This command dumps the microcode of KMX-0-1 to file LB:[1,6]KMXMC.DMP.

CFE

P

---

## LIST CIRCUIT

Use the LIST CIRCUIT command to display circuit information stored in the permanent database.

### Format:

LIST {CIRCUIT *circuit-id*  
KNOWN CIRCUITS}

where

CIRCUIT                    Displays information for the specified circuit  
*circuit-id*                    only.

KNOWN CIRCUITS Displays information for all known circuits.

### Example:

```
CFE>LIST CIRCUIT DMC-0
```

This command causes the system to display static information for circuit DMC-0.

---

**LIST EXECUTOR**

Use the **LIST EXECUTOR** command to display executor node information stored in the permanent database.

**Format:**

**LIST EXECUTOR**

**LIST LINE**

Use the LIST LINE command to display line information stored in the permanent database.

**Format:**
$$\text{LIST } \left\{ \begin{array}{l} \text{LINE } \textit{line-id} \\ \text{KNOWN LINES} \end{array} \right\}$$

where

LINE *line-id*            Displays information for the specified line only.

KNOWN LINES            Displays information for all known lines.

**Example:**

```
CFE>LIST KNOWN LINES
```

This command causes the system to display static information for every line attached to the executor node.

---

**LIST LOGGING**

Use the LIST LOGGING command to display logging information stored in the permanent database.

**Format:**

```
LIST { KNOWN LOGGING
      LOGGING CONSOLE
      LOGGING FILE
      LOGGING MONITOR }
```

where

KNOWN LOGGING	Displays information for all known logging components.
LOGGING CONSOLE	Displays information for the console logging component.
LOGGING FILE	Displays information for the file logging component.
LOGGING MONITOR	Displays information for the monitor logging component.

**Example:**

```
CFE>LIST LOGGING MONITOR EVENTS
```

This command causes the system to display static information for the logging monitor program.

**LIST MODULE  
X25-ACCESS**

Use the LIST MODULE X25-ACCESS command to display remote DTE destination information stored in the permanent database.

**Format:**

```
LIST MODULE X25-ACCESS { DESTINATION dest-name
                        { KNOWN DESTINATIONS } }
```

where

DESTINATION            Displays information for the specified destination  
*dest-name*            only.

KNOWN                 Displays information for all known destinations.  
DESTINATIONS

**Example:**

```
CFE>LIST MODULE X25-ACCESS DESTINATION NYC42
```

This command causes the system to display static information for destination NYC42.

---

**LIST MODULE  
X25-PROTOCOL**

Use the LIST MODULE X25-PROTOCOL command to display protocol module information stored in the permanent database. The information must be qualified by either DTE or group.

**Format:**

```
LIST MODULE X25-PROTOCOL { DTE dte-address
                           KNOWN DTES
                           GROUP group-name
                           KNOWN GROUPS }
```

where

DTE *dte-address*      Displays information for the specified DTE only.

KNOWN DTES            Displays information for all known DTEs.

GROUP *group-name*      Displays information for the specified group only.

KNOWN GROUPS        Displays information for all known groups.

**Example:**

```
CFE>LIST MODULE X25-PROTOCOL GROUP CUG98
```

This command causes the system to display static information for group CUG98.

CFE

P

LIST MODULE  
X25-SERVER  
LIST MODULE  
X29-SERVER

Use the LIST MODULE X25/X29-SERVER commands to display server module information stored in the permanent database. This information may be qualified by destination.

**Format:**

```
LIST MODULE { X25-SERVER } [ DESTINATION dest-name ]  
             { X29-SERVER } [ KNOWN DESTINATIONS ]
```

where

DESTINATION            Displays information for the specified destination  
*dest-name*            only.

KNOWN                    Displays information for all known destinations.  
DESTINATIONS

**Example:**

```
CFE>LIST MODULE X29-SERVER DESTINATION PG49
```

This command causes the system to display static server module information for destination PG49.

---

**LIST NODE**

Use the LIST NODE command to display node information stored in the permanent database.

**Format:**

LIST {*NODE node-id* }  
      {*KNOWN NODES* }

where

*NODE node-id*       Displays information for the specified node only.

*KNOWN NODES*       Displays information for all known nodes.

**Example:**

```
CFE>LIST NODE BOS51
```

This command causes the system to display static information for node BOS51.

**LIST OBJECT**

Use the LIST OBJECT command to display object information stored in the permanent database.

**Format:**

LIST {OBJECT *type-code*  
KNOWN OBJECTS }

where

OBJECT *type-code* Displays information for the specified object only.

KNOWN OBJECTS Displays information for all known objects.

**Example:**

```
CFE>LIST KNOWN OBJECTS
```

This command causes the system to display static information for all objects known to the executor node.

---

**LIST PROCESS**

Use the LIST PROCESS command to display process information stored in the permanent database.

**Format:**

LIST  $\left\{ \begin{array}{l} \text{PROCESS } \textit{process-name} \\ \text{KNOWN PROCESSES} \end{array} \right\}$

where

PROCESS *process-name*      Displays information for the specified process only.

KNOWN PROCESSES      Displays information for all known processes.

**Example:**

CFE>LIST PROCESS KDZ

This command causes the system to display static process information.

CFE

---

**LIST SYSTEM** P

Use the **LIST SYSTEM** command to display system information stored in the permanent database.

**Format:**

**LIST SYSTEM**

---

LOAD NODE

Use the LOAD NODE command to down-line load a specified RSX-11S remote node. For more information about down-line loading system images, see the *DECnet-RSX Network Management Concepts and Procedures* manual.

**Format:**

```
LOAD  NODE node-id      ADDRESS node-address
                               FROM file
                               HOST node-id
                               NAME node-name
                               PHYSICAL ADDRESS E-address
                               SECONDARY [LOADER] file
                               SERVICE DEVICE device-type
                               SERVICE NODE VERSION { PHASE III }
                                                       { PHASE IV }
                               [SERVICE] PASSWORD service-password
                               TERTIARY [LOADER] file
                               VIA circuit-id
```

where

NODE <i>node-id</i>	Specifies the node to be down-line loaded.
ADDRESS <i>node-address</i>	Specifies the address that the node is to use when it comes up.
FROM <i>file</i>	Specifies the file specification of the load file containing the system software to be down-line loaded.
HOST <i>node-id</i>	Specifies the default host that the node is to use when it comes up.
NAME <i>node-name</i>	Specifies the node name that the node is to use when it comes up.
PHYSICAL ADDRESS <i>E-address</i>	(For Ethernet nodes only) Identifies the Ethernet physical address that the node currently uses to identify itself. (Required for Ethernet circuits if the hardware address parameter has not been specified in the volatile database; see SET NODE.)

SECONDARY  
LOADER  
*file* Specifies a file containing secondary loader software for down-line loading to the node.

SERVICE  
DEVICE  
*device-type* Specifies the node's line controller for the service line over which the operation is to take place. Possible devices are:

DA	DMC	DPV	DV	KDZ
DL	DMP	DU	DZ	QNA
DLV	DMV	DUP	KDP	UNA

SERVICE NODE  
VERSION Specifies the node as a Phase III or Phase IV (default) node.

SERVICE  
PASSWORD  
*service-  
password* Specifies the password required to trigger the bootstrap mechanism on the node.

#### NOTE

If you do not want a password echoed to your terminal while you enter it, press the RETURN key after the keyword PASSWORD. NCP prompts for the password and turns off echoing until the next prompt.

TERTIARY  
LOADER *file* Specifies a file containing tertiary loader software for down-line loading to the node.

VIA *circuit-id* Specifies the circuit over which the load is to take place.

**Examples:**

```
NCP>LOAD NODE ATL25
```

This command loads node ATL25. Default load parameters are obtained from the volatile database.

```
NCP>LOAD NODE ATL25 VIA DMC-1
```

This command loads node ATL25 by way of circuit DMC-1.

**LOAD VIA**

Use the LOAD VIA command to down-line load an RSX-11S remote node via the specified circuit. For more information about down-line loading system images, see the *DECnet-RSX Network Management Concepts and Procedures* manual.

**Format:**

```
LOAD VIA circuit-id          ADDRESS node-address
                                FROM file
                                HARDWARE ADDRESS E-address
                                HOST node-id
                                NAME node-name
                                PHYSICAL ADDRESS E-address
                                SECONDARY [LOADER] file
                                SERVICE DEVICE device-type
                                SERVICE NODE VERSION {PHASE III }
                                                       {PHASE IV  }
                                [SERVICE] PASSWORD service-password
                                TERTIARY [LOADER] file
```

where

- VIA *circuit-id*                Specifies a circuit over which the load is to take place.
- ADDRESS *node-address*        Specifies the address the node is to use when it comes up.
- FROM *file*                    Specifies the file specification of the load file containing the system software to be down-line loaded.
- HOST *node-id*                Specifies the default host that the node is to use when it comes up.
- NAME *node-name*               Specifies the node name that the node is to use when it comes up.
- PHYSICAL ADDRESS *E-address*    (For Ethernet nodes only) Identifies the Ethernet physical address that the node currently uses to identify itself. Required for Ethernet circuits if the hardware address parameter has not been specified in the volatile database; see SET NODE.

SECONDARY LOADER <i>file</i>	Specifies the file specification of a secondary load program to be down-line loaded.															
SERVICE DEVICE <i>device-type</i>	Specifies the node's line controller for the service line over which the operation is to take place. Possible devices are:  <table border="0" style="margin-left: 40px;"> <tr> <td>DA</td> <td>DMC</td> <td>DPV</td> <td>DV</td> <td>KDZ</td> </tr> <tr> <td>DL</td> <td>DMP</td> <td>DU</td> <td>DZ</td> <td>QNA</td> </tr> <tr> <td>DLV</td> <td>DMV</td> <td>DUP</td> <td>KDP</td> <td>UNA</td> </tr> </table>	DA	DMC	DPV	DV	KDZ	DL	DMP	DU	DZ	QNA	DLV	DMV	DUP	KDP	UNA
DA	DMC	DPV	DV	KDZ												
DL	DMP	DU	DZ	QNA												
DLV	DMV	DUP	KDP	UNA												
SERVICE NODE VERSION	Specifies the node as a Phase III or Phase IV (default) node.															
SERVICE PASSWORD <i>service- password</i>	Specifies the password required to trigger the bootstrap mechanism on the node.															

#### NOTE

If you do not want a password echoed to your terminal while you enter it, press the RETURN key after the keyword PASSWORD. NCP prompts for the password and turns off echoing until the next prompt.

TERTIARY LOADER <i>file</i>	Specifies the file specification of a third load program to be down-line loaded.
-----------------------------------	--

**Example:**

```
NCP>LOAD VIA DMC-0
```

This command loads the node connected to the executor node via circuit DMC-0.

---

**LOOP CIRCUIT**

Use the LOOP CIRCUIT command to test a DECnet circuit in the network. See the *DECnet-RSX Network Management Concepts and Procedures* manual for more information on loopback testing.

When testing an Ethernet circuit, you can specify a destination node using either its node name or its physical address. You can also use the HFLP parameter to specify a third node to assist with the test. If you do not specify a destination node, the loop request is sent to the multicast address and the first node to respond will complete the loop test.

**Format:**

```

LOOP CIRCUIT circuit-id [
  NODE node-name
  PHYSICAL ADDRESS E-address
  HELP {FULL
        {RECEIVE}
        {TRANSMIT}}
        {(ASSISTANT NODE node-name)
         (ASSISTANT PHYSICAL ADDRESS E-address)}
  COUNT count
  LENGTH length
  WITH {MIXED
        {ONES}
        {ZEROES}}
]

```

**CIRCUIT***circuit-id*

Specifies the circuit to use for the loopback test.

**NODE** *E-node-name*

Specifies the name of an Ethernet node that is to be the destination of a loop test message.

**PHYSICAL ADDRESS***E-address*

Specifies the physical address (not a multicast address) of an Ethernet node that is to be the destination of a loop test message.

**HELP**  
*help-type*

Specifies the degree to which a third node is to assist with an Ethernet loop circuit test.

**FULL**                    The assisting node is to both receive and transmit the test packet (see the example).

**RECEIVE**                The assisting node is only to receive the test packet.

**TRANSMIT**              The assisting node is only to transmit the test packet.

If you specify **HELP**, you must also specify either the **PHYSICAL ADDRESS** and **ASSISTANT PHYSICAL ADDRESS** parameters or, if the addresses are not known, you must specify the **NODE** and **ASSISTANT NODE** parameters. If you specify **ASSISTANT NODE** or **ASSISANT ADDRESS** and do not specify **HELP**, the default is **HELP FULL**.

**ASSISTANT NODE**  
*E-node-name*

Specifies the name of an Ethernet node that is to assist in the loop circuit test.

**ASSISTANT PHYSICAL ADDRESS**  
*E-address*

Specifies the physical address (not a multicast address) of an Ethernet node that is to assist in the loop circuit test.

**COUNT**

*count* Specifies the number of blocks to be sent during loopback testing (range: 1 to 65535; default: 1 block is looped).

**LENGTH**  
*length*

Specifies the length (in bytes) of blocks to be sent during loopback testing. The length must be a decimal integer in the range of 1 to *n*, where *n* must be less than the smaller of either the local looper buffer size or the remote mirror buffer size (default: 40 bytes).

When testing over the Ethernet, the maximum length of the data pattern varies according to the level of assistance:

<b>Level of Assistance</b>	<b>Allowable Length</b>
No assistance	32 - 1486 bytes
Transmit or receive assistance	32 - 1478 bytes
Full assistance	32 - 1470 bytes

**WITH**

Specifies the type of binary information to be sent during testing (default: MIXED -- a combination of ones and zeroes).

**Example:**

```
NCP>LOOP CIRCUIT UNA-0 HELP FULL PHYSICAL ADDRESS AA-00-04-00-F9-04 -<RET>  
NCP>ASSISTANT PHYSICAL ADDRESS AA-00-04-00-04-A9
```

This command, shown in continuation format, tests circuit UNA-0 with the assistance of the node specified in ASSISTANT PHYSICAL ADDRESS. The initiating node sends a test packet to the assisting node. The assisting node processes the packet and passes the packet to the destination node specified in PHYSICAL ADDRESS. The destination node receives the packet and transmits the packet back to the assisting node. The assisting node then returns the packet to the initiating node.

---

**LOOP EXECUTOR/NODE**

Use the LOOP NODE command to test a node in the network. You can include access control information if the node requires it. If you are testing the executor node, you can use the LOOP EXECUTOR command. Either command causes test blocks of data to be transmitted to the specified node. See the *DECnet-RSX Network Management Concepts* manual for more information on loopback testing.

**Format:**

```

LOOP { NODE node-id[acc-con-info] } COUNT count
      { EXECUTOR } LENGTH length
                          WITH { MIXED
                                { ONES
                                  ZEROES }

```

where

NODE <i>node-id</i>	Specifies a node for loopback testing.
<i>acc-con-info</i>	Specifies access control information, if required.
EXECUTOR	Specifies the executor node for loopback testing.
COUNT <i>count</i>	Specifies the number of blocks to be sent during loopback testing (range: 1 to 65535; default: 1 block is looped).
LENGTH <i>length</i>	Specifies the length (in bytes) of blocks to be sent during loopback testing. When testing over the Ethernet, the length must be a decimal integer in the range of 32 to 1484. Otherwise, the length must be a decimal integer in the range of 1 to <i>n</i> , where <i>n</i> must be less than the smaller of either the local looper buffer size or the remote mirror buffer size (default: 40 bytes).
WITH	Specifies the type of binary information to be sent during testing (default: MIXED -- a combination of ones and zeroes).

**Example:**

```
MCP>LOOP NODE NYC COUNT 10
```

This command loops 10 blocks of mixed test messages to remote node NYC. Each block is 40 bytes.

---

LOOP LINE

Use the LOOP LINE command to test a PSI line in the network. This command causes test blocks of data to be transmitted over the specified line. See the *DECnet-RSX Network Management Concepts and Procedures* manual for more information on loopback testing.

**Format:**

```
LOOP LINE line-id      COUNT count
                        LENGTH length
                        WITH { MIXED
                              ONES
                              ZEROS }
```

where

LINE <i>line-id</i>	Specifies the line to use for the loopback test.
COUNT <i>count</i>	Specifies the number of blocks to be sent during loopback testing (range: 1 to 65535; default: 1 block is looped).
LENGTH <i>length</i>	Specifies the length (in bytes) of blocks to be sent during loopback testing. The length must be a decimal integer in the range of 1 to <i>n</i> , where <i>n</i> must be less than the smaller of either the local looper buffer size or the remote mirror buffer size (default: 40 bytes).
WITH	Specifies the type of binary information to be sent during testing (default: MIXED -- a combination of ones and zeroes).

**Example:**

```
NCP>LOOP LINE SDP-0 COUNT 10 LENGTH 100
```

This command loops a 100 byte message over line SDP-0 10 times. The data consists of mixed ones and zeroes.

**PURGE CIRCUIT**

Use the PURGE CIRCUIT command to clear the specified PSI circuit counter timer(s) in the permanent database.

**Format:**

```
PURGE {CIRCUIT circuit-id} COUNTER TIMER  
      {KNOWN CIRCUITS}
```

where

CIRCUIT                    Clears the counter timer for the specified circuit  
*circuit-id*                    only.

KNOWN                    Clears the counter timers for all known circuits.  
CIRCUITS

**Example:**

```
CFE>PURGE KNOWN CIRCUITS COUNTER TIMER
```

This command clears the counter timers for all known PSI circuits.

---

**PURGE LINE**

Use the PURGE LINE command to clear the specified PSI line counter timer(s) in the permanent database.

**Format:**

```
PURGE { LINE line-id } COUNTER TIMER  
      { KNOWN LINES }
```

where

LINE <i>line-id</i>	Clears the counter timer for the specified line only.
KNOWN LINES	Clears the counter timers for all known lines.

**Example:**

```
CFE>PURGE LINE SDP-0 COUNTER TIMER
```

This command clears the counter timer for line SDP-0.

**PURGE LOGGING**

Use the PURGE LOGGING command to remove the specified logging event parameters from the permanent database.

**Format:**

```
PURGE { KNOWN LOGGING
        LOGGING CONSOLE
        LOGGING FILE
        LOGGING MONITOR } { ALL EVENTS
                            EVENTS event-list
                            KNOWN EVENTS }
```

where

KNOWN LOGGING	Removes parameters for all known logging components.
LOGGING CONSOLE	Removes parameters for the console logging component.
LOGGING FILE	Removes parameters for the file logging component.
LOGGING MONITOR	Removes parameters for the monitor logging component.
ALL EVENTS	Removes parameters for all logging event classes and types.
EVENTS <i>event-list</i>	Removes the event class and type(s) specified in <i>event-list</i> for the specified component.
KNOWN EVENTS	Removes all logging events that DECnet-RSX can generate for the specified component.

**Example:**

```
CFE>PURGE KNOWN LOGGING ALL EVENTS
```

This command removes all classes and types of events for all known logging components.

---

**PURGE MODULE  
X25-ACCESS**

Use the PURGE MODULE X25-ACCESS command to remove parameters for the specified destination(s) from the permanent database.

**Format:**

```
PURGE MODULE X25-ACCESS { DESTINATION dest-name } ALL  
                        { KNOWN DESTINATIONS }
```

where

DESTINATION        Removes parameters for the specified destination  
*dest-name*            only.

KNOWN  
DESTINATIONS        Removes parameters for all known destinations.

ALL                (Required) Removes all parameters for the specified  
destination(s).

**Example:**

```
CFE>PURGE MODULE X25-ACCESS DESTINATION DALLAS ALL
```

This command removes destination DALLAS and all associated parameters.

## PURGE MODULE X25-PROTOCOL

Use the PURGE MODULE X25-PROTOCOL command to zero the counter timer(s) for the specified DTE(s) or to remove all parameters for the specified DTE(s) or group(s) from the permanent database.

### Format:

```
PURGE MODULE X25-PROTOCOL { DTE dte-address { ALL  
                           KNOWN DTES      COUNTER TIMER  
                           GROUP group-name ALL  
                           KNOWN GROUPS
```

where

**DTE**  
*dte-address*                      Clears the specified DTE only. Be aware that purging a DTE removes all PVCs and CUGs associated with it and purges them from the permanent database.

**KNOWN DTES**                      Clears all known DTEs.

**GROUP**  
*group-name*                      Removes parameters for the specified group only.

**KNOWN GROUPS**                  Removes parameters for all known groups.

**ALL**                                  Removes all parameters for the specified DTE(s) or group(s).

**COUNTER TIMER**                  Clears the counter timer(s) for the specified DTE(s).

### Example:

```
CFE>PURGE MODULE X25-PROTOCOL GROUP CUG111 ALL
```

This command removes parameters for group CUG111.

---

```

PURGE MODULE
X25-SERVER
PURGE MODULE
X29-SERVER

```

Use the PURGE MODULE X25/X29-SERVER commands to zero the counter timer or to remove parameters for the specified destination(s) from the permanent database.

**Format:**

```

PURGE MODULE {X25-SERVER} {COUNTER TIMER
                {X29-SERVER} {DESTINATION dest-name ALL
                {KNOWN DESTINATIONS ALL}

```

where

COUNTER TIMER	Clears the server module counter timer.
DESTINATION <i>dest-name</i>	Removes parameters for the specified destination only.
KNOWN DESTINATIONS	Removes parameters for all known destinations.
ALL	Required when you specify a destination parameter; removes all parameters for the specified destination(s).

**Example:**

```
CFE>PURGE MODULE X25-SERVER DESTINATION ALBANY ALL
```

This command removes destination ALBANY and all associated parameters.

**PURGE NODE**

Use the PURGE NODE command to remove specified node parameters from the permanent database on the local node.

**Format:**

```
PURGE { NODE node-id
        KNOWN NODES } ALL
        DIAGNOSTIC FILE
        DUMP ADDRESS
        DUMP COUNT
        DUMP FILE
        HARDWARE ADDRESS
        HOST
        LOAD FILE
        SECONDARY [LOADER]
        SERVICE CIRCUIT
        SERVICE DEVICE
        [SERVICE] PASSWORD
        TERTIARY [LOADER]
```

where

NODE <i>node-id</i>	Removes parameters for the specified node only.
KNOWN NODES	Removes parameters for all known nodes.
ALL	Removes all parameters for the specified node(s). If you specify ALL, you cannot include any other parameters.
DIAGNOSTIC FILE	Removes the identification of the down-line load diagnostics file.

DUMP ADDRESS	Removes the node's up-line dump address.
DUMP COUNT	Removes the up-line dump count.
DUMP FILE	Removes the up-line dump file identification.
HARDWARE ADDRESS	Removes the Ethernet address of the system hardware.
HOST	Removes the host node identification.
LOAD FILE	Removes the down-line load file identification.
SECONDARY LOADER	Removes the parameter associated with the file containing secondary loader software for down-line loading to the node.
SERVICE CIRCUIT	Removes the circuit parameter associated with the node for down-line load operations.
SERVICE DEVICE	Removes the service device type.
SERVICE PASSWORD	Removes the password parameter required to trigger the bootstrap mechanism.
TERTIARY LOADER	Removes the parameter associated with the file containing tertiary loader software for down-line loading to the node.

**Example:**

```
CFE>PURGE NODE BOS ALL
```

This command removes node BOS from the database.

**PURGE OBJECT**

Use the PURGE OBJECT command to remove all parameters for the specified object(s) from the permanent database.

**Format:**

PURGE { OBJECT *object-type* }  
          { KNOWN OBJECTS }

where

OBJECT                   Removes parameters for the specified object only.  
*object-type*

KNOWN                   Removes parameters for all known objects.  
OBJECTS

**Example:**

CFE>PURGE OBJECT 200

This command removes all parameters for object 200.

---

**SET ALIAS**

Use the NCP SET ALIAS command to specify an alias name for a node in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
SET ALIAS alias-name DESTINATION dest-node [[SCOPE] scope]
```

where

ALIAS <i>alias-name</i>	Specifies the alias name to be assigned to the destination node.
DESTINATION <i>dest-node</i>	Specifies the destination node and any access control parameters associated with this alias for that node. See Section 1.5.2 for access control information format.
SCOPE <i>scope</i>	See definition in Section 1.5.2.

**Example:**

```
NCP>SET ALIAS PETSKY DESTINATION MAINE/DAVIES/LESLIE -<RET>  
NCP>SCOPE TERMINAL TT22:
```

This command (shown in continuation format) sets alias PETSKY to terminal TT22: on node MAINE. DAVIES is the account name, and LESLIE is the password.

**SET CIRCUIT**

Use the NCP SET CIRCUIT command to create or modify specified circuit parameters in the volatile database. Use the VNP command to do the same with the system image file.

**Restriction:**

The VNP SET CIRCUIT command is invalid for PSI circuits.

**Format:**

```
SET {CIRCUIT circuit-id} COST cost
    {KNOWN CIRCUITS} COUNTER TIMER seconds
                    HELLO TIMER seconds
                    MULTIPOINT ACTIVE active-ratio
                    OWNER {DLX}
                          {XPT}
                    SERVICE {DISABLE}
                             {ENABLE}
                    STATE {OFF
                           ON
                           SERVICE}
                    TRIBUTARY trib-address
```

where

CIRCUIT <i>circuit-id</i>	Sets parameters for the specified circuit only.
KNOWN CIRCUITS	Sets parameters for all known circuits.
COST <i>cost</i>	(DECnet circuits only) Specifies the routing cost of the circuit. Messages will travel between nodes along the path with the smallest total cost (range: 1 to 25).
COUNTER TIMER <i>seconds</i>	(PSI circuits only; invalid for VNP) Sets a timer whose expiration causes the circuit counters to be logged and zeroed (range: 1 to 65,535).

HELLO TIMER <i>seconds</i>	(DECnet circuits only) Specifies the frequency of routing hello messages sent to adjacent nodes over the circuit (range: 1 to 8191).
MULTIPOINT ACTIVE <i>active-ratio</i>	(DECnet non-DMP/DMV multipoint control circuits only) Specifies the rate at which a tributary in the active multipoint state is polled (range: 1 to 255).
OWNER	(DECnet circuits only) Specifies the circuit owner.
	DLX            DLX (See the <i>DECnet-RSX Programmer's Reference Manual</i> .)
	XPT            DECnet
SERVICE	(DECnet circuits only) Specifies whether the circuit is enabled or disabled for down-line loading and loopback testing.
STATE	Sets the circuit's operational state. For circuits owned by the executor, the state of the corresponding lines must be ON.
	OFF            Not available.
	ON             Available for normal use.
	SERVICE      Available for service operations only.
TRIBUTARY <i>trib-address</i>	(DECnet multipoint circuits only) Specifies the data link physical tributary address of the multipoint circuit (range: 1 to 255).

**Example:**

```
NCP>SET CIRCUIT KDZ-1-3 STATE OFF
```

This command sets the state of circuit KDZ-1-3 to OFF.

**SET EXECUTOR**

Use the NCP SET EXECUTOR command to create or modify executor node parameters in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```

SET EXECUTOR HOST node-id
                INACTIVITY TIMER seconds
                INCOMING TIMER seconds
                OUTGOING TIMER seconds
                RECEIVE PASSWORD password
                RETRANSMIT FACTOR number
                ROUTING TIMER seconds
                SEGMENT BUFFER SIZE number
                STATE { OFF
                     ON [ FIXED
                        UNFIXED ]
                     SHUT }
                SUBADDRESSES range
                TRANSMIT PASSWORD password
                VERIFICATION [ STATE ] { OFF
                                       ON }

```

where

HOST <i>node-id</i>	Specifies the host node identification for down-line load, trigger, and up-line dump.
INACTIVITY TIMER <i>seconds</i>	Specifies the maximum time the executor will allow a link to remain idle (no user data traffic) before it checks to see if the circuit still works (range: 1 to 255).
INCOMING TIMER <i>seconds</i>	Specifies the maximum time a process has to answer an incoming connect request. If the process does not answer the connect request within this time interval, the node will reject the connect request on behalf of the process (range: 1 to 255).

OUTGOING TIMER <i>seconds</i>	Specifies the maximum time the executor node will wait for a pending connect request to be answered at a destination node. If the request is not answered in this time interval, the source process receives an error indication (range: 1 to 255).
RECEIVE PASSWORD <i>password</i>	Specifies a 1- to 8-character ASCII password that the executor expects to receive during initialization with an adjacent Phase III node.
RETRANSMIT FACTOR <i>number</i>	Specifies the number of times the executor will restart the retransmission timer before the logical link is disconnected (range: 1 to 255).
ROUTING TIMER <i>seconds</i>	Sets a timer whose expiration forces a routing update on non-Ethernet circuits (range: 1 to 65,535).
SEGMENT BUFFER SIZE <i>number</i>	Specifies the maximum size of transmit buffers (in bytes), thereby controlling the size of the NSP message segment to be sent. This value is the maximum size message that the End Communications layer can transmit; it does not include routing or data link overhead (range: 1 to 4096; default: 18 bytes less than the large buffer size specified in the DEFINE SYSTEM command). The segment buffer size can never be greater than the large buffer size minus 18 bytes.
STATE	Sets the operational state of the executor node. This parameter cannot be specified if you use the TELL prefix.
	OFF      Allows no new logical links; terminates existing links; forces the release of all mailboxes as quickly as possible.

ON (Invalid for VNP on RSX-11M/M-PLUS; see *DECnet-RSX Network Management Concepts and Procedures*) Allows normal logical link activity. When using VNP to set the executor state to ON for DECnet-11S systems, you can specify one of the following task states for NTINIT after it has initialized the node:

FIXED (Default) NTINIT remains in memory.

UNFIXED NTINIT is removed from memory.

SHUT (Invalid for VNP) Allows no new logical links; does not destroy existing links; goes to the OFF state when all logical links terminate and all mailboxes are released.

SUBADDRESSES (Invalid for VNP) Specifies a range of local DTE subaddresses that are valid on any DLM circuit for incoming calls to the executor node.  
*range*

TRANSMIT PASSWORD Specifies a 1- to 8-character ASCII password that the executor sends to the adjacent Phase III node during a node initialization sequence.  
*password*

<b>VERIFICATION STATE</b>	Sets the logical link access control verification state for the executor node.
<b>OFF</b>	The executor node does not verify access control on any incoming connect requests.
<b>ON</b>	The executor node verifies access control on all incoming connect requests according to the options set for each object (see the discussion of access control in the <i>DECnet-RSX Network Management Concepts and Procedures</i> manual).

**Example:**

```
NCP>SET EXECUTOR HOST 12
```

This command sets node 12 as the host node for down-line load, trigger, and up-line dump operations.

---

**SET EXECUTOR NODE**

Use the SET EXECUTOR NODE command to set a default executor for all NCP commands. This causes subsequent remotely executable NCP commands to be executed at the specified destination. The *DECnet-RSX Network Management Reference Card* lists all NCP commands and flags those that are not remotely executable.

**Restriction:**

You cannot use the TELL prefix with this command (that is, this command is not remotely executable).

**Format:**

```
SET EXECUTOR NODE node-id[acc-con-info]
```

where

<i>NODE node-id</i>	Specifies the node by address, alias, or name, that is to serve as the executor for subsequent NCP commands.
<i>acc-con-info</i>	Specifies access control information (if required by the node).

**Example:**

```
NCP>SET EXECUTOR NODE ATL25/NELSON/PAT
```

This command sets node ATL25 (user NELSON, password PAT) to executor status. Future commands will be sent to ATL25 for execution.

---

SET KNOWN LINES

Use the NCP SET KNOWN LINES command either to load all known lines and create their associated parameters in the volatile database or to create or modify parameters in the database for all known lines that are already loaded. Use the VNP command to do the same with the system image file.

**Restriction:**

Except for STATE, the options allowed when you are loading lines are different from those used to create or modify parameters for lines that are already loaded. Any lines that you are loading must be in the CLEARED state. Lines that are already loaded must be in the ON or OFF states (that is, not CLEARED).

**Format:**

SET KNOWN LINES

STATE	{	OFF	}
		ON	}

Loading options:

ALL
DEAD TIMER <i>milliseconds</i>
DELAY TIMER <i>milliseconds</i>
DUPLEX {
FULL
HALF
}
[LOCATION] {
FIRSTFIT
TOPDOWN
}

Loaded options:

COUNTER TIMER <i>seconds</i>
OWNER {
DLX
PLI
}

where

STATE (Invalid for VNP) Sets the operational state for all PSI lines to either OFF or ON.

## Loading options:

ALL	Loads the lines with all of the default parameters specified in the permanent database. If you specify ALL, you cannot include any other parameters.
DEAD TIMER <i>milliseconds</i>	(For DECnet DMP/DMV multipoint control lines only) Specifies the polling interval for a dead tributary (range: 1 to 65,535).
DELAY TIMER <i>milliseconds</i>	(For DECnet DMP/DMV multipoint control lines only) Specifies the minimum time to delay between polls in order to limit the effect of a fast control station on a slow tributary (range: 1 to 65,535).
DUPLEX	(DECnet DDCMP lines only) Sets the hardware transmission mode to full duplex or half duplex.
LOCATION	(Invalid for VNP) Specifies the type of dynamic memory allocation.
FIRSTFIT	(Default) Loads the driver processes at the first available space that is large enough in the partition.
TOPDOWN	Loads the driver processes at the top of the partition.

**Loaded options:**

COUNTER	(PSI lines only) Sets a timer whose expiration
TIMER	causes the line counters to be logged and zeroed
<i>seconds</i>	(range: 0 to 65.535).
OWNER	Specifies line ownership.
DLX	DLX (See the <i>DECnet-RSX Programmer's Reference Manual</i> .)
PLI	PSI (Invalid for VNP)

**Example:**

```
NCP>SET KNOWN LINES DUPLEX FULL
```

This command sets all lines known to the system to full duplex.

**SET LINE**

Use the NCP SET LINE command either to load the specified line and create its associated parameters in the volatile database or to create or modify parameters in the database for a line that is already loaded. Use the VNP command to do the same with the system image file.

**Restriction:**

Except for STATE, the options allowed when you are loading a line are different from those used to create or modify parameters for a line that is already loaded. Any line that you are loading must be in the CLEARED state. A line that is already loaded must be in the ON or OFF state (that is, not CLEARED).

**Format:**

SET LINE *line-id*

STATE { OFF }  
          { ON }

**Loading options:**

ALL  
[CONTROLLER] CSR *csr-address*  
DEAD TIMER *milliseconds*  
DELAY TIMER *milliseconds*  
DUPLEX { FULL }  
          { HALF }  
[LOCATION] { FIRSTFIT }  
           { TOPDOWN }  
MULTIPOINT DEAD *dead-ratio*  
PRIORITY *hardware-priority*  
UNIT CSR *csr-address*  
VECTOR *vector-address*

**Loaded options:**

CONTROLLER { LOOPBACK }  
              { NORMAL }  
~~COUNTER TIMER *seconds*~~  
OWNER { DLX }  
          { PLI }

where

LINE *line-id* Specifies the line to be loaded and for which parameters are to be created or modified.

