DECnet for OpenVMS Network Management Utilities

Order Number: AA–PV61A–TK

May 1993

This document contains information about commands for the Network Control Program (NCP) and for the DTS/DTR utility. This document is for users responsible for DECnet for OpenVMS network management.

Revision/Update Information: This is a new manual.

Software Version: OpenVMS AXP Version 1.5
OpenVMS VAX Version 6.0

Digital Equipment Corporation
Maynard, Massachusetts

May 1993

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 be used or copied only 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 Equipment Corporation or its affiliated companies.

© Digital Equipment Corporation 1993.

All Rights Reserved.

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation: AXP, Bookreader, CI, DDCMP, DEC, DECnet, DECnet RSX, DELUA, DEQNA, DEUNA, Digital, DECdns, OpenVMS, VAX, VAX P.S.I., VMS, VMScluster, VMS RMS, and the Digital logo.

The following is a third-party trademark:

Telenet is a registered trademark of GTE Telenet Communication Corporation.

All other trademarks and registered trademarks are the property of their respective holders.

This document was prepared using VAX DOCUMENT, Version 2.1.

Contents

Preface vii

Part I The Network Control Program Utility

1 Description

1.1 Command Syntax 1–1
1.2 Entering Commands 1–2
1.3 Command Prompting 1–4
1.3.1 Exiting the Prompting Sequence 1–5
1.3.2 Prompting Examples 1–6
1.4 Component and Parameter Syntax Rules 1–7
1.5 NCP HELP Facility 1–12

2 Using NCP

2.1 Hints for Using Commands 2–3

3 NCP Commands

CLEAR/PURGE CIRCUIT 3–2
CLEAR/PURGE EXECUTOR 3–6
CLEAR EXECUTOR NODE 3–11
CLEAR/PURGE LINE 3–12
CLEAR/PURGE LOGGING 3–16
CLEAR/PURGE LOGGING EVENTS 3–17
CLEAR/PURGE LOGGING NAME 3–19
CLEAR/PURGE MODULE CONFIGURATOR 3–20
CLEAR/PURGE MODULE X25-ACCESS (Remote management only) 3–21
CLEAR/PURGE MODULE X25-PROTOCOL (Remote management only) 3–22
CLEAR/PURGE MODULE X25-SERVER/X29-SERVER (Remote management only) 3–25
CLEAR/PURGE NODE 3–27
CLEAR/PURGE NODE CIRCUIT 3–31
CLEAR/PURGE OBJECT 3–32
CONNECT NODE 3–34
CONNECT VIA 3–35
COPY KNOWN NODES 3–36
DISCONNECT LINK 3–40

HELP 3–42
LOAD NODE 3–44
LOAD VIA 3–47
LOOP CIRCUIT 3–49
LOOP EXECUTOR 3–51
LOOP LINE (Remote management only) 3–52
LOOP NODE 3–54
SET CIRCUIT ALL 3–56
SET/DEFINE CIRCUIT 3–57
SET EXECUTOR ALL 3–63
SET/DEFINE EXECUTOR 3–64
SET EXECUTOR NODE 3–73
SET KNOWN PROXIES ALL 3–75
SET LINE ALL 3–76
SET/DEFINE LINE 3–77
SET LOGGING ALL 3–85
SET/DEFINE LOGGING EVENTS 3–86
SET/DEFINE LOGGING STATE 3–88
SET/DEFINE MODULE CONFIGURATOR 3–90
SET/DEFINE MODULE X25-ACCESS (Remote management only) . . . 3–92
SET/DEFINE MODULE X25-PROTOCOL (Remote management only) 3–94
SET/DEFINE MODULE X25-SERVER/X29-SERVER (Remote management only) 3–100
SET NODE ALL 3–105
SET/DEFINE NODE 3–106
SET/DEFINE NODE CIRCUIT 3–111
SET OBJECT ALL 3–112
SET/DEFINE OBJECT 3–113
SHOW AREA 3–118
SHOW/LIST CIRCUIT 3–120
SHOW/LIST EXECUTOR 3–125
SHOW/LIST LINE 3–128
SHOW LINKS 3–131
SHOW/LIST LOGGING 3–133
SHOW/LIST MODULE CONFIGURATOR 3–136
SHOW/LIST MODULE X25-ACCESS (Remote management only) . . . 3–139
SHOW/LIST MODULE X25-PROTOCOL (Remote management only) 3–141
SHOW/LIST MODULE X25-SERVER/X29-SERVER (Remote management only) 3–145
SHOW/LIST NODE 3–147
SHOW/LIST OBJECT 3–152
TELL 3–154
TRIGGER NODE 3–156
TRIGGER VIA 3–158
ZERO CIRCUITS 3–159

ZERO EXECUTOR 3–160
ZERO LINE 3–161
ZERO MODULE X25-PROTOCOL (Remote management only) 3–162
ZERO MODULE X25-SERVER (Remote management only)/X29-SERVER 3–163
ZERO NODE 3–164

Part II DECnet for OpenVMS DTS/DTR Utility

4 DECnet Test Sender/DECnet Test Receiver (DTS/DTR) Commands

4.1 DTS Command Syntax 4–1
4.2 DTS/DTR Usage Summary 4–2
4.3 DTS Commands 4–3
CONNECT 4–4
DATA 4–6
DISCONNECT 4–8
HELP 4–10
INTERRUPT 4–11

A DECnet Circuit and Line Devices

B Network Counter Summary

B.1 Overview B–1
B.2 Circuit Counters B–4
B.3 Line Counters B–7
B.4 Node Counters B–11
B.5 X.25 Protocol Module DTE Counters B–13
B.6 X.25 and X.29 Server Module Counters B–14

C States and Transitions

C.1 Overview C–1
C.2 Circuit and Line States and Transitions C–1
C.3 Link States C–5
C.4 DTE States and Transitions C–5
C.5 X.25 Server Module States and Transitions C–8

D Event Class and Type Summary

D.1 Network Management Layer Events D–2
D.2 Session Control Layer Events D–4
D.3 End Communications Layer Events D–4
D.4 Routing Layer Events D–4
D.5 Data Link Layer Events D–9
D.6 X.25 Packet Level Events D–11
D.7 System-Specific Events D–12
D.8 Digital Distributed Name Service (DECdns) Clerk Events D–12

Index

Tables

1–1	Command Syntax Example	1–2
1–2	NCP Syntax Summary	1–7
2–1	Privileges for NCP Operations	2–2
3–1	Object Type Codes	3–116
A–1	Ethernet Circuit and Line Devices	A–1
A–2	DDCMP and CI Circuit and Line Devices	A–1
A–3	X.25 Circuit and Line Devices	A–2
A–4	FDDI Circuit and Line Devices	A–2
C–1	Circuit and Line States and Substates	C–1
C–2	Circuit and Line State Transitions	C–2
C–3	Link States	C–5
C–4	DTE States and Substates	C–6
C–5	DTE State Transitions	C–6
C–6	X.25 Server States	C–8
C–7	X.25 Server State Transitions	C–8

Preface

Intended Audience

The *DECnet for OpenVMS Network Management Utilities* is a reference to the formats and syntax of NCP and DTS/DTR commands. This manual is for users already familiar with DECnet for OpenVMS concepts. System managers can use NCP to configure, control, monitor, or test DECnet for OpenVMS software.

Document Structure

This document consists of four chapters and one appendix:

- Chapter 1 provides a full description of the NCP utility, inclu the following topics:
  - Command syntax
  - Entering commands
  - Command prompting
  - Component and parameter syntax rules
  - NCP HELP
- Chapter 2 outlines the following NCP information:
  - Invoking the utility
  - Exiting the utility
  - Directing output
  - Restrictions or privileges for specific operations
- Chapter 3 describes NCP commands, including format, paramete qualifiers, and examples.
- Chapter 4 provides a description of the DTS/DTR utility
- Appendix A lists DECnet for OpenVMS circuit and line devices.
- Appendix B includes descriptions of circuit, line and node counters.
- Appendix C contains descriptions of circuit, line, and link states, and their substates and transitions.This section also describes X25-SERVER module and DTE states, substates, and their transitions.
- Appendix D includes a summary of NCP events in terms of their class and type.

Associated Documents

Before using NCP, you should be familiar with the basic networking concepts described in the *DECnet for OpenVMS Guide to Networking*. DECnet for OpenVMS concepts and NCP usage information are presented in the *DECnet for OpenVMS Networking Manual*, which also describes the procedures needed to configure, manage, test, and monitor your network. Refer to that manual for more information about any NCP command listed in this manual.

The following functional specifications define Digital Network Architecture (DNA) Phase IV protocols to which all implementations of DECnet Phase IV adhere:

- *DECnet Digital Network Architecture General Description*
- *Digital Data Communications Message Protocol Functional Specification*
- *Network Services Protocol Functional Specification*
- *Maintenance Operation Protocol Functional Specification*
- *Data Access Protocol Functional Specification*
- *Routing Layer Functional Specification*
- *DNA Session Control Functional Specification*
- *DNA Phase IV Network Management Functional Specification*
- *Ethernet Node Product Architecture Specification*
- *Ethernet Data Link Functional Specification*

Conventions

In this manual, every use of OpenVMS AXP means the OpenVMS AXP operating system, every use of OpenVMS VAX means the OpenVMS VAX operating system, and every use of OpenVMS means both the OpenVMS AXP operating system and the OpenVMS VAX operating system.

The following conventions are used to identify information specific to OpenVMS AXP or to OpenVMS VAX:

AXP	The AXP icon denotes the beginning of information specific to OpenVMS AXP.
VAX	The VAX icon denotes the beginning of information specific to OpenVMS VAX.
♦	The diamond symbol denotes the end of a section of information specific to OpenVMS AXP or to OpenVMS VAX.

The following conventions are also used in this manual:

Ctrl/*x*	A sequence such as Ctrl/*x* indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.

. . .	A horizontal ellipsis in examples indicates one of the following possibilities: • Additional optional arguments in a statement have been omitted. • The preceding item or items can be repeated one or more times. • Additional parameters, values, or other information can be entered.
. . .	A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.
[ ]	In format descriptions, brackets indicate optional elements. You can choose one, none, or all of the options. (Brackets are not optional, however, in the syntax of a directory name in an OpenVMS file specification, or in the syntax of a substring specification in an assignment statement.)
{ }	In format descriptions, braces surround a required choice of options; you must choose one of the options listed.
boldface text	Boldface text is used to show user input in Bookreader versions of the manual.
italic text	Italic text emphasizes important information, indicates variables, and indicates complete titles of manuals. Italic text also represents information that can vary in system messages (for example, Internal error *number*), command lines (for example, /PRODUCER=*name*), and command parameters in text.
UPPERCASE TEXT	Uppercase text indicates a command, the name of a routine, the name of a file, or the abbreviation for a system privilege.
-	A hyphen in code examples indicates that additional arguments to the request are provided on the line that follows.
numbers	All numbers in text are assumed to be decimal, unless otherwise noted. Nondecimal radixes—binary, octal, or hexadecimal—are explicitly indicated.

Notes

In this document, discussions that refer to VMScluster environments apply to both VAXcluster systems that include only VAX nodes and VMScluster systems that include at least one AXP node unless indicated otherwise.

In this document, discussions that refer to DECdns apply to the Distributed Name Service (DNS) V1.1.

Part I

The Network Control Program Utility

1
Description

The Network Control Program (NCP) is the utility program used by system and network managers to configure and control DECnet nodes. System managers can also use NCP to monitor network resources and test network components. This manual presents the syntax of the NCP commands used to perform these tasks.

Consult the *DECnet for OpenVMS Networking Manual* for detailed descriptions explaining the use of the NCP commands. After you read the appropriate sections of the *DECnet for OpenVMS Networking Manual* and know which NCP command (or series of commands) you need, use this manual for the syntax of the commands.

The description of NCP covers the following topics:

- Command syntax
- Entering commands
- Command prompting
- Component and parameter syntax rules
- NCP HELP Facility

The *DECnet for OpenVMS Networking Manual* also contains an overview of networking concepts and descriptions of the components that make up the network.

1.1 Command Syntax

To invoke NCP, type RUN SYS$SYSTEM:NCP at the DCL prompt. The *NCP>* prompt indicates that you are using the NCP utility program. When you receive this prompt, you can enter NCP commands.

Description
1.1 Command Syntax

The NCP command syntax has four parts: a command verb, a component, one or more parameters and, optionally, one or more qualifiers. Table 1–1 is an example of this syntax.

Table 1–1 Command Syntax Example

Command Verb	Component	Parameter	Qualifier
SHOW	ACTIVE LINES	CHARACTERISTICS	TO filespec
	KNOWN LINES	COUNTERS	
	LINE line-id	STATUS	
		SUMMARY	

For each command, you must supply a command verb and a component. Some commands also require one or more parameters. Command qualifiers are optional. Unless otherwise indicated, you can specify parameters in any order.

1.2 Entering Commands

Enter NCP commands as verbs, components, parameters, and qualifiers separated by spaces or tabs. To continue a long command to the next line, use the standard continuation line convention (a hyphen as the last character in the line). The *NCP>* prompt is not displayed on continuation lines; instead, continuation lines prompt with an underscore character. For example:

```
NCP>SET EXECUTOR -
_ ADDRESS 11 -
_ INCOMING TIMER 30 -
_ STATE ON
NCP>
```

In command files, use an exclamation point (!) to designate a comment line. NCP ignores hyphens within and at the end of a comment line. However, NCP does not ignore lines beginning with an exclamation point if they follow a command line ending with a hyphen.

When entering an NCP command, you can truncate any command verb, component name, parameter name, or qualifier name to its fewest unique letters. For example, the following two commands perform the same function:

```
NCP>SET EXECUTOR ADDRESS 11 STATE ON
```

```
NCP>SET EXE AD 11 STA ON
```

For clarity, all examples show the full command syntax.

Recalling Commands

You can recall NCP commands that you previously entered by pressing Ctrl/B or the arrow keys in response to the *NCP>* prompt.

You can press Ctrl/B or the up-arrow key up to 20 times to recall the last 20 commands you have typed. After you recall an NCP command, you can edit it and then reenter it. You can press the down-arrow key to recall the next (most recent) command in the NCP command recall buffer.

Using Wildcard Characters to Enter Commands

You can use the asterisk (*) and the percent sign (%) as wildcard characters in an NCP command line to represent NCP components by a general name, rather than specify each component name individually.

You can use the wildcard characters to represent the following component names:

- Node name
- Node address
- Circuit name
- Line name
- Object name
- Events

The asterisk wildcard represents one or more characters, while the percent sign represents a single character.

Rules for Using Wildcard Characters

The following rules define how you can use wildcard characters with component names:

- If the component name is a string, the wildcard character may occur at any location in the string. For example:

  ```
  NCP>LIST NODE ST%R STATUS
  NCP>SHOW OBJECT M* CHARACTERISTICS
  ```

  The first command requests a list of status information for all nodes with four-letter node names beginning with ''ST'' and ending with ''R.'' The second command requests a listing of characteristics for all objects with names beginning with ''M.''

- For node addresses, which are represented by the format *area-number.node-number*, only the *node-number* portion of the node address (the numeral on the right side of the period) can contain a wildcard. For example, the following command sets a COUNTER TIMER value of 45 seconds for all nodes in area 4:

  ```
  NCP>SET NODE 4.* COUNTER TIMER 45
  ```

  Specifying a node address such as *.5 is invalid because the *area-number* cannot contain a wildcard.

- In a node address, a wildcard character cannot be combined with a numeral to represent a partial *node-number*. The node addresses 4.* and 4.% contain valid uses of the wildcard characters, but the node addresses 4.%2 and 4.1* are invalid.

- For events, which are represented by the format *class.type*, only the *type* portion of the event (the numeral on the right side of the period) can contain a wildcard. For example, the following command specifies that all class 2 events are to be logged:

  ```
  NCP>SET KNOWN LOGGING EVENTS 2.*
  ```

- Except in the case of events, only component names can contain wildcards. Parameter values cannot contain wildcards. The following command is invalid because the circuit name UNA-* is not the component name in the command. Rather, it is a parameter used to modify the component named BOSTON. Only component names can be represented by wildcard characters.

  ```
  NCP>SET NODE BOSTON SERVICE CIRCUIT UNA-*    !INVALID COMMAND
  ```

  The component name EVENT is used as a parameter to the LOGGING commands and can contain wildcard characters, as long as only the *type* portion of the event number (the numeral to the right of the period) contains the wildcard.

  For example, the following command clears logging to the logging file for all class 2 events:

  ```
  NCP>CLEAR LOGGING FILE EVENTS 2.*
  ```

- Unit numbers of circuit and line devices can contain wildcard characters, but device names of circuits and lines cannot contain wildcard characters. Circuit and line devices are typically identified by the format *dev-c*, where *dev* is a mnemonic device name, and *c* is a device unit number. In the following example, the asterisk replaces the unit number in this request for circuit information for all DMC devices:

  ```
  NCP>SHOW CIRCUIT DMC-*
  ```

  However, the *device-name* portion of a circuit or line name cannot contain wildcard characters. Therefore, the following commands are invalid:

  ```
  NCP>SHOW CIRCUIT D* STATUS  !INVALID COMMAND
  NCP>SHOW LINE %NA-0 SUMMARY !INVALID COMMAND
  ```

Note that substituting a wildcard character for an entire component name is equivalent to specifying the command component KNOWN. For example:

```
NCP>SHOW NODE * STATUS
```

This command is equivalent to the following command:

```
NCP>SHOW KNOWN NODES STATUS
```

1.3 Command Prompting

NCP prompts for selected components and parameters if you do not supply them when you enter the command. These components and parameters are those that you use most often. Each prompt is either a list of valid choices for that position in the command string or the name of a parameter and a description of the value required.

Note

Prompting does not occur when NCP receives input from a command file. Parameter names must be supplied in command files.

Each prompt consists of two parts: the component or parameter desired, and its format. The syntax is as follows:

component (format):

The component part indicates the name of either the component or parameter. The format part indicates the range of acceptable values. The format part presents command words in uppercase, value classes in lowercase as descriptive strings, and numeric ranges as two numbers separated by a hyphen. Commas indicate alternatives that are mutually exclusive.

Prompting does not guarantee that you are entering a valid command. Even if parameters are required to make the command meaningful, prompting does not demand a response for all the required parameter values or names, as the following example indicates:

```
NCP>SET OBJECT NML     Return
Object number                (0-255):   Return
File name             (filename.type):   Return

Privileges  (List of VMS privileges):   Return
User ID             (1-16 characters):   Return
Password             (1-8 characters):   Return
Account             (1-16 characters):   Return
Proxy access INCOMING, OUTGOING, BOTH, NONE:)   Return
%NCP-F-NOPARM, no parameters
NCP>
```

1.3.1 Exiting the Prompting Sequence

You can exit a prompting sequence by pressing Ctrl/Z. When you do so, the NCP> prompt returns, and you can then enter another NCP command. To exit NCP completely, press Ctrl/Z at the NCP> prompt.

The following example shows how to interrupt a prompting sequence and exit NCP.

```
NCP>SET    Return
(CIRCUIT, EXECUTOR, KNOWN, LINE,
   LOGGING, MODULE, NODE, OBJECT): LINE
Line ID                         (dev-c-u.t): FZA-1
Number of receive buffers         (1-1024):  Ctrl/Z
.
.
.
NCP>  Ctrl/Z
$
```

If, after responding to one or more prompts, you want to exit the prompting sequence but retain whatever information you have provided in the database, you can respond to the next prompt with _DONE, as follows:

```
NCP>DEFINE OBJECT       Return
Object Name               (12 characters): TALOS
Object Number               (0-255): 101
File name                 (filename): TALOS.EXE
Privileges (List of VMS privileges): _DONE
NCP>
```

In this example, NCP makes a new entry in the permanent database for object TALOS. Because _DONE is used in response to a prompt, NCP writes no other parameter information to the database for this object, and NCP exits the prompting sequence. Note that NCP returns a success message whenever it modifies the permanent database successfully.

Using _DONE may not be successful, however, if you do not provide all of the necessary parameters for a given component. For example:

```
NCP>DEFINE OBJECT     Return
Object Name               (12 characters): RIGEL
Object Number               (0-255):   Return
File name                (filename): RIGEL.EXE
Privileges (List of VMS privileges): _DONE
%NCP-I-NMLRSP, listener response - Parameter missing, number
Object = RIGEL
NCP>
```

This example is similar to the previous example except that the object number is omitted. Because each object must have an object number defined in the database, and because none currently exists for object RIGEL, NCP does not update the database with this information.

1.3.2 Prompting Examples

The following examples illustrate three types of prompting that NCP provides:

- Prompts to supply missing component

  ```
  NCP>CLEAR     Return
  (CIRCUIT, EXECUTOR, KNOWN, LINE, LOGGING, MODULE, NODE, OBJECT):
  ```

  No component is specified for the CLEAR command. The prompt format is a list of valid components separated by commas. Note that the plural component KNOWN is included in this list.

- Prompts to supply parameter values

  ```
  NCP>CLEAR LINE     Return
  LINE ID (dev-c-u):
  ```

  NCP prompts for the *line-id* for this component because it is not provided with the command.

  The following example shows variations of the prompting scheme for parameter values:

  ```
  NCP>SET EXECUTOR     Return
  Node address              (1.1-63.1023): 2.11
  State       (ON, OFF, SHUT, RESTRICTED): ON
  System id string (32 characters:)  "DECnet for OpenVMS"
  .
  .
  ```

- Prompts to select parameter names

  ```
  NCP>CLEAR LINE     Return
  Line ID       (dev-c-u.t): FZA-1
  All line parameters (Y,N):
  ```

  NCP prompts for the selected parameter names whose values are to be removed from the database. In this case, no parameter name is given; consequently, NCP asks if all values should be removed. The Y parameter indicates that all values should be removed. Using N or RETURN causes NCP to provide a different response.

```
NCP> CLEAR NODE BOSTON     Return
All node parameters (Y,N):   Return
Host node           (Y,N): Y
.
.
.
Service password    (Y,N): Y
Tertiary loader     (Y,N): N
```

Because RETURN is the response to the first prompt, NCP prompts for all of the parameter names.

1.4 Component and Parameter Syntax Rules

When you enter NCP commands, many components, parameters, and qualifiers require you to supply additional information. For the most part, their syntax follows a standard set of rules. Exceptions to these rules are documented in the description of the component, parameter, or qualifier to which they apply.

The syntax of the various component-name, parameter, and qualifier values is summarized in Table 1–2. In the following table, all numeric values are in decimal and have a range of 0 to 65,535 unless otherwise specified:

Table 1–2 NCP Syntax Summary

account	A string of up to 39 alphanumeric and hyphen characters. OpenVMS operating systems do not currently use this string for access control.
area-number	A decimal value in the range 1 to 63 to be specified in the beginning of the *node-address* and separated from the *node number* by a period. If you do not specify an area number, the area number of the executor is used. The default area number for the executor is 1.

(continued on next page)

Table 1–2 (Cont.) NCP Syntax Summary

circuit-id	A string of characters whose exact syntax is that for a DECnet for OpenVMS circuit. The string may also be that for an X.25 circuit identification. For X.25 circuits, *circuit-id* is a string of up to 15 characters beginning with ''X25-''. Circuit identification takes one of the following formats: dev-c dev-c-u dev-c.t dev-c-u.t dev — Is a device name. Refer to the description of *device-type* for a list of device mnemonic names. For a complete list of DECnet for OpenVMS circuit devices and descriptions, see Appendix A. c — Is a decimal number (0 or a positive integer) designating the device's hardware controller. u — Is a decimal unit or circuit number (0 or a positive integer) included only if more than one unit is associated with the controller. t — Is a decimal number (0 or a positive integer) that identifies a tributary on a multipoint circuit. This is a logical tributary number, not to be confused with the tributary address used to poll the tributary.
cpu-type	A string of characters consisting of one of the following: DECSYSTEM1020 PDP11 PDP8 VAX
dest-name	An identification string consisting of 1 to 16 alphanumeric characters.
device-type	A string of characters representing the mnemonic name for the device. Appendix A in this book contains a list of devices, current as of this publication date. The following list represents typical devices: BNA FZA MFA MNA QNA SVA UNA
dte-address	A string of up to 15 decimal digits. See the *Public Network Information* manual for the exact format of this parameter on your network.

(continued on next page)

Table 1–2 (Cont.) NCP Syntax Summary

event-list

A list of event types for a given class in the format class.type. When specifying an event list, you may specify only one class; however, you can specify a range of types by using commas and hyphens, for example, 4.3–5,7–10. You can use the asterisk wildcard character in an event list, but only to replace an event type. The following table provides examples of these formats.

Event List	Meaning
4.4	Identifies event class 4, type 4.
4.5–7	Identifies event class 4, types 5 through 7.
4.5,7–9,11	Identifies event class 4, types 5, 7 through 9, and 11. Note that types must be specified in ascending order.
4.*	Identifies all event types for class 4 events.

filespec

A file specification string in the following general format:

node-spec::device:[directory]filename.type;version

Logical names are permitted.

Refer to the *VMS DCL Concepts Manual* for the format of a file specification, and the *Guide to OpenVMS File Applications* for more information about specifying files.

h-address

A string of 6 pairs of hexadecimal digits separated by hyphens (for example 08–00–2B–2B–AF–F3). The string represents the hardware address of a LAN device controller.

hex-byte

A string of exactly 2 hexadecimal digits.

hex-password

A string of up to 8 hexadecimal digits.

hex-value

A string of up to 32 hexadecimal digits.

id-string

A string of up to 32 characters. If the string includes spaces or tabs, enclose it within quotation marks.

(continued on next page)

Table 1–2 (Cont.) NCP Syntax Summary

line-id	A string of characters whose exact syntax is that for a DECnet for OpenVMS line identification. The string may also be that for an X.25 line identification. For OpenVMS operating systems, *line-id* takes one of the following formats: dev-c dev-c-u dev — Is a device name. Refer to the description of *device-type* for a list of device mnemonic names. For a complete list of DECnet for OpenVMS line devices and descriptions, see Appendix A. c — Is a decimal number (0 or a positive integer) designating the device's hardware controller. u — Is a decimal unit or line number (0 or a positive integer) included only if more than one unit is associated with the controller.
microseconds	A decimal numeric value.
milliseconds	A decimal numeric value.
network-name	For X.25 circuits, specifies a 1- to 16-character alphanumeric identifier that indicates the network to be used.
node-address	A numeric value in the range 1.1 to 63.1023, composed of an area number to the left of the period followed by a node number to the right of the period. (The node number indicates the address of the node within the specified area.) If the area number is not supplied, the area number of the executor node is used. The default area number for the executor is 1.
node-id	Either a *node-name* or a *node-address*.
node-name	A string of up to six alphanumeric characters containing at least one alphabetic character.
node-spec	A *node-id* followed by optional access control information in the following format: node-id"user password" node-id"user password account"
node-type	A string of characters consisting of one of the following: Nonrouting III Nonrouting IV Routing IV †Routing III †Area
number	A decimal numeric value.
object-name	A string of up to 16 printable characters, consisting of alphanumeric characters, the dollar sign ($), or the underscore (_).

†VAX specific

(continued on next page)

Table 1–2 (Cont.) NCP Syntax Summary

p-address	A string of 6 pairs of hexadecimal digits separated by hyphens (for example AA–00–04–00–03–10). The value is the unique physical address that the node has set for itself or, if the node has not set a physical address, the hardware address associated with the node's device controller.
password	A string of up to 39 printable characters.
privilege-list	A list of OpenVMS privilege names delimited by space characters.
seconds	A decimal numeric value.
sink-name	A string of alphanumeric characters that specifies the name of an event sink. The logging console name is a string in the form TTcnn (if a terminal) or a file specification if it is a file that receives events in the console format. The logging file name is a file specification. The logging monitor name is a 1- to 9-character name. If the string includes spaces or tabs, enclose it in quotation marks.
software-id	A string of up to 16 characters.
software-type	A string of characters consisting of one of the following: MANAGEMENT FILE SECONDARY LOADER SYSTEM TERTIARY LOADER
†tributary-address	A numeric value in the range of 0 to 255.
user-id	A string of up to 39 alphanumeric and hyphen characters. The operating system may include additional restrictions; for example, OpenVMS systems limit the string to 12 characters.

†VAX specific

Quotation mark delimiters are valid for the *node-spec* format. In addition, you can use quotation marks as delimiters when providing receive and transmit passwords for the SET NODE and DEFINE NODE commands. For example:

```
NCP>SET NODE TRANSMIT PASSWORD "HI OpenVMS"
```

Also, use quotation marks to define a software identification string with text and spaces for the IDENTIFICATION parameter of the SET EXECUTOR command. For example:

```
NCP>SET EXECUTOR IDENTIFICATION "VMS HOST SYSTEM"
```

1.5 NCP HELP Facility

NCP provides an extensive HELP Facility that contains information about each NCP command and its parameters and qualifiers, including general examples of its use. The information in the HELP file is tree structured, which makes it easy to retrieve information quickly and efficiently. The HELP description in the Command Section of this manual provides examples to illustrate the use of the HELP command for a typical terminal session.

Use the HELP command to display information at various levels of detail. For example, each HELP display includes a list of command words under an ''Additional Information Available'' section. These command words serve as topics for the next level of information accessible by the HELP command. In the general format of the HELP command shown in the following example, the topic is a command word from the list:

```
NCP>HELP
Information available:

   CLEAR      Commands   CONNECT    COPY       DEFINE     DISCONNECT EVENTS
   EXIT       HELP       LIST       LOAD       LOOP       PARAMETERS Prompting
   PURGE      SET        SHOW       TELL       TRIGGER    ZERO

Topic?
```

You can abbreviate command words when you enter the command. When you enter a particular level of detail, that level becomes the current level for accessing information. Through the use of the prompts *Topic?* and *Subtopic?*, you can return to a higher or lower level of information, respectively. The highest level of information is the general description of the HELP Facility.

ertain

2
Using NCP

The Network Control Program (NCP) is a DECnet for OpenVMS utility that accepts commands to configure, control, monitor, and test a DECnet network.

Entering Commands

To invoke NCP, enter the following DCL command:

```
$ RUN SYS$SYSTEM:NCP
```

NCP returns the following prompt:

```
NCP>
```

Alternatively, you can enter a single NCP command from DCL by first entering a DCL string assignment statement. For example:

```
$ NCP=="$NCP"
$ NCP SHOW EXECUTOR STATUS

Node Volatile Status as of 12-JUL-1992 11:57:59

Executor node = 4.306 (INTER)

State                     = on
Physical address          = AA-00-04-00-32-11
Active links              = 2
Delay                     = 0

$
```

NCP executes the SHOW EXECUTOR STATUS command and returns control to DCL.

To exit from an NCP session, type EXIT or press CTRL/Z after the *NCP>* prompt.

Output for the SHOW and LIST commands is normally displayed on the default output device, SYS$OUTPUT. Alternatively, you may direct output to a specified file using the TO qualifier with the SHOW or LIST command.

Certain NCP commands require privileges, as shown in Table 2–1. For DECnet purposes, an account is considered privileged if any privileges other than NETMBX and TMPMBX are held.

Table 2–1 Privileges for NCP Operations

Operation	Required Privileges
Start the network	ACNT, CMKRNL, SYSNAM, and DETACH
Perform task-to-task communication	NETMBX is required to assign a channel to the NET device. TMPMBX is required to optionally associate a temporary mailbox with a network channel.
Create a logical link	NETMBX
Declare a name or object number in a user task	SYSNAM (Refer to *DECnet for OpenVMS Networking Manual* for information about user tasks.)
CLEAR parameters from the volatile database	NETMBX and OPER
Issue CONNECT commands	NETMBX
COPY KNOWN NODES	NETMBX, TMPMBX and OPER (SYSPRV is also needed if access to the permanent node database is required or if the WITH option is used.)
DEFINE parameters in the permanent database	SYSPRV and OPER[1]
DISCONNECT LINKs	NETMBX and OPER
LIST parameters kept in the permanent database	SYSPRV[1]
LIST service passwords kept in the permanent database	SYSPRV and OPER[1]
LIST receive passwords, transmit passwords, and object and executor access control passwords kept in the permanent database	BYPASS
Issue LOAD commands	NETMBX, TMPMBX and OPER
Use LOOP CIRCUIT	NETMBX, TMPMBX and OPER
Use LOOP EXECUTOR	NETMBX and TMPMBX
PURGE parameters from the permanent database	SYSPRV and OPER[1]
SET parameters kept in the volatile database	NETMBX and OPER
Issue SET EXECUTOR NODE	Requires NETMBX and TMPMBX privileges on the local node, and NML on the executor node must have appropriate privileges to perform the commands issued after the SET EXECUTOR NODE command.
Use "SET component ALL" to load parameters from the permanent database to the volatile database	NETMBX, OPER and SYSPRV[1]
SHOW parameters kept in the volatile database	NETMBX

[1]You can perform operations requiring access to the permanent database without the SYSPRV privilege if you have read and write access to all configuration database files or hold BYPASS privilege. However, Digital recommends that you protect these network configuration files from unauthorized access by requiring SYSPRV to access these files.

(continued on next page)

Table 2–1 (Cont.) Privileges for NCP Operations

Operation	Required Privileges
SHOW service passwords kept in the volatile database	NETMBX and OPER
SHOW receive passwords, transmit passwords, and access control passwords kept in the volatile database	NETMBX and BYPASS
Issue TELL commands	Requires NETMBX and TMPMBX on the local node, and NML on the executor node must have sufficient privileges to perform the command that follows TELL.
Issue TRIGGER commands	NETMBX, TMPMBX and OPER
ZERO counters	NETMBX and OPER

2.1 Hints for Using Commands

You can abbreviate any command verb, component, parameter, or qualifier as long as the abbreviation is not ambiguous.

Commands that have the same components and parameters, yet different command verbs—depending upon whether they access the volatile or the permanent database—are listed together in Chapter 3. Examples of commands listed together are CLEAR and PURGE, SET and DEFINE, and SHOW and LIST, where CLEAR, SET, and SHOW verbs apply to the volatile database and PURGE, DEFINE and LIST verbs apply to the permanent database. When two commands are grouped together, components and parameters are described for the command that accesses the volatile database. Typically, the actions described for the volatile database also apply to the permanent database. However, if certain commands apply only to a particular database or if a different action is required for the volatile and permanent databases, these exceptions are specifically noted in the command description.

In general, the NCP commands and parameters are common to most network configurations. Exceptions to this general rule are noted under the command or parameter to which they apply. The VAX margin icon indicates functions supported only on VAX systems. Other restrictions are noted as part of the command or parameter description.

Note

Although NCP may accept a particular command syntax, this is no guarantee that some other DECnet implementation supports the function.

For a complete description of the network components specified in NCP commands, see the *DECnet for OpenVMS Networking Manual*. That manual also explains how to use the NCP commands to manage the DECnet network.

3
NCP Commands

This chapter presents NCP command descriptions. Commands are presented in alphabetical order with the associated command parameters and appropriate examples.

NCP allows for management of the local node and for remote management of any DECnet node that you specify as the executor. When you issue NCP commands, there are two ways to specify the executor — the location at which the command executes:

- Begin individual NCP commands with the TELL prefix.
- Set the executor with the SET EXECUTOR NODE command. All subsequent NCP commands execute at the remote node until you:
  - Issue another SET EXECUTOR NODE command,
  - Issue a CLEAR EXECUTOR NODE command, or
  - Exit NCP.

To enter one or two NCP commands for remote execution, it is probably easier to use the TELL prefix. To execute multiple NCP commands at a remote node, first issue the SET EXECUTOR NODE command, specifying that node.

Note

NCP commands and command parameters are not all supported on every hardware platform. In some cases, NCP allows you to set and display an unsupported parameter. However, the unsupported parameter is not functional and has no effect on the operation of the DECnet software.

CLEAR/PURGE CIRCUIT

The CLEAR CIRCUIT command resets circuit parameters in the volatile database to the default value (if any), or removes them from the volatile database. The PURGE CIRCUIT command removes circuit parameters from the permanent database.

Format

CLEAR circuit-component parameter [...]

PURGE circuit-component parameter [...]

Circuit Components

CIRCUIT	KNOWN CIRCUITS

Circuit Components

CIRCUIT circuit-id
Resets to the default value (if any) or removes from the database the parameters for a specific circuit. The *circuit-id* specifies the name of the circuit you want to clear.

KNOWN CIRCUITS
Resets to the default value (if any) or removes from the database the parameters for all known circuits.

Command Parameters

ACTIVE BASE (VAX only)
VAX
Applies only to DDCMP CONTROL circuits. Resets to its default value the ACTIVE BASE value in volatile the database. Removes the ACTIVE BASE parameter from the permanent database.

ACTIVE INCREMENT (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the ACTIVE INCREMENT value in the volatile database. Removes the ACTIVE INCREMENT parameter from the permanent database.♦

ALL
Removes from the database the specified circuit or all known circuits. The circuit must be in the OFF state before you specify this parameter.

BABBLE TIMER (VAX only)
VAX
Applies only to DDCMP CONTROL circuits. Resets to its default value the BABBLE TIMER value in the volatile database. Removes the BABBLE TIMER parameter from the permanent database.♦

COST
Resets to its default value the COST value in the volatile database. Removes the COST parameter from the permanent database. Does not apply to X.25 PVCs.

COUNTER TIMER
Cancels the event logging timer. This cancellation prevents any further circuit counter event logging for the indicated circuit or circuits.

VAX

DEAD THRESHOLD (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the DEAD THRESHOLD number in the volatile database. Removes the DEAD THRESHOLD parameter from the permanent database.♦

DTE (Remote management only)
Applies only to DLM circuits. Allows the X.25 call to be set up using any local DTE.

VAX

DYING BASE (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the DYING BASE value in the volatile database. Removes the DYING BASE parameter from the permanent database.

DYING INCREMENT (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the DYING INCREMENT value in the volatile database. Removes the DYING INCREMENT parameter from the permanent database.

DYING THRESHOLD (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the DYING THRESHOLD number in the volatile database. Removes the DYING THRESHOLD parameter from the permanent database.♦

HELLO TIMER
Resets to its default value the HELLO TIMER value in the volatile database. Removes the HELLO TIMER parameter from the permanent database.

VAX

INACTIVE BASE (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the INACTIVE BASE value in the volatile database. Removes the INACTIVE BASE parameter from the permanent database.

INACTIVE INCREMENT (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the INACTIVE INCREMENT value in the volatile database. Removes the INACTIVE INCREMENT parameter from the permanent database.

INACTIVE THRESHOLD (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the INACTIVE THRESHOLD value in the volatile database. Removes the INACTIVE THRESHOLD parameter from the permanent database.

MAXIMUM BUFFERS (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the MAXIMUM BUFFERS value in the volatile database. Removes the MAXIMUM BUFFERS parameter from the permanent database.♦

MAXIMUM RECALLS (Remote management only)
Applies only to X.25 DLM circuits. Resets to its default value the limit on the number of call retries in the volatile database. Removes the MAXIMUM RECALLS parameter from the permanent database.

CLEAR/PURGE CIRCUIT

MAXIMUM ROUTERS
Applies only to Ethernet and FDDI circuits. Resets to its default value the MAXIMUM ROUTER value in the volatile database. Removes the MAXIMUM ROUTERS parameter from the permanent database.

VAX

MAXIMUM TRANSMITS (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the MAXIMUM TRANSMIT number in the volatile database. Removes the MAXIMUM TRANSMITS parameter from the permanent database.♦

NETWORK (Remote management only)
Applies only to purging X.25 PVCs and X.25 DLM circuits. Removes the circuit's network parameters from the permanent database.

NUMBER (Remote management only)
Applies only to incoming DLM circuits. Removes the restriction on accepting calls only from a particular remote DTE number.

RECALL TIMER (Remote management only)
Applies only to X.25 DLM circuits. Resets to its default value the RECALL TIMER value in the volatile database. Removes the RECALL TIMER parameter from the permanent database.

ROUTER PRIORITY
Applies only to Ethernet and FDDI circuits. Resets to its default value the priority assigned to the router in the volatile database. Removes the ROUTER PRIORITY parameter from the permanent database.

VAX

TRANSMIT TIMER (VAX only)
Applies only to DDCMP CONTROL circuits. Resets to its default value the TRANSMIT TIMER value in the database.♦

Examples

VAX

1. NCP>CLEAR CIRCUIT DMB-0 ALL

   This command removes all parameter entries for circuit DMB–0 in the volatile database. As a result, the circuit no longer exists for the local DECnet software.

2. NCP>CLEAR CIRCUIT DMP-0.4 COUNTER TIMER

   This command removes the COUNTER TIMER parameter for circuit DMP–0.4 from the volatile database.♦

3. NCP>PURGE CIRCUIT SVA-0 ALL

   This command removes all parameter entries in the permanent database for the Ethernet circuit SVA-0.

4. NCP>CLEAR CIRCUIT SVA-* ALL

   This command clears all SVA circuits and the associated parameters for all SVA circuits in the volatile database.

CLEAR/PURGE EXECUTOR

The CLEAR EXECUTOR command resets selected parameters to the default value (if any) or removes them from the volatile database. You cannot clear the BUFFER SIZE, SEGMENT BUFFER SIZE, MAXIMUM BROADCAST ROUTERS, ALIAS NODE, or STATE parameters from the volatile database when the executor is in the ON state.

The PURGE EXECUTOR command removes selected parameters from the permanent database.

Format

CLEAR EXECUTOR parameter [...]

PURGE EXECUTOR parameter [...]

Command Parameters

ALIAS INCOMING
Removes from the database the ALIAS INCOMING parameter.

ALIAS MAXIMUM LINKS
Resets to its default value the ALIAS MAXIMUM LINKS value in the volatile database. Removes the ALIAS MAXIMUM LINKS parameter from the permanent database.

ALIAS NODE
Removes from the permanent database the alias node identifier.

ALL
Removes from the permanent database all executor parameters.

VAX

AREA MAXIMUM COST (VAX only)
Resets to its default value the AREA MAXIMUM COST value for the local node in the volatile database. Removes the AREA MAXIMUM COST parameter for the local node from the permanent database.

AREA MAXIMUM HOPS (VAX only)
Resets to its default value the AREA MAXIMUM HOPS value for the local node in the volatile database. Removes the AREA MAXIMUM HOPS parameter for the local node from the permanent database.♦

BROADCAST ROUTING TIMER
Resets to its default value the BROADCAST ROUTING TIMER value in the volatile database. Removes the BROADCAST ROUTING TIMER parameter from the permanent database.