STATE (Invalid for VNP) Sets a PSI line's operational state to OFF or ON.

**Loading options:**

ALL Loads the line with all of the default parameters specified in the permanent database. If you specify ALL, you cannot include any other parameters.

CONTROLLER Specifies the address of the first CSR for the  
CSR line controller.  
*csr-address*

DEAD TIMER (For DECnet DMP/DMV multipoint control lines only)  
*milliseconds* Specifies the polling interval for a dead tributary (range: 1 to 65,535).

DELAY TIMER (For DECnet DMP/DMV multipoint control lines only)  
*milliseconds* Specifies the minimum time to delay between polls in order to limit the effect of a fast control station on a slow tributary (range: 1 to 65,535).

DUPLEX (DECnet DDCMP lines only) Sets the hardware transmission mode to full duplex or half duplex.

LOCATION (Invalid for VNP) Specifies the type of dynamic memory allocation.

FIRSTFIT (Default) Loads the driver process at the first available space that is large enough in the partition.

TOPDOWN Loads the driver process at the top of the partition.

<b>MULTIPOINT DEAD</b> <i>dead-ratio</i>	(DECnet non-DMP/DMV multipoint control lines only) Specifies the polling rate for a tributary in the dead multipoint state (range: 1 to 255).
<b>PRIORITY</b> <i>hardware-priority</i>	Specifies the hardware priority of the controller (range: 4 to 7).
<b>UNIT CSR</b> <i>csr-address</i>	Specifies the address of the first CSR for a DECnet device controlled by a KMC-11.

### NOTE

With the KDP, a unit CSR can be specified for any unit on the controller.

With the KDZ, a unit CSR can be specified only for the first unit set in the associated DZ multiplex group.

<b>VECTOR</b> <i>vector-address</i>	Specifies the vector address of the line controller. You must use the vector address selected on the device or 0 if the address is unknown (range: 0 to 774 octal).
--	---

### Loaded options:

**CONTROLLER** (For DMC/DMR, DMP, DHU, DHV, QNA, and UNA lines; invalid for VNP) Specifies the controller mode for the line. The line's circuit must be in the OFF state

**LOOPBACK** Internal device loopback mode

**NORMAL** Normal operating mode

**COUNTER TIMER** (PSI lines only, invalid for VNP) Sets a timer whose expiration causes the line counters to be logged and zeroed (range: 0 to 65,535)  
*seconds*

**OWNER** Specifies line ownership.

**DLX** DLX (See the *DECnet-RSX Programmer's Reference Manual*.)

**PLI** PSI (Invalid for VNP)

**Example:**

```
NCP>SET LINE SDP-0 OWNER PLI
```

This command sets the owner of line SDP-0 as PLI, which assigns the line for PSI operations only.

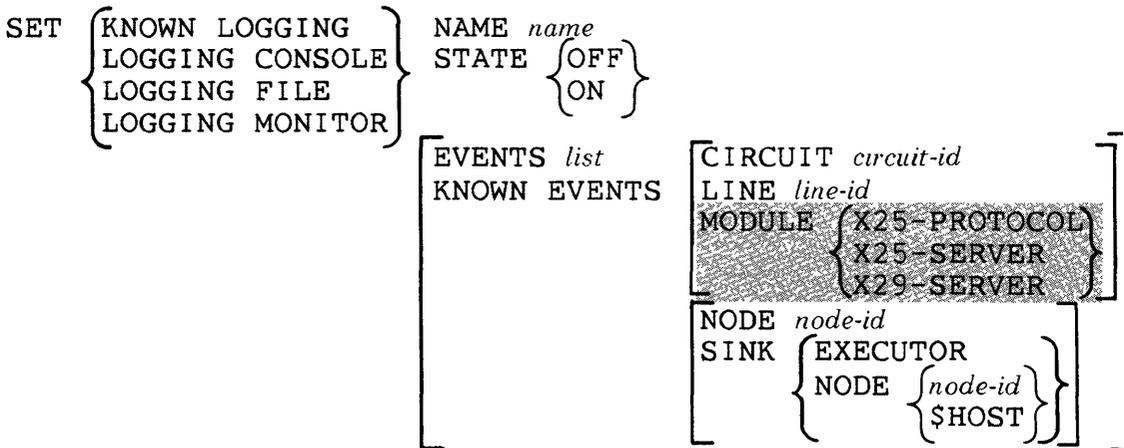
**SET LOGGING**

Use the NCP SET LOGGING command to create or modify logging component parameters in the volatile database. Use the VNP command to do the same with the system image file.

**Restriction:**

Whenever a circuit, line, module, node, or sink is specified in a SET LOGGING command, an event list or KNOWN EVENTS parameter must also be included.

**Format:**



where

**KNOWN LOGGING** Indicates that the specified parameters are to be created or modified for all known logging components.

**LOGGING CONSOLE** Indicates that the specified parameters are to be created or modified for the console logging component.

**LOGGING FILE** Indicates that the specified parameters are to be created or modified for the file logging component.

**LOGGING MONITOR** Indicates that the specified parameters are to be created or modified for the monitor logging component.

NAME <i>name</i>	Specifies the name of the console (default: CO0:), file (default: LB:[1,6]EVENTLOG.SYS), or monitor program (default: MON...) to which events are to be logged.
STATE	Sets the operational state of the logging component on the executor node. When the state is OFF, events are discarded.
EVENTS <i>event-list</i>	Specifies the event class and type(s) to be logged.
KNOWN EVENTS	Specifies that all events that DECnet-RSX can generate are to be logged.
CIRCUIT <i>circuit-id</i>	(Invalid for VNP) Logs the specified event(s) occurring on the specified circuit (see restriction).
LINE <i>line-id</i>	Logs the specified event(s) occurring on the specified line (see restriction).
MODULE	(Invalid for VNP) Logs the specified event(s) occurring on the specified module (see restriction).
NODE <i>node-id</i>	Logs the specified event(s) occurring on the specified node (see restriction).
SINK	Identifies the node where the specified event(s) are to be logged (see restriction).
	EXECUTOR            (Default) Executor node
	NODE <i>node-id</i> The specified remote node
	NODE \$HOST        Host node

**Examples:**

```
NCP>SET KNOWN LOGGING KNOWN EVENTS CIRCUIT DMC-0 SINK NODE BOSTON
```

This command dictates that known events on circuit DMC-0 be sent to node BOSTON.

```
NCP>SET LOGGING CONSOLE EVENT 2.1 SINK NODE ATL
```

This command dictates that any occurrence of event 2.1 be logged on the console at node ATL.

---

**SET MODULE  
X25-ACCESS**

Use the SET MODULE X25-ACCESS command to specify a destination name for a remote DTE in the volatile database.

**Format:**

```
SET MODULE X25-ACCESS DESTINATION dest-name NUMBER dte-address [[SCOPE]scope]
```

where

**DESTINATION**            Specifies a destination name for the identified  
*dest-name*                remote DTE address.

**NUMBER**                 Specifies the remote DTE address to be  
*dte-address*             associated with the specified destination name.

**SCOPE** *scope*            See definition in Section 1.5.2.

**Example:**

```
NCP>SET MODULE X25-ACCESS DESTINATION RDG724 -<RET>  
NCP>NUMBER 41106700200 SCOPE TERMINAL TT51:
```

This command (shown in continuation format) sets a destination name of RDG724 for DTE address 41106700200 at terminal TT51.

## SET MODULE X25-PROTOCOL

Use the SET MODULE X25-PROTOCOL command to create or modify DTE or group parameters in the volatile database.

### Format:

```

SET MODULE X25-PROTOCOL { DTE dte-address COUNTER TIMER seconds
                          KNOWN DTES STATE { OFF
                                             ON
                                             SHUT }
                          GROUP group-name DTE dte-address
                          KNOWN GROUPS NUMBER group-number
                          TYPE BILATERAL }

```

where

### DTE-related parameters:

DTE <i>dte-address</i>	Sets parameters for the specified local DTE only. See <i>RSX-11 PSI Network-specific Information</i> for the address format on your network.
KNOWN DTES	Sets parameters for all local DTEs.
COUNTER TIMER <i>seconds</i>	Sets a timer whose expiration causes the DTE counters to be logged and zeroed (range: 1 to 65,535).
STATE	Sets the local DTE's operational state.
OFF	Not available.
ON	Available for normal use.
SHUT	Not available for new virtual circuit connections; turns off when the last virtual circuit disconnects.

### Group-related parameters:

GROUP <i>group-name</i>	Sets parameters for the specified group only.
KNOWN GROUPS	Sets parameters for all known groups.
DTE <i>dte-address</i>	Specifies the local DTE associated with the specified group. This parameter must be used with a group number.
NUMBER <i>group-number</i>	Specifies a 2-digit closed user group (CUG) number or a 4-digit bilateral closed user group (BCUG) number (range: 0 to 9999; you can omit leading zeroes). This parameter must be used with a DTE address.
TYPE BILATERAL	Required when the specified group is bilateral (BCUG).

### Example:

```
NCP>SET MODULE X25-PROTOCOL GROUP CUG247 NUMBER 95 -<RET>  
NCP>DTE 16175554236
```

This command (shown in continuation format) assigns group number 95 to closed user group CUG247 on DTE 16175554236.

---

**SET MODULE  
X25-SERVER**


---

**SET MODULE  
X29-SERVER**

Use the NCP SET MODULE X25/X29-SERVER commands to create or modify server module parameters in the volatile database. Use the VNP SET MODULE X25-SERVER command to change the server module state in the system image file. (There is no VNP SET MODULE X29-SERVER command.)

**Restrictions:**

- The STATE parameter is valid only for SET MODULE X25-SERVER commands. It is the only valid parameter for VNP.
- You cannot modify parameters for existing destinations. You can only create new destinations. Existing destinations can be cleared, then reset with new parameters. If you specify a destination name, you must also specify the object.
- If you specify CALL MASK or CALL VALUE, you must specify both, and the *hex-value* of each must contain the same number of digits.

**Format:**

```

SET MODULE {X25-SERVER}
           {X29-SERVER}
           COUNTER TIMER seconds
           STATE {OFF
                 ON
                 SHUT}
           DESTINATION dest-name OBJECT object-id
                                (CALL MASK hex-value)
                                (CALL VALUE hex-value)
                                GROUP group-name
                                NUMBER dte-address
                                PRIORITY priority
                                SUBADDRESSES range

```

where

**COUNTER TIMER** *seconds* Sets a timer whose expiration causes the server module counters to be logged and zeroed (range: 0 to 65.535).

**STATE** (Valid only for X25-SERVER commands) Sets the module's operational state.

	OFF	Not available.
	ON	Available for normal use.
	SHUT	(Invalid for VNP) Not available for new virtual circuit connections; turns off when the last virtual circuit disconnects.
DESTINATION		Specifies the destination for which server parameters are to be created.
<i>dest-name</i>		
OBJECT		Specifies the task that runs when an incoming call activates it. Use either an object name or an object type code to identify the task. See Section 1.5.2.
<i>object-id</i>		
CALL MASK		Specifies the mask applied to incoming call data before it is tested against the call value (default: no mask). Use a hexadecimal string of 2 to 32 digits (must be an even number of digits); see restrictions.
<i>hex-value</i>		
CALL VALUE		Specifies a string of information used to test incoming call data (default: no test string). Use a hexadecimal string of 2 to 32 digits (must be an even number of digits); see restrictions.
<i>hex-value</i>		
GROUP		Specifies the name of a closed user group (CUG) or a bilateral closed user group (BCUG) (default: no group).
<i>group-name</i>		
NUMBER		Specifies the remote DTE that originates the call for the specified destination (default: no DTE).
<i>dte-address</i>		
PRIORITY		Specifies the priority of the destination. This is used to select one of a set of destinations for which the incoming call may be valid (range: 0 to 255, where 255 is the highest priority; default: 128).
<i>priority</i>		
SUBADDRESSES		Specifies a range of local DTE subaddresses that identifies the destination for the incoming call.
<i>range</i>		

**Examples:**

```
NCP>SET MODULE X25-SERVER STATE SHUT
```

This command makes the module unavailable for new virtual circuit connections and turns it off when the last virtual circuit disconnects.

```
NCP>SET MODULE X29-SERVER COUNTER TIMER 600
```

This command sets the module counter timer to 600.

---

**SET NODE**

Use the NCP SET NODE command to create or modify parameters in the volatile database for down-line loading and up-line dumping to the specified node. If you are not performing service operations on the node, NAME is the only meaningful parameter. Use the VNP command to associate a node name with a node address in the system image file. For more information on down-line loading and up-line dumping, see the *DECnet-RSX Network Management Concepts and Procedures* manual.

**Restriction:**

NAME is the only valid parameter for VNP.

**Format:**

```
SET NODE node-id      DIAGNOSTIC FILE file
                        ADDRESS node-address
                        DUMP ADDRESS address
                        DUMP COUNT number
                        DUMP FILE file
                        HARDWARE ADDRESS E-address
                        HOST node-id
                        LOAD FILE file
                        NAME node-name
                        SECONDARY [LOADER] file
                        SERVICE CIRCUIT circuit-id
                        SERVICE DEVICE device-type
                        SERVICE NODE VERSION {PHASE III}
                                              {PHASE IV}
                        [SERVICE] PASSWORD service-password
                        TERTIARY [LOADER] file
```

where

**NODE** *node-id* Specifies the node for which parameters are to be created or modified.

**ADDRESS** Specifies a node address to be associated with the name given in *node-id*.

DIAGNOSTIC FILE <i>file</i>	(For Ethernet nodes only) Specifies the file to be read when the node is down-line loaded and requests diagnostics.
DUMP ADDRESS <i>address</i>	Specifies an octal address in memory at which to begin an up-line dump of the node (range: 0 to 77777 (octal)).
DUMP COUNT <i>number</i>	Specifies the default number of blocks to up-line dump from the node (range: 0 to 65535 decimal).
DUMP FILE <i>file-id</i>	Specifies the file that is to receive a copy of the system when the node is up-line dumped.
HARDWARE ADDRESS <i>E-address</i>	Identifies the Ethernet hardware address that was originally assigned to the DEUNA controller for the system on the node. This address is used during operations such as down-line load to communicate with the system before it has set up its physical address.
HOST <i>node-id</i>	Specifies (for the executor node) the node from which the executor receives its services, or represents (for the adjacent node) the host identification received by the node when it is down-line loaded (default: executor node).
LOAD FILE <i>file</i>	Specifies a file containing the system software for down-line loading to the node.
NAME <i>node-name</i>	Specifies a node name to be associated with the <i>node-id</i> .
SECONDARY LOADER <i>file</i>	Specifies a file containing secondary loader software for down-line loading to the node.

SERVICE CIRCUIT <i>circuit-id</i>	Specifies the circuit to be used for down-line loading and up-line dumping. This circuit is the default value for the VIA parameter of the LOAD command.
SERVICE DEVICE <i>device-type</i>	Specifies the line controller for the service line over which the operation is to take place. Possible devices are:  DA    DMC    DPV    DV    KDZ DL    DMP    DU    DZ    QNA DLV   DMV    DUP    KDP   UNA
SERVICE NODE VERSION	Specifies the node as a Phase III or Phase IV (default) node.
SERVICE PASSWORD <i>service- password</i>	Specifies the password required to trigger the bootstrap mechanism on the node.
TERTIARY LOADER <i>file</i>	Specifies a file containing tertiary loader software for down-line loading to the node.

**Example:**

```
NCP>SET NODE 12 NAME BLAKE
```

This command assigns the name BLAKE to node 12.

---

**SET NODE CIRCUIT**

Use the NCP SET NODE CIRCUIT command to associate a DECnet circuit with a loop node name in the volatile database. Use the VNP command to do the same with the system image file.

**Restriction:**

The *circuit-id* used in this command cannot specify an Ethernet circuit; loop nodes are not supported for Ethernet circuits. For more details about loop nodes, see the *DECnet-RSX Network Management Concepts and Procedures* manual.

**Format:**

```
SET NODE node-name CIRCUIT circuit-id
```

where

NODE <i>node-name</i>	Specifies the loop node with which the circuit is to be associated.
CIRCUIT <i>circuit-id</i>	Specifies a DECnet circuit to be used for all traffic to the specified loop node.

**Example:**

```
NCP>SET NODE LASVGS CIRCUIT DMP-0.2
```

This command sets node LASVGS as the loop node name for circuit DMP-0.2.

---

SET OBJECT

Use the NCP SET OBJECT command to create or modify object parameters in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```

SET OBJECT type-code      COPIES { number
                               SINGLE }
                               NAME object-name
                               USER { DEFAULT
                                       LOGIN }
                               VERIFICATION { INSPECT
                                              OFF
                                              ON }

```

where

OBJECT <i>type-code</i>	Specifies the object for which parameters are to be created or modified.
COPIES <i>number</i>	Specifies the maximum number of copies of a task that can be run at once (range: 2 to 64). Use the keyword SINGLE (default) if the task is not multicopy.
NAME <i>object-name</i>	Specifies a name to be associated with the object. Use any valid RSX task name. If the task is multicopy, the name must end with \$\$\$.
USER	Specifies the UIC under which a task is to run when the object's VERIFICATION option is set to ON or INSPECT.
	DEFAULT (Default) The task runs under the default UIC under which it was built or installed.
	LOGIN The task runs under the log-in UIC.
VERIFICATION	Specifies the degree of access to the controlled object.

INSPECT	Allows access to the object while indicating to it whether or not the access control information is valid.
OFF	(Default) Allows access to the object regardless of the validity of the access control information.
ON	Allows access to the object only for inbound connections with valid access control information.

**Example:**

```
NCP>SET OBJECT 19 USER LOGIN VERIFICATION ON
```

This command specifies that object 19 is to run under the log-in UIC and that access is to be allowed only with valid access control information.

---

**SET PROCESS**

Use the NCP SET PROCESS command to load processes and to modify process parameters in the volatile database. With NCP, a network process remains in memory until you clear the system or the process (see CLEAR SYSTEM and CLEAR PROCESS) or until you reboot the system. Use the VNP SET PROCESS command to load processes and to modify process parameters in the system image file.

**Restrictions:**

- If you reload NW after clearing it, you must also issue the SET MODULE X25-PROTOCOL command (first with STATE OFF and then with STATE ON) to update the process pointers to the database.
- Data space is allocated within the process's addressing space only.

**Format:**

```
SET PROCESS process-name ALL
                [LOCATION] {FIRSTFIT}
                        {TOPDOWN }
                MAXIMUM CONTROLLERS count
                MAXIMUM LINES count
                PARTITION partition-name
```

where

PROCESS <i>process-name</i>	Specifies the process for which parameters are to be modified.
ALL	Loads the process with all process parameter defaults specified in the permanent database. If you specify ALL, you cannot include any other parameters.

LOCATION	(Invalid for VNP) Specifies the type of dynamic memory allocation for the driver process.
FIRSTFIT	(Default) Loads the process at the first available space that is large enough.
TOPDOWN	Loads the process at the top of the partition.
MAXIMUM CONTROLLERS <i>count</i>	Specifies the maximum number of hardware controllers (range: 0 to 64) for which the process is to allocate data space (see restrictions).
MAXIMUM LINES <i>count</i>	Specifies the maximum number of lines (range: 0 to 64) for which the process is to allocate data space (see restrictions).
PARTITION <i>partition-name</i>	Specifies the partition name. Use a name consisting of 1 to 6 Radix-50 characters.

**Example:**

```
NCP>SET PROCESS DPV MAXIMUM LINES 2
```

This command limits process DPV to the control of two lines.

---

**SET SYSTEM**

Use the NCP or VNP SET SYSTEM command to load the Communications Executive and the volatile database into the system. The volatile database is loaded from the permanent database, which includes all the network process, line, circuit, module, and node defaults that have been set or defined.

**Restriction:**

On RSX-11M/M-PLUS systems, you must have installed NTINIT using VMR before you can execute a VNP SET SYSTEM command.

**Format:**

```
SET SYSTEM [TOP]
```

where

TOP (Invalid for VNP) Specifies that the system be loaded at the top of memory.

**SET TRACE**

Use the SET TRACE command to start tracing the specified PSI line(s). For more information on the trace interpreter task, see Chapter 9. Note that only one trace file can be active at a time. To turn off tracing, use the CLEAR TRACE command.

**Format:**

```
SET TRACE { LINE line-id } BUFFER SIZE block-count
           { ACTIVE LINES } FILE file
```

where

- |                                   |  |
|-----------------------------------|--|
| LINE <i>line-id</i>               | Sets tracing for the specified line only.  |
| ACTIVE LINES                      | Sets tracing for all active lines.   |
| BUFFER SIZE<br><i>block-count</i> | Specifies the size (in 32-word blocks) of the trace buffer used to collect trace data (range: 1 to 255; default: 4). |
| FILE <i>file</i>                  | Specifies the name of the file to which trace data is to be copied (default: LB:[1.6]PSITRACE.SYS).                  |

**Example:**

```
NCP>SET TRACE LINE SDP-0 STATE ON BUFFER SIZE 10
```

This command turns tracing on for line SDP-0 and allocates a 10-block buffer in which to collect trace data.

---

SHOW ALIAS

Use the NCP SHOW ALIAS command to display alias information stored in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
SHOW { ALL ALIASES
      ALIAS alias-name
      KNOWN ALIASES } [CHARACTERISTICS] [[SCOPE] scope] [TOfile]
```

where

ALL ALIASES	Displays information for all aliases.
ALIAS <i>alias-name</i>	Displays information for the specified alias only.
KNOWN ALIASES	Displays information for all aliases for the specified scope (default scope is your terminal).
CHARACTERISTICS SUMMARY	See definitions in Section 1.5.2.
SCOPE <i>scope</i>	(Invalid for ALL ALIASES) See definition in Section 1.5.2.
TO <i>file</i>	(Valid for NCP only) See definition in Section 1.5.2.

**Example:**

```
NCP>SHOW ALIAS SMITH CHARACTERISTICS TERMINAL TT24: -<RET>
NCP>TO LB:[1,9]ALIAS.SYS
```

This command (shown in continuation format) directs static display information for alias SMITH on terminal TT24: to file LB:[1.9]ALIAS.SYS.

## NCP

---

### SHOW AREA NP

Use the NCP SHOW AREA command to display area information stored in the volatile database.

#### Format:

```
SHOW { AREA area-number
      ACTIVE AREAS
      KNOWN AREAS
      SIGNIFICANT AREAS } { CHARACTERISTICS
                          STATUS
                          SUMMARY } [ TO file ]
```

where

AREAS <i>area-number</i>	Displays information for the specified area only.
ACTIVE AREAS	Displays information for all active areas.
KNOWN AREAS	Displays information for all areas.
SIGNIFICANT AREAS	Displays information for all areas for which information is available.
CHARACTERISTICS STATUS SUMMARY	See definitions in Section 1.5.2.
TO <i>file</i>	See definition in Section 1.5.2.

#### Example:

```
NCP>SHOW AREA 4 CHARACTERISTICS
```

This command displays static information for area number 4.

---

SHOW CIRCUIT

Use the NCP SHOW CIRCUIT command to display circuit information stored in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
SHOW {CIRCUIT circuit-id
      ACTIVE CIRCUITS
      KNOWN CIRCUITS
      SIGNIFICANT CIRCUITS} [CHARACTERISTICS
                              COUNTERS
                              STATUS
                              SUMMARY] [TO file]
```

where

CIRCUIT <i>circuit-id</i>	Displays information for the specified circuit only.
ACTIVE CIRCUITS	(Invalid for VNP) Displays information for all active circuits.
KNOWN CIRCUITS	Displays information for all known circuits.
SIGNIFICANT CIRCUITS	(Invalid for VNP) Displays information about all circuits for which information is available.
CHARACTERISTICS COUNTERS STATUS SUMMARY	See definitions in Section 1.5.2.
TO <i>file</i>	(Valid for NCP only) See definition in Section 1.5.2.

**Example:**

```
NCP>SHOW ACTIVE CIRCUITS COUNTERS
```

This command displays circuit error and performance statistics for all active circuits.

---

**SHOW EXECUTOR**

Use the NCP SHOW EXECUTOR command to display local node information stored in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
SHOW EXECUTOR [ CHARACTERISTICS  
                COUNTERS  
                STATUS  
                SUMMARY ] [ TO file ]
```

where

CHARACTERISTICS                      See definitions in Section 1.5.2.  
COUNTERS  
STATUS  
SUMMARY

TO *file*                                (Valid for NCP only) See definition in  
Section 1.5.2.

**Example:**

```
NCP>SHOW EXECUTOR STATUS
```

This command displays local node dynamic information.

---

SHOW LINE

Use the NCP SHOW LINE command to display line information stored in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
SHOW { LINE line-id
      ACTIVE LINES
      KNOWN LINES
      SIGNIFICANT LINES } [ CHARACTERISTICS
                          COUNTERS
                          STATUS
                          SUMMARY ] [ TO file ]
```

where

LINE <i>line-id</i>	Displays information for the specified line only.
ACTIVE LINES	(Invalid for VNP) Displays information for all active lines.
KNOWN LINES	Displays information for all known lines.
SIGNIFICANT LINES	(Invalid for VNP) Displays information about all lines for which information is available.
CHARACTERISTICS COUNTERS STATUS SUMMARY	See definitions in Section 1.5.2.
TO <i>file</i>	(Valid for NCP only) See definition in Section 1.5.2.

**Example:**

```
NCP>SHOW LINE DMC-1 SUMMARY TO LB:[1,6]LINSUM.SYS
```

This command directs the most useful information about line DMC-1 to a file named LB:[1,6]LINSUM.SYS.

SHOW LOGGING

Use the NCP SHOW LOGGING command to display logging information stored in the volatile database. Use the VNP command to do the same with the system image file.

Format:

```

SHOW { ACTIVE LOGGING
      KNOWN LOGGING
      LOGGING CONSOLE
      LOGGING FILE
      LOGGING MONITOR
      SIGNIFICANT LOGGING } [ CHARACTERISTICS ] [ TO file ]
                               [ STATUS
                               SUMMARY
                               EVENTS ] [ KNOWN SINKS
                                          SINK NODE node-id ]

```

where

- ACTIVE LOGGING (Invalid for VNP) Displays all active logging information.
- KNOWN LOGGING Displays information for all known logging components.
- LOGGING CONSOLE Displays information for the console logging component.
- LOGGING FILE Displays information for the file logging component.
- LOGGING MONITOR Displays information for the monitor logging component.
- SIGNIFICANT LOGGING (Invalid for VNP) Displays logging information about all sink types for which information is available.
- CHARACTERISTICS STATUS SUMMARY See definitions in Section 1.5.2.

EVENTS	Displays event class and type information for the given logging component.
TO <i>file</i>	(Valid for NCP only) See definition in Section 1.5.2.
KNOWN SINKS	(Default) Displays logging information for all known sink nodes.
SINK NODE <i>node-id</i>	Displays logging information for the specified sink node only.

**Example:**

```
NCP>SHOW LOGGING FILE EVENTS SINK NODE CHI42
```

This command displays event class and type information for the logging file on node CHI42.

---

**SHOW MODULE  
X25-ACCESS**

Use the NCP SHOW MODULE X25-ACCESS command to display remote DTE destination information stored in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
SHOW MODULE X25-ACCESS { ALL DESTINATIONS
                        DESTINATION dest-name
                        KNOWN DESTINATIONS } [ CHARACTERISTICS SUMMARY ]
[[SCOPE] scope] [TO file]
```

where

ALL DESTINATIONS	Displays information for all destinations, regardless of scope.
DESTINATION <i>dest-name</i>	Displays information for the specified destination only.
KNOWN DESTINATIONS	Displays information for all known destinations within the specified scope.
CHARACTERISTICS SUMMARY	See definitions in Section 1.5.2.
SCOPE <i>scope</i>	See definition in Section 1.5.2.
TO <i>file</i>	(Valid for NCP only) See definition in Section 1.5.2.

**Example:**

```
NCP>SHOW MODULE X25-ACCESS ALL DESTINATIONS CHARACTERISTICS
```

This command displays static information for all destinations.

---

**SHOW MODULE  
X25-PROTOCOL**

Use the NCP SHOW MODULE X25-PROTOCOL command to display protocol module information stored in the volatile database. Use the VNP command to do the same with the system image file. Some types of information can be qualified by either DTE or group.

**Format:**

```

SHOW MODULE X25-PROTOCOL [ DTE dte-address ] [ CHARACTERISTICS ] [ TO file ]
                          KNOWN DTES      [ COUNTERS ]
                                          [ SUMMARY ]

                          GROUP group-name [ CHARACTERISTICS ] [ TO file ]
                          KNOWN GROUPS     [ SUMMARY ]

CHARACTERISTICS [ TO file ]
STATUS
SUMMARY

```

where

DTE <i>dte-address</i>	Displays information for the specified DTE only.
KNOWN DTES	Displays information for all known DTEs.
GROUP <i>group-name</i>	Displays information for the specified group only.
KNOWN GROUPS	Displays information for all known groups.
CHARACTERISTICS COUNTERS STATUS SUMMARY	See definitions in Section 1.5.2.
TO <i>file</i>	(Valid for NCP only) See definition in Section 1.5.2.

**Example:**

```
NCP>SHOW MODULE X25-PROTOCOL KNOWN GROUPS SUMMARY
```

This command displays the most useful information about all known groups in the network.

SHOW MODULE  
X25-SERVER  
SHOW MODULE  
X29-SERVER

Use the NCP SHOW MODULE X25/X29-SERVER commands to display server module information stored in the volatile database. Use the VNP commands to do the same with the system image file. Some types of information can be qualified by destination.

**Format:**

```
SHOW MODULE {X25-SERVER } [DESTINATION dest-name ] [CHARACTERISTICS ] [TO file ]
             {X29-SERVER } [KNOWN DESTINATIONS ] [SUMMARY ]
                                     CHARACTERISTICS [TO file ]
                                     COUNTERS
                                     STATUS
                                     SUMMARY
```

where

DESTINATION <i>dest-name</i>	Displays information for the specified destination only.
KNOWN DESTINATIONS	Displays information for all known destinations.
CHARACTERISTICS COUNTERS STATUS SUMMARY	See definitions in Section 1.5.2.
TO <i>file</i>	(Valid for NCP only) See definition in Section 1.5.2.

**Examples:**

```
NCP>SHOW MODULE X25-SERVER COUNTERS TO LB:[1,6]X25CTR.SYS
```

This command directs the error and performance statistics for the module to a file named LB:[1.6]X25CTR.SYS.

```
NCP>SHOW MODULE X29-SERVER DESTINATION CHI55 CHARACTERISTICS
```

This command displays static module information for destination CHI55.

**SHOW NODE**

Use the NCP SHOW NODE command to display node information stored in the volatile database. Use the VNP command to do the same with the system image file. See the SHOW EXECUTOR command for executor node displays.

**Restriction:**

No information will be displayed for an end node until a link has been established to it. The node may appear to be unreachable even when it is not.

**Format:**

```
SHOW  { NODE node-id
        ACTIVE NODES
        ADJACENT NODES
        KNOWN NODES
        LOOP NODES
        SIGNIFICANT NODES } [ CHARACTERISTICS
                             COUNTERS
                             STATUS
                             SUMMARY ] [ TO file ]
```

where

NODE <i>node-id</i>	Displays information for the specified node only.
ACTIVE NODES	(Invalid for VNP) For a routing node, displays information for all reachable nodes. For a nonrouting node, displays information for all nodes that are (1) adjacent, (2) designated routers, or (3) connected to the executor by a logical link.
KNOWN NODES	Displays information for all known nodes.
LOOP NODES	Displays information for all loop nodes.
SIGNIFICANT NODES	(Invalid for VNP) Displays information about all nodes for which any information is available.

CHARACTERISTICS  
COUNTERS  
STATUS  
SUMMARY

See definitions in Section 1.5.2.

TO *file*

(Valid for NCP only) See definition in  
Section 1.5.2.

**Example:**

```
NCP>SHOW KNOWN NODES COUNTERS
```

This command displays error and performance statistics for all known nodes.

---

**SHOW OBJECT**

Use the NCP SHOW OBJECT command to display object information stored in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
SHOW {OBJECT type-code} [CHARACTERISTICS] [TO file]  
     {KNOWN OBJECTS} [SUMMARY]
```

where

OBJECT <i>type-code</i>	Displays information for the specified object only.
KNOWN OBJECTS	Displays information for all known objects.
CHARACTERISTICS SUMMARY	See definitions in Section 1.5.2.
TO <i>file</i>	(Valid for NCP only) See definition in Section 1.5.2.

**Example:**

```
NCP>SHOW OBJECT 25
```

This command displays only the most useful information about object 25.

**SHOW PROCESS**

Use the NCP SHOW PROCESS command to display process information stored in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
SHOW { PROCESS process-name
      ACTIVE PROCESSES
      KNOWN PROCESSES } [STATUS
                        SUMMARY] [TO file]
```

where

PROCESS <i>process-name</i>	Displays information for the specified process only.
ACTIVE PROCESSES	(Invalid for VNP) Displays information for all active processes.
KNOWN PROCESSES	Displays information for all known processes.
STATUS SUMMARY	See definitions in Section 1.5.2.
TO <i>file</i>	(Valid for NCP only) See definition in Section 1.5.2.

**Example:**

```
NCP>SHOW PROCESS KDP STATUS
```

This command displays dynamic information for process KDP.

---

**SHOW SYSTEM**

Use the NCP SHOW SYSTEM command to display system information stored in the volatile database. Use the VNP command to do the same with the system image file.

**Format:**

```
SHOW SYSTEM [ CHARACTERISTICS ] [ TO file ]
             [ COUNTERS
             [ STATUS
             [ SUMMARY
```

where

CHARACTERISTICS                    See definitions in Section 1.5.2.  
COUNTERS  
STATUS  
SUMMARY

TO *file*                            (Valid for NCP only) See definition in  
Section 1.5.2.

**Example:**

```
NCP>SHOW SYSTEM COUNTERS
```

This command displays system error and performance statistics.

## NCP

NP

### SHOW TRACE

Use the SHOW TRACE command to display trace information stored in the volatile database (see Chapter 9 for more information on the trace utility). Trace information is kept only for PSI lines.

#### Format:

```
SHOW TRACE [STATUS] [SUMMARY] [TO file]
```

where

STATUS                    See definitions in Section 1.5.2.  
SUMMARY

TO *file*                   See definition in Section 1.5.2.

#### Example:

```
NCP>SHOW TRACE TO LB:[1,6]TRACE.SYS
```

This command directs a summary (by default) of trace information to file LB:[1,6]TRACE.SYS.