BUFFER SIZE
Removes from the permanent database the BUFFER SIZE value.

COUNTER TIMER
Removes from the database the logging timer to prevent any further counter logging for the local node.

DEFAULT ACCESS
Resets to its default value the DEFAULT ACCESS value in the volatile database. Removes the DEFAULT ACCESS parameter from the permanent database.

DELAY FACTOR
Resets to its default value the DELAY FACTOR value for the local node in the volatile database. Removes the DELAY FACTOR parameter for the local node from the permanent database.

DELAY WEIGHT
Resets to its default value the DELAY WEIGHT value for the local node in the volatile database. Removes the DELAY WEIGHT parameter for the local node from the permanent database.

VAX

DNS INTERFACE (VAX only)
Removes from the database the DNS INTERFACE parameter.

DNS NAMESPACE (VAX only)
Removes from the database the DNS NAMESPACE parameter.♦

IDENTIFICATION
Removes from the database the identification string.

VAX

IDP (VAX only)
Removes from the database the IDP parameter.♦

INACTIVITY TIMER
Resets to its default value the INACTIVITY TIMER value for the local node in the volatile database. Removes the INACTIVITY TIMER parameter for the local node from the permanent database.

INCOMING PROXY
Resets to its default value the INCOMING PROXY value for the local node in the volatile database. Removes the INCOMING PROXY parameter for the local node from the permanent database.

INCOMING TIMER
Resets to its default value the INCOMING TIMER value for the local node in the volatile database. Removes the INCOMING TIMER parameter for the local node from the permanent database.

MAXIMUM ADDRESS
Resets to its default value the MAXIMUM ADDRESS value for the local node in the volatile database. Removes the MAXIMUM ADDRESS parameter for the local node from the permanent database.

VAX

MAXIMUM AREA (VAX only)
Resets to its default value the MAXIMUM AREA value for the local node in the volatile database. Removes the MAXIMUM AREA parameter for the local node from the permanent database.♦

MAXIMUM BROADCAST NONROUTERS
Resets to its default value the MAXIMUM BROADCAST NONROUTERS value in the volatile database. Removes the MAXIMUM BROADCAST NONROUTERS parameter for the local node from the permanent database.

MAXIMUM BROADCAST ROUTERS
Removes the MAXIMUM BROADCAST ROUTERS parameter from the permanent database.

MAXIMUM BUFFERS
Resets to its default value the MAXIMUM BUFFERS value for the local node in the volatile database. Removes the MAXIMUM BUFFERS parameter for the local node from the permanent database.

MAXIMUM CIRCUITS
Removes the MAXIMUM CIRCUITS parameter from the permanent database.

MAXIMUM COST
Resets to its default value the MAXIMUM COST value for the local node in the volatile database. Removes the MAXIMUM COST parameter for the local node from the permanent database.

MAXIMUM DECLARED OBJECTS
Resets to its default value the MAXIMUM DECLARED OBJECTS parameter in the volatile database. Removes the MAXIMUM DECLARED OBJECTS parameter for the local node from the permanent database.

MAXIMUM HOPS
Resets to its default value the MAXIMUM HOPS value for the local node in the volatile database. Removes the MAXIMUM HOPS parameter for the local node from the permanent database.

MAXIMUM LINKS
Resets to its default value the MAXIMUM LINKS value for the local node in the volatile database. Removes the MAXIMUM LINKS parameter for the local node from the permanent database.

VAX

MAXIMUM PATH SPLITS (VAX only)
Resets to its default value the MAXIMUM PATH SPLITS value for the local node in the volatile database. Removes the MAXIMUM PATH SPLITS parameter for the local node from the permanent database. ♦

MAXIMUM VISITS
Resets to its default value the MAXIMUM VISITS value for the local node in the volatile database. Removes the MAXIMUM VISITS parameter for the local node from the permanent database.

NONPRIVILEGED item
Removes from the database the specified nonprivileged access control information. Specify any or all of the following items for removal:

ACCOUNT
PASSWORD
USER

OUTGOING PROXY
Resets to its default value the OUTGOING PROXY value for the local node in the volatile database. Removes the OUTGOING PROXY parameter for the local node from the permanent database.

OUTGOING TIMER
Resets to its default value the OUTGOING TIMER value for the local node in the volatile database. Removes the OUTGOING TIMER parameter for the local node from the permanent database.

VAX

PATH SPLIT POLICY (VAX only)
Resets to its default value the PATH SPLIT POLICY value for the local node in the volatile database. Removes the PATH SPLIT POLICY parameter for the local node from the permanent database.♦

PIPELINE QUOTA
Resets to its default value the PIPELINE QUOTA value for the local node in the volatile database. Removes the PIPELINE QUOTA parameter for the local node from the permanent database.

PRIVILEGED item
Removes from the database the specified privileged access control information. Specify any or all of the following items for removal:

ACCOUNT
PASSWORD
USER

RETRANSMIT FACTOR
Resets to its default value the RETRANSMIT FACTOR value for the local node in the volatile database. Removes the RETRANSMIT FACTOR parameter for the local node from the permanent database.

VAX

ROUTING TIMER
Resets to its default value the ROUTING TIMER value for the local node in the volatile database. Removes the ROUTING TIMER parameter for the local node from the permanent database.♦

SEGMENT BUFFER SIZE
Removes the SEGMENT BUFFER SIZE parameter for the local node from the permanent database.

STATE
Removes from the permanent database the local node's state.

SUBADDRESSES (Remote management only)
Removes from the database the value of the range of local DTE subaddresses.

TYPE
Removes the TYPE parameter for the local node from the permanent database.

Examples

1. `NCP>PURGE EXECUTOR IDENTIFICATION INCOMING TIMER`

   This command removes the local node's identification string and incoming timer value from the permanent database.

2. `NCP>PURGE EXECUTOR COUNTER TIMER`

   This command removes the local node's counter timer value from the permanent database.

VAX

3. `NCP>CLEAR EXECUTOR AREA MAXIMUM COST`

   This command resets to its default value the AREA MAXIMUM COST parameter in the volatile database for the executor. ♦

CLEAR EXECUTOR NODE

The CLEAR EXECUTOR NODE command clears the default executor designation. The executor of commands becomes the local node. You cannot use the TELL prefix with the CLEAR EXECUTOR NODE command.

After you enter the CLEAR EXECUTOR NODE command, NCP on the local node performs all operations using the privileges of your current process instead of the privilege associated with the default nonprivileged or privileged accounts.

Format

CLEAR EXECUTOR NODE

Parameters

None.

Examples

1.
```
NCP>SET EXECUTOR NODE TRNTO"GRAY MARY"
.
.
.
NCP>CLEAR EXECUTOR NODE
```

   The first command sets the executor node to node TRNTO. The second command clears the default executor node designation specified previously by the SET EXECUTOR NODE command. The executor node is now the local node.

CLEAR/PURGE LINE

The CLEAR LINE command resets line parameters to the default value (if any) or removes the entries and parameters from the volatile database.

The PURGE LINE command removes line parameters from the permanent database.

Format

CLEAR line-component parameter [...]

PURGE line-component parameter [...]

Line Components

LINE	KNOWN LINES

Line Components

LINE line-id
Identifies the specific line whose parameters are to be reset to their default values (if any) or removed from the database.

KNOWN LINES
Resets the parameters to the default value (if any) or removes them for all known lines from the database.

Command Parameters

ALL
Removes the specified line or all known lines from the database. The line must be in the OFF state before you specify this parameter.

CONTROLLER
Resets to its default value the CONTROLLER parameter in the volatile database. Removes the CONTROLLER parameter from the permanent database.

COUNTER TIMER
Removes the event logging timer. This prevents any further line counter event logging for the indicated line(s).

VAX

DEAD TIMER (VAX only)
Applies only to DDCMP lines. Resets the DEAD TIMER parameter to its default value in the volatile database. Removes the DEAD TIMER parameter from the permanent database.

DELAY TIMER (VAX only)
Applies only to DDCMP lines. Removes the DELAY TIMER parameter from the database.♦

ECHO DATA
Applies only to FDDI lines. Resets to its default value the ECHO DATA parameter in the volatile database. Permanent database operations cannot be performed on this parameter.

ECHO LENGTH
Applies only to FDDI lines. Resets to its default value the ECHO LENGTH parameter in the volatile database. Permanent database operations cannot be performed on this parameter.

ECHO TARGET
Applies only to FDDI lines. Resets to its default value the ECHO TARGET parameter in the volatile database. Permanent database operations cannot be performed on this parameter.

VAX

HANGUP (VAX only)
Applies only to asynchronous DDCMP lines. Removes the HANGUP parameter from the database.♦

HOLDBACK TIMER (Remote management only)
Applies only to X.25 lines. Removes the HOLDBACK TIMER parameter from the database, causing the X.25 software to acknowledge received data immediately.

VAX

LINE SPEED (VAX only)
Applies only to asynchronous DDCMP lines. Removes the LINE SPEED parameter from the database.♦

MAXIMUM BLOCK (Remote management only)
Applies only to X.25 lines. Resets to its default value the MAXIMUM BLOCKS parameter in the volatile database. Removes the MAXIMUM BLOCKS parameter from the permanent database.

MAXIMUM RETRANSMITS (Remote management only)
Applies only to X.25 lines. Resets the maximum number of retransmissions of a frame to the default value for the specified line or all known lines in the volatile database. Removes the MAXIMUM RETRANSMITS parameter from the permanent database.

NIF TARGET
Applies only to FDDI lines. Resets to its default value the NIF TARGET parameter in the volatile database. Permanent database operations cannot be performed on this parameter.

PROTOCOL
Removes the value for the PROTOCOL parameter from the permanent database.

RECEIVE BUFFERS
Resets the RECEIVE BUFFERS parameter to its default value in the volatile database. Removes the RECEIVE BUFFERS parameter from the permanent database. The line must be in the OFF state in order to CLEAR this parameter.

REQUESTED TRT
Applies only to FDDI lines. Resets to its default value the REQUESTED TRT parameter in the volatile database. Removes the REQUESTED TRT parameter from the permanent database.

RESTRICTED TOKEN TIMEOUT
Applies only to FDDI lines. Resets to its default value the RESTRICTED TOKEN TIMEOUT parameter in the volatile database. Removes the RESTRICTED TOKEN TIMEOUT parameter from the permanent database.

RING PURGER ENABLE
Applies only to FDDI lines. Resets to its default value the RING PURGER ENABLE parameter in the volatile database. Removes the RING PURGER ENABLE parameter from the permanent database.

VAX

SCHEDULING TIMER (VAX only)
Applies only to DDCMP lines. Resets the SCHEDULING TIMER parameter to its default value in the database.♦

SERVICE TIMER
Resets to its default value the SERVICE TIMER parameter in the volatile database. Removes the SERVICE TIMER parameter from the permanent database.

SIF CONFIGURATION TARGET
Applies only to FDDI lines. Resets to its default value the SIF CONFIGURATION TARGET parameter in the volatile database. Permanent database operations cannot be performed on this parameter.

SIF OPERATION TARGET
Applies only to FDDI lines. Resets to its default value the SIF OPERATION TARGET parameter in the volatile database. Permanent database operations cannot be performed on this parameter.

STATE
Resets to its default value the STATE parameter in the volatile database. Removes the STATE parameter from the permanent database.

VAX

STREAM TIMER (VAX only)
Applies only to DDCMP lines. Resets the STREAM TIMER parameter to its default value in the database.

SWITCH (VAX only)
Applies only to asynchronous DDCMP lines. Removes the SWITCH parameter from the database.

TRANSMIT PIPELINE (VAX only)
Applies only to DMR11 lines. Removes the TRANSMIT PIPELINE parameter from the database.♦

VALID TRANSMISSION TIME
Applies only to FDDI lines. Resets to its default value the VALID TRANSMISSION TIME parameter in the volatile database. Removes the VALID TRANSMISSION TIME parameter from the permanent database.

Examples

1. `NCP>CLEAR LINE SVA-0 ALL`

   This command removes the Ethernet line called SVA–0 from the volatile database.

VAX

2. `NCP>CLEAR LINE DMB-0 ALL`

   This command removes DDCMP line DMB–0 from the volatile database.♦

VAX

3. `NCP>CLEAR LINE DMB-* ALL`

   This command removes all DDCMP DMB lines from the volatile database.♦

CLEAR/PURGE LOGGING

The CLEAR LOGGING command removes logging parameters from the volatile database. The PURGE LOGGING command removes logging parameters from the permanent database.

Format

CLEAR logging-component parameter

PURGE logging-component parameter

Logging Components

KNOWN LOGGING	LOGGING FILE
LOGGING CONSOLE	LOGGING MONITOR

Logging Components

KNOWN LOGGING
Removes from the database the parameters for all known logging.

LOGGING CONSOLE
Removes from the database all parameters for logging to the console.

LOGGING FILE
Removes from the database all parameters for logging to a file.

LOGGING MONITOR
Removes from the database all parameters for logging to a monitor.

Command Parameters

ALL
Removes all parameters from the database. After you enter the CLEAR/PURGE LOGGING command with this parameter, the logging component is no longer recognized by the network.

CIRCUIT circuit-id
Removes from the database all logging parameters for the specified circuit.

LINE line-id
Removes from the database all logging parameters for the specified line.

NODE node-id
Removes from the database all logging parameters for the specified node.

Examples

1.
```
NCP>CLEAR LOGGING CONSOLE ALL
```

   This command clears logging to the logging console, after which the logging component is no longer recognized by the network.

CLEAR/PURGE LOGGING EVENTS

The CLEAR LOGGING EVENTS command removes event-source logging parameters from the volatile database. The PURGE LOGGING EVENTS command removes event-source logging parameters from the permanent database.

If no source is specified in the command, the source specific events will not be affected.

Format

CLEAR logging-component parameter [...]

PURGE logging-component parameter [...]

Logging Components

KNOWN LOGGING	LOGGING FILE
LOGGING CONSOLE	LOGGING MONITOR

Logging Components

KNOWN LOGGING
Removes from the database the parameters for all known logging.

LOGGING CONSOLE
Removes from the database the parameters for logging to the console.

LOGGING FILE
Removes from the database the parameters for logging to a file.

LOGGING MONITOR
Removes from the database the parameters for logging to a monitor program.

Command Parameters

EVENTS event-list
KNOWN EVENTS
Removes from the database an event class and type or all event classes and types. You can associate any of the following source parameters when you specify the EVENTS parameter.

CIRCUIT circuit-id	Identifies the circuit to which the events apply. Events will not be logged for this circuit.
LINE line-id	Identifies the line to which the events apply. Events will not be logged for this source.
NODE node-id	Identifies the node to which the events apply. Events will not be logged for this source.

SINK	Identifies the node where the event logger runs to log the events. There are two possibilities: EXECUTOR — Identifies the local node NODE node-id — Identifies the node at which events are being logged Only events being logged to this node are removed from the database. If this parameter is omitted, the local node is assumed to be the sink node affected.

Examples

1. `NCP>CLEAR LOGGING FILE EVENTS 2.*`

   This command clears logging to the logging file for all class 2 events.

2. `NCP>CLEAR LOGGING CONSOLE KNOWN EVENTS LINE SVA-1`

   This command clears logging to the console for all known events currently being logged for line SVA–1, an Ethernet line.

CLEAR/PURGE LOGGING NAME

The CLEAR LOGGING NAME command removes the name of the logging sink from the volatile database. The PURGE LOGGING NAME command removes the name of the logging sink from the permanent database.

You must enter a SET LOGGING *component* STATE OFF command before attempting to use the CLEAR command.

Format

CLEAR logging-component NAME

PURGE logging-component NAME

Logging Components

KNOWN LOGGING	LOGGING FILE
LOGGING CONSOLE	LOGGING MONITOR

Logging Components

KNOWN LOGGING

Indicates that the name of the logging sink for all known logging is to be removed from the database.

LOGGING CONSOLE

Indicates that the name of the logging console is to be removed from the database.

LOGGING FILE

Indicates that the name of the logging file is to be removed from the database.

LOGGING MONITOR

Indicates that the name of the logging monitor is to be removed from the database.

Examples

1.
```
NCP>CLEAR LOGGING CONSOLE NAME
```

   This command clears the name of the logging console from the volatile database.

CLEAR/PURGE MODULE CONFIGURATOR

The CLEAR MODULE CONFIGURATOR KNOWN CIRCUITS ALL command removes from the volatile database the Ethernet configurator module component parameters for all circuits. The CLEAR command can not be used to remove individual circuits. The PURGE MODULE CONFIGURATOR ALL command removes from the permanent database the Ethernet configurator module component parameters for all circuits. The PURGE command can be used to remove individual circuits.

Format

PURGE MODULE CONFIGURATOR circuit-component ALL

CLEAR MODULE CONFIGURATOR KNOWN CIRCUITS ALL

Circuit Components

CIRCUIT KNOWN CIRCUITS

Circuit Components

CIRCUIT circuit-id
For the PURGE command only. Applies only to Ethernet circuits. Removes from the permanent database at the local node all configurator module parameters related to the specified circuit.

KNOWN CIRCUITS
Applies only to Ethernet circuits. Removes from the volatile database all configurator module parameters related to all known Ethernet circuits.

Command Parameter

ALL
Removes all parameters for the specified circuit or circuits from the volatile database. After you issue this parameter, the circuit component is no longer recognized by the configurator.

Examples

1. `NCP>PURGE MODULE CONFIGURATOR KNOWN CIRCUITS ALL`

   This command removes from the permanent database at the local node all configurator module parameters related to all known Ethernet circuits.

2. `NCP>PURGE MODULE CONFIGURATOR CIRCUIT SVA-0 ALL`

   This command removes from the permanent database at the local node all configurator module parameters related to the SVA-0 circuit.

3. `NCP>CLEAR MODULE CONFIGURATOR KNOWN CIRCUITS ALL`

   This command removes from the volatile database all configurator module parameters related to all known Ethernet circuits.

CLEAR/PURGE MODULE X25-ACCESS (Remote management only)

The CLEAR MODULE X25-ACCESS command removes from the volatile database network names and parameters used by the host node to access a specified X.25 network or networks through a connector node. The PURGE MODULE X25-ACCESS command removes these same network names and parameters from the permanent database.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

CLEAR MODULE X25-ACCESS qualifier parameter [...]

PURGE MODULE X25-ACCESS qualifier parameter [...]

Qualifiers

NETWORK KNOWN NETWORKS

Qualifiers

NETWORK network-name
KNOWN NETWORKS

The name of a specific network or all of the network names to be removed from the database. After you specify the NETWORK qualifier, you should follow it with one or more of these parameters:

ACCOUNT	Removes user account access control information from the database.
ALL	Removes all X25-ACCESS module parameters from the database.
PASSWORD	Removes the password access control information from the database.
USER	Removes the user identification from the access control information in the database.

Examples

1.
```
NCP>CLEAR MODULE X25-ACCESS NETWORK TELENET1 ALL
```

   This command removes the network TELENET1 from the volatile database in the local node. The local node can no longer connect to X.25 network TELENET1 through the connector node.

2.
```
NCP>CLEAR MODULE X25-ACCESS NETWORK TELENET1 PASSWORD
```

   This command removes the password access control information for the network TELENET1 from the volatile database.

CLEAR/PURGE MODULE X25-PROTOCOL (Remote management only)

The CLEAR MODULE X25-PROTOCOL command removes the protocol module component from the volatile database or resets individual parameters. (Parameters are reset to their default values; the network values are specified by the PROFILE parameter of the SET MODULE X25-PROTOCOL command.) The PURGE MODULE X25-PROTOCOL command removes the parameters of the protocol module component from the permanent database. Note that the word MODULE is optional.

Use separate CLEAR or PURGE MODULE X25-PROTOCOL commands to clear or purge the network, the DTE, or the group.

You may not be able to change the values of some of the network parameters. (Refer to *Public Network Information* manual).

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

CLEAR MODULE X25-PROTOCOL [qualifier] parameter [...]

PURGE MODULE X25-PROTOCOL [qualifier] parameter [...]

Qualifiers

DTE	KNOWN DTES
GROUP	KNOWN GROUPS
NETWORK	KNOWN NETWORKS

Qualifiers

DTE dte-address
KNOWN DTES

Removes parameters for the specified DTE or all known DTEs in the database. If you specify the DTE qualifier, you must associate with it either of the following parameters.

NETWORK network-name	Selects the DTE(s) specified on a particular network. With the PURGE command, you can omit the network part of the qualifier if there is only one network defined.
KNOWN NETWORKS	Selects the DTE(s) on all known networks.

Additionally, you must follow the DTE qualifier with one or more of the following parameters:

ALL	Removes the specified DTE or all known DTEs from the database.

CLEAR/PURGE MODULE X25-PROTOCOL (Remote management only)

CALL TIMER	Resets the call timer to its default value in the database.
CLEAR TIMER	Resets the clear timer to its default value in the database.
COUNTER TIMER	Removes the counter timer for the specified DTE or all known DTEs from the database.
DEFAULT DATA	Resets the default packet size to its default value in the database.
DEFAULT WINDOW	Resets the default window size to its default value in the database.
INTERRUPT TIMER	Resets the interrupt timer to its default value in the database.
MAXIMUM CIRCUITS	Resets the maximum number of virtual circuits allowed at once on the DTE to its default value in the database.
MAXIMUM CLEARS	Resets to the default value in the database the maximum number of attempts to clear a circuit.
MAXIMUM DATA	Resets the maximum packet size to its default value in the database.
MAXIMUM RESETS	Resets to the default value in the database the maximum number of attempts to reset a circuit.
MAXIMUM RESTARTS	Resets to the default value in the database the maximum number of attempts to restart the DTE.
MAXIMUM WINDOW	Resets the maximum window size to its default value in the database.
RESET TIMER	Resets the reset timer to its default value in the database.
RESTART TIMER	Resets the restart timer to its default value in the database.

GROUP group-name
KNOWN GROUPS

Removes parameters for the specified closed user group (CUG) or bilateral closed user group (BCUG), or all known groups, in the database. If you specify a group qualifier, you must follow it with the following parameter:

ALL Removes all parameters for the specified group or all known groups.

NETWORK network-name
KNOWN NETWORKS

Removes all DTEs, groups, and parameters for the specified network or all known networks in the database. You must specify the following parameter:

ALL Removes all parameters for the network or networks.

Examples

1.
```
NCP>CLEAR MODULE X25-PROTOCOL GROUP ESECUG ALL
```

   This command removes the closed user group ESECUG from the volatile database.

2.
```
NCP>PURGE MODULE X25-PROTOCOL GROUP ESECUG ALL
```

   This command removes the parameters of the closed user group ESECUG from the permanent database. When the system is rebooted, the group no longer exists for VAX P.S.I.

3.
```
NCP>CLEAR MODULE X25-PROTOCOL DTE 123789456 NETWORK PSS1 ALL
```

   This command removes the specified DTE belonging to the network PSS1 from the volatile database.

4.
```
NCP>PURGE MODULE X25-PROTOCOL DTE 123789456 KNOWN NETWORKS ALL
```

   This command removes all parameter entries for all the DTEs of that DTE number, for all networks in which the DTE exists, from the permanent database. When the network is restarted, the DTE no longer exists for VAX P.S.I.

5.
```
NCP>CLEAR MODULE X25-PROTOCOL ALL
```

   This command removes all network, DTE, group, and X25-PROTOCOL module parameters from the volatile database.

6.
```
NCP>CLEAR MODULE X25-PROTOCOL NETWORK PSS1 ALL
```

   This command removes all DTEs and groups on network PSS1 and then removes the network name itself.

7.
```
NCP>CLEAR MODULE X25-PROTOCOL KNOWN DTES -
_ KNOWN NETWORKS COUNTER TIMER
```

   This command removes the COUNTER TIMER parameter for all DTEs on all networks from the volatile database.

CLEAR/PURGE MODULE X25-SERVER/X29-SERVER (Remote management only)

The CLEAR MODULE X25-SERVER or CLEAR MODULE X29-SERVER command resets or removes the X.25 or X.29 call handler from the volatile database or removes individual parameters. The PURGE MODULE X25-SERVER or PURGE MODULE X29-SERVER command removes the parameters of the X.25 or X.29 call handler from the permanent database.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

CLEAR MODULE X25-SERVER	[qualifier] parameter [...]
CLEAR MODULE X29-SERVER	[qualifier] parameter [...]
PURGE MODULE X25-SERVER	[qualifier] parameter [...]
PURGE MODULE X29-SERVER	[qualifier] parameter [...]

Qualifiers

DESTINATION	KNOWN DESTINATIONS

Qualifiers

DESTINATION dest-name
KNOWN DESTINATIONS

Identifies the destination or all known destinations whose parameters are to be reset or removed from the database. If you specify a destination qualifier, you must follow it with one or more of the following parameters.

ACCOUNT	Removes from the database the account for the specified destination or all known destinations.
ALL	Removes from the database all parameters for the specified destination or all known destinations.
CALL MASK	Removes from the database the call mask for the specified destination or all known destinations.
CALL VALUE	Removes from the database the call value for the specified destination or all known destinations.
CALLED ADDRESS	Removes from the database the called DTE address for the specified destination or all known destinations.

CLEAR/PURGE MODULE X25-SERVER/X29-SERVER (Remote management only)

EXTENSION MASK	Removes from the database the extension mask for the specified destination or all known destinations.
EXTENSION VALUE	Removes from the database the extension value for the specified destination or all known destinations.
GROUP	Removes from the database the group name for the specified destination or all known destinations.
INCOMING ADDRESS	Removes from the database the incoming address for the specified destination or all known destinations.
NETWORK	Removes from the database the network name for the specified destination or all known destinations.
OBJECT	Removes from the database the object identification for the specified destination or all known destinations.
PASSWORD	Removes from the database the password for the specified destination or all known destinations.
PRIORITY	Resets to the lowest value the priority for the specified destination or all known destinations.
RECEIVING DTE	Removes from the database the receiving DTE address for the specified destination or all known destinations.
REDIRECT REASON	Removes from the database the redirect reason code for the specified destination or all known destinations.
SENDING ADDRESS	Removes from the database the sending DTE address for the specified destination or all known destinations.
SUBADDRESSES	Removes from the database the subaddress range for the specified destination or all known destinations.
USER	Removes from the database the user identification for the specified destination or all known destinations.

Command Parameter

COUNTER TIMER

Resets the module counter timer to 0. This is the only parameter with which you are not required to supply a destination qualifier.

Examples

1. `NCP>CLEAR MODULE X25-SERVER KNOWN DESTINATIONS ALL`

   This command removes all X.25 destinations from the volatile database.

2. `NCP>PURGE MODULE X25-SERVER KNOWN DESTINATIONS ALL`

   This command removes all parameters for all X.25 destinations from the permanent database.

CLEAR/PURGE NODE

The CLEAR NODE command removes node parameters from the volatile database. The PURGE NODE command removes node parameters from the permanent database on the local node.

To remove a loop node name from the database, use the CLEAR/PURGE NODE node-id command and specify either the ALL parameter (with the loop node name or address as the node-id), or the CIRCUIT parameter.

Format

CLEAR node-component parameter [...]

PURGE node-component parameter [...]

Node Components

NODE	KNOWN NODES

Node Components

NODE node-id
Identifies the node (local or remote) whose parameters are to be removed from the database.

KNOWN NODES
Removes from the database the specified parameters for all known nodes.

Command Parameters

ACCESS
Removes from the database the value for the ACCESS parameter.

ALL
Removes from the database all parameters for the specified node or all known nodes. The component is no longer recognized by the network.

Caution

The PURGE KNOWN NODES ALL command purges both the executor and remote node databases.

COUNTER TIMER
Removes the COUNTER TIMER value from the database.

CPU
Removes the node's CPU identification from the database.

DIAGNOSTIC FILE
Removes from the database the identification of the downline loading diagnostics file.

DUMP ADDRESS
Removes from the database the upline dump address of the adjacent node.

DUMP COUNT
Removes the upline dump count from the database.

DUMP FILE
Removes the upline dump file identification from the database.

HARDWARE ADDRESS
Removes from the database the hardware address that is associated with a LAN controller that is connected to the node.

HOST
Removes from the database the identification of the host node.

VAX

INBOUND (VAX only)
Applies only to nodes connected by DDCMP point-to-point circuits. Removes the INBOUND parameter from the database.♦

LOAD ASSIST AGENT
Removes from the database the identification of the image used to assist in performing a downline load. The LOAD ASSIST AGENT applies only when the target is a VMScluster satellite.

LOAD ASSIST PARAMETER
Removes from the database the value to be passed to the load assist agent. The LOAD ASSIST PARAMETER value applies only when the target is a VMScluster satellite.

LOAD FILE
Removes from the database the name of the file to be downline loaded.

MANAGEMENT FILE
Removes from the database the identification of the downline loading management information file.

NAME
Removes from the database the node name parameter for the node.

NONPRIVILEGED item
Removes the specified nonprivileged access control information from the database. Specify any or all of the following items for removal:

ACCOUNT
PASSWORD
USER

PRIVILEGED item
Removes the specified privileged access control information from the database. Specify any or all of the following items for removal:

ACCOUNT
PASSWORD
USER

RECEIVE PASSWORD
Removes from the database the receive password expected from the remote node during a Routing initialization sequence.

SECONDARY LOADER
Removes from the database the identification of the secondary downline loading file.

SERVICE CIRCUIT
Removes from the database the circuit name associated with the node for downline loading purposes.

SERVICE DEVICE
Removes the service device type from the database. See Appendix A for a list of the DECnet for OpenVMS circuit and line devices.

VAX

Only the synchronous port on the DMF may be used for service operations.♦

SERVICE NODE VERSION
Removes from the database the software version number of the node to be downline loaded.

SERVICE PASSWORD
Removes from the database the password required to trigger the bootstrap mechanism of the node to be downline loaded.

SOFTWARE IDENTIFICATION
Removes the identification of the software from the database.

SOFTWARE TYPE
Removes the initial load software program type from the database.

TERTIARY LOADER
Removes from the database the identification of the tertiary downline loading file.

TRANSMIT PASSWORD
Removes from the database the transmit password sent to the remote node during a Routing initialization sequence.

Examples

1.
```
NCP>CLEAR NODE TRNTO ALL
```

   This command removes all parameter entries for node TRNTO from the volatile database. As a result, the node no longer exists in the volatile database.

2.
```
NCP>CLEAR NODE 14 NAME
.
.
.
NCP>SET NODE 15 NAME DENVER
```

   The first command disassociates the name DENVER from node 1.14. The second command associates the same name with node 1.15. In single area networks the default area is 1.

3.
```
NCP>CLEAR NODE 5.14 NAME
.
.
.
NCP>SET NODE 5.15 NAME DENVER
```

   The first command disassociates the name DENVER from node 5.14. The second command associates the same name with node 5.15. Use this example in place of the previous example when the network is divided into multiple areas.

4.
```
NCP>CLEAR NODE BOSTON RECEIVE PASSWORD
```

   This command removes the receive password from the record for the local node from the volatile database.

5.
```
NCP>CLEAR NODE 5.* NAME
```

   This command disassociates the node names from their addresses for all nodes in area 5.

CLEAR/PURGE NODE CIRCUIT

The CLEAR NODE CIRCUIT command removes a loop node from the volatile database. The PURGE NODE CIRCUIT command removes a loop node from the permanent database on the executor node.

Format

CLEAR node-component CIRCUIT

PURGE node-component CIRCUIT

Node Components

NODE	KNOWN NODES

Node Components

NODE node-id
Identifies the loop node to be removed from the database.

KNOWN NODES
Indicates that all loop nodes are to be removed from the database.

Examples

1.
```
NCP>SET NODE TESTER CIRCUIT FZA-0
.
.
.
NCP>CLEAR NODE TESTER ALL
```

The first command sets a loop node for circuit FZA–0. The second command removes the loop node from the volatile database.

CLEAR/PURGE OBJECT

The CLEAR OBJECT command removes object parameters from the volatile database on the local node. The PURGE OBJECT command removes object parameters from the permanent database.

Format

CLEAR object-component parameter [...]

PURGE object-component parameter [...]

Object Components

KNOWN OBJECTS OBJECT

Object Components

KNOWN OBJECTS

Indicates that parameters for all known objects are to be removed from the database.

OBJECT object-name

Identifies the object whose parameters are to be removed from the database.

Command Parameters

ACCOUNT

Removes the account access control information from the database.

ALIAS INCOMING

Removes the ALIAS INCOMING parameter for the specified object or for all known objects from the database.

ALIAS OUTGOING

Removes the ALIAS OUTGOING parameter for the specified object or for all known objects from the database.

ALL

Removes all parameters for the specified object or all known objects from the database. After you enter this parameter, the component is no longer recognized by the network.

FILE

Removes the name of the object's startup command file from the database.

OUTGOING CONNECT PRIVILEGES

Removes from the database the privileges required to make an outbound connection to the object.

PASSWORD

Removes the password access control information from the database.

PRIVILEGES

Removes from the database the privileges required to access the object.

PROXY
Removes from the database the proxy access information for the object.

USER
Removes the user access control information from the database.

Examples

1. `NCP>CLEAR OBJECT MYOBJ ALL`

   This command removes all parameter entries for the object MYOBJ from the volatile database. This command removes the object from the volatile database.

CONNECT NODE

The CONNECT NODE command sets up a logical connection between the host node and the console interface on a specified target node. Both the host node and the target node must be on the same Ethernet or FDDI circuit.

To define information for the target node in the database, use the NCP command SET NODE to specify the ADDRESS, SERVICE CIRCUIT, SERVICE PASSWORD, and HARDWARE ADDRESS parameters. Once you set these parameters, you can override them by specifying new parameters in the CONNECT command.

Format

CONNECT node-component [parameter] [...]

Node Component
NODE

Node Component

NODE node-id
Identifies the node name or address of the target node to be connected to the local node.

Command Parameters

PHYSICAL ADDRESS p-address
Specifies the physical address of the target node. The value is the physical address that the target node has set for itself or, if the target node has not set a physical address, the hardware address associated with the target node's device controller.

SERVICE PASSWORD hex-password
Identifies the password required to create the link between the host node and the target node. The password is a 16-digit hexadecimal number in the range 0 to FFFFFFFFFFFFFFFF.

VIA circuit-id
Specifies the circuit to be used to create the logical link between the host node and the target node. This circuit must be an Ethernet or FDDI circuit.

Examples

1.
```
NCP>CONNECT NODE RTRDEV SERVICE PASSWORD FEFEFEFEFEFEFEFE -
_ VIA UNA-0 PHYSICAL ADDRESS AA-00-04-00-38-04
```

   This command connects the host node to the console interface on the target node RTRDEV specifying the service password FEFEFEFEFEFEFEFE, the service circuit UNA–0, and the physical address AA–00–04–00–38–04.

CONNECT VIA

The CONNECT VIA command sets up a logical connection between the host node and the console interface on a target node using the specified circuit. If the DECnet node address of the target node is not known, use the CONNECT VIA command instead of the CONNECT NODE command. Both the host node and the target node must be on the same circuit.

Format

CONNECT VIA circuit-id PHYSICAL ADDRESS p-address parameter

Command Parameters

PHYSICAL ADDRESS p-address
Specifies the physical address of the target node. The value is the physical address that the target node has set for itself. If the target node has not set a physical address, the value is the hardware address associated with the node's controller. This parameter is required.

SERVICE PASSWORD hex-password
Identifies the password required to create the link between the host node and the target node. The password is a hexadecimal number in the range 0 to FFFFFFFFFFFFFFFF.

Examples

1.
```
NCP>CONNECT VIA UNA-0 PHYSICAL ADDRESS AA-00-04-00-38-00
```

   This command connects the host node to the console interface on the target node by specifying the circuit over which the connection is to be made and the physical address of the target node.

COPY KNOWN NODES

The COPY KNOWN NODES command updates the node database on the local node. You can copy the volatile or permanent node database from a remote node to either or both the volatile and permanent node databases on the local node. You also have the option of clearing or purging the node database on the local node before beginning the copy operation.

Only the node name and node address are copied. A node entry will not be copied into the node database if a node with the same name or address already exists in the database.

You cannot use the TELL prefix with the COPY KNOWN NODES command.

Format

COPY node-component parameter [qualifier] [...]

Node Component

KNOWN NODES

Qualifiers

USING WITH
TO

Node Component

KNOWN NODES

Indicates that names and addresses of all known nodes stored in the database of the specified remote node are to be copied.

Command Parameter

FROM node-id

Specifies the remote node from which node database information is to be copied. The remote node can be any node in the network to which you have access. The word FROM is optional.

Qualifiers

USING option

Specifies the node database on the remote node from which the information is to be copied. There are two possible options:

VOLATILE Indicates that the volatile database on the remote node is to be copied.

PERMANENT Indicates that the permanent database on the remote node is to be copied.

The default is VOLATILE.

TO option

Specifies the node database on the local node to which the information is to be copied. There are three possible options:

VOLATILE Indicates that the information is to be copied to the volatile database on the local node.

PERMANENT Indicates that the information is to be copied to the permanent database on the local node.

BOTH Indicates that the information is to be copied to both the volatile and permanent databases on the local node.

The default is VOLATILE.

WITH option

Clears or purges the node database on the local node before the copy operation is performed. Retains the executor node characteristics and the name and address of the remote node from which the node information is to be copied. The node database to be cleared or purged is the local database to which the information will be copied. There are two options:

CLEAR Clears the volatile node database at the local node.

PURGE Purges the permanent node database at the local node.

Note that you can specify CLEAR or PURGE for either database or for both databases.

If you do not specify the WITH qualifier, the node entries copied are added to the existing node databases.

Examples

1.
```
NCP>LIST KNOWN NODES

Known Node Permanent Summary as of 15-JUN-1992 13:50:20

Executor node = 2.20 (ROBIN)
State                    = on

Remote node =   2.21 (THRUSH)
No information available

Remote node =   2.22 (LARK)
No information available

NCP>TELL LARK LIST KNOWN NODES

Known Node Permanent Summary as of 15-JUN-1992 13:50:20

Executor node = 2.22 (LARK)
State                    = on

Remote node =   2.20 (ROBIN)
No information available

Remote node =   2.23 (DOVE)
No information available

NCP>COPY KNOWN NODES FROM LARK USING PERMANENT -
_ TO PERMANENT WITH PURGE
```

```
%NCP-I-SUCCESS - Success
Remote node = 2.21 (THRUSH)
%NCP-I-RECDELET, Database entry deleted
%NCP-I-SUCCESS - Success
Remote node = 2.22 (LARK)
%NCP-I-RECDELET, Database entry deleted
%NCP-I-SUCCESS - Success
Executor node = 2.20 (ROBIN)
%NCP-I-RECDELET, Database entry deleted


NCP>LIST KNOWN NODES

Known Node Permanent Summary as of 15-JUN-1992 14:01:05

Executor node = 2.20 (ROBIN)
State                    = on


Remote node =   2.22 (LARK)
No information available

Remote node =   2.23 (DOVE)
No information available
```

This copy command copies the node information from the permanent node database on node LARK into the permanent node database on the local node (ROBIN). The node database is purged before the copy operation is begun.

2.
```
NCP>SHOW KNOWN NODES

Known Node Volatile Summary as of 15-JUN-1992 13:50:20

Executor node = 2.20 (ROBIN)
State                    = on
Identification           = DECnet for OpenVMS
Active links             = 2

    Node           State      Active  Delay   Circuit     Next node
                              Links
 2.21 (THRUSH)                                QNA-0         2.19 (EGRET)
 2.22 (LARK)                                  QNA-0         2.19 (EGRET)

NCP>TELL LARK SHOW KNOWN NODES

Known Node Volatile Summary as of 15-JUN-1992 13:53:27

Executor node = 2.22 (LARK)
State                    = on
Identification           = DECnet for OpenVMS
Active links             = 2

    Node           State      Active  Delay   Circuit     Next node
                              Links
 2.20 (ROBIN)                                 QNA-0         2.19 (EGRET)
 2.23 (DOVE)                                  QNA-0         2.19 (EGRET)

NCP>COPY KNOWN NODES FROM LARK
```

```
NCP>SHOW KNOWN NODES

Known Node Volatile Summary as of 15-JUN-1992 14:01:05

Executor node = 2.20 (ROBIN)
State                    = on
Identification           = DECnet for OpenVMS
Active links             = 2

    Node           State      Active  Delay   Circuit     Next node
                              Links
 2.21 (THRUSH)                                QNA-0         2.19 (EGRET)
 2.22 (LARK)                                  QNA-0         2.19 (EGRET)
 2.23 (DOVE)                                  QNA-0         2.19 (EGRET)
```

This COPY command copies the node information from the volatile nodes database on node LARK into the volatile nodes database on the local node (ROBIN). In this example, the volatile nodes database on the local node is not cleared before the copy operation is begun.

DISCONNECT LINK

The DISCONNECT LINK command disconnects the logical links active on the local node.

Format

DISCONNECT link-component [qualifier]

Link Components

LINK KNOWN LINKS

Qualifier

WITH NODE

Link Components

KNOWN LINKS

Indicates that all links are to be disconnected.

LINK number

Identifies the particular logical link to be disconnected. Use the SHOW KNOWN LINKS command to display the active logical links.

Qualifier

WITH NODE node-id

Identifies the node with which all logical links are to be disconnected. The *node-id* is either a node name or a node address. You can use this command verb only with the KNOWN LINKS component.

Examples

1.
```
NCP>SHOW KNOWN LINKS

Known Link Volatile Summary as of 15-JUN-1992 15:53:42

   Link       Node           PID     Process     Remote link  Remote user
  4104   11.9 (DALLAS)     20600090  GRAY               3116  MAIL
  8229    3.5 (TRNTO)      2060014C  DAVIS              7217  NML
  3125    2.17 (NYC)       2060023D  REMACP             5175  THOMPSON_1

NCP>DISCONNECT KNOWN LINKS WITH NODE DALLAS
```

   The first command, SHOW KNOWN LINKS, displays the link addresses. The second command disconnects all known links to remote node DALLAS.

2.
```
NCP>SET EXECUTOR NODE DENVER
.
.
.
NCP>DISCONNECT KNOWN LINKS
NCP>CLEAR EXECUTOR NODE
```

   The first command sets the executor node to DENVER. The DISCONNECT KNOWN LINKS command disconnects the link to node DENVER. As a result, you must either enter the SET EXECUTOR NODE command again, or you must enter the CLEAR EXECUTOR NODE command to reestablish the local node as executor.

HELP

The HELP command displays general information about NCP commands and parameters. See Section 1.5 for more information about the NCP HELP Facility.

Format

HELP [topic...]

Command Parameter

topic

Specifies one or more NCP keywords that refer to a topic or subtopic on which you can obtain information from the NCP help library. Information in the NCP help library is organized in a hierarchial manner, in topics and subtopics. You can specify up to eight topics and subtopics, separated by spaces.

Examples

1.
```
NCP>HELP
Information available:

   CLEAR      Commands   CONNECT    COPY       DEFINE     DISCONNECT EVENTS
   EXIT       HELP       LIST       LOAD       LOOP       PARAMETERS Prompting
   PURGE      SET        SHOW       TELL       TRIGGER    ZERO

Topic?
```

   This command displays all command verbs for which further information exists.

2.
```
NCP>HELP CLEAR CIRCUIT
CLEAR

CIRCUIT

    The CLEAR CIRCUIT command resets circuit parameters in the
    volatile database to the default value (if any), or
    removes them from the volatile database on the local node or DTE.

    The PURGE CIRCUIT command always removes circuit parameters from
    the permanent database on the local node or DTE.

    CLEAR   KNOWN CIRCUITS        (parameters...)
            CIRCUIT circuit-id

Additional information available:

    KNOWN CIRCUITS        CIRCUIT circuit-id    ACTIVE BASE
    ACTIVE INCREMENT      ALL        BABBLE TIMER          COUNTER TIMER
    DEAD THRESHOLD        DYING BASE DYING INCREMENT       DYING THRESHOLD
    INACTIVE BASE         INACTIVE INCREMENT    INACTIVE THRESHOLD
    MAXIMUM BUFFERS       MAXIMUM RECALLS       MAXIMUM ROUTERS
    MAXIMUM TRANSMITS     NETWORK    RECALL TIMER          ROUTER PRIORITY
    TRANSMIT TIMER        Examples
```

   This command provides a description of the NCP command CLEAR CIRCUIT and displays command words for which further information exists.

3. NCP>HELP SHOW

```
SHOW

  Use the SHOW command to display information from the volatile database
  on the executor node.  Use the LIST command to display information
  from the permanent database on the executor node.

Additional information available:

  CHARACTERISTICS       COUNTERS   EVENTS     STATUS     SUMMARY
  TO file-id AREA       CIRCUIT    EXECUTOR   LINE       LINKS      LOGGING
  MODULE     NODE       OBJECT
```

This command provides a description of the NCP command SHOW and displays command words for which further information exists.

LOAD NODE

The LOAD NODE command downline loads software to a target node, using either the identified circuit (if specified) or the circuit obtained from the volatile database. Any parameter left unspecified in the command defaults to whatever entry is specified in the volatile node database on the executor node.

Format

LOAD node-component parameter [...]

Node Component
NODE

Node Component

NODE node-id
Identifies the node name or address of the target node to be downline loaded.

Command Parameters

ADDRESS node-address
Identifies the address that the target node is to use (when it comes up).

CPU cpu-type
Identifies the node's CPU type. There are four possibilities:

DECSYSTEM1020
PDP11
PDP8
VAX

FROM filespec
Identifies the file specification of the load file containing the system software to be downline loaded.

HOST node-id
Identifies the default host that the target node is to use (when it comes up).

LOAD ASSIST AGENT filespec
Specifies the image that defines the system software to be downline loaded to the target node. The load assist agent can be used to tailor the system software to be loaded. The LOAD ASSIST AGENT parameter applies only when the target is a VMScluster satellite.

LOAD ASSIST PARAMETER item
Specifies a parameter to be passed to a load assist agent. The LOAD ASSIST PARAMETER value applies only when the target is a VMScluster satellite.

MANAGEMENT FILE filespec
Specifies a file containing the management information to be downline loaded to the target node.

NAME node-name
Identifies the name that the target node is to use (when it comes up).

PHYSICAL ADDRESS p-address
Applies only to nodes on Ethernet or FDDI circuits. Specifies the address that the target node to be downline loaded currently uses to identify itself. The value is the physical address that the target node has set for itself or, if the target node has not set a physical address, the hardware address associated with the target node's device controller.

SECONDARY LOADER filespec
Identifies the file specification of a secondary load file containing the software to be downline loaded.

SERVICE DEVICE device-type (VAX only)
Identifies the target node's line controller for the service line over which the operation is to take place. See Appendix A for a list of the DECnet for OpenVMS circuit and line devices and what they support.

VAX

Only the synchronous port on the DMF service device may be used for service operations.♦

SERVICE PASSWORD hex-password
Identifies the password required to trigger the bootstrap mechanism on the target node. The password is a hexadecimal number. For FDDI or Ethernet circuits, the number is in the range 0 to FFFFFFFFFFFFFFFF.

VAX

For DDCMP circuits, the number is in the range 0 to FFFFFFFF.♦

SOFTWARE IDENTIFICATION software-id
Identifies the name of the software to be downline loaded.

SOFTWARE TYPE software-type
Identifies the particular file type to be loaded first. There are four possibilities:

MANAGEMENT FILE
SECONDARY LOADER
SYSTEM
TERTIARY LOADER

TERTIARY LOADER filespec
Identifies the file specification of a tertiary load file containing the software to be downline loaded.

VIA circuit-id
Identifies a circuit over which the loading sequence is to take place.

Examples

1. `NCP>LOAD NODE NYC HOST BOSTON VIA FZA-1`

   This command initiates a downline load operation for node NYC over circuit FZA-1. When loaded, node NYC has node BOSTON as its default host specification.

2. `NCP>LOAD NODE BANGOR`

   This command initiates a downline load operation for node BANGOR. Any required default information is retrieved from the volatile database on the executor node.

3. `NCP>LOAD NODE SWIFT PHYSICAL ADDRESS AA-00-04-00-07-04`

   This command initiates a downline load operation for node SWIFT on an Ethernet circuit. The executor node uses the physical address specified in the command to address node SWIFT.

LOAD VIA

The LOAD VIA command downline loads software to a target node over the specified circuit. The target node identification that is associated with the service circuit is obtained from the volatile database on the executor node. If the target node is on an Ethernet or FDDI circuit, you must specify the PHYSICAL ADDRESS parameter in this command.

Format

LOAD VIA circuit-id parameter [...]

Command Parameters

ADDRESS node-address
Identifies the address that the target node is to use (when it comes up).

CPU cpu-type
Identifies the node's CPU type. There are four possibilities:

DECSYSTEM1020
PDP11
PDP8
VAX

FROM filespec
Identifies the file specification of the load file containing the system software to be downline loaded.

HOST node-id
Identifies the default host that the target node is to use (when it comes up).

LOAD ASSIST AGENT filespec
Specifies the image that defines the system software to be downline loaded to the target node. The load assist agent can be used to tailor the system software to be loaded. The LOAD ASSIST AGENT parameter applies only when the target is a VMScluster satellite.

LOAD ASSIST PARAMETER item
Specifies a parameter to be passed to a load assist agent. The LOAD ASSIST PARAMETER value applies applies only when the target is a VMScluster satellite.

MANAGEMENT FILE filespec
Specifies a file containing the management information to be downline loaded to the target node.

NAME node-name
Identifies the name that the target node is to use (when it comes up).

PHYSICAL ADDRESS p-address
Applies only to nodes on Ethernet or FDDI circuits. Specifies the physical address that the target node to be downline loaded currently uses to identify itself. The value is the address that the target node has set for itself, or, if the target node has not set a physical address, the hardware address associated with the target node's device controller.

LOAD VIA

SECONDARY LOADER filespec
Identifies the file specification of a secondary load file containing the software to be downline loaded.

SERVICE DEVICE device-type (VAX only)
Identifies the target node's line controller for the service line over which the operation is to take place. See Appendix A for a list of the DECnet for OpenVMS circuit and line devices and what they support.

VAX

Only the synchronous port on the DMF service device may be used for service operations.♦

SERVICE PASSWORD hex-password
Identifies the password required to trigger the bootstrap mechanism on the target node. The password is a hexadecimal number. For Ethernet or FDDI circuits, it is in the range 0 to FFFFFFFFFFFFFFFF.

VAX

For DDCMP circuits, it is in the range 0 to FFFFFFFF.♦

SOFTWARE IDENTIFICATION software-id
Identifies the name of the software to be downline loaded.

SOFTWARE TYPE software-type
Identifies the particular file type to be loaded first. There are four possibilities:

MANAGEMENT FILE
SECONDARY LOADER
SYSTEM
TERTIARY LOADER

TERTIARY LOADER filespec
Identifies the file specification of a tertiary load file containing the software to be downline loaded.

Examples

VAX

1.
```
NCP>LOAD VIA DMP-0.4
```

   This command initiates a downline load operation over circuit DMP-0.4. Any required default information is retrieved from the volatile database on the executor node. The node database is scanned until a node is found whose service circuit matches the load circuit.

2.
```
NCP>LOAD VIA DMC-0 SOFTWARE TYPE SECONDARY LOADER
```

   This command initiates a downline load operation over circuit DMC–0, commencing with the secondary loader file.♦

3.
```
NCP>LOAD VIA SVA-0 PHYSICAL ADDRESS AA-00-04-00-07-04
```

   This command initiates a downline load operation over Ethernet circuit SVA–0, to the target node whose physical address is specified in the command.

LOOP CIRCUIT

The LOOP CIRCUIT command tests a circuit by transmitting test blocks of data over a specified circuit. The parameters are optional and can be entered in any order.

Format

LOOP circuit-component [parameter] [...]

Circuit Component
CIRCUIT

Circuit Component

CIRCUIT circuit-id
Identifies the circuit for loopback testing.

Command Parameters

ASSISTANT NODE node-id
Applies only to Ethernet and FDDI circuits. Identifies the name or the address of the node that is to perform the role of loopback assistant for third-party loopback testing.

ASSISTANT PHYSICAL ADDRESS p-address
Applies only to Ethernet and FDDI circuits. Identifies the physical address of the node that is to perform the role of loopback assistant for third-party loopback testing. If the HELP parameter is included in this command, you must specify ASSISTANT PHYSICAL ADDRESS. The address cannot be a multicast address.

COUNT number
Specifies the number of blocks to be sent during loopback testing over the circuit. The number must be a decimal integer in the range of 1 through 65,535. If you omit this parameter, only one block is looped.

HELP help-type
Applies only to Ethernet and FDDI circuits. Indicates the type of assistance to be provided during loopback testing by the assistant node, whose address is specified in the ASSISTANT PHYSICAL ADDRESS or *node-id* as specified in the ASSISTANT NODE parameter. There are three possible values for *help-type*.

FULL	The assistant node relays the request and the reply between the executor node and the destination node.
RECEIVE	The executor node sends the request directly to the destination node, which relays the reply to the assistant node for transmission to the executor node.
TRANSMIT	The assistant node relays the request to the destination node, which replies directly to the executor node.

If you specify the HELP parameter, you must also specify the ASSISTANT PHYSICAL ADDRESS or ASSISTANT NODE parameter.

LENGTH number

Specifies the length (in bytes) of the blocks to be sent during loopback testing. The length must be a decimal integer in the range of 1 through 4096. If you omit this parameter, a block length of 4096 bytes is used.

For further information on increasing the loop length, refer to the description of the MIRROR$SIZE logical in the *DECnet for OpenVMS Networking Manual*.

VAX

The LENGTH parameter must be less than 50 for the DMC–11 circuit operating in either controller loopback mode or cable loopback (a DMC–11 with an attached loopback cable).♦

NODE node-id

Applies only to Ethernet and FDDI circuits. Identifies the destination node to be used for loopback testing of the specified broadcast circuit. You can use this parameter instead of the PHYSICAL ADDRESS parameter.

PHYSICAL ADDRESS p-address

Applies only to Ethernet and FDDI circuits. Identifies the physical address of the destination node to be used for loopback testing of the specified broadcast circuit.

WITH data-type

Specifies the type of binary information to be sent during testing. If you omit this parameter, a combination of ones and zeros (the MIXED data type) is sent. There are three types of data:

MIXED
ONES
ZEROS

Examples

VAX

1.
```
NCP>LOOP CIRCUIT DMB-0 COUNT 5 LENGTH 20
```

   This command initiates a circuit-level loopback test over circuit DMB–0. The software loops five messages, 20 bytes in length with mixed binary information.♦

2.
```
NCP>LOOP CIRCUIT SVA-0 PHYSICAL ADDRESS AA-00-04-00-FF-04
```

   This command initiates a circuit-level loopback test with a node whose physical address is AA–00–04–00–FF–04.

3.
```
NCP>LOOP CIRCUIT SVA-0 NODE 224
```

   This command initiates a circuit-level loopback test with a node whose address is 1.224. In this example, the NODE parameter with a *node-id* value of 1.224 is used in place of the PHYSICAL ADDRESS parameter. In single area networks the default area is 1.

4.
```
NCP>LOOP CIRCUIT SVA-0 PHYSICAL ADDRESS -
_ AA-00-04-00-12-04 ASSISTANT NODE GULL HELP RECEIVE
```

   This command initiates a circuit-level loopback test by transmitting a message directly to a node whose physical address is AA–00–04–00–12–04. Additionally, the command requests the assistance of an Ethernet node named GULL to receive the message from the target node and then relay the message back.♦

LOOP EXECUTOR

The LOOP EXECUTOR command tests the executor node by causing test data to be transmitted to the executor node. The parameters are optional and can be entered in any order. You can supply explicit access control information for the LOOP EXECUTOR command.

Format

LOOP EXECUTOR [parameter] [...]

Command Parameters

ACCOUNT account

Identifies the user's account for access control verification for the executor node.

COUNT number

Specifies the number of blocks to be sent during loopback testing over the executor node. The number must be a decimal integer in the range of 1 through 65,535. If you omit this parameter, only one block is looped.

LENGTH number

Specifies the length (in bytes) of the blocks to be sent during loopback testing. The length must be a decimal integer in the range of 1 through 4096. If you omit this parameter, a block length of 4096 bytes is used.

For further information on increasing the loop length, refer to the description of the MIRROR$SIZE logical in the *DECnet for OpenVMS Networking Manual*.

PASSWORD password

Identifies the user's password for access control verification for the executor node.

USER user-id

Specifies the user's identification for access control verification for the executor node.

WITH data-type

Specifies the type of binary information to be sent during testing. If you omit this parameter, a combination of ones and zeros (the MIXED data type) is sent. There are three types of data:

MIXED
ONES
ZEROS

Examples

1. `NCP>LOOP EXECUTOR`

   This command initiates a loopback test on the executor node.

LOOP LINE (Remote management only)

The LOOP LINE command initiates loopback tests of X.25 lines only. The parameters are optional and can be entered in any order. Before beginning loopback testing of an X.25 line, use the SET LINE command to set the STATE SERVICE and CONTROLLER LOOPBACK parameters for internal loopback testing or the STATE SERVICE and CONTROLLER NORMAL parameters for all other loopback testing.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

LOOP line-component [parameter] [...]

Line Component

LINE

Line Component

LINE line-id

Applies only to VAX P.S.I. Identifies the X.25 line for loopback testing. Specify a line name using the *dev-c-u* format for the *line-id*.

Command Parameters

COUNT number

Specifies the number of blocks to be sent during loopback testing over the line. The number must be a decimal integer in the range of 1 through 65,535. If you omit this parameter, only one block is looped.

LENGTH number

Specifies the length (in bytes) of the blocks to be sent during loopback testing. The length must be a decimal integer in the range of 1 through 4096. If you omit this parameter, a block length of 4096 bytes is used.

For further information on increasing the loop length, refer to the description of the MIRROR$SIZE logical in the *DECnet for OpenVMS Networking Manual*.

WITH data-type

Specifies the type of binary information to be sent during testing. If you omit this parameter, a combination of ones and zeros (the MIXED data type) is sent. There are three types of data:

MIXED
ONES
ZEROS

Examples

1. `NCP>LOOP LINE DUP-0`

   This command initiates a line-level loopback test over X.25 line DUP–0. The software loops one message, 4096 bytes in length, with mixed binary information.

2. `NCP>LOOP LINE DUP-1 COUNT 10 WITH ONES`

   This command initiates a line-level loopback test over X.25 line DUP–1. The software loops 10 messages, 4096 bytes in length, with all binary ones information.

LOOP NODE

The LOOP NODE command tests a specified node in the network by causing test blocks of data to be transmitted to the node and then looped back to the executor. The parameters are optional and can be entered in any order. You can supply explicit access control information for the LOOP NODE command.

To test the executor node, use the LOOP EXECUTOR command.

Format

LOOP node-component [parameter] [...]

Node Component
NODE

Node Component

NODE node-id
Identifies the node on which the loopback mirror is to run.

Command Parameters

ACCOUNT account
Identifies the user's account for access control verification on the designated node.

COUNT number
Specifies the number of blocks to be sent during the loopback test with the remote node. The number must be a decimal integer in the range of 1 through 65,535. If you omit this parameter, only one block is looped.

LENGTH number
Specifies the length (in bytes) of the blocks to be sent during loopback testing. The length must be a decimal integer in the range of 1 through 4096. If you omit this parameter, a block length of 4096 bytes is used.

VAX

The LENGTH parameter must be less than 50 for the DMC–11 circuit operating in either controller loopback mode or cable loopback (a DMC–11 with an attached loopback cable).♦

PASSWORD password
Identifies the user's password for access control verification on the designated node.

USER user-id
Specifies the user's identification for access control verification on the designated node.

WITH data-type

Specifies the type of binary information to be sent during testing. If you omit this parameter, a combination of ones and zeros (the MIXED data type) is sent. There are three types of data:

MIXED
ONES
ZEROS

Examples

1.
```
NCP>SET NODE TESTER CIRCUIT FZA-0
NCP>LOOP NODE TESTER
.
.
.
NCP>CLEAR NODE TESTER ALL
```

The first command creates a loop node name, TESTER, for the associated circuit. The second command initiates a node-level loopback test with the loop node. The third command removes the loop node from the volatile database when the loop test is completed.

SET CIRCUIT ALL

The SET CIRCUIT ALL command updates the volatile copy of the database on the local node with all the circuit parameters stored for a particular circuit or for all known circuits in the permanent database on the local node.

Format

SET circuit-component ALL

Circuit Components

CIRCUIT	KNOWN CIRCUITS

Circuit Components

CIRCUIT circuit-id
Updates the parameters for a specific circuit.

KNOWN CIRCUITS
Updates all parameters for all known circuits.

Examples

1. `NCP>SET KNOWN CIRCUITS ALL`

   This command loads all permanent database entries stored for all known circuits into the volatile database.

SET/DEFINE CIRCUIT

The SET CIRCUIT command creates or modifies circuit parameters in the volatile database. The DEFINE CIRCUIT command creates or modifies circuit parameters in the permanent database. Most parameters cannot be SET while the circuit is in the ON state. Exceptions to this are the COST, COUNTER TIMER, HELLO TIMER, MAXIMUM ROUTERS, STATE, and TRANSMIT TIMER parameters. These parameters can be changed while the circuit is in the ON state.

Format

SET circuit-component parameter [...]

DEFINE circuit-component parameter [...]

Circuit Components

CIRCUIT	KNOWN CIRCUITS

Circuit Components

CIRCUIT circuit-id
Identifies the circuit whose parameters are to be updated.

KNOWN CIRCUITS
Indicates that parameters for all known circuits are to be updated.

Command Parameters

ACTIVE BASE base (VAX only)

VAX

Applies only to DDCMP CONTROL circuits. Sets the base priority of an ACTIVE tributary after the tributary has been polled. You can set a separate base for each of the indicated polling states. The base value must be a decimal integer from 0 to 255. The default is 255.

ACTIVE INCREMENT increment (VAX only)
Applies only to DDCMP CONTROL circuits. Sets the increment value added to the ACTIVE tributary priority each time the scheduling timer expires. The *increment* value must be a decimal integer from 0 to 255. The default is 0.

BABBLE TIMER milliseconds (VAX only)
Applies only to DDCMP CONTROL circuits. Represents the number of milliseconds that a selected tributary or remote half-duplex station is allowed to transmit. The *milliseconds* value must be a decimal integer in the range 1 to 65,535. The default is 6000 (6 seconds).♦

CHANNEL number (Remote management only)
Applies only to X.25 and DLM permanent virtual circuits (PVCs). Identifies the logical channel number for the X.25 PVCs. Specify a value in the range 1 to 4095. When you specify an X.25 PVC or a DLM PVC for the first time, this parameter is mandatory.

COST cost
Does not apply to X.25 PVCs. Specifies the routing cost of the circuit. Messages travel between nodes along the path with the smallest total cost. The *cost* value must be a decimal integer in the range 1 to 63. The default is 10.

COUNTER TIMER seconds
Specifies the number of seconds that the circuit counter timer will run. When the counter timer expires, a circuit counter logging event occurs. The *seconds* value must be a decimal integer in the range 0 to 65,535. If no value is set for COUNTER TIMER, the circuit counters are not logged automatically.

VAX

DEAD THRESHOLD number (VAX only)
Applies only to DDCMP CONTROL circuits. Defines the number of times to poll the active, inactive, or dying tributary before changing that tributary's polling state to DEAD because of receive timeouts. The *number* value must be a decimal integer in the range 0 to 255. The default is 8.♦

DTE dte-address (Remote management only)
Applies only to X.25 PVCs and DLM circuits. Identifies the local DTE for the circuit. Specify a decimal integer of 1 to 15 digits. See the *Public Network Information* manual for the format of the address on your network. This parameter is optional for DLM switched virtual circuits (SVCs) and mandatory when you specify an X.25 PVC or DLM PVC for the first time.

VAX

DYING BASE base (VAX only)
Applies only to DDCMP CONTROL circuits. Sets the base priority of a DYING tributary after the tributary has been polled. You can set a separate base for each of the polling state parameters. The *base* value must be a decimal integer from 0 to 255. The default is 0.

DYING INCREMENT increment (VAX only)
Applies only to DDCMP CONTROL circuits. Sets the increment value added to the DYING tributary priority each time the scheduling timer expires. The *increment* value must be a decimal integer from 0 to 255. The default is 16.

DYING THRESHOLD number (VAX only)
Applies only to DDCMP CONTROL circuits. Specifies the number of times to poll the active or inactive tributary before changing that tributary's polling state to dying because of receive timeouts. The *count* value must be a decimal integer in the range 0 to 255. The default is 2.♦

HELLO TIMER seconds
Specifies the frequency of Routing Hello messages sent to adjacent nodes on the circuit. The *seconds* value must be a decimal integer in the range 0 to 8191. The default is 15. The value of the read-only circuit parameter LISTEN TIMER is three times the value of the HELLO TIMER parameter. Does not apply to X.25 PVCs.

VAX

INACTIVE BASE base (VAX only)
Applies only to DDCMP CONTROL circuits. Sets the base priority of an inactive tributary after the tributary has been polled. You can set a separate base for each of the indicated polling states. The *base* value must be a decimal integer from 0 to 255. The default is 0.

INACTIVE INCREMENT increment (VAX only)
Applies only to DDCMP CONTROL circuits. Sets the increment value added to the inactive tributary priority each time the scheduling timer expires. The *increment* value must be a decimal integer from 0 to 255. The default is 64.

INACTIVE THRESHOLD number (VAX only)
Applies only to DDCMP CONTROL circuits. Specifies the number of times to poll the active tributary before changing that tributary's polling state to inactive because of no data response. The *number* value must be a decimal integer in the range 0 to 255. The default is 8.

MAXIMUM BUFFERS number (VAX only)
Applies only to DDCMP CONTROL circuits. Specifies the maximum number of buffers from a common buffer pool that the tributary can use. If you do not set this parameter, there is no common buffer pool and the higher level will explicitly supply the buffers. The *number* value must be a decimal integer up to 254 or the word UNLIMITED.♦

MAXIMUM DATA number (Remote management only)
Applies only to X.25 PVCs and DLM circuits. Specifies the maximum packet size for the X.25 circuit. This value must be at least 5 bytes less than the MAXIMUM BLOCK value that you specify in the SET or DEFINE LINE command and must be a power of 2. Specify a value in the range 16 to 4096 bytes. By default, the *packet size* takes the value specified by the PROFILE parameter of the SET or DEFINE MODULE X25-PROTOCOL command. See the *Public Network Information* manual for the network value of this parameter.

MAXIMUM RECALLS number (Remote management only)
Applies only to DLM outgoing switched virtual circuits (SVCs). Specifies the maximum number of call retries generated on an outgoing DLM circuit before the circuit is declared nonoperational. The range of possible values for MAXIMUM RECALLS is 0 to 255. A *number* value of 0 specifies an infinite number of call retries. The default is 100.

MAXIMUM ROUTERS number
Applies only to Ethernet and FDDI circuits. Specifies the maximum number of routers (including the executor node) allowed by the Routing layer on this circuit. Use a number in the range 1 to 33. The default is 33.

VAX

MAXIMUM TRANSMITS number (VAX only)
Applies only to DDCMP CONTROL circuits. Specifies the maximum number of data messages that can be transmitted at one time. The *number* value must be a decimal integer in the range 1 to 255. The default is 4.♦

MAXIMUM WINDOW number (Remote management only)
Applies only to X.25 PVCs and DLM circuits. Specifies the maximum window size allowed for the X.25 circuit—that is, the maximum number of packets for which outstanding acknowledgments are allowed. Specify a value in the range 1 to 127. By default, *window size* takes the value specified by the PROFILE parameter of the SET or DEFINE MODULE X25-PROTOCOL command. See the *Public Network Information* manual for the network value of this parameter.

NETWORK network-name (Remote management only)
Applies only to X.25 PVCs and DLM circuits. Specifies the network that the circuit runs through. If only one network is set up, this parameter is not required. Otherwise, the parameter is mandatory.

NUMBER dte-address (Remote management only)
Applies only to DLM switched virtual circuits. For outgoing circuits, specifies the remote DTE address and subaddress that this DLM circuit will call. For incoming circuits, specifies the address of the remote DTE from which this circuit

will accept calls. If the NUMBER parameter is not specified for an incoming circuit, a call from any remote DTE can be directed to this circuit. Specify a decimal integer of 1 to 15 digits.

OWNER EXECUTOR (Remote management only)
Applies only to DLM circuits. Identifies the circuit owner as the executor. This parameter is mandatory for a DLM circuit.

VAX

POLLING STATE polling-state (VAX only)
Applies only to DDCMP CONTROL circuits. Identifies the multipoint polling state of the tributary. There are five possible states:

ACTIVE	The tributary is locked in the ACTIVE state.
AUTOMATIC	The tributary's state varies according to the operation of the polling algorithm. This is the default.
DEAD	The tributary is locked in the DEAD state.
DYING	The tributary is locked in the DYING state.
INACTIVE	The tributary is locked in the INACTIVE state.♦

RECALL TIMER seconds (Remote management only)
Applies only to outgoing DLM switched virtual circuits (SVCs). Sets a timer whose expiration causes a remote DTE to be called again during an attempt to set up a DLM circuit. Specify a decimal integer up to a maximum of 65,535.

ROUTER PRIORITY number
Applies only to Ethernet and FDDI circuits. Specifies the priority this router (the executor node on this circuit) is to have in the selection of a designated router for this circuit. Use a value in the range 0 to 127. The default is 64.

SERVICE service-mode
Specifies whether service operations (loading and loop testing) are allowed for the circuit. Does not apply to X.25 circuits.

There are two possible modes:

DISABLED	The circuit cannot be put into service state and cannot perform service functions. The default mode is DISABLED.
ENABLED	The circuit can be put into service state and perform service functions.

STATE circuit-state
Specifies the circuit's operational state. There are three possible states:

OFF	The circuit is not in use.
ON	The circuit is available for normal use or service functions.
SERVICE	The circuit is available for service functions only.

VAX

TRANSMIT TIMER milliseconds (VAX only)
Applies to DDCMP circuits. Defines the number of milliseconds to delay between data message transmits. The *milliseconds* value must be a decimal integer up to 65,535. The default is 0.

TRIBUTARY tributary-address (VAX only)
Applies only to DDCMP CONTROL circuits. Defines the data link physical tributary address of the circuit. The *tributary-address* value must be a decimal integer up to 255.♦

TYPE X25 (Remote management only)
Applies only to X.25 circuits. Specifies the type of circuit.

USAGE usage-type (Remote management only)
Applies only to X.25 circuits. Defines the usage type of an X.25 virtual circuit. There are three possible usage types:

INCOMING	Applies only to DLM switched virtual circuits (SVCs). The circuit is used only for incoming calls.
OUTGOING	Applies only to DLM SVCs. The circuit is used only for outgoing calls.
PERMANENT	Mandatory for X.25 PVCs. The circuit is permanently connected to a remote DTE and does not need to be switched dynamically.

VAX

VERIFICATION option (VAX only)
Applies only to synchronous and asynchronous circuits. Determines whether the remote node needs to send its routing initialization password. There are three options:

DISABLED	Does not require the remote node to send its routing initialization password. This is the default.
ENABLED	Requires the remote node to send its routing initialization password.
INBOUND	Applies to any DDCMP point-to-point circuit. Specifies that the executor node expects to receive a routing initialization password for verification from a remote node before a connection is made between the nodes. This parameter is specified automatically for dynamic asynchronous DDCMP circuits. If you specify the VERIFICATION INBOUND parameter for a circuit, you must specify the INBOUND node parameter (by using the SET/DEFINE NODE command) for the remote node.♦

Comments (VAX only)

VAX

If you plan to run DECnet over a CI, you must first install the driver CNDRIVER. To do this, add the following lines to the site-specific startup procedure in SYS$MANAGER:

```
$ RUN SYS$SYSTEM:SYSGEN
CONNECT CNA0/NOADAPTER
```

Refer to the *DECnet for OpenVMS Networking Manual* for a table of circuit parameters grouped according to circuit type.♦

Examples

VAX

1. `NCP>SET CIRCUIT DMB-0 COST 4 STATE ON`

   This command sets the STATE to ON and the circuit cost to 4 for circuit DMB–0.♦

2. `NCP>SET CIRCUIT MNA-0 STATE ON MAXIMUM ROUTERS 5`

   This command sets Ethernet circuit MNA–0 to ON and sets the maximum number of routers permitted on the circuit to 5.

VAX

3. `NCP>SET CIRCUIT DMB-* COST 5`

   This command sets the circuit cost to 5 for all DMB circuits.♦

SET EXECUTOR ALL

The SET EXECUTOR ALL command updates the volatile copy of the local node's database with all the local node parameters stored for that node in its permanent database.

Format

SET EXECUTOR ALL

Parameters

None.

Examples

1. `NCP>SET EXECUTOR ALL`

   This command loads all permanent database entries stored for the local node into the volatile database.

SET/DEFINE EXECUTOR

The SET EXECUTOR command creates or modifies parameters in the volatile database. The DEFINE EXECUTOR command creates or modifies parameters in the permanent database.

After the executor node's state is set to ON, you cannot change its ADDRESS, ALIAS NODE, BUFFER SIZE, MAXIMUM BROADCAST ROUTERS, MAXIMUM CIRCUITS, NAME, SEGMENT BUFFER SIZE, or TYPE parameter.

Format

SET EXECUTOR parameter [...]

DEFINE EXECUTOR parameter [...]

Command Parameters

ADDRESS node-address

Establishes a node address for the local node, in the following format:

area-number.node-number

where:

area-number	Is in the range 1 to 63.
node-number	Is in the range 1 to 1023.

If you do not specify *area-number*, the default area is 1. You need not supply the area number in the *node-address* if your node is in area 1. When you configure the local node, this parameter is required.

ALIAS INCOMING option

Specifies whether the local node accepts incoming connect requests directed to the alias node identifier specified for the local node. The alias node identifier is described under the ALIAS NODE parameter. There are two options for ALIAS INCOMING:

DISABLED	Specifies that the local node will not accept incoming connect requests directed to the alias node identifier.
ENABLED	Specifies that the local node will accept incoming connect requests directed to the alias node identifier. This is the default if an alias node identifier has been specified.

ALIAS MAXIMUM LINKS number

Specifies the maximum number of logical links for the local node that can use the alias node identifier. The alias node identifier is described under the ALIAS NODE parameter. The maximum value for ALIAS MAXIMUM LINKS is 200. The default is 32.

ALIAS NODE node-id

Establishes a cluster alias node identifier for use by the local node. The *node-id* is a DECnet node identifier that can be either a node name or a node address. This alias permits the local node to be associated with a cluster node identifier common to some or all nodes in the cluster, in addition to its own unique *node-id*.

If you do not specify this parameter, the local node is not associated with a cluster alias node identifier. If a node name is to be used as the alias *node-id*, the node name must previously have been defined in the database.

VAX

AREA MAXIMUM COST number (VAX only)
Applies only to an executor node whose type is AREA. Specifies the maximum total path cost allowed from the executor to any other level 2 routing node (area router). You can specify a decimal value in the range 1 to 1022. The default is 1022.

AREA MAXIMUM HOPS number (VAX only)
Applies only to an executor node whose type is AREA. Specifies the maximum number of routing hops allowable from the executor to any other level 2 routing node. You can specify a decimal value in the range 1 to 30. The default is 30.♦

BROADCAST ROUTING TIMER seconds
Specifies the maximum amount of time allowed between routing updates on Ethernet or FDDI circuits. When the timer expires before a routing update occurs, a routing update is forced. The routing update produces a routing configuration message for each adjacent node. You can specify a number in the range 1 to 65,535. The default is 180.

BUFFER SIZE number
Specifies the size of the receive buffers in bytes, thereby controlling the maximum size of NSP message segments that can be received and forwarded. (The size includes protocol overhead down to and including the End Communication layer, but does not include the Data Link layer overhead.)

The value of the BUFFER SIZE parameter must be greater than or equal to the value of the SEGMENT BUFFER SIZE parameter. The two values are normally equal but may differ to permit the network manager to alter the buffer size on all nodes without interuption of service. Refer to the *DECnet for OpenVMS Networking Manual* for information about how to modify buffer sizes. This buffer size applies to all circuits known to the executor. The default is equal to the value of the SEGMENT BUFFER SIZE, if specified; otherwise it is 576.

COUNTER TIMER seconds
Specifies a timer whose expiration causes a node counter logging event.

DEFAULT ACCESS option
Assigns the default access to all nodes that do not have a specific node ACCESS entry in the volatile database. There are four options:

BOTH	Allows incoming and outgoing logical link connections. This is the default.
INCOMING	Allows logical link connections from the remote node.
NONE	Does not allow incoming or outgoing logical link connections to this node.
OUTGOING	Allows the local node to initiate connections to the remote node, but does not allow connections from the remote node.

If you have OPER privilege on the local system, you can override the default access restriction specified by this parameter.

DELAY FACTOR number
Specifies the number by which to multiply one-sixteenth of the estimated round trip delay to a node to set the retransmission timer to that node. Use a number ranging from 16 to 255. The default is 80.

DELAY WEIGHT number
Specifies the weight to apply to a new round-trip delay data point when updating the estimated round-trip delay to a node. The range is from 1 to 255. The default is 5.

VAX

DNS INTERFACE option (VAX only)
Specifies whether the node will obtain node and address information from the DECdns namespace to update the volatile node database. For information about using an existing DECdns namespace for node-name-to-address mapping, see *DECnet for OpenVMS Networking Manual*.

ENABLED	Enables the use of the DECdns namespace, which must already exist, for updating the volatile node database.
DISABLED	Specifies that the node will not use the DECdns namespace to search for node name and address information. Only the local node database is searched for this information. This is the default.

DNS NAMESPACE string (VAX only)
Set this parameter if you want to use the DNS INTERFACE. Specifies an existing DECdns namespace. If DNS INTERFACE is enabled and you omit the DNS NAMESPACE parameter, DECnet for OpenVMS uses the default DECdns namespace for node name and address lookups. The string value must be 1 to 256 alphanumeric characters and can include the dollar sign ($), underscore (_), and hyphen (-) characters. ♦

IDENTIFICATION id-string
Specifies a text string that describes the executor node (for example, ''VMS Host System''). The string can be a maximum of 32 characters. If it contains blanks or tabs, you must enclose the string in quotation marks. If you do not set this parameter, the default string includes the name and version of the product.

VAX

IDP string (VAX only)
Set this parameter if you want to use the DNS INTERFACE. Specifies the IDP of the network's network service access point (NSAP) (global network address). The string value must be 1 to 22 hexadecimal digits. The default is 49.♦

INACTIVITY TIMER seconds
Specifies the maximum duration of inactivity (no data in either direction) on a logical link before the node checks to see if the logical link still works. The range is between 1 and 65,535. The default is 60.

INCOMING PROXY option
Indicates whether proxy login requests present on incoming logical links are to be honored. There are two options for INCOMING PROXY:

DISABLED	Ignores all incoming proxy requests. Instead, validates the logical link by using the access control information in the connect request or the access control information associated with the object.
ENABLED	Invokes the appropriate proxy, based on the source user, source node, and supplied access control information (if any). This is the default.

Note that proxy access characteristics established in the object database take preference over the proxy access characteristics established in the executor database.

INCOMING TIMER seconds
Specifies the maximum amount of elapsed time between the time a connection is received for a process and the time that process accepts or rejects the connection. For very busy systems, use a value in the range of 45 to 60 seconds. Otherwise use a value of 30 seconds. The default is 45.

MAXIMUM ADDRESS number
Defines the highest node address and, consequently, the greatest number of nodes that can be addressed by the local node. The default is 1023.

If you receive an ''invalid value'' error, you probably have not allocated enough nonpaged pool space. You must lower the MAXIMUM ADDRESS value or increase the NPAGEDYN value. (Use the System Generation utility (SYSGEN) to change the NPAGEDYN parameter.)

VAX

MAXIMUM AREA number (VAX only)
Applies only to an executor node whose type is AREA. Specifies the largest area number and, therefore, the greatest number of areas that can be known about by the executor node's Routing layer. You can specify a decimal value up to 63. The default is 63.♦

MAXIMUM BROADCAST NONROUTERS number
Specifies the maximum total number of nonrouting nodes (end nodes) the executor node can have on its broadcast circuits. Use a number from 1 to 1023. The default is 64.

MAXIMUM BROADCAST ROUTERS number
Specifies the maximum total number of routers the executor node can have on its broadcast circuits. The value of this parameter should be obtained by multiplying 32 by the value of the executor MAXIMUM CIRCUITS parameter. The default is 32.

MAXIMUM BUFFERS number
Specifies the maximum number of buffers in the transmit buffer pool. DECnet normally allocates only what it needs. At minimum, use a value that is 15 times the square root of the number of lines. Increase this value if you experience congestion loss. The default is 100.

MAXIMUM CIRCUITS number
Defines the maximum number of circuits that the local node can use. The number must be in the range 1 to 127. The default is 16.

MAXIMUM COST number
Specifies the maximum total path cost allowed from the local node to any node. The path cost is the sum of the circuit costs along a path between two nodes. Use as small a number as possible in the range of 1 to 1022. The default is 1022.

MAXIMUM DECLARED OBJECTS number
Specifies the maximum number of objects that all processes on a system can declare. To determine the current number of declared objects on your system, use the NCP SHOW KNOWN OBJECTS command. Each of the objects with a PID displayed is one declared object. A single process can declare more than one object. The default is 31, which is sufficient for most configurations. The valid range of objects is 8 to 16383. Dynamically setting the number lower than the number of declared objects currently in use has no effect on those objects.

MAXIMUM HOPS number
Specifies the maximum routing hops from the local node to any other reachable node. A hop is the logical distance over a circuit between two adjacent nodes. Use as small a number as possible in the range of 1 to 30, and be sure that this value is less than or equal to the MAXIMUM VISITS parameter. The default is 30.

MAXIMUM LINKS number
Specifies the maximum number of logical links for the local node. The maximum value for MAXIMUM LINKS is 3885. The default is 32.

VAX

MAXIMUM PATH SPLITS number (VAX only)
Indicates the maximum number of equal cost paths to a given destination node among which the packet load may be split. The range is from 1 to 31. The default is 1.♦

MAXIMUM VISITS number
Specifies the maximum number of nodes a message can visit before it is received by the destination node. Use a number in the range of the value of the MAXIMUM HOPS parameter to 63. You should specify a number that is twice the MAXIMUM HOPS value. The default is 63.

NAME node-name
Specifies the node name to be associated with the executor node. You can assign only one name to a node address.

NONPRIVILEGED item
Specifies nonprivileged inbound access control information for the node. Associate any of the following parameters with the NONPRIVILEGED parameter:

ACCOUNT account	Identifies the account for the default nonprivileged DECnet account on the executor node.
PASSWORD password	Identifies the password for the default nonprivileged DECnet account on the executor node.
USER user-id	Identifies the user name for the default nonprivileged DECnet account on the executor node.

OUTGOING PROXY option

Indicates whether proxy login may be used on outgoing connect requests. There are two options for OUTGOING PROXY.

DISABLED	Specifies that proxy login is not requested on any outgoing logical links.
ENABLED	Specifies that proxy login is requested on outgoing logical links. This is the default.

Note that proxy access characteristics established in the object database take preference over the proxy access characteristics established in the executor database.

OUTGOING TIMER seconds

Specifies the timeout value for the elapsed time between the moment a connection is requested and the moment that connection is acknowledged by the destination node. A value in the range of 30 to 60 seconds is recommended. The default is 60.

VAX

PATH SPLIT POLICY policy (VAX only)

Specifies the policy for equal cost path splitting of network traffic. There are two values for PATH SPLIT POLICY:

INTERIM	Specifies that traffic will be split over all equal-cost paths while forcing packets for individual network sessions to follow the same paths in order to guarantee that packets will be received by the destination node in the correct order. The INTERIM value should be set if some of the nodes in the network do not support out-of-order packet caching. (DECnet–VAX Version 4.5 and earlier versions do not support out-of-order packet caching.)
NORMAL	Specifies that all traffic will be split equally over all equal-cost paths to a destination node. All destination nodes must support out-of-order packet caching (supported by DECnet–VAX Version 4.6 or later and by DECnet for OpenVMS); otherwise, network performance may suffer. NORMAL is the default.♦

PIPELINE QUOTA quota

Specifies the maximum number of bytes of nonpaged pool that DECnet will use for transmission over logical links. Use this parameter for buffering at the NSP level. The default is 10000 bytes. For satellite communications, a value of 6000 or greater is recommended.

PRIVILEGED item

Specifies privileged inbound access control information for the node. Associate any of the following parameters with the PRIVILEGED parameter:

ACCOUNT account	Identifies the account for the default privileged DECnet account on the executor node.
PASSWORD password	Identifies the password for the default privileged DECnet account on the executor node.
USER user-id	Identifies the user name for the default privileged DECnet account on the executor node.