Use the TELL prefix to send an NCP command to a remote node for execution. TELL sets the executor for only one command and must prefix the command for which it is intended. The TELL part of the command always executes at the local node, even though the executor may be set to a remote node identification. You can specify access control information (if required) to connect to the remote node.

**Format:**

TELL *node-id* [*acc-con-info*] *ncp-command*

where

*node-id* Specifies the node, by name or alias, to receive and execute the NCP command.

*acc-con-info* Specifies access control information (if required) to connect to the node. (See format description in Section 1.5.2.)

*ncp-command* Represents any valid NCP command that is remotely executable. The *DECnet-RSX Network Management Reference Card* lists all NCP commands and flags those that are not executable remotely.

**Example:**

```
NCP>TELL BOS LOAD NODE ALB
```

This command sends the LOAD NODE command to node BOS, where it executes.

**TRIGGER NODE**

Use the TRIGGER NODE command to trigger the bootstrap mechanism of a node so that the node loads itself. This command initiates the loading sequence for an unattended system. For information on down-line load operations, see the *DECnet-RSX Network Management Concepts and Procedures* manual.

**Format:**

```
TRIGGER NODE node-id    VIA circuit-id
                        PHYSICAL ADDRESS E-address
                        [SERVICE] PASSWORD service-password
```

where

NODE <i>node-id</i>	Specifies the node whose bootstrap is to be triggered.
VIA <i>circuit-id</i>	Specifies the circuit over which the operation is to take place.
PHYSICAL ADDRESS <i>E-address</i>	(For Ethernet nodes only) Identifies the Ethernet physical address that the node currently uses to identify itself. Required for Ethernet circuits if the hardware address parameter has not been specified in the volatile database; see SET NODE.
SERVICE PASSWORD <i>service- password</i>	Specifies the password required to trigger the bootstrap mechanism on the node.

**NOTE**

If you do not want a password echoed to your terminal while you enter it, press the RETURN key after the keyword PASSWORD. NCP prompts for the password and turns off echoing until the next prompt.

**Examples:**

```
NCP>TRIGGER NODE NYC
```

This command triggers the bootstrap mechanism on node NYC to initiate a down-line load.

```
NCP>TRIGGER NODE NYC VIA DMC-O PASSWORD 5A5A5A5A
```

This command provides service password 5A5A5A5A to trigger the bootstrap mechanism on node NYC and initiate a down-line load over circuit DMC-0.

---

**TRIGGER VIA**

Use the TRIGGER VIA command to trigger the bootstrap mechanism of a node so that the node loads itself. This command initiates the loading sequence for an unattended system by triggering the node via the specified circuit. The node identification is obtained from the volatile database on the executor node. For information on down-line load operations, see the *DECnet-RSX Network Management Concepts and Procedures* manual.

**Format:**

```
TRIGGER VIA circuit-id    PHYSICAL ADDRESS E-address  
                    [SERVICE] PASSWORD service-password
```

where

VIA <i>circuit-id</i>	Specifies the circuit over which the operation is to take place.
PHYSICAL ADDRESS <i>E-address</i>	(For Ethernet nodes only) Identifies the Ethernet physical address that the node currently uses to identify itself. Required for Ethernet circuits if the hardware address parameter has not been specified in the volatile database; see SET NODE.
SERVICE PASSWORD <i>service- password</i>	Specifies the password required to trigger the bootstrap mechanism on the node.

**NOTE**

If you do not want a password echoed to your terminal while you enter it, press the RETURN key after the keyword PASSWORD. NCP prompts for the password and turns off echoing until the next prompt.

**Examples:**

```
NCP>TRIGGER VIA DMC-0
```

This command triggers the bootstrap mechanism on the node connected to circuit DMC-0.

```
NCP>TRIGGER VIA DMC-0 PASSWORD FFFFFFFF
```

This command provides service password FFFFFFFF to trigger the bootstrap mechanism on the node connected to circuit DMC-0.

**ZERO CIRCUIT**

Use the ZERO CIRCUIT command to zero circuit counters for the specified circuit(s). The executor node maintains these counters on a per circuit basis.

**Format:**

ZERO {CIRCUIT *circuit-id* } [COUNTERS]  
      {KNOWN CIRCUITS }

where

CIRCUIT                   Zeroes counters for the specified circuit only.  
*circuit-id*

KNOWN                   Zeroes counters for all known circuits.  
CIRCUITS

**Example:**

NCP>ZERO CIRCUIT DMC-0

This command zeroes circuit counters for circuit DMC-0.

---

**ZERO EXECUTOR**

Use the ZERO EXECUTOR command to zero node counters associated with and maintained on the executor node.

**Format:**

ZERO EXECUTOR [COUNTERS]

**ZERO LINE**

Use the ZERO LINE command to zero line counters for the specified line(s). The executor node maintains these counters on a per line basis.

**Format:**

ZERO  $\left\{ \begin{array}{l} \text{LINE } \textit{line-id} \\ \text{KNOWN LINES} \end{array} \right\} [\text{COUNTERS}]$

where

LINE *line-id*           Zeroes counters for the specified line only.

KNOWN LINES           Zeroes counters for all known lines.

**Example:**

NCP>ZERO LINE DUV-3

This command zeroes line counters for line DUV-3.

---

**ZERO MODULE  
X25-PROTOCOL**

Use the ZERO MODULE X25-PROTOCOL command to zero module counters for the specified DTE(s). The executor node maintains these counters on a per DTE basis.

**Format:**

```
ZERO MODULE X25-PROTOCOL { DTE dte-address } [COUNTERS]  
                          { KNOWN DTES }
```

where

DTE                      Zeroes counters for the specified DTE only.  
*dte-address*

KNOWN DTES              Zeroes counters for all known DTEs.

**Example:**

```
NCP>ZERO MODULE X25-PROTOCOL DTE 123456789
```

This command zeroes X.25 protocol module counters for DTE 123456789.

NCP

\_\_\_\_\_ P

ZERO MODULE  
X25-SERVER  
ZERO MODULE  
X29-SERVER

Use the ZERO MODULE X25/X29-SERVER commands to zero X.25/X.29 server module counters that are maintained on the executor node.

**Format:**

```
ZERO MODULE { X25-SERVER } [ COUNTERS ]  
              { X29-SERVER }
```

---

**ZERO NODE**

Use the ZERO NODE command to zero node counters for the specified node(s). The executor node maintains node counters on a per node basis.

**Format:**

ZERO  $\left\{ \begin{array}{l} \text{NODE } \textit{node-id} \\ \text{KNOWN NODES} \end{array} \right\} [\text{COUNTERS}]$

where

NODE *node-id*      Zeroes counters for the specified node only.

KNOWN NODES      Zeroes counters for all known nodes.

**Example:**

```
NCP>ZERO NODE GEERT
```

This command zeroes node counters for node GEERT.

**NCP**

---

**ZERO SYSTEM** P

Use the ZERO SYSTEM command to zero system counters associated with and maintained on the executor node.

**Format:**

ZERO SYSTEM [COUNTERS]

## 1.5.4 RSX-11S NCP Command Summary

This section summarizes the NCP commands that are supported by RSX-11S. CFE and VNP cannot be used on an RSX-11S system directly, however, they can be used to modify an RSX-11S system image stored on a host system. The supported NCP commands are a subset of the commands supported on RSX-11M and RSX-11M-PLUS operating systems. Unless otherwise noted in the restriction section of an individual command description, all of the following commands can be initiated both locally (that is, at the RSX-11S node) and remotely (not at the RSX-11S node).

The commands are presented in alphabetical order, using the same definitions described in Section 1.6.2. To distinguish this subset from the full set, NCP/S is printed at the top of each page above the command name. In addition, each command is designated as privileged (P) or nonprivileged (NP).

---

**LOOP EXECUTOR/NODE**

Use the LOOP NODE command to test a node in the network. You can include access control information if the node requires it. If you are testing the executor node, you can use the LOOP EXECUTOR command. Either command causes test blocks of data to be transmitted to the specified node. See the *DECnet-RSX Network Management Concepts and Procedures* manual for more information on loopback testing.

**Format:**

```

LOOP {NODE node-id[acc-con-info]} COUNT count
    {EXECUTOR} LENGTH length
    WITH {MIXED
        ONES
        ZEROES}

```

where

NODE <i>node-id</i>	Specifies a node for loopback testing.
<i>acc-con-info</i>	Specifies access control information, if required.
EXECUTOR	Specifies the executor node for loopback testing.
COUNT <i>count</i>	Specifies the number of blocks to be sent during loopback testing (range: 1 to 65535; default: 1 block is looped).
LENGTH <i>length</i>	Specifies the length (in bytes) of blocks to be sent during loopback testing. When testing over the Ethernet, the length must be a decimal integer in the range of 32 to 1484. Otherwise, the length must be a decimal integer in the range of 1 to <i>n</i> , where <i>n</i> must be less than the smaller of either the local looper buffer size or the remote mirror buffer size (default: 40 bytes).
WITH	Specifies the type of binary information to be sent during testing (default: MIXED -- a combination of ones and zeroes).

**Example:**

```
NCP>LOOP NODE NYC COUNT 10
```

This command loops 10 blocks of mixed test messages to remote node NYC. Each block is 40 bytes.

**SET CIRCUIT**

Use the SET CIRCUIT command to set the operational state for the specified circuit in the volatile database

**Format:**

```
SET CIRCUIT circuit-id STATE { OFF }  
                                { ON  }
```

where

**CIRCUIT**                      Specifies the circuit whose state is to be set.  
*circuit-id*

**STATE**                        Sets the circuit's operational state to ON or OFF.

**Example:**

```
NCP>SET CIRCUIT KDZ-1-3 STATE OFF
```

This command sets the state of circuit KDZ-1-3 to OFF.

---

**SET EXECUTOR HOST**

Use the SET EXECUTOR HOST command to specify a host node in the volatile database.

**Format:**

```
SET EXECUTOR HOST {node-address
                   $HOST
                   $LOCAL }
```

where

HOST	Specifies the address for the host node or
<i>node-address</i>	the local node to be used for down-line load, trigger, and up-line dump service operations.

**Example:**

```
NCP>SET EXECUTOR HOST 08
```

This command sets node 08 as the host node for down-line load, trigger, and up-line dump.

**SET LINE**

Use the SET LINE command to set the controller mode for a specified loaded line in the volatile database

**Restrictions:**

- SET LINE cannot be initiated from a remote node.
- The specified line must be in the OFF state (that is, not CLEARED).

**Format:**

```
SET LINE line-id CONTROLLER { LOOPBACK
                                NORMAL }
```

where

LINE *line-id* Specifies the line for which the controller mode is to be specified.

CONTROLLER For Ethernet, DMC/DMR, DMP, DHU, and DHV lines; Specifies the controller mode for the line. The line's circuit must be in the OFF state.

LOOPBACK Internal device loopback mode

NORMAL Normal operating mode

**Example:**

```
NCP>SET LINE DMC-0 CONTROLLER LOOPBACK
```

This command sets the operating mode for line DMC-0 to loopback mode.

Use the SET LOGGING CONSOLE command to modify the operational state of the logging console in the volatile database.

**Format:**

SET LOGGING CONSOLE STATE { OFF }  
                                  { ON }

where

STATE                   Sets the operational state of the logging console on the executor node. When the state is OFF, events are discarded.

**Example:**

```
NCP>SET LOGGING CONSOLE STATE OFF
```

This command sets the state of the logging console to OFF.

**SHOW CIRCUIT**

Use the SHOW CIRCUIT command to display circuit information stored in the volatile database.

**Restriction:**

ACTIVE, SIGNIFICANT, and KNOWN CIRCUITS cannot be initiated locally.

**Format:**

```
SHOW {CIRCUIT circuit-id
      ACTIVE CIRCUITS
      KNOWN CIRCUITS
      SIGNIFICANT CIRCUITS} [COUNTERS
                             STATUS
                             SUMMARY]
```

where

CIRCUIT <i>circuit-id</i>	Displays information for the specified circuit only.
ACTIVE CIRCUITS	Displays information for all active circuits.
KNOWN CIRCUITS	Displays information for all known circuits.
SIGNIFICANT CIRCUITS	Displays information for all significant circuits.
COUNTERS STATUS SUMMARY	See definitions in Section 1.5.2.

**Example:**

```
NCP>SHOW CIRCUIT DMC-0 COUNTERS
```

This command displays error and performance statistics for circuit DMC-0.

---

**SHOW EXECUTOR**

Use the **SHOW EXECUTOR** command to display local node information stored in the volatile database.

**Format:**

```
SHOW EXECUTOR [CHARACTERISTICS  
                COUNTERS  
                STATUS  
                SUMMARY]
```

where

**CHARACTERISTICS**      See definitions in Section 1.5.2.  
**COUNTERS**  
**STATUS**  
**SUMMARY**

**Example:**

```
NCP>SHOW EXECUTOR STATUS
```

This command displays local node dynamic information.

**SHOW LINE**

Use the SHOW LINE command to display line information stored in the volatile database.

**Restriction:**

ACTIVE LINES, SIGNIFICANT LINES, and KNOWN LINES cannot be initiated locally.

**Format:**

```
SHOW { LINE line-id
      ACTIVE LINES
      KNOWN LINES
      SIGNIFICANT LINES } [ COUNTERS
                          STATUS
                          SUMMARY ]
```

where

LINE <i>line-id</i>	Displays information for the specified line only.
ACTIVE LINES	Displays information for all active lines.
KNOWN LINES	Displays information for all known lines.
SIGNIFICANT CIRCUITS	Displays information for all significant circuits.
COUNTERS STATUS SUMMARY	See definitions in Section 1.5.2.

**Example:**

```
NCP>SHOW LINE DMC-1 SUMMARY
```

This command displays the most useful information about line DMC-1.

Use the **SHOW LOGGING CONSOLE** command to display dynamic logging console information stored in the volatile database.

**Format:**

**SHOW LOGGING CONSOLE STATUS**

where

**STATUS**                    NCP only; displays dynamic information for the specified component.

**SHOW NODE**

Use the **SHOW NODE** command to display node information stored in the volatile database. See the **SHOW EXECUTOR** command for executor node displays.

**Restrictions:**

- **ACTIVE NODES** and **KNOWN NODES** cannot be initiated locally.
- No information will be displayed for an end node until a link has been established to it. The node may appear to be unreachable even when it is not.
- Only name and address static characteristics kept on 11S.

**Format:**

```
SHOW { NODE node-id
      ACTIVE NODES
      KNOWN NODES } [ CHARACTERISTICS
                    COUNTERS
                    STATUS
                    SUMMARY ]
```

where

<b>NODE</b> <i>node-id</i>	Displays information for the specified node only.
<b>ACTIVE NODES</b>	Displays information for all active nodes.
<b>KNOWN NODES</b>	Displays information for all known nodes.
<b>CHARACTERISTICS</b>	See definitions in Section 1.5.2.
<b>COUNTERS</b>	
<b>STATUS</b>	
<b>SUMMARY</b>	

**Example:**

```
NCP>SHOW NODE LOSANG COUNTERS
```

This command displays error and performance statistics for node LOSANG.

---

**SHOW SYSTEM**

Use the **SHOW SYSTEM** command to display system information stored in the volatile database.

**Format:**

```
SHOW SYSTEM [CHARACTERISTICS  
            COUNTERS  
            STATUS  
            SUMMARY]
```

where

**CHARACTERISTICS**      See definitions in Section 1.5.2.  
**COUNTERS**  
**STATUS**  
**SUMMARY**

**Example:**

```
NCP>SHOW SYSTEM COUNTERS
```

This command displays system error and performance statistics.

---

**ZERO CIRCUIT**

Use the ZERO CIRCUIT command to zero circuit counters for the specified circuit(s). The executor node maintains these counters on a per circuit basis.

**Restriction:**

KNOWN CIRCUITS cannot be initiated locally.

**Format:**

ZERO  $\left\{ \begin{array}{l} \text{CIRCUIT } \textit{circuit-id} \\ \text{KNOWN CIRCUITS} \end{array} \right\}$  [COUNTERS]

where

CIRCUIT *circuit-id*                      Zeroes counters for the specified circuit only.

KNOWN CIRCUITS                      Zeroes counters for all known circuits.

**Example:**

```
NCP>ZERO CIRCUIT DUP-0
```

This command zeroes circuit counters for circuit DUP-0.

---

**ZERO EXECUTOR**

Use the ZERO EXECUTOR command to zero node counters associated with and maintained on the executor node.

**Format:**

ZERO EXECUTOR [COUNTERS]

**ZERO LINE**

Use the ZERO LINE command to zero line counters for the specified line(s). The executor node maintains these counters on a per line basis.

**Restriction:**

KNOWN LINES cannot be initiated locally.

**Format:**

ZERO  $\left\{ \begin{array}{l} \text{LINE } \textit{line-id} \\ \text{KNOWN LINES} \end{array} \right\}$  [COUNTERS]

where

KNOWN LINES      Zeroes counters for all known lines.

LINE *line-id*      Zeroes counters for the specified line only.

**Example:**

```
NCP>ZERO LINE DUV-3
```

This command zeroes line counters for line DUV-3.

---

**ZERO NODE**

Use the ZERO NODE command to zero node counters for the specified node(s). The executor node maintains node counters on a per node basis.

**Restriction:**

This command cannot be initiated locally.

**Format:**

ZERO  $\left\{ \begin{array}{l} \text{NODE } \textit{node-id} \\ \text{KNOWN NODES} \end{array} \right\}$  [ COUNTERS ]

where

NODE *node-id*        Zeroes counters for the specified node only.

KNOWN NODES        Zeroes counters for all known nodes.

**Example:**

```
NCP>TELL NET11S ZERO KNOWN NODES
```

This command zeroes node counters for all known nodes.

NCP/S

---

P  
ZERO SYSTEM

Use the ZERO SYSTEM command to zero system counters associated with and maintained on the executor node.

**Format:**

ZERO SYSTEM [COUNTERS]

## Console Carrier Requester (CCR)

The console carrier requester (CCR) uses the DLX interface to communicate with the console carrier server (CCS) residing on a target node to provide remote access to normal console services. More information about the use of CCR can be found in the *DECnet-RSX Network Management Concepts and Procedures* manual.

### 2.1 Running CCR

To use CCR use the following command:

```
CCR NODE node-id SERVICE CIRCUIT circuit-id
SERVICE PASSWORD password
PHYSICAL ADDRESS E-address
```

where

NODE <i>node-id</i>	specifies the target node by address (range: 1 to 1023) or name (1 to 6 alphanumeric characters, including at least 1 alphabetic character).
SERVICE CIRCUIT <i>circuit-id</i>	identifies the circuit to the target node. ( <i>circuit-id</i> has the format <i>dev-c[u][.t]</i> .)
SERVICE PASSWORD <i>password</i>	defines the password required to access the target node. The password is a hexadecimal number in the range of 0 to FFFFFFFFFFFFFFFF (up to 16 hexadecimal digits).

**PHYSICAL ADDRESS** identifies the Ethernet physical address that the node currently uses to identify itself. (Required if the hardware address parameter has not been specified in the volatile database; see the SET NODE command in Chapter 1.)  
*E-address*

The *node-id* is required. If the other parameters are not specified in the command line, they must be specified in the down-line load database. If they are specified in the command line, they override the parameters set in the down-line load database.

## 2.2 CCR Special Characters

Two special characters are supplied with the CCR software

<CTRL/B> - Operates as a BREAK command to get the attention of the console on-line debugging tool (ODT).

<CTRL/D> - Initiates an exit from console carrier mode.

## 2.3 CCR Error Handling

CCR error messages are listed in Chapter 10.

## Event File Interpreter (EVF)

The Event File Interpreter (EVF) is part of the event logging facility provided by DECnet-RSX. This facility enables you to collect events in a machine-readable file for later formatting by EVF. The event collector and Event File Interpreter are similar to the Error Logger and Error Report Generator provided by RSX-11M/M-PLUS.

### 3.1 Invoking EVF

Any user can use EVF to create a formatted event listing because EVF is a nonprivileged task. EVF can be invoked using either the DCL CLI or the MCR CLI. To invoke EVF using the DCL CLI type the following command:

```
ANALYZE/EVENT_LOG[qualifiers] [input_file]
```

where *qualifiers* is a list of command qualifiers and *input\_file* is the name of the input file.

There are three methods of invoking EVF using the MCR CLI:

#### 1. MCR>EVF

EVF then prompts:

```
EVF>
```

Enter an EVF command immediately following the prompt. If the command executes successfully, the utility will redisplay the prompt on the next line. If an error occurs, an error message is displayed indicating the reason for the error followed by a prompt.

## 2. MCR>EVF *command*

where *command* is a valid EVF command. With this method you will return to MCR after the command is executed.

## 3. MCR>EVF @*command-file*

where *command-file* is the name of an indirect command file containing valid EVF commands. Command files can be nested up to six deep. You will return to MCR after the commands in the file(s) have been executed.

## 3.2 Exiting EVF

To exit the EVF utility, enter <CTRL-Z> in response to the utility prompt.

## 3.3 EVF Command Format

The EVF command line has the following format:

**DCL format:** ANALYZE/EVENT\_LOG[*qualifiers*] [*infile*]

**MCR format:** EVF [*outfile*][*output-switches*] = [*infile*][*input-switches*]

where:

*outfile* is the output file name. If this file is not specified, the output file defaults to EVENTLOG.LST in the current UIC. This file will contain the formatted events.

*infile* is the input file name. If this file is not specified, the input file defaults to LB:[1,6]EVENTLOG.SYS. This file contains the machine-readable events. The file is generated by using the NCP SET LOGGING FILE command. For information on logging and NCP control of logging see the *DECnet-RSX Network Management Concepts* manual and the NCP command descriptions in Chapter 1.

*qualifiers* is one or more of the command qualifiers shown in Table 3-1 and 3-2. For more information on the use of these qualifiers, see the description of *input-switches*.

*output-switches* is one or more of the switches shown in Table 3-1. These switches control the output of the file.

*input-switches* is one or more of the switches shown in Table 3-2. These switches control which events in the event logging file will be formatted by EVF. The switches allow for event filtering based on the time of the event, the entity the event is associated with, the type of event, and the source node of the event. Many of the switches have a positive and a negated form. The positive form (*/sw*) causes all events specified by the switch to be formatted. The negated form (*/-sw* or */NOsw*) causes all events EXCEPT those specified by the switch to be formatted. Most of the filtering switches allow an argument to be specified with the switch (*/sw:argument*). The switches can be combined to provide a greater degree of event filtering; however, care must be taken to avoid specifying too many event filters in order to avoid generating an empty report.

### 3.3.1 Using the EVF Default Command Line

To use the EVF default command line, enter the following command:

**DCL format:** DCL>ANALYZE/EVENT\_LOG

**MCR format:** EVF>=**<RET>**

This command causes EVF to use the file specification defaults and qualifier or switch defaults. In general this command creates a full format report using all the events in the event log file.

The EVF default command line invokes the following qualifiers or switches:

**DCL qualifier:** /FULL

**MCR switch:** /FORMAT:FULL

Creates a full format report containing the complete event information for each event.

**DCL qualifier:** /ALL

**MCR switch:** /ALL

Creates a report with all events found in the error log file.

### 3.4 EVF Option Descriptions

This section describes in detail the EVF options. These options tell EVF the type of filtering desired on events as well as govern the disposition of the files used during the operation requested.

When using the MCR CLI, these options are specified as switches that are appended to either the input or output file name depending on which file the switch affects. You also have the option of using the default file names by specifying the switches without any input or output file name. The MCR switches may be abbreviated to the first two letters of the switch name. If the switch is abbreviated to one letter you will receive the following error message:

Unknown switch - *s*

where *s* is the one letter switch.

When using the DCL CLI, the options are specified as command qualifiers and the qualifiers are classified as input or output qualifiers only to indicate their logical use.

The EVF output options are summarized in Table 3-1. The EVF input options are summarized in Table 3-2.

**Table 3-1: EVF Output Options**

<b>MCR Switch</b>	<b>DCL Qualifier</b>	<b>Function</b>
(specified in command)	<i>/OUTPUT:filename</i>	Specifies output file name
<i>/SPOOL</i>	<i>/SPOOL</i>	Controls output file spooling
<i>/FORMAT:BRIEF</i>	<i>/BRIEF</i>	Specifies brief report
<i>/FORMAT: FULL</i>	<i>/FULL</i>	Specifies full report

**Table 3-2: EVF Input Options**

<b>MCR Switch(See Note 1)</b>	<b>DCL Qualifier</b>	<b>Function</b>
/DELETE	/DELETE	Controls input file disposition
/ALL	/ALL	Selects all events
/DATE:PREVIOUS: <i>n_days</i>	/PREVIOUS: <i>n_days</i>	Selects time frame for formatting
/DATE:RANGE: <i>start:end</i>	/SINCE: <i>start</i>	
	/THROUGH: <i>end</i>	
/DATE:TODAY	/TODAY	
/DATE:YESTERDAY	/YESTERDAY	
/ENTITY:ALL	/ENTITY:ALL	Selects entity (See Note 2)
/ENTITY:AREA[: <i>area-id</i> ]	/ENTITY:AREA[: <i>area-id</i> ]	
/ENTITY:CIRCUIT[: <i>cir-id</i> ]	/ENTITY:CIRCUIT[: <i>cir-id</i> ]	
/ENTITY:LINE[: <i>line-id</i> ]	/ENTITY:LINE[: <i>line-id</i> ]	
/ENTITY:MODULE[: <i>mod-id</i> ]	/ENTITY:MODULE[: <i>mod-id</i> ]	
/ENTITY:NODE[: <i>node-id</i> ]	/ENTITY:NODE[: <i>node-id</i> ]	
/ENTITY:NULL	/ENTITY:NULL	
/EVENT: <i>event-id</i>	/EVENT: <i>event-id</i>	Selects event types for formatting (See Note 2)
/SOURCE: <i>node-id</i>	/SOURCE: <i>node-id</i>	Selects source node for formatting (See Note 2)

1. All MCR switches and switch argument keywords must be at least two letters.
2. The /ENTITY, /EVENT, and /SOURCE options can have multiple arguments as described in Section 3.4.1.

### 3.4.1 Using Multiple Option Arguments

You can specify each EVF option only once in a command line. However, the same options provide an alternative syntax that allows you to specify a list of arguments for the option. The options that support multiple arguments are shown in Tables 3-1 and 3-2.

To specify more than one argument for an EVF option that allows multiple arguments, use the following command syntax:

```
/option:(argument1,argument2...argumentn)
```

The parentheses, which are a required part of the command syntax, allow EVF to use more than one argument for the option. If you do not use the parenthesis, EVF displays the following error message on your terminal:

Command line syntax error

For example, to specify multiple event types, use the following EVF option:

```
/EVENT:(4.*,5.*,6.*)
```

EVF will format all events for event classes 4, 5, and 6.

### 3.4.2 File Control Options

The file control options determine the disposition of the files after the execution of the EVF command.

### 3.4.2.1 The Output File Option -

**MCR Format:**

(Specified in command)

**DCL Format:**

*/OUTPUT:filename*

**Default:**

EVENTLOG.LST

*/OUTPUT:EVENTLOG.LST*

The OUTPUT option is only available when using DCL. This option specifies the output file name that EVF is to use. When using MCR you specify the output file name as part of the command syntax.

### 3.4.2.2 The Spooling Option -

**MCR Format:**

*/SP[OOL]*

*/-SP[OOL]*

*/NOSP[OOL]*

**DCL Format:**

*/SPOOL*

*/NOSPOOL*

**Default:**

*/-SP*

*/NOSPOOL*

The SPOOL option is used to control the spooling of the resultant formatted file. If the SPOOL option is used the file will be spooled to the system line printer.

### 3.4.2.3 The Delete Option -

**MCR Format:**                    **DCL Format:**

/DE[LETE]                    /DELETE  
/-DE[LETE]                /NODELETE  
/NODE[LETE]

**Default:**

/-DE                        /NODELETE

The DELETE option is used to control the disposition of the input file. If the DELETE option is used the input file will be deleted after formatting is complete.

### 3.4.3 Output File Format Control

**MCR Format:**                    **DCL Format:**

/FO[RMAT]: BR[IEF]        /BRIEF  
                  FU[LL]        /FULL

**Default:**

/FO:FU                        /FULL

The FORMAT option (or BRIEF and FULL option in DCL) is used to control the format of the output file.

The BRIEF argument (or BRIEF option in DCL) causes an abbreviated event description to be used for all events formatted in the report. This shortened form consists of the first three lines of the event text.

The FULL argument (or FULL option in DCL) causes the entire event text to be included in the report.

### 3.4.4 Event Selection Options

The following options are used to make a selection of the events to be formatted. You can select events based on the time of the event, the entity type, the event type, and source node of the event. These options can be combined to further limit the events formatted.

#### 3.4.4.1 The ALL Event Option -

**MCR Format:**            **DCL Format:**

/AL[L]                    /ALL

**Default:**

/AL                        /ALL

The ALL option specifies that all events are to be formatted. This option has the effect of specifying /DATE:RA:'.', ENTITY:AL, EVENT:'.\*', and SOURCE:'.\*.

#### 3.4.4.2 The DATE Options -

**MCR Format:**

/DA[TE]:PR[EVIOUS]:*n\_days*  
          RA[NGE]:*start:end*  
  
          TO[DAY]  
          YE[STERDAY]

**DCL Format:**

/PREVIOUS:*n\_days*  
/SINCE:*start*  
/THROUGH:*end*  
/TODAY  
/YESTERDAY

**Default:**

/DA:RA:\*.\*

/SINCE:\*/THROUGH:\*

The DATE option allows you to select events based on the date that an event occurred. This option includes arguments to specify a range of dates or to specify a particular day.

The RANGE argument (or SINCE and THROUGH options in DCL) accepts starting and ending dates in the standard RSX format:

DD-*MMM*-YY or (DD-*MMM*-YY HH:MM:SS)

When you use the date only format, EVF uses a starting time of 00:00:00 and an ending time of 23:59:59. If you use the second format, you must include the parentheses.

In MCR command lines the asterisk (\*) can be used at the beginning of a range specification to indicate any date through the specified ending date. For example, \*:12-MAR-85 specifies all events from the beginning of the event file through March 12, 1985 are to be formatted.

In MCR command lines the asterisks (\*) can be used at the end of a range specification to indicate any date since the specified starting date. For example, 23-FEB-85:\* specifies all events since February 23, 1985 are to be formatted.

You may not combine the arguments of the MCR DATE switch. You can combine the SINCE and THROUGH qualifiers when using DCL.

### 3.4.4.3 The Entity Filter Option -

#### MCR Format:

```
/EN[TITY]:AL[L]
    AR[EA][:area-id]
    CI[RCUIT][:cir-id]
    LI[NE][:line-id]
    MO[DULE][:mod-id]
    NO[DE][:node-id]
    NU[LL]
```

#### DCL Format:

```
/ENTITY:ALL
    AREA[:area-id]
    CIRCUIT[:cir-id]
    LINE[:line-id]
    MODULE[:mod-id]
    NODE[:node-id]
    NULL
```

#### Default:

```
/EN:AL
```

```
/ENTITY:ALL
```

The ENTITY option allows you to select events based on the entity type. This option includes arguments to specify the entity type desired as well as the ALL argument. The arguments (with the exception of the ALL argument) can be negated using the *-argument*(MCR only) or *NOargument* syntax. The arguments can be combined using the syntax described in Section 3.4.1. For example:

```
/ENTITY: (CIRCUIT, LINE, NONODE: 4.456)
```

will cause EVF to format all circuit and line events and node events for all nodes except node 4.456.

For a list of the entities associated with each event, see the *DECnet-RSX Network Management Concepts* manual.

#### NOTE

Although you may specify multiple option arguments or the /ENTITY option, you cannot specify more than one specific ID value for any one of the arguments.

The following is a detailed description of each of the /ENTITY option arguments.

- The ALL argument specifies that there is to be no entity filtering. The ALL argument cannot be negated.
- The AREA argument allows filtering on area entity events. You can specify an area ID in the range 0-63. The negated form of this argument (NOAREA:*area-id*) will cause all events EXCEPT the specified area events to be included in the report.
- The CIRCUIT argument allows filtering on circuit entity events. You can specify a circuit ID of the form *dev-ctl[-unit][.trib]* where *dev* is the device name, *ctl* is the controller number, *unit* is the unit number, and *trib* is the tributary address. When specifying the circuit ID, you can use the wildcard symbol (\*) for *ctl*, *unit*, or *trib*. If you specify a wildcard symbol, you cannot specify any explicit values for the parameters following the wildcard symbol (ie. PCL- is correct, PCL-.0 is not correct). The negated format of this argument (NOCIRCUIT:*cir-id*) will cause all events EXCEPT the specified circuit events to be included in the report.
- The LINE argument allows filtering on line entity events. You can specify a line ID of the form *dev-ctl[-unit]* where *dev* is the device name, *ctl* is the controller number, and *unit* is the unit number. When specifying the line ID, you can use the wildcard symbol (\*) for *ctl* or *unit*. If you specify a wildcard symbol for *ctl*, you cannot specify an explicit value for *unit*. (ie. DZ- is correct, DZ--0 is not correct). The negated form of this argument (NOLINE:*line-id*) will cause all events EXCEPT the specified line events to be included in the report.

- The **MODULE** argument allows filtering on module entity events. You can specify a module ID where the module ID is an alphanumeric string. The negated form of this argument (**NOMODULE:module-id**) will cause all events **EXCEPT** the specified module events to be included in the report.
- The **NODE** argument allows filtering on node entity events. You can specify a node ID where the node ID is of the form *[area.]number*. The value for *area* must be in the range 0-63 and *number* must be in the range 1-1023. The node id can also be a node name. The negated form of this argument (**NONODE:node-id**) will cause all events **EXCEPT** the specified node events to be included in the report.
- The **NULL** argument allows filtering on events that do not have an entity. The negated form of this argument (**NONULL**) will cause all events **EXCEPT** the events without entities to be included in the report.

#### 3.4.4.4 The Event Specification Filter Option -

##### MCR Format:

```
/EV[ENT]: event_type
/-EV[ENT]: event_type
/NOEV[ENT]:event_type
```

##### DCL Format:

```
/EVENT: event_type
/NOEVENT:event_type
```

##### Default:

```
/EV:.*          /EVENT:.*
```

The **EVENT** option allows you to select events based on the event class and type. If the negated format is used (**/NOEVENT:event-type** ) all events **EXCEPT** those with the specified event ID will be formatted. The *event-type* argument is of the form *class.type* where *class* is the event class (0-511) and *type* is the event type (0-63). This option allows multiple events to be specified using the syntax described in Section 3.4.1.

If the asterisk (\*) is used for *type*, all events of the requested class are included in the report.

For a list of the event classes and types associated with each event, see the *DECnet-RSX Network Management Concepts* manual.

#### 3.4.4.5 The Source Node Filter Option -

##### MCR Format:

*/SO[URCE]:node\_id*  
*/-SO[URCE]:node\_id*  
*/NOSO[URCE]:node\_id*

##### DCL Format:

*/SOURCE: node\_id*  
*/NOSOURCE:node\_id*

##### Default:

*/SO:\*.\**                      */SOURCE:\*.\**

The SOURCE option allows you to select events based on the source node. If the negated format is used (*/NOSOURCE:node\_id*), EVF formats all events EXCEPT those with a source of *node\_id*. The *node\_id* value is a node address of the form *[area.]number* where *area* is in the range 1-63 and *number* is in the range 0-1023. The node ID can also be a node name. This option allows multiple nodes to be specified using the syntax described in Section 3.4.1.

If *area* is not specified, only events with an area value of 0 will be formatted. If the asterisk (\*) is used for *number*, all events in the specified area will be formatted.

#### NOTE

The node identified by this option is the node that generated the event. This node may or may not be the node entity associated with the event. The node argument on the ENTITY option should be used to filter events based on the node entity associated with the event.

#### 3.4.4.6 /SO Switch - Specify Source Node Filter -

##### Format:

*/SO[URCE]:node\_id*  
*/-SO[URCE]:node\_id*  
*/NOSO[URCE]:node\_id*

##### Default:

*/SO:\*.\**

The `/SOURCE` switch allows you to select events based on the source node. If the negated format is used (`/-SO:node_id` or `/NOSO:node_id`), EVF formats all events EXCEPT those with a source of `node_id`. The `node_id` value is a node address of the form `[area.]number` where `area` is in the range 1-63 and `number` is in the range 0-1023. If `area` is not specified, only events with an area value of 0 or 1 will be formatted.

If the asterisk (\*) is used for `number`, all events in the specified area will be formatted.

#### NOTE

The node identified by this switch is the node that generated the event. This node may or may not be the node associated with the event. The node qualifier on the `ENTITY` switch should be used to filter events based on the node associated with the event.

### 3.5 EVF Error Messages

EVF prints an error message on your terminal when an invalid command request is detected. EVF error messages are summarized in Chapter 10. EVF also generates errors if the input file is found to be in an unreadable format.

## KMX/KMY Dump Analyzer (KDA)

If the performance of a KMX/KMY device deteriorates and this cannot be attributed to a hardware fault, you can dump the KMX/KMY microcode for analysis by your local software specialist. The NCP KMX-DUMP command controls the dumping of the microcode to disk (see Chapter 1). The KMX/KMY dump analyzer task (KDA) formats and prints the contents of the KMX/KMY dump disk file.

You must have included both the dump KMX/KMY task (DUK) and the KMX/KMY dump analyzer task (KDA) in your system during NETGEN if you want to use this facility. The *RSX-11 PSI Generation Guide* gives details of the DUK and KDA tasks.

### 4.1 Invoking and Exiting KDA

Use any RSX-11 method of invoking tasks to enter KDA. The simplest method is to type:

```
MCR>KDA
```

Once KDA is running, it returns the prompt:

```
KDA>
```

Exit from KDA by entering <CTRL/Z>.

## 4.2 Using the KDA Command

The KDA command has the following format:

```
[outfile[switches]=] [infile[switches]]
```

where

*outfile* is the file specification for the analyzed dump listing file (default is SY:[UFD]*inputname*.LST, where UFD is the user's default user file directory). The = symbol marks this as the output file specification.

*infile* is the file specification for the binary input file containing the dump (default is LB:[1.6]PSIKMXMEM.SYS). Specify the same file that you specified for the output file in the KMX-DUMP LINE command.

*switches* are options to control the listing that KDA produces. Any compatible combination of the switches shown in Table 4-1 can be specified for either the input file or the output file.

**Table 4-1: KDA Command Switches**

Switch	Meaning	Comments
/ALL	Analyze and print the control memory	
/-ALL	Analyze and print only the data memory and registers	Default switch
/SP	Spool the output file	Default switch (Output is spooled unless /-SP is specified.)
/-SP	Do not spool the output file	Output file is retained for printing later

**Example:**

```
>KDA  
KDA>VORTEX/ALL=DK0:[7,21]FROGST.DMP
```

This command uses input file FROGST.DMP from UFD [7,21] on DK0: and prints an analysis of control memory to output file SY:[UFD]VORTEX.LST.



# 5

## LAT Control Program (LCP)

This chapter discusses the Local Area Transport process (LAT) used in the network generation procedure.

LAT control program (LCP) commands enable you to monitor and control the LAT process on a DECnet-RSX11M-PLUS or a DECnet-Micro/RXX service node. Using these commands, you can:

- *CREATE* the LAT terminals.
- *START* the LAT protocol.
- *STOP* the LAT protocol.
- Modify (*SET*) the LAT service node characteristics in the volatile database.
- Modify (*DEFINE*) the LAT service node characteristics in the permanent database.
- *SHOW* the LAT service node characteristics in the volatile database.
- *LIST* the LAT service node characteristics in the permanent database.
- *SHOW* the characteristics of a LAT terminal server.
- *SHOW* the LAT counters on the service node.
- *ZERO* the LAT counters on the service node.

## 5.1 INVOKING and EXITING LCP

You invoke the control program by typing

```
>LCP
```

The following prompt appears on your screen:

```
LCP> <RET>
```

Enter the LCP command after the prompt and press RETURN. You can also invoke LCP and enter a command on the same line:

```
>LCP command <RET>
```

To exit from LCP, type EXIT or CTRL/z.

## 5.2 LAT Characteristics

When you select LAT during the network generation procedure, the procedure automatically writes the default characteristics to the permanent database. When the process is started (see the START command in section 5.3.9), LCP uses the information stored in the permanent database to set up the volatile database. Characteristics recorded in the volatile database define the LAT protocol to the running system.

You can make changes to the permanent database using the DEFINE command (section 5.3.2). These changes take effect the next time the LAT process is started. Any changes made to the volatile database using the SET command (section 5.3.6) are effective immediately and are not saved when the LAT process is stopped

### 5.3 LCP COMMANDS: Format and Syntax

The following sections describe the format and syntax of each LCP command and specify the default values initially assigned to the LAT parameters. For information on how these commands affect the Ethernet, the service node, and the LAT terminals, refer to the *LAT Terminal Server Network Manager's Guide*.

As shown in Table 5-1, LCP command syntax is similar to that of the Digital Command Language (DCL).

**Table 5-1: LAT Command Format**

CREATE [/TERMINALS = n]

DEFINE CHARACTERISTICS /DEVICE = *line-id*  
/GROUPS = (*group1*, [*group2*, ..., *groupn*])  
/MAXIMUM\_BUFFERS = *number*  
/MULTICAST\_TIMER = *seconds*  
/NODE\_IDENTIFICATION = *announcement-string*  
/NODE\_NAME = *node\_name*  
/RECEIVE\_TIMEOUT = *seconds*  
/RETRANSMIT\_TIMER = *seconds*  
/SERVICE\_IDENTIFICATION = *announcement-string*  
/SERVICE\_NAME = *service\_name*

EXIT

HELP

LIST CHARACTERISTICS] [/NODE]

SET CHARACTERISTICS] /DEVICE = *line-id*  
/GROUPS = (*group1*, [*group2*, ..., *groupn*])  
/MAXIMUM\_BUFFERS = *number*  
/MULTICAST\_TIMER = *seconds*  
/NODE\_IDENTIFICATION = *announcement-string*  
/NODE\_NAME = *node\_name*  
/RECEIVE\_TIMEOUT = *seconds*  
/RETRANSMIT\_TIMER = *seconds*  
/SERVICE\_IDENTIFICATION = *announcement-string*  
/SERVICE\_NAME = *service\_name*

### Table 5-1 (cont.): LAT Command Format

SHOW     {CHARACTERISTICS} [/NODE]  
          {COUNTERS}            [/SERVER[ = *server\_name*]]

START     /DEVICE = *line-id*  
          /GROUPS = (*group1* [, *group2*, ..., *groupn*])  
          /MAXIMUM\_BUFFERS = *number*  
          /MULTICAST\_TIMER = *seconds*  
          /NODE\_IDENTIFICATION = *announcement-string*  
          /NODE\_NAME = *node\_name*  
          /RECEIVE\_TIMEOUT = *seconds*  
          /RETRANSMIT\_TIMER = *seconds*  
          /SERVICE\_IDENTIFICATION = *announcement-string*  
          /SERVICE\_NAME = *service\_name*

STOP     /ALL  
          /DEVICE = *line-id*

ZERO COUNTERS    /NODE  
                  /SERVER[ = *server-name*]

### 5.3.1 CREATE

Use the CREATE command to create the system data structures to support the LAT terminals. Before issuing the CREATE command make sure that the communication executive is loaded.

#### NOTE

You must issue the CREATE command before using the START command.

Once the LAT terminals are added to the system, they cannot be removed until the system has been rebooted.

#### Format:

```
CREATE [/TERMINALS = n]
```

where:

[/TERMINALS = n]            Creates *n* terminals where *n* is an integer in the range of 1 to 64. The default is 4.

### 5.3.2 DEFINE

Use the DEFINE command to change the LAT process characteristics in the permanent database. The DEFINE command can execute before or after LAT is started, but it does not affect the running system.

#### Format:

```
DEFINE CHARACTERISTICS /DEVICE = line-id
                        /GROUPS = (group1 [, group2 ..., groupn])
                        /MAXIMUM_BUFFERS = number
                        /MULTICAST_TIMER = seconds
                        /NODE_IDENTIFICATION = announcement-string
                        /NODE_NAME = node_name
                        /RECEIVE_TIMEOUT = seconds
                        /RETRANSMIT_TIMER = seconds
                        /SERVICE_IDENTIFICATION = announcement-st
                        /SERVICE_NAME = service_name
```

where:

*/DEVICE = line-id*

Specifies the Ethernet device that is to be used for LAT connections. In a multi-Ethernet device environment, specify the Ethernet device to be used for LAT terminal support. The default is the first Ethernet line in the system.

*/GROUPS = (group1  
[,group2,...groupn])*

Sets the groups to which the host belongs. There are 256 groups, numbered 0 to 255 decimal. The default is */GROUP = (0)*. For more information on group and group codes, see the individual server operations guides and the *LAT Network Manager's Guide*.

<p><code>/MAXIMUM_BUFFERS = number</code></p>	<p>Sets the upper limit for the number of large data buffers (LDBs) that the LAT process can have in use at any one time. For more information about LDBs and their use in the system, see the chapter on system buffers in the <i>DECnet-RSX Network Management Concepts and Procedures</i> manual. The default is 3.</p>
<p><code>/MULTICAST_TIMER = seconds</code></p>	<p>Sets the minimum rate at which the node will send multicast messages in seconds. The default is 20.</p>
<p><code>/NODE_IDENTIFICATION = announcement_string</code></p>	<p>Specifies the announcement string for the node. It can be up to 32 characters in length. Nonprintable characters are translated to spaces. You must enclose the string in double quotes (") if it contains one or more spaces.</p>
<p><code>/NODE_NAME = node_name</code></p>	<p>Specifies the name by which the service node is known to the terminal servers. The name is a string of 1 to 6 alphanumeric characters. The default is the RSX system name.</p>
<p><code>/RECEIVE_TIMEOUT = seconds</code></p>	<p>Sets the maximum time interval, in seconds, that the service node waits for a reply from a terminal server before declaring to the server that the circuit is "down". The default is 3.</p>
<p><code>/RETRANSMIT_TIMER = seconds</code></p>	<p>Specifies, in seconds, the interval between retransmission attempts. The default is 3. For more information about this parameter, see the <i>LAT Network Manager's Guide</i>.</p>

**/SERVICE\_IDENTIFICATION**  
= *announcement\_string*

Specifies the announcement string for the service. It can be up to 32 characters in length. Nonprintable characters are translated to spaces. You must enclose the string in double quotes (") if it contains one or more spaces.

**/SERVICE\_NAME**  
= *service\_name*

Specifies the service name for the service node. Service names allow a service node to be known by more than just the node name. The service name is a string of 1 to 6 alphanumeric characters. The default is the node name.

#### **NOTE**

Terminal server users always connect to service names, NOT service node names.

### 5.3.3 EXIT

Use the EXIT command to exit LCP.

### 5.3.4 HELP

Use the HELP command for help in LAT control programming.

Format:

```
CREATE [/terminal = n]  
SET [characteristics]host-characteristics  
SHOW CHARACTERISTICS [/NODE]  
SHOW COUNTERS [/NODE]  
STOP [/ALL]
```

where host characteristics are:

```
/DEVICE = lineid  
/GROUPS = (grpn,grpn [,grpn...])  
/MAXIMUM_BUFFERS = n  
/MULTICAST_TIMER = n  
/NODE IDENTIFICATION = "node announcement string"  
/NODE NAME = node name  
/RECEIVE_TIMEOUT = n  
/RETRANSMIT_TIMER = n  
/SERVICE_IDENTIFICATION = "service announcement string"  
/SERVICE_NAME = servicename  
/EXIT  
/START [host_characteristics]  
/SHOW CHARACTERISTICS/SERVER  
/SHOW COUNTERS/SERVER
```

### 5.3.5 LIST CHARACTERISTICS

Use the LIST command to display the characteristics of the LAT service node. The LIST command displays characteristics in the permanent database. You can use the LIST command before or after LAT is started.

Format:

```
LIST [CHARACTERISTICS] [/NODE]
```

where

/NODE                   Displays the characteristics of the service node.

#### Example:

```
LCP>LIST CHARACTERISTICS/NODE
```

causes the following display:

```
LCP Static Characteristics
```

```
Node name= \Boston\                   Service name= \Boston\
```

```
Node Identification= \Boston-Home of the Red Sox\
```

```
Service Identification= \Boston-Home of the Red Sox\
```

```
Groups: (0)
```

```
Multicast Timer=20
```

```
Maximum Buffers=3
```

```
Receive Timeout=3
```

```
Retransmit Timer=3
```

```
Terminals=4
```

```
Device=UNA-0
```

### 5.3.6 SET

Use the SET command to change the LAT process characteristics in the volatile database. The SET command can execute only after the LAT process has been started, and while it is active.

#### Format:

```
SET CHARACTERISTICS      /DEVICE = line-id
                        /GROUPS = (group1, [group2, ..., groupn])
                        /MAXIMUM_BUFFERS = number
                        /MULTICAST_TIMER = seconds
                        /NODE_IDENTIFICATION = announcement-string
                        /NODE_NAME = node_name
                        /RECEIVE_TIMEOUT = seconds
                        /RETRANSMIT_TIMER = seconds
                        /SERVICE_IDENTIFICATION = announcement-string
                        /SERVICE_NAME = service_name
```

where:

<i>/DEVICE = line-id</i>	Specifies the Ethernet device that is to be used for LAT connections. In a multi-Ethernet device environment, specify the Ethernet device to be used for LAT terminal support. The default is the first Ethernet line in the system.
<i>/GROUPS = (group1[,group2, ...groupn])</i>	Sets the groups to which the host belongs. There are 256 groups, numbered 0 to 255 decimal. The default is <i>/GROUP = (0)</i> . For more information on group and group codes, see the individual server operations guides and the <i>LAT Network Manager's Guide</i> .

/MAXIMUM_BUFFERS = <i>number</i>	Sets the upper limit for the number of large data buffers (LDBs) that the LAT process can have in use at any one time. For more information about LDBs and their use in the system, see the chapter on system buffers in the <i>DECnet-RSX Network Management Concepts and Procedures</i> manual. The default is 3.
/MULTICAST_TIMER = <i>seconds</i>	Sets the minimum rate at which the node will send multicast messages in seconds. The default is 20.
/NODE_IDENTIFICATION = <i>announcement_string</i>	Specifies the announcement string for the node. It can be up to 32 characters in length. Nonprintable characters are translated to spaces. You must enclose the string in double quotes (") if it contains one or more spaces.
/NODE_NAME = <i>node_name</i>	Specifies the name by which the service node is known to the terminal servers. The name is a string of 1 to 6 alphanumeric characters. The default is the RSX system name.
/RECEIVE_TIMEOUT = <i>seconds</i>	Sets the maximum time interval, in seconds, that the service node waits for a reply from a terminal server before declaring to the server that the circuit is "down". The default is 3.
/RETRANSMIT_TIMER = <i>seconds</i>	Specifies, in seconds, the interval between retransmission attempts. The default is 3. For more information about this parameter, see the <i>LAT Network Manager's Guide</i> .

**/SERVICE\_IDENTIFICATION**  
= *announcement\_string*

Specifies the announcement string for the service. It can be up to 32 characters in length. Nonprintable characters are translated to spaces. You must enclose the string in double quotes (") if it contains one or more spaces.

**/SERVICE\_NAME**  
= *service\_name*

Specifies the service name for the service node. Service names allow a service node to be known by more than just the node name. The service name is a string of 1 to 6 alphanumeric characters. The default is the node name.

#### **NOTE**

Terminal server users always connect to service names, NOT service node names.

### 5.3.7 SHOW CHARACTERISTICS

Use the SHOW command to display the characteristics of the LAT service node or the LAT terminal server. The SHOW command displays characteristics in the volatile database. You can use the SHOW command only after LAT is started and while it is running.

**Format:**

```
SHOW CHARACTERISTICS  /NODE
                      /SERVER[ = server_name]
```

where

/NODE                   (Default) Displays information about the characteristics of the service node.

**Example:**

```
LCP>SHOW CHARACTERISTICS/NODE
```

causes the following display:

LCP Characteristics

```
Node name= \Boston\                   Service name= \Boston\  
Node Identification= \Boston-Home of the Red Sox\  
Service Identification= \Boston-Home of the Red Sox\  

```

```
Groups: (0)                           LAT Version=5.0  
Multicast Timer=20                   Maximum Buffers=3  
Receive Timeout=3                   Retransmit Timer=3  
Terminals=4                          Device=UNA-0  
LAT Protocol is active
```

`/SERVER = server_name`

Displays information about the specified server.  
If you do not supply a server name, you receive  
the characteristics of all servers.

`LCP>SHOW CHARACTERISTICS/SERVER=GREENFIELD`

causes the following display:

`LCP Server Characteristics for GREENFIELD:`

`Ethernet address=AA-00-03-01-05-CD`

`Server is active`

`Active Users=2`

### 5.3.8 SHOW COUNTERS

Use the SHOW command to display the counters which are kept by the LAT process.

#### Format:

```
SHOW COUNTERS /NODE
                /SERVER[ = server-name]
```

where:

/NODE (Default) Shows the counters kept for the service node.

#### Example:

```
LCP>SHOW COUNTERS/NODE
```

causes the following display:

#### LCP Node Counters

```
2052      seconds since last zeroed
  83      messages received
  92      messages transmitted
5256      bytes received
3712      bytes transmitted
  0      retransmissions
  0      out of sequence
  0      invalid messages
  0      invalid slots
  0      illegal messages
  0      illegal slots
  0      resource allocation errors
```

*/SERVER = server\_name*

Displays for the specified server. If you do not supply a server name, LCP displays the counters of all servers.

**Example:**

LCP>SHOW COUNTERS/SERVER=GREENFIELD

causes the following display:

LCP Server Counter for Greenfield

2052	seconds since last zeroed
83	messages received
92	messages transmitted
5256	bytes received
3712	bytes transmitted
0	retransmissions
0	out of sequence
0	invalid messages
0	invalid slots

### 5.3.9 START

Use the START command to start the LAT process and to override parameter values in the permanent database.

#### NOTE

Be sure that you have loaded the communications executive and created the LAT terminals with the CREATE command before using the START command.

If the LAT process has not been loaded, and you issue the START command, the following message appears on the screen:

```
LAT process being loaded
```

When you issue the START command, it creates the volatile database using information contained in the permanent database. If you specify parameters when you issue the START command, they are placed in the volatile database.

#### Format:

```
START /DEVICE = line-id  
      /GROUPS = (group1[,group2, ..., groupn])  
      /MAXIMUM_BUFFERS = number  
      /MULTICAST_TIMER = seconds  
      /NODE_IDENTIFICATION = announcement-string  
      /NODE_NAME = node_name  
      /RECEIVE_TIMEOUT = seconds  
      /RETRANSMIT_TIMER = seconds  
      /SERVICE_IDENTIFICATION = announcement-string  
      /SERVICE_NAME = service_name
```

where:

<i>/DEVICE = line-id</i>	Specifies the Ethernet device that is to be used for LAT connections. In a multi-Ethernet device environment, specify the Ethernet device to be used for LAT terminal support. The default is the first Ethernet line in the system.
--------------------------	--

`/GROUPS=(group1  
[,group2...,groupn])`

Sets the groups to which the host belongs. There are 256 groups, numbered 0 to 255 decimal. The default is `/GROUP=(0)`. For more information on group and group codes, see the individual server operations guides and the *LAT Network Manager's Guide*.

`/MAXIMUM_BUFFERS=  
number`

Sets the upper limit for the number of large data buffers (LDBs) that the LAT process can have in use at any one time. For more information about LDBs and their use in the system, see the chapter on system buffers in the *DECnet-RSX Network Management Concepts and Procedures* manual. The default is 3.

`/MULTICAST_TIMER=  
seconds`

Node will send multicast messages in seconds. The default is 20.

`/NODE_IDENTIFICATION  
=announcement_string`

Specifies the announcement string for the node. It can be up to 32 characters in length. Nonprintable characters are translated to spaces. You must enclose the string in double quotes (") if it contains one or more spaces.

`/NODE_NAME=node_name`

Specifies the name by which the service node is known to the terminal servers. The name is a string of 1 to 6 alphanumeric characters. The default is the RSX system name.

`/RECEIVE_TIMEOUT=  
seconds`

Sets the maximum time interval, in seconds, that the service node waits for a reply from a terminal server before declaring to the server that the circuit is "down". The default is 3.

**/RETRANSMIT\_TIMER**  
= *seconds*

Specifies, in seconds, the interval between retransmission attempts. The default is 3. For more information about this parameter, see the *LAT Network Manager's Guide*.

**/SERVICE\_IDENTIFICATION**  
= *announcement\_string*

Specifies the announcement string for the service. It can be up to 32 characters in length. Nonprintable characters are translated to spaces. You must enclose the string in double quotes (") if it contains one or more spaces.

**/SERVICE\_NAME**  
= *service\_name*

Specifies the service name for the service node. Service names allow a service node to be known by more than just the node name. The service name is a string of 1 to 6 alphanumeric characters. The default is the node name.

#### **NOTE**

Terminal server users always connect to service names, NOT service node names.

### 5.3.10 STOP

Use the STOP command to shut down the LAT process. Active connections are broken without warning. You can use the RSX11M-PLUS BROADCAST command to warn LAT users before this command is used.

#### Format:

```
STOP    /ALL
        /DEVICE = line-id
```

where:

*/ALL* (Default) Stops the use of all the terminals associated with the LAT process.

*/DEVICE = line-id* Stops all terminals associated with the particular line.

#### NOTE

To stop the LAT process, use the following procedure:

1. Issue an LCP SET command with the */NODE\_IDENTIFICATION* set to a message informing users the server is coming down. For example,

```
LCP> SET/NODE_IDENTIFICATION = "LAT shutdown in 10
minutes."
```

2. Issue a broadcast command.

```
BROADCAST LOG: LAT shutdown in 10 minutes
```

3. Issue the LCP STOP command.

### 5.3.11 ZERO [COUNTERS]

Use the ZERO command to zero the counters kept by the LAT process. To test the performance of the system over a specified period of time, first zero the counters and then analyze the information that accumulates during the specified period.

**Format:**

```
ZERO [COUNTERS] /NODE  
                /SERVER[ = server-name]
```

where:

/NODE                                      Zeroes the counters kept for the local node.

/SERVER = *server-name*                  Zeroes the counters kept for the specified server.

#### NOTE

The Ethernet lines that the LAT process uses also have counters. You can show these counters with the NCP SHOW LINE COUNTERS command and zero them with the NCP ZERO LINE COUNTERS command (see Chapter 1 for NCP command descriptions).

## Network Crash Dump Analyzer (NDA)

The Network Crash Dump Analyzer (NDA) utility helps you to determine the cause of network crashes. NDA is a nonprivileged task for use only with DECnet-RSX systems, including those with a PSDN capability. If you have selected crash dump analysis support during RSX system generation, then you can also select network crash dump analysis support during DECnet-RSX network generation. Refer to the *RSX-11M System Generation and Installation Guide* for information on obtaining crash dump support for your operating system. Refer to the *DECnet-RSX Network Generation and Installation Guide* for information on obtaining network crash dump support for DECnet-RSX.

NDA performs crash dump analysis for the network data structures in the same way that the RSX crash dump analyzer (CDA) analyzes the operating system data structures. Specifically, NDA compiles and analyzes information from the system's executive symbol table file, the network symbol table file, and the crashed system image. The information is then output in a listing file or at a line printer in a format that is useful to you. Note that an effective use of NDA requires an extensive knowledge of DECnet-RSX internal data structures.

Command switches provide you with the means to control NDA operation. There are two types of switches: analysis control switches and function control switches. Analysis control switches determine which NDA analysis routines are applied to the input file for analysis. These switches provide such options as:

- Dumping the contents of a specified octal address range of memory
- Listing general system data (time of crash, registers in use, and so forth)
- Listing the routing parameters and databases

- Listing the remote node names and alias node names
- Listing PSI data structures (local and remote DTE descriptor blocks, NW: database and window blocks, and so forth)
- Listing the network services protocol (ECL) databases

Function control switches provide a number of NDA control options, such as:

- Identifying the starting block number of the secondary storage device
- Terminating the analysis after a specified number of errors has been encountered
- Limiting the number of pages of analysis output
- Creating a save file and transferring a specified amount of the crash dump to it

You can type NDA commands at a terminal, or you can place them in an indirect command file.

## 6.1 NDA Operation

NDA operates in the following manner:

1. First, it reads the contents of the memory dump created by the executive crash dump routine.
2. Next, it analyzes the dump in accordance with information contained in a combined executive/network symbol table file.
3. Finally, it formats and outputs the dump to a line printer or a listing file for your evaluation.

## 6.2 System Requirements

To obtain a crash dump that can be handled by NDA, you must first build the executive crash dump routine into your system during phase 1 of an RSX-11 SYSGEN. This is accomplished by entering B in response to the query:

50. Crash dump analysis: A- PANIC B- Crash dump analysis (CDA)

If you select this option, code is included in your system to dump memory to a secondary storage device following a system crash. Your secondary storage device can be any one of a number of different types of disk units, a magtape unit, or a DECTape unit. You can find complete information on RSX-11 SYSGEN in the *RSX-11M System Generation and Installation Guide*.

## 6.3 Obtaining a Crash Dump

To obtain a crash dump, control of the processor must be transferred to the executive crash dump routine following a system crash. How this control is transferred depends on the way in which the crash occurred and whether or not you have built the executive debugging tool (XDT) into your system at SYSGEN.

System crashes occur in three ways:

- The processor encounters an unintentional halt instruction (000000).
- An infinite loop condition occurs.
- The processor encounters a program condition causing it to trap.

When a system crash is the result of an unintentional halt instruction or an infinite loop condition, you must restart your processor at location 40.

When a program condition causes a processor trap and XDT is included in your system, control transfers automatically to XDT. When you type X at your terminal, XDT transfers control to the executive crash dump routine. When XDT is not included in your system, a processor trap causes control to be transferred to the crash dump routine. Refer to the *RSX-11M Guide to Writing an I/O Driver* for a description of XDT.

Regardless of the manner in which control is transferred, once the executive crash dump routine is entered, it prints the following message on the crash notification device specified during SYSGEN:

```
CRASH - CONT WITH SCRATCH MEDIA ON xxx
```

where *xxx* is the mnemonic for the device that is to receive the crashed system image.

You can then put the secondary crash dump device on line and depress the CONT switch on the CPU console. The executive crash dump routine dumps memory to the crash dump device and halts the processor upon completion.

## 6.4 Running NDA

You can run NDA as either an installed or an uninstalled task.

### 6.4.1 Running NDA as an Installed Task

When you run NDA as an installed task, the command format that you use determines where control is returned upon completion of the analysis:

- To return control to the executive:

```
>NDA command string
```

- To return control to NDA:

```
>NDA  
NDA> command string
```

### 6.4.2 Running NDA as an Uninstalled Task

When you run NDA as an uninstalled task, control is always returned to NDA upon completion of the analysis. The commands for running NDA are:

RUN \$NDA	This requires NDA.TSK to be present in the user
NDA> <i>command string</i>	file directory corresponding to the system UIC on
	device LB:.

RUN NDA	This requires NDA.TSK to be present in the user
NDA> <i>command string</i>	file directory corresponding to the current UIC on
	device SY:.

## 6.5 NDA Command Syntax

Use the following format to input commands to NDA:

```
[analysis[/sw1]], [save[/MEMSIZ:n]] = [symbols/STB][,input[/sw2]]
```

where

*analysis* is the crash dump analysis listing file. If a file extension is not specified, the default extension is .LST. If an analysis file is not specified, no analysis list file is created.

*/sw1* consists of the /EXIT:*n*, /LIMIT:*n*, and /-SP function control switches for the analysis file. A description of each switch and its default value are given in Table 6-3.

*save* is the crash dump save file. This file is a copy of the binary data that was written to the dump device by the dump routine. It allows you to create a historical record of crash dumps.

/MEMSIZ:*n* is the number of K words of information to be transferred to the save file. A description of the MEMSIZ:*n* function control switch and its default value is given in Table 6-3.

*symbols/STB* is the symbol table file for the crashed system. This file must define all executive and network symbols. A description of /STB and its default value is given in Table 6-3.

*input* is the binary input to NDA. This specification can be either a device name (your secondary storage device) or the name of a crash dump save file.

*/sw2* consists of all the analysis control switches and three function control switches -- /KMR, /SYM, and /BL:*n*. A description of the analysis control switches is provided in Section 6.7.1. A description of the /KMR, /SYM, and /BL:*n* switches and their default values is given in Table 6-3.

Output specifications (to the left of the equal sign) are position dependent. Therefore, when you include both output specifications (analysis, save), you must place them in the command string in the positions shown. If you do not enter a specification for the analysis file, you must place a comma before the save file specification. The comma can appear only if a save file is specified.

Input specifications (to the right of the equal sign) are position independent and can appear in either order.

Any of the specifications in the command string can appear in the complete FILES-11 format -- that is, complete with device, UIC, file name, file type, and version number. When you omit any of the command specifications, NDA uses the defaults shown in Table 6-1.

**Table 6-1: NDA File Default Values**

<b>File</b>	<b>Device</b>	<b>User File Directory (UFD)</b>	<b>Default File Name</b>	<b>Value Type</b>
Analysis file	SY:	Current	None	.LST
Save file	SY:	Current	None	.CDA
Symbols files:				
11M/S	SY:	Current	CEX	.STB
11M-PLUS and Micro/R SX	SY:	Current	RSX11M	.STB
Input file	SY:	Current	None	.CDA

The following examples illustrate the NDA file default values. Assume that the user in these examples is logged in under UIC [301,356] and that the secondary storage device is DK1:. Also assume that NDA is being operated as an installed task.

**Example 1:**

```
>NDA
NDA>18APR85,18APR85=CEX.STB/STB,DK1:/ALL
```

This command string creates an analysis file, 18APR85.LST, and a save file, 18APR85.CDA. The binary input is read from RK05 disk unit 1. NDA analyzes it in accordance with the information contained in the symbol definition file CEX.STB under UFD [301,356].

**Example 2:**

```
>NDA
NDA>,13JAN85=DK1:
```

This command string creates a save file, 13JAN85.CDA. In this case, NDA performs no analysis, since there is no analysis listing file. The binary input is read from RK05 disk unit 1.

**Example 3:**

```
>NDA
NDA>13JAN85=13JAN85/ALL
```

This command string produces an analysis output listing using the /ALL analysis switch (see Table 6-2 for a complete description of this switch). The binary input is read from the previously created save file (13JAN85.CDA). NDA analyzes it in accordance with the information contained in the symbol definition file CEX.STB under UFD [301,356].

**Example 4:**

```
>NDA
NDA>13JAN85,13JAN85/MEMSIZ:512=DK1:/ALL
```

This command string creates an analysis file named 13JAN85.LST and a save file named 13JAN85.CDA. The number of words of information to be transferred is 512K words. The input file is located on RK05 disk unit 1. NDA produces an analysis output listing using the /ALL analysis switch.

## 6.6 Indirect Command Files

You can enter command strings to NDA directly from the keyboard or indirectly through the use of the indirect command file facility. NDA indirect command files must not contain a reference to another command file.

## 6.7 NDA Switches

Switches provide you with the means to control NDA operation. There are two types: analysis control switches and function control switches.

Analysis control switches determine which NDA analysis routines are to be applied to the input file. Some switches apply only to the DECnet software, while others apply only to the PSI software running with DECnet on a DECnet-RSX system with a PSDN capability.

Function control switches provide a number of NDA control options. For example, such switches might terminate an analysis after NDA has encountered a specified number of errors. Or, a switch might limit the number of pages of output listing.

### 6.7.1 Analysis Control Switches

Table 6-2 provides a list of the analysis control switches and a brief description of their use. More than one analysis control switch can be specified. If none are specified, the default is */SYS/CEX*.

**Table 6-2: NDA Analysis Control Switches**

<b>Switch</b>	<b>Description</b>
/ALL	Executes the following switches (each is defined later in this table):  /CEX /LDTE /OBJ /PRO /RDTE /SYS /CTB /LIS /PDV /PSV /REM /TSK /CUG /NSP /PHB /PVC /ROU /XACP /DST /NW /PLI /QUE /SLT  DECnet, PSI, and X.25 Gateway components are analyzed only if the respective software was in the system that was dumped. /ALL is the recommended switch to use. If problems occur, then try using specific switches.
/CEX	Lists general Communications Executive data. This includes the node name and number, the current network process, and the current state of buffer usage.
/CTB	Prints information on all controller blocks assigned to network devices. (This is printed only on RSX-11M-PLUS systems.)
/CUG	Prints the PSI/X.25 Gateway closed user group name blocks.
/DST	Prints the PSI/X.25 Gateway X.25 and X.29 destination blocks.
/DMP: <i>a:b</i> /DUMP: <i>a:b</i>	Lists the contents of physical memory from octal address <i>a</i> through octal address <i>b</i> . Both <i>a</i> and <i>b</i> are 32-bit addresses.
/LDTE	Prints the PSI/X.25 Gateway local DTE descriptor blocks.
/LIS /LIST /LISTS /FREE	Scans the CCB, SDB, and RDB free lists for errors and prints the results.
/NSP	Prints the DECnet ECL databases, including the logical link blocks.
/NW	Prints the PSI NW: database and window blocks.