These parameters are not needed unless the PRIVILEGES parameter is used explicitly in any node's object database.

RETRANSMIT FACTOR number

Defines the maximum number of times any given message (except a connect initiate message) will be retransmitted before the logical link is disconnected. The range is 1 to 65,535. The default is 10.

VAX

ROUTING TIMER seconds (VAX only)

Specifies the maximum amount of elapsed time before a routing update is forced on non-broadcast circuits. The routing update produces a routing configuration message for each adjacent node. You can use a number up to a maximum of 65,535. The default is 600.♦

SEGMENT BUFFER SIZE number

Specifies the maximum size of transmit buffers in bytes, thereby controlling the maximum size NSP message segment that can be transmitted. (This value is the maximum size message the End Communications layer can transmit; it does not include Data Link layer overhead.) The default is equal to the value of BUFFER SIZE, if specified; otherwise, it is 576.

The SEGMENT BUFFER SIZE must always be less than or equal to the BUFFER SIZE. The two values are normally equal but may differ to permit the network manager to alter buffer sizes on all nodes without interruption of service. Refer to the *DECnet for OpenVMS Networking Manual* for information about how to modify buffer sizes.

STATE node-state

Specifies the operational state of the executor node. There are four possible states:

OFF	Allows no new logical links, terminates existing links, and stops route-through traffic.
ON	Allows logical links.
RESTRICTED	Allows no new inbound links from other nodes.
SHUT	Allows no new logical links, does not destroy existing links, and goes to the OFF state when all logical links are disconnected.

If you have OPER privilege, you can override the state value specified in this parameter.

SUBADDRESSES range (Remote management only)

Applies only to DLM circuits. Specifies a range of local DTE subaddresses that the Routing layer accepts as X.25 DLM calls. VAX P.S.I. routes all incoming X.25

calls within the specified subaddress range to the Routing layer to be handled as DLM calls.

TYPE node-type
Indicates the type of the executor node. There are five possible node types:

Nonrouting III

Nonrouting IV

Routing IV

†Routing III

†Area

†VAX specific

A ROUTING IV node has full routing capability. A NONROUTING IV node (or end node) can deliver packets to or receive them from any node over one circuit, but cannot route packets from other source nodes through to destination nodes.

AXP

Note

DECnet for OpenVMS AXP supports level 1 routing on only one circuit and does not support level 2 (area) routing.♦

An AREA node is a level 2 router that can route packets within its own area and between areas.

Comments

You can use the SET and DEFINE EXECUTOR commands to create or modify executor node parameters, including certain parameters common to both the executor and remote nodes. Refer to the *DECnet for OpenVMS Networking Manual* for a table of node parameters that indicates those used for the executor node only and those common to the executor and remote nodes.

Examples

1.
```
NCP>SET EXECUTOR ADDRESS 2.11
```

   This command sets the executor node's address to 2.11.

2.
```
NCP>SET NODE 2.13 NAME BIRDY
.
.
.
NCP>SET EXECUTOR ALIAS NODE BIRDY
```

   The SET NODE command establishes a node address 2.13 with the associated node name BIRDY. The SET EXECUTOR ALIAS NODE command then establishes the node name BIRDY as the alias node identifier.

3.
```
NCP>SET EXECUTOR ALIAS INCOMING ENABLED
```

   This command specifies that incoming connect requests directed to the alias node identifier are permitted.

4.
```
NCP>SET EXECUTOR ADDRESS 11
.
.
.
NCP>SHOW EXECUTOR SUMMARY
Node Volatile Summary as of 15-JUN-1992 15:40:20
Executor node = 1.11 (BOSTON)
State                     = on
Identification            = DECnet for OpenVMS
```

   This command sets the executor node's address to 1.11. As the display indicates, the system default value is 1 if the *area-number* is not specified.

5.
```
NCP>SET EXECUTOR STATE ON
```

   This command turns on the executor node.

6.
```
NCP>SET EXECUTOR NAME ROBIN
```

   This command sets the executor node's name to ROBIN.

SET EXECUTOR NODE

The SET EXECUTOR NODE command sets the default executor until you explicitly reset it. The executor is the node on which the network management listener (NML) runs to perform NCP commands. (Do not use the TELL prefix with the SET EXECUTOR NODE command.)

Format

SET EXECUTOR NODE parameter [...]

Command Parameter

NODE node-spec

Specifies a node name or address optionally followed by access control information as specified for OpenVMS operating systems. Use one of the following formats:

node-id
node-id"user password account"

The *node-spec* may be a logical name. You can override access control in a logical name by specifying access control information explicitly in the command.

Associate any or all of the following optional access control parameters with this parameter:

ACCOUNT account	Specifies the user's account for access control verification at the designated executor node.
PASSWORD password	Specifies the user's password for access control verification at the designated executor node.
USER user-id	Specifies the user's username for access control verification at the designated executor node.

Description

For this command, you must enter the command verb SET EXECUTOR followed by the NODE *node-spec* parameter. You can then enter the optional access control parameters in any order.

If the SET EXECUTOR NODE command fails, you must explicitly use the CLEAR EXECUTOR NODE or the SET EXECUTOR NODE command with a different node name or address.

Examples

1.
```
NCP>SET EXECUTOR NODE 5.14
.
.
.
NCP>CLEAR EXECUTOR NODE
```

   The first command sets the executor to node 5.14. The second resets the executor to the local node.

2.
```
NCP>SET EXECUTOR NODE 14"GRAY MARY"
```

   This command uses access control information to set the executor node to node 1.14. This example assumes a default area number of 1 for the executor node.

3.
```
NCP>SET EXECUTOR NODE TRNTO USER GRAY PASSWORD MARY
```

   This command uses an alternate access control format to set the executor to node TRNTO.

4.
```
$ DEFINE NODE$MY   TRNTO"""GRAY MARY"""
$ RUN SYS$SYSTEM:NCP
NCP>SET EXECUTOR NODE NODE$MY
```

   The first command defines a logical name (NODE$MY) for the given *node-spec*. When running NCP, you can use this logical name with the SET EXECUTOR NODE command. Note that you must use three sets of quotation marks to delimit access control information (if supplied) in the *node-spec*.

SET KNOWN PROXIES ALL

The SET KNOWN PROXIES ALL command clears the contents of the volatile proxy database and rebuilds the volatile proxy database from the contents of the permanent proxy database.

Format

SET proxy-component ALL

Proxy Component
KNOWN PROXIES

Proxy Component

KNOWN PROXIES
Indicates that all proxy values are to be updated.

Examples

1. `NCP>SET KNOWN PROXIES ALL`

   This command rebuilds the volatile database of proxy values from the permanent database.

SET LINE ALL

The SET LINE ALL command updates the volatile database with all the line parameters stored for a particular line in the permanent database.

Format

SET line-component ALL

Line Components

LINE	KNOWN LINES

Line Components

LINE line-id
Identifies the line whose parameters are to be updated.

KNOWN LINES
Indicates that all parameters for all known lines are to be updated.

Examples

1. `NCP>SET KNOWN LINES ALL`

   This command loads all permanent database entries stored for all known lines into the volatile database.

SET/DEFINE LINE

The SET LINE command creates or modifies line parameters in the volatile database. The DEFINE LINE command creates or modifies line parameters in the permanent database.

Most parameters canont be SET when the line is in the ON state except for the COUNTER TIMER, SERVICE TIMER, and STATE parameters. These parameters can be changed when the line is ON.

Format

SET line-component parameter [...]

DEFINE line-component parameter [...]

Line Components

LINE	KNOWN LINES

Line Components

LINE line-id
Identifies the line for which specified parameters are to be created or modified in the database.

KNOWN LINES
Indicates that the specified parameters for all known lines are to be created or modified in the database.

Command Parameters

BUFFER SIZE number
Specifies in bytes the size of receive buffers for the specified line, thereby controlling the maximum size of NSP message segments that can be received from or forwarded to an adjacent node that has accepted the line buffer size. Use a value up to a maximum of 65,535 for non-broadcast lines. On Ethernet and FDDI lines, BUFFER SIZE should not be raised to a value higher than 1498 bytes. For Ethernet and FDDI lines, the value of 1498 bytes is provided as the default. For all other types of line, the default is the executor BUFFER SIZE value (as specified in the SET EXECUTOR command). This parameter does not apply to X.25 lines.

You can use BUFFER SIZE to increase the size of NSP messages for logical links over this line.

VAX

CLOCK clock-mode (VAX only)
Applies only to synchronous DDCMP lines. Specifies the hardware clock mode for the line. There are two values for *clock-mode*:

EXTERNAL For normal clock operating mode. The clock signal is supplied externally to the controller.

INTERNAL For use of the clock in test mode. Setting this value causes the line device to supply a clock signal that will allow all transmitted messages to be looped back from outside the device. Note that, in order to use this parameter, the operator may have to connect a loopback plug in place of the normal line.♦

CONTROLLER mode

Specifies the controller mode for the line. There are two possible modes:

LOOPBACK Internal device loopback mode. Note that not all devices support this mode.

NORMAL Normal operating mode, which is the default

COUNTER TIMER seconds

Specifies a timer whose expiration causes a line counter logging event. Specify a decimal integer up to 65,535.

VAX

DEAD TIMER milliseconds (VAX only)

Applies only to DDCMP CONTROL lines. Specifies the number of milliseconds between polls of one of the sets of dead tributaries. The *milliseconds* value must be a decimal integer up to 65,535. The default is 10,000 (10 seconds).

DELAY TIMER milliseconds (VAX only)

Applies only to DDCMP CONTROL lines. Defines the minimum number of milliseconds to delay between polls. This timer limits the effect of a very fast control station on slower tributaries. The *milliseconds* value must be a decimal integer up to 65,535. If you do not set this parameter, there is no delay.

DUPLEX mode (VAX only)

Applies to DDCMP lines. Specifies the hardware duplex mode of the line. There are two possible modes:

FULL Full-duplex (default)

HALF Half-duplex ♦

ECHO DATA value

Applies only to FDDI lines. Specifies the value, in hexadecimal, used as data in an echo request frame. The size of the frame is determined by the ECHO LENGTH parameter. The frame is sent to the address specified by the ECHO TARGET parameter.

The value must be a string of exactly two hexadecimal digits. The default is 55. ECHO DATA can be set in the volatile database, but it cannot be defined in the permanent database.

ECHO LENGTH number

Applies only to FDDI lines. Specifies the number of bytes used in an echo request frame. The value of the bytes is specified by the ECHO DATA parameter. The frame is sent to the address specified by the ECHO TARGET parameter.

The number must be a decimal value from 0 to 4478. The default is 1. ECHO LENGTH can be set in the volatile database, but it cannot be defined in the permanent database.

ECHO TARGET p-address

Applies only to FDDI lines. Specifies the address to which an echo request frame will be sent. The default echo target is 00-00-00-00-00-00. The ECHO TARGET can be set in the volatile database, but it cannot be defined in the permanent database.

VAX

HANGUP option (VAX only)

Applies only to asynchronous DDCMP lines. Indicates whether the modem signals are dropped when the line is shut down. There are two possible options:

DISABLED Indicates that modem signals should not be dropped when the line is shut down. This is the default for static asynchronous DDCMP lines.

ENABLED Indicates that modem signals should be dropped when the line is shut down.

This parameter is supplied automatically for dynamic asynchronous DDCMP lines. The default is HANGUP ENABLED if the /HANGUP qualifier was specified for the DCL command SET TERMINAL, and HANGUP DISABLED if /NOHANGUP was specified.♦

HOLDBACK TIMER milliseconds (Remote management only)

Applies only to X.25 lines. Specifies the maximum delay before the X.25 software acknowledges successful receipt of data. If you do not specify this parameter, then the acknowledgment is sent immediately. If you specify a holdback timer, performance may be improved by allowing the acknowledgment to be included with a subsequent data message transmitted. This parameter cannot be set to a value larger than one half the value of the RETRANSMIT TIMER parameter.

INTERFACE interface-code (Remote management only)

Applies only to X.25 lines using the ISO8208 profile. Specifies the way in which the X.25 level 2 software acts when it uses the line. There are two values for the *interface-code*:

DCE Specifies that the software operates as a DCE.

DTE Specifies that the software operates as a DTE.

The default is DTE.

VAX

LINE SPEED number (VAX only)

Applies only to asynchronous DDCMP lines. Specifies the speed of the line in baud. This parameter must be set to the same value on both sides of an asynchronous DDCMP connection. It is specified automatically for dynamic asynchronous DDCMP lines. If not specified, the value of this parameter is equal to the current speed of the line.♦

MAXIMUM BLOCK number (Remote management only)
Applies only to X.25 lines. Specifies the maximum size in bytes of the frame for the line. This value must be at least 5 bytes larger than the maximum packet size you specify in the SET CIRCUIT (for PVCs) or SET MODULE X25-PROTOCOL (for SVCs) command. Specify a value in the range 21 to 4103. By default, the size of the frame takes the value specified by the PROFILE parameter of SET MODULE X25-PROTOCOL commands. See the *Public Network Information* manual for the network value of this parameter.

MAXIMUM RETRANSMITS number (Remote management only)
Applies only to X.25 lines. Specifies the maximum number of retransmissions of a frame on the specified line. Specify a value up to 255. By default, the retransmission value takes the value specified by the PROFILE parameter of the SET MODULE X25-PROTOCOL command. See the *Public Network Information* manual for the network value of this parameter.

MAXIMUM WINDOW number (Remote management only)
Applies only to X.25 lines. Specifies the maximum number of frames for which outstanding acknowledgments are allowed. Specify a value up to 127. By default, the outstanding frame value takes the value specified by the PROFILE parameter of the SET MODULE X25-PROTOCOL command. See the *Public Network Information* manual for the network value of this parameter.

MICROCODE DUMP filespec (Remote management only)
Applies only to X.25 lines. Dumps the microcode of the specified KMS11, KMS1P, or KMV1A device to the indicated file. By default, the output file takes the following format:

SYS$ERRORLOG:fileid.DMP

The KMS/KMV Dump Analyzer (PSIKDA) is the utility program used to process the dump file created by the MICROCODE DUMP parameter. Refer to the *VAX P.S.I. Problem Solving Guide* for more information about PSIKDA.

NETWORK network-name (Remote management only)
Applies only to X.25 lines. Specifies the network to which the line is connected. If only one network is set up, that network is the default. Otherwise, the NETWORK parameter is mandatory.

NIF TARGET p-address
Applies only to FDDI lines. Specifies the address to which the next Neighborhood Information Frame (NIF) request frame will be sent. The default NIF target is 00-00-00-00-00-00. You can set the NIF TARGET parameter in the volatile database, but you cannot define it in the permanent database.

PROTOCOL protocol-name

Defines the Data Link protocol to be used on this line. The following values can be used for *protocol-name*:

†DDCMP CONTROL	Specifies this line as a multipoint control station. You can specify multiple circuits for CONTROL lines, but each circuit must have a unique physical tributary address.
†DDCMP DMC	Specifies that this line is in DMC emulator mode. DMC is similar to POINT, except that DMC uses an older version of DDCMP (Version 3.2). This protocol should be set for the local line when the remote line is a DMC. Note that this protocol is valid only when a DMP11 or DMV11 is being used.
†DDCMP POINT	Defines this line as one end of a point-to-point DDCMP connection. You can specify only one circuit per POINT line.
†DDCMP TRIBUTARY	Specifies that this line is a tributary end of a DDCMP multipoint group. You may specify only one circuit per TRIBUTARY line.
ETHERNET	Specifies that this line uses the Ethernet protocol.
FDDI	Specifies that this line uses the FDDI protocol.
‡LAPB	Specifies that the line uses the X.25 level 2 protocol. The line must be a line for the X25-PROTOCOL module.
‡LAPBE	Specifies that the line uses the X.25 level 2 protocol with extended sequencing. The line must be used by the X25-PROTOCOL module.

†VAX specific

‡Remote management only

Default line protocols are based on line names. See Appendix A for a list of the DECnet for OpenVMS circuit and line devices.

RECEIVE BUFFERS number

Specifies the length of the line's receive queue. Use a value in the range 1 to 32.

REQUESTED TRT microseconds

Applies only to FDDI lines. Specifies the value for the token rotation timer in microseconds. Microseconds must be a decimal integer in the range of 4000 to 167772. The default is 8000 microseconds.

RESTRICTED TOKEN TIMEOUT milliseconds

Applies only to FDDI lines. Specifies the limit on how long a single restricted mode dialog may last before being terminated. Milliseconds must be a decimal integer in the range of 0 to 10000. The default is 1000 milliseconds.

RETRANSMIT TIMER milliseconds

VAX

For a DDCMP multipoint line, specifies the maximum amount of time allowed to elapse before a retransmission is necessary. This is the amount of time a control station will wait for a tributary to respond. The number of milliseconds must be a decimal integer up to 65,535. The default value is the network default.♦

For a remotely managed X.25 line, this parameter specifies the time before a frame is retransmitted. By default, the time takes the value specified in the PROFILE parameter of the SET MODULE X25-PROTOCOL command. See the *Public Network Information* manual for the network value of this parameter. The RETRANSMIT TIMER cannot be set to a value smaller than twice the value of the HOLDBACK TIMER parameter (if specified).

RING PURGER ENABLE option

Applies only to FDDI lines. The option values are:

ON	Participate in the Ring Purger election and, if elected, perform the Ring Purger function. This is the default.
OFF	Do not participate in the Ring Purger election.

This parameter is to allow operation when stations on your ring do not conform to the FDDI specification. Except for this case, it should be left ON for improved ring reliability.

VAX

SCHEDULING TIMER milliseconds

Applies only to DDCMP CONTROL lines. Specifies the number of milliseconds between recalculation of tributary polling priorities. The number of milliseconds must be a decimal integer up to a maximum of 65,535. The default is 50.♦

SERVICE TIMER milliseconds

Specifies the maximum amount of time allowed to elapse before a receive request completes while performing service operations on a line. Milliseconds is a decimal integer from 1 to 65,535. The default is 4000 (4 seconds). Does not apply to X.25 lines.

SIF CONFIGURATION TARGET p-address

Applies only to FDDI lines. Specifies the address to which a Status Information Frame (SIF) configuration request frame will be sent upon request. The default SIF configuration target is 00-00-00-00-00-00. SIF CONFIGURATION TARGET can be set in the volatile database, but it cannot be defined in the permanent database.

SIF OPERATION TARGET p-address

Applies only to FDDI lines. Specifies the address to which a Status Information Frame (SIF) operation request frame will be sent upon request. The default SIF operation target is 00-00-00-00-00-00. SIF OPERATION TARGET can be set in the volatile database, but it cannot be defined in the permanent database.

STATE line-state

Specifies the line's operational state. The possible states include the following:

OFF	The line is not in use.
ON	The line is available for normal use or service functions.
SERVICE	The line is available for active service functions.

VAX

STREAM TIMER milliseconds (VAX only)
Applies only to DDCMP CONTROL lines. Specifies the number of milliseconds that a tributary or half-duplex remote station is allowed to hold the line. The number of milliseconds must be a decimal integer up to 65,535. The default is 1000 (1 second).

SWITCH option (VAX only)
Applies only to asynchronous DDCMP lines. Specifies if a line currently being used as a DECnet asynchronous communications line should be switched back to a terminal line. There are two values for *option*:

DISABLED The line is not switched to a terminal line. This is the default for static lines.

ENABLED The line is switched to a terminal line after it is disconnected from the network (when the channel to the network is deassigned). This is the default for dynamic lines.

TRANSMIT PIPELINE number (VAX only)
Applies only to DMR11 lines. Specifies the maximum number of DDCMP messages for which outstanding acknowledgments are allowed. Specify a value in the range 1 to 32. By default, the value for outstanding DDCMP messages is 7. To avoid excessive use of system memory, do not arbitrarily set this value higher than necessary. Refer to the *DECnet for OpenVMS Networking Manual* for the procedure for determining an optimum TRANSMIT PIPELINE value.♦

VALID TRANSMISSION TIME microseconds
Applies only to FDDI lines. Specifies the maximum time between arrivals of a valid frame or unrestricted token. Microseconds must be a decimal integer in the range of 2500 to 5222. The default is 2621 microseconds.

Comments

VAX

Note that no protocol is specified for a CI line. The CI uses its own private protocol for communication between nodes. If you plan to run DECnet for OpenVMS over a CI, you must install the driver CNDRIVER. To do this, add the following lines to the site-specific start up procedure in SYS$MANAGER:

```
$ RUN SYS$SYSTEM:SYSGEN
CONNECT CNA0/NOADAPTER ♦
```

Refer to the *DECnet for OpenVMS Networking Manual* for a table of line parameters grouped according to line type.

Examples

1.
```
NCP>SET LINE SVA-0 STATE ON
```

   This command sets Ethernet line SVA–0 to the ON state.

2.
```
NCP>SET LINE FZA-1 SIF CONFIGURATION TARGET AA-00-04-00-03-10
```

   This command sets the SIF CONFIGURATION TARGET of FDDI line FZA–1 to AA-00-04-00-03-10.

VAX

3.
```
NCP>SET LINE DMC-0 DUPLEX FULL STATE ON
```

   This command sets line DMC–0 to the ON state in full duplex mode.

4.
```
NCP>SET LINE DMC-0 PROTOCOL DDCMP POINT
```

   This command sets the line protocol to DDCMP POINT for line DMC–0.♦

5.
```
NCP>TELL REMNOD SET LINE DUP-1 -
_ RETRANSMIT TIMER 500 MAXIMUM RETRANSMITS 10 -
_ MAXIMUM BLOCK 133 MAXIMUM WINDOW 2 STATE ON
```

   This command sets up a line called DUP–1 on remote node REMNOD that uses the LAPB protocol and has an associated DUP11–DA synchronous line interface with controller number 1. If a frame is not acknowledged within 500 milliseconds, it is retransmitted, and this operation is to be performed a maximum of 10 times. The maximum frame size is 133 bytes, and the maximum number of frames for which outstanding acknowledgments are allowed is 2. The line is operational.

6.
```
NCP>TELL REMNOD SET LINE KMX-0-0 MICROCODE DUMP BARRY
```

   This command dumps the microcode of the KMX to the file SYS$ERRORLOG:BARRY.DMP on node REMNOD.

7.
```
NCP>TELL REMNOD SET LINE KMV-0 NETWORK PSS STATE ON
```

   This command sets up the line KMV-0 on remote node REMNOD to the PSS network.

VAX

8.
```
NCP>SET LINE DMF-* CONTROLLER LOOPBACK
```

   This command sets the controller mode to loopback for all DMF lines.♦

SET LOGGING ALL

The SET LOGGING ALL command updates the volatile copy of the database on the executor node with all the logging parameters stored for the logging component in the permanent database on the executor node.

Format

SET logging-component ALL

Logging Components

KNOWN LOGGING	LOGGING FILE
LOGGING CONSOLE	LOGGING MONITOR

Logging Components

KNOWN LOGGING
Indicates that all parameters for all known logging sinks are to be updated.

LOGGING CONSOLE
Indicates that all parameters for the logging console are to be updated.

LOGGING FILE
Indicates that all parameters for the logging file are to be updated.

LOGGING MONITOR
Indicates that all parameters for the logging monitor are to be updated.

Examples

1.
```
NCP>SET LOGGING CONSOLE ALL
```

   This command loads all permanent database entries stored for the logging console into the volatile database.

SET/DEFINE LOGGING EVENTS

The SET LOGGING EVENTS command creates or modifies event-source logging parameters in the volatile database. The DEFINE LOGGING EVENTS command creates or modifies event-source logging parameters in the permanent database on the local node.

The logging state (in the SET/DEFINE LOGGING STATE command) must be set to ON for all the logging components for which you are defining events. Event-source and event-sink parameters are mutually exclusive; you cannot use parameters from both categories in a single command.

Format

SET logging-component parameter [...]

DEFINE logging-component parameter [...]

Logging Components

KNOWN LOGGING	LOGGING FILE
LOGGING CONSOLE	LOGGING MONITOR

Logging Components

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

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

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

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

Command Parameters

EVENTS event-list
KNOWN EVENTS
Indicates a specific event or all known events to be logged. The EVENTS parameter may be followed by one or more of the following event-source parameters:

CIRCUIT circuit-id	Identifies the circuit from which events are logged.
LINE line-id	Identifies the line from which events are logged.

NODE node-id	Identifies the node from which events are logged.
SINK node	Identifies the node that is to receive events. There are two possibilities:
	EXECUTOR — Identifies the local node (default).
	NODE node-id — Identifies a node in the network.

Comments

For a table of the event-source parameters used with the SET/DEFINE LOGGING EVENTS command, refer to the *DECnet for OpenVMS Networking Manual*. For a description of event-lists, refer to Section 1.4. For a summary of event class and types and information about specific events that operating systems will log, refer to Appendix D.

Examples

1. `NCP>SET LOGGING MONITOR LINE FZA-1 KNOWN EVENTS`

   This command causes all events for line FZA–1 to be logged to OPCOM, the default monitor.

2. `NCP>SET LOGGING CONSOLE KNOWN EVENTS SINK NODE TRNTO`

   This command causes all events generated locally to be logged to the logging console on remote node TRNTO.

3. `NCP>SET LOGGING CONSOLE EVENTS 4.* SINK EXECUTOR`

   This command causes all class 4 events to be logged on the console of the executor node.

4. `NCP>SET LOGGING MONITOR EVENTS 7.*`

   This command causes all X25-related events to be logged to OPCOM, the default monitor.

SET/DEFINE LOGGING STATE

The SET LOGGING STATE command creates or modifies event-sink logging parameters in the volatile database. The DEFINE LOGGING STATE command creates or modifies event-sink logging parameters in the permanent database on the local node. The STATE parameter is required for all logging.

Format

SET logging-component parameter [...]

DEFINE logging-component parameter [...]

Logging Components

KNOWN LOGGING	LOGGING FILE
LOGGING CONSOLE	LOGGING MONITOR

Logging Components

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

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

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

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

Command Parameters

NAME sink-name
Identifies the name of a console, file, or monitor program to which events will be logged. For the monitor sink, all network operator terminals (enabled by the command REPLY/ENABLE=NETWORK) receive formatted events. You can change this parameter only if the sink is in the OFF state.

STATE sink-state
Specifies the operational state of the sink. There are three possible states:

HOLD	The sink is temporarily unavailable and events are queued.
OFF	The sink is unavailable for receiving events; they are discarded.
ON	The sink is available for receiving events.

Comments

Refer to the *DECnet for OpenVMS Networking Manual* for a table of event-sink logging parameters and their functions.

Examples

1. `NCP>SET LOGGING CONSOLE NAME SYS$MANAGER:EVENTS.LOG`

   This command establishes a logging file, EVENTS.LOG, on the local node.

2. `NCP>SET LOGGING MONITOR STATE ON`

   This command turns on logging to OPCOM, the default monitor. All operator terminals that were previously declared as network operator terminals (with REPLY/ENABLE=NETWORK) will now receive all formatted events.

SET/DEFINE MODULE CONFIGURATOR

The SET MODULE CONFIGURATOR command creates or modifies the parameters of the Ethernet configurator module component in the volatile database. The DEFINE MODULE CONFIGURATOR command creates or modifies the parameters of the configurator module component in the permanent database. You must specify a qualifier following the CONFIGURATOR command verb.

The configurator module constructs a list of systems active on the Ethernet circuits specified.

Format

SET MODULE CONFIGURATOR [qualifier] parameter

DEFINE MODULE CONFIGURATOR [qualifier] parameter

Qualifiers

CIRCUIT	KNOWN CIRCUITS

Qualifiers

CIRCUIT circuit-id

Applies only to Ethernet circuits. Specifies that configurator information on the circuit identified by *circuit-id* is to be stored in the database.

KNOWN CIRCUITS

Applies only to Ethernet circuits. Specifies that configurator information on all known Ethernet circuits is to be stored in the database.

Command Parameters

ALL

Copies configurator module information from the permanent database into the volatile database.

SURVEILLANCE control

Indicates whether a list of active systems is to be kept for the Ethernet circuit or circuits specified. There are two control values:

DISABLED	The list is not to be kept. This is the default value.
ENABLED	The list is to be kept.

Examples

1.
```
NCP>SET MODULE CONFIGURATOR KNOWN CIRCUITS -
_ SURVEILLANCE ENABLED
```

   This command specifies that a list of active systems on all known Ethernet circuits is to be kept.

2.
```
NCP>SET MODULE CONFIGURATOR CIRCUIT MNA-0 SURVEILLANCE ENABLED
```

   This command specifies that a list of active systems on Ethernet circuit MNA–0 is to be kept.

3.
```
NCP>SET MODULE CONFIGURATOR KNOWN CIRCUITS ALL
```

   This command loads all permanent database entries for the configurator module into the volatile database.

SET/DEFINE MODULE X25-ACCESS (Remote management only)

The SET MODULE X25-ACCESS and DEFINE MODULE X25-ACCESS commands associate the host node with the specified multihost connector node for the purpose of accessing a specified X.25 network.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

SET MODULE X25-ACCESS qualifier parameter [...]

DEFINE MODULE X25-ACCESS qualifier parameter [...]

Qualifiers

NETWORK	KNOWN NETWORKS

Qualifiers

NETWORK network-name
KNOWN NETWORKS

Identifies the name of a network or all known networks to be accessed by the host node. The network name must be the same network name specified on the multihost node to which the access node connects. After you specify the network qualifier, you must include one or more of the following access control parameters:

ACCOUNT account	Identifies the user account for use when connecting to the multihost connector node. Specify a string from 1 to 39 characters. By default, no account is used.
ALL	Copies the access parameters from the permanent database into the volatile database. Use this parameter only with the SET MODULE X25-ACCESS command.
NODE node-id	Identifies a multihost connector node connected to the PSDN that you want to access. When you specify an X25-ACCESS network for the first time, this parameter is mandatory.

PASSWORD password	Specifies the password for use by the access routines when connecting to the multihost connector node. Specify a string from 1 to 39 characters. The default is that no password is used.
USER user-id	Specifies the user identification for use by the access routines in connecting to the multihost connector node. Specify a string from 1 to 16 characters. The default is that no user identification is used.

Examples

1.
```
NCP>SET MODULE X25-ACCESS NETWORK TELENET1 NODE FRED
NCP>SET MODULE X25-ACCESS NETWORK PSS1 NODE GINGER
```

   These two commands allow users on the host node to access the network called TELENET1 through node FRED, and the network called PSS1 through node GINGER.

2.
```
NCP>SET MODULE X25-ACCESS KNOWN NETWORKS ALL
```

   This command loads all permanent database entries stored for the X25-ACCESS database into the volatile database.

SET/DEFINE MODULE X25-PROTOCOL (Remote management only)

The SET MODULE X25-PROTOCOL command creates or modifies the parameters of the protocol module component in the volatile database. The DEFINE MODULE X25-PROTOCOL command creates or modifies the parameters of the protocol module component in the permanent database.

Use separate SET MODULE X25-PROTOCOL commands to specify a DTE with its associated parameters, a group with its associated parameters, and the network profile. The X25-PROTOCOL module contains records that identify the packet switching data networks (PSDNs) your DTE is connected to, associate your local DTE (and possibly a group name) with VAX P.S.I., and control the transmission of packets.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

SET MODULE X25-PROTOCOL qualifier [...] parameter [...]

DEFINE MODULE X25-PROTOCOL qualifier [...] parameter [...]

Qualifiers

DTE	KNOWN DTES
GROUP	KNOWN GROUPS
NETWORK	KNOWN NETWORKS

Qualifiers

DTE dte-address
KNOWN DTES

Identifies a local DTE or specifies all known DTEs. See the *Public Network Information* manual for the format of the DTE address. If you specify the DTE qualifier, you must associate with it either of the following parameters.

NETWORK network-name	Selects the DTE(s) specified on a particular network.
KNOWN NETWORKS	Selects the DTE(s) on all known networks.

If you have only one network set up, that network is the default for SET commands. For DEFINE commands the NETWORK parameter is mandatory. If you have more than one network set up, the NETWORK parameter is mandatory. In either case, you must follow the DTE qualifier with one or more of the following parameters:

ALL	Copies DTE parameters from the permanent database into the volatile database. Used only with the SET MODULE X25-PROTOCOL command.

CALL TIMER seconds	Specifies a timer that starts when a request to set up an X.25 virtual circuit is transmitted, and whose termination clears the request if no response has been received. Specify a value up to 255. By default, the timer takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.
CHANNELS list	Specifies a list of logical channel numbers (LCNs) reserved for outgoing calls. The *list* value is one or more LCNs. Separate multiple channel numbers with hyphens to indicate ranges and with commas to indicate individual numbers. Specify a value up to 4095 for each number in the list. If the DTE is to make outgoing calls, this parameter is mandatory when you specify a DTE for the first time. The LCN range at the DTE is defined by PSDN authorities at subscription time.
CLEAR TIMER seconds	Specifies a timer that starts when a request to clear an X.25 virtual circuit is transmitted, and whose termination retransmits the clear request if no response has been received. Specify a value up to 255. By default, the timer takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.
COUNTER TIMER seconds	Sets a timer whose expiration causes a DTE counter logging event. Specify a decimal integer up to 65,535.
DEFAULT DATA bytes	Specifies the default packet size for X.25 virtual circuits. This value must be at least 5 bytes less than the MAXIMUM BLOCK value you specify in the SET LINE command and must be a power of 2. Specify a value in the range 16 to 4096 bytes. By default, packet size takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.

SET/DEFINE MODULE X25-PROTOCOL (Remote management only)

DEFAULT WINDOW number

Specifies the default window size for X.25 virtual circuits, that is, the maximum number of packets for which outstanding acknowledgments are allowed. Specify a value up to 127. By default, window size takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.

INTERFACE interface-code

Specifies the way in which the processor operates. This parameter can be used only with the ISO8208 profile. The parameter can take one of the following values:

DTE	The processor operates as a DTE.
DCE	The processor operates as a DCE.
NEGOTIATED	The processor can operate either as a DTE or DCE. The restart protocol determines the appropriate method of operation.

The default is DTE.

INTERRUPT TIMER seconds

This is the interrupt timer (ISO T26 timer). This parameter can be used only with the ISO8208 profile. Specify a value up to 255. If you do not specify a value, there is no time limit on interrupt acknowledgments.

LINE line-id

Identifies the line associated with the DTE. Use the format *dev-c[-u]* to specify a line name. This parameter is mandatory when you specify a DTE for the first time.

MAXIMUM CIRCUITS number

Specifies the maximum number of virtual circuits for the DTE. By default, the number takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.

MAXIMUM CLEARS number

Specifies the maximum number of attempts to clear an X.25 virtual circuit. Specify a value up to 255. By default, the number takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.

MAXIMUM DATA bytes
: Specifies the maximum packet size for X.25 virtual circuits. This value must be at least 5 bytes less than the MAXIMUM BLOCK value you specify in the SET LINE command and must be a power of 2. Specify a value in the range 16 to 4096 bytes. By default, packet size takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.

MAXIMUM RESETS number
: Specifies the maximum number of attempts to reset an X.25 virtual circuit. If this number is exceeded, the virtual circuit will be cleared. Specify a value up to 255. By default, the number takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.

MAXIMUM RESTARTS number
: Specifies the maximum number of attempts to restart an X.25 virtual circuit. Specify a value up to 255. By default, the number takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.

MAXIMUM WINDOW number
: Specifies the window size allowed for X.25 virtual circuits, that is, the maximum number of packets for which outstanding acknowledgments are allowed. Specify a value up to 127. By default, the window size takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.

RESET TIMER seconds
: Specifies a timer that starts when a reset is transmitted, and whose termination retransmits the reset if no response has been received. Specify a value up to a maximum of 255. By default, the timer takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.

RESTART TIMER seconds	Specifies a timer that starts when a restart is transmitted, and whose termination retransmits the restart if no response has been received. Specify a value up to 255. By default, the timer takes the network value specified by the PROFILE parameter. See the *Public Network Information* manual for the network value of this parameter.
STATE dte-state	Specifies the operational state of the DTE. This parameter takes one of the following values: ON — The DTE is available for normal use. OFF — The DTE is not in use. This is the default state. SHUT — The DTE is to be closed down but only when all present activity has ceased. Used only with the SET MODULE X25-PROTOCOL command.

GROUP group-name
KNOWN GROUPS

Identifies a closed user group (CUG) or bilateral closed user group (BCUG) or specifies all known groups. Each group specified should have a unique *group-name*, from 2 to 16 characters long. If you specify the GROUP qualifier, you must follow it with one or more of the following parameters:

ALL	Copies the group parameters from the permanent database into the volatile database. Used only with the SET MODULE X25-PROTOCOL command.
DTE dte-address	Identifies the local DTE associated with the group name. This parameter is mandatory when you are creating a group.
NETWORK network-name(Remote management only)	Identifies a network to which the DTE is connected. This parameter is mandatory when you are creating a group.
NUMBER group-number	Specifies the CUG or BCUG number. You can omit leading zeros. This parameter is mandatory when you are associating a DTE with a group.
TYPE BILATERAL	Specifies the user group as a BCUG. This parameter is mandatory when you are associating a DTE with a BCUG. It is omitted if the user group is a CUG.

NETWORK network-name
KNOWN NETWORKS

Identifies a network or specifies all known networks. If you specify the NETWORK qualifier, you must follow it with one of the following parameters.

ALL	Copies the profile name from the permanent database to the volatile database. Use only with the SET MODULE X25-PROTOCOL command.
PROFILE profile-name	Specifies a profile to be used on the network. The *profile-name* is a character string of up to 32 characters. This parameter is mandatory. Refer to the *Public Network Information* manual for the valid *profile-name*.

Examples

1.
```
NCP>SET MODULE X25-PROTOCOL DTE 123789456 NETWORK TELENET -
_ CHANNELS 20-10,3,9 LINE DUP-0 MAXIMUM CIRCUITS 400 STATE ON
```

   This command creates a record for your local DTE (DTE address 123789456) and associates a line DUP–0 and a set of logical channels with this DTE.

2.
```
NCP>SET MODULE X25-PROTOCOL NETWORK NET1 PROFILE PSDNC
```

   This command specifies that you are going to use PSDNC, and that you are going to connect to this PSDN through the network NET1.

3.
```
NCP>SET MODULE X25-PROTOCOL GROUP ESECUG DTE 123789456 -
_ NETWORK NET1 NUMBER 12
```

   This command specifies that your DTE is a member of closed user group ESECUG with group number 12.

SET/DEFINE MODULE X25-SERVER/X29-SERVER (Remote management only)

The SET MODULE X25-SERVER and SET MODULE X29-SERVER commands create or modify the parameters of the X.25 or X.29 call handler in the volatile database. The DEFINE MODULE X25-SERVER and DEFINE MODULE X29-SERVER commands create or modify the parameters of the X.25 or X.29 call handler in the permanent database.

Use separate SET MODULE X25-SERVER or DEFINE MODULE X29-SERVER commands to specify the module parameters and the destinations. The server modules contain records that identify and specify parameters for a destination, specify the maximum number of circuits that each module (that is, all destinations for a particular module) may have, and specify the state of the module.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

SET MODULE X25-SERVER [qualifier] parameter [...]

SET MODULE X29-SERVER [qualifier] parameter [...]

DEFINE MODULE X25-SERVER [qualifier] parameter [...]

DEFINE MODULE X29-SERVER [qualifier] parameter [...]

Qualifiers

DESTINATION
KNOWN DESTINATIONS

Command Parameters

ALL
COUNTER TIMER
MAXIMUM CIRCUITS
STATE

Qualifiers

DESTINATION dest-name
KNOWN DESTINATIONS

Identifies a destination or specifies all known destinations. If you specify the destination qualifier, you must follow it with one or more of the following parameters:

ACCOUNT account

Identifies the user account used in connecting to a host node for incoming calls to a destination. Specify a string from 1 to 39 characters. By default, no account is used.

ALL	Copies the destination parameters from the permanent database into the volatile database. Use only with the SET MODULE X25-SERVER or the SET MODULE X29-SERVER command.
CALL MASK hex-value	Specifies the mask applied to the incoming call data before it is tested. By default, no value is used.
CALL VALUE hex-value	Specifies the string used to test the incoming call data. By default, no value is used.
CALLED ADDRESS dte-address	For redirected calls, identifies the DTE that was originally called. The parameter value is used to select a destination for a redirected call. By default, no called DTE is used.
EXTENSION MASK hex-string	Provides a mask for the called address extension facility. Using logical AND, the mask is merged with the called address extension in the incoming call and the result is compared with the extension value. By default, no extension mask is used.
EXTENSION VALUE hex-string	Provides a value for the called address extension facility. The value is compared with the address extension in the incoming call (after merging as described for EXTENSION MASK). By default, no extension mask is used.
GROUP group-name	Identifies a closed user group (CUG) or bilateral closed user group (BCUG). The *group-name* is an ID string. By default, no group name is used.
INCOMING ADDRESS	Identifies the address of the DTE receiving the call (as specified in the call packet).
NETWORK network-name	Specifies the network for which calls are accepted. By default, no network name is used.

NODE node-id	Identifies the P.S.I. Access node that is to be the destination of the incoming X.25 call. Specify this parameter if the executor node is a node with VAX P.S.I. software in multihost mode installed, serving as a connector node. This parameter is mandatory when you specify this destination for the first time. Used only with the SET MODULE X25-SERVER command or the DEFINE MODULE X25-SERVER command.
OBJECT object-id	Identifies the object used in connecting to a host node for incoming calls to a destination. The *object-id* is an ID string. If the object name looks like a number, enclose the string in quotation marks. This parameter is mandatory when you specify a destination for the first time.
PASSWORD password	Specifies the password used in connecting to a host node for incoming calls to a destination. Specify a string from 1 to 39 characters. The default is that no password is used.
PRIORITY priority	Specifies the priority of the destination record and is used to select one of a set of destinations for which the incoming call may be valid. Specify a value up to 255, where 255 is the highest priority. By default, *priority* takes a value of 0.
RECEIVING DTE dte-address	Identifies the local DTE. It is used to select a destination for calls received on the local DTE. By default, no receiving DTE is used.
REDIRECT REASON reason-code	Identifies a reason for redirecting an incoming call. The *reason-code* can take one of the following values: BUSY — The original DTE was busy and could not accept any more calls. OUT OF ORDER — The original DTE was out of order. SYSTEMATIC — All calls to the original DTE are automatically rerouted. The *reason-code* is used to select a destination for a redirected call. By default, no *reason-code* is used.

SENDING ADDRESS dte-address	Identifies the address of the remote DTE that originated a call. By default, no sending DTE is used.
SUBADDRESSES range	Specifies the range of local DTE subaddresses that identify the destination for the incoming call. The *range* consists of one or two subaddresses. The subaddress is a decimal integer up to 9999. See the *Public Network Information* manual for the maximum length of the subaddress on your network. Separate two subaddresses with a single hyphen to indicate a range. The second subaddress must be greater than the first. By default, no subaddress range is used.
USER user-id	Specifies the user identification for use by incoming calls to the destination. Specify a string from 1 to 16 characters. The default is that no user identification is used.

Command Parameters

ALL

Copies the call handler parameters from the permanent database into the volatile database. Used only for the SET MODULE X25-SERVER command or the SET MODULE X29-SERVER command.

COUNTER TIMER seconds

Sets a timer whose expiration causes a server module counter logging event. Specify a decimal integer up to 65,535.

MAXIMUM CIRCUITS number

Specifies the maximum number of circuits that the module (that is, all destinations) may have. By default, the maximum is 512.

STATE module-state

Specifies the operational state of the server module. This parameter takes one of the following values:

OFF	The module is not in use. For the X.29 server module, OFF is the default state.
ON	The module is available for normal use. For the X.25 server module, ON is the default state.
SHUT	The module is to be closed down but only when all present activity has ceased. The SHUT value applies only to the SET MODULE X25-SERVER or X29-SERVER command.

Examples

1.
```
NCP>SET MODULE X25-SERVER DESTINATION JOE SUBADDRESSES 12-24 -
_ SENDING ADDRESS 987321654 PRIORITY 3 OBJECT OBJONE
```

   This command creates a destination called JOE that will handle incoming X.25 calls from a DTE with an address of 987321654 and a subaddress in the range 12 to 24 specified in the DECnet object database.

2.
```
NCP>SET MODULE X25-SERVER DESTINATION DEFDES -
_ OBJECT LAST PRIORITY 0
```

   This command modifies the destination called DEFDES that will handle any incoming calls. The destination has the lowest priority and will only handle calls that fail to find another destination.

3.
```
NCP>SET MODULE X25-SERVER DESTINATION THRUSH SUBADDRESSES 11-20 -
_ OBJECT 36 NODE THRUSH
```

   This command creates a destination called THRUSH for node THRUSH that will match incoming X.25 calls with a subaddress in the range 11 to 20. This command is used if the executor node is a node with VAX P.S.I. software in multihost mode installed, serving as a connector node. Object 36 is the object number for VAX P.S.I. Access on the remote node.

SET NODE ALL

The SET NODE ALL command updates the volatile node database on the executor node with all the parameters stored for a particular node in the permanent database on the executor node.

Format

SET node-component ALL

Node Components

NODE	KNOWN NODES

Node Components

NODE node-id
Identifies the node whose parameters are to be updated.

KNOWN NODES
Indicates that all parameters for all known nodes are to be updated.

Examples

1. `NCP>SET KNOWN NODES ALL`

   This command loads all permanent database entries stored for all known nodes into the volatile database.

SET/DEFINE NODE

The SET NODE command creates or modifies node parameters in the volatile database. The DEFINE NODE command creates or modifies node parameters in the permanent database on the local node.

Format

SET node-component parameter [...]

DEFINE node-component parameter [...]

Node Components

NODE	KNOWN NODES

Node Components

NODE node-id

Identifies the node (local or remote) for which specified parameters are to be created or modified in the database.

KNOWN NODES

Indicates that the specified parameters for all known nodes are to be created or modified in the database.

Command Parameters

ACCESS option

Specifies the allowed logical link connections for the node. There are four options:

BOTH	Allows incoming and outgoing logical link connections. This is the default.
INCOMING	Allows logical link connections from the remote node.
NONE	Does not allow incoming or outgoing logical link connections to this node.
OUTGOING	Allows the local node to initiate connections to the remote node, but does not allow connections from the remote node.

If you have OPER privilege, you can override the access restriction specified in this parameter.

ADDRESS node-address

Specifies the address of the node.

COUNTER TIMER seconds

Specifies a timer whose expiration causes a node counter logging event. Seconds must be a decimal integer from 1 to 65,535.

CPU cpu-type
Identifies the node's CPU type. There are four possibilities:

DECSYSTEM1020
PDP11
PDP8
VAX

DIAGNOSTIC FILE filespec
Applies to nodes on Ethernet circuits. Identifies the file to be read when the adjacent node has been downline loaded and has requested diagnostics. The *filespec* is interpreted according to the file system of the executor node.

DUMP ADDRESS number
Identifies the address in memory to begin an upline dump of the adjacent node.

DUMP COUNT number
Specifies the default number of memory units to upline dump from the adjacent node.

DUMP FILE filespec
Identifies which file to write to when the adjacent node is dumped upline.

HARDWARE ADDRESS h-address
Identifies the address originally assigned to the controller for the system on the adjacent node. Used during operations such as downline loading to communicate with the system before the system has set up its physical address.

HOST node-id
Identifies the host node. For an adjacent node, the host address is a parameter that the adjacent node receives when it is downline loaded. If no host is specified, the default is the executor node.

VAX

INBOUND node-type (VAX only)
Required for nodes when the VERIFICATION INBOUND parameter is specified for the DDCMP circuit over which the connection is to be made. Specifies the type of the node. The *node-type* is checked by the executor node if the specified node attempts to form a dynamic connection with the executor node. If VERIFICATION INBOUND is not specified for the circuit, the INBOUND parameter for the node is ignored. There are two possible node types:

ENDNODE Allows the remote node to be connected only if it is configured as an end node.

ROUTER Allows the remote node to be connected whether it is configured as an end node or a router.♦

LOAD ASSIST AGENT filespec
Specifies the image that will define the system software to be downline loaded to an adjacent node. The load assist agent can be used to tailor the system software to be loaded. The LOAD ASSIST AGENT parameter applies only when the target is a VMScluster satellite.

LOAD ASSIST PARAMETER item
Specifies a parameter to be passed to a load assist agent. The LOAD ASSIST PARAMETER value applies only when the target is a VMScluster satellite. ♦

LOAD FILE filespec
Specifies a file containing the system software to be downline loaded to an adjacent node.

MANAGEMENT FILE filespec
Specifies a file containing the management information to be downline loaded to an adjacent node.

NAME node-name
Specifies the node name to be associated with the node identification. You can assign only one name to a node address.

NONPRIVILEGED item
Specifies nonprivileged inbound access control information for the node. Associate any of the following parameters with the NONPRIVILEGED parameter:

ACCOUNT account	Identifies the account for the default nonprivileged DECnet account on the designated node.
PASSWORD password	Identifies the password for the default nonprivileged DECnet account on the designated node.
USER user-id	Identifies the user name for the default nonprivileged DECnet account on the designated node.

PRIVILEGED item
Specifies privileged inbound access control information for the node. Associate any of the following parameters with the PRIVILEGED parameter:

ACCOUNT account	Identifies the account for the default privileged DECnet account on the designated node.
PASSWORD password	Identifies the password for the default privileged DECnet account on the designated node.
USER user-id	Identifies the user name for the default privileged DECnet account on the designated node.

VAX

RECEIVE PASSWORD password
Does not apply to nodes on a broadcast circuit. Defines the password (1 to 8 characters) that is expected from the remote node during a routing initialization sequence. You use this parameter only if verification is enabled or set to INBOUND for the circuit.♦

SECONDARY LOADER filespec
Specifies a file containing the secondary boot loader to be downline loaded to an adjacent node.

SERVICE CIRCUIT circuit-id
Specifies the circuit to be used for downline loading. This circuit is the default value for the VIA parameter of the LOAD command. The node identification must be that of the target node.

SERVICE DEVICE device-type
Identifies the target node's line controller for the service line over which the operation is to take place. See Appendix A, for a list of the DECnet for OpenVMS circuit and line devices and what they support.

VAX

Only the synchronous port on the DMF may be used for service operations.♦

SERVICE NODE VERSION version
Specifies the DECnet for OpenVMS software version of the node that downline loads its software to a target node. The two possibilities are PHASE III and PHASE IV. The default is PHASE IV.

SERVICE PASSWORD hex-password
Defines the password required to trigger the bootstrap mechanism on the target node. The password is a hexadecimal number. For broadcast circuits, the password is in the range 0 to FFFFFFFFFFFFFFFF.

VAX

For DDCMP circuits, the password is in the range 0 to FFFFFFFF.♦

SOFTWARE IDENTIFICATION software-id
Specifies the ID of the software to be downline loaded. You include this parameter only if the SOFTWARE TYPE parameter is specified.

SOFTWARE TYPE software-type
Identifies a particular file type to be downline loaded. There are four possibilities:

MANAGEMENT FILE
SECONDARY LOADER
SYSTEM
TERTIARY LOADER

The default is SECONDARY LOADER.

You include this parameter only if the SOFTWARE IDENTIFICATION parameter is specified.

TERTIARY LOADER filespec
Specifies a file containing a tertiary boot loader to be downline loaded to an adjacent node.

VAX

TRANSMIT PASSWORD password
Does not apply to nodes on an Ethernet or FDDI circuit. Specifies a password (1 to 8 characters) sent to the remote node during a routing initialization sequence. This parameter is used only if the VERIFICATION parameter has been set to ENABLED or INBOUND for the circuit.♦

Comments

You can use this command to create or modify parameters for a local or remote node. Refer to the *DECnet for OpenVMS Networking Manual* for a table of the parameters used with the SET NODE command according to their functions.

Examples

1.
```
NCP>SET NODE 5.14 NAME DENVER
```

   This command sets the node name of node 5.14 to DENVER.

2.
```
NCP>SET NODE 2.11 -
_ NONPRIVILEGED -
_ USER NETNONPRIV -
_ PASSWORD NONPRIV -
_ PRIVILEGED -
_ USER NETPRIV -
_ PASSWORD PRIV
```

   This command establishes default privileged and nonprivileged access control information for node 2.11.

3.
```
NCP>SET NODE 14 ADDRESS 2
```

   This command associates the information for node 1.14 with a new node whose address is 1.2. The executor is assumed to be in area 1.

4.
```
NCP>SET NODE LARK HARDWARE ADDRESS 08-00-2B-2D-88-7B
```

   This command associates with the node LARK the hardware address of the controller at node LARK. This information in the volatile database can be used during downline loading of target node LARK over a broadcast circuit.

5.
```
NCP>SET NODE 8.* ACCESS NONE
```

   This command prevents incoming or outgoing logical link connections to all nodes in area 8.

SET/DEFINE NODE CIRCUIT

The SET NODE command with the CIRCUIT parameter creates or modifies loop node parameters in the volatile database. The DEFINE NODE command with the CIRCUIT parameter creates or modifies loop node parameters in the permanent database.

The CIRCUIT parameter is the only valid parameter for loop nodes. You can assign only one loop node name to a particular circuit.

Format

SET NODE node-id CIRCUIT circuit-id

DEFINE NODE node-id CIRCUIT circuit-id

Node Component

NODE node-id
Identifies the loop node name or address for which the CIRCUIT parameter is to be created or modified in the database.

Command Parameter

CIRCUIT circuit-id
Identifies which circuit to use for all traffic to the node.

Examples

1. `NCP>SET NODE TESTER CIRCUIT FZA-0`

   This command sets the loop node name (TESTER) and identifies the circuit over which loop information is to be transmitted.

SET OBJECT ALL

The SET OBJECT ALL command updates the volatile database with all the object parameters stored for a particular object in the permanent database.

Format

SET object-component ALL

Object Components

OBJECT	KNOWN OBJECTS

Object Components

OBJECT object-name
Identifies the object whose parameters are to be updated.

KNOWN OBJECTS
Indicates that all parameters for all known objects are to be updated.

Examples

1. `NCP>SET KNOWN OBJECTS ALL`

   This command loads all permanent database parameter entries stored for all known objects into the volatile database.

SET/DEFINE OBJECT

The SET OBJECT command creates or modifies object parameters in the volatile database. The DEFINE OBJECT command creates or modifies object parameters in the permanent database on the local node.

Format

SET object-component parameter [...]

DEFINE object-component parameter [...]

Object Components

OBJECT	KNOWN OBJECTS

Object Components

OBJECT object-name

Identifies the object for which specified parameters are to be created or modified in the database.

KNOWN OBJECTS

Indicates that the specified parameters are applicable to all known objects.

Command Parameters

ACCOUNT account

Identifies the default user's account for access control on inbound connects to the object when no access control is specified by the remote node.

ALIAS INCOMING option

Specifies how a particular object responds to incoming connect requests directed to the alias node address. You establish the alias node address using the SET EXECUTOR command. There are two options for ALIAS INCOMING.

DISABLED — Does not allow a specified object to receive incoming connect requests that have been directed to the alias node address. An object whose resources are not accessible clusterwide should have ALIAS INCOMING disabled. If an attempt is made to connect to an object that does not have ALIAS INCOMING enabled, the status of NO SUCH OBJECT is returned.

ENABLED — Allows a specified object to receive incoming connect requests that have been directed to the alias node address. An object such as PHONE, which uses a protocol that depends on multiple links, should not be enabled for ALIAS INCOMING. By default, if an alias node identifier has been specified, ALIAS INCOMING is enabled for all objects except for PHONE.

ALIAS OUTGOING option

Specifies whether a particular object uses the alias node identifier specified in the SET EXECUTOR command in its outgoing connect requests and other protocols. Specify either of the following two options:

DISABLED — Does not allow a specified object to use the alias node address in outgoing connect requests.

ENABLED — Allows a specified object to use the alias node address in its outgoing connect requests. An object such as PHONE, which uses a protocol that depends on multiple links, should not have the ALIAS OUTGOING parameter enabled. By default, only the object MAIL has ALIAS OUTGOING enabled.♦

FILE filespec
Specifies the file name of the command procedure or image that is run as a result of an incoming connection to the object. If not specified, the default is SYS$SYSTEM:object-name.COM. When you specify an object for the first time, this parameter is mandatory.

NUMBER number
Specifies the object number. Numbers 1 through 127 are for use by Digital Equipment Corporation. Numbers 0 and 128 through 255 are for customer use. See Table 3–1 for a comprehensive list of reserved object numbers. When you specify an object for the first time, this parameter is mandatory.

OUTGOING CONNECT PRIVILEGES privilege-list
Specifies those privileges that a user must have in order to make an outbound connection to the object.

PASSWORD password
Identifies the default password for access control on inbound connects to the object when no access control is specified by the remote node. This password must match the password established for the username associated with the object.

PRIVILEGES privilege-list
Specifies those privileges normally required by the object. A user with those privileges may be supplied with default outbound privileged access control information when connecting to the object on a remote node.

PROXY option

Assigns the proxy login access defaults to individual objects. Specify one of the following four options:

BOTH	Allow both incoming and outgoing proxy login access. This is the default option.
INCOMING	Allows proxy login to the object.
NONE	Does not allow incoming or outgoing proxy login access.
OUTGOING	Allows logical links using the object to request proxy login on the remote node.

USER user-id

Identifies the default username for access control on inbound connects to the object when no access control is specified by the remote node.

Description

A DECnet object is identified by object name and object type. (The type is specified in the NUMBER parameter.)

The privilege list in the SET/DEFINE OBJECT PRIVILEGES command is used to validate the user privileges for outbound connections to that object. The access control information is used as the default access control for inbound connections.

Refer to the *DECnet for OpenVMS Networking Manual* for a table of object parameters and their functions. Table 3–1 lists the object type codes used with the SET OBJECT and DEFINE OBJECT commands. All values in Table 3–1 are expressed in decimal.

Table 3–1 Object Type Codes

Code	Object Type Mnemonic	Description
0	TASK	Digital or customer use
1–16		Reserved for Digital use
17	FAL	File Access Listener for remote file and record access
18	HLD	Host loader for RSX–11S downline task loading requests
19	NML	Network Management Listener object
20		RSTS/E media transfer program (NETCPY)
21–22		Reserved for Digital use
23	REMACP	Network terminal handler (host side)
24		Network terminal handler (terminal side)
25	MIRROR	Loopback mirror
26	EVL	Event receiver
27	MAIL	Mail utility
28		Reserved for Digital use
29	PHONE	VMS Phone utility and RSX–11M/M–PLUS Phone utility
30–41		Reserved for Digital use
42	CTERM	Network terminal handler
43–62		Reserved for Digital use
63	DTR	DECnet Test Receiver object
64–127		Reserved for Digital use
128–255		Reserved for customer use

Examples

1.
```
NCP>SET OBJECT NML NUMBER 19
```

   This command sets an object number, 19, for the NML object.

2.
```
NCP>SET OBJECT NML -
_ PRIVILEGES OPER DIAGNOSE -
_ USER NET_NONPRIV -
_ PASSWORD NET_NONPRIV
```

   This command establishes default access control information for the NML object and sets those privileges required for privilege access control information to be sent to a remote node when connecting to the object.

3.
```
NCP>SET OBJECT OBJONE FILE OBJONE.COM NUMBER 0 -
_ USER NET PASSWORD NET
```

   This command creates an object called OBJONE with a command procedure called OBJONE.COM. The incoming connection uses a password and user identification of NET.

4.
```
NCP>SET EXECUTOR ALIAS NODE 2.13
.
.
.
NCP>SET OBJECT FOX ALIAS OUTGOING ENABLED
```

   The SET EXECUTOR ALIAS NODE command establishes address 2.13 as the alias node identifier for the local node. The SET OBJECT command associates the object FOX with the alias node identifier for all outgoing connect requests.

5.
```
NCP>SET OBJECT T* PROXY NONE
```

   This command prevents incoming or outgoing proxy login access for all objects with names beginning with *T*.

6.
```
NCP>SET OBJECT TEST OUTGOING CONNECT PRIVILEGES READALL OPER
```

   This command specifies that a user process must have privileges READALL and OPER in order to make an outbound connection to object TEST.

SHOW AREA

The SHOW AREA command displays area information from the volatile database.

Format

SHOW area-component parameter [qualifier]

Area Components

ACTIVE AREAS
AREA
KNOWN AREAS

Qualifier

TO

Area Components

ACTIVE AREAS
Indicates that information for all active areas is to be displayed.

AREA area-id
Identifies a particular area for which information is to be displayed.

KNOWN AREAS
Indicates that information for all known areas is to be displayed.

Command Parameters

CHARACTERISTICS
Indicates that static area information is to be displayed.

STATUS
Indicates that dynamic area information is to be displayed.

SUMMARY
Indicates that only a summary of area information is to be displayed. This is the default display type.

Qualifier

TO filespec
Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Interpreting the Display

CIRCUIT circuit-id
This read-only parameter identifies the circuit used to get to a remote area.

COST cost
This read-only parameter represents the total cost over the current path to the destination area. This value is displayed only on routing nodes. The *cost* is a positive integer value associated with using a circuit. The Routing layer routes messages (data) along the path between two areas with the least cost.

HOPS hops

This read-only parameter represents the number of hops over to a destination area. A hop is the routing value representing the logical distance between two areas in a network.

NEXT NODE node-id

This read-only value indicates the next node on the circuit used to get to the destination area.

STATE state

This read-only value indicates the state of the area. The two possible states are REACHABLE and UNREACHABLE.

Examples

1.
```
NCP>SHOW KNOWN AREAS STATUS

Known Area Volatile Status as of 15-JUN-1992 09:50:34
  Area    State      Cost   Hops    Circuit         Next node to area
  2       reachable     0      0                     2.11 (BOSTON)
  3       reachable     4      1     SVA-0           3.5 (TRNTO)
  5       reachable     7      2     UNA-0           3.5 (TRNTO)
  11      reachable     4      1     SVA-1          11.9 (DALLAS)
  44      reachable    11      1     FZA-0          44.2 (LONDON)
```

   This command displays status information for all known areas in the network.

2.
```
NCP>SHOW KNOWN AREA CHARACTERISTICS

Known Area Volatile Characteristics as of 30-JAN-1992 11:16:27
Area = 2
State                     = reachable
Next node to area         =  2.11 (BOSTON)
Area = 3
State                     = reachable
Circuit                   = SVA-0
Next node to area         =  3.5 (TRNTO)
Area = 5
State                     = reachable
Circuit                   = UNA-0
Next node to area         =  3.5 (TRNTO)
Area = 11
State                     = reachable
Circuit                   = SVA-1
Next node to area         =  11.9 (DALLAS)

Area = 44
State                     = reachable
Circuit                   = FZA-0
Next node to area         =  44.2 (LONDON)
```

   This command displays characteristics for all known areas in the network.

3.
```
NCP>SHOW AREA 11 SUMMARY

Known Area Volatile Summary as of 15-JUN-1992 11:16:44
  Area    State          Circuit           Next node to area
  11      reachable       FZA-3              11.9 (DALLAS)
```

   This command displays only the most useful information for area 11 in the network.

SHOW/LIST CIRCUIT

The SHOW CIRCUIT command displays circuit information from the volatile database. The LIST CIRCUIT command displays circuit information from the permanent database.

Format

SHOW circuit-component parameter [qualifier] [...]

LIST circuit-component parameter [qualifier] [...]

Circuit Components

ACTIVE CIRCUITS	KNOWN CIRCUITS
CIRCUIT	

Qualifiers

ADJACENT NODE	TO

Circuit Components

ACTIVE CIRCUITS
Indicates that information for all active circuits is to be displayed (SHOW only).

CIRCUIT circuit-id
Identifies a particular circuit for which information is to be displayed.

KNOWN CIRCUITS
Indicates that information for all known circuits is to be displayed.

Command Parameters

CHARACTERISTICS
Indicates that static circuit information is to be displayed.

COUNTERS
Indicates that circuit error and performance statistics are to be displayed.

STATUS
Indicates that dynamic circuit information is to be displayed, including end node adjacencies and routing node adjacencies.

SUMMARY
Indicates that dynamic circuit information is to be displayed, including the routing adjacencies available to this circuit. SUMMARY is the default display type.

Qualifiers

ADJACENT NODE node-id
Indicates that the display of a list of circuits is to be restricted to those circuits leading to the specified adjacent node.

TO filespec
Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Interpreting the Display

Adjacent node node-id
This read-only parameter indicates an adjacent node on the circuit. There can be many adjacent nodes on an Ethernet or FDDI circuit.

Block size number
This read-only parameter is the block size in bytes for the adjacent node, as negotiated with the adjacent Routing layer during routing initialization.

Designated router node-id
This read-only value is the Routing layer identification of the node that is to be used for routing to nonrouting nodes (end nodes) on this circuit.

Listen timer seconds
This read-only parameter determines the maximum time allowed to elapse before a message (a Routing Hello message or a user message) is received from an adjacent node on the circuit. The value can be up to 65,535. Note that the LISTEN TIMER value is three times that of the HELLO TIMER circuit parameter.

Loopback name
This read-only parameter is the node name associated with a circuit for loopback testing. It identifies the circuit to be used for all traffic to the loop node.

Polling substate (VAX only)
Applies only to DDCMP CONTROL circuits. This read-only value represents the state of the tributary as determined by the polling algorithm when the multipoint polling state is AUTOMATIC. The polling substate is displayed as a tag on the polling state (for example, AUTOMATIC-INACTIVE). Possible values of the polling substate are as follows:

Active
Inactive
Dying
Dead

Substate
This read-only value is the operational substate of the circuit. The substate is displayed as a tag on the STATE parameter (for example, ON-SYNCHRONIZING). See Table C–1 for a complete list of circuit/line substates. Possible substate values are as follows:

Synchronizing
Starting
Reflecting
Looping
Loading
Dumping
Triggering
Autoservice
Autoloading
Autodumping
Autotriggering
Failed

Examples

1.
```
NCP>SHOW KNOWN CIRCUITS STATUS
Known Circuit Volatile Status as of 15-JUN-1992 15:39:04
   Circuit          State                    Loopback     Adjacent      Block
                                               Name          Node        Size
  DMC-0             on                                   3.5  (TRNTO)     576
  DMC-1             on       -starting
  UNA-0             on                                   2.22 (LARK)      576
                                                         2.23 (DOVE)      576
                                                         2.20 (ROBIN)     576
                                                         2.21 (THRUSH)    576
  X25-INC           on                        INC      44.2  (LONDON)    576
  X25-INC2          on       -synchronizing
  X25-RY1           off                       RY1
  X25-RY2           off                       RY2
  X25-ZK            off                       ZK
```

   This command displays status information for all known circuits connected to a DECnet for OpenVMS node. This information includes the current state of the circuit; a loop node name (if any) associated with the circuit; and the address, name, and block size of the adjacent node on that circuit.

2.
```
NCP>SHOW ACTIVE CIRCUITS CHARACTERISTICS

Active Circuit Volatile Characteristics as of 15-JUN-1992 15:39:21

Circuit = DMC-0

State                    = on
Service                  = enabled
Cost                     = 12
Hello timer              = 15
Listen timer             = 30
Maximum buffers          = 255
Verification             = disabled
Adjacent node            = 3.5 (TRNTO)
Listen timer             = 30

Circuit = UNA-0
State                    = on
Designated router        = 2.20 (ROBIN)
Cost                     = 1
Maximum routers allowed  = 33
Router priority          = 64
Hello timer              = 15
Verification             = disabled
Adjacent node            = 2.22 (LARK)
Listen timer             = 45


Circuit = X25-INC

State                    = on
Loopback name            = INC
Cost                     = 20
Hello timer              = 15
Listen timer             = 30
Owner                    = Executor
Usage                    = incoming
Type                     = X.25
Verification             = disabled
Adjacent node            = 44.2 (LONDON)
Listen timer             = 30
```

This command displays circuit characteristics for all circuits whose states are ON.

3. NCP>SHOW CIRCUIT MNA-0 STATUS

```
Circuit Volatile Status as of 30-JAN-1992 15:45:04
   Circuit          State         Loopback     Adjacent     Block
                                    Name         Node       Size
  MNA-0             on                      2.22 (LARK)       576
  MNA-0                                     2.23 (DOVE)       576
  MNA-0                                     2.20 (ROBIN)      576
  MNA-0                                     2.21 (THRUSH)     576
```

This command displays status information for circuit MNA–0.

4. NCP>TELL DOVE SHOW CIRCUIT MNA-0 CHARACTERISTICS

```
Circuit Volatile Characteristics as of 15-JUN-1992 15:46:20

Circuit = MNA-0

State                    = on
Designated router        = 2.20 (ROBIN)
Cost                     = 1
Maximum routers allowed  = 33
Router priority          = 64
Hello timer              = 15
Verification             = disabled
Adjacent node            = 2.20 (ROBIN)
Listen timer             = 45

Circuit = MNA-0

Adjacent node            = 2.11 (BOSTON)
Listen timer             = 45

Circuit = MNA-0

Adjacent node            = 2.22 (LARK)
Listen timer             = 45

Circuit = MNA-0

Adjacent node            = 2.21 (THRUSH)
Listen timer             = 45
```

This command displays circuit characteristics for circuit MNA–0 on node DOVE.

5. NCP>TELL MYNODE SHOW CIRCUIT TX-0-5 CHARACTERISTICS

```
Circuit Volatile Characteristics as of 30-JAN-1992 15:35:12
Circuit = TX-0-5
State                    = on
Substate                 = -synchronizing
Service                  = enabled
Cost                     = 10
Hello timer              = 15
Verification             = inbound
```

This command displays circuit characteristics for circuit TX-0-5 on MYNODE.

6. NCP>SHOW KNOWN CIRCUIT ADJACENT NODE BOSTON

```
Known Circuit Volatile Summary as of 15-JUN-1992 15:50:19
   Circuit          State              Loopback     Adjacent
                                         Name         Node
  FZA-0             on                        2.11 (BOSTON)
  SVA-0             on                        2.11 (BOSTON)
```

This command displays summary information for all circuits connected to adjacent node BOSTON.

7. NCP>SHOW ACTIVE CIRCUIT COUNTERS

```
Active Circuit Counters as of 30-JAN-1992 16:10:31
Circuit = FZA-0

      >65534  Seconds since last zeroed
      695803  Terminating packets received
     1847598  Originating packets sent
           0  Terminating congestion loss
      201179  Transit packets received
      200122  Transit packets sent
           0  Transit congestion loss
           0  Circuit down
           0  Initialization failure
           0  Adjacency down
           1  Peak adjacencies
     2057126  Data blocks sent
  1975840666  Bytes sent
      903317  Data blocks received
    32194824  Bytes received
           0  Unrecognized frame destination
           0  User buffer unavailable

Circuit = MNA-0
   >65534  Seconds since last zeroed
   184949  Terminating packets received
   214036  Originating packets sent
        0  Terminating congestion loss
   555679  Transit packets received
   555334  Transit packets sent
        2  Transit congestion loss
        1  Circuit down
        0  Initialization failure
  1142708  Data blocks sent
104475601  Bytes sent
  2389413  Data blocks received
277120757  Bytes received
        0  Unrecognized frame destination
        9  User buffer unavailable
```

This command displays counter information for all active circuits. See Appendix B for a description of each circuit counter.

8. NCP>SHOW CIRCUIT SVA-* SUMMARY

This command displays dynamic circuit information for all SVA circuits, including the routing adjacencies available to those circuits.

SHOW/LIST EXECUTOR

The SHOW EXECUTOR command displays node information from the volatile database. The LIST EXECUTOR command displays local node information from the permanent database.

Format

SHOW EXECUTOR parameter [qualifier]

LIST EXECUTOR parameter [qualifier]

Qualifier
TO

Command Parameters

CHARACTERISTICS
Indicates that static local node information is to be displayed.

COUNTERS
Indicates that local node error and performance statistics are to be displayed.

STATUS
Indicates that dynamic local node information is to be displayed.

SUMMARY
Indicates that a summary of local node information is to be displayed. This is the default display type.

Qualifier

TO filespec
Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Interpreting the Display

Active links number
This read-only parameter represents the number of active logical links from the executor to the destination node.

Delay seconds
This read-only parameter is the average round-trip delay in seconds from the executor to the destination node.

Management version n.n.n
This read-only parameter identifies the version number of the Network Management layer.

NSP version n.n.n
This read-only parameter identifies the version number of the End Communication layer.

Physical address p-address
This read-only parameter is the unique physical address used on a LAN to identify the executor node.

Routing version n.n.n

This read-only parameter identifies the version number of the Routing layer.

Examples

1.
```
NCP>SHOW EXECUTOR CHARACTERISTICS

Node Volatile Characteristics as of 15-JUN-1992 15:37:32

Executor node = 2.11 (RAVEN)

Identification           = DECnet for OpenVMS

Management version       = V4.0.0
Incoming timer           = 45
Outgoing timer           = 45
Incoming Proxy           = Enabled
Outgoing Proxy           = Enabled
NSP version              = V4.0.0
Maximum links            = 128
Delay factor             = 80
Delay weight             = 5
Inactivity timer         = 60
Retransmit factor        = 10
Routing version          = V2.0.0
Type                     = routing IV
Routing timer            = 600
Broadcast routing timer  = 40
Maximum address          = 1023
Maximum circuits         = 16
Maximum cost             = 1022
Maximum hops             = 15
Maximum visits           = 63
Maximum area             = 63
Max broadcast nonrouters = 64
Max broadcast routers    = 32
Maximum path splits      = 1
Area maximum cost        = 1022
Area maximum hops        = 30
Maximum buffers          = 100
Buffer size              = 576
Default access           = incoming and outgoing
Pipeline quota           = 1500
Alias incoming           = Enabled
Alias maximum links      = 32
Alias node               = 2.10 (CROW)
Path split policy        = Normal
Maximum Declared Objects = 31
DNS interface            = Enabled
IDP of ISO address       = 33
DNS namespace            = ENTERPRISE
```

This command displays characteristics for the executor, a DECnet for OpenVMS routing node. This display shows values that you have set for this node. In addition, it provides supplemental information about the software versions of Network Management, NSP, Routing, and the use of the DECdns namespace.♦

2. NCP>SHOW EXECUTOR STATUS

```
Node Volatile Status as of 15-JUN-1992 15:37:53

Executor node = 2.11 (BOSTON)

State                     = on
Physical address          = AA-00-04-00-AB-04
```

This command displays status information for the executor node. This format includes the operational state and the physical address.

3. NCP>SHOW EXECUTOR SUMMARY

```
Node Volatile Summary as of 15-JUN-1992 15:38:04

Executor node = 2.11 (BOSTON)

State                     = on

Identification            = DECnet for OpenVMS
```

This command displays summary information for the executor.

4. NCP>SHOW EXECUTOR COUNTERS

```
Node Counters as of 15-JUN-1992 15:38:17

Executor node = 2.11 (BOSTON)

   >65534  Seconds since last zeroed
  1073585  Bytes received
  1069826  Bytes sent
    96120  Messages received
    96120  Messages sent
     1267  Connects received
     1267  Connects sent
       11  Response timeouts
        0  Received connect resource errors
       15  Maximum logical links active
       34  Aged packet loss
       70  Node unreachable packet loss
        0  Node out-of-range packet loss
        0  Oversized packet loss
        0  Packet format error
        0  Partial routing update loss
        0  Verification reject
```

This command displays counter information for both the local node and the executor node. The counters for the executor node are always displayed. The counters for the local node are only displayed after the first link is established between the executor and itself.

Refer to Appendix B for a description of each node counter.

SHOW/LIST LINE

The SHOW LINE command displays line information from the volatile database. The LIST LINE command displays line information from the permanent database.

Format

SHOW line-component parameter [qualifier]

LIST line-component parameter [qualifier]

Line Components

ACTIVE LINES
KNOWN LINES
LINE

Qualifier

TO

Line Components

ACTIVE LINES
Indicates that information for all active lines is to be displayed (SHOW only).

KNOWN LINES
Indicates that information for all known lines is to be displayed.

LINE line-id
Identifies a particular line for which information is to be displayed.

Command Parameters

CHARACTERISTICS
Indicates that static line information is to be displayed.

COUNTERS
Indicates that line error and performance statistics are to be displayed.

STATUS
Indicates that dynamic line information is to be displayed.

SUMMARY
Indicates that a summary of line information is to be displayed. This is the default display type.

Qualifier

TO filespec
Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Examples

1. NCP>SHOW ACTIVE LINES CHARACTERISTICS

```
Active Line Volatile Characteristics as of 30-JAN-1992 17:54:19
Line = FZA-0

Receive buffers          = 10
Controller               = normal
Protocol                 = FDDI
Service timer            = 4000
Hardware address         = 08-00-2B-2D-88-7B
Device buffer size       = 1498
Requested TRT            = 8000
Valid transmission time  = 2620
Restricted token timeout = 1000
Ring purger enable       = on
NIF target               = 00-00-00-00-00-00
SIF configuration target = 00-00-00-00-00-00
SIF operation target     = 00-00-00-00-00-00
Echo target              = 00-00-00-00-00-00
Echo data                = 55
Echo length              = 1
```

This command displays line characteristics for all active lines (lines in ON state) for the executor. This display shows values for line parameters that you have set for individual lines.

2. NCP>SHOW KNOWN LINES STATUS

```
Known Line Volatile Status as of 15-JUN-1992 10:21:27

Line = MFA-0

State                     = on
Negotiated TRT            = 99840
Duplicate address flag    = unknown
Upstream neighbor         = 08-00-2B-1C-0D-BB
Old upstream neighbor     = 00-00-00-00-00-00
Upstream neighbor DA flag = unknown
Downstream neighbor       = 00-00-00-00-00-00
Old downstream neighbor   = 00-00-00-00-00-00
Ring purger state         = off
Ring error reason         = no error
Neighbor PHY type         = A
Link error estimate       = 15
Reject reason             = none
```

This command displays status information for all known lines connected to the local node. This display shows the current state of the line.

3. NCP>SHOW LINE MNA-0 SUMMARY

```
Line Volatile Summary as of 15-JUN-1992 10:22:11

   Line             State
   MNA-0             on
```

This command displays summary information for line MNA–0.

4. NCP>TELL LARK SHOW LINE MNA-0 CHARACTERISTICS

```
Line Volatile Characteristics as of 15-JUN-1992 10:23:41

Line = MNA-0

Receive buffers          = 9
Controller               = normal
Protocol                 = Ethernet
Hardware address         = AA-00-03-00-00-C0
Buffer size              = 1498
```

This command displays line characteristics for line MNA–0 on node LARK.

5. NCP>SHOW LINE FZA-0 COUNTERS

```
Line Counters as of  30-JAN-1992 10:25:43

Line = FZA-0

      >65534  Seconds since last zeroed
      879508  Data blocks received
       13438  Multicast blocks received
           0  Receive failure
    55687201  Bytes received
     1846203  Multicast bytes received
           0  Data overrun
     1972648  Data blocks sent
        9386  Multicast blocks sent
  1845189341  Bytes sent
     1058642  Multicast bytes sent
           0  Send failure
           0  Unrecognized frame destination
           0  System buffer unavailable
           0  User buffer unavailable
 >4294967294  MAC frame count
           0  MAC error count
           0  MAC lost count
           1  Ring initializations initiated
           2  Ring initializations received
           1  Ring beacons initiated
           0  Duplicate address test failures
           0  Duplicate tokens detected
           0  Ring purge errors
           0  FCI strip errors
           0  Traces initiated
           0  Traces received
           0  Directed beacons received
           0  Elasticity buffer errors
           0  LCT rejects
           0  LEM rejects
           0  Link errors
           2  Connections completed
```

This command shows the line counters for FDDI line FZA–0. Refer to Appendix B for a complete description of each line counter.

VAX

6. NCP>SHOW LINE DMC-* STATUS

This command displays status information for all DMC lines connected to the executor.♦

SHOW LINKS

The SHOW LINKS command displays link information from the volatile database.

Format

SHOW link-component parameter [qualifier]

Link Components

KNOWN LINKS LINK
KNOWN LINKS WITH NODE

Qualifier

TO

Link Components

KNOWN LINKS
Indicates that information for all known links is to be displayed.

KNOWN LINKS WITH NODE node-id
Indicates that information for all known links to the designated node is to be displayed.

LINK link-number
Displays information about the specific link.

Command Parameters

CHARACTERISTICS
Displays static link information.

STATUS
Displays dynamic link information.

SUMMARY
Displays a summary of link information. This is the default display type.

Qualifier

TO filespec
Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Interpreting the Display

State
This read-only value is the operational state of the link. See Appendix C for a complete list of link states. Possible state values are as follows:

CI sending
CI ACKed
CI receiving
Closed
CC sending

Run
DI received
DI sending

Examples

1. NCP>SHOW KNOWN LINKS

```
Known Link Volatile Summary as of 15-JUN-1992 15:53:42

   Link       Node           PID     Process     Remote link  Remote user
  4104   11.9 (DALLAS)     20600090  GRAY               3116  MAIL
  8229    3.5 (TRNTO)      2060014C  DAVIS              7217  NML
  3125    2.17 (NYC)       2060023D  REMACP             5175  THOMPSON_1
```

This command displays link information for all links known to the local node. This format is the same for all three display types. The display shows the local link number, node name and address to which the links apply, the process identification number (PID), the process associated with the link, the remote link, and the remote object or process name.

2. NCP>SHOW KNOWN LINKS WITH NODE DALLAS

```
Known Link Volatile Summary as of 15-JUN-1992 15:53:42

   Link       Node           PID     Process     Remote link  Remote user
  4104   11.9 (DALLAS)     20600090  GRAY               3116  MAIL
```

This command displays link information for all active links with remote node DALLAS.

3. NCP>SHOW KNOWN LINKS CHARACTERISTICS

```
Known Link Volatile Characteristics as of  30-JAN-1992 15:00:42

Link = 8245

State                    = run
PID                      = 264006AB
Remote node              = 2.58 (THRUSH)
Delay time               = 1
Remote link              = 9270
Remote user              = TERM
Username                 = MARTIN
Process name             = MARTIN

Link = 9270
State                    = CI received
PID                      = 26400091
Remote node              = 2.58 (THRUSH)
Delay time               = 1
Remote link              = 8245
Remote user              = BARNETT
Username                 = SYSTEM
Process name             = REMACP
```

This command displays link characteristics for all links. Refer to Appendix C for a complete description of each link state.

SHOW/LIST LOGGING

The SHOW LOGGING command displays logging information from the volatile database. The LIST LOGGING command displays logging information from the permanent database.

Format

SHOW logging-component parameter [qualifier] [...]

LIST logging-component parameter [qualifier] [...]

Logging Components

ACTIVE LOGGING
KNOWN LOGGING
LOGGING CONSOLE
LOGGING FILE
LOGGING MONITOR

Qualifiers

KNOWN SINKS
SINK NODE
TO

Logging Components

ACTIVE LOGGING
Indicates that information for all active logging is to be displayed (SHOW only).

KNOWN LOGGING
Indicates that information for all known logging is to be displayed.

LOGGING CONSOLE
Indicates that information for the logging console is to be displayed.

LOGGING FILE
Indicates that information for the logging file is to be displayed.

LOGGING MONITOR
Indicates that information for the logging monitor is to be displayed.

Command Parameters

EVENTS
Indicates that all static logging information is to be displayed.

STATUS
Indicates that dynamic logging information is to be displayed.

SUMMARY
Indicates that a summary of logging information is to be displayed. This is the default display type.

SHOW/LIST LOGGING

Qualifiers

KNOWN SINKS
Indicates that information for all known sinks is to be displayed.

SINK NODE node-id
Identifies a particular sink node for which information is to be displayed. If you do not include this parameter, NCP displays only local node logging.

TO filespec
Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Examples

1.
```
NCP>SHOW LOGGING FILE EVENTS KNOWN SINKS

Logging Volatile Events as of 15-JUN-1992 13:40:54

Logging sink type = file

Sink node         = 3.5 (TRNTO)
Events            = 4.0-5

Logging sink type = file

Sink node         = 11.9 (DALLAS)
Events            = 0.0-7
Events            = 2.0-1
Events            = 5.0-4
```

   This command displays events being logged to the logging file component as specified for all known sinks. This display lists the sink node, the source for events, and the actual events that are logged to the logging file at that node.