(continued on next page)

**Table 6-2 (cont.): NDA Analysis Control Switches**

<code>/OBJ</code> <code>/OBJECT</code>	Lists the network task object types.
<code>/PDV</code>	Prints interpreted contents of the process description vectors.
<code>/PHB</code>	Prints the contents of the PSI/X.25 Gateway PLI home block.
<code>/PLI</code>	Prints the PSI PLI/X.25 Gateway database and the X.25 circuit blocks.
<code>/PRO</code> <code>/PROC</code>	Dumps the line tables for all loaded network processes.
<code>/PSV</code>	Prints the X.25 Gateway PSV process databases, circuit databases, and buffer chains.
<code>/PVC</code>	Prints the PSI/X.25 Gateway permanent virtual circuit (PVC) name blocks.
<code>/QUE</code> <code>/QUEUE</code>	Lists the contents of the fork process queue.
<code>/RDTE</code>	Prints the PSI remote DTE descriptor blocks.
<code>/REM</code> <code>/REMOTE</code>	Lists the DECnet remote node names and alias names.
<code>/ROU</code> <code>/ROUT</code> <code>/ROUTING</code>	Lists the DECnet routing parameters, including the physical link blocks.
<code>/SLT</code>	Prints interpreted contents of the system line tables.
<code>/SYS</code>	Lists general system data. This includes the time of the crash, the current task, and the registers in use.
<code>/TSK</code> <code>/TASK</code>	Prints the network mailbox, window blocks, and I/O packets for each task using the network.
<code>/XACP</code>	Prints the CCBs and I/O packets waiting to be processed by the PSI X25ACP.

## 6.7.2 Function Control Switches

Table 6-3 provides a list of the function control switches and a brief description of their use. More than one function control switch can be specified. If switches are not specified, the default values listed in Table 6-3 are assumed. Note that when a switch requires a numeric argument, the argument is considered to be octal unless followed by a decimal point.

**Table 6-3: NDA Function Control Switches**

Switch	File	Description	Default
<code>/BL:<i>n</i></code>	Input file	Identifies the starting block number of the secondary storage device. The value of <i>n</i> must be less than 65535.  NDA reads the dump from the input device beginning at logical block <i>n</i> . If the crash dump device is not a disk or DECTape, NDA ignores this switch.	<code>/BL:1</code>
<code>/EXIT.<i>n</i></code>	Analysis file	Terminates analysis after encountering <i>n</i> analysis errors.  NDA maintains an error count. If you specify the <code>/EXIT:<i>n</i></code> switch, NDA terminates analysis after <i>n</i> errors. If you specify <code>/EXIT</code> but do not specify <i>n</i> , NDA exits after one error.	<code>/-EXIT</code>

(continued on next page)

**Table 6-3 (cont.): NDA Function Control Switches**

/KMR	Input file	When NDA reads incorrect page address register (PAR) values from the crash stack, it prints an error message on the user's terminal and attempts to continue with the analysis. In this event, you can use the /KMR switch when you restart the analysis.	/KMR forces NDA to use standard mapping values to convert kernel virtual addresses to physical memory addresses.
/LIMIT: <i>n</i>	Analysis file	Limits the number of pages of analysis output. When NDA has generated <i>n</i> pages, it terminates the analysis and prints a message on the user terminal indicating that it has done so.	/LIMIT:50

(continued on next page)

**Table 6-3 (cont.): NDA Function Control Switches**

<code>/MEMSIZ:n</code>	Save file	Creates a save file $4 \times n$ blocks long and transfers $nK$ words to it from the input file. The valid range of decimal values for $n$ is from 16 to 4096.	<code>/MEMSIZ:124</code>
		This switch is necessary only when saving dumps from secondary storage devices that have no EOFs (disk or DECTape). When the crash input resides on magtape, the save file is zero filled if the EOF is read before $nK$ words have been transferred.	
<code>/-SP</code>	Analysis file	Causes NDA to not spool the analysis output listing file to the line printer.	<code>/SP</code>

(continued on next page)

**Table 6-3 (cont.): NDA Function Control Switches**

<code>/STB</code>	Symbols file	Identifies the file that contains both the executive and the network symbol tables for the crashed system. NDA opens the specified file and extracts the necessary symbol values. If it fails to find any required symbol values, NDA aborts the analysis and prints an error message on the terminal.	CEX.STB in the current UIC for RSX-11M; RSX11M.STB in the current UIC for RSX-11M-PLUS
		If no symbol is specified via the <code>/STB</code> switch, CEX.STB under the current UIC is used.	
<code>/SYM</code>	Input file	If any required symbols are not found in the symbol table file, set their values to zero and proceed with the analysis. Normally, this condition would cause NDA to abort the analysis.	<code>/-SYM</code>

## 6.8 NDA Error Messages

NDA prints an error message on your terminal when it detects an error condition. NDA error messages are summarized in Chapter 10. Do not confuse these messages with the diagnostic analysis messages that NDA generates during the analysis and prints in the analysis listing.

## Network Display Program (NTD)

The Network Display Program (NTD) can provide three types of displays:

- A **resource display** of the current status of the host node or any specified remote RSX/IAS DECnet node in the network. There are two formats for this display:

**Default resource display** for current DECnet-RSX systems (see Example 7-1)

**Old resource display** for connections to DECnet-RSX systems having previous versions of the software (see Example 7-2)

- A **node summary display** of all reachable nodes or a user-specified range of nodes in the network (available on routing nodes only -- see Example 7-3)
- An **area summary display** of all reachable areas in the network (available on level 2 routing nodes only -- see Example 7-4)

You can review a continuously updated screen display on any VT100, VT52, VT220, or VT240 terminal, or you can use NTD with a hard-copy terminal to obtain a listing of the display information. The displayed information is collected by NTDEMO, a task that runs on the system for which you are displaying information.

This chapter describes the displays and outlines the commands required to invoke and customize the displays.

## 7.1 Invoking NTD

Use the following command to invoke NTD:

```
NTD [node-name][/PAGE = s]
```

where

*node-name* is a name consisting of 1 to 6 numeric or alphabetic characters, including at least 1 alphabetic character. If you do not specify *node-name*, NTD displays the status of the local node.

*/PAGE = s* is a switch that specifies the display type, where *s* is one of the following:

- R Default resource display
- O Old resource display
- N Node summary display
- A Area summary display

This switch is most useful when running NTD from a hard-copy terminal. Without this switch, you will always get a resource display. NTD automatically checks the version of the specified node to determine whether to display the default resource display or the old resource display. From a VT100 or a VT52, you can switch to a node summary display simply by entering <ESC>.

Error messages that can be generated when you invoke NTD are listed in Chapter 10.

## 7.2 NTD Commands

NTD commands control various aspects of the real-time NTD displays. There are two modes of operation for NTD commands:

- **Immediate mode.** With the NTD display on your screen, you enter commands to directly manipulate the display -- for example, to control scrolling of display information or to change to a different display.
- **Command mode.** When you switch from immediate mode to command mode, NTD displays the commands that you can enter to customize the display -- for example, to change the frequency of display updates or the range of nodes for which NTD is to display information. In this mode, NTD prompts for commands. You must return to immediate mode to view the displays.

The use of each mode and its commands is defined in the following sections.

### 7.2.1 Immediate Mode Commands

When you invoke NTD, you automatically enter immediate mode. Enter any of the commands described in Table 7-1 to directly manipulate the screen display. The terminal's bell will ring if you enter an unrecognized command.

**Table 7-1: Immediate Mode Commands**

<b>Character Key</b>	<b>Function</b>
R	Changes to default resource display
O	Changes to old resource display
N	Changes to node summary display
A	Changes to area summary display
<space bar>	Refreshes the currently selected display
D	Selects scrolling of node/circuit information only (default)
T	Selects scrolling of task information only

(continued on next page)

**Table 7-1 (cont.): Immediate Mode Commands**

B	Selects simultaneous scrolling of node and task information
+	Scrolls currently selected information forward 5 items at a time for resource displays or 10 nodes at a time for node summary displays *
-	Scrolls currently selected information backward 5 items at a time for resource displays or 10 nodes at a time for node summary displays *
S	Sorts selected information in ascending order by node address instead of alphabetically by node name
H	Displays help information for immediate mode commands for 5 seconds, then redisplay the currently selected NTD status information
<ESC>	Enters command mode
<CTRL/Z> or <CTRL/C>	Exits NTD

-----

\* Scrolling causes information to be displayed for a range of nodes, beginning with the next available node that is five reachable addresses ahead of (or behind) the currently specified starting node. On the node summary display, a maximum of 84 nodes can be displayed on the screen at once. Unless you enter the S command (see above), nodes are displayed in alphabetic order by name. Therefore, scrolling by node address will cause the order to change.

## 7.2.2 Command Mode Commands

When you enter <ESC> to switch from immediate mode to command mode, the possible commands you can enter are displayed on your screen, followed by a command prompt. Valid commands are described in Table 7-2. Enter one command for each prompt.

**Table 7-2: Command Mode Commands**

<b>Command</b>	<b>Function</b>
NTIME = <i>n</i>	Sets the time (in seconds) between display updates for remote nodes (default: 10 seconds)
TTIME = <i>n</i>	Sets the time (in seconds) between display updates for network tasks and circuits (default: 5 seconds)
ADDRESS = <i>n</i>	Sets the lowest node address for which information is to be displayed (range: 1 to 1023; default: node 1) †
MAXIMUM = <i>n</i>	Sets the highest node address for which information is to be displayed (range: 1 to 1023; default: node 1023) †
HOPS = <i>n</i>	Sets the maximum number of hops allowed to a remote node to be displayed (default: 31) †
COST = <i>n</i>	Sets the maximum cost allowed for a remote node to be displayed (default: 1023) *
<RET>	Reverts to the currently selected display and immediate mode
<CTRL/Z>	Exits NTD

-----  
\* If you specify 0 for *n*, the value will revert to the specified default.

When entering these commands, you need only use a single character to designate the command. For example, the commands ADDRESS=4 and A=4 both set 4 as the lowest node address for which information is to be displayed.

## 7.3 NTD Resource Display Format

Examples 7-1 and 7-2 are examples of the default resource display and the old resource display, respectively. The display information fields are described below.

### Top of the screen

- The current date and time at the node where NTDEMO runs. This is updated approximately every second.
- The name and identification of the node being displayed. For VT100 terminals, NTD uses the VT100 graphics capability to highlight the name of the node.
- Bar graphs for the following resources:

Logical links (LNKS)

Communication control buffers (CCBs)

### NOTE

Because CCBs can be allocated dynamically, the total number of CCBs available can vary.

Small data buffers (SDBs)

Large data buffers (LDBs)

The first set of numbers for each resource indicates the number of that resource currently in use and the total number generated into the system. For example, 12/50 in the LDB line in Example 7-1 indicates that 12 out of 50 generated large data buffers are currently being used.

The actual graphs represented by Xs in the examples illustrate the percentage of each resource currently in use. The vertical bars on the graph (represented by ! in the examples) illustrate the highest percentage of each resource used during the current invocation of NTD. For example, in Example 7-1, roughly 25 percent of the LDBs are currently in use, but the position of the ! shows that approximately 50 percent of the generated buffers have been in use at once during the current invocation of NTD. Bar graphs for each resource are scaled independently; for example, an X on one graph may represent usage of one of that resource, while an X on another graph may represent usage of five of that resource. Actual graphic representations can differ depending on terminal type.

At the end of the line, NTD displays the number of allocation failures for each resource type since network startup. On the LDB line, the second allocation failure field refers to receive data buffers.

The bar graph display is updated approximately every second.

**Lower left of the screen.** This data differs with the two resource display types, as described below. For either type, this information is updated at the interval specified by the NTIME command (default: every 10 seconds).

- Default resource display (see Example 7-1). NTD displays the following information for all known circuits:

The circuit name.

The cost to reach the adjacent node.

The maximum buffer size of network messages to the adjacent node (not applicable for Ethernet circuits, since each node on the Ethernet may have a different size).

For point-to-point circuits, the name and address of the adjacent node on the circuit, if it is reachable; otherwise, the status of the circuit.

- Old resource display (see Example 7-2). NTD displays the following information for any node that is currently reachable:

The remote node name and its node address.

The name of the circuit over which packets are sent to the node.

The minimum number of hops required to reach the node and the minimum cost to reach the node. These numbers may correspond to different routes, as the least costly route may not necessarily have the fewest hops.

The number of active logical links to that node and the current round trip delay in seconds. If the number of logical links is shown as 0, it indicates that a link has been active to that node since the network was loaded.

**Lower right of the screen.** NTD displays the following information for currently active network tasks (that is, those tasks that have issued an OPNS\$ to the network):

- The task name.
- The TI: device associated with the task. Tasks requested through the network normally run with a TI: of CO0: for RSX-11M and M-PLUS.
- The number of logical links that are active on the task, including any outgoing connect request that has not yet completed.
- The number of entries on the task's mailbox. Entries can include:

Incoming connect requests

Interrupt messages

Disconnect notifications

Link abort notifications

Network abort notifications

Network events

Refer to the *DECnet-RSX Programmer's Reference Manual* for more information about mailboxes.

- The number of outstanding logical link transmit requests.
- The number of outstanding logical link receive requests.

This information is updated at the interval specified by the TTIME command (default: every 5 seconds).

**Example 7-1: Sample Default Resource Display**

```

6-MAR-84 17:20:25 Node: BIGBOY(4.8) FINANCE SYSTEMS ROUTER
                RSX-11M-Plus, Area Routing Node, DECnet V2.0
LNKS   3 / 10 XXXXXXXXXXXXX!----- Alloc Fails:
CCB   30 /100 XXXXXXXXXXXXX!-----           0
SDB    6 / 22 XXXXXXXXXXXX!-----           0
LDB   23 /130 XXXXXXXX!-----           0 :   0

                Known Circuits:
Circuit  Cost Size Adj Node/Status          Task  TI Links MBX XMT RCV
UNA-0     3      ACCNTS( 4.126)      NTD... COO:   2  0  1  0
UNA-1     3      Off                NTDT1  TT1:   1  0  0  1
DMC-0    11      On-starting
DMC-1    11      On-starting
DMC-2    11      On-starting
DMC-3    11      On-starting
DMC-4     4      On-starting
DMC-5     4      On-starting
    
```

**Example 7-2: Sample Old Resource Display**

```

6-MAR-84 17:20:43 Node: BIGBOY(4.8) FINANCE SYSTEMS ROUTER
                RSX-11M-Plus, Area Routing Node, DECnet V2.0
LNKS   3 / 10 XXXXXXXXXXXXX!----- Alloc Fails:
CCB   29 /100 XXXXXXXXXXXXX!-----           0
SDB    6 / 22 XXXXXXXXXXXX!-----           0
LDB   22 /130 XXXXXXXX!-----           0 :   0

                Remote Nodes:
Node      Circuit Hop/Cst Lnk/Dly          Task  TI Links MBX XMT RCV
TULSA ( 4.6)  UNA-0  2/6   0/10      NTD... COO:   2  0  1  0
MIAMI ( 4.10) UNA-0  1/3   0/10      NTDT1  TT1:   1  0  0  1
BOSTON( 4.11) UNA-0  2/6
HARTFD( 4.17) UNA-0  1/3   0/5
NEWYRK( 4.18) UNA-0  3/15  0/4
SEATTL( 4.19) UNA-0  1/3   1/4
CHICAG( 4.21) UNA-0  1/3
DENVER( 4.22) UNA-0  2/7
SANFRN( 4.17) UNA-0  2/6
    
```

## 7.4 NTD Node Summary Display Format

Example 7-3 is an example of a node summary display. The display information fields are described below.

### Top of the screen

- The current date and time at the node where NTDEMO runs. This is updated approximately every second.
- The name and identification of the node being displayed. For VT100 terminals, NTD uses the VT100 graphics capability to highlight the name of the node.

### Node entries

- Node name or address. By default, named nodes are listed alphabetically, followed by nodes that are identified by address only (in ascending order). You can use the immediate mode S command to order all nodes by ascending node address (see Table 7-1).
- The minimum number of hops required to reach the node and the minimum cost to reach the node. These numbers may correspond to different routes, as the least costly route may not necessarily have the fewest hops.
- The number of active logical links to the node. If the number of logical links is shown as 0, this indicates that a link has been active to that node since the network was loaded.
- The total number of nodes actually displayed. Since the screen cannot display more than 84 nodes at once, this number may not necessarily reflect the total number of nodes in a large network. You will have to use scrolling to display additional nodes. This number also does not include any nodes that have been excluded because of filtering commands (see ADDRESS, MAXIMUM, HOPS, and COST in Table 7-2).

### Example 7-3: Sample Node Summary Display

```
6-MAR-84 17:20:43 Node: BIGBOY(4.8) FINANCE SYSTEMS ROUTER
                RSX-11M-Plus, Area Routing Node, DECnet V2.0
Node Hop/Cst Lnk Node Hop/Cst Lnk Node Hop/Cst Lnk Node Hop/Cst Lnk
ATLANT 4/20      DENVER 4/20  0  OMAHA  1/3      [ 21] 2/6
ALBERQ 5/24      FRISCO 2/6      QUEBEC 4/20  1  [ 25] 4/20
BOSTON 2/6        1  HARTFD 5/22      RICHMN 1/3      [ 16] 3/15
CHICAG 5/23      JACKVL 1/3      SEATTL 4/20  0  -Total Nodes: 19
DALLAS 5/23      2  NEWYRK 4/20  3  TULSA  4/20
```

## 7.5 NTD Area Summary Display Format

Example 7-4 is an example of an area summary display. The display information fields are described below.

### Top of the screen

- The current date and time at the node where NTDEMO runs. This is updated approximately every second.
- The name and identification of the node being displayed. For VT100 terminals, NTD uses the VT100 graphics capability to highlight the name of the node.

### Area entries

- Area ID. The ID of the area.
- The circuit name of the circuit being used to communicate with the area.
- The minimum number of hops required to reach the area and the minimum cost to reach the area. These numbers may correspond to different routes, as the least costly route may not necessarily have the fewest hops.
- The next node on the path to the area. This node may be another level 2 router within the local area or a level 2 router in another area. In either case, it should always be a level 2 router if the network was configured properly.
- The total number of areas actually displayed.

### Example 7-4: Sample Area Summary Display

```
6-MAR-84 17:20:43 Node: BIGBOY(4.8) FINANCE SYSTEMS ROUTER
                RSX-11M-Plus, Area Routing Node, DECnet V2.0
Area Circuit Hop/Cst  Next Node          Area Circuit Hop/Cst  Next Node
 2 UNA-0         4/20 DALLAS( 4.12)    20 DMC-0         2/6  OMAHA ( 4.18)
 3 UNA-0         3/15 DALLAS( 4.12)    21 UNA-0         1/5  DALLAS( 4.12)
 5 DMC-1         2/9  HARTFD( 4.8)    25 UNA-0         2/9  DALLAS( 4.12)
10
```

## Queue Manager (QUE)

The Queue Manager (QUE) utility is the interface to the RSX-11M V4.2, RSX-11M-PLUS V3.0, and Micro/RSX V3.0 queue manager. Any privileged user can employ QUE to initialize the File Transfer Spooler (FTS) queue and processor and to manipulate user-queued FTS requests. FTS is a DECnet-RSX utility that allows you to transfer files in much the same way as the Network File Transfer (NFT) utility, with the additional queue feature.

This chapter outlines components required by FTS and QUE and describes how to invoke and use QUE. For more information on both utilities, including error messages, see the following manuals:

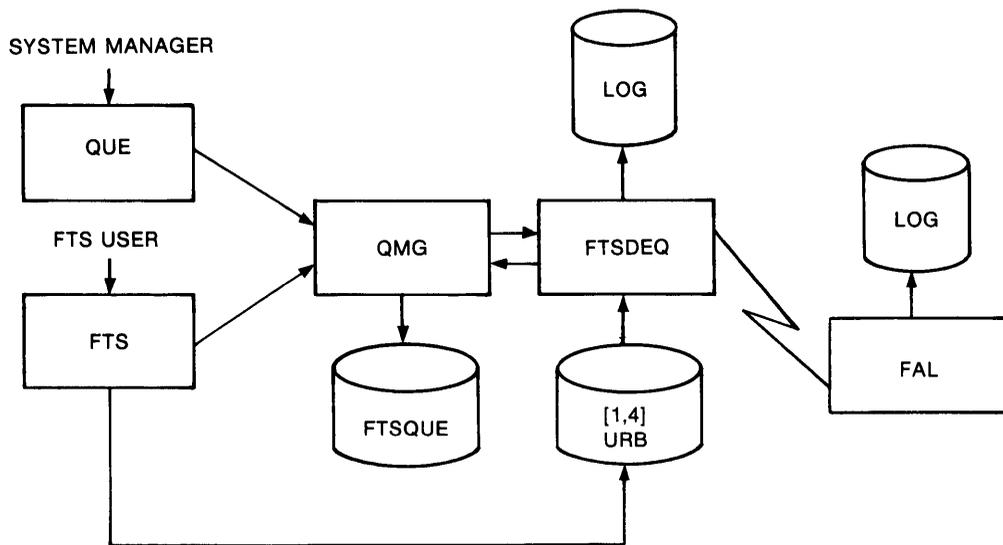
- *DECnet-RSX Guide to User Utilities*. Contains a complete description of FTS
- *RSX-11MIM-PLUS Batch and Queue Operations Manual*. Contains a description of QUE user commands and error messages
- *RSX-11MIM-PLUS System Management Guide*. Contains a description of QUE system manager commands

### 8.1 FTS and QUE Components

Table 8-1 lists the components required by FTS and QUE and describes their relationship to queue management. Figure 8-1 diagrams the relationships among these components.

**Table 8-1: FTS and QUE Components**

<b>Component</b>	<b>Description</b>
FTS	The user interface that spools requests and places them on a queue.
QUE	The system manager interface that processes queue management commands.
QMG	The RSX-11 queue manager task.
FTSDEQ	The processor task that services the queue.
FAL	The File Access Listener that handles file access and transfer requests over a line at the remote node.
FTSQUE	The queue of FTS jobs that the queue manager stores in a disk area.
URBs	User request block files that contain details of the FTS requests corresponding to FTS jobs on the queue. This information contains names that FTSDEQ needs to process the requests. FTS prints this information in response to the /LIST command.
FTSSYS.LOG	The log file that contains a record of both completed and uncompleted FTS jobs.
FAL.LOG	The log file that contains a record of completed FAL jobs.



TW145

Figure 8-1: FTS and QUE Components

## 8.2 Invoking QUE

To invoke QUE, enter the following command in response to an MCR> prompt:

*QUE command*

where *command* is any valid QUE command (as described in the following sections). After the QUE command executes, you are returned to MCR.

## 8.3 QUE Command Syntax

QUE supports three command formats. Each format allows you to perform various queue management functions. These functions include:

- Initializing/deleting and starting/stopping the queue and the processor
- Assigning/deassigning the processor to/from the queue
- Holding, releasing, or deleting FTS user jobs on the queue

The following sections describe the three command formats and their functions and give examples of their use.

### 8.3.1 Initializing/Deleting and Starting/Stopping the Queue and the Processor

Use the following QUE command syntax and the appropriate switch to cause the queue or the processor to be initialized, deleted, started, or stopped:

QUE *name/switch*

where

*name* is either the queue name (FTSQUE) or the processor's installed name (FTSDEQ).

*switch* is one of the following switch options:

CR	creates, names, and starts the FTS queue
SP/EX	creates, names, and starts the FTS processor
STA	starts the FTS queue or processor
STO:QUE	stops the FTS queue
STO	stops the FTS processor
DEL:Q	deletes the FTS queue
UNSP	deletes the FTS processor

In an emergency situation, delete either the queue or the processor. Users can delete their jobs from the queue using FTS.

#### NOTE

If FTS was selected during NETGEN, the FTS queue and processor will already be set up when the network is loaded. If they were not set up by commands in file NETINS.CMD, you must create the queue and start the processor to make the FTS queue available for users.

#### Example

```
MCR>QUE FTSQUE/CR
MCR>QUE FTSDEQ/SP/EX
```

The first command creates the FTS queue. The second command starts the FTS processor task.

### 8.3.2 Assigning and Deassigning the Processor

Use the following QUE command syntax to assign and deassign the processor to and from the queue:

```
QUE FTSDEQ:switch:FTSQUE
```

where

FTSDEQ is the processor name.

FTSQUE is the queue name.

*switch* is one of the following switch options:

AS assigns the processor to the queue. You must assign the processor upon initializing it in order to enable FTS operations.

DEA deassigns the processor from the queue. Deassigning the processor prevents it from performing any functions (for example, transferring files, despooling). However, jobs can be added to the queue when the processor is deassigned. You may need to deassign the processor in special situations (for example, to prevent jobs from being completed). To reactivate the processor, use the assign switch (/AS).

There are cases where a queue must be inspected, but you do not want any of the jobs processed before listing the queue's contents. In this case, deassign the processor before listing the queue. Refer to Section 8.4 for information on related queue management functions.

### 8.3.3 Holding, Releasing, and Deleting FTS User Jobs

Use the following QUE command syntax to hold, release, or delete FTS user jobs on the queue:

```
QUE /ENT:entry-number/switch
```

where

*entry-number* is the number assigned to the queue entry by the queue manager. The entry number is obtained by listing the contents of FTSQUE (see Section 8.3.2).

*switch* is one of the following switch options:

HO	holds the specified job
REL	releases the specified job from the hold state
DEL	deletes the specified job

Ordinarily, you do not delete or hold jobs. FTS users can delete their own jobs. You need to use these commands only in special situations (for example, when you want to delay a job that is about to begin).

#### Example:

```
MCR>QUE /ENT:857/DEL
```

This command deletes the job with entry number 857 from the queue.

## 8.4 Related Queue Management Functions

In addition to using QUE for queue management, you should periodically perform other related queue management functions. These functions include:

- Purging the FTS system log file (FTSSYS.LOG) and the FAL system log file (FAL.LOG)
- Listing the contents of FTSQUE and of FTS jobs
- Listing and purging the user request block (URB) files

The following sections describe procedures for performing these functions.

### NOTE

FTS will retry requested file operations if they fail due to network problems. After a number of unsuccessful retries (default = 5), the FTS job will be put into the hold state in FTSQUE. The user must decide whether to release the job for one more retry or to delete (abort) it.

#### 8.4.1 Purging FTSSYS.LOG and FAL.LOG

Periodically, you should purge the FTS system log file (FTSSYS.LOG) and the FAL system log file (FAL.LOG). FTSSYS.LOG contains a record of every job FTS has received, along with coded status information, time of receipt, and time of completion. The file is useful for statistical purposes. FAL.LOG records network file transfer information for each file copied over the network.

FTSDEQ automatically deletes URB file entries as jobs complete. However, it does not delete FTS system log file entries. Use the Peripheral Interchange Program (PIP) to purge these files when necessary. (See the *RSX-11MIM-PLUS Utilities Manual* for more information.) The following PIP command displays the FTS system log file:

```
>PIP TI:=[1,4]FTSSYS.LOG
```

Refer to the *DECnet-RSX Guide to User Utilities* for a description of the system log file display.

During network generation, you can choose to create the system log file in UFD [1,4] with no file protection. Optionally, log entries are copied to a log file in the UFD of the user who queued the request. The user is expected to maintain this file.

#### 8.4.2 Listing FTSQUE and FTS Jobs

Periodically, or when necessary, list the contents of FTSQUE and of FTS jobs. The following command lists the contents of FTSQUE:

```
MCR>QUE FTSQUE:/LI:P
```

A sample system display follows:

```
  ** PRINT QUEUES **  
  FTSQUE =>FTSDEQ  
[1,1]      223771  ENTRY:857          PRINT AFTER 2-FEB-85 11:37  
          1 DBO:[1,4]FTS4166.URB;1
```

The FTSQUE contains one job, number 223771. Note that FTSDEQ is the processor for FTSQUE. This field appears only when the processor has been assigned to the queue. If this field does not appear, the processor is deassigned, but there can still be entries in the queue. The job was originally queued from the UIC [4,166]. After it failed, the job was requeued by the FTSDEQ task from UIC [1,1]. The queue manager entry number is 857. The job is to be printed (in this case retried) at 11:37.

Note the format of the URB file specification:

```
device:[ufd]FTSnnnnnn.URB;xx
```

where

*nnnnnn* is the user's current UIC (with leading zeroes suppressed).

*xx* is the request number. If multiple jobs are queued from a single UIC, this number is greater than 1.

If, after inspecting the FTS queue, you want more information on a user's job on the queue, enter the following commands:

```
>SET /UIC=[user uic]
>FTS /LI
```

The following is a sample system display:

```
FTS USER REQUEST LISTING                                2-FEB-85 11:45

223553  TLR/50                                           /IM/LO
  Queued 2-FEB-85 11:45                                   Retried 0 times
  Source File -- DBO:[4,122]NODES.SYS;4
  Destination File -- XYZ::[4,131]NODES.SYS

223573  TLR/50                                           /IM/LO
  Queued 2-FEB-85 11:45                                   Retried 0 times
  Source File -- DBO:[4,122]NODFIL.MAC;10
  Destination File -- XYZ::[4,131]NODFIL.MAC

223507  TLR/50                                           /IM/LO
  Queued 2-FEB-85 11:45                                   Retried 0 times
  Source File -- DBO:[4,122]PRMFIL.MAC;1
  Destination File -- XYZ::[4,131]PRMFIL.MAC
```

The job numbers at the beginning of each entry are the only cross-reference you have to the system queue list. For more information on the FTS /LI switch, see the *DECnet-RSX Guide to User Utilities*.

### 8.4.3 Listing and Purging URB Files

Periodically, or when necessary, list and purge the URB files. The following PIP command produces a display of the URB directory:

```
>PIP [1,4]*.URB;*/LI
```

If you have deleted a job with QUE, you must also delete its associated URB file or files. QUE does not do this automatically. However, when an FTS user deletes a job, it also is deleted automatically from its URB file.



## Trace Interpreter Task (TRI)

The X.25 trace interpreter task (TRI) is part of the trace facility provided for RSX-11 PSI users. This facility enables you to diagnose software problems on a PSI line connected to a device. The facility traces message frames passing between the X.25 level 2 protocol and the device driver and copies the traced frames to a disk file. The contents of this file can then be analyzed and printed.

Use NCP TRACE commands to control the tracing of frames and to create the file (see the CLEAR TRACE, SET TRACE, and SHOW TRACE commands in Chapter 1). Then use TRI to analyze the contents of the trace disk file.

### 9.1 Invoking TRI

There are three methods of invoking TRI:

1. MCR> TRI

TRI then prompts

```
TRI>
```

Enter your TRI command immediately following the prompt. If the command executes, the utility prompt is displayed on the next line. If the command is unsuccessful, an error message is displayed on the next line indicating the reason for the error followed by the utility prompt.

## 2. MCR>TRI *command*

where *command* is any valid TRI command. With this method, you return to MCR after the TRI command is executed.

## 3. MCR>TRI @*command-file*

where *command-file* is the name of an indirect command file containing TRI commands. You can nest command files.

## 9.2 Exiting TRI

To exit TRI, enter <CTRL/Z> or <RET>.

## 9.3 TRI Command Format

The TRI command line has the following format:

```
[outfile[/switches]] = [[infile[/switches]]]
```

where

*outfile* is the listing file. If not specified, this defaults to the input file name with file type .LST.

*infile* is the name of the file containing the tracing data. If this name is not specified, this file defaults to LB:[1,6]PSITRACE.SYS.

*switches* are the options available for controlling the form and content of the listing produced by TRI. Any compatible combination of the switches shown in Table 9-1 can be specified on either the input file or the output file.

**Table 9-1: Trace Interpreter Switches**

<b>Switch</b>	<b>Meaning</b>	<b>Comments</b>
<code>/SP</code>	Spool the output file	Default switch. Output is spooled unless <code>/-SP</code> is specified.
<code>/-SP</code>	Do not spool the output file	
<code>/OC *</code>	Output user data in octal	Cannot be specified with <code>/-IN</code> .
<code>/AS *</code>	Output user data in ASCII	Cannot be specified with <code>/-IN</code> .
<code>/IN</code>	Interpret trace data	Default switch. (Data is interpreted unless <code>/-IN</code> specified.)
<code>/-IN</code>	Do not interpret trace data	Only <code>/SP</code> or <code>/-SP</code> can be specified with this switch.
<code>/LI:n</code>	Output trace data for system line <i>n</i> only	The <i>n</i> is octal. If you want to specify a decimal number, use <i>n.</i> instead. The default for <i>n</i> is 0. Cannot be specified with <code>/-IN</code> .
<code>/CH:n</code>	Output trace data for channel number <i>n</i> only	The <i>n</i> is octal. If you want to specify a decimal number use <i>n.</i> instead. The default for <i>n</i> is 0. Cannot be specified with <code>/-IN</code> .

-----

\* If you specify both `/OC` and `/AS`, trace data is printed in octal bytes with the corresponding ASCII characters underneath.

## Examples:

```
MCR> TRI =/OC/AS
```

This command uses the /OC and /AS switches to output user data in octal and ASCII characters.

```
MCR> TRI =/LI:1/OC
```

This command uses the /OC switch to output octal user data for system line 1 only.

## 9.4 Contents of Trace Output

TRI reads the trace file created by the tracing facility and lists the following information (depending upon the frame type) for each traced frame:

- Any error status from the CCB
- The system line number
- The direction of transfer (Rx or Tx)
- The frame address byte
- Setting of the P/F bit
- N(S) and/or N(R) numbers
- The frame type as text (for example, SABM)
- The logical channel identifier
- Settings of the Q, D, and M bits
- The packet type as text (for example, Call Req.)
- P(S) and/or P(R) numbers
- Cause and diagnostic fields
- The first few bytes of user data

## 9.5 TRI Error Messages

TRI prints an error message on your terminal when it detects an operating error. All TRI error messages are listed in Chapter 10.



# 10

## Utility Error Messages

This chapter lists the error messages sent by the following utility programs:

- Console carrier requester (CCR)
- Configuration File Editor (CFE)
- Event File Interpreter program (EVF)
- Host Task Loader (HLD)
- Network Control Program (NCP)
- Network Crash Dump Analyzer (NDA)
- Network Display Program (NTD)
- Network initializer (NTINIT)
- Network loader (NTL)
- X.25 trace interpreter task (TRI)
- Virtual Network Processor (VNP)

You would not be surprised to receive a CFE error message if you had just invoked a CFE command. However, NCP, VNP, and CFE use the NTINIT and NTL utilities to process your commands. Therefore, an error could occur in one of those utilities instead of in the one that you invoked. You will receive an error message from the utility in which the error occurred. Each error message has a prefix that identifies the utility that sent it.