2.
```
NCP>SHOW ACTIVE LOGGING STATUS KNOWN SINKS

Active Logging Status as of 15-JUN-1992 14:02:15

Logging sink type = console

  Sink Node    Source      Events              State   Name

3.5 (TRNTO)    DMC-0       4.0-5,8-10          on
                           5.0-4

Logging sink type = file

Sink Node    Source      Events              State   Name

3.5 (TRNTO)    DMC-0       4.0-5,8-10          on      NET.LOG
                           5.0-4
11.9 (DALLAS)  DMC-3       5.0-4
11.9 (DALLAS)              0.0-7
                           2.0-1
```

   This command displays status information for all active logging at all known sinks for executor TRNTO. This display identifies each logging component and information particular to that component. The status format is similar to the events format, except that the name of the component and its operational state are included in this display.

3. NCP>SHOW KNOWN LOGGING SUMMARY SINK NODE TRNTO

```
Known Logging Volatile Summary as of 15-JUN-1992 15:30:20

 Logging sink type = monitor

Sink Node       Source          Events          State  Name

4.430 (TRNTO)  (All sources)       0.0-9       on
                (All sources)   2.0-1
                (All sources)   4.2-13 15-16
                                18-19
                (All sources)   5.0-18
                (All sources)   128.0-4
```

This command displays summary information for all known logging to node TRNTO. The summary format is the same as the events format.

SHOW/LIST MODULE CONFIGURATOR

The SHOW MODULE CONFIGURATOR command displays information about the Ethernet configurator module from the volatile database. The LIST MODULE CONFIGURATOR command displays information about the configurator module from the permanent database.

Format

SHOW MODULE CONFIGURATOR circuit-qualifier parameter [qualifier]

LIST MODULE CONFIGURATOR circuit-qualifier parameter [qualifier]

Circuit Qualifiers

CIRCUIT KNOWN CIRCUITS

Qualifier

TO

Circuit Qualifiers

CIRCUIT circuit-id

Applies only to Ethernet circuits. Indicates that configurator information should be displayed only for the Ethernet circuit identified by *circuit-id*.

KNOWN CIRCUITS

Applies only to Ethernet circuits. Indicates that configurator information should be displayed for all known Ethernet circuits.

Command Parameters

CHARACTERISTICS

Indicates that dynamic information about all active systems on the specified Ethernet circuits is to be displayed. (Specifying CHARACTERISTICS or STATUS results in a similar display.)

STATUS

Indicates that dynamic information about all active systems on the specified Ethernet circuits is to be displayed. (Specifying CHARACTERISTICS or STATUS results in a similar display.)

SUMMARY

Indicates that the only information to be displayed is the list of circuits being monitored and the length of time surveillance has been maintained. This is the default display type.

Qualifier

TO filespec

Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Interpreting the Display

Circuit name
This read-only parameter identifies the circuit for which surveillance is being maintained.

Device type device-type
This read-only value identifies the type of device over which the remote system is communicating on the circuit. The *device-type* value represents one of the standard line devices (for example, UNA or FZA.)

Elapsed time hours:minutes:seconds
This read-only value indicates the amount of time that surveillance has been enabled on the circuit. The *hours* value is a decimal integer up to 65,535; the *minutes* and *seconds* values are decimal integers in the range 0 to 59 (for example, 19:48:24).

Function list
This read-only parameter lists the maintenance functions that the remote system supports. The list consists of one or more of the following items:

Item	Meaning
Boot	Remote controlled boot
Counters	Data link counter can be read
Dump	Upline dump
Load	Multiblock downline load
Loop	Loopback
Primary	Primary loader

Hardware address h-address
This read-only value is the hardware address assigned to the Ethernet controller at the remote system.

Maintenance version n.n.n
This read-only value indicates the maintenance protocol version of the remote system.

Physical address p-address
This read-only value represents the physical address of a remote system on the Ethernet. The value is the unique physical address that the node has set for itself or, if the node has not set a physical address, the hardware address associated with the node's device controller.

Surveillance flag
This read-only value indicates whether surveillance is enabled or disabled for the system indicated.

Surveillance control
This read-only value indicates whether a list of active systems is to be kept for the indicated circuit. A control value of disabled means that the list is not being kept (this is the default value). A control value of enabled means that the list is being kept.

Time of last report day-month hour:minute:second

This read-only value is the date and time the remote system last reported in on a circuit under surveillance. The *day* value is a decimal integer in the range 1 to 31, *month* is the name of the month, *hour* is a decimal integer in the range 0 to 23, and *minute* and *second* are decimal integers in the range 0 to 59 (for example, 16-JUL 14:30:03).

Examples

1.
```
NCP>SHOW CONFIGURATOR KNOWN CIRCUITS SUMMARY TO PRINTFILE

Module Configurator Volatile Summary as of 15-JUN-1992 08:55:21

  Circuit           Surveillance     Elapsed Time
  MNA-0             enabled          00:12:31
```

   This command causes summary information about surveillance on all known Ethernet circuits to be directed to the file PRINTFILE.

2.
```
NCP>SHOW MODULE CONFIGURATOR KNOWN CIRCUITS STATUS

Module Configurator Volatile Status as of 10-Jul-1992 13:01:23

Circuit name             = SVA-0
Surveillance flag        = enabled
Elapsed time             = 02:29:11
Physical address         = AA-00-04-00-5D-FF
Time of last report      = 10-Jul 13:03:35
Maintenance version      = V3.0.0
Function list            = Loop, Data link counters
Hardware address         = 08-00-2B-02-39-0F
Device type              = SVA

Circuit name             = SVA-0
Surveillance flag        = enabled
Elapsed time             = 02:29:11
Physical address         = AA-00-04-00-42-DD
Time of last report      = 10-Jul 13:00:49
Maintenance version      = V4.0.0
Function list            = Loop, Console carrier, Data link counters
Hardware address         = 08-00-2B-18-38-E2
Device type              = MNA

Circuit name             = SVA-0
Surveillance flag        = enabled
Elapsed time             = 02:29:11
Physical address         = 08-00-2B-24-42-21
Time of last report      = 10-Jul 13:03:40
Maintenance version      = V3.0.0
Function list            = Loop
Hardware address         = 08-00-2B-24-42-21
Device type              = MFA
```

   This command displays information about active systems on all Ethernet circuits known to the executor node.

SHOW/LIST MODULE X25-ACCESS (Remote management only)

The SHOW MODULE X25-ACCESS command displays network names and parameters of the X.25 Access module from the volatile database. The LIST MODULE X25-ACCESS command displays networks and parameters of the X.25 Access module from the permanent database.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

SHOW MODULE X25-ACCESS network-qualifier parameter [qualifier]

LIST MODULE X25-ACCESS network-qualifier parameter [qualifier]

Network Qualifiers

NETWORK KNOWN NETWORKS

Qualifier

TO

Network Qualifiers

KNOWN NETWORKS
NETWORK network-name
Displays information about a specific X.25 network or about all of the X.25 networks accessed through any connector nodes.

Command Parameters

CHARACTERISTICS
Indicates that static X25-ACCESS information is to be displayed.

STATUS
Indicates that dynamic X25-ACCESS information is to be displayed.

SUMMARY
Indicates that a summary of X25-ACCESS information is to be displayed. This is the default display type.

Qualifier

TO filespec
Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Examples

1.
```
NCP>SHOW MODULE X25-ACCESS KNOWN NETWORKS STATUS TO MYFILE.NET
```

   This command writes status information for all known X.25 networks to the file named MYFILE.NET.

2. NCP>SHOW MODULE X25-ACCESS NETWORK PSS1 SUMMARY

```
Module X25-Access Volatile Summary as of  30-JAN-1992 12:00:56
Network                   = PSS1
Node                      = 13.4 (EGRET)
```

This command displays static network information about the X.25 network named PSS1.

SHOW/LIST MODULE X25-PROTOCOL (Remote management only)

The SHOW MODULE X25-PROTOCOL command displays the parameters of the protocol module from the volatile database. The LIST MODULE X25-PROTOCOL command displays the protocol module parameters stored in the permanent database. Note that the component qualifier is optional.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

SHOW MODULE X25-PROTOCOL [component-qualifier] parameter [qualifier]

LIST MODULE X25-PROTOCOL [component-qualifier] parameter [qualifier]

Component Qualifiers

DTE	KNOWN DTES
GROUP	KNOWN GROUPS
NETWORK	KNOWN NETWORKS

Qualifier

TO

Component Qualifiers

DTE dte-address
KNOWN DTES
Displays information about the specified DTE or about all known DTEs.

If you specify the DTE qualifier, you must associate with it either of the following parameters:

NETWORK network-name	Selects the DTE(s) specified on a particular network.
KNOWN NETWORKS	Selects the DTE(s) on all known networks.

GROUP group-name
KNOWN GROUPS
Displays information about the specified group or about all known groups.

NETWORK network-name
KNOWN NETWORKS
Displays information about the specified network or about all known networks.

Command Parameters

CHARACTERISTICS
Displays static module information.

SHOW/LIST MODULE X25-PROTOCOL (Remote management only)

COUNTERS
Displays DTE error and performance statistics. Does not apply to the LIST MODULE X25-PROTOCOL command.

STATUS
Displays dynamic module information. Does not apply to the LIST MODULE X25-PROTOCOL command.

SUMMARY
Displays a summary of module information. This is the default display type.

Qualifier

TO filespec
Specifies the output file. If you omit this parameter, the default file is SYS$OUTPUT.

Interpreting the Display

Active channels count
This read-only parameter is the count of known switched virtual circuit (SVC) logical channel numbers currently in use. This counter applies only to those channels defined by the CHANNELS parameter of the SET MODULE X25-PROTOCOL command. The channels can be in use for either outgoing or incoming SVCs.

Active switched count
This read-only parameter is the count of switched virtual circuits (SVCs) currently in use.

Maximum channels number
This read-only parameter indicates the number of logical channels defined.

Examples

1.
```
NCP>SHOW MODULE X25-PROTOCOL NETWORK FRED CHARACTERISTICS

Module X25-Protocol Volatile Characteristics as of 30-JAN-1992 17:55:24
Network                   = FRED
Profile                   = PSS
```

   This command displays module characteristics for the X25-PROTOCOL module.

2. NCP>SHOW MODULE X25-PROTOCOL KNOWN DTES CHARACTERISTICS

```
Known Module X25-Protocol Volatile Characteristics as of 30-JAN-1992 18:07:49
DTE                     = 12345
Network                 = TALOS
Line                    = KMV-0-0
Channels                = 31-1
Maximum channels        = 31
Maximum circuits        = 255
Default data            = 128
Default window          = 2
Maximum data            = 1024
Maximum window          = 7
Maximum clears          = 6
Maximum resets          = 6
Maximum restarts        = 6
Call timer              = 200
Clear timer             = 180
Reset timer             = 180
Restart timer           = 180
Interface mode          = DTE

DTE                     = 54321
Network                 = FRED
Line                    = DPV-0-0
Channels                = 64-1
Maximum channels        = 64
Maximum circuits        = 255
Default data            = 128
Default window          = 2
Maximum data            = 1024
Maximum window          = 7
Maximum clears          = 6
Maximum resets          = 6
Maximum restarts        = 6
Call timer              = 200
Clear timer             = 180
Reset timer             = 180
Restart timer           = 180
Interface mode          = DTE
```

This command displays characteristics for all DTEs known to the X25-PROTOCOL module.

3. NCP>SHOW MODULE X25-PROTOCOL DTE 123456789 NETWORK -
_ PSS-COMBINATION STATUS

```
Module X25-Protocol Volatile Status as of  30-JAN-1992 11:50:12
  DTE              Network          State                Active    Active
                                                         Channels  Switched
  123456789      PSS_COMBINATION  on   -running            0         0
```

This command displays the status of DTE 123456789 in the network PSS-COMBINATION.

4. NCP>SHOW MODULE X25-PROTOCOL DTE 123456789 NETWORK FRED COUNTERS

```
Module X25-Protocol Counters as of 30-JAN-1992 18:08:26
DTE                     = 54321
Network                 = FRED
        7144  Seconds since last zeroed
       11705  Bytes received
       11705  Bytes sent
         501  Data blocks received
         501  Data blocks sent
           2  Calls received
           2  Calls sent
           0  Fast selects received
           0  Fast selects sent
           2  Maximum switched circuits active
           1  Maximum channels active
           0  Received call resource errors
           0  Locally initiated resets
           0  Remotely initiated resets
           0  Network initiated resets
           0  Restarts
```

This command displays information about each DTE counter. Refer to Appendix B for a description of each DTE counter.

5. NCP>SHOW MODULE X25-PROTOCOL KNOWN GROUPS CHARACTERISTICS

```
Known Module X25-Protocol Volatile Characteristics as of 30-JAN-1992 18:13:50
 Group             DTE              Network          Number    Type
  SEALS            54321            FRED                54   bilateral
  WALES            54321            FRED                 6
```

This command displays information about each closed user group.

SHOW/LIST MODULE X25-SERVER/X29-SERVER (Remote management only)

The SHOW MODULE X25-SERVER and SHOW MODULE X29-SERVER commands display the parameters of the X.25 and X.29 call handlers stored in the volatile database. The LIST MODULE X25-SERVER and LIST MODULE X29-SERVER commands display the parameters of the X.25 and X.29 call handlers stored in the permanent database.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

SHOW MODULE X25-SERVER [component-qualifier] parameter [qualifier]

SHOW MODULE X29-SERVER [component-qualifier] parameter [qualifier]

LIST MODULE X25-SERVER [component-qualifier] parameter [qualifier]

LIST MODULE X29-SERVER [component-qualifier] parameter [qualifier]

Component Qualifiers

DESTINATION KNOWN DESTINATIONS

Qualifier

TO

Component Qualifiers

DESTINATION name
KNOWN DESTINATIONS
Displays information about the specified destination or about all known destinations.

Command Parameters

CHARACTERISTICS
Displays static module information.

COUNTERS
Displays module error and performance statistics. Does not apply to the LIST MODULE X25-SERVER or LIST MODULE X29-SERVER command.

STATUS
Displays dynamic module information. Does not apply to the LIST MODULE X25-SERVER or LIST MODULE X29-SERVER command.

SUMMARY
Displays a summary of module information. This is the default display type.

SHOW/LIST MODULE X25-SERVER/X29-SERVER (Remote management only)

Qualifier

TO filespec

Specifies the output file. If you omit this parameter, the default file is SYS$OUTPUT.

Interpreting the Display

Active circuits count

This read-only parameter indicates the number of circuits the module currently has open.

Examples

1.
```
NCP>SHOW MODULE X25-SERVER CHARACTERISTICS

Module X25-Server Volatile Characteristics as of  30-JAN-1992 12:04:58

Maximum circuits         = 255
```

   This command displays characteristics for the X25-SERVER module.

2.
```
NCP>LIST MODULE X25-SERVER DESTINATION XTR CHARACTERISTICS

Module X25-Server Permanent Characteristics as of  30-JAN-1992 12:04:27

Destination              = XTR
Object                   = PSIXTR
Priority                 = 30
Sending DTE              = 234273412345
Subaddresses             = 33-35
```

   This command displays server module characteristics from the permanent database for destination XTR.

3.
```
NCP>SHOW MODULE X25-SERVER STATUS

Module X25-Server Volatile Status as of  30-JAN-1992 12:05:13

Active circuits          = 1
State                    = on
```

   This command displays status information for the X25-SERVER module.

4.
```
NCP>SHOW MODULE X25-SERVER COUNTERS

Module X25-Server Counters as of  30-JAN-1992 12:05:21

        4704  Seconds since last zeroed
           7  Maximum circuits active
           2  Incoming calls rejected, no resources
```

   This command displays counter information for each server module. See Appendix B for a description of each server module counter.

SHOW/LIST NODE

The SHOW NODE command displays node information from the volatile database. The LIST NODE command displays node information from the permanent database available to the local node.

Format

SHOW node-component parameter [qualifier]

LIST node-component parameter [qualifier]

Node Components

ACTIVE NODES
ADJACENT NODES
KNOWN NODES
LOOP NODES
NODE

Qualifier

TO

Node Components

ACTIVE NODES

For a routing node, indicates that information about all reachable nodes is to be displayed. For a nonrouting node (end node), indicates that information about the executor is to be displayed. Optionally, you can associate the following CIRCUIT parameter with this parameter:

CIRCUIT circuit-id — Specifies that the display of a list of nodes is to be restricted to those nodes adjacent to the specified circuit.

ADJACENT NODES

Indicates that information about all adjacent nodes is to be displayed. Adjacent nodes are those the executor perceives Routing can reach that are separated from the executor by a single circuit. Each occurrence of a node on a different circuit appears as a separate adjacent node. Optionally, you can associate the following CIRCUIT parameter with this parameter:

CIRCUIT circuit-id — Specifies that the display of a list of nodes is to be restricted to those nodes adjacent to the specified circuit.

KNOWN NODES

Indicates that information about all known nodes is to be displayed. Optionally, you can associate the following CIRCUIT parameter with this parameter:

CIRCUIT circuit-id — Specifies that the display of a list of nodes is to be restricted to those nodes adjacent to the specified circuit.

LOOP NODES

Indicates that information about all loop nodes is to be displayed.

NODE node-id

Identifies a particular node about which information is to be displayed.

Command Parameters

CHARACTERISTICS
Indicates that static node information is to be displayed.

COUNTERS
Indicates that node error and performance statistics are to be displayed.

STATUS
Indicates that dynamic node information is to be displayed.

SUMMARY
Indicates a summary of node information is to be displayed. This is the default display type.

Qualifier

TO filespec
Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Interpreting the Display

Active links number
This read-only parameter represents the number of active logical links from the executor to the destination node.

Circuit circuit-id
This read-only parameter identifies the circuit used to get to a remote node.

Cost number
This read-only parameter represents the total cost over the current path to the destination node. The DECnet Routing layer routes messages (data) along the path between two nodes with the smallest cost. Cost is a positive integer value.

Delay seconds
This read-only parameter is the average round-trip delay in seconds from the executor to the destination node.

Hops number
This read-only parameter indicates the number of hops from the executor node to a destination node. A hop is a value assigned by the Routing layer that represents the logical distance between two nodes on a network.

Management version n.n.n
This read-only parameter identifies the version number of the Network Management layer.

Next node node-id
This read-only parameter indicates the address and name of the next node on the circuit used to get to the node whose status is being displayed. Knowing which node is the partner on the next hop of the path to the destination node aids in tracing the path to that destination over a large number of hops.

NSP version n.n.n
This read-only parameter identifies the version number of the End Communication layer (ECL).

Physical address p-address
This read-only parameter is the physical address that identifies the executor node.

Routing version n.n.n
This read-only parameter identifies the version number of the Routing layer.

Type node-type
This read-only parameter indicates the type of the specified node. The values of *node-type* are as follows:

Nonrouting III

Nonrouting IV

Routing IV

Routing III

Area

If the specified node is not adjacent to the local node, the *node-type* will be blank.

Examples

1.
```
NCP>SHOW ACTIVE NODES CHARACTERISTICS

Active Node Volatile Characteristics as of 15-JUN-1992  13:38:34

Executor node = 2.11 (BOSTON)

Identification           = DECnet for OpenVMS
Management version       = V4.0.0
Incoming timer           = 45
Outgoing timer           = 45
Incoming Proxy           = Enabled
Outgoing Proxy           = Enabled
NSP version              = V3.2.0
Maximum links            = 128
Delay factor             = 80
Delay weight             = 5
Inactivity timer         = 60
Retransmit factor        = 10
Routing version          = V2.0.0
Type                     = routing IV
Routing timer            = 600
Maximum address          = 1023
Maximum circuits         = 16
Maximum cost             = 1022
Maximum hops             = 15
Maximum visits           = 63
Maximum area             = 63
Max broadcast nonrouters = 64
Max broadcast routers    = 32
Maximum path splits      = 1
Area maximum cost        = 1022
Area maximum hops        = 30
Maximum buffers          = 100
Buffer size              = 576
Default access           = incoming and outgoing
Pipeline quota           = 1500
Alias incoming           = Enabled
Alias maximum links      = 32
Alias node               = 2.10 (CLUSTR)
Path split policy        = Normal
```

```
Remote node =    3.5 (TRNTO)
Nonprivileged user id    = NETNONPRIV
Remote node =    11.9 (DALLAS)
Nonprivileged user id    = NETNONPRIV

Remote node =   12.34 (MYNODE)
Inbound                  = router
Remote node =   2.13 (KANSAS)
Nonprivileged user id    = NETNONPRIV

Remote node =   2.17 (NYC)
Nonprivileged user id    = NETNONPRIV

Loop node =    0 (TESTER)
```

This command displays characteristics for all active nodes. This display shows values that you have set for both the local node and remote nodes.

2. `NCP>SHOW NODE TRNTO STATUS`

```
Node Volatile Status as of 15-JUN-1992 13:39:05

  Node      State       Active  Delay    Type       Cost Hops  Line
                       Links
3.5 (ROBIN) reachable     1       2    routing IV    1    1    DMC-0
```

This command displays status information about remote node ROBIN. This format includes the following information:

- Node name and address
- Routing state (reachable or unreachable)
- Active links associated with the node
- Delay time required to set the retransmission (estimated round trip delay).
- Node type (phase II, routing III, routing IV, nonrouting III, nonrouting IV, or area).
- Minimum total path cost to the node. This value represents the cost of the actual path a packet will use to get to this node.
- Minimum total hops to the node. Note that the minimum cost path may be different from the minimum hops path. This value does not necessarily prevent the actual path from being used.
- Line associated with the node, that is, the line over which traffic to the node is currently being routed.

Full status information is displayed only for reachable nodes.

3. NCP>SHOW KNOWN NODES SUMMARY

```
Known Node Volatile Summary as of 15-JUN-1992 13:39:47

Executor node = 2.11 (BOSTON)

State                     = on
Identification            = DECnet for OpenVMS
Active links              = 2


Node             State    Active   Delay     Circuit    Next node
                          Links

1.2  (TRNTO)     reachable                   FZA-0      1.55
1.3  (DALLAS)    reachable    1       2      FZA-0      1.55
1.4  (KANSAS)    reachable    1       4      FZA-0      1.55
1.9  (DENVER)    unreachable
1.15 (BANGOR)    unreachable
1.17 (NYC)       reachable                   FZA-0      1.55
```

This command displays summary information for all known nodes. The summary format contains a subset of the information displayed in the status format. Note that the display also shows loop nodes and the line associated with the loop node name.

4. NCP>SHOW NODE TRNTO COUNTERS

```
Node Counters as of  15-JUN-1992 13:40:08

Remote node =    3.5 (TRNTO)

       18748  Seconds since last zeroed
      557304  Bytes received
      231849  Bytes sent
       27703  Messages received
       27556  Messages sent
          56  Connects received
           5  Connects sent
          12  Response timeouts
           0  Received connect resource errors
```

This command displays counter information about remote node TRNTO. Note that remote node counters are a subset of those maintained for the local node. Refer to Appendix B for a description of each node counter.

5. NCP>SET NODE LOCAL LINE DMC-0
NCP>SHOW LOOP NODES STATUS

```
Loop Node Volatile Status as of 15-JUN-1992 13:43:19

  Node    State     Active Delay    Type   Cost  Hops  Line
                    Links

0 (LOCAL)                                              DMC-0
```

This command displays status information about all loop nodes.

6. NCP>SHOW NODE M* STATUS

This command displays status information about all nodes whose names begin with *M*.

7. NCP>SHOW NODE %SU STATUS

This command requests a display of status information of all nodes that have three-letter node names and *SU* as the second and third characters of their names.

SHOW/LIST OBJECT

The SHOW OBJECT command displays object information from the volatile database. The LIST OBJECT command displays object information from the permanent database.

Format

SHOW object-component parameter [qualifier]

LIST object-component parameter [qualifier]

Object Components

OBJECT KNOWN OBJECTS

Qualifier

TO

Object Components

KNOWN OBJECTS

Indicates that information about all known objects is to be displayed.

OBJECT object-name

Identifies a particular object about which information is to be displayed.

Command Parameters

CHARACTERISTICS

Indicates that static object information is to be displayed. The SHOW OBJECT CHARACTERISTICS command displays only those parameters that you have defined.

STATUS

Indicates that dynamic object information is to be displayed.

SUMMARY

Indicates that a summary of object information is to be displayed. This is the default display type.

Qualifier

TO filespec

Specifies the output file. If none is specified, SYS$OUTPUT is the default.

Examples

1.
```
NCP>SHOW OBJECT MAIL CHARACTERISTICS

Object Volatile Characteristics as of 15-JUN-1992 13:46:22

Object = MAIL
Number                  = 27
File id                 = MAIL.EXE
User id                 = NETNONPRIV
Proxy access            = outgoing
Alias outgoing          = Enabled
Alias incoming          = Enabled
```

This command displays object characteristics for the MAIL object on a DECnet for OpenVMS node. This display shows values that you have set for the object.

2. `NCP>SHOW OBJECT MAIL CHARACTERISTICS TO MAIL_CHARS.NET`

   This command copies MAIL object characteristics from the volatile database to a file named MAIL_CHARS.NET.

3. `NCP>SHOW OBJECT NML SUMMARY`

```
Object Volatile Summary as of 15-JUN-1992 12:07:44

   Object   Number  File/PID                  User Id          Password
  NML           19  NML.EXE                   NML$SERVER       BLUESKIES
```

   This command displays object characteristics for the NML object.

4. `NCP>SHOW OBJECT I* SUMMARY`

   This command displays object characteristics for all objects whose names begin with *I*.

TELL

The TELL command specifies an NCP command for execution at a remote node. The NCP command is entered as a parameter to the TELL command. Note that the TELL command sets the executor to the remote node only for a single NCP command.

Format

TELL node-component parameter [...] command

Node Component
node-spec

Node Component

node-spec
Specifies a node name or address optionally followed by access control information. Use one of the following formats:

node-id

node-id"user password account"

The *node-id* can be a logical name. You can override access control in a logical name by specifying explicit access control information in the command.

Command Parameters

ACCOUNT account
Identifies the user's account for access control verification at the designated executor node. This parameter is not required for DECnet for OpenVMS nodes since DECnet for OpenVMS does not use this information to validate access control.

PASSWORD password
Identifies the user's password for access control verification at the designated executor node.

USER user-id
Identifies the user's ID for access control verification at the designated executor node.

Comments

You cannot use the TELL prefix with the CLEAR EXECUTOR NODE, COPY KNOWN NODES, or SET EXECUTOR NODE commands.

Examples

1. `NCP>TELL 3.5 SET KNOWN LINES ALL`

   This command sets the executor to node 3.5 where the SET KNOWN LINES ALL command executes.

2. `NCP>TELL TRNTO"GRAY MARY" SHOW KNOWN LINES`
   .
   .
   .

   This command uses access control information to set the executor to node TRNTO where the SHOW KNOWN LINES command executes. The information is displayed locally.

3. `NCP>TELL TRNTO USER SYSTEM PASSWORD MANAGER ZERO EXECUTOR`

   This command uses an alternate access control format to set the executor to node TRNTO where the ZERO EXECUTOR command executes.

TRIGGER NODE

The TRIGGER NODE command starts the bootstrap mechanism of a target node to allow the node to load itself. Use this command to initiate the loading sequence for an unattended system.

Format

TRIGGER node-component parameter [...]

Node Component
NODE

Node Component

NODE node-id
Identifies the node whose bootstrap is to be triggered.

Command Parameters

PHYSICAL ADDRESS p-address
Applies only to nodes on FDDI and Ethernet circuits. Specifies the physical address that the target node currently uses to identify itself. The value is the physical address that the target node has set for itself or, if the target node has not set a physical address, the hardware address associated with the target node's device controller.

SERVICE PASSWORD password
Identifies the password required to trigger the bootstrap mechanism on the target node. The password is a hexadecimal number, 8 to 16 digits long. For broadcast circuits, it is in the range 0 to FFFFFFFFFFFFFFFF.

VAX

For DDCMP circuits, it is in the range 0 to FFFFFFFF.♦

VIA circuit-id
Identifies the circuit over which the operation is to take place.

Examples

1. `NCP>TRIGGER NODE NYC`

   This command triggers the bootstrap mechanism on node NYC to initiate a downline load operation.

2. `NCP>TRIGGER NODE NYC SERVICE PASSWORD FEFEFEFEFEFEFEFE VIA FZA-1`

   This command provides a service password in order to trigger the bootstrap mechanism on node NYC and to initiate a downline load operation over FDDI circuit FZA–1.

3. `NCP>TRIGGER NODE SWIFT PHYSICAL ADDRESS AA-00-04-00-07-04 -`
   `_ VIA MNA-0`

   This command triggers the bootstrap mechanism on node SWIFT to initiate a downline load operation. The executor node uses the physical address specified in the command to address the node SWIFT on Ethernet circuit MNA–0.

4. `NCP>TRIGGER NODE SWIFT VIA MNA-0`

   This command triggers the bootstrap mechanism on node SWIFT on Ethernet circuit MNA–0 to initiate a downline load operation. DECnet for OpenVMS at the executor node attempts to address node SWIFT using a physical address derived from the node number. If node SWIFT does not recognize this address, the executor node uses the hardware address for node SWIFT if specified in the volatile database at the executor node.

TRIGGER VIA

The TRIGGER VIA command starts the bootstrap mechanism of a target node using the specified circuit to allow the node to load itself. Use this command to initiate the loading sequence for an unattended system.

Format

TRIGGER VIA circuit-id parameter [...]

Command Parameters

PHYSICAL ADDRESS p-address
Applies only to target nodes on FDDI and Ethernet circuits. Specifies the physical address that the target node currently uses to identify itself. The value is the physical address that the target node has set for itself or, if the target node has not set a physical address, the hardware address associated with the target node's device controller. This parameter must be included if an FDDI or Ethernet circuit is specified in VIA *circuit-id*.

SERVICE PASSWORD password
Identifies the password required to trigger the bootstrap mechanism on the target node. The password is a hexadecimal number. For broadcast circuits, it is in the range 0 to FFFFFFFFFFFFFFFF. The word SERVICE is optional.

VAX

For DDCMP circuits, it is in the range 0 to FFFFFFFF.♦

Comments

This command triggers the target node through the specified circuit. The target node identification is obtained from the volatile database on the executor node.

Examples

VAX

1.
```
NCP>TRIGGER VIA DMC-0
```

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

2.
```
NCP>TRIGGER VIA MNA-0 PHYSICAL ADDRESS AA-00-04-00-07-04
```

   This command triggers the bootstrap mechanism at the target node whose physical address on circuit MNA–0 is specified in the command.

ZERO CIRCUITS

The ZERO CIRCUITS command resets circuit counters. Note that the word COUNTERS is optional.

Format

ZERO circuit-component [COUNTERS]

Circuit Components

CIRCUIT	KNOWN CIRCUITS

Circuit Components

CIRCUIT circuit-id
Identifies a particular circuit for which counters are to be reset.

KNOWN CIRCUITS
Indicates that counters for all known circuits are to be reset.

Examples

1. `NCP>ZERO KNOWN CIRCUITS COUNTERS`

   This command resets all circuit counters for all known circuits.

ZERO EXECUTOR

The ZERO EXECUTOR command resets all counters. Note that the word COUNTERS is optional.

Format

ZERO EXECUTOR [COUNTERS]

Examples

1. `NCP>ZERO EXECUTOR COUNTERS`

   This command resets all counters for the local node.

ZERO LINE

The ZERO LINE command resets line counters. Note that the word COUNTERS is optional.

Format

ZERO line-component [COUNTERS]

Line Components

LINE	KNOWN LINES

Line Components

LINE line-id
Identifies a particular line for which counters are to be reset.

KNOWN LINES
Indicates that counters for all known lines are to be reset.

Examples

1. `NCP>ZERO KNOWN LINES COUNTERS`

   This command resets all line counters for all known lines.

2. `NCP>ZERO LINE FZA-0`

   This command resets line counters for the FDDI line FZA–0.

ZERO MODULE X25-PROTOCOL (Remote management only)

The ZERO MODULE X25-PROTOCOL command zeroes the counters for a particular DTE or for all known DTEs. Note that the word COUNTERS is optional.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

ZERO MODULE X25-PROTOCOL parameter [qualifier] [COUNTERS]

Command Parameters

DTE	KNOWN DTES

Qualifiers

NETWORK	KNOWN NETWORKS

Command Parameters

DTE dte-address

Zeroes the counters for the specified DTE only.

KNOWN DTES

Zeroes the counters for all known DTEs.

You must associate the DTE parameter with either of the following qualifiers:

NETWORK network-name	Selects the DTE(s) specified on a particular network.
KNOWN NETWORKS	Selects the DTE(s) on all known networks.

Examples

1. `NCP>ZERO MODULE X25-PROTOCOL KNOWN DTES COUNTERS`

   This command zeroes all DTE counters for all known DTEs on the default network.

2. `NCP>ZERO MODULE X25-PROTOCOL KNOWN DTES NETWORK TELENET COUNTERS`

   This command zeroes counters for all DTEs on the network Telenet.

ZERO MODULE X25-SERVER (Remote management only)/X29-SERVER

The ZERO MODULE X25-SERVER and ZERO MODULE X29-SERVER commands zero the counters for the X.25 and X.29 call handlers. Note that the word COUNTERS is optional.

Note

Use this command to manage VAX P.S.I. on remote nodes. To manage a remote node, enter this command as a parameter to the TELL command, or use the SET EXECUTOR NODE command first.

Format

ZERO MODULE X25-SERVER [COUNTERS]

ZERO MODULE X29-SERVER [COUNTERS]

Examples

1. `NCP>ZERO MODULE X25-SERVER COUNTERS`

   This command zeroes all module counters associated with the X.25 call handler.

ZERO NODE

The ZERO NODE command resets node counters. Note that the word COUNTERS is optional.

Format

ZERO node-component [COUNTERS]

Node Components

NODE	KNOWN NODES

Node Components

NODE node-id
Identifies a particular node for which counters are to be reset.

KNOWN NODES
Indicates that counters for all known nodes are to be reset.

Examples

1. `NCP>ZERO NODE TRNTO COUNTERS`

   This command resets all node counters maintained on the local node for remote node TRNTO.

Part II

DECnet for OpenVMS DTS/DTR Utility

4

DECnet Test Sender/DECnet Test Receiver (DTS/DTR) Commands

DTS and DTR are cooperating tasks that perform various functions to exercise the DECnet software and test communication lines. DTS initiates each test by issuing a connection request to DTR. DTS passes parameters pertinent to the type of test request to DTR. The user interface to DTS consists of a group of commands. These commands enable you to select the type of test and subtest and to specify options relative to the operation of the test. These options include parameters to regulate the duration of the test, the type of information to be used for the test, and buffer-level control.

Note

The OpenVMS DECnet Test Receiver image (DTRECV) cannot perform multiple buffering. In addition, it does not support any negative acknowledgment (NAK) or BACK options nor flow control other than segment. The OpenVMS DECnet Test Sender image (DTSEND) does not support multiple buffering either.

4.1 DTS Command Syntax

This section describes the DTS command syntax for all tests. Each command consists of a command keyword, optional general command qualifiers, each preceded by a slash (/) character, and optional command-specific qualifiers, each preceded by a slash character. You may abbreviate DTS commands and qualifiers, as long as the command is unambigious.

Note that, when using DTS commands in a batch job, you must use uppercase characters in the command.

The general DTS command syntax and the definition of each command component is as follows.

```
command-keyword/general-qualifier,.../command-specific-qualifier,...
```

command-keyword — Specifies the general test to be performed.

CONNECT	Indicates a connect test
DISCONNECT	Indicates a disconnect test
DATA	Indicates a data test
INTERRUPT	Indicates an interrupt test

general-qualifier	Specifies one or more general test characteristics regardless of the test to be performed. These qualifiers and their default values are described in each command (CONNECT, DATA, DISCONNECT, and INTERRUPT) where they apply.
command-specific-qualifier	Specifies one or more test-specific qualifiers that regulate the operation of the test.

The general command syntax adheres to the DCL rules for comments and continuation lines, as shown in the following example:

```
$ RUN SYS$SYSTEM:DTSEND
_TEST:   DATA/PRINT/TYPE=SEQ         !Perform Data Test
   .
   .
   .
_TEST:   INTERRUPT/NODENAME=DALLAS -
         _/PRINT/TYPE=PAT            !Perform Interrupt Test
   .
   .
   .
_TEST:   EXIT
$
```

4.2 DTS/DTR Usage Summary

DECnet Test Sender (DTS) and DECnet Test Receiver (DTR) are the DECnet transmitter and receiver test programs that exercise network task-to-task capabilities. DTSEND and DTRECV are the DECnet for OpenVMS implementations of these programs. The images and command files to use DTSEND and DTRECV are included with the DECnet for OpenVMS software.

There are four basic network tests provided by DTS and DTR:

- Connection tests
- Data tests
- Disconnection tests
- Interrupt tests

Each test is composed of a set of subtests. See the following sections for a description of each test and its subtests.

Format

The following shows of the DCL command that invokes DTSEND:

```
RUN SYS$SYSTEM:DTSEND
```

Command Parameters

This command has no command parameters.

Before invoking DTS, be sure that the necessary line (if any) is in the ON state. To invoke DTS, type RUN SYS$SYSTEM:DTSEND in response to the DCL prompt.

DTS returns with the following information and prompt:

```
DTS Version xxx initiated on dd-mmm-yy hh:mm:ss
_Test:
```

DTS accepts input entered interactively at the terminal. DTS also accepts a command procedure as input.

To exit DTS and thereby terminate the test, enter EXIT at the *_Test:* prompt. For example:

```
_Test:   EXIT

DTS terminated on dd-mmm-yy hh:mm:ss
$
```

Use the /PRINT qualifier to print test results. Also, you can use the /STATISTICS and /DISPLAY qualifiers to print statistics and messages transmitted to DTR, respectively. The /STATISTICS and /DISPLAY qualifiers apply only to the data and interrupt tests. See the individual commands for more information.

4.3 DTS Commands

This section explains how to use the four DTS tests. Examples are provided for each test.

DTS Commands

VAX

CONNECT
DATA
DISCONNECT
HELP ♦
INTERRUPT

CONNECT

Invokes a connection test.

Format

CONNECT [/general-qualifier[,...] /command-specific-qualifier[,...]]

General Qualifiers

/NODENAME=node-id
Identifies the node (by name or address) on which DTR is to run. The default is the local node. Because you cannot use access control information as part of the node name, DTR runs on the default nonprivileged account on the remote node. To run DTR on an account other than the default, you associate the account with the DTR object.

/PRINT
Instructs DTS to print (or log) test results.

/NOPRINT
Disables printing of test results. The default is /NOPRINT.

/SPEED=number
Indicates the line speed (bits per second) of the test line. DTS uses this qualifier for reporting statistics (DTS does not control line speeds). The default is /SPEED=1000000.

Command- Specific Qualifiers

/TYPE=test-type
Indicates the subtest for the connection test. There are two subtests:

ACCEPT	Indicates a connect accept test. The default is /TYPE=ACCEPT.
REJECT	Indicates a connect reject test.

/RETURN=return-option
Indicates the type of optional user data DTR returns. There are two possibilities:

STANDARD	Indicates standard user data.
RECEIVED	Indicates received user data.

/NORETURN
Disables return of user data. The default is /NORETURN.

Description

Connection tests evaluate the ability of the DECnet software to process connect, connect accept, and connect reject requests with and without optional user data. Connection tests that you may perform include the following:

- Connect reject without user data
- Connect accept without user data
- Connect reject with 16 bytes of standard user data
- Connect accept with 16 bytes of standard user data

- Connect reject with received user data used as reject user data
- Connect accept with received user data used as accept user data

Examples

1. `_Test: CONNECT/TYPE=ACCEPT/RETURN=RECEIVED`

   This command invokes the connection test with the connect accept subtest. DTS attempts to connect to DTR on the local node (by default). Connect user data is to be returned as part of the test.

2. `_Test: CONNECT/NODENAME=TRNTO`

   This command invokes the connection test with the connect accept subtest. DTS attempts to connect to DTR on remote node TRNTO. DTR returns no optional user data.

DATA

Invokes a data test.

Format

DATA [/general-qualifier[,...] /command-specific-qualifier[,...]]

General Qualifiers

/NODENAME=node-id
Identifies the node (by name or address) on which DTR is to run. The default is the local node. Because you cannot use access control information as part of the node name, DTR runs on the default nonprivileged account on the remote node. To run DTR on an account other than the default, you must associate the account with the DTR object.

/PRINT
Instructs DTS to print (or log) test results.

/NOPRINT
Disables printing of test results. The default is /NOPRINT.

/STATISTICS
Instructs DTS to print statistics on data and interrupt tests. The default is /STATISTICS.

/NOSTATISTICS
Disables the output of statistics.

/DISPLAY=number
Instructs DTS to print the specified number of bytes (in hexadecimal) of data and interrupt messages transmitted to DTR.

/NODISPLAY
Disables the display of data and interrupt messages that are transmitted to DTR. The default is /NODISPLAY.

/SPEED=number
Indicates the line speed (bits per second) of the test line. DTS uses this qualifier for reporting statistics. (DTS does not control line speeds.) The default is /SPEED=1000000.

Command- Specific Qualifiers

/TYPE=test-type
Indicates the subtest for the data test. There are four subtests:

SINK Indicates a sink test. The default is /TYPE=SINK.

SEQ Indicates a sequence test.

PAT Indicates a pattern test.

ECHO Indicates an echo test.

/SIZE=number
Indicates the data message length in bytes. The lower bounds are 0 for a sink or echo test; 4 for a sequence test; and 5 for a pattern test. The upper bound is 4096 bytes for all tests. The default is /SIZE=128.

/[test-duration]
Indicates the duration of the test. There are three units of time for this parameter:

/SECONDS=number
/MINUTES=number
/HOURS=number

The number specifies the duration of the test. The default unit is /SECONDS=30. The maximum test duration is 3,600,000 seconds or 1000 hours.

Description

Data tests provide a full spectrum of test capabilities, ranging from data sink operations to data integrity checking. Data tests that you may perform include the following:

- Sink Test. DTR ignores all data received during this test. DTR performs neither sequence nor content validation.
- Sequence Test. Data messages transmitted by DTS to DTR include a 4-byte sequence number. During the sequence test, if a message is received out of sequence, DTR aborts the logical link and the test.
- Pattern Test. Data messages transmitted to DTR have both a sequence number and a standard data pattern. During the pattern test, if either the sequence number or the received data does not match the expected data, DTR aborts the logical link and the test.
- Echo Test. DTR transmits all data messages received back to DTS during this test. DTS performs data validity tests.

Example

```
_Test:   DATA/PRINT/TYPE=SEQ/SIZE=128/SECONDS=10

         _DTS-S-NORMAL, normal successful completion

         Test parameters:
          Test duration (sec) 10
          Target nodename         ""
          Line speed (baud) 1000000
          Message size (bytes) 128

         Summary statistics:
          Total messages XMIT 788     RECV    0
          Total bytes XMIT 100864
          Messages per second 78.8
          Bytes per second 10086
          Line throughput (baud) 80691
          _ Line utilization 8.0
```

This command invokes the data test with the sequence subtest. DTS sends messages to DTR on the local node where test information is to be printed. The message size is 128 bytes, and the duration of the test is 10 seconds.

DISCONNECT

Invokes a disconnection test.

Format

DISCONNECT [/general-qualifier[,...] /command-specific-qualifier[,...]]

General Qualifiers

/NODENAME=node-id
Identifies the node (by name or address) on which DTR is to run. The default is the local node. Because you cannot use access control information as part of the node name, DTR runs on the default nonprivileged account on the remote node. To run DTR on an account other than the default, you must associate the account with the DTR object.

/PRINT
Instructs DTS to print (or log) test results.

/NOPRINT
Disables printing of test results. The default is /NOPRINT.

Command- Specific Qualifiers

/TYPE=test-type
Indicates the subtest for the disconnection test. There are two subtests:

SYNCHRONOUS	Indicates a synchronous disconnect test.
ABORT	Indicates a disconnect abort test. The default is /TYPE=ABORT.

/RETURN=return-option
Indicates the type of optional user data DTR returns. There are two possibilities:

STANDARD	Indicates standard user data.
RECEIVED	Indicates received user data.

/NORETURN
Disables return of user data. The default is /NORETURN.

Description

Disconnection tests evaluate whether DTS detects the difference between disconnection and abort sequences generated by DTR as well as the ability of DTS to receive the proper optional user data. Disconnection tests that you can perform include the following:

- Disconnect without user data
- Abort without user data
- Disconnect with 16 bytes of standard user data
- Abort with 16 bytes of standard user data
- Disconnect with received connect user data used as disconnect user data
- Abort with received connect user data used as abort user data

Example

```
_Test:   DISCONNECT/NODENAME=DENVER/TYPE=ABORT
```

This command invokes the disconnection test with the disconnect abort subtest. DTS performs this test with DTR on remote node DENVER. No optional user data is to be returned.

HELP

HELP

The HELP command displays general information about DTS commands and parameters.

Format

HELP [topic]

Command Parameter

topic

Specifies DTS and DTR keyword that refers to a topic or subtopic on which you can obtain information.

Example

```
_Test:   HELP

  Information available:

  CONNECT    DATA       DISCONNECT DTS        INTERRUPT

Topic? dts

DTS

     DTS and DTR are cooperationg tasks that perform various functions to
     exercise network software. DTR functions as a slave to DTS. DTS
     initiates each test by issuing a connection request to DTR. DTS
     passes parameters pertinent to the type of test request to DTR in
     the optional data of the connection request. The user interface to
     DTS consists of a group of commands:

        CONNECT
        DATA
        DISCONNECT
        INTERRUPT

     You may abbreviate command keywords and qualifiers to four characters
     or less (if unique).

     Note that, when using DTS commands in a batch job, you must use uppercase
     characters in the command.

Topic?
```

This command invokes HELP and displays information about DTS.

INTERRUPT

Invokes an interrupt test.

Format

INTERRUPT [/general-qualifier[,...] /command-specific-qualifier[,...]]

General Qualifiers

/NODENAME=node-id
Identifies the node (by name or address) on which DTR is to run. The default is the local node. Because you cannot use access control information as part of the node name, DTR runs on the default nonprivileged account on the remote node. To run DTR on an account other than the default, you must associate the account with the DTR object.

/PRINT
Instructs DTS to print (or log) test results.

/NOPRINT
Disables printing of test results. The default is /NOPRINT.

/STATISTICS
Instructs DTS to print statistics on data and interrupt tests. The default is /STATISTICS.

/NOSTATISTICS
Disables the output of statistics.

/DISPLAY=number
Instructs DTS to print data and interrupt messages transmitted to DTR.

/NODISPLAY
Disables the display of data and interrupt messages that are transmitted to DTR. The default is /NODISPLAY.

/SPEED=number
Indicates the line speed (bits per second) of the test line. DTS uses this qualifier for reporting statistics. (DTS does not control line speeds.) The default is /SPEED=1000000.

Command- Specific Qualifiers

/TYPE=test-type
Indicates the subtest for the interrupt test. There are four subtests:

SINK	Indicates a sink test. The default is /TYPE=SINK.
SEQ	Indicates a sequence test.
PAT	Indicates a pattern test.
ECHO	Indicates an echo test.

INTERRUPT

/SIZE=number
Indicates the data message length in bytes. The lower bounds are 0 for a sink or echo test; 4 for a sequence test; and 5 for a pattern test. The upper bound is 16 bytes for all tests. The default is /SIZE=16.

/[test-duration]
Indicates the duration of the test. There are three units of time for this parameter:

/SECONDS=number
/MINUTES=number
/HOURS=number

The number specifies the duration of the test. The default unit is /SECONDS=30. The maximum test duration is 3,600,000 seconds or 1000 hours.

Description

As with data tests, interrupt tests provide a full range of test capabilities from data sink operations to data integrity checking. Interrupt tests that you can perform include the following:

- Sink Test. DTR ignores all interrupt data received during this test. DTR performs neither sequence nor content validation.
- Sequence Test. Interrupt messages transmitted by DTS to DTR include a 4-byte sequence number. During the sequence test, if a message is received out of sequence, DTR aborts the logical link and the test.
- Pattern Test. Interrupt messages transmitted to DTR have both a sequence number and a standard data pattern. During the pattern test, if either the sequence number or the received data does not match the expected data, DTR aborts the logical link and the test.
- Echo Test. DTR transmits all interrupt messages received back to DTS during this test. DTS performs data validity tests.

Example

```
_Test:   INTERRUPT/NODENAME=DALLAS/PRINT/TYPE=PAT/SIZE=16

         _DTS-S-NORMAL, normal successful completion

         Test parameters:
          Test duration (sec) 30
          Target nodename   "DALLAS"
          Line speed (baud) 1000000
          Message size (bytes) 16

         Summary statistics:
          Total messages XMIT 2734 RECV 0
          Total bytes XMIT 43744
          Messages per second 91.1
          Bytes per second 1458
          Line throughput (baud) 11665
          __ Line utilization 1.1
```

This command invokes the interrupt test with the pattern subtest. DTS sends interrupt messages to DTR on node DALLAS where test information is to be printed. The interrupt message size is 16 bytes, and the duration of the test is 30 seconds (by default).

A

DECnet Circuit and Line Devices

This appendix lists DECnet for OpenVMS circuit and line devices. NCP allows for management of the local node and for remote management of any DECnet node that you specify as the **executor**. The display that you see and the devices that you manage depend on the supported features of the particular DECnet system that you are managing.

Note that some of the devices shown may not be supported on the your system. Check the Software Product Description for the latest device information.

Table A–1 shows the devices for Ethernet circuits and lines. Devices on this table conform to the Ethernet and/or IEEE 802.3 protocol.

Table A–1 Ethernet Circuit and Line Devices

Mnemonic	Driver	Description
BNA	ET	DEBNA, DEBNI communications link
ISA	EZ	SGEC communications link
KFE	EF	VAXft 3000 communications link
MNA	EX	DEMNA communications link
MXE	EC	PMAD communications link
QNA	XQ	DEQNA, DELQA, DESQA, DEQTA communications link
SVA	ES	DESVA communications link
UNA	XE	DEUNA, DELUA communications link

VAX

Table A–2 shows the devices for DDCMP and CI circuits and lines. DDCMP devices in this table conform to the DDCMP protocol. CI devices communicate using protocols defined by the System Communications Architecture (SCA).

Table A–2 DDCMP and CI Circuit and Line Devices

Mnemonic	Driver	Description
DIV	SD	DIV32, ISDN controller
DMB	SI	DMB32 synchronous line unit
DMC	XM	DMC11, DMR11 synchronous line units
DMF	XG	DMF32 synchronous line unit
DMP	XD	DMP11, DMV11 synchronous point-to-point or multipoint line device
DSB	SL	DSB32 synchronous line unit

(continued on next page)

Table A–2 (Cont.) DDCMP and CI Circuit and Line Devices

Mnemonic	Driver	Description
DSF	SF	VAXft 3000 synchronous communications controller
DST	ZS	DST32, DSH32 synchronous device
DSV	SJ	DSV11 synchronous line interface
DSW	ZT	DSW-21, DSW-41, DSW-42 synchronous communication device
TT	NO	DZ11, DZ32, DZQ11, DZV11 asynchronous device
TX	NO	DMB32, DHB32, DHU11, DHV11, DHT32, DMF32 or DMZ32, DHQ11, DSH32,CXY08 asynchronous device
CI	CN	CI–780, CI–750, CIBCA, or CIBCI computer interconnect ♦

Table A–3 shows the devices for X.25 circuits and lines. Devices on this table conform to the LAPB protocol.

Note

Although some systems do not support VAX P.S.I., you can manage remote VAX P.S.I. systems from the local node.

Table A–3 X.25 Circuit and Line Devices

Mnemonic	Driver	Description
DMB	SI	DMB32 synchronous line unit
DMF	XG	DMF32 synchronous line unit
DPV	UE	DPV11–DA synchronous line interface
DSB	SL	DSB32 synchronous line unit
DSV	SJ	DSV11 synchronous line interface
DSW	ZT	DSW-21, DSW-41, DSW-42 synchronous communication devices
DUP	XW	DUP11–DA synchronous line interface
KMV	UV	KMV1A synchronous line interface combined with X.25 level 2 microcode
KMX	XS	KMS11–BD or –BE synchronous line interface combined with X.25 level 2 microcode
KMY	XN	KMS1P synchronous line interface combined with X.25 level 2 microcode

Table A–4 shows the devices for FDDI circuits and lines. Devices on this table conform to the FDDI protocol.

Table A–4 FDDI Circuit and Line Devices

Mnemonic	Driver	Description
FZA	FC	DEFZA FDDI communications link
MFA	FX	DEMFA FDDI communications link

B

Network Counter Summary

B.1 Overview

This appendix includes descriptions of circuit, line and node counters. Where possible, the description of each counter includes the probable causes for the particular type of occurrence that causes the counter to be incremented. In some cases, the counters correspond to network events. The events and event descriptions provide additional information relative to the specific occurrence. The individual counter descriptions state which counters correspond to events.

NCP allows for management of the local node and for remote management of any DECnet node that you specify as the executor. The displays, counters, and messages that you receive by issuing NCP commands originate at the executor node. This appendix shows all the information that DECnet for OpenVMS can display. The actual display you receive depends on the supported features of the particular DECnet system that you are managing.

To see what management features are supported on DECnet for OpenVMS, see Chapter 3.

NCP reports the contents of all counters in decimal notation. Counter content displays with an angle bracket (>) indicate that the counter has overflowed. In some cases where the counter has more than one error associated with it, the standard counter is expanded to include a reason.

The following example illustrates the display of circuit counters:

```
NCP>SHOW KNOWN CIRCUITS COUNTERS

Known Circuit Counters as of 12-JUN-1992 06:45:10

Circuit = FZA-0

      >65534  Seconds since last zeroed
      695803  Terminating packets received
     1847598  Originating packets sent
           0  Terminating congestion loss
      201179  Transit packets received
      200122  Transit packets sent
           0  Transit congestion loss
           0  Circuit down
           0  Initialization failure
           0  Adjacency down
           1  Peak adjacencies
     2057126  Data blocks sent
  1975840666  Bytes sent
      903317  Data blocks received
    32194824  Bytes received
           0  Unrecognized frame destination
           0  User buffer unavailable


Circuit = SVA-0

      >65534  Seconds since last zeroed
           0  Terminating packets received
           0  Originating packets sent
           0  Terminating congestion loss
           0  Transit packets received
           0  Transit packets sent
           0  Transit congestion loss
           0  Circuit down
           0  Initialization failure
           0  Adjacency down
           0  Peak adjacencies
           0  Data blocks sent
           0  Bytes sent
           0  Data blocks received
           0  Bytes received
           0  Unrecognized frame destination
           0  User buffer unavailable
```

This example illustrates the display of line counters:

```
NCP>SHOW KNOWN LINE COUNTERS

Known Line Counters as of 12-JUN-1992 06:42:51

Line = FZA-0

      >65534  Seconds since last zeroed
      879508  Data blocks received
       13438  Multicast blocks received
           0  Receive failure
    55687201  Bytes received
     1846203  Multicast bytes received
           0  Data overrun
     1972648  Data blocks sent
        9386  Multicast blocks sent
  1845189341  Bytes sent
     1058642  Multicast bytes sent
           0  Send failure
           0  Unrecognized frame destination
           0  System buffer unavailable
           0  User buffer unavailable
 >4294967294  MAC frame count
           0  MAC error count
           0  MAC lost count
           1  Ring initializations initiated
           2  Ring initializations received
           1  Ring beacons initiated
           0  Duplicate address test failures
           0  Duplicate tokens detected
           0  Ring purge errors
           0  FCI strip errors
           0  Traces initiated
           0  Traces received
           0  Directed beacons received
           0  Elasticity buffer errors
           0  LCT rejects
           0  LEM rejects
           0  Link errors
           2  Connections completed


Line = SVA-0

      >65534  Seconds since last zeroed
      103374  Data blocks received
       71226  Multicast blocks received
           0  Receive failure
    10714578  Bytes received
     8119764  Multicast bytes received
           0  Data overrun
       52938  Data blocks sent
       23861  Multicast blocks sent
         436  Blocks sent, multiple collisions
         329  Blocks sent, single collision
        2201  Blocks sent, initially deferred
     4990338  Bytes sent
     2711186  Multicast bytes sent
           0  Send failure
           0  Collision detect check failure
           0  Unrecognized frame destination
           1  System buffer unavailable
           0  User buffer unavailable
```

B.2 Circuit Counters

This section contains a description of each circuit counter.

Adjacency down

This counter indicates a loss of connection to an adjacent node that may result from any of the following:

- Node listener timeout
- Invalid data received at a node listener
- Unexpected control (initialization or verification) message received
- Adjacency idled
- Routing message received with a checksum error
- Node identification from a routing message or a Hello message that is not the one expected
- Hello message received indicating that connectivity became one-way

Bytes received

This counter indicates the number of bytes of data received by the local node over the circuit. You can use this information together with the data blocks received counter to determine the inbound traffic load.

Bytes sent

This counter indicates the number of bytes of data sent by the local node over the circuit. You can use this information together with the data blocks sent counter to determine the outbound traffic load.

Circuit down

This counter indicates the number of failures—operator or software induced—for the circuit. These failures may include any number of hardware- software- or operator-caused problems. This counter corresponds to events 4.7–9 (circuit down).

Data blocks received

This counter indicates the number of data blocks received by the local node. You can use this information as a statistical base when evaluating the number of inbound data errors, remote reply timeouts, and local buffer errors.

Data blocks sent

This counter indicates the number of data blocks sent by the local node. You can use this information as a statistical base when evaluating the number of outbound data errors, local reply timeouts, and remote buffer errors.

Data errors inbound

This counter indicates the number of data errors that normally result from errors on the inbound communications channel to the local node. These errors are caused usually by a noisy communications line or a poorly functioning modem. This counter may include either or both of the following qualifiers if they contribute any of the following errors:

NAKs sent, header block check error
NAKs sent, data field block check error

Data errors outbound

This counter indicates the number of data errors that normally result from errors on the outbound communications channel from the local node. These errors are caused usually by a noisy communications line or a poorly functioning modem. This counter represents the sum of all NAKs received.

Initialization failure

This counter indicates the number of times the circuit failed to initialize with remote Routing software. These failures may include any number of hardware-software- or operator-caused problems. This counter corresponds to events 4.11–13 (initialization failure).

Local buffer errors

This counter indicates the number of local buffer errors that normally result from a local receive buffer that is too small. These errors are caused when a remote node sends a block that is too big for the local node's receive buffer. This counter may include the following qualifier if it contributes to the error:

NAKs sent, buffer unavailable

This counter corresponds to event 5.9 (local buffer too small) and event 4.3 (oversized packet loss).

Local reply timeouts

This counter indicates the number of timeouts that normally result from the following:

- The loss of communication between nodes while the local node has data to transmit
- An inappropriate value set for the reply timer on the local node

Locally initiated resets

This counter indicates the number of resets sent over the circuit.

Network initiated resets

This counter indicates the number of resets originated by the PSDN received over the circuit.

Originating packets sent

This counter indicates the number of data packets sent by the Routing layer on the local node.

Peak adjacencies

This counter indicates the maximum number of nodes that have been up on a circuit.

Remote buffer errors

This counter indicates the number of remote buffer errors that normally result from a remote receive buffer that is too small.

Remote reply timeouts

This counter indicates the number of timeouts that normally result from the following:

- The loss of communication between nodes while the remote node has data to transmit
- An inappropriate value set for the reply timer on the remote node

Remotely initiated resets

This counter indicates the number of resets received over the circuit.

Seconds since last zeroed

This counter indicates the number of seconds that have elapsed since the circuit counters were zeroed. This counter provides a time frame for other counter values. The software increments this counter every second and clears it when other counters are cleared.

Selection intervals elapsed

This counter indicates the number of selection intervals for half-duplex and multipoint control stations. This is the number of times a station selects another station. You can use this information as a statistical base when evaluating selection timeout errors.

Selection timeouts

This counter indicates the number of selection timeouts that normally result from the following:

- Loss of communication with a remote station.
- Data errors on the communications channel to or from that station.
- An inappropriate value for the station's select timer. This counter corresponds to event 5.5 (select error threshold) and may include either the *No reply to select* or the *Incomplete reply to select* qualifiers if they contribute to the errors.

Terminating congestion loss

This counter indicates the number of packets intended for the node that were discarded because Routing could not buffer them.

Terminating packets received

This counter indicates the number of data packets received by the Routing layer on the local node.

Transit congestion loss

This counter indicates the number of transit data packets discarded for congestion reasons. This counter is maintained only on full-routing nodes. If congestion loss increases, increase the MAXIMUM BUFFERS parameter for the local node.

Transit packets received

This counter indicates the number of data packets received over the circuit and to be routed through the local node to another node. It is maintained only on full-routing nodes.

Transit packets sent

This counter indicates the number of data packets sent over the circuit and being routed through the local node to another node. It is maintained only on full-routing nodes.

Unrecognized frame destination

This counter indicates the number of times a frame was discarded because the frame was addressed to a multicast address not enabled on the circuit.

User buffer unavailable

This counter indicates the total number of times that no user buffer was available for an incoming frame that passed all filtering. User buffers are supplied by users on receive requests.

B.3 Line Counters

This section contains a description of each line counter. The counters in this section may be displayed for devices other than the DMC11 on remote nodes other than DECnet for OpenVMS.

Blocks sent, initially deferred

This counter indicates the total number of times a frame transmission was deferred on its first transmission attempt. Used in measuring Ethernet contention with no collisions.

Blocks sent, multiple collisions

This counter indicates the total number of times a frame was successfully transmitted on the third or later attempt after normal collisions on previous attempts.

Blocks sent, single collision

This counter indicates the total number of times a frame was successfully transmitted on the second attempt after a normal collision on the first attempt.

Bytes received

This counter indicates the number of bytes of data received over the line.

Bytes sent

This counter indicates the number of bytes of data sent over the line.

Collision detect check failure

This counter indicates the approximate number of times a collision detect was not sensed after a transmission. This counter is also reported each time a packet is transmitted using either a DEUNA or DELUA on a broadband Ethernet.

Connections completed

An FDDI counter that indicates the number of times the PHY Port entered the In Use state, after having completed the initialization process.

Data blocks received

This counter indicates the number of data blocks received over the line.

Data blocks sent

This counter indicates the number of data blocks sent over the line.

Data errors inbound

This counter indicates the number of incoming data errors that result from faults on the channel between the local DTE and DCE. The counter can include any of the following qualifiers:

Block too long
Block check error
Reject sent

Data errors outbound

This counter indicates the number of outgoing data errors that result from faults on the channel between the local DTE and DCE. The counter can include the following qualifier:

Reject received

Data overrun

This counter indicates the total number of times the hardware lost an incoming frame because it was unable to keep up with the data rate.

Directed beacons received

An FDDI counter that indicates the number of times the link received a unique directed beacon.

Duplicate address test failures

An FDDI counter that indicates the number of times the duplicate address test detected that the link address was a duplicate.

Duplicate tokens detected

An FDDI counter that indicates the number of times the media access control (MAC) detected a duplicate token either by means of the duplicate token-detection algorithm or by the receipt of a token while already holding one.

Elasticity buffer errors

An FDDI counter that indicates the number of times the Elasticity Buffer function in the PHY Port had an overflow or underflow. This indicates a transmit clock error somewhere on the ring.

FCI strip errors

An FDDI counter that indicates the number of times the receipt of a token terminated a Frame Content Independent Strip.

Local buffer errors

This counter indicates the number of receive-not-ready (RNR) frames sent. The counter can include the following qualifier:

RNR sent, buffer unavailable

Local process errors

This counter indicates errors for both VAX P.S.I. and DDCMP lines. For VAX P.S.I. operations, this counter indicates that a frame reject (FRMR) has been received over the line or that your system is being overloaded. The counter can include any of the following qualifiers:

Transmit underrun
Receive overrun
FRMR received, header format error

The first two qualifiers usually indicate that the system is overloaded and the third usually indicates that the VAX P.S.I. software may be incorrectly configured.

For DDCMP operations, this counter indicates the number of local DDCMP processing errors. DECnet for OpenVMS maintains this counter only for DMP and DMF lines. For a multipoint circuit, this counter reflects the total errors for the control station, and not for each tributary. These errors may be caused by a programming error at the local node, or by a circuit error that was not detected by the Data Link protocol. This counter may include any of the following qualifiers if they contribute to the errors:

NAKs sent, receive overrun
Receive overrun, NAK not sent
Transmit overruns
NAKs received, header format error

For DDCMP lines, this counter corresponds to events 5.3–4 and 5.6.

Local reply timeouts

This counter indicates the number of times a frame with a poll bit set has been received over the line, that is, the number of errors that result from faults on the line.

LCT rejects

An FDDI counter that indicates the number of times a connection on this PHY Port was rejected because the Link Confidence Test (LCT) at either end of the physical connection failed. The LCT rejects counter only counts rejections that cause the link to enter into the Watch State. The counter, therefore, indicates the number of *distinct* link quality problems rather than the total length of time such problems persisted.

LEM rejects

An FDDI counter that indicates the number of times an active connection on this PHY Port was disconnected because the Link Error Monitor at this end of the physical connection rejected the connection, or because the Noise timer (TNE) expired.

Link errors

An FDDI counter that indicates the total number of *raw* Link Error input events detected by the Link Error Monitor.

MAC error count

An FDDI counter that indicates the total number of times the MAC Layer changed the error indicator in a frame from reset to set. This tells you the number of times the local FDDI adapter detected an error in a frame.

MAC frame count

An FDDI counter that indicates the total number of frames on the FDDI media, other than tokens.

MAC lost count

An FDDI counter that indicates the total number of times a frame (other than a token) was improperly terminated.

Multicast blocks received

This counter indicates the total number of multicast blocks successfully received.

Multicast bytes received

This counter indicates the total number of multicast data bytes successfully received (includes bytes in Ethernet data field but not the Ethernet data link headers).

Multicast blocks sent

This counter indicates the total number of multicast blocks successfully sent.

Multicast bytes sent

This counter indicates the total number of multicast data bytes successfully sent (including bytes in the Ethernet data field, but not in the Ethernet data link headers).

Receive failure

This counter indicates the total number of blocks received with some data error (the blocks are data frames) that passed either physical or multicast address comparison. For each increment of the counter, a type of failure is recorded, as follows:

Block check error
Framing error
Frame too long

Remote buffer errors

This counter indicates the number of receive-not-ready (RNR) frames received. The counter can include the following qualifier:

RNR received, buffer unavailable

Remote process errors

This counter indicates errors for both VAX P.S.I. and DDCMP lines. For VAX P.S.I. operations, this counter indicates an invalid n(R) and a frame reject (FRMR) sent over the line. The counter can include the following qualifiers:

Invalid n(R) received
FRMR sent, header format error

These errors usually indicate that the DCE is functioning incorrectly.

For DDCMP operations, this counter indicates the number of remote DDCMP processing failures. DECnet for OpenVMS maintains this counter only for DMP and DMF lines. For a multipoint circuit, this counter reflects the total errors for the control station, and not for each tributary. These errors may be caused by a programming error or a hardware malfunction at the adjacent node. They could also be caused by a circuit error that was not detected by the Data Link protocol. This counter may include any of the following qualifiers if they contribute to these errors:

NAKs received, receive overrun
NAKs sent, header format error
Selection address errors
Streaming tributaries

For DDCMP lines, this counter corresponds to events 5.4 and 5.6–8.

Ring beacons initiated

An FDDI counter that indicates the number of times this station initiated the ring beacon process.

Ring initializations initiated

An FDDI counter that indicates the number of times this station initiated a ring reinitialization.

Ring initializations received

An FDDI counter that indicates the number of times another station initiated ring reinitialization.

Ring purge errors

An FDDI counter that indicates the number of times the ring purger received a token while still in the ring purge state.

Remote reply timeouts

This counter indicates the number of times the retransmit timer for that line has expired. Either the line is not connected to a modem, the PSDN is not responding fast enough, or the retransmit timer is set too low.

Seconds since last zeroed

This counter indicates the number of seconds that elapsed since the line counters were zeroed. This counter provides a time frame for other counter values. The software increments this counter every second and clears it when other counters are cleared.

Send failure

This counter indicates the total number of times a transmit attempt failed. For each increment of the counter, a type of failure is recorded, as follows:

Excessive collisions
Carrier check failed
Short circuit
Open circuit
Frame too long
Remote failure to defer

The ‘‘short circuit’’ failure is also reported each time a packet is transmitted using a DEQNA with a DECOM transceiver.

System buffer unavailable

This counter indicates the total number of times no system buffer was available for an incoming frame. This can be any buffer between the hardware and the user buffers (those supplied on receive requests).

Traces initiated

An FDDI counter that indicates the number of times this link initiated the PC-Trace process.

Traces received

An FDDI counter that indicates the number of times another link initiated the PC-Trace process.

Unrecognized frame destination

This counter indicates the number of times a frame was discarded because there was no portal with the protocol enabled. The count includes frames received for the physical address only. It does not include frames received for the multicast or broadcast address.

User buffer unavailable

This counter indicates the total number of times no user buffer was available for an incoming frame that passed all filtering. The user buffer is one supplied by the user on a receive request.

B.4 Node Counters

This section contains a description of each node counter.

Aged packet loss

This counter indicates the number of data packets discarded for visiting too many nodes. This usually occurs while the nodes throughout the network are recovering from a disruption (for example, when a circuit or line goes down) in the former path to a destination. This counter is maintained only on routing nodes and corresponds to event 4.0 (aged packet loss).

Buffer unavailable

This counter indicates the number of data segments transferred from the remote node that were discarded because of insufficient cache buffering.

Bytes received

This counter indicates the number of user data bytes received from a remote node. This includes interrupt messages, but excludes Connect, Accept, Reject, and Disconnect messages.

Bytes sent

This counter indicates the number of user data bytes sent to a remote node.

Connects received

This counter indicates the number of logical link connection requests received by the local node.

Connects sent

This counter indicates the number of logical link connection requests sent by the local node.

Maximum logical links active

This counter indicates the largest number of logical links that have been active since DECnet for OpenVMS software was started or since executor counters were zeroed.

Messages received

This counter indicates the number of NSP messages received from a remote node.

Messages sent

This counter indicates the number of NSP messages sent to a remote node.

Node out-of-range packet loss

This counter indicates the number of data packets discarded because the destination node's address is greater than the maximum address defined for the local node. This counter corresponds to event 4.2 (node out-of-range packet loss).

Node unreachable packet loss

This counter indicates the number of data packets lost because the destination node could not be accessed. This counter is maintained only on routing nodes. This counter corresponds to event 4.1 (node unreachable packet loss).

Oversized packet loss

This counter indicates the number of received data packets that were too large to forward because of the block size of the data link that would be used. This counter is maintained only on routing nodes, and corresponds to event 4.3 (oversized packet loss).

Packet format error

This counter indicates the number of packet format errors that occur because of invalid packet control information. This counter corresponds to event 4.4 (packet format error).

Partial routing update loss

This counter indicates the number of received routing messages that were too long to process. Part of a routing update may be lost if it contains a reachable node with an address greater than the maximum address defined for the local node. This counter is maintained only on routing nodes, and corresponds to event 4.5 (partial routing update loss).

Received connect resource errors

This counter indicates the number of inbound connect messages for which the local node did not have sufficient resources. These errors may result from dynamic memory problems or too few logical link slots; that is, the MAXIMUM LINKS parameter value is too small.

Response timeouts

This counter indicates the number of times there was no response to an NSP segment within the allotted timeout period. This implies that the local node is having to retransmit messages. Such retransmission can be caused either by messages being discarded in the network or by a wide variance in the round-trip delay to the node. Normally, it indicates an overload condition in the network.

Seconds since last zeroed

This counter indicates the number of seconds that have elapsed since the node counters were zeroed. It provides a time frame for other counter values. The software increments this counter every second and clears it when other counters are cleared.

Verification reject

This counter indicates the number of received verification messages that were invalid. It corresponds to event 4.6 (verification reject).

B.5 X.25 Protocol Module DTE Counters

This section contains a description of each X.25 protocol module DTE counter. These counters apply only to VAX P.S.I. operations.

Note

Although some systems do not support VAX P.S.I., you can manage remote VAX P.S.I. systems from the local node.

Bytes received

This counter indicates the number of bytes of data received by the local DTE. You can use this information together with the data blocks received counter to determine the total traffic load.

Bytes sent

This counter indicates the number of bytes of data sent by the local DTE. You can use this information together with the data blocks sent counter to determine the total traffic load.

Calls received

This counter indicates the number of incoming calls received.

Calls sent

This counter indicates the number of outgoing calls made.

Data blocks received

This counter indicates the number of data blocks received by the local DTE.

Data blocks sent

This counter indicates the number of data blocks sent by the local DTE.

Fast selects received

This counter indicates the number of calls received with the fast select facility specified.

Fast selects sent

This counter indicates the number of calls sent with the fast select facility specified.

Locally initiated resets

This counter indicates the number of resets sent by the local DTE.

Maximum channels active

This counter indicates the maximum number of channels from the logical channels list that were active at any one time since the counters were last zeroed.

This counter applies only to those channels defined by the CHANNELS parameter of the SET MODULE X25-PROTOCOL command.

Maximum switched circuits active

This counter indicates the number of switched virtual circuits that were active at any one time since the counters were last zeroed.

Network initiated resets

This counter indicates the number of resets (originated by the PSDN) received by the local DTE.

Received call resource errors

This counter indicates the number of times an incoming call has been rejected because of insufficient resources or an incorrect configuration (for example, no destination or object specified).

Remotely initiated resets

This counter indicates the number of resets (originated by a remote DTE) received by the local DTE.

Restarts

This counter indicates the number of times the restart protocol procedure was used on the DTE.

Seconds since last zeroed

This counter indicates the number of seconds that have elapsed since the module counters were zeroed. This counter provides a time frame for other counter values. The VAX P.S.I. software increments this counter every second and clears it when the counters are zeroed.

B.6 X.25 and X.29 Server Module Counters

This section contains a description of each X.25 and X.29 server module counter. These counters apply only to VAX P.S.I. operations.

Note

Although some systems do not support VAX P.S.I., you can manage remote VAX P.S.I. systems from the local node.

Incoming calls rejected, no resources

This counter indicates the number of times the incoming call handler rejected a request to set up a virtual circuit because of insufficient resources. It includes calls rejected by security for the X.25 server module.

Maximum circuits active

This counter indicates the number of switched virtual circuits that have been set up since the counters were last zeroed.

Seconds since last zeroed

This counter indicates the number of seconds that have elapsed since the module counters were zeroed. This counter provides a time frame for other counter values. The VAX P.S.I. software increments this counter every second and clears it when the counters are zeroed.

C
States and Transitions

C.1 Overview

Each circuit, line, and logical link has sets of states and substates that reflect internal network software activity. The following sections contain descriptions of circuit, line, and link states, and their substates and transitions.This section also describes X25-SERVER module and DTE states, substates, and their transitions. You cannot set or clear these substates, but you can display them by using NCP.

NCP allows for management of the local node and for remote management of any DECnet node that you specify as the executor. The display that you see for the circuit, line, or link state you specify depends on the supported features of the particular DECnet system that you are managing.

To see what management features are supported on DECnet for OpenVMS, see Chapter 3.

C.2 Circuit and Line States and Transitions

Table C–1 lists all possible network management circuit and line states and substates. Table C–2 provides a list of state transitions that take place in response to certain user-initiated NCP functions. Some substates in these tables have an AUTO- prefix. These substates occur when circuit or line service operations take place. Refer to the *DNA Phase IV Network Management Functional Specification* for further information about circuit or line states, substates, and their transitions.

Table C–1 Circuit and Line States and Substates

State	Substate	Meaning
OFF	None	The circuit/line cannot be used.
ON	None	The circuit/line is in normal use by the owner.
	-SYNCHRONIZING	The circuit/line is in the hardware initialization cycle. For X.25 permanent virtual circuits, the circuit is being reset, restarted, or an error has occurred.
	-STARTING	The circuit/line is in the owner routing initialization cycle.
	-REFLECTING	The circuit/line is in use for passive loopback testing.
	-AUTODUMPING	The circuit/line is reserved for upline dumping.

(continued on next page)

Table C–1 (Cont.) Circuit and Line States and Substates

State	Substate	Meaning
	-AUTOSERVICE	The circuit/line is reserved for service use.
	-AUTOLOADING	The circuit/line is in use for automatic loading.
	-AUTOTRIGGERING	The circuit/line is in use for automatic triggering.
	-DUMPING	The circuit/line is in use by the operator for upline dumping.
	-LOADING	The circuit/line is in use by the operator for loading.
	-LOOPING	The circuit/line is in use by the operator for active circuit or line loopback testing.
	-TRIGGERING	The circuit/line is in use by the operator for triggering.
	-FAILED	The DLM SVC has not been able successfully to call the remote DTE, and the maximum recalls limit has been reached.
SERVICE	None	The circuit/line is reserved by the operator for an active service function.
	-REFLECTING	The circuit/line is in use for passive loopback testing.
	-LOADING	The circuit/line is in use by the operator for loading.
	-LOOPING	The circuit/line is in use by the operator for loopback testing.
	-TRIGGERING	The circuit/line is in use by the operator for triggering.

Table C–2 Circuit and Line State Transitions

Old State	New State	Cause of Change
Any	OFF	Operator commands for non-X.25 lines and circuits: SET LINE STATE OFF, SET CIRCUIT STATE OFF
OFF	ON-STARTING	Operator command for non-X.25 lines and circuits: SET LINE STATE ON, SET CIRCUIT STATE ON
	ON-SYNCHRONIZING	Operator command for X.25 lines and circuits: SET LINE STATE ON, SET CIRCUIT STATE ON
	SERVICE	Operator command: SET LINE STATE SERVICE

(continued on next page)

Table C–2 (Cont.) Circuit and Line State Transitions

Old State	New State	Cause of Change
ON	OFF	Operator command: SET LINE STATE OFF and SET CIRCUIT STATE OFF
	ON-STARTING	Routing restarted the data link (from either end).
	ON-SYNCHRONIZING	Hardware is being restarted on either end of the link, or X.25 virtual circuit has been cleared.
	ON-REFLECTING	The local node received a circuit /line loopback message from a remote system.
	ON-AUTOSERVICE	NETACP received a service request.
	ON-LOADING	Operator command: LOAD
	ON-LOOPING	Operator command: LOOP CIRCUIT
	ON-TRIGGERING	Operator command: TRIGGER
	SERVICE	Operator command: SET LINE STATE SERVICE, SET CIRCUIT STATE SERVICE
ON-STARTING	ON	Routing initialization completed.
	ON-REFLECTING	The local node received a circuit /line loopback message from a remote node.
	ON-AUTOSERVICE	NETACP received a service request.
	ON-LOADING	Operator command: LOAD
	ON-LOOPING	Operator command: LOOP CIRCUIT
	ON-TRIGGERING	Operator command: TRIGGER
	SERVICE	Operator command: SET LINE STATE SERVICE, SET CIRCUIT STATE SERVICE
ON-SYNCHRONIZING	None	The circuit/line is attempting to come up.
	OFF	Operator command: SET CIRCUIT STATE OFF
	ON	For an X.25 PVC, the hardware has been restarted and both ends of the PVC have agreed to communicate.
	ON-FAILED	For an outgoing DLM SVC, the MAXIMUM RECALLS limit has been reached and the remote DTE still cannot be called successfully.

(continued on next page)

Table C–2 (Cont.) Circuit and Line State Transitions

Old State	New State	Cause of Change
ON-REFLECTING	ON-SERVICE	A passive circuit/line loopback operation terminated.
	ON-AUTOSERVICE	NETACP received a service request.
	ON-LOADING	Operator command: LOAD
	ON-LOOPING	Operator command: LOOP CIRCUIT
	ON-TRIGGERING	Operator command: TRIGGER
	SERVICE	Operator command: SET LINE STATE SERVICE, SET CIRCUIT STATE SERVICE
ON-AUTOSERVICE	ON-STARTING	NML released the line.
	ON-AUTOLOADING	NML initiated a load operation.
	ON-AUTOTRIGGERING	NML initiated a trigger operation.
ON-AUTOLOADING	ON-AUTOSERVICE	The load operation completed.
ON-AUTOTRIGGERING	ON-AUTOSERVICE	The trigger operation completed.
ON-LOADING	ON-STARTING	The load operation completed.
ON-LOOPING	ON-STARTING	The active circuit/line loopback test completed.
ON-TRIGGERING	ON-STARTING	The trigger operation completed.
SERVICE	SERVICE-SYNCHRONIZING	The circuit/line is attempting to come up in MOP mode.
	SERVICE-REFLECTING	The local node received a circuit /line loopback message from a remote node.
	SERVICE-LOADING	Operator command: LOAD
	SERVICE-LOOPING	Operator command: LOOP CIRCUIT
	SERVICE-TRIGGERING	Operator command: TRIGGER
	OFF	Operator command: SET LINE STATE OFF, SET CIRCUIT STATE OFF
	ON-STARTING	Operator command: SET LINE STATE ON, SET CIRCUIT STATE ON
SERVICE-REFLECTING	SERVICE	A passive circuit/line loopback test completed.
	SERVICE-LOADING	Operator command: LOAD
	SERVICE-LOOPING	Operator command: LOOP CIRCUIT
	SERVICE-TRIGGERING	Operator command: TRIGGER

(continued on next page)

Table C–2 (Cont.) Circuit and Line State Transitions

Old State	New State	Cause of Change
SERVICE-LOADING	SERVICE	The load operation completed.
SERVICE-LOOPING	OFF	Operator command: SET CIRCUIT STATE OFF
	SERVICE	The active circuit/line loopback test completed.
SERVICE-TRIGGERING	SERVICE	The trigger operation completed.

C.3 Link States

Table C–3 lists all possible network management states for logical links. Refer to the *Network Services Protocol Functional Specification* for further information about link states.

Table C–3 Link States

State	Meaning
CC sending	Connect confirm sending. Confirmation of a connection is being sent from the destination node in response to a connect initiate from the originating node.
CI ACKed	Connect initiate acknowledged. The originating node has received a connect confirm from the destination node.
CI received	Connect initiate received. The destination node has received a connect initiate from the originating node.
CI sending	Connect initiate sending. The originating node is sending a connect initiate request to the destination node.
closed	The destination port is not available. The logical link has been disconnected, but the user program has not yet destined the channel to the network device.
DI received	Disconnect initiate received. The destination node has received a disconnect initiate message from the originating node.
DI sending	Disconnect initiate sending. The originating node is sending a disconnect initiate message to the destination node.
run	The logical link may be used for sending and receiving data.

C.4 DTE States and Transitions

Table C–4 lists all possible VAX P.S.I. management states and substates for DTEs. Table C–5 provides a list of DTE state transitions that take place.

Note

Although some systems do not support VAX P.S.I., you can manage remote VAX P.S.I. systems from the local node.

Table C–4 DTE States and Substates

State	Substate	Meaning
OFF	RUNNING	X.25 level 2 and level 3 software is operational but the DTE is not available for use. Incoming calls are cleared.
	SYNCHRONIZING	X.25 level 2 software is operational but level 3 software is not. The DTE is not available for use.
	UNSYNCHRONIZED	X.25 levels 2 and 3 are not operational and the DTE is not available for use.
ON	RUNNING	The DTE is available for normal use.
	SYNCHRONIZING	X.25 level 2 software is operational, level 3 software is starting up, and the DTE will soon be available for use.
	UNSYNCHRONIZED	X.25 level 2 software is starting up and the DTE will soon be available for use.
SHUT	RUNNING	X.25 levels 2 and 3 are operational but the DTE is not to be used for any new activity; that is, all existing virtual circuits will be allowed to complete their operations. Incoming calls are cleared.
	SYNCHRONIZING	X.25 level 2 software is operational and level 3 software is starting up. When the DTE is available for use, no circuits may be established.
	UNSYNCHRONIZED	X.25 level 2 software is starting up. When the DTE is available for use, no circuits may be established.

Table C–5 DTE State Transitions

Old State	New State	Cause of Change
OFF-RUNNING	ON-RUNNING	Operator command: SET MODULE X25-PROTOCOL DTE STATE ON
	OFF-SYNCHRONIZING	X.25 level 3 software is resynchronizing.
	OFF-UNSYNCHRONIZED	X.25 level 2 software is resynchronizing.
OFF-UNSYNCHRONIZED	ON-UNSYNCHRONIZED	Operator command: SET MODULE X25-PROTOCOL DTE STATE ON
	OFF-SYNCHRONIZING	X.25 level 2 startup has completed.
OFF-SYNCHRONIZING	ON-SYNCHRONIZING	Operator command: SET MODULE X25-PROTOCOL DTE STATE ON
	OFF-RUNNING	X.25 level 3 startup has completed.
	OFF-UNSYNCHRONIZED	X.25 level 2 software is resynchronizing.