In this chapter, error messages are arranged in alphabetical order by utility and message text. Variable fields within messages are ignored for purposes of alphabetization.

## 10.1 CCR Error Messages

### **CCR -- Component in wrong state, *xxx***

The specified component was not in the correct state to proceed with the operation of console carrier (*xxx* = circuit or system).

### **CCR -- File I/O error, *xxx***

An I/O error occurred while trying to read the specified file File I/O (*xxx* = secondary loader or load file).

### **CCR -- File open error, *xxx***

CCR was unable to open the specified file (*xxx* = secondary loader or load file).

### **CCR -- Invalid file contents, *xxx***

The contents of the specified file are not in the required format (*xxx* = secondary loader or load file).

### **CCR -- Invalid identification format, *xxx***

The format for the specified parameter is invalid (*xxx* = node, service circuit, hardware address, or service password).

### **CCR -- Line communication error**

The operation failed due to a communication error on the line.

## **CCR**

---

### **Error Messages**

#### **CCR -- Line protocol error**

The operation failed due to a protocol error.

#### **CCR -- Management programming error**

An internal programming error has been detected.

#### **CCR -- Message format error**

The format of a received protocol message was invalid.

#### **CCR -- Parameter missing, *xxx***

The required parameter was not defined in the command line and the down-line load database (*xxx* = node, service circuit, hardware address, or service password).

#### **CCR -- Privilege violation**

The user does not have sufficient privileges to run CCR.

#### **CCR -- Protocol/address pair in use**

The protocol/address pair used to communicate with the target system is already in use.

#### **CCR -- Remote console already in use**

Another user on another node has already reserved the remote console on the target system.

## 10.2 CFE Error Messages

All CFE error messages documented in this chapter are classified according to severity and cause. Both classifications are described below and are noted at the beginning of each error description.

### Severity classifications

Informational	There is a minor problem, but it does not affect the task's exit status.
Warning	Minor error. Processing continues.
Error	Major error. The command was accepted, but failed to execute completely. The system waits for a new command.
Severe Error	The command was not accepted. Processing stops and control returns to the monitor.

### Cause classifications

CETAB Parse	These errors occur during the parsing of CETAB.MAC. They usually occur when someone has edited CETAB using a text editor. Restore CETAB to its original content.
CETAB	These errors are generated in response to valid commands. They usually occur when someone has edited CETAB using a text editor. Restore CETAB to its original content.
Configuration	These errors are generated by valid commands that refer to features or components that are not in the CETAB file being edited. Check the CETAB file and enter the correct command.
Command	These errors usually are caused by a mistyped command or by an incorrect or inapplicable parameter. Correct the command and reenter it.
General	These errors usually indicate a problem with the CFE installation or with the HELP or CETAB files. The cause is probably an invalid UIC or something similar.

## **CFE**

---

### **Error Messages**

#### **CFE -- Access verification not supported**

(Error/Configuration) You do not have access verification supported in your system, so you cannot change the verification state.

#### **CFE -- Bad port value**

(Severe Error/CETAB Parse) The number of ports on the PSN\$SDF macro is out of range or contains nonnumeric characters.

#### **CFE -- Bad transport subaddress value**

(Severe Error/CETAB Parse) The transport subaddress values on the PSN\$SDF macro are out of range or contain invalid characters.

#### **CFE -- Buffer allocation failure**

(Error/Command) CFE tried to allocate an internal buffer to complete the command but failed. If you must make a new entry in CETAB, then remove CFE and reinstall it with an increment.

#### **CFE -- Call mask and call value are different lengths**

(Error/Command) The call mask and call value of a destination must be of the same length.

#### **CFE -- Circuit not in configuration**

(Error/Configuration) The circuit in the command is not in the database.

#### **CFE -- Command line error**

(Error/Command) The command has a bad character or equivalent.

#### **CFE -- Consistency error**

(Severe Error/General) There is an internal consistency error in CFE. Please contact your software specialist to determine the cause of this error.

**CFE -- Destination not in configuration**

(Error/Configuration) The destination in the command is not in the database.

**CFE -- DTE not in configuration**

(Error/Configuration) The DTE in the command is not in the database.

**CFE -- DTE number missing from outgoing SVC: DLM-*u.t***

(Severe Error/CETAB Parse) The DTE number is required for outgoing SVCs. Reenter the command with the DTE number.

**CFE -- DTE with DLM circuits not purged: *DTE-address***

(Error/Command) The indicated DTE was not purged because some of its circuits are data-link-mapping circuits. Data-link-mapping circuits may not be purged.

**CFE -- Duplicate PSN\$DF in CETAB**

(Severe Error/CETAB Parse) There is more than one PSN\$DF macro in the CETAB.MAC file. Remove one of the PSN\$DF macros.

**CFE -- Error in call mask or call value**

(Error/Command) The format in the call mask or call value in the command is invalid.

**CFE -- Error in channel list**

(Error/Command) The format of the channel list in the command is invalid.

**CFE -- Error in channel number**

(Severe Error/CETAB Parse) The channel number on a CHN\$DF macro is out of range or contains nonnumeric characters.

## **CFE**

---

### **Error Messages**

#### **CFE -- Error in controller number**

(Severe Error/CETAB Parse) The controller number on an SLT\$DF macro is invalid.

#### **CFE -- Error in DDM name**

(Severe Error/CETAB Parse) The DDM name on an SLT\$DF macro is too long.

#### **CFE -- Error in DLC name**

(Severe Error/CETAB Parse) The DLC name on an SLT\$DF macro is too long.

#### **CFE -- Error in DTE address**

(Severe Error/CETAB Parse) The DTE address field on an SVC\$DF macro is out of range.

#### **CFE -- Error in flags**

(Severe Error/CETAB Parse) The flags field on a CUG\$DF, PSN\$DF, PVC\$DF, SLT\$DF, or SVC\$DF macro is unrecognized or invalid.

#### **CFE -- Error in line ID**

(Severe Error/CETAB Parse) The line identification on a DTE\$DF macro is of an invalid format.

#### **CFE -- Error in line type**

(Severe Error/CETAB Parse) The line type on an SLT\$DF macro is unrecognized.

#### **CFE -- Error in LLC name**

(Severe Error/CETAB Parse) The LLC name on an SLT\$DF macro is too long.

**CFE -- Error in maximum circuits**

(Severe Error/CETAB Parse) The maximum circuits value on an X2PSDF macro is out of range or has an invalid format.

**CFE -- Error in maximum recalls**

(Severe Error/CETAB Parse) The maximum recall field on an SVC\$DF macro is out of range.

**CFE -- Error in network name**

(Severe Error/CETAB Parse) The network name on a DTE\$DF or PSN\$DF macro is too long or contains invalid characters.

**CFE -- Error in opening file**

(Severe Error/General) CFE could not open the specified CETAB source file. Make sure that the file is where you specified, that the file is not locked, and that you have write privileges to the file.

**CFE -- Error in owner**

(Severe Error/CETAB Parse) The owner process field on an SLT\$DF macro is unrecognized.

**CFE -- Error in packet size**

(Severe Error/CETAB Parse) The packet size field on an SVC\$DF or X3PSDF macro is out of range.

**CFE -- Error in recall timer**

(Severe Error/CETAB Parse) The recall timer value field on an SVC\$DF macro is out of range.

## **CFE**

---

### **Error Messages**

#### **CFE -- Error in SVC name**

(Severe Error/CETAB Parse) The SVC circuit ID on an SVC\$DF macro has an invalid format.

#### **CFE -- Error in unit number**

(Severe Error/CETAB Parse) The unit number on an SLT\$DF is invalid.

#### **CFE -- Error in window size**

(Severe Error/CETAB Parse) The window size on an SVCSDF or X3PSDF macro is out of range.

#### **CFE -- Error writing file**

(Severe Error/General) CFE was unable to write a record to the output file; therefore the CETAB source file was not updated. Check to make sure that the disk is not full.

#### **CFE -- Executor node not allowed**

(Error/Command) An attempt has been made to define the executor node as a remote node.

#### **CFE -- Executor node not allowed with command**

(Error/Command) The command parameters specified in the command are not permitted on the executor node.

#### **CFE -- Group not in configuration**

(Error/Configuration) The closed user group in the command is not in the database.

**CFE -- Illegal baud rate**

(Error/Command) The baud rate specified in the command is out of range or not in the table of valid baud rates. Reenter the command with a legal baud rate.

**CFE -- Illegal call mask**

(Severe Error/CETAB Parse) The call mask on a DSC\$DF macro is too long or contains invalid characters.

**CFE -- Illegal call value**

(Severe Error/CETAB Parse) The call value on a DSC\$DF macro is too long or contains invalid characters.

**CFE -- Illegal circuit type**

(Error/Command) The circuit type in the command is illegal for the requested operation. For example, it is illegal to purge a DDCMP circuit.

**CFE -- Illegal command**

(Error/Command) The command you gave to CFE is incorrect.

**CFE -- Illegal CUG name**

(Severe Error/CETAB Parse) The user group name on a CUG\$DF or DSA\$DF macro is too long or contains invalid characters.

**CFE -- Illegal destination address**

(Severe Error/CETAB Parse) The destination address on a DSA\$DF macro is not a valid DTE address.

## **Error Messages**

### **CFE -- Illegal destination name**

(Severe Error/CETAB Parse) The destination name on a DST\$DF macro is too long or contains invalid characters.

### **CFE -- Illegal destination priority**

(Severe Error/CETAB Parse) The list insertion priority on a DST\$DF macro is out of range or contains nonnumeric characters.

### **CFE -- Illegal destination type**

(Severe Error/CETAB Parse) The destination type on a DST\$DF macro is not legal for PSI.

### **CFE -- Illegal device priority**

(Error/Command) The device priority specified in the command is out of range.

### **CFE -- Illegal DTE name**

(Severe Error/CETAB Parse) The DTE name on a RDT\$DF macro is too long or contains invalid characters.

### **CFE -- Illegal executor node address**

(Error/Command) An attempt was made to define the executor node address to greater than the maximum address for the network.

### **CFE -- Illegal group number**

(Severe Error/CETAB Parse) The group number on a CUG\$DF macro is out of range or contains nonnumeric characters.

**CFE -- Illegal logical channel number**

(Severe Error/CETAB Parse) The logical channel number on a PVC\$DF macro is out of range or contains nonnumeric characters.

**CFE -- Illegal maximum data size**

(Severe Error/CETAB Parse) The maximum data size on a PVC\$DF macro is out of range or contains nonnumeric characters.

**CFE -- Illegal object number**

(Severe Error/CETAB Parse) The object number on a DST\$DF macro is out of range or contains nonnumeric characters.

**CFE -- Illegal owner process**

(Severe Error/CETAB Parse) The owner process identification field on a PVC\$DF macro is not a recognized owner.

**CFE -- Illegal PVC name**

(Severe Error/CETAB Parse) The circuit identification on a PVC\$DF macro is too long or contains invalid characters.

**CFE -- Illegal state**

(Severe Error/CETAB Parse) The state on a PVC\$DF macro is not a recognized state.

**CFE -- Illegal subaddress high range**

(Severe Error/CETAB Parse) The indicated subaddress on a DSA\$DF macro is out of range or contains nonnumeric characters.

**CFE -- Illegal subaddress low range**

(Severe Error/CETAB Parse) The indicated subaddress on a DSA\$DF macro is out of range or contains nonnumeric characters.

## **CFE**

---

### **Error Messages**

#### **CFE -- Illegal task name**

(Severe Error/CETAB Parse) The task name on a DST\$DF macro is too long or contains invalid characters. This message is also generated if a task name is included and the destination object number is not zero.

#### **CFE -- Illegal window size**

(Severe Error/CETAB Parse) The window size on a PVC\$DF macro is out of range or contains nonnumeric characters.

#### **CFE -- Insufficient buffer space**

(Severe Error/General) CFE was not built with enough buffer space to accommodate the CETAB source file information. Remove CFE and reinstall it with an increment.

#### **CFE -- Invalid block size**

(Severe Error/CETAB Parse) The block size on an X2P\$DF macro is out of range or of an invalid format.

#### **CFE -- Invalid call timer value**

(Severe Error/CETAB Parse) The call timer field on the X3P\$DF macro is out of range or is of an invalid format.

#### **CFE -- Invalid circuit identification**

(Error/Command) The format of the circuit ID in the command is invalid.

#### **CFE -- Invalid clear timer value**

(Severe Error/CETAB Parse) The clear timer field on the X3P\$DF macro is out of range or is of an invalid format.

**CFE -- Invalid counter timer value**

(Severe Error/CETAB Parse) The counter timer value on a DTE\$DF, PVC\$DF, or SLT\$DF macro is out of range or contains invalid characters.

**CFE -- Invalid default block size**

(Severe Error/CETAB Parse) The default block size on the X3P\$DF macro is out of range or is of an invalid format.

**CFE -- Invalid default window size**

(Severe Error/CETAB Parse) The default window size field on the X3P\$DF macro is out of range or is of an invalid format.

**CFE -- Invalid destination sequence**

(Error/CETAB) An invalid destination sequence has been detected. This usually indicates a corrupt CETAB.MAC file.

**CFE -- Invalid DTE address**

(Severe Error/CETAB Parse) The DTE address on a DTE\$DF or RDT\$DF macro is too long or contains invalid characters.

**CFE -- Invalid DTE state**

(Severe Error/CETAB Parse) The state on a DTE\$DF macro is not a recognized DTE state.

**CFE -- Invalid hash table size**

(Severe Error/CETAB Parse) The hash table size on a DTE\$DF macro is out of range, is not a power of 2, or contains invalid characters.

**CFE -- Invalid holdback timer value**

(Severe Error/CETAB Parse) The holdback timer value on an X2P\$DF macro is out of range or is of an invalid format.

## **CFE**

---

### **Error Messages**

#### **CFE -- Invalid line identification**

Two error conditions can display this message:

- (Severe Error/CETAB Parse) CFE found an X2P\$DF macro containing an error in the line identification. This error occurs during the initial scan of the CETAB file when CFE is invoked.
- (Error/Command) The line ID specified in the command is incorrect.

#### **CFE -- Invalid maximum block size**

(Severe Error/CETAB Parse) The default block size on the X3P\$DF macro is out of range or is of an invalid format.

#### **CFE -- Invalid maximum clears value**

(Severe Error/CETAB Parse) The maximum clears field on the X3P\$DF macro is out of range or is of an invalid format.

#### **CFE -- Invalid maximum resets value**

(Severe Error/CETAB Parse) The maximum resets field on the X3P\$DF macro is out of range or is of an invalid format.

#### **CFE -- Invalid maximum restarts value**

(Severe Error/CETAB Parse) The maximum restarts field on the X3P\$DF macro is out of range or is of an invalid format.

#### **CFE -- Invalid maximum window size**

(Severe Error/CETAB Parse) The maximum window size field on the X3P\$DF macro is out of range or is of an invalid format.

**CFE -- Invalid multicopy task name**

(Error/Command) The task name of a multicopy object must be of the format `xxx$$$`, where `xxx` is 1 to 3 alphanumeric characters. This message is generated when an attempt is made to define a multicopy object without a multicopy name or a single copy object with a multicopy name.

**CFE -- Invalid number**

(Error/Command) The command has a number out of range.

**CFE -- Invalid process identification**

(Error/Command) The process name specified in the command is too long.

**CFE -- Invalid reset timer value**

(Severe Error/CETAB Parse) The reset timer field on the `X3P$DF` macro is out of range or is of an invalid format.

**CFE -- Invalid restart timer value**

(Severe Error/CETAB Parse) The restart timer field on the `X3P$DF` macro is out of range or is of an invalid format.

**CFE -- Invalid retransmit count**

(Severe Error/CETAB Parse) The retransmit count on an `X2P$DF` macro is out of range or is of an invalid format.

**CFE -- Invalid retransmit timer value**

(Severe Error/CETAB Parse) The retransmit timer value on an `X2P$DF` macro is out of range or is of an invalid format.

## CFE

---

### Error Messages

#### CFE -- Invalid wildcard syntax

(Error/Command) The wildcard line or circuit ID in the command is of an invalid format.

#### CFE -- Line not in configuration

(Error/Configuration) The line specified in the command is not in the CETAB source file.

#### CFE -- Logging monitor not supported

(Error/Configuration) The logging monitor is not supported in your system configuration.

#### CFE -- Missing LLC\$DF for NW

(Error/CETAB) The LLC\$DF macro for NW is missing from the database. This usually indicates a corrupt CETAB.MAC file.

#### CFE -- Missing parameter: *parameter*

(Error/Command) The indicated parameter is required for the type of operation requested and is missing from the command. Correct the command and reenter it.

#### CFE -- Missing X2P\$DF for line *line-id*

(Error/CETAB) There is no X2P\$DF macro for the indicated line. This usually indicates a corrupt CETAB.MAC file.

#### CFE -- No buffer definition

(Error/CETAB) The CETAB source file does not contain a BUF\$DF macro defining your buffers.

**CFE -- No HELP available**

(Informational/General) CFE could not find the help file.

**CFE -- No node definition**

(Error/CETAB) The CETAB source file does not contain a NOD\$DF macro defining the local node.

**CFE -- No partition definition**

(Error/CETAB) The CETAB source file does not contain a PAR\$DF macro defining the network pool.

**CFE -- No PSI support in system**

(Error/Configuration) The CETAB being processed does not include support for PSI.

**CFE -- No routing definition**

(Error/CETAB) The CETAB source file does not contain a ROU\$DF macro defining the routing parameters.

**CFE -- No station template for *line-id***

(Error/CETAB) There are no station templates for the indicated line. This usually indicates a corrupt CETAB.MAC file.

**CFE -- No unit CSR on this line**

(Error/Command) An attempt was made to change the unit CSR for a line that does not have a unit CSR. This part of the command was ignored.

## **CFE**

---

### **Error Messages**

#### **CFE -- No XPT features macro**

(Error/CETAB) The CETAB source file does not contain an FEASDF macro defining the XPT features. You will not be able to modify any of the states defined by this macro.

#### **CFE -- No XPT process macro**

(Error/CETAB) The CETAB source file does not contain an LLC\$DF macro defining the XPT process.

#### **CFE -- No X.29 support in configuration**

(Error/Configuration) The CETAB.MAC file being processed does not contain the X29\$DF macro.

#### **CFE -- No X3P\$DF in database**

(Error/CETAB) The X3P\$DF macro is not in the database. This usually indicates a corrupt CETAB.MAC file.

#### **CFE -- Node address already in use**

(Error/Command) The address you want to give this node is already in use.

#### **CFE -- Node name already in use**

(Error/Command) The name you want to give this node is already in use.

#### **CFE -- Node not in configuration**

(Error/Configuration) The node specified in the command is not in the CETAB source file.

**CFE -- Number of SDBs exceeds number of CCBs**

(Warning/CETAB) The maximum number of SDBs specified in the BUF\$DF macro exceeds the number of CCBs specified in the BUF\$DF macro.

**CFE -- Object not in configuration**

(Error/Configuration) The object specified in the command is not in the CETAB source file.

**CFE -- Object task name must be specified**

(Error/Command) An attempt was made to create an object type entry without an associated task name. Reenter the command with the task name supplied.

**CFE -- Operation invalid for line type**

(Error/Command) The requested operation is not applicable to the line type indicated in the command.

**CFE -- Parameter applicable to PSI systems only**

(Error/Configuration) One or more of the parameters in the command is inapplicable to systems without PSI, and the CETAB being processed is for a non-PSI system.

**CFE -- Parameter illegal for circuit type**

(Error/Command) One or more of the parameters in the command cannot be set for the type of circuit indicated.

**CFE -- Parameter illegal for wildcard operation**

(Error/Command) One or more of the parameters in the command is not allowed for wildcard or KNOWN operations.

## CFE

---

### Error Messages

#### **CFE -- Parameter not applicable**

(Error/Configuration) One or more of the parameters in the command is not applicable to the CETAB configuration (for example, routing parameters for an end node).

#### **CFE -- Pool byte extension less than minimum Defaulting to *nnn*.**

(Warning/Command) The pool byte extension on the command is less than the minimum required for structures defined in the CETAB. The minimum value *nnn*. is being used instead.

#### **CFE -- Process not in configuration**

(Error/Configuration) The process name specified in the command is not in the CETAB source file.

#### **CFE -- Routing parameters not applicable**

(Error/Command) The routing parameters in the command are not applicable to an end node.

#### **CFE -- Syntax error**

(Severe Error/CETAB Parse) The indicated macro contains an invalid or unrecognized value.

#### **CFE -- Task name not allowed with object type zero**

(Error/Command) An attempt was made to force a single task to be associated with the general task connect (type zero). This part of the command was ignored.

#### **CFE -- Too many stations**

(Severe Error/CETAB) There are more stations on a particular line than CFE is prepared to handle. This message is followed by the syntax error message and the entry read from CETAB that caused the station number overflow.

**CFE -- Unknown HELP qualifier**

(Informational/General) CFE has no help on the qualifier requested.

**CFE -- Usage cannot be permanent**

(Error/Command) The usage of an SVC must be either incoming or outgoing.

**CFE -- XPT not allowed to load line tables in process space**

(Error/Configuration) The command included an extension for line tables for XPT. Since XPT cannot have line tables in process space, this part of the command was ignored.

**CFE -- X25-SERVER not in configuration**

(Error/Configuration) There is no support for the X.25 server module in the system.

**CFE -- X29-SERVER not in configuration**

(Error/Configuration) There is no support for the X.29 server module in the system.

## **EVF**

---

### **Error Messages**

#### **10.3 EVF Error Messages**

##### **EVF -- Argument specified multiple times - *xxx***

An /ENTITY argument is used more than once. *xxx* is the argument found more than once.

##### **EVF -- Command line syntax error**

You have entered an invalid command.

##### **EVF -- Error in get command line - *reason***

An error occurred in getting the command line. *Reason* is either a text string or a decimal error number.

##### **EVF -- Failed to close input file - *reason***

The input file could not be closed. *Reason* is either a text string or a decimal error number.

##### **EVF -- Failed to close output file - *reason***

The output file could not be closed. *Reason* is either a text string or a decimal error number.

##### **EVF -- Failed to delete input file - *reason***

The input file could not be deleted. *Reason* is either a text string or a decimal error number. Possible text strings include:

1. Privilege violation

**EVF -- Failed to open input file - *reason***

The input file could not be opened. *Reason* is either a text string or a decimal error number. Possible text strings include:

1. Accessed for write
2. Bad version number
3. Bad file name
4. Bad device name
5. Invalid device or unit
6. No such file
7. Privilege violation

**EVF -- Failed to open output file - *reason***

The output file could not be opened. *Reason* is either a text string or a decimal error number. Possible text strings include:

1. Bad file name
2. Bad device name
3. Device full
4. Invalid device or unit
5. Privilege violation

**EVF -- Failed to spool output file - *reason***

The output file could not be spooled. *Reason* is either a text string or a decimal error number.

## **EVF**

---

### **Error Messages**

#### **EVF -- Illegal argument combination**

You have used the ALL argument in the /ENTITY option along with another entity argument.

#### **EVF -- Illegal data range**

You have entered a start date that is greater than the end date in the /DATE:RANGE or /SINCE and /THROUGH option.

#### **EVF -- Illegal switch combination**

You have used the ALL option along with another option.

#### **EVF -- Input file format error**

The input file you specified is not a binary event log file.

#### **EVF -- Invalid area number**

You have specified an area number in the /ENTITY:AREA option that is not in the allowed range or is not a number.

#### **EVF -- Invalid *circuit-id***

You have specified a circuit-id in the /ENTITY:CIRCUIT option that is not a valid circuit identifier. If the error occurred because of an incorrect use of the wildcard character (\*), the error is qualified with '- illegal use of wildcard'.

#### **EVF -- Invalid date format**

You have specified a date and time in the /DATE:RANGE or /SINCE and /THROUGH options is in the wrong format.

**EVF -- Invalid *event-id***

You have specified an event-id in the /EVENT option that is not a valid event identifier. If the error involved the incorrect use of wildcards, this error is qualified with '- illegal use of wildcard'. If too many events were specified (more than 6), this error is qualified with '- too many event-ids declared'.

**EVF -- Invalid *line-id***

You have specified a line-id in the /ENTITY:LINE option that is not a valid line identifier. If the error occurred because of an incorrect use of the wildcard character (\*), the error is qualified with '- illegal use of wildcard'.

**EVF -- Invalid *module-id***

You have specified a module-id in the /ENTITY:MODULE option that is not a valid module identifier. The module identifier is not alphanumeric.

**EVF -- Invalid *node-id***

You have specified a node-id in the /ENTITY:NODE or /SOURCE option that is not a valid node identifier. The node-id was not in the correct format, the address is out of range, or the node name is greater than 6 characters.

**EVF -- Switch specified multiple times - *xxx***

You have used an option more than once in the command. *xxx* is the option found more than once.

**EVF -- Too few arguments - *xxx***

You have used an option that required an argument but no argument was found. *xxx* is the option.

## **EVF**

---

### **Error Messages**

#### **EVF -- Unknown argument - *xxx***

You have used an argument after the DATE, ENTITY or FORMAT option that was not recognized. *xxx* is the unrecognized argument.

#### **EVF -- Unknown switch - *xxx***

You have used an unrecognized option. If you abbreviate options to one letter you will receive this error message. *xxx* is the option that was in error.

## 10.4 HLD Error Messages

Some of the HLD error messages include strings that are replaced with run-time variables. A list of these strings and the type of information they contain is shown below.

*(-nnn.)* A DSW or IOSB error code (always negative).

*(nnn.)* A network code (always positive).

*node* The name of the remote RSX-11S node.

*task* The installed task name received from SLD.

*filespec* A copy of the file specification from the HTASK\$ macro. This is followed by (File) to indicate that HLD found a match from the external mapping tables, or by (G-P) to indicate that the task was found on the general purpose list, or by (File, G-P) to indicate that both conditions were satisfied.

Variable fields are ignored in the alphabetization of error messages in this chapter.

**HLD -- *nodeltask* - Base address not a multiple of 4K**

**HLD -- *filespec***

The file was not built with the /MM switch. Rebuild the file.

**HLD -- *nodeltask* - Connect accept failure (-nnn.)**

**HLD -- *filespec***

The attempt to accept a connection from a remote SLD has failed. The remote node may have become unreachable.

## HLD

---

### Error Messages

**HLD -- *nodeltask* - Disk read failure (-*nnn*.)**

**HLD -- *filespec***

This error message indicates a disk error while reading the file (initial load or checkpoint read). Validate the integrity of the file and the disk. The second error message line may or may not appear.

**HLD -- *nodeltask* - Disk write failure (-*nnn*.)**

This error message indicates a disk error while writing to the file (checkpoint write). Validate the integrity of the file and the disk.

**HLD -- *nodeltask* - File has never been installed**

The task file has not been installed (via VMR) into the RSX-11S system. This error message appears if you rebuild the task file but neglect to install it. Use VMR to install the task into the RSX-11S system.

**HLD -- *nodeltask* - File has no task header**

**HLD -- *filespec***

The file was built with the */-HD* switch. It is not a valid RSX-11S task image. Rebuild the file.

**HLD -- *nodeltask* - File is larger than partition**

**HLD -- *filespec***

The initial load size of the file is larger than the partition size in the RSX-11S system. Either make the partition larger or rebuild the file to use a smaller partition size.

**HLD -- *nodeltask* - File is not contiguous**

**HLD -- *filespec***

The file is not contiguous. Either rebuild the file or use PIP to make the file contiguous (see example below).

```
>PIP /NV/CO=filespec
```

```
>PIP filespec/PU
```

**HLD -- *nodeltask* - File spec parse error**  
**HLD -- *filespec***

The file specification for the file has an invalid format. Rebuild the HLD database.

**HLD -- *nodeltask* - G-P tasks may not be checkpointed**  
**HLD -- *filespec***

A checkpoint operation was attempted for an HTASK\$ macro from the general purpose list. Edit and rebuild the appropriate mapping table to move the HTASK\$ macro from the general purpose list into a node-specific list.

**HLD -- Invalid connect request from *node-id***

A task attempted to connect to HLD but did not supply a valid optional connect data message. Refer to your object type tables within NETACP. HLD is connected to by the object NO.RTL and not by the installed task name. NO.RTL is defined in the NSSYMS\$ macro. Its object type number is 18.

**HLD -- *nodeltask* - Invalid data type in mailbox (*nnn.*)**

A network data type was found in the mailbox that is not used by HLD/SLD. If the data type is a connect, NSP has tried to place more connects onto the mailbox than HLD can handle.

**HLD -- *nodeltask* - Label block read failure (*-nnn.*)**  
**HLD -- *filespec***

The label block (TKB information) could not be accessed successfully. Rebuild the file.

## HLD

---

### Error Messages

#### **HLD -- *nodeltask* - LUN fixing not supported**

**HLD -- *filespec***

The remote SLD was not built with the LUN-fixing module. Either rebuild the remote SLD, or edit and rebuild the appropriate mapping table to move the HTASK\$ macro from general purpose list into a node-specific list. This error also occurs if the HTASK\$ macro is already in a node-specific list but uses the LUN control argument.

#### **HLD -- Mailbox LUN assignment failure (-*nnn*.)**

HLD could not successfully assign the network mailbox LUN to the NS: device. Check to see if the local node's state is ON.

#### **HLD -- *nodeltask* - Mapping file base address not zero**

This error is caused by an incorrect TKB command file. Edit the TKB command file, add the */MM* switch to the output file specifier, and then add or modify the PAR statement to read PAR=ANY:0:20000. Then rebuild the file.

#### **HLD -- *nodeltask* - Mapping file contains a task header**

This error is caused by an incorrect TKB command file. Edit the TKB command file and add the */HD* switch to the output file specifier. Then rebuild the file.

#### **HLD -- *nodeltask* - Mapping file contains garbage**

HLD can successfully read the external file, but the file is internally inconsistent. Rebuild the file.

#### **HLD -- *nodeltask* - Mapping file is larger than HLDFIL buffer space**

The external mapping file is larger than the buffer space available. Merge the external table into the internal table. Discontinue use of the external table.

**HLD -- *nodetask* - Mapping file is not contiguous**

The file is not contiguous. Either rebuild the file or use PIP to make the file contiguous (see example below).

```
>PIP /NV/CO=HLDTAB.TSK  
>PIP HLDTAB.TSK/PU
```

**HLD -- *nodetask* - Mapping file label block read failure (-*nnn*.)**

HLD could not read the file's label block (which contains TKB descriptive information). Rebuild the file.

**HLD -- *nodetask* - Mapping file open failure (-*nnn*.)**

HLD could not open the file. Rebuild the file.

**HLD -- *nodetask* - Mapping file read failure (-*nnn*.)**

HLD could not read the external mapping file. Rebuild the file.

**HLD -- Network data queue empty**

HLD found nothing in the network data queue. This error occurs if you issue MCR>RUN HLD... or if the connection times out before HLD can remove the connect request from the network data queue. Increase the connect timeout period.

**HLD -- *nodetask* - Network disconnect (*nnn*.)**

NSP has broken the logical link before the transfer has been completed.

**HLD -- Network open failure (-*nnn*.)**

HLD could not successfully issue the OPNS\$ macro. Check to see if the local node's state is ON.

## HLD

---

### Error Messages

#### **HLD -- *nodeltask* - Network read failure (-*nnn*.)**

A communication error has occurred at the remote SLD. The logical link is broken.

#### **HLD -- Network SPA failure (-*nnn*.)**

HLD could not successfully issue the SPAS\$ macro.

#### **HLD -- *nodeltask* - Network write failure (-*nnn*.)**

A communication error has occurred at the remote SLD. The logical link is broken.

#### **HLD -- *nodeltask* - No entry in *nodeltask* tables**

Your mapping tables do not contain an HTASK\$ macro that corresponds to the task name from the error message.

#### **HLD -- NS: Work LUN assignment failure (-*nnn*.)**

HLD could not successfully assign one of its nonmailbox LUNs to the NS: device.

#### **HLD -- *nodeltask* - Open failure (-*nnn*.)**

#### **HLD -- *filespec***

The file named by the HTASK\$ macro could not be opened successfully. If there is an FCS problem, you must rebuild the file. If not, change either the file or the mapping table. If you change the file name, make it match the mapping table. If you change the mapping table, make it match the file name.

**HLD -- *node/task* - Partition is larger than CHKPT *space***

**HLD -- *filespec***

The partition size in the RSX-11S system is larger than the checkpoint space inside the file. Typically, this indicates that the partition size in your PAR= statement is smaller than the partition's real size in the RSX-11S system. Although the load size of a task can be much smaller than its partition, the entire partition is transferred during checkpoint operations. Rebuild the file with the same partition size as the RSX-11S system.

**HLD -- *node/task* - SLD abort/disconnect**

This error message indicates that SLD detected a file error during an initial load or checkpoint read. Reinstall the task into the RSX-11S system.

**HLD -- *node/task* - SLD premature disconnect**

SLD has disconnected during a checkpoint write before the transfer is complete.

## NCP

---

### Error Messages

#### 10.5 NCP Error Messages

There are two sets of error messages for NCP: a full set for RSX-11M and RSX-11M-PLUS systems and a subset for RSX-11S systems. There are two groups of messages within the full set: one group is RSX system-specific, and the other is standard for all DECnet systems. The error message descriptions below identify each message as being RSX-specific, RSX-11S, or standard DECnet.

##### 10.5.1 RSX-11M/M-PLUS Error Message Format

The general format for an RSX-11M/M-PLUS NCP error message is:

```
NCP -- [<component>,<command> <diagnostic>, <error message>
      [,<error detail>] [<extra text>]
```

where

*component* is the affected component (for example, LINE *line-id*). Appears only in a command affecting multiple components.

*command* is the command that caused the error.

*diagnostic* is one of the following values:

**FAILED**

This command was accepted by NCP, but execution failed.

## NOT ACCEPTED

NCP would not accept the command, and execution was not attempted. When NCP rejects a command, it displays the command line with two right angle brackets at the point where NCP stopped parsing. If NCP was parsing a pair of words, it points to the first word of the pair even if the second word was incorrect -- for example,

```
SET EXECUTOR >>ROUTING RIMER 50
```

NCP does not recognize ROUTING RIMER. Since both words make up one parameter, NCP rejects the parameter and places brackets in front of it.

*error message* is the major reason for the failure. In most cases, these reasons are identical for all DECnet implementations (see the *DNA Network Management Functional Specification*).

If NCP cannot interpret an internal error code, the error message field contains one of the following:

**Management return #*n***

An error code of *n* (decimal) was returned by network management software.

**System return #*n***

An error code of *n* (decimal) was returned by the RSX operating system software.