(continued on next page)

Table C–5 (Cont.) DTE State Transitions

Old State	New State	Cause of Change
ON-RUNNING	OFF-RUNNING	Operator command: SET MODULE X25-PROTOCOL DTE STATE OFF
	SHUT-RUNNING	Operator command: SET MODULE X25-PROTOCOL DTE STATE SHUT
	ON-SYNCHRONIZING	X.25 level 3 software is resynchronizing.
	ON-UNSYNCHRONIZED	X.25 level 2 software is resynchronizing.
ON-UNSYNCHRONIZED	OFF-UNSYNCHRONIZED	Operator command: SET MODULE X25-PROTOCOL DTE STATE OFF
	SHUT-UNSYNCHRONIZED	Operator command: SET MODULE X25-PROTOCOL DTE STATE SHUT
	ON-SYNCHRONIZING	X.25 level 2 startup has completed.
ON-SYNCHRONIZING	OFF-SYNCHRONIZING	Operator command: SET MODULE X25-PROTOCOL DTE STATE OFF
	SHUT-SYNCHRONIZING	Operator command: SET MODULE X25-PROTOCOL DTE STATE SHUT
	ON-RUNNING	X.25 level 3 startup has completed.
	ON-UNSYNCHRONIZED	X.25 level 2 software is resynchronizing.
SHUT-RUNNING	OFF-RUNNING	Operator command: SET MODULE X25-PROTOCOL DTE STATE OFF
	ON-RUNNING	Operator command: SET MODULE X25-PROTOCOL DTE STATE ON
	SHUT-SYNCHRONIZING	X.25 level 3 software is resynchronizing.
	SHUT-UNSYNCHRONIZED	X.25 level 2 software is resynchronizing.
SHUT-UNSYNCHRONIZED	OFF-UNSYNCHRONIZED	Operator command: SET MODULE X25-PROTOCOL DTE STATE OFF
	ON-UNSYNCHRONIZED	Operator command: SET MODULE X25-PROTOCOL DTE STATE ON
	SHUT-SYNCHRONIZING	X.25 level 2 startup has completed.

(continued on next page)

Table C–5 (Cont.) DTE State Transitions

Old State	New State	Cause of Change
SHUT-SYNCHRONIZING	OFF-SYNCHRONIZING	Operator command: SET MODULE X25-PROTOCOL DTE STATE OFF
	ON-SYNCHRONIZING	Operator command: SET MODULE X25-PROTOCOL DTE STATE ON
	SHUT-RUNNING	X.25 level 3 startup has completed.
	SHUT-UNSYNCHRONIZED	X.25 level 2 software is resynchronizing.

C.5 X.25 Server Module States and Transitions

Table C–6 lists all possible VAX P.S.I. management states for the X.25 server module. Table C–7 summarizes state transitions that take place.

Note

Although some systems do not support VAX P.S.I., you can manage remote VAX P.S.I. systems from the local node.

Table C–6 X.25 Server States

State	Meaning
OFF	The module is not in use.
ON	The module is available for normal use.
SHUT	The module is to be closed down but only when all present activity has ceased.

Table C–7 X.25 Server State Transitions

Old State	New State	Cause of Change
OFF	ON	Operator command: SET MODULE X25-SERVER STATE ON
ON	OFF	Operator command: SET MODULE X25-SERVER STATE OFF
	SHUT	Operator command: SET MODULE X25-SERVER STATE SHUT
SHUT	ON	Operator command: SET MODULE X25-SERVER STATE ON
	OFF	Operator command: SET MODULE X25-SERVER STATE OFF

D

Event Class and Type Summary

This appendix includes a summary of NCP events in terms of their class and type. In general, event classes relate to specific layers of the DNA architecture. The event logging components support the event classes summarized.

NCP allows for management of the local node and for remote management of any DECnet node that you specify as the executor. The display that you see for the event that you specify depends on the supported features of the particular DECnet system that you are managing.

To see what management features are supported on DECnet for OpenVMS, see Chapter 3.

Event Class	Description
0	Network Management layer
1	Applications layer
2	Session Control layer
3	End Communications layer
4	Routing layer
5	Data Link layer
6	Physical Link layer
7	X.25 packet level events
8–31	Reserved for other common classes
32–63	RSTS system specific
64–95	RSX system specific
96–127	TOPS 10/20 system specific
128–159	OpenVMS system specific
160–191	RT system specific
192–223	CT system specific
224–255	Communication server specific
256–352	Reserved for future use
353	Digital Distributed Name Service (DECdns) clerk events
354–479	Reserved for future use
480–511	Customer specific

DECnet for OpenVMS logs events only for event classes 0, 3, 4, and 128–159. If you attempt to turn on logging for any other event, it does not get logged. All other classes are supported for logging of processed events from other remote nodes. Events in class 353 are logged only in networks where the Digital Distributed Name Service (DECdns) software is running.

Events in class 7 are logged only on nodes that have VAX P.S.I. installed.

If the logging sink is the monitor and no name is specified, DECnet for OpenVMS uses the OPCOM facility to display event messages on all terminals enabled as NETWORK (by means of REPLY/ENABLE=NETWORK), including the console. Event messages have the following format:

```
event type class.type, event-text
from node address [(node-name)] dd-mmm-yy hh:mm:ss.ms
component-type, event-qualifiers,...
```

The event text is a standard text message for each event class and type. The message format also includes the source node (address and node name, if available) and time stamp for when the event occurred. For most events, the message format includes the component type and name for which the event applies. Finally, the message format may include one or more event qualifiers that provide additional information as to the cause of the event. The following example is a typical message that may be displayed on the console:

```
%%%%%%%%%%  OPCOM  29-JUN-1992 13:49:05.45  %%%%%%%%%%
Message from user DECNET on FALSE
DECnet event 4.15, adjacency up
From node 4.884 (FALSE), 29-JUN-1992 13:49:02.09
Circuit BNA-0, Adjacent node = 4.894 (PASSWD)
```

The information accompanying each event description in Section D.1 includes a brief description of event qualifiers that may be displayed for the event.

Note that OPCOM does not display line or node counters on the terminal device. However, they can be written to the console sink.

D.1 Network Management Layer Events

The following specific event classes and types are supported for the network management layer. Only those events and entire event classes marked with an asterisk (*) are logged by DECnet for OpenVMS components.

0.0 Event records lost*

Events occurred too rapidly for the event logger to buffer them.

0.1 Automatic node counters

A node counter timer expired, thus prompting this event. This message displays the name of the node to which the event applies, along with the node counters for that node.

0.2 Automatic line counters

A line counter timer expired, thereby producing this event. This message displays the name of the line to which the event applies, along with the line counters for that line.

0.3 Automatic circuit service*

An adjacent node requested an automatic circuit service operation. This message displays the name of the circuit to which the event applies, along with the following event qualifiers: the service function performed (load or dump), the status of the operation (requested, successful, or failed), the node address, the file specification, and the software type. If the operation fails, this status includes an NML error message and details.

0.4 Line counters zeroed

Line counters were zeroed. This message displays the name of the line to which the event applies. The event logger logs these counters prior to the execution of a request to zero them.

0.5 Node counters zeroed

Node counters were zeroed. This message displays the name of the node to which the event applies. The event logger logs these counters prior to the execution of a request to zero them.

0.6 Passive loopback*

The software initiated or terminated a passive loopback test on behalf of an adjacent node. This message displays the name of the line to which the event applies, along with one event qualifier that specifies the state of the operation (initiated or terminated).

0.7 Aborted service request*

An adjacent node requested a service operation over a line connected to the local node. However, a problem prevented it from being processed at the local node. This message displays the name of the line to which the event applies, along with one event qualifier that specifies the reason for the failure. The reason may be one of the following:

- Line open error

  DECnet received an MOP message and was unable to acquire control of the line.

- Line state change by higher level

  The line was preempted by a higher priority function. For example, you used NCP to turn off the line.

- Receive error

  A line error occurred while the Network Management layer was trying to receive the request.

- Receive timeout

  The line message receive timer expired before the request could be received from the adjacent node. Either the timer was too short, the line error level was too great for any message to get through, or the adjacent node stopped requesting.

- Unrecognized request

  A message was received but was not recognizable as a request for upline dumping, downline loading, or passive loopback testing. The adjacent node may be running an incompatible version of the line service protocol.

0.8 Automatic counters*

A counter timer for a node, circuit, or line has expired, thereby producing this event. This message displays the name of the component to which the event applies, along with the counters for that component.

0.9 Counters zeroed*

Counters were zeroed for the node, circuit, or line. This message displays the name of the component to which the event applies. The event logger logs these events prior to the execution of a request to zero them.

D.2 Session Control Layer Events

The following specific event classes and types are supported for the session control layer.

2.0 Local node state change

The operational state of the local node changed because of an operator command. Note, however, that the transition from SHUT to OFF also happens automatically when the last logical link is disconnected (under normal operation).

This message displays three event qualifiers: the reason for the state change (operator command or normal operation), the old state (ON, OFF, SHUT, or RESTRICTED), and the new state.

2.1 Access control reject

The local node rejected a connection request because of invalid access control information.

This message displays four event qualifiers: the name and address of the source node, the object type number and process ID of the source process requesting the connection, the object type number and process ID of the destination process to receive the connection request, and the invalid access control information.

D.3 End Communications Layer Events

The following specific event classes and types are supported for the End Communications layer. The Network Services Protocol (NSP) is the protocol that operates within the End Communications layer. Only those events and entire event classes marked with an asterisk (*) are logged by DECnet for OpenVMS components.

3.0 Invalid message*

NSP received a message that could not be interpreted. This may indicate a software malfunction in either the local or remote NSP. This message displays one event qualifier—the NSP message that was invalid. Refer to the *Network Services Protocol Functional Specification* for a description of NSP messages.

3.1 Invalid flow control*

The remote NSP attempted to modify the local flow control value in an invalid manner. This may indicate a software malfunction in either the local or remote NSP. This message displays one event qualifier—the current flow control value. Refer to the *Network Services Protocol Functional Specification* for a description of flow control.

3.2 Database reused

The local node received a connection request from a node for which there is no counter block. All counter blocks have been previously used, and one of the previously used blocks is available for this new node. This results in the loss of node counters for the node that formerly occupied the database entry.

This message displays the name of the node for which the database entry was formerly used, along with the node counters for that node.

D.4 Routing Layer Events

The following specific event classes and types are supported for the Routing layer. Only those events and entire event classes marked with an asterisk (*) are logged by DECnet for OpenVMS components.

4.0 Aged packet loss*

Routing discarded a packet because it had visited too many nodes. This can be a normal occurrence when the network is reconfiguring its routing databases. It can be a failure when the MAXIMUM HOPS value is set too small. This can cause the MAXIMUM VISITS value to be too small for a path that should be usable.

This message displays the name of the line to which the event applies, along with one event qualifier—the packet header. This is information from the beginning of the packet. For non-broadcast packets, it consists of a hexadecimal byte of flags, the decimal destination and source node addresses, and a hexadecimal byte of forwarding data. For broadcast packets, it also includes the physical address of the destination and source, the service type, and the protocol type. Refer to the *Routing Layer Functional Specification* for additional information.

4.1 Node unreachable packet loss*

Routing discarded a packet because the local node found that the destination node was unreachable. This event provides a trace of what has happened to packets that are not reaching their destination.

This message displays the name of the line to which the event applies, along with one event qualifier—the packet header (as described for event 4.0).

4.2 Node out-of-range packet loss*

Routing discarded a packet because the destination node number was greater than the maximum node number known to the local node. Typically, this results from the addition of a new node to the network without increasing the MAXIMUM ADDRESS value on the local node, yet expecting the local node to route packets to that node.

This message displays the name of the line to which the event applies, along with one event qualifier—the packet header (as described for event 4.0).

4.3 Oversized packet loss*

Routing discarded a packet because it was too large to forward to the appropriate adjacent node. Typically, this occurs when the adjacent node's buffer size is too small or when the source node sends a packet that is too large.

This message displays the name of the line over which the packet was to be forwarded, along with one event qualifier—the packet header (as described for event 4.0).

4.4 Packet format error*

Routing discarded a packet because of a format error in the packet header. This usually results from a programming error in the packet formatting by the adjacent node, though it could result from a line error that was not detected by the line protocol.

This message displays the name of the line to which the event applies, along with one event qualifier—the packet beginning. This consists of the first 6 bytes of the packet, displayed as hexadecimal.

4.5 Partial routing update loss*

Routing received a routing message that contained node addresses greater than the maximum address known to the local node. Subsequently, information on these nodes was lost. This occurs when the MAXIMUM ADDRESS value on an adjacent node has been increased to accommodate more nodes, but the local node's has not.

This message displays the name of the line over which this message was received, along with two event qualifiers—the packet header (as described for event 4.0) and the highest node address in the routing update that was lost.

4.6 Verification reject*

An attempt to initialize with another node failed. The local node received an invalid password in the verification requested of the adjacent node during routing initialization over the line. Either the local node expected the wrong receive password, or the adjacent node sent the wrong transmit password.

This message displays the name of the line to which the event applies, along with one event qualifier—the address of the adjacent node that failed to initialize.

4.7 Circuit down, circuit fault*

An error has occurred for the circuit. This message displays the name of the circuit to which the event applies, along with one event qualifier—the reason for the event. The reason could be one of the following:

- Adjacent node address change

  The adjacent node changed addresses without going through the normal initialization sequence. This is also logged when an adjacent node attempts to initialize with the local node, but the address of the adjacent node is not in the database.

- Adjacent node address out of range

  The adjacent node's address is greater than the maximum address defined for the local node. This may be caused by an incorrectly defined node address or by a failure to update the local node's database when a new node was added.

- Adjacent node block size too small

  The line block size provided by the adjacent node is too small for normal network operation. The block size may be set incorrectly at the adjacent node.

- Adjacent node listener receive timeout

  The node has received no message over the data link within the last 30 seconds. This usually means that the remote node is not running.

- Adjacent node listener received invalid data

  A test message sent by the adjacent node contained invalid or corrupted data. This is most likely caused by a hardware problem.

- Call failed

  An outgoing SVC call failed. This is an X.25 event.

- Data errors

  The line was declared down by the line protocol of the local node handler when the line exceeded an error threshold.

- Dropped by adjacent node

  The adjacent node was responsible for breaking the circuit connection.

- Invalid verification seed value

  A Routing initialization message sent by an adjacent node is not formatted properly. This is most likely caused by a remote network software problem.

- Line synchronization lost

The normal line protocol was restarted or terminated by the adjacent node. Either a line exceeded an error threshold, or network management initiated a line state change. DMR/DMC failures that cause a line synchronization error are as follows:

- Threshold errors, including more than eight attempts to transmit a message, or eight NAKs received in a row.
- Start message received in the ON state (that is, the remote system detected an error and restarted the line).
- Maintenance requested while in the ON state (that is, the remote system tried to perform a maintenance operation such as LOOP CIRCUIT).
- Message was lost because no buffer was available in CPU memory.
- Nonexistent memory error.
- Procedure error, because of driver failure or hardware failure.
- Timeout on request to transmit a message in 255 seconds.
- Power failure.

- Routing update checksum error

  A routing update packet failed its internal integrity test.

- Unexpected packet type

  A packet was received out of the normal protocol sequence. For example, the local node received a normal data packet when it expected a verification packet.

- Verification password required from Phase III node

  A required routing initialization password was not specified before an attempt was made to initialize the Phase III node in a Phase IV network.

- Verification receive timeout

  A required verification packet was not received from the adjacent node within the required response time. Either packets were lost on the line or a failure occurred at the adjacent node.

- Version skew

  The routing version of the adjacent node is unacceptable to the local node. The operator may have installed incorrect software at the adjacent node.

4.8 Circuit down*

An error has occurred for the circuit. This message displays the name of the circuit to which the event applies, along with the following event qualifiers: the packet header (as described for event 4.0), the reason (as described for event 4.7), and the address of the adjacent node.

4.9 Circuit down, operator initiated*

An operator initiated state change has occurred for the circuit. This message displays the name of the circuit to which the event applies, along with the following event qualifiers—the packet header (as described for event 4.0), the reason (as described for event 4.7), and the addresses of the expected node and the adjacent node.

4.10 Circuit up*

A remote node has initialized on one of the physical lines connected to the local node. This message displays the name of the line to which the event applies, along with one event qualifier—the address of the newly initialized node.

Be sure to note that this event does not imply that the node is reachable. Reachability is determined by the higher-level routing algorithms.

4.11 Initialization failure, line fault*

A remote node failed to initialize with the local node because of a physical line error. This message displays the name of the line to which the event applies, along with one event qualifier—the reason for the event (as described for event 4.7).

4.12 Initialization failure*

A remote node failed to initialize with the local node because of a software error. This message displays the name of the line to which the event applies, along with two event qualifiers—the packet header (as described for event 4.0) and the reason (as described for event 4.7).

4.13 Initialization failure, operator initiated*

A remote node failed to initialize with the local node because of an operator error. This message displays the name of the line to which the event applies, along with three event qualifiers—the packet header (as described for event 4.0), the reason (as described for event 4.7), and the version received from the adjacent node.

4.14 Node reachability change*

Because of Routing operation, the reachability of a remote node has changed. This message displays the name of the node to which the event applies, along with one event qualifier—the routing status of the node (reachable or unreachable).

4.15 Adjacency up*

The adjacent node on the circuit is initialized. This message displays the name of the circuit to which the event applies, and one event qualifier—the address of the adjacent node.

4.16 Adjacency rejected*

The adjacent node on the circuit is not initialized. This message displays the name of the circuit to which the event applies, and two event qualifiers—the address of the adjacent node and the reason for the event (as described for event 4.7).

4.17 Area reachability change*

Because of Routing operation, the reachability of an area has changed. This message displays the name of the area to which the event applies, along with one event qualifier—the routing status of the area (reachable or unreachable).

4.18 Adjacency down*

An error has occurred for an adjacency on the circuit. This message displays the name of the circuit to which the event applies, along with the following event qualifiers—the reason (as described for event 4.7), the packet header (as described for event 4.4), and the address of the adjacent node on the circuit.

4.19 Adjacency down, operator initiated*

An adjacency on the circuit is down because of an operator request. This message displays the name of the circuit to which the event applies, along with the following event qualifiers—the reason (as described for event 4.7), the packet header (as described for event 4.0), and the addresses of the expected node and the adjacent node on the circuit.

D.5 Data Link Layer Events

The following specific event classes and types are supported for the Data Link layer. Only those events and entire event classes marked with an asterisk (*) are logged by DECnet for OpenVMS components.

5.0 Locally initiated state change

The line state changed because of an operator command. This message displays the name of the line to which the event applies, along with two event qualifiers—the old DDCMP state (HALTED, ISTRT, ASTRT, RUNNING, or MAINTENANCE) and the new DDCMP state. Refer to the *Digital Data Communications Message Protocol Functional Specification* for a description of these states.

5.1 Remotely initiated state change

A remote user changed the line state. This message displays the name of the line to which the event applies, along with two event qualifiers (see event 5.0).

5.2 Protocol restart received in maintenance mode

The remote node restarted normal operation while the local node had the line in maintenance mode. This message displays the name of the line to which the event applies.

5.3 Send error threshold

Too many data transmission errors occurred. This message displays the name of the line to which the event applies, along with the line counters for that line and one event qualifier—the address of the received station (node).

5.4 Receive error threshold

Too many data reception errors occurred. This message displays the name of the line to which the event applies, along with the line counters for that line and one event qualifier (see event 5.3).

5.5 Select error threshold

Too many selection errors occurred. This message displays the name of the line to which the event applies, along with the line counters for that line and one event qualifier (see event 5.3).

5.6 Block header format error

DDCMP received an invalid block header. This message displays the name of the line to which the event applies, along with one event qualifier—the invalid block header. Refer to the *Digital Data Communications Message Protocol Functional Specification* for a description of the block header format.

5.7 Selection address error

The wrong tributary responded in the polling process. This event occurs only for a multipoint control station when one receives a message that does not match the address of the currently selected tributary.

This message displays the name of the line to which the event applies, along with three event qualifiers—the tributary addresses of the selected tributary, the received tributary, and the previous tributary.

5.8 Streaming tributary

A tributary on the line is impeding the use of that line. This message displays the name of the line to which the event applies, along with two event qualifiers—the tributary address of the received tributary and the status of the tributary. The status may be any of the following:

- Streaming
- Continued send after timeout
- Continued send after deselect
- Ended streaming

5.9 Local buffer too small

A local buffer is too small for receiving a block of data. This message displays the name of the line to which the event applies, along with two qualifiers—the length (in bytes) of the received block and the length (in bytes) of the buffer.

5.13 Line initialization failure

An initialization failure occurred over a broadcast line. This message displays the name of the line to which the event applies.

5.14 Send failure on line

A data transmission attempt failed on a broadcast line. This message displays the name of the line to which the event applies, along with two event qualifiers—the reason for the failure and the distance. Failure reasons can include excessive collisions, short or open circuits, too long a frame, a framing error, an unrecognized frame destination, a remote failure to defer, a block check error, or data overrun.

5.15 Receive failed on line

Data was not received on a broadcast line. This message displays the name of the line to which the event applies, along with two event qualifiers—the reason for the failure (as described in event 5.14) and the header, which includes the source and destination node addresses and the protocol type.

5.16 Collision detect check failed on line

A check for collision detection failed on a broadcast line. This message displays the name of the line to which the event applies.

5.17 DTE up*

A local DTE has synchronized X.25 levels 2 and 3. The DTE address and network name are displayed.

5.18 DTE down*

A local DTE is resynchronizing X.25 levels 2 and 3. The DTE address and network name are displayed.

5.19 Retransmit maximum exceeded*

A frame has been retransmitted the maximum number of times for this line without having received an acknowledgment. This means that either the PSDN or the line to the PSDN has failed.

5.20 FRMR received*

A frame reject was received. This is an error that you cannot correct by retransmitting the same frame again. This is usually due to a protocol violation.

5.21 Illegal frame received*

The frame received was not correctly formatted, or was not the frame expected by the X.25 protocol.

D.6 X.25 Packet Level Events

The following specific event classes and types are supported for X.25 packet level events. Only those events and entire event classes marked with an asterisk (*) are logged by DECnet for OpenVMS components on nodes that have VAX P.S.I. installed.

7.0 DTE state change*

The operator has altered the DTE state to that displayed.

7.1 Illegal packet received*

VAX P.S.I. has received a packet from the PSDN that is invalid in some way. This is usually because of a protocol violation by the PSDN.

7.2 Invalid LCN*

This is used whenever the logical channel number (LCN) in the received packet is invalid. For example: the network has sent a restart packet with a nonzero LCN, a call request packet on an LCN already in use, or a packet on an unused LCN, or tried to use an LCN reserved for a PVC. This is usually because of a protocol violation by the PSDN, but can be caused by not correctly configuring the PVCs in the database.

7.3 Flow control invalid*

Invalid P(S) received or invalid P(R) received. This is because of a protocol error.

7.4 Restart*

Either a restart packet was received or sent (in which case another event may also be logged). This indicates either a protocol error or that the DCE or DTE has reinitialized.

7.5 Clear*

A clear packet with a nonzero cause code was either received or sent because of a protocol error. See the PSDN documentation for the cause of the message and the diagnostic codes displayed.

7.6 Reset*

A reset packet with a nonzero cause code was either received or sent because of a protocol error. See the PSDN documentation for the cause of the message and the diagnostic codes displayed.

7.7 Diagnostic*

A diagnostic packet was either received or sent because of a protocol error. See the PSDN documentation for the cause of the message and the diagnostic codes displayed.

7.8 Reject*

A reject packet was received. VAX P.S.I. does not support receiving reject packets, because PSDNs are not permitted to send them.

7.9 Retransmission maximum exceeded*

The specified control packet has been transmitted the maximum number of times without being acknowledged by the PSDN. Usually this is caused by a failure by the PSDN or the connection to the PSDN, but may be because the DTE is not correctly configured.

7.10 Call failed*

The call (incoming or outgoing) has failed because of one of the following reasons: the application process has exited, the application has rejected the call, no destination has been set up, insufficient resources are available, a call has collided, security checks have failed, or a PSDN or a remote DTE has cleared the call.

7.11 State change*

The operator has initiated a state change, such as shutting down the node. This event is displayed when you shut down VAX P.S.I. (or a part of VAX P.S.I., such as the X.29 server).

D.7 System-Specific Events

The following system-specific event classes and types are supported for OpenVMS operating systems.

128.1 DAP CRC error detected

A CRC error was detected at the DAP level by the file access listener (FAL) or by RMS. The event displays the name of the node to which the connection was established.

128.2 Duplicate PHASE 2 address error

A circuit connection was attempted to a PHASE 2 node that was already reachable. You are allowed only one circuit connection to any PHASE 2 node.

128.3 Process created

A process has been created to handle an incoming X.25 call. The message displays the process name, process ID, and the status returned by the $CREPRC system service.

128.4 Process terminated

A process created to handle an incoming X.25 call has terminated. The message displays the process ID and the exit status of the process.

D.8 Digital Distributed Name Service (DECdns) Clerk Events

The following specific event classes and types are supported for the Digital Distributed Name Service (DECdns) clerk.

353.5 DECdns clerk unable to communicate with server

The DECdns clerk was unable to communicate with a DECdns server. This message displays the name of the clearinghouse where the communication failed, the node on which the DECdns server servicing the clearinghouse exists, and the reason why the communication failed, which may be any of the following:

- Unknown clearinghouse

  The requested clearinghouse is not serviced by the DECdns server that was contacted. This may happen when the cache maintained by the local DECdns clerk contains outdated information for a directory.

- Clearinghouse down

A DECdns server is unable to service a request because the clearinghouse is not operational (stopped state).

- Wrong state

  A DECdns server is unable to service a request because the clearinghouse is currently starting up or shutting down.

- Data corruption

  A DECdns server is unable to service the request because the clearinghouse file has been corrupted.

- No communication

  A network error occurred on the local system or on the system containing the DECdns server. The local OpenVMS error is displayed as a part of this message.

353.20 Local DECdns Advertiser error

This event communicates errors that are local to the DECdns Advertiser. All of these errors have the prefix ADV. The DECdns Advertiser (DNS$ADVER) has encountered an error that prevents proper operation of the process. Exact errors are listed in the system error messages chapter in the *OpenVMS System Messages and Recovery Procedures Reference Manual*.

Index

A

Abbreviating NCP commands, 1–2
Abort, 4–7, 4–8, 4–9, 4–12
Addresses
  physical, 1–9, 1–10
AUTO prefix, C–1

C

Circuit and line devices
  BNA, A–1
  CI, A–1
  DIV, A–1
  DMB, A–1, A–2
  DMC, A–1
  DMF, A–1, A–2
  DMP, A–1
  DPV, A–2
  DSB, A–1, A–2
  DSF, A–1
  DST, A–1
  DSV, A–1, A–2
  DSW, A–1, A–2
  DUP, A–2
  FZA, A–2
  ISA, A–1
  KFE, A–1
  KMV, A–2
  KMX, A–2
  KMY, A–2
  MFA, A–2
  MNA, A–1
  MXE, A–1
  QNA, A–1
  SVA, A–1
  TT, A–1
  TX, A–1
  UNA, A–1
Circuits
  clearing counters, 3–159
  counters, 3–159, B–1, B–4
  devices, A–1
  information, 3–120
  states, C–1 to C–4
  testing, 3–31, 3–49, 3–111
  transitions, C–1 to C–4
CLEAR CIRCUIT command, 3–2
CLEAR EXECUTOR command, 3–6
CLEAR EXECUTOR NODE command, 3–11
CLEAR LINE command, 3–12
CLEAR LOGGING command, 3–16
CLEAR LOGGING EVENTS command, 3–17
CLEAR LOGGING NAME command, 3–19
CLEAR MODULE X25-ACCESS command, 3–21
CLEAR MODULE X25-PROTOCOL command, 3–22
CLEAR MODULE X25-SERVER command, 3–25
CLEAR MODULE X29-SERVER command, 3–25
CLEAR NODE CIRCUIT command, 3–31
CLEAR NODE command, 3–27
CLEAR OBJECT command, 3–32
Command prompting
  See Prompt syntax
Command syntax
  See Syntax
Component names
  using wildcards with, 1–3
CONNECT command, 4–4
Connection tests, 4–4
  invoking, 4–4
  types of, 4–4
CONNECT NODE command, 3–34
CONNECT VIA command, 3–35
Consoles
  connecting to host node, 3–34
  connecting to host node on a specified circuit, 3–35
COPY KNOWN NODES command, 3–36
Counters
  circuit, 3–120, B–1, B–4
  line, 3–161, B–7
  node, 3–160, 3–162, 3–164, B–11
  summary of network, B–1
  X.25 protocol module, B–13
  X.25 server module, B–14

D

Databases
  circuit
    clearing permanent values, 3–2
    creating volatile or permanent parameters, 3–57

Databases
circuit (cont'd)
loading volatile area with all permanent parameters, 3–56
resetting volatile parameters to default, 3–2
setting volatile or permanent parameters, 3–57
clearing selected permanent parameters, 3–6
copying remote node data, 3–36
creating volatile or permanent parameters, 3–64
Ethernet configurator module
clearing permanent values of a specified circuit, 3–20
creating volatile or permanent parameters, 3–90
executor node
clearing volatile or permanent loop node and circuit values, 3–111
creating volatile or permanent network control parameters, 3–64
loading volatile area with all permanent parameters, 3–63, 3–85
setting volatile or permanent network control values, 3–64
host node
accessing X.25 target node parameters, 3–21
line
clear and load volatile parameters with permanent data, 3–76
clearing all values, 3–12
creating parameters, 3–77
resetting volatile parameters to default, 3–12
setting parameters, 3–77
logging
clearing event-source values, 3–17
clearing sink-name, 3–19
clearing volatile data, 3–16
creating event-sink parameters, 3–88
creating event-source parameters, 3–86
resetting event-source values, 3–17
resetting volatile data, 3–16
setting event-sink parameters, 3–88
setting event-source parameters, 3–86
loop node
clearing volatile or permanent values on executor node, 3–31
creating volatile or permanent parameters, 3–111
setting volatile or permanent values, 3–111
node
clearing volatile or permanent values, 3–27
creating volatile or permanent parameters, 3–106
Databases
node (cont'd)
setting volatile or permanent values, 3–106
object
creating volatile or permanent parameters, 3–113
loading volatile area with all permanent parameters, 3–112
setting volatile or permanent values, 3–113
proxy
clear and load volatile area with all permanent parameters, 3–75
resetting selected volatile parameters to default, 3–6
setting volatile or permanent parameters, 3–64
updating all volatile or permanent data, 3–36
X.25 network connector-node access
clearing volatile or permanent names and parameters, 3–21
for host node, 3–21, 3–92
X.25 or X.29 server module
clearing volatile or permanent call handler values, 3–25
creating volatile or permanent call handler parameters, 3–100
resetting volatile or permanent call handler values, 3–25
setting volatile or permanent call handler values, 3–100
X.25 protocol module
clearing volatile or permanent values, 3–22
creating volatile or permanent parameters, 3–94
resetting volatile parameters to default, 3–22
setting volatile or permanent values, 3–94
Databases on local node
displaying
See Displaying network parameters
DATA command, 4–6
Data integrity checking, 4–7
Data Link layer
events, D–9
Data message length, 4–6, 4–11
Data tests, 4–6
DECdns
event messages, D–12
events, D–12
IDP, 3–7
interface, 3–7, 3–66
namespace, 3–7, 3–66
DECnet
circuit and line devices, A–1
devices supported by, A–1
event class and type summary, D–1 to D–13

DECnet (cont'd)
using the Network Control Program to control, 1–1, 2–1
DECnet for OpenVMS
See DECnet
DECnet Test Sender/DECnet Test Receiver utility (DTS/DTR), 4–1
abort, 4–7, 4–8, 4–9, 4–12
commands, 4–3 to 4–12
abbreviating, 4–1
qualifiers, 4–1
syntax, 4–1
using in a batch job, 4–1
directing output, 4–3
invoking, 4–2
printing, 4–3
testing network, 4–4 to 4–12
tests
connection, 4–4
data, 4–6
disconnection, 4–8
exiting, 4–3
interrupt, 4–11
subtests, 4–2
terminating, 4–3
using, 4–3
DEFINE CIRCUIT command, 3–57
DEFINE EXECUTOR command, 3–64
DEFINE LINE command, 3–77
DEFINE LOGGING EVENTS command, 3–86
DEFINE LOGGING STATE command, 3–88
DEFINE MODULE CONFIGURATOR command, 3–90
DEFINE MODULE X25-ACCESS command, 3–92
DEFINE MODULE X25-PROTOCOL command, 3–94
DEFINE MODULE X25-SERVER command, 3–100
DEFINE MODULE X29-SERVER command, 3–100
DEFINE NODE CIRCUIT command, 3–111
DEFINE NODE command, 3–106
DEFINE OBJECT command, 3–113
Directing output, 2–1
DISCONNECT command, 4–8
Disconnection tests, 4–8
types of, 4–9
DISCONNECT LINK command, 3–40
Displaying network parameters
circuit information, 3–120
Ethernet configurator data, 3–136
line information, 3–128
local node information, 3–125
logging information, 3–133
network names and parameters from the X.25 access module, 3–139
node information, 3–147
object information, 3–152
Displaying network parameters (cont'd)
volatile database information, 3–118
volatile link information, 3–131
X.25 and X.29 server parameters for call handler, 3–145
X.25 protocol module parameters, 3–141
DNS
See DECdns
DTE (data terminal equipment)
states and transitions, C–5
state transitions, C–6
DTE (Data Terminal Equipment)
database parameters
See Database on local node
DTR
See DECnet Test Sender/DECnet Test Receiver utility (DTS/DTR)
DTRECV (DECnet Test Receiver image), 4–1
DTS
See DECnet Test Sender/DECnet Test Receiver utility (DTS/DTR)
DTS/DTR utility
See DECnet Test Sender/DECnet Test Receiver utility (DTS/DTR)
DTSEND (DECnet Test Sender image), 4–1
DTSEND command, 4–2
Duration tests, 4–7, 4–12

E

Echo tests, 4–7, 4–12
End Communications layer events, D–4
Entering NCP commands, 1–2
Event class and type summary, D–1 to D–13
Event lists, 1–9
Events
Data Link layer, D–9
DNS (Distributed Name Service), D–12
End Communications layer, D–4
Network Management layer, D–2
OpenVMS specific, D–12
Routing layer, D–4
Session Control layer, D–4
X.25 Packet level, D–11
Executor nodes
clearing counters, 3–160
commands
prefix identifier, 3–154
management listener (NML), 3–73
setting as the default command point, 3–73
testing, 3–51
Exiting NCP, 2–1
External clock, 3–77

F

FDDI (Fiber Distributed Data Interface), 3–81
  ECHO DATA, 3–12, 3–78
  ECHO LENGTH, 3–12, 3–78
  ECHO TARGET, 3–13, 3–78
  NIF TARGET, 3–13, 3–80
  REQUESTED TRT, 3–13, 3–81
  RESTRICTED TOKEN TIMEOUT, 3–13, 3–81
  RING PURGER ENABLE, 3–13, 3–82
  SIF CONFIGURATION TARGET, 3–14, 3–82
  SIF OPERATION TARGET, 3–14, 3–82
  VALID TRANSMISSION TIME, 3–14, 3–83

H

h-addresses
  See Physical addresses
Hardware addresses
  See Physical addresses
HELP command, 3–42
Help facility, 1–12
Hex password, 1–9

I

Identifiers
  downline load, 3–45, 3–48
IDP string parameter, 3–66
Internal clock, 3–77
INTERRUPT command, 4–11
Interrupt tests, 4–11
  types of, 4–12
Invoking NCP, 1–1

K

KMS/KMV DUMP Analyzer
  See PSIKDA

L

Line counters summary, B–7
Lines
  clearing counters, 3–161
  counters, 3–161, B–7
  devices, A–1
  states, C–1 to C–4
  transitions, C–1 to C–4
  X.25 testing, 3–52
Line speeds
  setting, 4–6, 4–11
Links
  disconnecting, 3–40
  disconnecting logical from local node, 3–40
  states, C–5
LIST CIRCUIT command, 3–120
LIST EXECUTOR command, 3–125
LIST LINE command, 3–128
LIST LOGGING command, 3–133
LIST MODULE CONFIGURATOR command, 3–136
LIST MODULE X25-ACCESS command, 3–139
LIST MODULE X25-PROTOCOL command, 3–141
LIST MODULE X25-SERVER command, 3–145
LIST MODULE X29-SERVER command, 3–145
LIST NODE command, 3–147
LIST OBJECT command, 3–152
LOAD NODE command, 3–44
LOAD VIA command, 3–47
Loopback mirrors
  See MIRRORs
LOOP CIRCUIT command, 3–49
Loop circuit for testing, 3–49
LOOP EXECUTOR command, 3–51
LOOP LINE command, 3–52
LOOP NODE command, 3–54
Loop nodes, 3–54

M

MIRRORs (loopback mirrors), 3–54
Multiple buffering, 4–1

N

NCP (Network Control Program)
  usage summary, 2–1
NCP commands
  continuing, 1–2
  entering, 1–2
  recalling, 1–2
Network counter summary, B–1 to B–15
Network management
  layer events, D–2
Networks
  circuit devices, A–1
  line devices, A–1
  parameters
    See Database on local node
  testing, 3–49, 3–51, 3–52, 3–54, 4–1
Node counters, 3–160, 3–164, B–11
  summary, B–11
Nodes
  address, 4–8, 4–11
  clearing all counters, 3–160, 3–164
  name, 4–8, 4–11
  remote, 4–8
Nonprivileged DECnet accounts, 4–8

O

Object code values, 3–116
Objects
  code values, 3–115
  parameters, 3–112, 3–113, 3–152
OpenVMS systems
  events specific to, D–12
Optional user data, 4–4, 4–8

P

P.S.I. DTEs
  See also X.25 server modules
  states and substates, C–5
  state transitions, C–6
P.S.I. server modules
  states, C–8
  state transitions, C–8
p-addresses
  See Physical addresses
Pattern tests, 4–7, 4–12
Physical addresses, 1–9, 1–10
PIPELINE QUOTA parameter, 3–69
Pipeline quotas, 3–69
Printing
  test results, 4–4, 4–6, 4–8, 4–11
Prompting sequence
  examples, 1–6 to 1–7
  exiting, 1–5
Prompt syntax, 1–4
Protocol module counters, B–13
Proxies
  parameters, 3–75
PURGE CIRCUIT command, 3–2
PURGE EXECUTOR command, 3–6
PURGE LINE command, 3–12
PURGE LOGGING command, 3–16
PURGE LOGGING EVENTS command, 3–17
PURGE LOGGING NAME command, 3–19
PURGE MODULE CONFIGURATOR command, 3–20
PURGE MODULE X25-ACCESS command, 3–21
PURGE MODULE X25-PROTOCOL command, 3–22
PURGE MODULE X25-SERVER command, 3–25
PURGE MODULE X29-SERVER command, 3–25
PURGE NODE CIRCUIT command, 3–31
PURGE NODE command, 3–27
PURGE OBJECT command, 3–32

Q

Quotas
  pipeline, 3–69

R

Recalling NCP commands, 1–2
Routing layer events, D–4

S

Sequence tests, 4–7, 4–12
Server module counters, 3–163, B–14
Session Control layer events, D–4
SET CIRCUIT ALL command, 3–56
SET CIRCUIT command, 3–57
SET EXECUTOR ALL command, 3–63
SET EXECUTOR command, 3–64
SET EXECUTOR NODE command, 3–73
SET KNOWN PROXIES ALL command, 3–75
SET LINE ALL command, 3–76
SET LINE command, 3–77
SET LOGGING ALL command, 3–85
SET LOGGING EVENTS command, 3–86
SET LOGGING STATE command, 3–88
SET MODULE CONFIGURATOR command, 3–90
SET MODULE X25-ACCESS command, 3–92
SET MODULE X25-PROTOCOL command, 3–94
SET MODULE X25-SERVER command, 3–100
SET MODULE X29-SERVER command, 3–100
SET NODE ALL command, 3–105
SET NODE CIRCUIT command, 3–111
SET NODE command, 3–106
SET OBJECT ALL command, 3–112
SET OBJECT command, 3–113
SHOW AREA command, 3–118
SHOW CIRCUIT command, 3–120
SHOW EXECUTOR command, 3–125
SHOW LINE command, 3–128
SHOW LINKS command, 3–131
SHOW LOGGING command, 3–133
SHOW MODULE CONFIGURATOR command, 3–136
SHOW MODULE X25-ACCESS command, 3–139
SHOW MODULE X25-PROTOCOL command, 3–141
SHOW MODULE X25-SERVER command, 3–145
SHOW MODULE X29-SERVER command, 3–145
SHOW NODE command, 3–147
SHOW OBJECT command, 3–152
Sink tests, 4–7, 4–11, 4–12
Software transmissions
  loading downline to target node, 3–44
  loading downline to target node on specified circuit, 3–47

States
substates, C–1
transitions, C–2
Statistics
printing, 4–6, 4–11
reporting, 4–4, 4–6, 4–11
Syntax
for event list identification, 1–9
of NCP commands, 1–2
rules for naming NCP components and parameters, 1–7 to 1–11

T

Target nodes
starting bootstrap and self-load program, 3–156
through a specified circuit, 3–158
testing, 3–54
TELL command, 3–154
Testing
the network, 4–1
Tests
See DECnet Test Sender/DECnet Test Receiver utility (DTS/DTR) tests
TRIGGER NODE command, 3–156
TRIGGER VIA command, 3–158

U

User data
See Optional user data

V

VAX P.S.I.
See P.S.I.

W

Wildcard characters
using with NCP, 1–3

X

X.25 networks
accessing with specified multihost connector, 3–92
clearing DTE counters, 3–162
testing lines, 3–52
X.25 packet level events, D–11
X.25 protocol modules
counters, 3–162, B–13
parameters, 3–22, 3–94, 3–141
X.25 server modules
See also P.S.I.
clearing call handler counters, 3–163
counters, 3–163, B–14
X.25 server modules (cont'd)
parameters, 3–25, 3–100, 3–145
states, C–8
state transitions, C–8
X.29 server modules
See X.25 server modules

Z

ZERO CIRCUITS command, 3–159
ZERO EXECUTOR command, 3–160
ZERO LINE command, 3–161
ZERO MODULE X25-PROTOCOL command, 3–162
ZERO MODULE X25-SERVER command, 3–163
ZERO MODULE X29-SERVER command, 3–163
ZERO NODE command, 3–164