## NCP

---

### Error Messages

*error detail* is a detailed explanation of the failure (for certain error messages). All DECnet implementations have the same definitions for these messages. See the *DNA Network Management Functional Specification* for details

If NCP cannot interpret an error detail code, the error detail field contains *detail #n*, meaning that an error detail of *n* (decimal) was returned by network management software.

*extra text* is an additional system-specific explanation of the error condition. For a description of the contents of this field in an error from a non-DECnet-RSX implementation, consult the documentation for that system.

### 10.5.2 RSX-11S Error Message Format

The general format for an RSX-11S NCP error message is:

NCP -- <*error message*>

where

*error message* See description given above for RSX-11M/M-PLUS.

### 10.5.3 NCP Error Messages

#### **NCP -- Bad loopback response**

(Standard DECnet) The message that was returned in a loopback test did not match the message that was sent. This can be due to a loopback protocol violation, bad data returned, or bad message length returned.

#### **NCP -- Bad management response**

(Standard DECnet) The information returned to NCP either from its internal network management functions or from the network management listener was improperly formatted or contained an invalid value. This error condition implies a programming error in the module that responded to NCP.

**NCP -- Circuit in wrong state**

(Standard DECnet) The current operational state of the circuit precludes the requested operation. When this message is received on an end node, it usually means that a circuit is already active. There can be only one circuit active on an end node at a given time.

**NCP -- Component in wrong state**

(Standard DECnet) The current operational state of the component precludes the requested operation (for example, you cannot down-line load a line that is off). The error detail included with this message identifies the component (for example, circuit, line, node, and so on).

When a LOOP CIRCUIT command is given on an RSX system, one of the following RSX system-specific extra text messages may be included:

**Circuit is loopback node**

The circuit is currently associated with a loop node name, and its state cannot be changed to do the circuit test.

**Circuit not restarted**

An attempt was made to return the circuit to normal use before the circuit had been restarted.

**Device off line**

The circuit is no longer on the I/O bus. This message is specific to RSX-11M-PLUS.

**Other management circuit service active**

The circuit is currently being used to perform some other network management circuit service function, such as an up-line dump or a down-line load.

## **NCP**

---

### **Error Messages**

#### **Reassign circuit to former owner failed**

At the completion of the test, the owner attempted to return the circuit to normal use. The attempt failed because the circuit could not be given back to the owner.

#### **NCP -- Error reading command**

(RSX-specific) NCP could not read a command from either the terminal or a command file because of a hardware error or an unrecognized command read error.

#### **NCP -- Exceeded command file nesting depth**

(RSX-specific) A command file attempted to start another command file when there were already six command files nested.

#### **NCP -- File I/O error**

(Standard DECnet) A hardware error was encountered while reading or writing one of the files necessary to the requested operation. The error detail included with this message identifies the type of file:

<b>Dump file</b>	Output file for an up-line dump
<b>Load file</b>	Input file for a down-line load
<b>Permanent database</b>	File containing default parameters
<b>Secondary dumper</b>	Input file for the secondary dumper program that runs in the target node
<b>Secondary loader</b>	Input file for the secondary loader program that runs in the target node
<b>Tertiary loader</b>	Input file for the tertiary loader program that runs in the target node

**NCP -- File open error**

(Standard DECnet) One of the files necessary for the requested operation could not be reserved for use. The error detail is the same as that described above for the File I/O error message.

**NCP -- File privilege violation**

(RSX-specific) The file specified with the TO option in a SHOW command is not available to the user of NCP.

**NCP -- Hardware failure**

(Standard DECnet) The requested operation could not be performed because a related hardware component failed to function as needed.

**NCP -- Incompatible management version**

(Standard DECnet) The network management listener version is incompatible with NCP.

**NCP -- Invalid device name syntax**

(RSX-specific) The device name in a command file or TO specification did not have legal syntax.

**NCP -- Invalid directory syntax**

(RSX-specific) The directory identification (UIC) in a command file or a TO specification did not have legal syntax.

**NCP -- Invalid file contents**

(Standard DECnet) The requested operation could not be performed because the files contained data of an invalid form or value. The error detail is the same as that described previously for the File I/O error message.

## **NCP**

---

### **Error Messages**

#### **NCP -- Invalid file name syntax**

(RSX-specific) The file name in a command file or a TO file specification did not have legal syntax.

#### **NCP -- Invalid identification format**

(Standard DECnet) The identification of the component involved in the requested operation did not have proper syntax (for example, a numeric character in a node name). The error detail included with this message identifies the type of component (for example, line, node, circuit, and so on).

#### **NCP -- Invalid length value, maximum loop length = 128**

(RSX-11S) The value of the length parameter in a LOOP NODE command exceeds the buffering capability of subset XPT. Use a smaller value for the length parameter.

#### **NCP -- Invalid message format**

(Standard DECnet) The information sent by NCP either to its internal network management functions or to a network management listener was improperly formatted or contained an invalid value. This error condition implies a programming error in NCP.

### **NOTE**

This error message is also used to indicate standard DECnet options not supported by DECnet-RSX.

#### **NCP -- Invalid parameter grouping**

(Standard DECnet) The parameters furnished by the user for the requested operation cannot be included in a single command.

**NCP -- Invalid parameter value**

(Standard DECnet) The value of a parameter furnished by the user for the requested operation was not acceptable (for example, a numeric parameter out of range). The error detail included with this message identifies the type of parameter.

In the case of the events parameter for the LOGGING command, the following RSX system-specific extra text message may be provided:

**Event cannot be filtered**

The ability to enable or disable one of the specified events is not implemented.

In the case of the length parameter for a LOOP command, one of the following RSX system-specific extra text messages may be included. In each of these cases, the length was more than could be handled, and the maximum length will be included with the error message.

**Active looper size exceeded**

The requested length exceeds the buffering capability of the active looper task. Reduce the looper's size or reinstall the looper with a memory increment.

**Mirror size exceeded**

The requested length exceeds the buffering capability of the network management loopback mirror. Either use a smaller length or (on an RSX system) reinstall the mirror with a memory increment.

**NCP -- Line communication error**

(Standard DECnet) The requested operation failed because of communication errors on the involved line. This error condition applies only to functions that involve direct use of a line, such as down-line load and line loop tests.

## **NCP**

---

### **Error Messages**

In the case of a LOOP LINE command, one of the following RSX system-specific extra text messages may be included:

#### **Receiver**

The looper failed to receive a message that should have been looped back on the line.

#### **Transmitter**

The looper failed to transmit a message on the line. This is specifically a failure of the transmit function.

### **NCP -- Line protocol error**

(Standard DECnet) The requested operation failed because of protocol errors on the involved line. This usually implies either incompatible line protocols or protocol-programming errors. There is a possibility that the condition is caused by a line hardware error that was not detected by the line protocol.

Line protocol can mean either the Data Link Protocol or the service operation protocol.

This error condition applies only to functions involving direct use of a line, such as down-line load and line loop tests.

### **NCP -- Listener connect failed**

(Standard DECnet) The logical link from NCP to the network management listener could not be connected. This error message generally has one of the following error details:

#### **Access control rejected**

The remote node or the network management listener could not understand or would not accept the access control information.

**Invalid node name format**

The executor rejected the format of the remote node name (for example, the name contained illegal characters or was too long).

**Invalid object name format**

The remote node did not understand the object name format used by NCP to identify the network management listener.

**Local node shut down**

The executor node is in the process of shutting down and will not accept logical link connections.

**Network resources**

Either the local or the remote network management listener had insufficient network resources to handle the connect request.

**No response from object**

The network management listener did not respond. This could be, for example, due to its responding too slowly or terminating abnormally.

**Node unreachable**

No path exists to the remote node.

**Object too busy**

The remote node had insufficient resources available to forward the connect request to the network management listener.

**Remote node shut down**

The remote node is in the process of shutting down and will not accept logical link connections.

## **NCP**

---

### **Error Messages**

#### **Unrecognized node name**

The destination node name does not correspond to any known node address.

#### **Unrecognized object**

The remote node does not have a network management listener.

### **NCP -- Listener link disconnected**

(Standard DECnet) The logical link from NCP to the network management listener was unexpectedly disconnected. This error message will have one of the following error details:

#### **Abort by management**

An operator or program used network management to abort the logical link. NCP recognizes this condition, although DECnet-RSX does not support the ability to cause it.

#### **Abort by object**

The network management listener aborted the logical link. This indicates a programming error in the network management listener.

#### **Disconnect by object**

The network management listener disconnected the logical link. This indicates a programming error in the network management listener.

#### **Node or object failed**

The network aborted the logical link for one of the following reasons: The remote node terminated abnormally, or the network management listener terminated abnormally.

**Node unreachable**

The network aborted the logical link because it could no longer find a path to the remote node.

**NCP -- Management program error**

(Standard DECnet) The network management software has detected an internal programming error.

**NCP -- Mirror connect failed**

(Standard DECnet) The logical link to the network management loopback mirror could not be connected. This error message generally has one of the following error details:

**Access control rejected**

The remote node or the network management loopback mirror either could not understand or would not accept the access control information.

**Invalid node name format**

The executor rejected the format of the remote node name (for example, the name contained illegal characters or was too long).

**Invalid object name format**

The remote node did not understand the object name format used to identify the network management loopback mirror.

**Local node shut down**

The executor node is in the process of shutting down and is accepting no more logical link connections.

### **Error Messages**

#### **Network resources**

Either the local or the remote node had insufficient network resources to connect the logical link.

#### **No response from object**

The network management loopback mirror did not respond. This could be, for example, due to its responding too slowly or terminating abnormally.

#### **Node unreachable**

No path exists to the remote node.

#### **Object too busy**

The remote node had insufficient resources available to forward the connect request to the network management loopback mirror.

#### **Rejected by object**

The logical link could not be connected because the network management loopback mirror rejected the connection. This most likely implies that the loopback mirror is too busy to accept another logical link.

#### **Remote node shut down**

The remote node is in the process of shutting down and will accept no more logical link connections.

**Unrecognized node name**

The destination node name does not correspond to any known node address.

**Unrecognized object**

The remote node does not have a network management loopback mirror.

**NCP -- Mirror link disconnected**

(Standard DECnet) The logical link from NCP to the network management listener was unexpectedly disconnected. This error message generally has one of the following error details:

**Abort by management**

An operator or program used network management to abort the logical link. NCP recognizes this condition, although DECnet-RSX does not support the ability to cause it.

**Abort by object**

The network management loopback mirror aborted the logical link. This indicates a programming error in the network management loopback mirror.

**Disconnect by object**

The network management loopback mirror disconnected the logical link. This indicates a programming error in the network management loopback mirror.

**Node or object failed**

The network aborted the logical link because either the remote node or the network management loopback mirror terminated abnormally.

## **NCP**

---

### **Error Messages**

#### **Node unreachable**

The network aborted the logical link because it could no longer find a path to the remote node.

#### **NCP -- NCP program error**

(RSX-specific) NCP detected an internal programming error.

#### **NCP -- No room for new entry**

(Standard DECnet) The requested operation could not be performed because it required the addition of a new entry in some database and that database was full.

#### **NCP -- Not remotely executable**

(Standard DECnet) The requested operation cannot be sent to a remote node.

#### **NCP -- Operation failure**

The requested operation failed for one of the following reasons: a circuit was not owned by XPT or could not be turned off, a node was not in the XPT database, or XPT was not loaded.

#### **NCP -- Operation failure**

(Standard DECnet) The requested operation failed. For some RSX system-specific failures, one of the following extra text messages may be included:

#### **DLX error #*n***

A direct line control (DLX) function failed with decimal error code *n*. Consult DLX documentation for an explanation of the error code.

**Insufficient buffer space for additional aliases**

The network management software could not internally buffer all the aliases that could have been displayed.

**Insufficient buffer space for additional nodes**

The network management software could not internally buffer all the nodes that could have been displayed.

**Insufficient buffer space for additional objects**

The network management software could not internally buffer all the objects that could have been displayed.

**Network initializer function failed**

The network initializer (NTINIT) failed to perform its part of the requested operation. The network initializer will write an error message to the requesting terminal or, in the case of a remotely executed command, to the system console.

**Network initializer not installed**

The network initializer could not run because it is not properly installed.

**Network loader function failed**

The network loader (NTL) failed to perform its part of the requested operation. The network loader will write an error message to the requesting terminal or, in the case of a remotely executed command, to the system console.

**Network loader not installed**

The network loader could not run because it is not properly installed.

## **NCP**

---

### **Error Messages**

#### **Node in wrong state**

The requesting operation could not be performed because the executor node is in the wrong state.

#### **PSI not generated**

An attempt was made to perform a PSI operation on a system that does not support PSI.

#### **Server task not installed**

The separate task needed to perform the requested operation is not properly installed. This could be the active looper for a loop test, the down-line loader for a down-line load or trigger, or the up-line dumper for an up-line dump.

#### **NCP -- Oversized management command response**

(Standard DECnet) The command message sent by NCP was too big for the network management listener.

#### **NCP -- Oversized management response**

(Standard DECnet) The response message returned by the network management listener was too big for NCP.

#### **NCP -- Parameter missing**

(Standard DECnet) The user did not supply a parameter that is necessary for the requested operation. The error detail included with this message identifies the type of the parameter.

#### **NCP -- Parameter not applicable**

(Standard DECnet) The user supplied a parameter that is not applicable to the requested operation on the specified component. The error detail included with this message identifies the type of parameter.

**NCP -- Parameter value too long**

(Standard DECnet) The user supplied a parameter value that was too long to be accepted by the network management listener. The error detail included with this message identifies the type of the parameter.

**NCP -- Privilege violation**

(Standard DECnet) The user does not have sufficient privilege to perform the requested operation.

In the case of a LOOP LINE or LOOP CIRCUIT command, the following RSX system-specific extra text message may print out:

**Service access denied**

The line or circuit is not in SERVICE state and the nonprivileged user cannot perform a line or circuit test.

**NCP -- Resource error**

(Standard DECnet) Network management had insufficient internal resources to perform the requested operation.

**NCP -- System in wrong state**

(RSX-specific) NCP cannot perform the requested function because the DECnet system (the Communications Executive) has not been properly initialized.

**NCP -- System-specific management function not supported**

(Standard DECnet) The requested operation is system specific to some DEC systems and is not supported by the network management listener.

**NCP -- Unrecognized circuit identification**

(RSX-11S) The circuit specified by the user does not exist or did not contain necessary information (such as a tributary number).

## **NCP**

---

### **Error Messages**

#### **NCP -- Unrecognized command**

(Standard DECnet) NCP does not have the command the user entered.

#### **NCP -- Unrecognized command**

(RSX-11S) NCP does not have the command that the user entered, or the command contained an unrecognized keyword or an invalid parameter value (for example, a node name contained 7 characters instead of the allowed 6-character name).

#### **NCP -- Unrecognized component**

(Standard DECnet) The component specified by the user does not exist. The error detail included with this message identifies the type of component (for example, line, node, circuit, and so forth).

In the case of the LOOP LINE command, the following RSX system-specific extra text message may be displayed:

#### **Or line cleared**

The line may be defined in the system configuration, but it is in the CLEARED state.

#### **NCP -- Unrecognized device or unit**

(RSX-specific) The device name or unit number in a command file or a TO file specification could not be found by the RSX operating system.

#### **NCP -- Unrecognized file, device, or directory**

(RSX-specific) The file name, device name, or directory (UIC) in a command file or a TO file specification could not be found by the RSX operating system.

#### **NCP -- Unrecognized function or option**

(Standard DECnet) The requested operation is not implemented by the executor.

**NCP -- Unrecognized keyword**

(Standard DECnet) One of the keywords in a command is unknown to NCP.

**NCP -- Unrecognized line identification**

(RSX-11S) The line specified by the user does not exist or did not contain necessary information (such as a tributary number).

**NCP -- Unrecognized node identification**

(RSX-11S) The node specified by the user does not exist.

**NCP -- Unrecognized parameter type**

(Standard DECnet) One of the parameters given by the user is not implemented by the executor. The error detail included with this message identifies the type of parameter.

**NCP -- Unrecognized value**

(Standard DECnet) A parameter value given by the user was unknown to NCP.

**NCP -- Value out of range**

(Standard DECnet) A parameter value given by the user is out of the range implemented in NCP.

## **NDA**

---

### **Error Messages**

#### **10.6 NDA Error Messages**

NDA prints an error message on your terminal when it detects one of the error conditions described below. These error messages reflect operational conditions and must not be confused with the diagnostic analysis messages that NDA generates during the analysis and prints in the analysis listing.

Note that a few of the messages listed below are informational in nature, while most are fatal errors resulting in termination of the analysis.

##### **NDA -- Analysis output must be directed to an explicit device or file**

(Fatal) NDA requires an explicit output file specification. There are no default output file names.

##### **NDA -- Analysis terminated after $n$ pages**

(Informational) NDA terminated the analysis after generating  $n$  pages of analysis output.

##### **NDA -- Command I/O error**

(Fatal) The system returned an error when NDA attempted to read a command line.

##### **NDA -- Command line syntax error**

(Fatal) NDA detected an error in the syntax of an NDA command line. NDA points to the error within the command line.

##### **NDA -- Crash dump must be input from an explicit device or file**

(Fatal) The crash dump input file specification must be explicit. There is no default file specification for the crash dump input.

##### **NDA -- Device driver missing**

(Fatal) You have not loaded the driver for the crash dump input device.

**NDA -- Error reading crash dump**

(Fatal) NDA either could not read or could not access the crash dump input device. The device may not be properly allocated or may contain bad blocks.

**NDA -- Error reading file *filename***

(Fatal) The system returned an error when NDA attempted to read the crash dump file. This could be caused by a device error or a bad block on the volume.

**NDA -- Error reading symbol file *filename***

(Fatal) The system returned an error when NDA attempted to read the symbol table definition file.

**NDA -- Error writing analysis file**

(Fatal) The system returned an error when NDA attempted to write a line into the analysis listing file. This could be caused by a problem with the device, a full volume, or a bad block on the volume.

**NDA -- Error writing dump file *filename* Error=*n* DSW=*n***

(Fatal) The system returned an error when NDA attempted to write into the binary output file. This condition could be caused by a problem with the device, a full volume, or a bad block on the volume.

**NDA -- Errors detected: *n***

(Informational) NDA detected *n* analysis errors during the run.

**NDA -- Failed to assign LUN to input device *ddu***

(Fatal) A directive to assign a LUN failed when NDA attempted to use it to attach the specified input device before reading the crash dump from the device. The directive to assign a LUN fails if the device name in the NDA command line is invalid.

## **NDA**

---

### **Error Messages**

#### **NDA -- Failed to extend page buffer - $n$ pages available**

(Informational) A directive to extend the task failed when NDA attempted to use it to expand the page buffer. This problem causes the analysis to take longer, but the analysis continues with a buffer of  $n$  pages, each 266 words long.

#### **NDA -- Failed to open input file *filename***

(Fatal) One of the following conditions exists:

- The specified device does not exist.
- The volume is not mounted.
- A problem exists with the device.
- The specified UFD does not exist.
- The specified file does not exist.
- You do not have read access privileges.

#### **NDA -- Failed to open output file *filename***

(Fatal) One of the following conditions exists:

- The specified device does not exist.
- The volume is not mounted.
- A problem exists with the device.
- The specified UFD does not exist.
- The volume is full or the device is write protected.
- You do not have write access privileges to UFD.

**NDA -- Illegal switch**

(Fatal) You have specified an unknown switch or have used a valid switch after the wrong file specification.

**NDA -- Illegal trap - snapshot dump being attempted**

(Fatal) NDA has aborted after detecting an odd address or some other type of fault. This message indicates a software problem.

**NDA -- Indirect command syntax error**

(Fatal) The name of the indirect command file (*@filename*) is syntactically incorrect.

**NDA -- Indirect file open failure**

(Fatal) NDA could not open an indirect command file specified as *@filename* in the NDA command line.

**NDA -- Maximum indirect file depth exceeded**

(Fatal) You have exceeded the maximum allowable number of nested indirect command files (only one level of indirection is permitted in NDA). The rest of the top level command file is executed.

**NDA -- No input file specified**

(Fatal) You did not supply an input file specification for the crash dump.

**NDA -- No output file specified**

(Fatal) You have not specified an output file.

**NDA -- Output dump file name must be explicit**

(Fatal) You have not specified an output dump file.

## NDA

---

### Error Messages

#### **NDA -- Premature end of dump input - *filename* being zero filled**

(Informational) NDA reached the end of the medium (or end-of-file mark, if a magnetic tape) before the crash dump output file had been completely filled. If you expected the file to be completely filled by the dump, this condition could indicate a problem.

#### **NDA -- Symbol file *filename* has illegal format**

(Fatal) The specified executive symbol table file has an improper format, probably caused by entry of the wrong file name. However, this message could also indicate a problem with the device or medium on which the file is located.

#### **NDA -- Symbol *symbol-name* not defined in symbol file**

(Fatal) NDA did not find a symbol it required for the analysis in the specified executive symbol table file. The analysis may be restarted specifying the /SYM switch, forcing NDA to continue with the analysis, using a zero value for all undefined symbols.

#### **NDA -- Transfer complete - *ddu* may be unloaded**

(Informational) The transfer of the crash dump to the output file is finished; you may unload the crash dump device. This message occurs only when you have specified a binary save file in the command string to NDA.

#### **NDA -- Unknown get command line error**

(Fatal) Unrecognized error has occurred when NDA attempted to read a command line.

#### **NDA -- Warning kernel PARs do not contain proper values**

(Informational) This message indicates that the values contained in the page address registers are invalid. NDA attempts to finish the analysis using the indicated values. When restarting the analysis, you must specify the /KMR switch (Section 6.7.2). This switch forces NDA to use standard mapping values when converting kernel virtual addresses to physical memory addresses.

## 10.7 NTD Error Messages

NTDEMO prints an error message on your terminal if it detects one of the conditions described below when you invoke the NTD utility.

### **NTDEMO -- Network not loaded**

The network at the node is not loaded (that is, no network software is running at the local node).

### **NTDEMO -- Unable to connect to remote server**

NTD is unable to communicate with NTDEMO on the node you requested (that is, the host node if you typed >NTD, or the node you specified if you typed >NTD *node-id*). This could be due to any one of the following conditions:

- The node is unreachable.
- The node does not support the NTDEMO server. (This software runs only on DECnet-RSX/IAS nodes.)
- The node does not have NTDEMO installed.
- The maximum number of links to NTDEMO are busy. (This value is defined during network generation.)

### **NTDEMO -- Unable to open network**

The network at the node is loaded, but not currently available for network operations.

## NTINIT

---

### Error Messages

#### 10.8 NTINIT Error Messages

The network initializer (NTINIT) is used with either NCP or VNP to initialize the network software. All NTINIT errors are fatal. The attempted operation cannot succeed unless the problem identified in the message is corrected.

When an NCP or a VNP command executes locally, error messages are displayed at the requesting terminal. When the command executes remotely, error messages are displayed on the remote system console.

Some of the error messages include words printed in italics. The italicized words are variables. NTINIT replaces them with a specific term. For example, in the message NTINIT -- Failed to load line *line-id*, the initializer replaces *line-id* with the actual line identification.

#### **NTINIT -- Clearing system**

NTINIT is removing the network software from the system because of a failure on the marked-for-load scan. To recover, use CFE to correct the condition that caused the failure. Then execute another SET SYSTEM command and try to load again. If the failure recurs, shut down more lines and circuits before loading again. Repeat this procedure until the load is successful or until all lines and circuits are shut down.

If the error continues after all lines and circuits are shut down, one of the processes is at fault. Set one process to the CLEAR state and try the SET SYSTEM command again. Repeat this action using a different process until the SET SYSTEM command is successful. The last process to be cleared is the faulty one. Then use CFE DEFINE commands to reload the lines, circuits, and processes (except the erring one).

#### **NTINIT -- Comm Exec not loaded**

The Communications Executive is not loaded in main memory. Use the NCP SET SYSTEM command to load the network software.

**NTINIT -- Device *controller-id* offline**

The CSR address for the device does not exist on the bus. Use CFE to change the CSR value in the configuration file or use the CSR keyword when issuing the NCP SET LINE command. If the line is set to ON during NETGEN, you must use the following NCP commands to clear the line and reload it:

```
NCP>CLEAR LINE line-id ALL
NCP>SET LINE line-id CSR device-id STATE ON
```

**NTINIT -- Failed to load line *line-id***

A SET LINE request failed. You will also receive an NTL error message that gives a more detailed description of the cause of the error (for example, NTL -- Device off line).

**NTINIT -- Failed to load process *process-id***

A SET PROCESS request failed. You will also receive an NTL error message that gives a more detailed description of the cause of the error (for example, NTL -- Process not in system).

**NTINIT -- Illegal function**

An attempt to reprogram the network software resulted in a programming error.

**NTINIT -- Incompatible with RSX-11M system**

The initializer was built with the wrong RSX11M.STB file. Rebuild the network software with the correct RSX11M.STB file.

## **NTINIT**

---

### **Error Messages**

#### **NTINIT -- Incompatible with RSX-11S system**

The initializer was built with the wrong RSX11S.STB file. Rebuild the network software with the correct RSX11S.STB file.

#### **NTINIT -- Network ACP (NETACP) not installed**

NETACP must be installed to perform the SET EXECUTOR STATE ON operation. To clear the problem, use the MCR INSTALL NETCAP command to install the network ACP.

#### **NTINIT -- Network loader (NTL...) not installed**

NTL must be installed to perform the NCP SET SYSTEM operation. To clear the problem, use the MCR INSTALL NTL command to install the network loader.

#### **NTINIT -- NS: assignment failure**

The initializer could not assign one of its LUNs to the network device. There is no recovery procedure from this failure.

#### **NTINIT -- NW: assignment failure**

The initializer could not assign one of its LUNs to the PSI network device. There is no recovery procedure from this failure.

#### **NTINIT -- PLB allocation failure**

There is not enough free space in the dynamic storage region (DSR) to allocate the XPT physical link block database. You must run a system generation again, allocating enough free space in the DSR.

#### **NTINIT -- Router priority database allocation failure**

There is not enough free space in the dynamic storage region (DSR) to allocate the router priority database.

**NTINIT -- Routing control task not installed**

The routing control task must be installed to perform the SET EXECUTOR STATE ON command. Use the MCR INSTALL <netuic> RCP1 command to install the task.

**NTINIT -- X25 ACP (X25ACP) not installed**

The X25ACP must be installed to set module X25-SERVER to ON. Use the MCR INSTALL X25ACP command to install the X25ACP.

**Error Messages****10.9 NTL Error Messages**

The network loader (NTL) loads and unloads network software in response to NCP commands. When an NCP command executes locally, NTL error messages are displayed on the requesting terminal. When the command executes remotely, NTL error messages are displayed on the remote console terminal.

There are ten types of NTL error messages. The format of each is outlined below. The term *message-text* used in these sample formats is replaced with the actual message in the error message descriptions that follow. The italicized word *var* is replaced with a run-time variable in the actual message. Some of the variables are network processes, device driver names, controller names, line names, partition names, file names, vector values, and RSX device names. Variable fields are ignored in the alphabetization of error messages in this chapter.

**NTL -- *message-text***

**NTL -- *var message-text***

**NTL -- Config File -- *message-text***

**NTL -- *var*: Device Tables -- *message-text***

**NTL -- *var* Image File -- *message-text***

**NTL -- *var* Microcode File -- *message-text***

**NTL -- *var* Process File -- *message-text***

**NTL -- *var* Symbol Name -- *message-text***

**NTL -- *var* Template -- *message-text***

**NTL -- *var* Template File -- *message-text***

#### NOTE

Only recoverable NTL errors are listed in this guide. Any unlisted error message that you receive reflects an unrecoverable error. Call your Digital representative for assistance.

**NTL -- *dev-id*: allocation failure**

There is not enough space to allocate the RSX-11M device data structures (DCB, UCB, and SCB) for one of the network "devices." In this case, device does not refer to hardware. It refers to a software structure to which the user program assigns its LUNs and directs its QIOs.

**NTL -- CCB allocation failure**

There is not enough space for all the CCBs. It is not known how many were successfully allocated.

**NTL -- CETAB allocation failure**

There is not enough dynamic memory to allocate the CETAB database.

**NTL -- Channel *var* still active**

An attempt was made to issue a CLEAR PROCESS command for an LLC before all its lines were set to OFF. Set all the lines to OFF and repeat the command.

## **NTL**

---

### **Error Messages**

#### **NTL -- Circuit not in system**

An attempt was made to load a circuit that is not in the system.

#### **NTL -- Comm Exec has been deallocated**

The space for the Communications Executive has been deallocated.

#### **NTL -- Comm Exec incompatible with RSX-11M system**

The Communications Executive was built with the wrong RSX11M.STB file.

#### **NTL -- Comm Exec not loaded**

An attempt was made to perform an operation that requires the Communications Executive (CEX) to be loaded and it is not. Load the CEX with the NCP SET SYSTEM command.

#### **NTL -- Config File -- Binary buffer overflow**

The NTL end-of-task buffers are not large enough to hold the input. Install NTL with an INC option.

#### **NTL -- Config File -- Device *var* UNIBUS run mask missing**

The UNIBUS run mask is missing for the specified device.

**NTL -- Config File -- Event filter control block allocation failure**

There is not enough network pool and DSR to allocate an event filter block.

**NTL -- Config File -- Illegal CSR address**

The specified CSR address is not a legal value. Use the correct CSR address.

**NTL -- Config File -- LLC\$DF process extension too big**

One system process has an extension that makes it larger than the allowed 4K maximum.

**NTL -- Config File -- Non-UMR-mapped area is too large**

The non-UMR-mapped area specified in the PAR\$DF macro is too large for the network pool partition.

**NTL -- Config File -- Not enough contiguous UMRs**

Contiguous UMRs are needed to map the network buffer pool. Issue the SET SYSTEM command before you begin fragmenting the UMRs by running other jobs. If this fails, you can use VNP to load the network software into the operating system file instead of using NCP on the in-core image. VNP is not available on RSX-11M-PLUS systems.

## NTL

---

### Error Messages

#### **NTL -- Config File -- Object or remote block allocation failure**

There is not enough network pool or DSR to allocate the object or remote block. The line from CETAB where the failure occurred is displayed so that you can determine which object or remote block was not allocated.

#### **NTL -- Config File -- Open failure (-error code.)**

The system was unable to open the CETAB.MAC file. See the RSX system documentation for the meaning of the RSX I/O error code.

#### **NTL -- Config File -- Partition *var* busy**

The partition name specified for the network pool is busy.

#### **NTL -- Config File -- Partition *var* not common**

The partition specified for the network pool is not a common partition. Remove that partition and re-create it with the system-controlled attribute:

```
MCR>SET/NOMAIN=partition-name  
MCR>SET/MAIN=partition-name:aaaa:bbCOM
```

#### **NTL -- Config File -- Partition *var* not in system**

The partition specified for the network pool is not in the system.

#### **NTL -- Config File -- Partition *var* not system controlled**

NTL requires the partition to be system controlled. This error indicates that the partition was created as a task or common partition. Remove that partition and re-create it with the system-controlled attribute:

```
MCR>SET/NOMAIN=partition-name  
MCR>SET/MAIN=partition-name:aaaa:bbSYS
```

**NTL -- Config File -- Partition *var* too fragmented**

There is not enough contiguous space in the partition to load a process.

**NTL -- Config File -- Partition *var* too small for RDBs/SDBs**

This error pertains only to the network buffer pool partition. The network buffer pool is not large enough for the large and small data buffers. Use CFE to reduce the buffer counts and/or sizes or to re-create a larger partition.

**NTL -- Config File -- Read failure (*-error code.*)**

The system was unable to read the CETAB.MAC file. See the RSX system documentation for the meaning of the RSX I/O error code.

**NTL -- Config File -- Secondary CSR off line**

The specified secondary CSR (from CETAB.MAC) is off line.

**NTL -- Config File -- Sub-PCB allocation failure**

There is not enough space for a partition control block for the network buffer pool.

**NTL -- Config File -- SVC descriptor allocation failure**

There was not enough DSR or network pool to allocate the SVC descriptor.

**NTL -- Config File -- UMR block allocation failure**

There is not enough space for a UMR assignment block.

## NTL

---

### Error Messages

#### NTL -- Config File -- Vector not in system

The controller's vector address specified in the configuration file does not exist in your system's vector space.

**NTL -- *var* Device I/O error, CSR = *var*, RAM address = *var* was written, *var* was read**

An attempt to load microcode for a KMC device failed due to an input/output failure. One possible cause could be a hardware failure.

#### NTL -- Device off line

An attempt was made to load a device that is off line.

#### NTL -- Device *var* off line

An attempt was made to load a device that is off line.

#### NTL -- Device *dev-id*: still active

An attempt was made to clear an LLC process when the associated device is still active.

#### NTL -- Device *dev-id*: still mounted

An attempt was made to clear an LLC process when the device associated with it is still mounted.

#### NTL -- *dev-id*: driver already resident

The executive data structure for the driver associated with the LLC process being loaded already exists.

#### NTL -- *var* ICB allocation failure

There is not enough space in the dynamic storage region (DSR) to allocate an interrupt control block (ICB).

**NTL -- *var* Image File -- Label block read failure (-error code.)**

The system is unable to read a label block. See the RSX system documentation for the meaning of the RSX I/O error code.

**NTL -- *var* Image File -- Open failure (-error code.)**

The attempt to open the specified file failed. See the RSX system documentation for the meaning of the RSX I/O error code.

**NTL -- *var* Image File -- Read failure (-error code.)**

The attempt to read a record from a specified file failed. See the RSX system documentation for the meaning of the RSX I/O error code.

**NTL -- Incompatible with Comm Exec**

NTL has been built with the wrong CEX.STB file. Rebuild NTL with the correct CEX.STB file.

**NTL -- Incompatible with RSX-11M system**

NTL has been built with the wrong RSX11M.STB file. Rebuild NTL with the correct RSX11M.STB file.

**NTL -- *var* KRB allocation failure**

There is not enough space in either the network pool or the dynamic storage region for the controller request block (KRB).

**NTL -- Line not in system**

An attempt was made to set a line that does not exist.

## NTL

---

### Error Messages

#### **NTL -- Line still active**

An attempt was made to clear a line that was still ON. Set the line to OFF and reenter the command.

#### **NTL -- Line *var* still ready**

An attempt was made to perform a CLEAR PROCESS on a DLC or a DDM controller before all of its circuits were turned off and unloaded. Set the circuits to OFF and reenter the command.

#### **NTL -- *var* Line table allocation failure**

There is not enough space in DSR for either a device line table or an LLC process database.

#### **NTL -- MUX update only**

This is a reminder that the vector and the CSR for the device specified in the command line have not been changed. All lines on a multiplexer device are loaded when any line on the device is loaded. Therefore, all subsequent loads for other lines on this device do not change the vector or the CSR.

#### **NTL -- *var* Process File -- Incompatible with Comm Exec**

The process has been built with the wrong CEX.STB file.

#### **NTL -- *var* Process File -- Label block read failure (-*error-code*.)**

An attempt to read a label block for a process task image file failed due to an input/output error. See the RSX system documentation for the meaning of the RSX I/O error code.

**NTL -- *var* Process File -- Open failure (-*error-code*.)**

An attempt to open the task image file for the specified process failed due to an input/output error. See the RSX system documentation for the meaning of the RSX I/O error code.

**NTL -- *var* Process File -- Partition *var* busy**

The partition is occupied by either another network process or an RSX-11M resident library. The current occupant must be unloaded.

**NTL -- *var* Process File -- Partition *var* not common**

NTL requires the partition to be common. The partition has been created as either task or system controlled. Remove the partition and re-create it with the common attribute:

```
MCR>SET/NOMAIN=partition-name  
MCR>SET/MAIN=partition-name:aaaa:bbb:COM
```

**NTL -- *var* Process File -- Partition *var* not in Exec space**

The Communications Executive (CEX) partition must be contained completely in the first 20K words of physical memory. This error indicates that the ending address of the CEX partition is larger than this limit. The CEX contains an end-of-partition buffer pool. If you exceed the limit by a small amount (100 or 200 octal), you can try to reduce the CEX's partition size. Otherwise, you should repeat the NETGEN.

**NTL -- *var* Process File -- Partition *var* not in system**

The partition specified in your command does not exist. Either create the partition that NTL is seeking, or rebuild the component to change the partition assignment.

## NTL

---

### Error Messages

#### **NTL -- *var* Process File -- Partition *var* sub-PCB allocation failure**

There is not enough space for a partition control block (PCB) for the named process.

#### **NTL -- *var* Process File -- Partition *var* too fragmented**

This error pertains to system-controlled partitions only. The size of the largest free space in the partition is smaller than the size of the process being loaded. The current occupants of the partition must be rearranged to make a free space of the size needed. NTL does not activate the RSX-11M shuffler task. Manual recovery is necessary.

#### **NTL -- *var* Process File -- Partition *var* too small**

This error pertains to the common partition only. The partition is smaller than the size of the process being loaded. The partition must be created with a larger size. Refer to the TKB map for the process to determine the size needed.

#### **NTL -- *var* Process File -- Read failure (*-error-code.*)**

An attempt to read a record from the process task image file failed due to an input/output error. See the RSX system documentation for the meaning of the RSX I/O error code.

#### **NTL -- Process may not be unloaded**

An attempt was made to unload a process that is required for network operation and cannot be unloaded.

#### **NTL -- Process not in system**

An attempt was made to set a nonexistent process.

#### **NTL -- Reconfiguration device not in system**

The reconfiguration device for the network devices is not loaded. Load the device with the DECnet process called DLX.

**NTL -- *var* Symbol Table File -- Open failure (-*error-code*.)**

An attempt to open the system table file for the specified process failed due to an input/output error. See the RSX system documentation for the meaning of the RSX I/O error code.

**NTL -- *var* Symbol Table File -- Read failure (-*error-code*.)**

An attempt to read a record from the specified system table file failed due to an input/output error. See the RSX system documentation for the meaning of the RSX I/O error code.

**NTL -- System not built with network support**

This message needs no explanation.

**NTL -- *var* Template -- Library block allocation failure**

There is no space in the network buffer pool.

**NTL -- *var* Template -- Resource allocation failure**

There is not enough space for the allocation size needed. (Another error message follows that contains the allocation type being attempted.)

**NTL -- *var* Template -- Vector *var* not available**

The indicated vector is assigned to another user. The vector must be deassigned from the current user.

**NTL -- *var* Template -- Vector *var* not in system**

The vector specified in the command exceeds the maximum value assigned at NETGEN.

## **NTL**

---

### **Error Messages**

#### **NTL -- *var* Template File -- Open failure (-*error-code*.)**

An attempt to open the template file for the specified process failed due to an input/output error. See the RSX system documentation for the meaning of the RSX I/O error code.

#### **NTL -- *var* Template File -- Read failure (-*error-code*.)**

An attempt to read a record from the specified template file failed due to an input/output error. See the RSX system documentation for the meaning of the RSX I/O error code.

#### **NTL -- Timer cells not found**

NTL could not find the process timer cells in the AUX process.

#### **NTL -- Vector not in system**

The vector specified in the command line does not exist in your system's vector space. The upper limit of the vector space is determined during SYSGEN. Repeat both SYSGEN and NETGEN.

#### **NTL -- *var* Vector set-up error**

Some other user has stolen one or more of the interrupt vectors. This occurred after NTL first verified that the interrupt vectors were available, but before NTL built the line table and pointed the vector at the table. The vector must be deassigned from the other user.

#### **NTL -- \* WARNING \* -- Failed to reinitialize HRC**

The network loader was unable to successfully reinitialize HRC after loading a network device. This could affect the use of the system utility ...CON.

#### **NTL -- \* WARNING \* -- System name changed to "*name*"**

The system name specified during SYSGEN does not match the node name specified during NETGEN. The system name has been changed to the node name in NETGEN and will remain so until the system is rebooted.

## 10.10 TRI Error Messages

TRI displays an error message on your terminal when it detects one of the errors described below. These messages reflect operational conditions. While most result in termination of the trace interpreter task, a few are for information only.

### **TRI -- All frames were excluded by /LI or /CH**

(Informational) There were no frames traced on the lines or channels that you specified with the /LI or /CH switch.

### **TRI -- Cannot open input file**

(Fatal) One of the following conditions exists:

- The specified device does not exist.
- The volume is not mounted.
- A problem exists with the device.
- The specified UIC does not exist.
- The specified file does not exist.
- You do not have read access privileges.

## **TRI**

---

### **Error Messages**

#### **TRI -- Cannot open output file**

(Fatal) One of the following conditions exists:

- The specified device does not exist.
- The volume is not mounted.
- A problem exists with the device.
- The specified UIC does not exist.
- The volume is full or the device is write protected.
- You do not have write access privileges to the UIC.

#### **TRI -- Command I/O error**

(Fatal) The system returned an error when TRI attempted to read a command line.

#### **TRI -- Error closing output file**

(Fatal) TRI failed to close the trace-listing file.

#### **TRI -- Error reading input file**

(Fatal) The system returned an error when TRI tried to read the trace input file. This could be caused by a device error or a bad block on the volume.

**TRI -- Error writing output file**

(Fatal) The system returned an error when TRI attempted to write a line into the trace-listing file. This could be caused by a problem with the device, a full volume, or a bad block on the volume.

**TRI -- Failed to spool output file**

(Fatal) The system failed to spool the trace-listing file.

**TRI -- Illegal character in *filespec***

(Fatal) One of your file specifications is incorrect.

**TRI -- Illegal command string**

(Fatal) TRI detected an error in the syntax of the TRI command string.

**TRI -- Illegal switch or switch value .**

(Fatal) You have specified an invalid switch in the TRI command string, or you have given an invalid value to a switch.

**TRI -- Incompatible switches**

(Fatal) You have specified an illegal combination of switches. Check that you have not specified any switches other than /SP or /-SP with the /-IN switch.

**TRI -- Input file is not a trace dump file**

(Fatal) The input file you have specified is not a trace dump file produced as output from an NCP SET TRACE command.

**TRI -- Trace file is empty**

(Informational) The trace file specified as the input file is empty.

## Error Messages

### 10.11 VNP Error Messages

#### **VNP -- Account field too long**

The account field is limited to 16 characters.

#### **VNP -- Alias not in system**

An attempt was made to reference an alias that is not in the system.

#### **VNP -- Attempt to reference block number that is not within the system image file *block-number* BIAS:*n* ADDRESS:*m***

An attempt was made to reference the specified block number, which is not in the system image file. This error is often caused by building a system image file that is smaller than the real memory size while top-down loading. Use CFE to disable top-down loading.

#### **VNP -- CETAB data structures are inconsistent**

The data structures in the Communications Executive configuration file (CETAB) are inconsistent. Possibly the CETAB.TSK file is corrupt. Perform another NETGEN to obtain an uncorrupt CETAB.TSK file.

#### **VNP -- Circuit not in system**

The specified circuit is not in the system.

#### **VNP -- COMEXEC symbol missing from symbol table *symbol-id***

The specified symbol is not in the Communications Executive symbol table.

#### **VNP -- Communications Executive load failed**

VNP could not successfully load the Communications Executive.

**VNP -- Communications Executive symbol table file open failure** *error-code filename*

The Communications Executive symbol table file (*filename*) failed to open. The error code is returned by the operating system's file services. All the network files must reside in the UIC defined by the NETUIC in the system image file. Check the NETUIC in the system image file by using the VMR SET /NETUIC command.

**VNP -- Configuration file read error**

VNP could not read a record from the CETAB source file.

**VNP -- Controller number too large**

The controller number on the line identification is illegal.

**VNP -- Error parsing file name - *filename***

The file name of the operating system disk image file is invalid.

**VNP -- Event cannot be filtered**

This message indicates that the ability to enable or disable one of the specified events is not implemented.

**VNP -- Executive symbol missing from symbol table** *symbol-id*

The specified symbol is not in the Communications Executive symbol table.

**VNP -- Executive symbol table file open failure** *error-code filename*

The RSX executive symbol table file (*filename*) failed to open. The error code is returned by the operating system's file services. The system image file, its symbol table, and all network files must reside under the same UIC.

## VNP

---

### Error Messages

#### **VNP -- Failed to close system image file *error-code***

The system image file failed to close. The error code is returned by the operating system's file services.

#### **VNP -- Failed to get command line**

VNP could not read a command line.

#### **VNP -- Failed to load a process**

VNP did not successfully load a process that was marked for load. You can list all processes to find out which process did not load.

#### **VNP -- Failed to load a system line**

VNP did not successfully load a line that was marked for load. You can list all lines to find out which line did not load.

#### **VNP -- Function not implemented**

An attempt was made to execute an unimplemented VNP function.

#### **VNP -- Illegal event code**

The event code given is invalid.

#### **VNP -- Illegal executive size *nnn***

The executive size (*nnn*) is illegal. This is probably caused by a corrupt system image file.

#### **VNP -- Illegal line cost**

The line cost must be in the range of 1 to 1022.

#### **VNP -- Illegal multipoint active ratio**

The multipoint active ratio must be in the range of 1 to 255.

**VNP -- Illegal multipoint dead ratio**

The multipoint dead ratio must be in the range of 1 to 255.

**VNP -- Illegal node address**

The node address must be in the range of 1 to 1023.

**VNP -- Illegal object type**

The object type numbers must be in the range of 1 to 255.

**VNP -- Illegal tributary address**

The tributary addresses must be in the range of 1 to 255.

**VNP -- Illegal unit CSR address**

The unit CSR address is invalid.

**VNP -- Insufficient space in template-processing buffer (TMP\$\$A *psect*)  
Current size is *n* bytes, but it should be expanded to at least *m* bytes**

VNP's processing buffer is *n* bytes long, but it requires *m* bytes for the configuration. Edit VNPGEN.CMD and extend the buffer size. Then perform another task build on VNP and try again.

**VNP -- Invalid hello timer value**

The value specified for the hello timer is invalid. The value must be within the range of 1 to 65,535.

**VNP -- Invalid line identification**

The line identification is not in the proper format.

## **VNP**

---

### **Error Messages**

#### **VNP -- Invalid node option**

Certain node options are valid only on the executor node, while others are not allowed. You must verify that all options selected are valid for the node specified.

#### **VNP -- Invalid parameter grouping**

The combination of parameters specified is invalid. For example,

```
VNP>SET LOGGING CONSOLE EVENTS 2.0 CIRCUIT DMC-O NODE A
```

is invalid because you cannot specify two entities to qualify events that are to be logged. You can specify the circuit or the node, but not both. If you want two entities to be logged, you must enter two separate commands.

#### **VNP -- I/O error accessing system image file *error-code***

A disk I/O error occurred when VNP accessed the system image file. The error code is returned by the operating system's file services.

#### **VNP -- KMC microcode load failed**

VNP failed to load the KMC microcode.

#### **VNP -- Line name too long**

Line names must be 1 to 3 Radix-50 characters.

#### **VNP -- Line not assigned to XPT or DLX**

An attempt was made to access a line that is not on a network communications device assigned to XPT or DLX.

#### **VNP -- Line not in system**

The specified line is not in the system.

**VNP -- Line parameter not applicable**

The line option is valid on SET NODE requests only if the node identification is a name.

**VNP -- Maximum controllers limited to 64**

The maximum controller count for the SET PROCESS request is limited to 64.

**VNP -- Maximum lines limited to 64**

The maximum line count for the SET PROCESS request is limited to 64.

**VNP -- NETACP must be "fixed" in RSX-11S systems**

NETACP is not fixed in an RSX-11S System. Fix NETACP using VMR before starting VNP.

**VNP -- NETACP not installed**

NETACP is not installed. Install NETACP using VMR before restarting VNP.

**VNP -- Network buffer pool is too small**

The network buffer pool is not large enough to load the system software. You will have to use VMR on your system image to change the size of the network pool partition.

**VNP -- Network initializer not installed**

The network initializer (NTINIT) is not installed. Install NTINIT with VMR before restarting VNP.

**VNP -- Network pool partition not in the system**

VNP could not find the network pool partition as specified during NETGEN.

## **VNP**

---

### **Error Messages**

#### **VNP -- Node in wrong state**

The node is not in the proper state to perform the requested function.

#### **VNP -- Node name too long**

Node names are limited to a maximum of 6 characters.

#### **VNP -- Node not in system**

An attempt was made to reference a node that is not in the system.

#### **VNP -- Non-UMR-mapped area is too large**

The non-UMR-mapped portion of the network pool is larger than the entire network pool.

#### **VNP -- Not enough contiguous UMRs to map network pool**

The network pool needs too many UMRs to be loaded. You should try to reduce the size of the network pool.

#### **VNP -- NTINIT must be "fixed" in RSX-11S systems**

NTINIT is not fixed in an RSX-11S system. Fix NTINIT using VMR before restarting VNP.

#### **VNP -- Object not in system**

An attempt was made to reference an object that is not contained in the system.

#### **VNP -- Object/remote block allocation failure**

There is not enough network pool or dynamic storage space to allocate a remote name block or an object block.

**VNP -- Object task name too long**

Object task names must be 1 to 6 Radix-50 characters.

**VNP -- Open failure on configuration file**

VNP could not open the CETAB source file.

**VNP -- Open failure on system image file *error-code filename***

The system image file (*filename*) cannot be opened. The error code is returned by the operating system's file services.

**VNP -- Owner process name too long**

Process names must be 1 to 3 Radix-50 characters.

**VNP -- Partition name too long**

The partition name must be 1 to 6 Radix-50 characters.

**VNP -- Password field too long**

The password field is limited to 8 characters.

**VNP -- PLB allocation failure**

VNP was not able to allocate enough space for the XPT databases.

**VNP -- Process name too long**

Process names must be 1 to 3 Radix-50 characters.

**VNP -- Resource error trying to allocate CCBs**

There is not enough dynamic storage space in your system to allocate the control buffers.

## VNP

---

### Error Messages

#### **VNP -- Reverse mapping table inconsistent**

The reverse mapping table has been corrupted.

#### **VNP -- RSX-11M system pool allocation failure**

The system pool is empty. Use VMR to increase the size of the system pool.

#### **VNP -- Supposed system image file not actually a system image *filename***

The indicated file is not a system image file.

#### **VNP -- Syntax error**

The command line is not correct.

#### **VNP -- System image does not contain a GEN partition**

The default partition for network software is GEN. The system image file does not contain a GEN partition.

#### **VNP -- System image file corrupted because VNP was terminated abnormally *filename***

A previous execution of VNP terminated abnormally and may have corrupted the system image file (*filename*). Recopy the system image file before restarting VNP.

#### **VNP -- System image must be extended to *n* blocks if the GEN partition is to be contained within the image**

The default partition GEN is not within the system image. Use the PIP command for copying files with the /BL:*n*. switch to enlarge the output file.

**VNP -- The Communications Executive data structures are too large for VNP's internal buffer. The buffer (CET\$SA *psect*) can be expanded to 2776 bytes maximum (from the current *n* bytes), but this configuration needs *m* bytes. This error is not recoverable without source modifications.**

VNP's internal buffers cannot accommodate the Communications Executive configuration file (CETAB). Reduce the size of the configuration.

**VNP -- Too many object task copies**

A maximum of 63 object task copies is allowed.

**VNP -- Tributary number too large**

Tributary numbers must be in the range of 1 to 255.

**VNP -- UMR block allocation failure**

There is not enough dynamic storage space to allocate a UMR block.

**VNP -- Unable to dynamically allocate network buffer pool**

VNP could not allocate enough partition space to load the network pool.

**VNP -- Unit number too large**

The unit number on the line identification is illegal.

**VNP -- Unspecified error during CONFIG file scan**

VNP could not successfully parse all the definitions in the CETAB source file.

**VNP -- User identification field too long**

The user identification field is limited to a maximum of 16 characters.

## **VNP**

---

### **Error Messages**

#### **VNP -- XPT channel table inconsistent**

The XPT channel table has been corrupted.

#### **VNP -- XPT database not allocated**

The XPT process is loaded but the data structures are not.

#### **VNP -- XPT process not loaded**

The XPT process is not loaded.

# A

## Process Names

Table A-1 lists processes alphabetically by type, including device driver processes and software protocol-related processes for DECnet and PSI.

**Table A-1: DECnet-RSX Processes**

<b>Process Name</b>	<b>Function</b>
<b>Communications Executive processes</b>	
AUX	Auxiliary process
DLX	Direct Line Access Controller
EVL	Event Logger process
<b>DECnet device drivers</b>	
DH	DHU-11 driver
DHV	DHV-11 driver
DL	DL-11 driver
DLV	DLV-11 driver
DMC	DMC-11/DMR-11 driver
DMP	DMP-11 driver
DMV	DMV-11 driver
DPV	DPV-11 driver
DU	DU-11 driver
DUP	DUP-11 driver
DUV	DUV-11 driver

**Table A-1 (cont.): DECnet-RSX Processes**

<b>Process Name</b>	<b>Function</b>
DV	DV-11 driver
DZ	DZ-11 driver
DZV	DZV-11/DZQ-11 driver
KDP (KMC/DUP)	KDP-11 driver
KDZ (KMC/DZ)	KDZ-11 driver
PCL	PCL driver
UNA	DEUNA driver
QNA	DEQNA driver

**DECnet processes**

DCP	Implements DDCMP; included for all systems with devices other than DMC-11, DMP-11, DMR-11, DMV-11, PCL-11B, DEUNA, and DEQNA
EPM	Ethernet Protocol Manager; included for all systems with Ethernet devices
ECL	End Communications layer process
XPT	Routing layer process
NCT	Network Command Terminal process
RTH	Remote Terminal Host process
LAT	LAT device driver process

**PSI device drivers**

KMX	KMX-11 driver
KMY	KMY-11 driver
SDP	DUP-11 driver
SDV	DPV-11 driver

**PSI processes**

DLM	Data-link-mapping process
LAB	X.25 level 2 LAPB protocol process
NW	PSI user interface process
PLI	X.25 level 3 (packet level) process

## **B**

### **Object Type Codes**

Table B-1 defines valid object type code values and describes their process type for network management. The values are expressed as decimal byte values. Digital reserves the right to add object types or to make changes to the descriptor formats used by the object types.

**Table B-1: Object Type Codes**

<b>Object Type Code</b>	<b>Process Type</b>
0	General task, user program
1	File Access Listener -- FAL/DAP, Version 1
2-4	Reserved for DECnet use
5	RSX-11M Task Control -- Version 1
6-14	Reserved for DECnet use
15	RSX-11M Task Control -- Version 2
16	TLK utility
17	File Access Listener -- FAL/DAP, Version 4
18	Host Task Loader
19	Network Information and Control Exchange
20	RSTS/E media transfer program
21-22	Reserved for DECnet use
23	Network terminal handler
24	Reserved for DECnet use
25	Network management loopback mirror

(continued on next page)

**Table B-1 (cont.): Object Type Codes**

<b>Object Type Code</b>	<b>Process Type</b>
26	Network management event receiver
27-28	Reserved for DECnet use
29	PHONE utility
30-28	Reserved for DECnet use
42	Heterogeneous terminal host
43-62	Reserved for DECnet use
63	RSX DECnet test tool
64-127	Reserved for DECnet use
128-255	Reserved for customer extensions

# Index

## A

Abbreviating command keywords,  
1-7

Access control information,  
to specify for remote node,  
1-102

Access control verification,  
to specify in permanent  
database, 1-36, 1-57  
to specify in volatile database,  
1-98, 1-160, 1-162, 1-166

Alias node name,  
to display alias information,  
1-129  
to remove, 1-14  
to specify, 1-95

Areas,  
to display area information,  
1-130

## B

Buffers,  
*see also* System commands

## C

CCBs,  
bar graphs of in NTD displays,  
7-6

CCR (console carrier requester), 2-1

CCR,  
error messages, 10-3  
to use CCR,, 2-1

CETAB.MAC,  
errors in, 10-5  
UIC for, 1-1

CFE,  
*see also* Utility commands  
command descriptions, 1-13  
error messages, 10-5  
general description, 1-1  
KILL command, 1-3  
to exit, 1-2  
to invoke, 1-1

Circuit commands,  
CLEAR CIRCUIT (to zero PSI  
circuit counter timer),  
1-15  
DEFINE CIRCUIT, 1-32  
LIST CIRCUIT, 1-64  
LOOP CIRCUIT, 1-80  
PURGE CIRCUIT, 1-86  
SET CIRCUIT, 1-96  
SET CIRCUIT (for RSX-11S),  
1-162  
SHOW CIRCUIT, 1-131  
SHOW CIRCUIT (for RSX-11S),  
1-166  
ZERO CIRCUIT, 1-152  
CLEAR commands (descriptions),  
1-14 to 1-31

Clear,  
    how to clear the PSI circuit  
        counter timer, 1-86  
    how to clear the PSI line  
        counter timer, 1-87  
Comment line format, 1-7  
Console carrier requester,  
    *see* CCR  
Continuation lines within  
    commands, 1-7  
Counter timer,  
    to zero PSI circuit counter  
        timer, 1-15  
    to zero PSI line counter timer,  
        1-18  
Crash analysis,  
    crash causes, 6-3  
    obtaining a crash dump, 6-3  
    using NDA,  
        *see also* NDA, 6-1

## D

DECnet-RSX,  
    device drivers and processes  
        (table), A-1  
DEFINE commands (descriptions),  
    1-32 to 1-62  
Device drivers and processes  
    (table), A-1  
Display commands, *see* LIST and  
    SHOW commands  
Down-line loading,  
    NCP load commands,  
        LOAD NODE, 1-75  
        LOAD VIA, 1-78

## E

Error messages,  
    for LCP *see* DECnet-RSX Release  
        Notes  
    for CCR, 10-3  
    for CFE, 10-5  
    for EVF, 10-24  
    for HLD, 10-29  
    for NCP, 10-38  
    for NCP (message formats),  
        10-36  
    for NDA, 10-56

Error messages (Cont.)  
    for NTD, 10-61  
    for NTDEMO, 10-61  
    for NTINIT, 10-62  
    for NTL, 10-66  
    for TRI, 10-79  
    for VNP, 10-82  
ESCAPE key usage for help  
    information, 1-8  
EVF,  
    error messages, 10-24  
Executor commands,  
    CLEAR EXECUTOR, 1-16  
    CLEAR EXECUTOR NODE, 1-17  
    DEFINE EXECUTOR, 1-36  
    LIST EXECUTOR, 1-65  
    LOOP EXECUTOR, 1-83  
    LOOP EXECUTOR (for RSX-11S),  
        1-160  
    SET EXECUTOR, 1-98  
    SET EXECUTOR HOST (for RSX-11S),  
        1-163  
    SET EXECUTOR NODE, 1-102  
    SHOW EXECUTOR, 1-132  
    SHOW EXECUTOR (for RSX-11S),  
        1-167  
    ZERO EXECUTOR, 1-153  
    ZERO EXECUTOR (for RSX-11S),  
        1-173

## F

File Transfer Spooler, *see* FTS  
FTS (File Transfer Spooler), 8-1  
FTS,  
    components (figure), 8-3  
    components (table), 8-2  
    to list and purge URB files,  
        8-9  
    to list FTSQUE and FTS jobs,  
        8-8  
    to purge FTSSYS.LOG and FAL.LOG,  
        8-7

## H

Help information, 1-8  
HLD,  
    error messages, 10-29

## K

KDA (KMX/KMY dump analyzer), 4-1

KDA,

    KDA command, 4-2

    KDA command switches (table),

        4-2

    to exit, 4-1

    to invoke, 4-1

KILL command (CFE), 1-3

KMX-Dump command, 1-63

KMX/KMY dump analyzer, *see* KDA

## L

LAT control program, *see* LCP

LCP (LAT control program), 5-1

LDBs,

    bar graphs of in NTD displays,

        7-6

Line commands,

    CLEAR LINE (to unload line/zero

        PSI line counter timer),

        1-18

    DEFINE LINE, 1-40

    LIST LINE, 1-66

    LOOP LINE, 1-85

    PURGE LINE, 1-87

    SET KNOWN LINES, 1-103

    SET LINE, 1-106

    SET LINE (for RSX-11S), 1-164

    SHOW LINE, 1-133

    SHOW LINE (for RSX-11S), 1-168

    ZERO LINE, 1-154

    ZERO LINE (for RSX-11S), 1-174

Line,

*see also* Line commands and line

        entries under PSI

    to unload, 1-18

LIST commands (descriptions),

    1-64 to 1-74

LOAD commands (descriptions),

    1-75 to 1-79

Logging commands,

    CLEAR LOGGING, 1-19

    DEFINE LOGGING, 1-44

    LIST LOGGING, 1-67

    PURGE LOGGING, 1-88

    SET LOGGING, 1-110

Logging commands (Cont.)

    SET LOGGING (for RSX-11S),

        1-165

    SHOW LOGGING, 1-134

    SHOW LOGGING CONSOLE (for

        RSX-11S), 1-169

Logging,

    to display logging information,

        1-67, 1-134

    to display logging information

        (for RSX-11S), 1-169

    to specify logging parameters,

        1-110

LOOP command (RSX-11S), 1-160

LOOP commands (descriptions),

    1-80 to 1-85

Loopback tests,

*see also* LOOP commands

## N

NCP,

*see also* Utility commands

    command descriptions (full set),

        1-13

    command descriptions (RSX-11S

        only), 1-159

    error message formats, 10-36

    error messages (listing of),

        10-38

    general description, 1-3

    remote execution, 1-4

    to exit, 1-4

    to invoke, 1-3

NDA (Network Crash Dump Analyzer),

    6-1

NDA,

    analysis control switches, 6-8

    analysis control switches

        (table), 6-9

    command syntax, 6-5

    crash causes, 6-3

    error messages, 10-56

    file default values (table),

        6-6

    function control switches, 6-8,

        6-11

    function control switches (table),

        6-11

## NDA (Cont.)

- operation of, 6-2
- requirements for use, 6-3
- to obtain a crash dump, 6-3
- to run NDA as an installed task, 6-4
- to run NDA as an uninstalled task, 6-4
- to use indirect command files, 6-8

## NETUIC,

- to set in system image file, 1-5

## Network Crash Dump Analyzer, *see* NDA

## Network Display Program, *see* NTD

## Network initializer, *see* NTINIT

## Network loader, *see* NTL

## Network management,

- see also* Utility commands
- command descriptions, 1-1
- command usage, 1-6

## Node commands,

- see also* Executor commands
- CLEAR NODE, 1-25
- DEFINE NODE, 1-54
- LIST NODE, 1-71
- LOAD NODE, 1-75
- LOAD VIA, 1-78
- LOOP NODE, 1-83
- LOOP NODE (for RSX-11S), 1-160
- PURGE NODE, 1-92
- SET NODE, 1-119
- SET NODE CIRCUIT, 1-122
- SHOW NODE, 1-141
- SHOW NODE (for RSX-11S), 1-170
- TRIGGER NODE, 1-148
- TRIGGER VIA, 1-150
- ZERO NODE, 1-157
- ZERO NODE (for RSX-11S), 1-175

## Node,

- to bootstrap, 1-148
- to down-line load a remote node, 1-75, 1-78

## NTD (Network Display Program), 7-1

## NTD,

- command mode commands, 7-3, 7-5
- command mode commands (table), 7-5

## NTD (Cont.)

- command modes, 7-3
- display types, 7-1
- error messages, 10-61
- format of area summary display, 7-11
- format of area summary display (example), 7-12
- format of default resource display (example), 7-9
- format of node summary display, 7-10
- format of node summary display (example), 7-11
- format of old resource display (example), 7-9
- format of resource display, 7-6
- immediate mode commands, 7-3
- immediate mode commands (table), 7-3
- scrolling commands, 7-3
- to invoke, 7-2

## NTDEMO error messages, 10-61

## NTINIT error messages, 10-62

## NTL error messages, 10-66

## O

### Object commands,

- CLEAR OBJECT, 1-28
- DEFINE OBJECT, 1-57
- LIST OBJECT, 1-72
- PURGE OBJECT, 1-94
- SET OBJECT, 1-123
- SHOW OBJECT, 1-143

### Object type codes,

- table, B-1

### Object,

- see also* Object commands

## P

### Password,

- receive password, to remove, 1-16
- receive password, to specify, 1-98
- transmit password, to remove, 1-16

## Password (Cont.)

transmit password, to specify,  
1-98

## Process commands,

CLEAR PROCESS, 1-29  
DEFINE PROCESS, 1-59  
LIST PROCESS, 1-73  
SET PROCESS, 1-125  
SHOW PROCESS, 1-144

## Process,

*see also* Process commands  
names and types (table), A-1

## PSI,

device drivers and processes  
(table), A-1  
to zero the circuit counter  
timer, 1-15  
to zero the DTE counter timer,  
1-23, 1-90  
to zero the line counter timer,  
1-18

PURGE commands (descriptions),  
1-86 to 1-94

## Q

QUE (Queue Manager), 8-1

## QUE,

commands, 8-3  
components (figure), 8-3  
components (table), 8-2  
to invoke, 8-3  
to list and purge URB files, 8-9

Queue management functions, 8-7

Queue Manager, *see* QUE

## R

Receive password, *see* Password

Remote command execution,

of NCP commands, 1-4  
to initiate, 1-102  
to return control to local node,  
1-17

## RSX-11S,

NCP command descriptions, 1-159  
to down-line load a remote node,  
1-75, 1-78

## S

## SDBs,

bar graphs of in NTD displays, 7-6

SET commands (descriptions), 1-95  
to 1-128

SET commands for RSX-11S

(descriptions), 1-162 to 1-165

SET EXECUTOR HOST command (for  
RSX-11S), 1-163

SET EXECUTOR NODE command,  
format, 1-102

usage, 1-4

SHOW commands (descriptions),  
1-129 to 1-146

SHOW commands for RSX-11S  
(descriptions), 1-166 to 1-171

## System commands,

CLEAR SYSTEM, 1-30

DEFINE SYSTEM, 1-61

LIST SYSTEM, 1-74

SET SYSTEM, 1-127

SHOW SYSTEM, 1-145

SHOW SYSTEM (for RSX-11S),  
1-171

ZERO SYSTEM, 1-158

ZERO SYSTEM (for RSX-11S),  
1-176

## T

TELL command,

format, 1-147

usage, 1-4

## Trace commands,

CLEAR TRACE, 1-31

SET TRACE, 1-128

SHOW TRACE, 1-146

Trace facility, 9-1

Trace facility, *see also* TRI

Trace interpreter task, *see* TRI

Trace output information, 9-4

Transmit password, *see* Password

TRI (trace interpreter task), 9-1

## TRI,

command format, 9-2

command switches (table), 9-3

contents of trace output, 9-4

TRI (Cont.)  
error messages, 10–79  
to exit TRI, 9–2  
to invoke TRI, 9–1  
TRIGGER command,  
formats, 1–148 to 1–151

## U

URB,  
to list and purge URB files, 8–9  
Utility commands,  
*see also* CFE, NCP, VNP, and  
Command summaries  
abbreviating a keyword, 1–7  
command prompting (for help), 1–8  
comment lines, 1–7  
continuing a command line, 1–7  
descriptions of (for RSX–11S),  
1–159  
descriptions of,  
formats, 1–13  
introduction, 1–10  
parameter definitions, 1–11  
error handling, 1–9  
exit status, 1–9  
format, 1–6  
HELP command usage, 1–8

## V

VMR,  
used to set NETUIC in system  
image file, 1–5

VNP,  
*see also* Utility commands  
command descriptions, 1–13  
error messages, 10–82  
general description, 1–5  
to exit, 1–6  
to invoke, 1–5  
use of NETUIC, 1–5

## X

X.25 access module,  
to remove destination name,  
1–89  
to remove destination name(s),  
1–22

X.25 access module (Cont.)  
to specify a destination name,  
1–46, 1–113  
XDT, 6–3

## Z

ZERO commands (descriptions),  
1–152 to 1–158  
ZERO commands for RSX–11S  
(descriptions), 1–172 to 1–176  
Zero,  
how to zero the DTE counter  
timer, 1–23, 1–90  
how to zero the PSI circuit  
counter timer, 1–15  
how to zero the PSI line counter  
timer, 1–18

READER'S COMMENTS

What do you think of this manual? Your comments and suggestions will help us to improve the quality and usefulness of our publications.

Please rate this manual:

	Poor			Excellent	
Accuracy	1	2	3	4	5
Readability	1	2	3	4	5
Examples	1	2	3	4	5
Organization	1	2	3	4	5
Completeness	1	2	3	4	5

Did you find errors in this manual? If so, please specify the error(s) and page number(s).

---

---

---

---

General comments:

---

---

---

---

Suggestions for improvement:

---

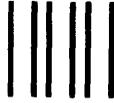
---

---

---

Name \_\_\_\_\_ Date \_\_\_\_\_  
Title \_\_\_\_\_ Department \_\_\_\_\_  
Company \_\_\_\_\_ Street \_\_\_\_\_  
City \_\_\_\_\_ State/Country \_\_\_\_\_ Zip Code \_\_\_\_\_

DO NOT CUT FOLD HERE AND TAPE



NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY LABEL**  
FIRST CLASS PERMIT NO 33 MAYNARD MASS

POSTAGE WILL BE PAID BY ADDRESSEE

**digital**

**SOFTWARE DOCUMENTATION**  
1925 ANDOVER STREET TWO/E07  
TEWKSBURY, MASSACHUSETTS 01876



DO NOT CUT FOLD HERE



CUT ALONG DOTTED LINE

